Horizon DaaS Platform 6.0 – Monitoring the Horizon DaaS Platform

Apex Release

A VMware Technical Note

This document describes basic monitoring of the Horizon DaaS environment. It also provides links to more detailed information about Horizon DaaS CIM providers and information about connectivity and ports.

March 2014

vmware

Revision History

Date	Version	Description
03/31/2014	NA	Based on 5.4.0 doc; re-branding and content changes (new CIM provider added)

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

Contents

1 Introduction 1

1.1 Critical Nodes 1
1.2 Basic System Functions 2
1.3 Web Application Monitoring 2
1.4 Port Response 2
1.5 Monitoring CIM Classes 2

2 CIM Providers on Horizon DaaS Management Nodes 3

2.1 Operating Environment CIM Providers for Horizon DaaS Nodes 3
2.1.1 Linux_OperatingSystem 3
2.1.2 Linux_EthernetPort 4
2.1.3 Linux_ComputerSystem 4
2.1.4 CIM_FileSystem 5
2.2 Application-Specific CIM Providers for Horizon DaaS Management Appliances 5
2.2.1 Service Provider Appliances 5
2.2.2 Tenant Appliances 6
2.2.3 Resource Manager Appliances 6
2.2.4 Desktone_CommonDatabase 6
2.2.5 Desktone_DatabaseService 7
2.2.6 Desktone_DatabaseReplicationService 8
2.2.7 Desktone_InstalledProduct 9
2.2.8 Desktone_ApplicationServer 9
2.2.9 Desktone_ApplicationServerStatistics 10
2.2.10 Desktone_RemoteAccessManagerStatistics 10
2.2.11 Desktone_ActiveDirectoryStatus 11
2.2.12 Desktone_HypervisorManagerStatus 11
2.2.13 Desktone_NTPService 12

3 WBEM and CIM 14

3.1 Connecting to the WBEM/CIM Server of a Horizon DaaS Management Appliance 14
3.2 Using WBEM/CIM to Monitor ESX Hosts 15

This page intentionally left blank

1 Introduction

This document describes basic monitoring of the Horizon DaaS environment. It also provides links to more detailed information about Horizon DaaS CIM providers and information about connectivity and ports.

The intent of this document is to provide information on the major items that should be monitored in the Horizon DaaS environment. At this time VMware does not have preference for the monitoring tool to be used, and the choice is left to the provider. Therefore the methods of implementation will depend upon the monitoring tool selected.

1.1 Critical Nodes

There are several nodes that are critical to proper functioning in a Horizon DaaS environment. In many cases the Horizon DaaS software is able to "self-heal". However, any impairment to these nodes should still be noted and potential action taken regardless of the Horizon DaaS software capability to "self-heal". Providing feedback on these occurrences is also important to improving the quality of the Horizon DaaS software. The nodes (whether iron or virtual) that should be actively monitored are listed below. Some of these are Horizon DaaS appliances and some are not. More details of the items that can be monitored are outlined later in this document.

Service provider nodes:

- Active Directory
- ESX hosts
- Load balancer
- NFS server
- Network routers
- Time server

Horizon DaaS nodes:

- Service Provider
- Tenant
- Resource Manager

1.2 Basic System Functions

For each of the nodes listed under "Critical Nodes", these basic functions should be monitored:

- File system space
- CPU usage
- Memory usage

The method of monitoring this information will vary depending upon the OS being monitored and the monitoring software itself. Please consult your monitoring software documentation for details.

1.3 Web Application Monitoring

Basic verification of a Horizon DaaS installation includes connecting to the following web pages (both through a load balancer, if applicable, and directly to each node):

- Desktop Portal
- Enterprise Center
- Service Center

1.4 Port Response

In addition to using ping, monitoring software can check response of specific ports - that is, if they respond to an "open socket" request. DNS and DHCP are exceptions which use UDP, and may require more intelligent monitoring.

