

Horizon 7 Integration

MAR 2020

VMware Horizon 7 7.12



vmware®

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

<https://docs.vmware.com/>

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

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2016-2020 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

| | |
|---|-----------|
| Horizon 7 Integration | 5 |
| 1 Introduction to Horizon 7 Integration | 6 |
| Horizon 7 Components | 6 |
| Integration Interfaces to Horizon 7 | 7 |
| 2 Integrating Horizon 7 with the Event Database | 8 |
| Event Database Tables and Schemas | 8 |
| Horizon Connection Server Events | 11 |
| Horizon Agent Events | 17 |
| Horizon Administrator Events | 17 |
| Event Message Attributes | 26 |
| Sample Database Queries and Views | 27 |
| 3 Enabling Horizon 7 for Cloud-Hosted Subscriptions and Services | 30 |
| 4 Deploying Horizon 7 on VMware Cloud on AWS | 32 |
| 5 Customizing LDAP Data | 33 |
| Introduction to LDAP Configuration Data | 33 |
| Modifying LDAP Configuration Data | 34 |
| Export LDAP Configuration Data | 34 |
| Defining a Desktop Pool in an LDIF Configuration File | 35 |
| Import LDAP Configuration Data | 38 |
| 6 Examining PCoIP Session Statistics with WMI | 40 |
| Using PCoIP Session Statistics | 40 |
| General PCoIP Session Statistics | 41 |
| PCoIP Audio Statistics | 42 |
| PCoIP Imaging Statistics | 43 |
| PCoIP Network Statistics | 43 |
| PCoIP USB Statistics | 45 |
| Examples of Using PowerShell cmdlets to Examine PCoIP Statistics | 46 |
| 7 Setting Desktop Policies with Start Session Scripts | 47 |
| Obtaining Input Data for a Start Session Script | 47 |
| Best Practices for Using Start Session Scripts | 47 |
| Preparing a Horizon 7 Desktop to Use a Start Session Script | 48 |

| | |
|---|----|
| Enable the VMware View Script Host Service | 49 |
| Add Windows Registry Entries for a Start Session Script | 49 |
| Sample Start Session Scripts | 51 |

8 Using the Horizon PowerCLI Module 52

| | |
|--------------------------------------|----|
| Set Up the Horizon PowerCLI Module | 52 |
| Run Example Horizon PowerCLI Scripts | 53 |

Horizon 7 Integration

The *Horizon 7 Integration* document describes how to integrate Horizon 7™ software with third-party software such as Windows PowerShell and business intelligence reporting engines.

Intended Audience

This document is intended for anyone who wants to customize or integrate software to work with Horizon 7. The information in this document is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology and datacenter operations.

Introduction to Horizon 7 Integration

1

With Horizon 7, system administrators can provision desktops and control user access to these desktops. Client software connects users to virtual machines running in VMware vSphere™, or to physical systems running within your network environment. In addition, Horizon 7 administrators can configure Remote Desktop Services (RDS) hosts to provide Horizon 7 desktop and application sessions to client devices.

This chapter includes the following topics:

- [Horizon 7 Components](#)
- [Integration Interfaces to Horizon 7](#)

Horizon 7 Components

You can use Horizon 7 with VMware vCenter Server to create desktops from virtual machines that are running on VMware ESX® or VMware ESXi™ hosts and deploy these desktops to end users. You can also install Horizon 7 on RDS hosts to deploy desktops and applications to end users. Horizon 7 uses your existing Active Directory infrastructure for user authentication and management.

After you create a desktop or application, authorized end users can use Web-based or locally installed client software to securely connect to centralized virtual machines, back-end physical systems, or RDS hosts.

Horizon 7 consists of the following major components.

| | |
|----------------------------------|--|
| Horizon Connection Server | A software service that acts as a broker for client connections by authenticating and then directing incoming user requests to the appropriate virtual machine, physical system, or RDS host. |
| Horizon Agent | A software service that is installed on all guest virtual machines, physical systems, or RDS hosts to allow them to be managed by Horizon 7. Horizon Agent provides features such as connection monitoring, virtual printing, USB support, and single sign-on. |
| Horizon Client | A software application that communicates with Connection Server to enable users to connect to their desktops. |

| | |
|------------------------------|--|
| Horizon Administrator | A Web application that enables Horizon 7 administrators to configure Connection Server, deploy desktop and application pools, manage machines, control user authentication, initiate and examine system events, and perform analytical activities. |
| vCenter Server | A server that acts as a central administrator for ESX/ESXi hosts that are connected on a network. A vCenter Server instance provides the central point for configuring, provisioning, and managing virtual machines in the datacenter. |
| View Composer | A software service that is installed on a vCenter Server instance to enable Horizon 7 to rapidly deploy multiple linked-clone desktops from a single centralized base image. |

Integration Interfaces to Horizon 7

You can use several interfaces to integrate Horizon 7 with external applications.

| | |
|---|--|
| Event database | You can configure Horizon 7 to record events to a Microsoft SQL Server or Oracle database. You can then use business intelligence reporting engines to access and analyze this database. |
| Lightweight Directory Access Protocol (LDAP) | You can export and import LDAP configuration data from and into Horizon 7. You can create scripts that update this configuration data without accessing Horizon Administrator directly. |
| Windows Management Instrumentation (WMI) | You can examine performance statistics for a PCoIP session. |

Integrating Horizon 7 with the Event Database

2

You can configure Horizon 7 to record events to a Microsoft SQL Server or Oracle database. Horizon 7 records events such as end-user actions, administrator actions, alerts that report system failures and errors, and statistical sampling.

End-user actions include logging and starting desktop and application sessions. Administrator actions include adding entitlements and creating desktop and application pools. An example of statistical sampling is recording the maximum number of users over a 24-hour period.

You can use business intelligence reporting engines such as Crystal Reports, IBM Cognos, MicroStrategy 9, and Oracle Enterprise Performance Management System to access and analyze the event database.

This chapter includes the following topics:

- [Event Database Tables and Schemas](#)
- [Horizon Connection Server Events](#)
- [Horizon Agent Events](#)
- [Horizon Administrator Events](#)
- [Event Message Attributes](#)
- [Sample Database Queries and Views](#)

Event Database Tables and Schemas

Horizon 7 uses database tables to implement the event database. The event database prepends the names of these tables with a prefix that you define when you set up the database.

Event Database Tables

The following table shows the database tables that implement the event database in Horizon 7.

Table 2-1. Event Database Tables

| Table Name | Description |
|------------|--|
| event | Metadata and search optimization data for recent events. |
| event_data | Data values for recent events. |

Table 2-1. Event Database Tables (continued)

| Table Name | Description |
|-----------------------|---|
| event_data_historical | Data values for all events. |
| event_historical | Metadata and search optimization data for all events. |

Horizon 7 records details about events to all the database tables. After a certain period of time has elapsed since writing an event record, Horizon 7 deletes the record from the event and event_data tables. You can use Horizon Administrator to configure the time period for which the database keeps a record in the event and event_data tables.

Important Horizon 7 does not restrict the growth of the event_historical and event_data_historical tables. You must implement a space management policy for these tables.

A unique primary key, EventID, identifies each event that Horizon 7 records in the event and event_historical tables. Horizon 7 records data values for each event in the event_data and event_data_historical tables. You can obtain the complete set of information for an event by joining the event and event_data tables or the event_historical and event_data_historical tables on the EventID column.

The EventType, Severity, and Time columns in the event and event_historical tables identify the type and severity of an event and the time at which it occurred.

For information about setting up the event database, see the *Horizon 7 Installation* document.

Note To purge data from the historical tables, see <http://kb.vmware.com/kb/2150309>.

Event Database Schemas

The following table shows the schema for the event and event_historical database tables.

Table 2-2. Schema for the event and event_historical Tables

| Column Name | Oracle Data Type | SQL Server Data Type | Description |
|--------------|------------------|----------------------|--|
| Acknowledged | SMALLINT | tinyint | Whether Horizon 7 acknowledged the event. ■ 0 = false ■ 1 = true |
| DesktopID | NVARCHAR2(512) | nvarchar(512) | Desktop ID of the associated pool. |
| EventID | INTEGER | int | Unique primary key for the event. |
| EventType | NVARCHAR2(512) | nvarchar(512) | Event name that corresponds to an item in the message catalog. For example, BROKER_USERLOGGEDIN. |
| FolderPath | NVARCHAR2(512) | nvarchar(512) | Full path of the folder that contains the associated object. |
| GroupID | NVARCHAR2(512) | nvarchar(512) | SID of the associated group in Active Directory. |

Table 2-2. Schema for the event and event_historical Tables (continued)

| Column Name | Oracle Data Type | SQL Server Data Type | Description |
|--------------------|------------------|----------------------|---|
| LUNId | NVARCHAR2(512) | nvarchar(512) | ID of the LUN that stores the associated object. |
| Machineld | NVARCHAR2(512) | nvarchar(512) | ID of the associated physical or virtual machine. |
| Module | NVARCHAR2(512) | nvarchar(512) | Horizon 7 component that raised the event. For example, Admin, Broker, Tunnel, Framework, Client, or Agent. |
| ModuleAndEventText | NVARCHAR2(512) | nvarchar(512) | Event message with values substituted for attribute parameters. |
| Node | NVARCHAR2(512) | nvarchar(512) | Name of the virtual device node. |
| Severity | NVARCHAR2(512) | nvarchar(512) | Severity level. For example, INFO, WARNING, ERROR, AUDIT_SUCCESS, AUDIT_FAIL. |
| Source | NVARCHAR2(512) | nvarchar(512) | Identifier for the source of the event. |
| ThinAppld | NVARCHAR2(512) | nvarchar(512) | ID of the associated ThinApp™ object. |
| Time | TIMESTAMP | datetime | Time at which the event occurred, measured from the epoch (January 1, 1970). |
| UserDiskPathId | NVARCHAR2(512) | nvarchar(512) | ID of the user disk. |
| UserSID | NVARCHAR2(512) | nvarchar(512) | SID of the associated user in Active Directory. |

The following table shows the schema for the event_data and event_data_historical database tables.

Table 2-3. Schema for the event_data and event_data_historical Tables

| Column Name | Oracle Data Type | SQL Server Data Type | Description |
|--------------|------------------|----------------------|--|
| BooleanValue | SMALLINT | tinyint | Value of a Boolean attribute. ■ 0 = false ■ 1 = true |
| EventID | INTEGER | int | Unique primary key for the event. |
| IntValue | INTEGER | int | Value of an integer attribute. |
| Name | NVARCHAR2(512) | nvarchar(512) | Attribute name (for example, UserDisplayName). |
| StrValue | NVARCHAR2(512) | nvarchar(512) | Value of a string attribute. For other types of attributes, this column contains an interpretation of the data type as a string. |

Table 2-3. Schema for the event_data and event_data_historical Tables (continued)

| Column Name | Oracle Data Type | SQL Server Data Type | Description |
|-------------|------------------|----------------------|--|
| TimeValue | TIMESTAMP | datetime | Value of a date and time attribute. |
| Type | SMALLINT | tinyint | <p>The data type of the attribute.</p> <ul style="list-style-type: none"> ■ 0 = StrValue ■ 1 = IntValue ■ 2 = TimeValue ■ 3 = BooleanValue |

Horizon Connection Server Events

Horizon Connection Server events report Connection Server-related information, such as desktop and application sessions, user authentication failures, and provisioning errors.

The BROKER_DAILY_MAX_DESKTOP_SESSIONS event reports the maximum number of concurrent desktop sessions over a 24-hour period. If a user runs multiple desktop sessions concurrently, each desktop session is counted separately.

The BROKER_DAILY_MAX_APP_USERS event reports the maximum number of concurrent application users over a 24-hour period. If a user runs multiple applications concurrently, the user is counted only once. Short-lived sessions might not be included in the count because the sampling is performed every five minutes.

The BROKER_VC_DISABLED and BROKER_VC_ENABLED events report the state of the vCenter driver that Horizon 7 uses to track a vCenter Server instance.

The BROKER_VC_STATUS_* events report the state of a vCenter Server instance.

The following table lists all the event types for Connection Server.