1.5 Monitoring CIM Classes

Horizon DaaS management nodes run a variety of CIM classes that provide information about system operation. See CIM Providers on Horizon DaaS Management Nodes for more details.

2 CIM Providers on Horizon DaaS Management Nodes

This chapter describes the CIM providers to monitor for a Horizon DaaS installation. Key properties for monitoring are **highlighted** in the descriptions below.

2.1 Operating Environment CIM Providers for Horizon DaaS Nodes

These CIM providers report on the operating environment for Horizon DaaS management nodes. They should be monitored on all Horizon DaaS nodes:

- Linux_OperatingSystem
- Linux_EthernetPort
- Linux_ComputerSystem
- CIM_FileSystem

2.1.1 Linux_OperatingSystem

Description

There will only be a single instance of this class per appliance.

Properties

- FreePhysicalMemory: If this reaches 0 that is a critical fault and needs to be resolved immediately . (see the calculation below).
- **FreeVirtualMemory**: If this reaches 0 0 that is a critical fault and needs to be resolved immediately (see the calculation below).
- **HealthState**: Anything but a value of 5 indicates a problem.
- **OperationalStatus**: Anything but a value of 2 (OK) indicates a problem. However, an occasional value of 4 (stressed) may appear. If repeated samplings indicate a value other than 2, you should raise an alert.
- **TotalVirtualMemorySize**: The total amount of swap space available to the system.

Calculations

- **PercentSwapUsed**: `(100 * TotalVirtualMemorySize - FreeVirtualMemory ) / TotalVirtualMemorySize`.
- It is useful to monitor for swap space usage. Once the system begins using swap space, performance will degrade. The free memory alert should be triggered prior to the system using swap space so the use of swap should be considered a serious problem.

Mitigation

Recommendation is to warn if PercentSwapUsed > 5% and alert if PercentSwapUsed > 20%.

If the memory used reaches high levels, you should check to see if there are any memory-intensive processes that need to be restarted using top and shift-M on the node in question:

```
$ top

PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND

6816 root      20   0 2069m 389m  13m S  0.0 19.6   3:36.97 java

6634 root      20   0  755m  84m 9.8m S  0.0  4.2   1:21.70 java

...
```

If no single application appears to be the culprit, restart the node.

2.1.2 Linux_EthernetPort

Description

There will typically be two instances of this class, one for the eth0 interface (tenant or service-provider network) and one for the eth1 (management backbone) interface.

Properties

- **EnabledState**: Anything but the value 2 is a problem.
- **Status**: Anything but OK is a problem.

Mitigation

If the eth0 status is not OK use `ifconfig` to check that the interfaces are up and have an IP address. You should also be able to ping the IPv4 gateway for each node.

If the eth1 status is not OK try to connect to that appliance via `ssh` from the transit server. If this works, then the eth1 interface is OK.

2.1.3 Linux_ComputerSystem

Description

There will only be a single instance of this class per appliance.

Properties

- **EnabledState**: Anything but a value of 2 indicates an issue.

Mitigation

If EnabledState is anything but 2, attempt to ping the node, ssh to the node, and check the status of the dtService (`service dtService status`) on the node.

2.1.4 CIM_FileSystem

Description

There are several subclasses of this. (You can also check the CIM_LocalFileSystem class if you don't want to view remote file systems.) Focus on the all the Linux_Ext4FleSystem instances. In addition to the root file system, there may be others that are important to check that they are not in ReadOnly mode. Currently you should check these file systems:

- `/(root)`
- `/boot`
- `/data`
- `/tmp`
- `/usr/local`
- `/var`

Additionally on the Resource Manager nodes and the DB nodes there will be some number of Linux_NFS instances. These are remotely mounted file systems. You can choose to monitor these mounts via our appliances or an alternate mechanism based on the storage system.