Table 2-4. Connection Server Events

| Event Type | Severity | ModuleAndEventText |
|-----------------------------------|------------|---|
| BROKER_AGENT_OFFLINE | WARNING | The agent running on machine \${MachineName} has not responded to queries, marking it as offline |
| BROKER_AGENT_ONLINE | WARNING | The agent running on machine \${MachineName} is responding again, but did not send a startup message |
| BROKER_APPLICATION_LAUNCH_FAILURE | ERROR | Unable to launch from Pool \${PoolId} for user \${UserDisplayName}: The broker encountered an error while processing the request, please contact support for assistance |
| BROKER_APPLICATION_MISSING | WARNING | At least \${ApplicationMissingCount} applications, including \${ApplicationExecutable}, are not installed on \${MachineName} in Pool \${PoolId} |
| BROKER_APPLICATION_NOT_ENTITLED | AUDIT_FAIL | Unable to launch from Pool \${PoolId} for user \${UserDisplayName}: User is not entitled to this Pool |

Table 2-4. Connection Server Events (continued)

| Event Type | Severity | ModuleAndEventText |
|---|------------|--|
| BROKER_APPLICATION_PROTOCOL_NOT_SUPPORTED | AUDIT_FAIL | Unable to launch from Pool \${PoolId} for user \${UserDisplayName}: Requested protocol \${ProtocolId} is not supported |
| BROKER_APPLICATION_REQUEST | INFO | User \${UserDisplayName} requested Application \${ApplicationId} |
| BROKER_APPLICATION_SESSION_REQUEST | INFO | User \${UserDisplayName} requested an application session from Pool \${PoolId} |
| BROKER_DAILY_MAX_DESKTOP_SESSIONS | INFO | \$(Time): Over the past 24 hours, the maximum number of concurrent desktop sessions was \${UserCount} |
| BROKER_DAILY_MAX_APP_USERS | INFO | \$(Time): Over the past 24 hours, the maximum number of users with concurrent application sessions was \${UserCount} |
| BROKER_DESKTOP_LAUNCH_FAILURE | ERROR | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: The broker encountered an error while processing the request, please contact support for assistance |
| BROKER_DESKTOP_NOT_ENTITLED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: User is not entitled to this Pool |
| BROKER_DESKTOP_PROTOCOL_NOT_SUPPORTED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Requested protocol \${ProtocolId} is not supported |
| BROKER_DESKTOP_REQUEST | INFO | User \${UserDisplayName} requested Pool \${DesktopId} |
| BROKER_EVENT_HANDLING_STARTED | INFO | Broker \${BrokerName} has started handling events |
| BROKER_EVENT_HANDLING_STOPPED | INFO | \$(BrokerName) has stopped handling events |
| BROKER_MACHINE_ALLOCATED | INFO | User \${UserDisplayName} requested Pool \${DesktopId}, allocated machine \${MachineName} |
| BROKER_MACHINE_ASSIGNED_UNAVAILABLE | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Assigned machine \${MachineName} is unavailable |
| BROKER_MACHINE_CANNOT_CONNECT | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Failed to connect to Machine \${MachineName} using \${ProtocolId} |
| BROKER_MACHINE_CONFIGURED_VIDEO_SETTINGS | INFO | Successfully configured video settings for Machine VM \${MachineName} in Pool \${DesktopId} |
| BROKER_MACHINE_NOT_READY | WARNING | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} is not ready to accept connections |
| BROKER_MACHINE_OPERATION_DELETED | INFO | machine \${MachineName} has been deleted |
| BROKER_MACHINE_PROTOCOL_NOT_SUPPORTED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} does not support protocol \${ProtocolId} |

Table 2-4. Connection Server Events (continued)

| Event Type | Severity | ModuleAndEventText |
|--|------------|--|
| BROKER_MACHINE_PROTOCOL_UNAVAILABLE | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} did not report protocol \${ProtocolId} as ready |
| BROKER_MACHINE_REJECTED_SESSION | WARNING | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} rejected the start session request |
| BROKER_MACHINE_SESSION_TIMEDOUT | WARNING | Session for user \${UserDisplayName} timed out |
| BROKER_MULTIPLE_DESKTOPS_FOR_KIOSK_USER | WARNING | User \${UserDisplayName} is entitled to multiple desktop pools |
| BROKER_POOL_CANNOT_ASSIGN | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There are no machines available to assign the user to |
| BROKER_POOL_COMANAGER_REQUIRED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No co-management availability for protocol \${ProtocolId} |
| BROKER_POOL_EMPTY | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: The Desktop Pool is empty |
| BROKER_POOL_NO_MACHINE_ASSIGNED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No machine assigned to this user |
| BROKER_POOL_NO_RESPONSES | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No machines in the Desktop Pool are responsive |
| BROKER_POOL_OVERLOADED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: All responding machines are currently in use |
| BROKER_POOL_POLICY_VIOLATION | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: This Desktop Pool does not allow online sessions |
| BROKER_POOL_PROTOCOL_NOT_SUPPORTED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There were no machines available that support protocol \${ProtocolId} |
| BROKER_POOL_PROTOCOL_UNAVAILABLE | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There were no machines available that reported protocol \${ProtocolId} as ready |
| BROKER_POOL_TUNNEL_NOT_SUPPORTED | AUDIT_FAIL | Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Tunneling is not supported for protocol \${ProtocolId} |
| BROKER_PROVISIONING_ERROR_CONFIG_CLEARED | INFO | The previously reported configuration problem is no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_CONFIG_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because of a configuration problem |
| BROKER_PROVISIONING_ERROR_DISK_CLEARED | INFO | The previously reported disk problem is no longer present on Pool \${DesktopId} |

Table 2-4. Connection Server Events (continued)

| Event Type | Severity | ModuleAndEventText |
|--|----------|--|
| BROKER_PROVISIONING_ERROR_DISK_LC_RESERVATION_CLEARED | INFO | The previously reported error due to available free disk space reserved for linked clones is no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_DISK_LC_RESERVATION_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because available free disk space is reserved for linked clones |
| BROKER_PROVISIONING_ERROR_DISK_SE T | WARNING | Provisioning error occurred on Pool \${DesktopId} because of a disk problem |
| BROKER_PROVISIONING_ERROR_LICENCE_CLEARED | INFO | The previously reported licensing problem is no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_LICENCE_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because of a licensing problem |
| BROKER_PROVISIONING_ERROR_NETWORKING_CLEARED | INFO | The previously reported networking problems with Horizon Agent are no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_NETWORKING_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because of a networking problem with Horizon Agent |
| BROKER_PROVISIONING_ERROR_RESOURCE_CLEARED | INFO | The previously reported resource problem is no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_RESOURCE_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because of a resource problem |
| BROKER_PROVISIONING_ERROR_TIMEOUT_CUSTOMIZATION_CLEARED | INFO | The previously reported timeout while customizing is no longer present on Pool \${DesktopId} |
| BROKER_PROVISIONING_ERROR_TIMEOUT_CUSTOMIZATION_SET | ERROR | Provisioning error occurred on Pool \${DesktopId} because of a timeout while customizing |
| BROKER_PROVISIONING_ERROR_VM_CLONING | ERROR | Provisioning error occurred for Machine \${MachineName}: Cloning failed for Machine |
| BROKER_PROVISIONING_ERROR_VM_CUSTOMIZATION_ERROR | ERROR | Provisioning error occurred for Machine \${MachineName}: Customization failed for Machine |
| BROKER_PROVISIONING_ERROR_VM_CUSTOMIZATION_NETWORKING | ERROR | Provisioning error occurred for Machine \${MachineName}: Customization error due to no network communication between Horizon Agent and Connection Server |
| BROKER_PROVISIONING_ERROR_VM_CUSTOMIZATION_TIMEOUT | ERROR | Provisioning error occurred for Machine \${MachineName}: Customization operation timed out |
| BROKER_PROVISIONING_SVI_ERROR_COMPOSER_AGENT_INIT_FAILED | ERROR | Provisioning error occurred for Machine \${MachineName}: View Composer agent initialization failed |
| BROKER_PROVISIONING_SVI_ERROR_RECONFIG_FAILED | ERROR | Provisioning error occurred for Machine \${MachineName}: Reconfigure operation failed |
| BROKER_PROVISIONING_SVI_ERROR_REFIT_FAILED | ERROR | Provisioning error occurred for Machine \${MachineName}: Refit operation \${SVIOperation} failed |

Table 2-4. Connection Server Events (continued)

| Event Type | Severity | ModuleAndEventText |
|---|---------------|--|
| BROKER_PROVISIONING_SVI_ERROR_REMOVING_VM | ERROR | Provisioning error occurred for Machine \${MachineName}: Unable to remove Machine from inventory |
| BROKER_PROVISIONING_VERIFICATION_FAILED_USER_ASSIGNED | WARNING | Provisioning verification failed for Machine \${MachineName}: User is already assigned to a machine in Pool \${DesktopId} |
| BROKER_PROVISIONING_VERIFICATION_FAILED_USER_CANNOT_BE_ASSIGNED | WARNING | Provisioning verification failed for Machine \${MachineName}: A user cannot be assigned because Pool \${DesktopId} is not persistent |
| BROKER_PROVISIONING_VERIFICATION_FAILED_VMNAME_IN_USE | WARNING | Provisioning verification failed for Machine \${MachineName}: A machine already exists in Pool \${DesktopId} with name \${MachineName} |
| BROKER_SECURITY_SERVER_ADD_FAILED | AUDIT_FAIL | Failed to add security server \${SecurityServerId} |
| BROKER_SECURITY_SERVER_ADD_FAILED_PASSWORD_EXPIRED | AUDIT_FAIL | Failed to add security server \${SecurityServerId}, pairing password expired |
| BROKER_SECURITY_SERVER_ADD_FAILED_PASSWORD_INCORRECT | AUDIT_FAIL | Failed to add security server \${SecurityServerId}, pairing password incorrect |
| BROKER_SECURITY_SERVER_ADD_FAILED_PASSWORD_NOT_SET | AUDIT_FAIL | Failed to add security server \${SecurityServerId}, pairing password not set |
| BROKER_SECURITY_SERVER_ADDED | AUDIT_SUCCESS | Security server \${SecurityServerId} added |
| BROKER_SVI_ARCHIVE_UDD_FAILED | AUDIT_FAIL | Failed to archive user data disk \${UserDiskName} to location \${SVIPath} |
| BROKER_SVI_ARCHIVE_UDD_SUCCEEDED | AUDIT_SUCCESS | Archived user data disk \${UserDiskName} to location \${SVIPath} |
| BROKER_SVI_ATTACH_UDD_FAILED | AUDIT_FAIL | Failed to attach user data disk \${UserDiskName} to VM \${SVIVMID} |
| BROKER_SVI_ATTACH_UDD_SUCCEEDED | AUDIT_SUCCESS | Attached user data disk \${UserDiskName} to VM \${SVIVMID} |
| BROKER_SVI_DETACH_UDD_FAILED | AUDIT_FAIL | Failed to detach user data disk \${UserDiskName} from VM \${SVIVMID} |
| BROKER_SVI_DETACH_UDD_SUCCEEDED | AUDIT_SUCCESS | Detached user data disk \${UserDiskName} from VM \${SVIVMID} |
| BROKER_USER_AUTHFAILED_ACCOUNT_DISABLED | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because the account is disabled |
| BROKER_USER_AUTHFAILED_ACCOUNT_EXPIRED | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because the account has expired |
| BROKER_USER_AUTHFAILED_ACCOUNT_LOCKED_OUT | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because the account is locked out |
| BROKER_USER_AUTHFAILED_ACCOUNT_RESTRICTION | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because of an account restriction |
| BROKER_USER_AUTHFAILED_BAD_USER_PASSWORD | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because of a bad username or password |