Properties

- **EnabledState**: Any value other than 2 (enabled) on a remotely mounted NFS file system is cause for alarm. However, local file systems in management nodes may show up with an EnabledState of 3.
- **ReadOnly**: This value should be FALSE. A value of TRUE is cause for alarm. If the CIM_FileSystem class does not respond for a particular file system, the file system may be read-only and you should restart the node. Contact Horizon DaaS support if the restart fails.
- **Status**: Any value other than OK is cause for alarm.

  Go to the node and use mount to check that the file system is mounted. If the file system is mounted, try to create a file.
- **PercentageSpaceUsed**: Displays percent of available disk space that is used. Recommendation is to warn at 70% and then increase the alert priority in 10% increments (that is, 70, 80, 90).

Mitigation

If any of the file systems report high usage please contact Horizon DaaS support for corrective action.

2.2 Application-Specific CIM Providers for Horizon DaaS Management Appliances

Note: For non-Horizon DaaS-specific CIM provider classes, see Operating Environment CIM Providers for Horizon DaaS Nodes.

2.2.1 Service Provider Appliances

The CIM providers for Service Provider Appliances are as follows:

- Desktone_ApplicationServer
- Desktone_ApplicationServerStatistics
- Desktone_InstalledProduct

- Desktone_CommonDatabase
- Desktone_DatabaseService
- Desktone_DatabaseReplicationService
- Desktone_ActiveDirectoryStatus
- Desktone_HypervisorManagerStatus
- Desktone_NTPService

2.2.2 Tenant Appliances

The CIM providers for Tenant appliances are as follows:

- Desktone_ApplicationServer
- Desktone_ApplicationServerStatistics
- Desktone_InstalledProduct
- Desktone_CommonDatabase
- Desktone_DatabaseService
- Desktone_DatabaseReplicationService
- Desktone_RemoteAccessManagerStatistics
- Desktone_ActiveDirectoryStatus
- Desktone_NTPService

2.2.3 Resource Manager Appliances

The CIM providers for Horizon DaaS Resource Manager nodes are as follows:

- Desktone_ApplicationServer
- Desktone_ApplicationServerStatistics
- Desktone_InstalledProduct
- Desktone_NTPService

2.2.4 Desktone_CommonDatabase

Description

Describes the PostgreSQL server running on database nodes.

Properties

- InstanceID: Key to uniquely identify the instance of this class. Set to Desktone_hostName_postgreSQL.
- HomeDirectory: Home directory of the PostgreSQL service.
- DataDirectory: Data directory of the PostgreSQL service.
- DatabaseVersion: Version number of the database.

- **MaxConnections**: Maximum number of connections that the PostgreSQL server can manage concurrently. The value is extracted from the PostgreSQL configuration file from the parameter "max_connections".
- **Status**: Indicates the current status of the PostgreSQL server. OK indicates PostgreSQL is running. STOPPED indicates that the database is stopped. If the database is down (status STOPPED), any other data provided should be ignored.
- ListenAddress: The port and ip address on which postmaster process is listening for new connections.

Calculations

- **Percent maximum connections used**: You should total up the ActiveConnections used by each database instance on the server (see Desktone_DatabaseService provider) and divide by the MaxConnections from this class to determine the load on the database server. That is: 100*(Sum(ActiveConnections)/MaxConnections).

Mitigation

If the database is stopped, check the database server:

```
$ service postgresql status
```

If PostgreSQL is not running, start the service, then run the status command again:

```
$ service postgresql start
$ service postgresql status
```

If the database will not start, examine the PostgreSQL logs and contact Horizon DaaS support.

The recommendation is to warn at 80%, critical at 90% of Percent maximum connections used.

If the percent maximum connections reaches the critical level, you should examine the database server to determine which cache node or nodes is consuming a large number of connections (5-10 connections is the normal range for a cache node):

```
$ netstat -an | grep 5432
```

2.2.5 Desktone_DatabaseService

Description

Specifies the details of database instances running on a database server.