Table 2-4. Connection Server Events (continued)

| Event Type | Severity | ModuleAndEventText |
|---|-----------------|--|
| BROKER_USER_AUTHFAILED_GENERAL | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate |
| BROKER_USER_AUTHFAILED_NO_LOGON_SERVERS | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because there are no logon servers |
| BROKER_USER_AUTHFAILED_PASSWORD_EXPIRED | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because the password has expired |
| BROKER_USER_AUTHFAILED_PASSWORD_MUST_CHANGE | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because the password must change |
| BROKER_USER_AUTHFAILED_SECUREID_ACCESS_DENIED | AUDIT_FAIL | SecurID access denied for user \${UserDisplayName} |
| BROKER_USER_AUTHFAILED_SECUREID_NEWPIN_REJECTED | AUDIT_FAIL | SecurID access denied for user \${UserDisplayName} because new pin was rejected |
| BROKER_USER_AUTHFAILED_SECUREID_WRONG_NEXTTOKEN | AUDIT_FAIL | SecurID access denied for user \${UserDisplayName} because wrong next token entered |
| BROKER_USER_AUTHFAILED_SECUREID_WRONG_STATE | AUDIT_FAIL | SecurID access denied for user \${UserDisplayName} because of incorrect state |
| BROKER_USER_AUTHFAILED_TIME_RESTRICTION | AUDIT_FAIL | User \${UserDisplayName} failed to authenticate because of a time restriction |
| BROKER_USER_NOT_AUTHORIZED | AUDIT_FAIL | User \${UserDisplayName} has authenticated, but is not authorized to perform the operation |
| BROKER_USER_NOT_ENTITLED | AUDIT_FAIL | User \${UserDisplayName} has authenticated, but is not entitled to any Pools |
| BROKER_USERCHANGEDPASSWORD | AUDIT_SUCCESS | Password for \${UserDisplayName} has been changed by the user |
| BROKER_USERLOGGEDIN | AUDIT_SUCCESS | User \${UserDisplayName} has logged in |
| BROKER_USERLOGGEDOUT | AUDIT_SUCCESS | User \${UserDisplayName} has logged out |
| BROKER_VC_DISABLED | INFO | vCenter at address \${VCAddress} has been temporarily disabled |
| BROKER_VC_ENABLED | INFO | vCenter at address \${VCAddress} has been enabled |
| BROKER_VC_STATUS_CHANGED_CANNOT_LOGIN | WARNING | Cannot log in to vCenter at address \${VCAddress} |
| BROKER_VC_STATUS_CHANGED_DOWN | INFO | vCenter at address \${VCAddress} is down |
| BROKER_VC_STATUS_CHANGED_INVALID_CREDENTIALS | WARNING | vCenter at address \${VCAddress} has invalid credentials |
| BROKER_VC_STATUS_CHANGED_NOT_YET_CONNECTED | INFO | Not yet connected to vCenter at address \${VCAddress} |
| BROKER_VC_STATUS_CHANGED_RECONNECTING | INFO | Reconnecting to vCenter at address \${VCAddress} |
| BROKER_VC_STATUS_CHANGED_UNKNOWN | WARNING | The status of vCenter at address \${VCAddress} is unknown |
| BROKER_VC_STATUS_CHANGED_UP | INFO | vCenter at address \${VCAddress} is up |

Horizon Agent Events

Horizon Agent events report Horizon Agent-related information, such as the users who have logged in to or disconnected from a specific machine, whether Horizon Agent has shut down on a specific machine, and whether Horizon Agent has sent a start up message from a specific machine to Horizon Connection Server.

Table 2-5. Horizon Agent Events

| Event Type | Severity | ModuleAndEventText |
|-----------------------|----------|---|
| AGENT_CONNECTED | INFO | User \${UserDisplayName} has logged in to a new session on machine \${MachineName} |
| AGENT_DISCONNECTED | INFO | User \${UserDisplayName} has disconnected from machine \${MachineName} |
| AGENT_ENDED | INFO | User \${UserDisplayName} has logged off machine \${MachineName} |
| AGENT_PENDING | INFO | The agent running on machine \${MachineName} has accepted an allocated session for user \${UserDisplayName} |
| AGENT_PENDING_EXPIRED | WARNING | The pending session on machine \${MachineName} for user \${UserDisplayName} has expired |
| AGENT_RECONFIGURED | INFO | Machine \${MachineName} has been successfully reconfigured |
| AGENT_RECONNECTED | INFO | User \${UserDisplayName} has reconnected to machine \${MachineName} |
| AGENT_RESUME | INFO | The agent on machine \${MachineName} sent a resume message |
| AGENT_SHUTDOWN | INFO | The agent running on machine \${MachineName} has shut down, this machine will be unavailable |
| AGENT_STARTUP | INFO | The agent running on machine \${MachineName} has contacted the connection server and sent a startup message |
| AGENT_SUSPEND | INFO | The agent on machine \${MachineName} sent a suspend message |

Horizon Administrator Events

Horizon Administrator events report information about actions that users initiate in Horizon Administrator.

Table 2-6. Horizon Administrator Events

| EventType | Severity | ModuleAndEventText |
|-------------------------------|---------------|--|
| ADMIN_ADD_DESKTOP_ENTITLEMENT | AUDIT_SUCCESS | \${EntitlementDisplay} was entitled to Pool \${DesktopId} by \${UserDisplayName} |
| ADMIN_ADD_LICENSE | AUDIT_SUCCESS | \${UserDisplayName} added license |
| ADMIN_ADD_LICENSE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add license |
| ADMIN_ADD_PM | AUDIT_SUCCESS | \${UserDisplayName} added physical machine \${MachineName} to Pool \${DesktopId} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|--|---------------|---|
| ADMIN_ADD_PM_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to add physical machine `\${MachineName}` to Pool `\${DesktopId}` |
| ADMIN_ADD_THINAPP_ENTITLEMENT | AUDIT_SUCCESS | Application `\${ThinAppDisplayName}` was assigned to Desktop `\${MachineName}` by `\${UserDisplayName}` |
| ADMIN_ADD_THINAPP_ENTITLEMENT_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to add Application entitlement |
| ADMIN_ADD_THINAPP_POOL_ENTITLEMENT | AUDIT_SUCCESS | Application `\${ThinAppDisplayName}` was assigned to Pool `\${DesktopId}` by `\${UserDisplayName}` |
| ADMIN_ADMINISTRATOR_REMOVE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to remove all permissions for Administrator `\${AdminPermissionEntity}` |
| ADMIN_ADMINISTRATOR_REMOVED | AUDIT_SUCCESS | `\${UserDisplayName}` removed all permissions for Administrator `\${AdminPermissionEntity}` |
| ADMIN_CONNECTION_BROKER_UPDATE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to update connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_BROKER_UPDATED | AUDIT_SUCCESS | `\${UserDisplayName}` updated connection broker `\${BrokerId}`: (`\${AttrChangeType}`: `\${AttrName}` = `\${AttrValue}`) |
| ADMIN_CONNECTION_SERVER_BACKUP_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to initiate a backup of connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_SERVER_BACKUP_INITIATED | AUDIT_SUCCESS | `\${UserDisplayName}` initiated a backup of connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_SERVER_DISABLE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to disable connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_SERVER_DISABLED | AUDIT_SUCCESS | `\${UserDisplayName}` is disabling connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_SERVER_ENABLE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to enable connection broker `\${BrokerId}` |
| ADMIN_CONNECTION_SERVER_ENABLED | AUDIT_SUCCESS | `\${UserDisplayName}` is enabling connection broker `\${BrokerId}` |
| ADMIN_DATABASE_CONFIGURATION_ADD_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to add database configuration |
| ADMIN_DATABASE_CONFIGURATION_ADDED | AUDIT_SUCCESS | `\${UserDisplayName}` has added database configuration |
| ADMIN_DATABASE_CONFIGURATION_DELETE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to delete database configuration |
| ADMIN_DATABASE_CONFIGURATION_DELETE_FAILED | AUDIT_SUCCESS | `\${UserDisplayName}` has deleted database configuration |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|--|---------------|--|
| ADMIN_DATABASE_CONFIGURATION_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update database configuration |
| ADMIN_DATABASE_CONFIGURATION_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} has updated database configuration |
| ADMIN_DEFAULT_DESKTOPPOOL_ASSIGN | AUDIT_SUCCESS | \${UserDisplayName} assigned Pool \${DesktopId} for default desktop to \${UserName} |
| ADMIN_DEFAULT_DESKTOPPOOL_ASSIGN_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to assign Pool \${DesktopId} for default desktop to \${UserName} |
| ADMIN_DEFAULT_DESKTOPPOOL_UNASSIGN | AUDIT_SUCCESS | \${UserDisplayName} removed pool assignment for default desktop to \${UserName} |
| ADMIN_DEFAULT_DESKTOPPOOL_UNASSIGN_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove Pool assignment for default desktop to \${UserName} |
| ADMIN_DESKTOP_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added Pool \${DesktopId} |
| ADMIN_DESKTOP_ASSIGN | AUDIT_SUCCESS | \${UserDisplayName} assigned Desktop \${MachineName} to \${UserName} |
| ADMIN_DESKTOP_ASSIGN_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to assign Desktop \${MachineName} to \${UserName} |
| ADMIN_DESKTOP_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited Pool \${DesktopId} (\${AttrChangeType}): \${AttrName} = \${AttrValue} |
| ADMIN_DESKTOP_MAINTENANCE_MODE_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update desktop \${MachineName} to \${MaintenanceMode} maintenance mode |
| ADMIN_DESKTOP_MAINTENANCE_MODE_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated desktop \${MachineName} to \${MaintenanceMode} maintenance mode |
| ADMIN_DESKTOP_UNASSIGN | AUDIT_SUCCESS | \${UserDisplayName} removed assignment for Desktop \${MachineName} |
| ADMIN_DESKTOP_UNASSIGN_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove assignment for Desktop \${MachineName} |
| ADMIN_ENABLE_DESKTOP_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to set Pool \${DesktopId} to \${EnableStatus} |
| ADMIN_ENABLE_DESKTOP_SUCCEEDED | AUDIT_SUCCESS | \${UserDisplayName} set Pool \${DesktopId} to \${EnableStatus} |
| ADMIN_ENABLED_DESKTOP_PROVISION_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to set provisioning for Pool \${DesktopId} to \${EnableStatus} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|---|---------------|--|
| ADMIN_ENABLED_DESKTOP_PROVISION_SUCCEEDED | AUDIT_SUCCESS | \${UserDisplayName} set provisioning for Pool \${DesktopId} to \${EnableStatus} |
| ADMIN_EVENT_CONFIGURATION_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update event configuration |
| ADMIN_EVENT_CONFIGURATION_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} has updated global configuration |
| ADMIN_FOLDER_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add folder \$ {AdminFolderName} |
| ADMIN_FOLDER_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added folder \$ {AdminFolderName} |
| ADMIN_FOLDER_CHANGE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to change object \${ObjectID}(type=\${ObjectType}) to folder \$ {AdminFolderName} |
| ADMIN_FOLDER_CHANGED | AUDIT_SUCCESS | \${UserDisplayName} changed object \$ {ObjectID}(type=\${ObjectType}) to folder \$ {AdminFolderName} |
| ADMIN_FOLDER_DELETE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to delete folder \$ {AdminFolderName} |
| ADMIN_FOLDER_DELETED | AUDIT_SUCCESS | \${UserDisplayName} deleted folder \$ {AdminFolderName} |
| ADMIN_GLOBAL_CONFIGURATION_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update global configuration |
| ADMIN_GLOBAL_CONFIGURATION_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated global configuration (\${AttrChangeType}: \$ {AttrName} = \$ {AttrValue}) |
| ADMIN_GLOBAL_POLICY_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update global policies |
| ADMIN_GLOBAL_POLICY_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated global policy (\${AttrChangeType}: \$ {AttrName} = \$ {AttrValue}) |
| ADMIN_PERFMON_CONFIGURATION_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update performance monitoring configuration |
| ADMIN_PERFMON_CONFIGURATION_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} has updated performance monitoring configuration |
| ADMIN_PERMISSION_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add Permission to \$ {AdminPermissionEntity} with Role \$ {AdminRoleName} on Folder \$ {AdminFolderName} |
| ADMIN_PERMISSION_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added Permission to \$ {AdminPermissionEntity} with Role \$ {AdminRoleName} on Folder \$ {AdminFolderName} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|---|---------------|--|
| ADMIN_PERMISSION_REMOVE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to remove Permission to `\${AdminPermissionEntity}` with Role `\${AdminRoleName}` on Folder `\${AdminFolderName}` |
| ADMIN_PERMISSION_REMOVED | AUDIT_SUCCESS | `\${UserDisplayName}` removed Permission to `\${AdminPermissionEntity}` with Role `\${AdminRoleName}` on Folder `\${AdminFolderName}` |
| ADMIN_POOL_POLICY_UPDATE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to update Pool `\${DesktopId}` policies |
| ADMIN_POOL_POLICY_UPDATED | AUDIT_SUCCESS | `\${UserDisplayName}` updated Pool `\${DesktopId}` policy (`\${AttrChangeType}`: `\${AttrName}` = `\${AttrValue}`) |
| ADMIN_REMOVE_DESKTOP_ENTITLEMENT | AUDIT_SUCCESS | `\${EntitlementDisplay}` was unentitled from Pool `\${DesktopId}` by `\${UserDisplayName}` |
| ADMIN_REMOVE_DESKTOP_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to removed Pool `\${DesktopId}` |
| ADMIN_REMOVE_DESKTOP_SUCCEEDED | AUDIT_SUCCESS | `\${UserDisplayName}` removed Pool `\${DesktopId}` |
| ADMIN_REMOVE_THINAPP_ENTITLEMENT | AUDIT_SUCCESS | Application `\${ThinAppDisplayName}` was unassigned from Desktop `\${MachineName}` by `\${UserDisplayName}` |
| ADMIN_REMOVE_THINAPP_ENTITLEMENT_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to remove Application entitlement |
| ADMIN_REMOVE_THINAPP_POOL_ENTITLEMENT | AUDIT_SUCCESS | Application `\${ThinAppDisplayName}` was unassigned from Pool `\${DesktopId}` by `\${UserDisplayName}` |
| ADMIN_RESET_THINAPP_STATE | AUDIT_SUCCESS | Application `\${ThinAppDisplayName}` state are reset for Desktop `\${DesktopDisplayName}` by `\${UserDisplayName}` |
| ADMIN_RESET_THINAPP_STATE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to reset Application state for `\${ThinAppDisplayName}` |
| ADMIN_ROLE_ADD_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to add Role `\${AdminRoleName}` with privileges `\${AdminPrivilegeName}` |
| ADMIN_ROLE_ADDED | AUDIT_SUCCESS | `\${UserDisplayName}` added Role `\${AdminRoleName}` with privileges `\${AdminPrivilegeName}` |
| ADMIN_ROLE_PRIV_UPDATE_FAILED | AUDIT_FAIL | `\${UserDisplayName}` failed to update Role `\${AdminRoleName}` to privileges `\${AdminPrivilegeName}` |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|--|---------------|--|
| ADMIN_ROLE_PRIV_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated Role \$ {AdminRoleName} to privileges \$ {AdminPrivilegeName} |
| ADMIN_ROLE_REMOVE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove Role \$ {AdminRoleName} |
| ADMIN_ROLE_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed Role \$ {AdminRoleName} |
| ADMIN_ROLE_RENAME_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to rename Role \$ {AdminRoleName} to \$ {AdminRoleNewName} |
| ADMIN_ROLE_RENAMED | AUDIT_SUCCESS | \${UserDisplayName} renamed Role \$ {AdminRoleName} to \$ {AdminRoleNewName} |
| ADMIN_SECURITY_SERVER_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add security server \$ {SecurityServerId} |
| ADMIN_SECURITY_SERVER_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added security server \$ {SecurityServerId} |
| ADMIN_SECURITY_SERVER_EDIT_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to edit security server \$ {SecurityServerId} |
| ADMIN_SECURITY_SERVER_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited security server \$ {SecurityServerId} (\$ {AttrChangeType}: \$ {AttrName} = \$ {AttrValue}) |
| ADMIN_SECURITY_SERVER_REMOVE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove security server \$ {SecurityServerId} |
| ADMIN_SECURITY_SERVER_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed security server \$ {SecurityServerId} |
| ADMIN_SESSION_SENDMSG | AUDIT_SUCCESS | \${UserDisplayName} sent message (\$ {SessionMessage}) to session (User \$ {UserName}, Desktop \$ {MachineName}) |
| ADMIN_SESSION_SENDMSG_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to send message (\$ {SessionMessage}) to session \$ {ObjectId} |
| ADMIN_SVI_ADD_DEPLOYMENT_GROUP_FAILED | AUDIT_FAIL | Failed to add deployment group for \$ {SVIParentVM} : \$ {SVISnapshot} |
| ADMIN_SVI_ADD_DEPLOYMENT_GROUP_SUCCEEDED | AUDIT_SUCCESS | Added deployment group \$ {SVIDeploymentGroupId} for \$ {SVIParentVM} : \$ {SVISnapshot} |
| ADMIN_SVI_ADD_UDD_FAILED | AUDIT_FAIL | Failed to add user data disk \$ {UserDiskName} |
| ADMIN_SVI_ADD_UDD_SUCCEEDED | AUDIT_SUCCESS | Added user data disk \$ {UserDiskName} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|--|---------------|---|
| ADMIN_SVI_ADMIN_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added SVI QuickPrep domain \${SVIAdminFqdn}(\${SVIAdminName}) |
| ADMIN_SVI_ADMIN_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed SVI QuickPrep domain (id=\${SVIAdminID}) |
| ADMIN_SVI_ADMIN_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated SVI QuickPrep domain \${SVIAdminFqdn}(\${SVIAdminName}) |
| ADMIN_SVI_ATTACH_UDD_FAILED | AUDIT_FAIL | Failed to request attach user data disk \${UserDiskName} to VM \${SVIVMID} |
| ADMIN_SVI_ATTACH_UDD_SUCCEEDED | AUDIT_SUCCESS | Requested attach user data disk \${UserDiskName} to VM \${SVIVMID} |
| ADMIN_SVI_DELETE_UDD_FAILED | AUDIT_FAIL | Failed to delete user data disk \${UserDiskName} |
| ADMIN_SVI_DELETE_UDD_SUCCEEDED | AUDIT_SUCCESS | Deleted user data disk \${UserDiskName} |
| ADMIN_SVI_DETACH_UDD_FAILED | AUDIT_FAIL | Failed to request detach user data disk \${UserDiskName} from VM \${SVIVMID} |
| ADMIN_SVI_DETACH_UDD_SUCCEEDED | AUDIT_SUCCESS | Requested detach user data disk \${UserDiskName} from VM \${SVIVMID} |
| ADMIN_SVI_REBALANCE_VM_FAILED | AUDIT_FAIL | Failed to rebalance VM \${SVIVMID} |
| ADMIN_SVI_REBALANCE_VM_SUCCEEDED | AUDIT_SUCCESS | Rebalanced VM \${SVIVMID} |
| ADMIN_SVI_REFRESH_VM_FAILED | AUDIT_FAIL | Failed to refresh VM \${SVIVMID} |
| ADMIN_SVI_REFRESH_VM_SUCCEEDED | AUDIT_SUCCESS | Refreshed VM \${SVIVMID} |
| ADMIN_SVI_RESYNC_VM_FAILED | AUDIT_FAIL | Failed to resync VM \${SVIVMID} to deployment group \${SVIDeploymentGroupID} |
| ADMIN_SVI_RESYNC_VM_SUCCEEDED | AUDIT_SUCCESS | Resyncd VM \${SVIVMID} to deployment group \${SVIDeploymentGroupID} |
| ADMIN_SVI_UPDATE_POOL_DEPLOYMENT_GROUP_FAILED | AUDIT_FAIL | Failed to update pool \${DesktopID} to deployment group \${SVIDeploymentGroupID} |
| ADMIN_SVI_UPDATE_POOL_DEPLOYMENT_GROUP_SUCCEEDED | AUDIT_SUCCESS | Updated pool \${DesktopID} to deployment group \${SVIDeploymentGroupID} |
| ADMIN_SVI_UPDATE_UDD_FAILED | AUDIT_FAIL | Failed to update user data disk \${UserDiskName} |
| ADMIN_SVI_UPDATE_UDD_SUCCEEDED | AUDIT_SUCCESS | Set user data disk \${UserDiskName} pool to \${DesktopID} and user to \${UserName} |
| ADMIN_THINAPP_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add Application \${ThinAppDisplayName} |
| ADMIN_THINAPP_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added Application \${ThinAppDisplayName} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|---------------------------------------|---------------|---|
| ADMIN_THINAPP_DESKTOP_AVAILABLE | AUDIT_SUCCESS | Application \${ThinAppDisplayName} is now available on Desktop \$ {DesktopDisplayName} |
| ADMIN_THINAPP_DESKTOP_REMOVED | AUDIT_SUCCESS | Application \${ThinAppDisplayName} has been removed from Desktop \$ {DesktopDisplayName} |
| ADMIN_THINAPP_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited Application \$ {ThinAppDisplayName} |
| ADMIN_THINAPP_FAILED_DESKTOP_DELIVERY | AUDIT_FAIL | Failed to deliver Application \$ {ThinAppDisplayName} to Desktop \$ {DesktopDisplayName} |
| ADMIN_THINAPP_FAILED_DESKTOP_REMOVAL | AUDIT_FAIL | Failed to remove Application \$ {ThinAppDisplayName} from Desktop \$ {DesktopDisplayName} |
| ADMIN_THINAPP_GROUP_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add Application Template \$ {ThinAppGroupName} |
| ADMIN_THINAPP_GROUP_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added Application Template \$ {ThinAppGroupName} with Applications \$ {ThinAppGroupApplications} |
| ADMIN_THINAPP_GROUP_EDIT_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to edit Application Template \$ {ThinAppGroupName} |
| ADMIN_THINAPP_GROUP_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited Application Template \$ {ThinAppGroupName} with Applications \$ {ThinAppGroupApplications} |
| ADMIN_THINAPP_GROUP_REMOVE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove Application Template \$ {ThinAppGroupName} |
| ADMIN_THINAPP_GROUP_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed Application Template \$ {ThinAppGroupName} |
| ADMIN_THINAPP_REMOVE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove Application \$ {ThinAppDisplayName} |
| ADMIN_THINAPP_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed Application \$ {ThinAppDisplayName} |
| ADMIN_THINAPP_REPO_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add Repository \$ {ThinAppRepositoryName}, path \$ {ThinAppRepositoryPath} |
| ADMIN_THINAPP_REPO_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added Repository \$ {ThinAppRepositoryName}, path \$ {ThinAppRepositoryPath} |
| ADMIN_THINAPP_REPO_EDIT_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to edit Repository \$ {ThinAppRepositoryName}, path \$ {ThinAppRepositoryPath} |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|---------------------------------|---------------|--|
| ADMIN_THINAPP_REPO_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited Repository \${ThinAppRepositoryName}, path \${ThinAppRepositoryPath} |
| ADMIN_THINAPP_REPO_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed Repository \${ThinAppRepositoryName} |
| ADMIN_UNREGISTER_PM | AUDIT_SUCCESS | \${UserDisplayName} unregistered physical machine \${MachineName}) |
| ADMIN_UNREGISTER_PM_FAILED | AUDIT_FAIL | \${UserDisplayName} fails to unregister physical machine \${MachineName}) |
| ADMIN_USER_INFO_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update user info with AD server for \${UserName} |
| ADMIN_USER_INFO_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated user info with AD server for \${UserName} |
| ADMIN_USER_POLICY_DELETE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to delete Pool \${DesktopId} override policies for user \${UserName} |
| ADMIN_USER_POLICY_DELETED | AUDIT_SUCCESS | \${UserDisplayName} deleted Pool \${DesktopId} override policy for user \${UserName} (\${AttrChangeType}): \${AttrName} = \${AttrValue}) |
| ADMIN_USER_POLICY_UPDATE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to update Pool \${DesktopId} policies for user \${UserName} |
| ADMIN_USER_POLICY_UPDATED | AUDIT_SUCCESS | \${UserDisplayName} updated Pool \${DesktopId} policy for user \${UserName} (\${AttrChangeType}): \${AttrName} = \${AttrValue}) |
| ADMIN_USERLOGGEDIN | AUDIT_SUCCESS | User \${UserDisplayName} has logged in to View Administrator |
| ADMIN_USERLOGGEDOUT | AUDIT_SUCCESS | User \${UserDisplayName} has logged out from View Administrator |
| ADMIN_VC_ADD_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to add VC server \${VCAddress} |
| ADMIN_VC_ADDED | AUDIT_SUCCESS | \${UserDisplayName} added VC server \${VCAddress} |
| ADMIN_VC_EDITED | AUDIT_SUCCESS | \${UserDisplayName} edited VC server \${VCAddress} (\${AttrChangeType}): \${AttrName} = \${AttrValue}) |
| ADMIN_VC_LICINV_ALARM_DISABLED | AUDIT_SUCCESS | Alarm on VC server \${VCAddress} for License Inventory monitoring was disabled as all Hosts have desktop licenses |

Table 2-6. Horizon Administrator Events (continued)

| EventType | Severity | ModuleAndEventText |
|------------------------|---------------|--|
| ADMIN_VC_REMOVE_FAILED | AUDIT_FAIL | \${UserDisplayName} failed to remove VC server \${VCAddress} |
| ADMIN_VC_REMOVED | AUDIT_SUCCESS | \${UserDisplayName} removed VC server \${VCAddress} |

Event Message Attributes

ModuleAndEventText messages use certain attributes. To determine the data type for an attribute, you can examine its value in the type column in the event_data or event_data_historical table.

Table 2-7. Attributes that ModuleAndEventText Messages Use

| Attribute Name | Description |
|-----------------------|---|
| AdminFolderName | Name of a folder that requires privileged access. |
| AdminPermissionEntity | Name of an object that requires privileged access. |
| AdminPrivilegeName | Name of an administrative privilege. |
| AdminRoleName | Name of an administrative role. |
| AdminRoleNewName | New name of an administrative role. |
| AttrChangeType | Type of change that was applied to a generic attribute. |
| AttrName | Name of a generic attribute. |
| AttrValue | Value of a generic attribute. |
| BrokerId | Identifier of a Connection Server instance. |
| BrokerName | Name of a Connection Server instance. |
| DesktopDisplayName | Display name of a desktop pool. |
| DesktopId | Identifier of a desktop pool. |
| EntitlementDisplay | Display name of a desktop entitlement. |
| MachineId | Name of a physical or virtual machine. |
| MachineName | Name of a physical or virtual machine. |
| MaintenanceMode | Maintenance mode state. |
| ObjectID | Identifier of an inventory object. |
| ObjectType | Type of an inventory object. |
| PolicyDisplayName | Display name of a policy. |
| PolicyObject | Identifier of a policy object. |
| PolicyValue | Value of a policy object. |
| ProtocolId | Identifier of a display protocol. |
| SecurityServerId | Identifier of a security server. |
| SVIAdminFqdn | FQDN of a QuickPrep domain. |

Table 2-7. Attributes that ModuleAndEventText Messages Use (continued)

| Attribute Name | Description |
|-----------------------|--|
| SVIAdminID | Identifier of a QuickPrep domain. |
| SVIAdminName | Name of a QuickPrep domain. |
| SVIDeploymentGroupID | Identifier of a View Composer deployment group. |
| SVIOperation | Name of a View Composer operation. |
| SVIParentVM | Parent virtual machine in View Composer. |
| SVIPath | Path of an object in View Composer. |
| SVISnapshot | Snapshot in View Composer. |
| SVIVMID | Identifier of a virtual machine in View Composer. |
| ThinAppDisplayName | Display name of a ThinApp object. |
| ThinAppId | Identifier of a ThinApp object. |
| ThinAppRepositoryName | Name of a ThinApp repository |
| ThinAppRepositoryPath | Path of a ThinApp repository. |
| Time | Date and time value. |
| UserCount | Maximum number of desktop users over a 24-hour period. |
| UserDiskName | Name of a user data disk. |
| UserDisplayName | User name in the form DOMAIN\username. |
| UserName | Name of a user in Active Directory. |
| VCAddress | URL of a vCenter Server. |

Sample Database Queries and Views

You can query the event_historical database to display error events, warning events, and specific recent events.

Note Replace the dbo.VE_ prefix in the following examples with the appropriate prefix for your event database.

List Error Events

The following query displays all error events from the event_historical table.

```
CREATE VIEW error_events AS
(
  SELECT ev.EventID, ev.Time, ev.Module, ev.EventType, ev.ModuleAndEventText
  FROM dbo.VE_event_historical AS ev
  WHERE ev.Severity = 'ERROR'
);
```

List Warning Events

The following query displays all warning events from the event_historical table.

```
CREATE VIEW warning_events AS
(
  SELECT ev.EventID, ev.Time, ev.Module, ev.EventType, ev.ModuleAndEventText
    FROM dbo.VE_event_historical AS ev
   WHERE ev.Severity = 'WARNING'
);
```

List Recent Events

The following query lists all recent events that are associated with the user fred in the domain MYDOM.

```
CREATE VIEW user_fred_events AS
(
  SELECT ev.EventID, ev.Time, ev.Module, ev.EventType, ev.Severity, ev.Acknowledged
    FROM dbo.VE_event_historical AS ev,
         dbo.VE_event_data_historical AS ed
   WHERE ev.EventID = ed.EventID AND ed.Name = 'UserDisplayName' AND ed.StrValue =
         'MYDOM\fred'
);
```

The following query lists all recent events where the agent on a machine shut down.

```
CREATE VIEW agent_shutdown_events AS
(
  SELECT ev.EventID, ev.Time, ed.StrValue
    FROM dbo.VE_event_historical AS ev,
         dbo.VE_event_data_historical AS ed
   WHERE ev.EventID = ed.EventID AND ev.EventType = 'AGENT_SHUTDOWN' AND
         ed.Name = 'MachineName'
);
```

The following query lists all recent events where a desktop failed to launch because the desktop pool was empty.

```
CREATE VIEW desktop_launch_failure_events AS
(
  SELECT ev.EventID, ev.Time, ed1.StrValue, ed2.StrValue
    FROM dbo.VE_event_historical AS ev,
         dbo.VE_event_data_historical AS ed1,
         dbo.VE_event_data_historical AS ed2
   WHERE ev.EventID = ed1.EventID AND ev.EventID = ed2.EventID AND
         ev.EventType = 'BROKER_POOL_EMPTY' AND
         ed1.Name = 'UserDisplayName' AND ed2.Name = 'DesktopId'
);
```

The following query lists all recent events where an administrator removed a desktop pool.

```
CREATE VIEW desktop_pool_removed_events AS
(
```

```

SELECT ev.EventID, ev.Time, ed1.StrValue, ed2.StrValue
FROM dbo.VE_event_historical AS ev,
     dbo.VE_event_data_historical AS ed1,
     dbo.VE_event_data_historical AS ed2
WHERE ev.EventID = ed1.EventID AND ev.EventID = ed2.EventID AND
      ev.EventType = 'ADMIN_DESKTOP_REMOVED' AND
      ed1.Name = 'UserDisplayName' AND ed2.Name = 'DesktopId'
);

```

The following query lists all recent events where an administrator added a ThinApp repository.

```

CREATE VIEW thinapp_repository_added_events AS
(
  SELECT ev.EventID, ev.Time, ed1.StrValue, ed2.StrValue, ed3.StrValue
  FROM dbo.VE_event_historical AS ev,
       dbo.VE_event_data_historical AS ed1,
       dbo.VE_event_data_historical AS ed2,
       dbo.VE_event_data_historical AS ed3
  WHERE ev.EventID = ed1.EventID AND ev.EventID = ed2.EventID AND ev.EventID = ed3.EventID
  AND
        ev.EventType = 'ADMIN_THINAPP_REPO_ADDED' AND
        ed1.Name = 'UserDisplayName' AND ed2.Name = 'ThinAppRepositoryName' AND
        ed3.Name = 'ThinAppRepositoryPath'
);

```

Enabling Horizon 7 for Cloud-Hosted Subscriptions and Services

3

You can enable your Horizon 7 deployment to use a cloud-hosted subscription which includes leveraging cloud-hosted services for use with cloud-connected Horizon 7 pods. You must use the Horizon 7 Cloud Connector virtual appliance to connect your Horizon 7 deployment with the cloud-based management plane provided by VMware Horizon Cloud Service.

Subscription Licenses for Horizon 7

Horizon 7 subscription licenses are available through the Horizon universal license, which is available standalone and as part of the Workspace ONE Enterprise bundle.

The Horizon 7 subscription license provides the same Horizon 7 product components with more flexible deployment options. Horizon 7 subscription licenses enable Horizon 7 deployment in the customer's data center, private cloud, and in VMware Cloud on AWS. After you configure the Horizon 7 Cloud Connector virtual appliance to connect your pod to Horizon Cloud, you can view the subscription license term in Horizon Console.

Note The subscription license for Horizon 7 is managed by VMware only after you deploy the Horizon 7 Cloud Connector virtual appliance. You will not receive the license key for VMware Horizon 7 with this subscription license. However, you will receive license keys for vSphere, vCenter Server, vSAN, App Volumes, and Dynamic Environment Manager with this subscription license. You will receive these keys in an email with the following subject: Welcome to VMware Horizon On-Premises Subscription.

Cloud-Hosted Services for Horizon 7 Pods

After you configure the Horizon 7 Cloud Connector virtual appliance to connect your pod to VMware Horizon Cloud Service, you can also leverage the cloud-hosted management, features, and workflows that are provided by Horizon Cloud and which are available to you according to that subscription license. For information about those cloud-hosted services, see [Introduction to Horizon Cloud](#) in the VMware Horizon Cloud Service documentation.

Horizon 7 Cloud Connector

The Horizon 7 Cloud Connector is a virtual appliance that is deployed alongside the Horizon 7 pod and connects that Horizon 7 pod with VMware Horizon Cloud Service. Horizon 7 Cloud Connector is a required component that bridges your Horizon 7 pods with VMware Horizon Cloud Service. The Horizon 7 Cloud Connector is required for cloud-hosted services including Horizon 7 subscription licenses, health status dashboard, Horizon Help Desk Tool, and other cloud-hosted management features and workflows in Horizon Cloud.

You must have an active My VMware account to purchase a Horizon 7 license from <https://my.vmware.com>. You then receive an email with the link to download the Horizon 7 Cloud Connector as an OVA file.

When you deploy the Horizon 7 Cloud Connector virtual appliance from vSphere Web Client, you pair the Cloud Connector with the Connection Server pod you want to connect to Horizon Cloud Service. As part of the pairing process, the Horizon 7 Cloud Connector virtual appliance connects the Connection Server to Horizon Cloud Service to manage the Horizon 7 subscription license and other cloud-hosted management services. With a Horizon 7 subscription license, you do not need to manually enter a Horizon 7 license key for the VMware Horizon 7 product activation. However, you do need to use the license keys to activate supporting components such as vSphere, vCenter Server, App Volumes, and others.

Note The Horizon 7 Cloud Connector virtual appliance does not support an IPv6 environment.

For more information about deploying the Horizon 7 Cloud Connector virtual appliance and completing that connection between a Horizon 7 pod with VMware Horizon Cloud Service, see the following topics in the Horizon Cloud Service documentation:

- [End-to-End Workflow When Your Very First Cloud-Connected Pod is from Connecting Horizon Cloud with an Existing Manually Deployed Horizon 7 Pod](#)
- [Connect Horizon Cloud with an Existing Manually Deployed Horizon 7 Pod](#)

Deploying Horizon 7 on VMware Cloud on AWS

4

VMware Cloud on AWS is a cloud service where you can deploy Horizon 7 desktops and applications.

For more information about deploying Horizon 7 on VMware Cloud on AWS, see the *"Horizon 7 on VMware Cloud on AWS Deployment Guide"* at <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/vmw-deploy-horizon-seven-on-vmware-cloud-on-aws.pdf>.

For a list of Horizon 7 features supported on VMware Cloud on AWS, see the VMware Knowledge Base article <https://kb.vmware.com/s/article/58539>.

For more information about VMware Cloud on AWS, see the VMware Cloud on AWS documentation at <https://docs.vmware.com/en/VMware-Cloud-on-AWS/index.html>.

For more information about the impact of SDDC upgrade on a Horizon 7 deployment on VMware Cloud on AWS, see the VMware Knowledge Base article <https://kb.vmware.com/s/article/74599>.

For more information about updating the vSAN Storage Policy FTT Level on a Horizon 7 deployment on VMware Cloud on AWS, see the VMware Knowledge Base article <https://kb.vmware.com/s/article/76366>.

Customizing LDAP Data

5

You can use VMware and Microsoft command-line tools to import and export LDAP configuration data to and from Horizon 7. These command-line tools import and export LDAP configuration data in LDAP Data Interchange Format (LDIF) configuration files.

This feature is intended for use by advanced administrators who want to perform automatic bulk configuration operations. To create scripts to update the Horizon 7 configuration, use Horizon 7 PowerCLI.

This chapter includes the following topics:

- [Introduction to LDAP Configuration Data](#)
- [Modifying LDAP Configuration Data](#)

Introduction to LDAP Configuration Data

All Horizon 7 configuration data is stored in an LDAP directory. Each Horizon Connection Server standard or replica instance contains a local LDAP configuration repository and a replication agreement between each of the Connection Server instances. This arrangement ensures that changes to one repository are automatically replicated to all other repositories.

When you use Horizon Administrator to modify the Horizon 7 configuration, the appropriate LDAP data is updated in the repository. For example, if you add a desktop pool, Horizon 7 stores information about users, user groups, and entitlements in LDAP. Connection Server instances manage other LDAP configuration data automatically, and they use the information in the repository to control Horizon 7 operations.

You can use LDIF configuration files to perform a number of tasks, including transferring configuration data between Connection Server instances and backing up your Horizon 7 configuration so that you can restore the state of a Connection Server instance.

You can also use LDIF configuration files to define a large number of Horizon 7 objects, such as desktop pools, and add those objects to your Connection Server instances without having to use Horizon Administrator to perform the task manually.

Horizon 7 performs regular backups of the LDAP repository.

LDAP configuration data is transferred as plain ASCII text and conforms to the Internet Engineering Task Force (IETF) RFC 2849 standard.

Modifying LDAP Configuration Data

You can export LDAP configuration data on a Horizon Connection Server instance to an LDIF configuration file, modify the LDIF configuration file, and import the modified LDIF configuration file into other Connection Server instances to perform automatic bulk configuration operations.

You can obtain examples of LDIF syntax for any item of LDAP configuration data in Horizon by examining the contents of an exported LDIF configuration file. For example, you can extract the data for a desktop pool and use that data as a template to create a large number of desktop pools.

Export LDAP Configuration Data

You can use the `vdmexport` command-line utility to export configuration data from a standard or replica Connection Server instance to an LDIF configuration file.

Procedure

- 1 Log in to a standard or replica Connection Server instance as a user in the Administrators or Administrators (Read only) role.

You must be logged in as a user in the Administrators or Administrators (Read only) role to export configuration data from the Horizon configuration repository.

- 2 At the command prompt, type the `vdmexport` command.

By default, the `vdmexport` command-line utility is installed in the `C:\Program Files\VMware\VMware View\Server\tools\bin` directory.

The `vdmexport` command has the following options.

| Option | Description |
|-----------------|--|
| <code>-f</code> | Output file name for LDAP backup. |
| <code>-v</code> | The output file is verbatim (not encrypted). |
| <code>-c</code> | Similar to the <code>-v</code> option, but sensitive attribute values are not included in the output file. |
| <code>-k</code> | Outputs only kiosk client entries and related FSPs. |
| <code>-g</code> | Take a backup of the Cloud Pod Architecture global LDAP, rather than of local LDAP. |

For example, the following command exports a local LDIF configuration file.

```
vdmexport -f mylocalexport.LDF
```

The following command takes a backup of the Cloud Pod Architecture global LDAP.

```
vdmexport -f myglobalexport.LDF -g
```

Results

The `vdmexport` command writes the configuration of your Connection Server instance to the file that you specify. The command displays errors if your role has insufficient privileges to view the data in the configuration repository.

Defining a Desktop Pool in an LDIF Configuration File

You can define a desktop pool in an LDIF configuration file and import the customized LDIF configuration file to create a large number of desktop pools.

Note You can also create customized LDIF configuration files for other objects that are defined in the LDAP repository, including global configuration settings, configuration settings for a specific Horizon Connection Server instance or security server, and configuration settings for a specific user.

To define a desktop pool in an LDIF configuration file, you must add the following entries to the file.

- A Virtual Desktop VM entry for each virtual desktop in the desktop pool
- A VM Pool entry for each desktop pool
- A Desktop Application entry that defines the entitlement of the desktop pool

You associate each VM Pool entry with one Desktop Application entry in a one-to-one relationship. A Desktop Application entry cannot be shared between VM Pool entries, and a VM Pool entry can only be associated with one Desktop Application entry.

The following table describes the attributes you must specify when you modify a desktop pool definition in an LDIF configuration file.

Table 5-1. Important Attributes for Defining a Desktop Pool

| Entry | Attribute | Description |
|--|-----------|---|
| Virtual Desktop VM VM Pool Desktop Application | cn | Common name of an entry. If you require names to be generated automatically, specify globally unique identifier (GUID) strings. You can use any reliable GUID generator, such as the mechanism provided by .NET (for example, by calling <code>System.Guid.NewGuid().ToString()</code> in Visual Basic). |
| Desktop Application | member | <p>A list of Active Directory (AD) users and groups who are entitled to access the desktop pool. The attribute is specified in the form of a Windows Security Identifier (SID) reference. A member value of <code><SID=S-1-2-3-4></code> represents an AD user or group with the SID value S-1-2-3-4.</p> <p>In LDIF format, the left angle (<) character is reserved, so you must place two colons (::) after the attribute name and specify the SID value in base 64 format (for example, <code>PFNJRD1TLTEtMi0zLTQ+IA==</code>). Because this attribute is multivalued, you can use it on multiple lines to represent each entry in a list of SIDs.</p> |

Sample LDIF Configuration File Desktop Pool Entries

The following example is an excerpt from an LDIF configuration file. It shows sample entries for a desktop pool named Pool1, which contains two virtual desktops named VM1 and VM2. The desktop pool entry is paired with the Desktop Application entry, which is also named Pool1.

```
#
# Virtual Desktop VM entry VM1
#
DN: CN=vm1,OU=Servers,DC=vdi,DC=vmware,DC=int
changetype: add
objectClass: top
objectClass: pae-Server
objectClass: pae-WinServer
objectClass: pae-ThinWinServer
objectClass: pae-VM
cn: vm1
description: sample virtual desktop entry
pae-VmSuspended:: IA==
pae-OptIgnoreProcessList: 0
pae-MOID: vm-1
pae-VmState: READY
pae-ServerManaged: 1
pae-SSOEnabled: 1
pae-DisplayName: virtual desktop 1
pae-TunneledConnection: 1
pae-pwdEncryption: KERB5
ipHostNumber: vm1
pae-ClientProtVersion: 1
pae-WinDomain: NULL
pae-thinProto: XP_RDP
pae-Services: SESSION |, HEARTBEAT |, EVENTS |, USED |
pae-VmPath: /New Datacenter/vm/vm-1
pae-OptSuspendTimeout: 0
pae-OptDisconnectLimitTimeout: 0
pae-OptMaximumSessions: 0
pae-Disabled: 0

#
# Virtual Desktop VM entry VM2
#
DN: CN=vm2,OU=Servers,DC=vdi,DC=vmware,DC=int
changetype: add
objectClass: top
objectClass: pae-Server
objectClass: pae-WinServer
objectClass: pae-ThinWinServer
objectClass: pae-VM
cn: vm2
description: sample virtual desktop entry
pae-VmSuspended:: IA==
pae-OptIgnoreProcessList: 0
pae-MOID: vm-2
pae-VmState: READY
pae-ServerManaged: 1
```

```

pae-SSOEnabled: 1
pae-DisplayName: virtual desktop 2
pae-TunneledConnection: 1
pae-pwdEncryption: KERB5
ipHostNumber: vm2
pae-ClientProtVersion: 1
pae-WinDomain: NULL
pae-thinProto: XP_RDP
pae-Services: SESSION |, HEARTBEAT |, EVENTS |, USED |
pae-VmPath: /New Datacenter/vm/vm-2
pae-OptSuspendTimeout: 0
pae-OptDisconnectLimitTimeout: 0
pae-OptMaximumSessions: 0
pae-Disabled: 0
#
# Further Virtual Desktop VM entries as required
#
#
# VM Pool entry Pool1
#
DN: CN=Pool1,OU=Server Groups,DC=vdi,DC=vmware,DC=int
changetype: add
objectClass: top
objectClass: pae-ServerPool
cn: Pool1
pae-VCDN: CN=b180b93b-2dd3-4b58-8a81-b8534a4b7565,OU=VirtualCenter,OU=Properties,DC=vdi,
DC=vmware,DC=int
pae-MemberDN: CN=vm1,OU=Servers,DC=vdi,DC=vmware,DC=int
pae-MemberDN: CN=vm2,OU=Servers,DC=vdi,DC=vmware,DC=int
pae-VmPowerPolicy: remainon
pae-VmProvEnabled: 1
pae-VmProvSuspendOnError: 1
pae-VmStartClone: 1
pae-VmPoolCalculatedValues: 1
pae-ServerPoolType: 0
pae-VmMinimumCount: 0
pae-VmHeadroomCount: 0
pae-VmMaximumCount: 0
pae-Disabled: 0
#
# Desktop Application entry Pool1 -- one entry is required for each VM Pool
#
DN: CN=Pool1,OU=Applications,DC=vdi,DC=vmware,DC=int
changetype: add
objectClass: top
objectClass: pae-Entity
objectClass: pae-App
objectClass: pae-WinApp
objectClass: pae-ThinWinApp
objectClass: pae-DesktopApplication
cn: Pool1
member:: PFNJRDI1LTETMi0zLTQ+IA==
pae-Icon: /thinapp/icons/desktop.gif
pae-URL: \

```

```

pae-Servers: CN=Pool1,OU=Server Groups,DC=vdi,DC=vmware,DC=int
pae-ServerProtocolLevel: OSX_NETOP
pae-ServerProtocolLevel: OS2_NETOP
pae-ServerProtocolLevel: NT4_NETOP
pae-ServerProtocolLevel: WIN2K_NETOP
pae-ServerProtocolLevel: NT4_RDP
pae-ServerProtocolLevel: WIN2K_RDP
pae-ServerProtocolLevel: XP_RDP
pae-Disabled: 0

```

Import LDAP Configuration Data

You can use the `vdmimport` command to import configuration data from an LDIF configuration file into a standard or replica Connection Server instance.

Prerequisites

- Export LDAP configuration data to an LDIF configuration file. See [Export LDAP Configuration Data](#).
- If you are importing a Cloud Pod Architecture global LDIF configuration file, verify that the Cloud Pod Architecture feature is initialized on the Connection Server instance.

Procedure

- 1 Log in to a Connection Server instance as a user in the Administrators role.

You must be logged in as a user in the Administrators role to import configuration data into the Horizon configuration repository.

- 2 At the command prompt, type the `vdmimport` command.

By default, the `vdmimport` command-line utility is installed in the `C:\Program Files\VMware\VMware View\Server\tools\bin` directory.

The `vdmimport` command has the following options.

| Option | Description |
|-----------------|--|
| <code>-f</code> | Input file name. |
| <code>-i</code> | Shows file information about the specified LDIF configuration file. |
| <code>-d</code> | Decrypts the specified LDIF configuration file. |
| <code>-p</code> | Specifies the recovery password for decryption of an encrypted LDIF configuration file. Type "" to enter the password at the prompt. |
| <code>-g</code> | Specifies that the restore is for a Cloud Pod Architecture environment. |

For example, the following commands decrypt and import a local LDIF configuration file.

```
vdmimport -d -p mypassword -f MyEncryptedexport.LDF > MyDecryptedexport.LDF
```

```
vdmimport -f MyDecryptedexport.LDF
```

The following commands decrypt and import a Cloud Pod Architecture global LDIF configuration file.

```
vdmimport -d -p mypassword -f MyEncryptedCPAexport.LDF > MyDecryptedCPAexport.LDF
```

```
vdmimport -g -f MyDecryptedCPAexport.LDF
```

Results

After the `vdmimport` command runs, the configuration of your Connection Server instance is updated with the data from the file, and the number of records that have been successfully updated is displayed. Errors appear if some records could not be updated because your role has insufficient privileges.

Examining PColP Session Statistics with WMI

6

You can use Windows Management Instrumentation (WMI) to examine performance statistics for a PColP session by using any of the supported programming interfaces, including C#, C++, PowerShell, VBScript, VB .NET, and Windows Management Instrumentation Command-line (WMIC).

You can also use the Microsoft WMI Code Creator tool to generate VBScript, C#, and VB .NET code that accesses the PColP performance counters. For more information about WMI, WMIC, and the WMI Code Creator tool, go to <http://technet.microsoft.com/en-us/library/bb742610.aspx>.

This chapter includes the following topics:

- [Using PColP Session Statistics](#)
- [General PColP Session Statistics](#)
- [PColP Audio Statistics](#)
- [PColP Imaging Statistics](#)
- [PColP Network Statistics](#)
- [PColP USB Statistics](#)
- [Examples of Using PowerShell cmdlets to Examine PColP Statistics](#)

Using PColP Session Statistics

The WMI namespace for the PColP session statistics is `root\CIMV2`. The names of the statistics are suffixed with (Server) or (Client), according to whether the statistic is recorded on the PColP server or PColP client.

You can use Windows Performance Monitor (PerfMon) with the counters to calculate averages over a specified sampling period. You must have administrator privileges to access the performance counters remotely.

All statistics are reset to 0 when a PColP session is closed. If the WMI `SessionDurationSeconds` property is a non-zero value and stays constant, the PColP server was forcefully ended or crashed. If the `SessionDurationSeconds` property changes from a non-zero value to 0, the PColP session is closed.

To avoid a division-by-zero error, verify that the denominator in the expressions for calculating bandwidth or packet-loss percentage does not evaluate to zero.

USB statistics are recorded for zero clients, but not for thin clients or software clients.

General PCoIP Session Statistics

The WMI class name for PCoIP general session statistics is
Win32_PerfRawData_TeradiciPerf_PCoIPSessionGeneralStatistics.

Table 6-1. General Session Statistics

| WMI Property Name | Description |
|------------------------|--|
| BytesReceived | Total number of bytes of PCoIP data that have been received since the PCoIP session started. |
| BytesSent | Total number of bytes of PCoIP data that have been transmitted since the PCoIP session started. |
| PacketsReceived | Total number of packets that have been received successfully since the PCoIP session started. Not all packets are the same size. |
| PacketsSent | Total number of packets that have been transmitted since the PCoIP session started. Not all packets are the same size. |
| RXPacketsLost | Total number of received packets that have been lost since the PCoIP session started. |
| SessionDurationSeconds | Total number of seconds that the PCoIP Session has been open. |
| TXPacketsLost | Total number of transmitted packets that have been lost since the PCoIP session started. |

Calculating Bandwidth for Received PCoIP Data

To calculate the bandwidth in kilobits per second for received PCoIP data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{BytesReceived}[t_2] - \text{BytesReceived}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Calculating Bandwidth for Transmitted PCoIP Data

To calculate the bandwidth in kilobits per second for transmitted PCoIP data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{BytesSent}[t_2] - \text{BytesSent}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Calculating Packet Loss for Received PCoIP Data

To calculate the percentage of received packets that are lost, use the following formula.

$$100 / (1 + ((\text{PacketsReceived}[t_2] - \text{PacketsReceived}[t_1]) / (\text{RXPacketsLost}[t_2] - \text{RXPacketsLost}[t_1])))$$

Calculating Packet Loss for Transmitted PCoIP Data

To calculate the percentage of transmitted packets that are lost, use the following formula.

$$100 * (TXPacketsLost[t2] - TXPacketsLost[t1]) / (PacketsSent[t2] - PacketsSent[t1])$$

PCoIP Audio Statistics

The WMI class name for PCoIP audio statistics is
Win32_PerfRawData_TeradiciPerf_PCoIPSessionAudioStatistics.

Note Audio statistics do not include audio data that is carried within USB data.

Table 6-2. PCoIP Audio Statistics

| WMI Property Name | Description |
|--------------------------|--|
| AudioBytesReceived | Total number of bytes of audio data that have been received since the PCoIP session started. |
| AudioBytesSent | Total number of bytes of audio data that have been sent since the PCoIP session started. |
| AudioRXBkbitPersec | Bandwidth for ingoing audio packets averaged over the sampling period, in seconds. |
| AudioTXBkbitPersec | Bandwidth for outgoing audio packets averaged over the sampling period, in seconds. |
| AudioTXBWLimitkbitPersec | Transmission bandwidth limit in kilobits per second for outgoing audio packets. The limit is defined by a GPO setting. |

Calculating Bandwidth for Received Audio Data

To calculate the bandwidth in kilobits per second for received audio data over the time interval from time t1 to time t2, use the following formula.

$$(\text{AudioBytesReceived}[t2] - \text{AudioBytesReceived}[t1]) * 8 / (1024 * (t2 - t1))$$

Do not use AudioRXBkbitPersec for this calculation.

Calculating Bandwidth for Transmitted Audio Data

To calculate the bandwidth in kilobits per second for transmitted audio data over the time interval from time t1 to time t2, use the following formula.

$$(\text{AudioBytesSent}[t2] - \text{AudioBytesSent}[t1]) * 8 / (1024 * (t2 - t1))$$

Do not use AudioTXBkbitPersec for this calculation.

PCoIP Imaging Statistics

The WMI class name for PCoIP imaging statistics is
Win32_PerfRawData_TeradiciPerf_PCoIPSessionImagingStatistics.

Table 6-3. PCoIP Imaging Statistics

| WMI Property Name | Description |
|------------------------------------|--|
| ImagingBytesReceived | Total number of bytes of imaging data that have been received since the PCoIP session started. |
| ImagingBytesSent | Total number of bytes of imaging data that have been transmitted since the PCoIP session started. |
| ImagingDecoderCapabilitykbitPersec | Estimated processing capability of the imaging decoder in kilobits per second. This statistic is updated once per second. |
| ImagingEncodedFramesPersec | Number of imaging frames that were encoded over a one-second sampling period. |
| ImagingActiveMinimumQuality | Lowest encoded quality value on a scale from 0 to 100. This statistic is updated once per second. This counter does not correspond to the GPO setting for minimum quality. |
| ImagingRXBwkbPersec | Bandwidth for incoming imaging packets averaged over the sampling period, in seconds. |
| ImagingTXBwkbPersec | Bandwidth for outgoing imaging packets averaged over the sampling period, in seconds. |

Calculating Bandwidth for Received Imaging Data

To calculate the bandwidth in kilobits per second for received imaging data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{ImagingBytesReceived}[t_2] - \text{ImagingBytesReceived}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Do not use ImagingRXBwkbPersec for the calculation.

Calculating Bandwidth for Transmitted Imaging Data

To calculate the bandwidth in kilobits per second for transmitted imaging data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{ImagingBytesSent}[t_2] - \text{ImagingBytesSent}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Do not use ImagingTXBwkbPersec for the calculation.

PCoIP Network Statistics

The WMI class name for PCoIP network statistics is
Win32_PerfRawData_TeradiciPerf_PCoIPSessionNetworkStatistics.

Table 6-4. PCoIP Network Statistics

| WMI Property Name | Description |
|---------------------------|--|
| RoundTripLatencymms | Round trip latency in milliseconds between the PCoIP server and the PCoIP client. |
| RXBWkbitPersec | Overall bandwidth for incoming PCoIP packets averaged over the sampling period, in seconds. |
| RXBWPeakkbitPersec | Peak bandwidth in kilobits per second for incoming PCoIP packets over a one-second sampling period. |
| RXPacketLossPercent | Percentage of received packets lost during a sampling period. |
| TXBWkbitPersec | Overall bandwidth for outgoing PCoIP packets averaged over the sampling period, in seconds. |
| TXBWActiveLimitkbitPersec | Estimated available network bandwidth in kilobits per second. This statistic is updated once per second. |
| TXBWLimitkbitPersec | Transmission bandwidth limit in kilobits per second for outgoing packets. The limit is the minimum of the following values. <ul style="list-style-type: none"> ■ GPO bandwidth limit for the PCoIP client ■ GPO bandwidth limit for the PCoIP server ■ Bandwidth limit for the local network connection ■ Negotiated bandwidth limit for the Zero Client firmware based on encryption limits |
| TXPacketLossPercent | Percentage of transmitted packets lost during a sampling period. |

Calculating Bandwidth for Received Network Data

To calculate the bandwidth in kilobits per second for received data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{BytesReceived}[t_2] - \text{BytesReceived}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Do not use RXBWkbitPersec for the calculation.

Calculating Bandwidth for Transmitted Network Data

To calculate the bandwidth in kilobits per second for transmitted data over the time interval from time t_1 to time t_2 , use the following formula.

$$(\text{BytesSent}[t_2] - \text{BytesSent}[t_1]) * 8 / (1024 * (t_2 - t_1))$$

Do not use TXBWkbitPersec for the calculation.

Calculating Packet Loss for Received Network Data

To calculate the packet loss in percentage for received data over the time interval from time t_1 to time t_2 , use the following formula.

$$\text{PacketsReceived during interval} = (\text{PacketsReceived}[t_2] - \text{PacketsReceived}[t_1])$$

```

RXPacketsLost during interval = (RXPacketsLost[t2]-RXPacketsLost[t1])

RXPacketsLost % = RXPacketsLost during interval /
(RXPacketsLost during interval + PacketsReceived during interval) * 100

```

Do not use RXPacketLostPercent or RXPacketLostPercent_Base for the calculation.

Calculating Packet Loss for Transmitted Network Data

To calculate the packet loss in percentage for transmitted data over the time interval from time t1 to time t2, use the following formula.

```

PacketsSent during interval = (PacketsSent[t2]-PacketsSent[t1])

TXPacketsLost during interval = (TXPacketsLost[t2]-TXPacketsLost[t1])

TXPacketsLost % = TXPacketsLost during interval /
(TXPacketsLost during interval + PacketsSent during interval) * 100

```

Do not use TXPacketLostPercent or TXPacketLostPercent_Base for the calculation.

Use this formula to prevent the packet loss percent from becoming greater than 100 percent. This calculation is required because PacketsLost and PacketsSent are asynchronous.

PCoIP USB Statistics

The WMI class name for PCoIP USB statistics is Win32_PerfRawData_TeradiciPerf_PCoIPSessionUSBStatistics.

Table 6-5. PCoIP USB Statistics

| WMI Property Name | Description |
|-------------------|---|
| USBBytesReceived | Total number of bytes of USB data that have been received since the PCoIP session started. |
| USBBytesSent | Total number of bytes of USB data that have been transmitted since the PCoIP session started. |
| USBRXBkbitPersec | Bandwidth for incoming USB packets averaged over the sampling period, in seconds. |
| USBTXBkbitPersec | Bandwidth for outgoing USB packets averaged over the sampling period, in seconds. |

Calculating Bandwidth for Received USB Data

To calculate the bandwidth in kilobits per second for received USB data over the time interval from time t1 to time t2, use the following formula.

```

(USBBytesReceived[t2]-USBBytesReceived[t1]) * 8 / (1024 * (t2-t1))

```

Do not use USBRXBkbitPersec for the calculation.

Calculating Bandwidth for Transmitted USB Data

To calculate the bandwidth in kilobits per second for transmitted USB data over the time interval from time t1 to time t2, use the following formula.

$$(\text{USBBytesSent}[t2] - \text{USBBytesSent}[t1]) * 8 / (1024 * (t2 - t1))$$

Do not use `USBTxBWkbitPersec` for the calculation.

Examples of Using PowerShell cmdlets to Examine PCoIP Statistics

You can use PowerShell cmdlets to examine PCoIP statistics.

In the following example, the `Get-WmiObject` cmdlet retrieves the PCoIP network statistics for the client `cm-02`.

```
Get-WmiObject -namespace "root\cimv2" -computername cm-02 -class
Win32_PerfRawData_TeradiciPerf_PCoIPSessionNetworkStatistics
```

In the following example, the `Get-WmiObject` cmdlet retrieves the PCoIP general session statistics for desktop `dt-03` if any transmitted packets have been lost.

```
Get-WmiObject -namespace "root\cimv2" -computername desktop-03 -query "select * from
Win32_PerfRawData_TeradiciPerf_PCoIPSessionGeneralStatistics where TXPacketsLost > 0"
```

Setting Desktop Policies with Start Session Scripts

7

With start session scripts, you can configure specific Horizon 7 desktop settings before a desktop session starts based on information received from Horizon Client and Horizon Connection Server.

For example, you can use a start session script to configure desktop policies based on client device and user location instead of setting up multiple desktop pools that have different desktop policies. A start session script can enable mapped drives, clipboard redirection, and other desktop features for a user who has an IP address in your organization's internal domain, but disallow these features for a user who has an IP address in an external domain.

This chapter includes the following topics:

- [Obtaining Input Data for a Start Session Script](#)
- [Best Practices for Using Start Session Scripts](#)
- [Preparing a Horizon 7 Desktop to Use a Start Session Script](#)
- [Sample Start Session Scripts](#)

Obtaining Input Data for a Start Session Script

Start session scripts cannot run interactively. A start session script runs in an environment created by Horizon 7 and must obtain its input data from that environment.

Start session scripts gather input data from environment variables on the client computer. Start session environment variables have the prefix `VDM_StartSession_`. For example, the start session environment variable that contains the client system's IP address is `VDM_StartSession_IP_Address`. You must ensure that a start session script validates the existence of any environment variable that it uses.

For a list of variables similar to start session environment variables, see "Client System Information Sent to Remote Desktops" in the *Configuring Remote Desktop Features in Horizon 7* document.

Best Practices for Using Start Session Scripts

Follow these best practices when using start session scripts.

When to Use Start Session Scripts

Use start session scripts only if you need to configure desktop policies before a session starts.

As a best practice, use the Horizon Agent `CommandsToRunOnConnect` and `CommandsToRunOnReconnect` group policy settings to run command scripts after a desktop session is connected or reconnected. Running scripts within a desktop session, rather than using start session scripts, satisfies most use cases.

For more information, see “Running Commands on Horizon Desktops” in the *Configuring Remote Desktop Features in Horizon 7* document.

Managing Start Session Timeouts

Make sure your start session scripts run quickly.

If you set the `WaitScriptsOnStartSession` value in the Windows registry, your start session script must finish running before Horizon Agent can respond to the `StartSession` message that Horizon Connection Server sends. A long-running script is likely to cause the `StartSession` request to time out.

If a timeout occurs and the pool uses floating assignments, Connection Server tries to connect the user to another virtual machine. If a timeout occurs and no virtual machine is available, Connection Server rejects the user's connection request.

As a best practice, set a hard timeout for the script host operation so that a specific error can be returned if a script runs too long.

Making Start Session Scripts Accessible

The path where you configure your start session scripts must be accessible only to the SYSTEM account and to local administrators. Set the ACL for the base key to be accessible to these accounts only.

As a best practice, place start session scripts in the `View_Agent_install_path\scripts` directory, for example:

```
%ProgramFiles%\VMware\VMware View\Agent\scripts\sample.vbs
```

By default, this directory is accessible only by the SYSTEM and administrator accounts.

Preparing a Horizon 7 Desktop to Use a Start Session Script

To prepare a Horizon 7 desktop to use a start session script, you must enable the VMware View Script Host service and add entries in the Windows registry.

You must configure all Horizon 7 desktops that need to run start session scripts. Horizon 7 does not provide a mechanism to propagate registry changes, VMware View Script Host service configuration changes, and start session scripts to multiple Horizon 7 desktop virtual machines.

Enable the VMware View Script Host Service

You must enable the VMware View Script Host service on each Horizon 7 desktop virtual machine where you want Horizon 7 to run a start session script. The VMware View Script Host service is disabled by default.

When you configure the VMware View Script Host service, you can optionally specify the user account under which the start session script runs. Start session scripts run in the context of the VMware View Script Host service. By default, the VMware View Host Script service is configured to run as the SYSTEM user.

Important Start session scripts are run outside a desktop user session and not by the desktop user account. Information is sent directly from the client computer within a script running as the SYSTEM user.

Procedure

- 1 Log in to the Horizon 7 desktop virtual machine.
- 2 At the command prompt, type `services.msc` to start the Windows Services tool.
- 3 In the details pane, right-click the VMware View Script Host service entry and select **Properties**.
- 4 On the **General** tab, select **Automatic** from the **Startup type** drop-down menu.
- 5 (Optional) If you do not want the local System account to run the start session script, select the **Log On** tab, select **This account**, and type the user name and password of the account to run the start session script.
- 6 Click **OK** and exit the Windows Services tool.

Add Windows Registry Entries for a Start Session Script

You must add Windows registry entries on each Horizon desktop virtual machine where you want Horizon to run a start session script.

Prerequisites

- Verify that the path where you configured your start session scripts is accessible only to the SYSTEM account and local administrators. For more information, see [Making Start Session Scripts Accessible](#).
- Make sure your start session scripts run quickly. If you set the `WaitScriptsOnStartSession` value in the Windows registry, your start session script must finish running before Horizon Agent can respond to the `StartSession` message that Horizon Connection Server sends. For more information, see [Managing Start Session Timeouts](#).

Procedure

- 1 Log in to the Horizon desktop virtual machine.
- 2 At the command prompt, type `regedit` to start the Windows Registry Editor.
- 3 In the registry, navigate to `HKLM\SOFTWARE\VMware, Inc.\VMware VDM\ScriptEvents`.

4 Add the path to the start session script to the registry.

- a In the navigation area, right-click `ScriptEvents`, select **New > Key**, and create a key named `StartSession`.
- b In the navigation area, right-click `StartSession`, select **New > String Value**, and create a string value that identifies the start session script to run, for example, `SampleScript`.

To run more than one start session script, create a string value entry for each script under the `StartSession` key. You cannot specify the order in which these scripts run. If the scripts must run in a particular order, invoke them from a single control script.

- c In the topic area, right-click the entry for the new string value and select **Modify**.
- d In the **Value data** text box, type the command line that invokes the start session script and click **OK**.

Type the full path of the start session script and any files that it requires.

5 Add and enable a start session value in the registry.

- a Navigate to `HKLM\SOFTWARE\VMware, Inc.\VMware VDM\Agent\Configuration`.
- b (Optional) If the `Configuration` key does not exist, right-click **Agent**, select **New > Key**, and create the key.
- c In the navigation area, right-click `Configuration`, select **New > DWORD (32 bit) Value**, and type `RunScriptsOnStartSession`.

- d In the topic area, right-click the entry for the new DWORD value and select **Modify**.

- e In the **Value data** text box, type 1 to enable start session scripting and click **OK**.

You can type 0 to disable this feature. The default value is 0.

- f (Optional) To delay the `StartSession` response by Horizon Agent, add a second DWORD value to the `Configuration` key called `WaitScriptsOnStartSession`.

A `WaitScriptsOnStartSession` data value of 1 causes Horizon Agent to delay sending a `StartSession` response and fail if the scripts do not complete. A value of 0 means that Horizon Agent does not wait for the scripts to complete or check script exit codes before sending the `StartSession` response. The default value is 0.

6 Set a registry value to specify timeout values in seconds rather than minutes to prevent scripts from timing out.

Setting this timeout value in seconds enables you to configure the VMware View Script Host service timeout value in seconds. For example, if you set the VMware View Script Host service timeout to 30 seconds, you can ensure that a start session script either finishes running or times out before a Connection Server timeout occurs.

- a Navigate to `HKLM\SOFTWARE\VMware, Inc.\VMware VDM\ScriptEvents`.
- b Add a DWORD value called `TimeoutsInMinutes`.
- c Set a data value of 0.

7 (Optional) To enable the VMware View Script Host service to time out the start session script, set a timeout value.

- a Navigate to HKLM\SOFTWARE\VMware, Inc.\VMware VDM\ScriptEvents\StartSession.
- b In the topic area, right-click the Default (@) key and select **Modify**.
- c In the **Value data** text box, type the timeout value and click **OK**.

A value of 0 means that no timeout is set.

8 Exit the Registry Editor and restart the system.

Sample Start Session Scripts

These sample start session scripts illustrate how to write environment variables to a file, test the timeout functionality, and test a non-zero exit code.

The following sample Visual Basic script writes all the environment variables provided to the script into a file. You can use this sample script to see example data in your own environment. You might save this script as C:\sample.vbs.

```
Option Explicit
Dim WshShell, FSO, outFile, strOutputFile, objUserEnv, strEnv

strOutputFile = "c:\setvars.txt"

Set FSO = CreateObject("Scripting.FileSystemObject")
Set outFile = FSO.CreateTextFile(strOutputFile, TRUE)
outFile.WriteLine("Script was called at (" & Now & ")")

Set WshShell = CreateObject("WScript.Shell")
Set objUserEnv = WshShell.Environment("PROCESS")
For Each strEnv In objUserEnv
    outFile.WriteLine(strEnv)
Next

outFile.Close
```

The following sample script tests the timeout functionality.

```
Option Explicit
WScript.Sleep 60000
```

The following sample script tests a non-zero exit code.

```
Option Explicit
WScript.Quit 2
```

Using the Horizon PowerCLI Module

8

The Horizon PowerCLI Module includes Horizon PowerCLI cmdlets that you can use to perform various administration tasks on Horizon components. You can use Horizon PowerCLI with API specifications to create community-based open-source scripts.

You can install the Horizon PowerCLI module when you install VMware PowerCLI.

For more information about Horizon PowerCLI cmdlets, read the *VMware PowerCLI Cmdlets Reference* document available at <https://code.vmware.com/docs/6978/cmdlet-reference>.

For information on the API specifications to create advanced functions and scripts to use with Horizon PowerCLI, see the View API Reference at <https://code.vmware.com/apis/405/view>.

For more information on sample scripts that you can use to create your own Horizon PowerCLI scripts, visit the PowerCLI community at <https://github.com/vmware/PowerCLI-Example-Scripts>.

This chapter includes the following topics:

- [Set Up the Horizon PowerCLI Module](#)
- [Run Example Horizon PowerCLI Scripts](#)

Set Up the Horizon PowerCLI Module

You can setup the Horizon PowerCLI module with VMware PowerCLI and use the Horizon PowerCLI cmdlets to connect or disconnect from Connection Server. After you connect to the Connection Server, you can write PowerShell scripts that invoke the Horizon APIs.

Procedure

1 Install VMware PowerCLI.

Install VMware PowerCLI from the PowerShell Gallery. To install VMware PowerCLI, run the following command in the Windows PowerShell prompt:

```
Install-Module -Name VMware.PowerCLI
```

This command installs all the VMware PowerCLI modules into Windows PowerShell. The `VMware.VimAutomation.HorizonView` module is the Horizon PowerCLI module.

You can also download and install VMware PowerCLI from <https://code.vmware.com/web/dp/tool/vmware-powercli>.

For more information on how to install VMware PowerCLI, see the *VMware PowerCLI User's Guide* available at <https://code.vmware.com/web/dp/tool/vmware-powercli>.

- 2 Import the Horizon PowerCLI module named `VMware.VimAutomation.HorizonView` in the Windows PowerShell session.

Use the following command to import `VMware.VimAutomation.HorizonView` into the Windows PowerShell session:

```
Import-Module -Name VMware.VimAutomation.HorizonView
```

`VMware.VimAutomation.HorizonView` contains the `Connect-HVServer` and `Disconnect-HVServer` cmdlets that you can use to connect to a Connection Server or disconnect from a Connection Server.

- 3 Pull sample scripts from the github repository.

After you use the `Connect-HVServer` cmdlet to connect to the Horizon API service of the Connection Server, you can run PowerShell scripts that invoke the Horizon APIs. For more information about Horizon APIs, see the *View API Reference* documentation available at <https://code.vmware.com/apis/405/view>.

Example scripts for the Horizon PowerCLI module are available as the `VMware.Hv.Helper` module in the Modules section at <https://github.com/vmware/PowerCLI-Example-Scripts>.

What to do next

Use the example scripts directly or modify the scripts to suit your automation needs. Apart from example scripts, you can also develop new scripts that invoke Horizon APIs based on your needs. See, [Run Example Horizon PowerCLI Scripts](#).

Run Example Horizon PowerCLI Scripts

You can use example scripts that invoke Horizon APIs and use these scripts to perform Horizon 7 administrator tasks. You can also modify these scripts to perform administrative tasks based on your requirements.

Prerequisites

- Complete the steps to install VMware PowerCLI and set up the Horizon PowerCLI module. See, [Set Up the Horizon PowerCLI Module](#).

Procedure

- 1 Download the `VMware.Hv.Helper` module from the Modules section at <https://github.com/vmware/PowerCLI-Example-Scripts>.
- 2 Use the `$env:PSModulePath` command to find out the modules path in your Windows PowerShell session and copy the `VMware.Hv.Helper` module to that location.

- 3 Use the following command to load the VMware.Hv.Helper module into your Windows PowerShell session and start using the scripts.

```
Get-Module -ListAvailable 'Vmware.Hv.Helper' | Import-Module Get-Command -Module  
'VMware.Hv.Helper'
```