Properties

- Name: Unique identification of the service. Set to hostName_DBInstanceName. For rollback purposes, upgrades will create a db name_version instance. You do not need to monitor the database instances that have the version appended.
- **ActiveConnections**: Specifies the number of active connections to this database instance at the time of sampling/monitoring. See the calculation for Desktone_CommonDatabase using this number totaled across all database instances on a server compared to the maximum connections permitted on a single database server.

2.2.6 Desktone_DatabaseReplicationService

Description

Provides information about Fabric database instances that are replicated. This provider runs on all Fabric database servers.

Properties

- SystemCreationClassName: Name of the class used to create the database instance.
- SystemName: Name of the system on which the database instance is running. Set to host name in our case.
- CreationClassName: Name of the class used to create the database instance.
- Name: Unique identification of the service. Set to hostName_databaseInstanceName.
- NodeID: Represents the UID of the node in the context of the replication system.
- **Role**: Indication of whether the database instance is master or slave instance.
- **SyncStatus**: Synchronization status applies to the slave instance only. This property does not have any significance in case of master instance. For a slave instance, the SyncStatus value will be the number of milliseconds since the last synchronization. For example, SyncStatus = 1200 means that the last successful sync was 1.2 seconds before. Warn if the SyncStatus is more than 40 seconds old. Critical if SyncStatus is more than 2 minutes old.
- **Status**: Indicates the current status of the replication service. OK indicates the replication service is running. STOPPED indicates that the replication service is stopped. The replication service should be running for all database instances in use.

Mitigation

If replication is stopped (or if the SyncStatus is out of date), you should check that the replication daemon (slony) is running properly on the database server:

```
$ ps -ef | grep db.conf
root 1062     1  0 Sep17 ? 00:00:00 /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_edb.conf
root 1121     1  0 Sep17 ? 00:00:00 /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_fdb.conf
root 1443  1062  0 Sep17 ? 00:07:39 /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_edb.conf
root 1446  1121  0 Sep17 ? 00:06:01 /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_fdb.conf
```

There should be 2 processes for each database instance. If replication is not running properly for any of the instances, you can restart replication:

```
$ nohup /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_fdb.conf >/dev/null 2>&1 &
$ nohup /usr/local/pgsql/bin/slon -f
/usr/local/desktone/release/static/conf/slon_edb.conf >/dev/null 2>&1 &
```

2.2.7 Desktone_InstalledProduct

Description

Provides information about the Horizon DaaS software, including the version and build number.

Properties

- ProductIdentifyingNumber: Product identification. This property contains build information.
- ProductName: Product's commonly used name. Set to "Virtual-D."
- ProductVendor: Vendor's name: Desktone.
- ProductVersion: Product version information
- SystemID: Host name where the product is installed.

2.2.8 Desktone_ApplicationServer

Description

Provides information about the application server used by the Horizon DaaS software.

Properties

- Name: Name by which the application server is identified. Set to "Jboss" for Element manager and Resource manager.
- SoftwareElementID: Identifier for software element to be used in conjuction with other keys to uniquely identify the element. Set to host name on which the application server is running.
- Version: Version of the application server.
- **SoftwareElementState**: This property defines the various states of software element's life cycle. For example: Running, Executable, Deployable etc. A SoftwareElementState of 3 indicates that the application server is running.

Mitigation

If the application server is not running, go to the node in question and check the status:

```
$ service dtService status
Desktone Service is running under PID 6761
```

If the Desktone Service is not running, start it (and watch the log file):

```
$ service dtService start
```

- TargetOperatingSystem: Specifies the node's operating system environment. Set to 36 (LINUX).

2.2.9 Desktone_ApplicationServerStatistics

Description

There will be a single instance of this class for all of the application appliances (that is, this will not be present in DB appliances).

Properties

These properties report on operations of the JVM (Java virtual machine) used for the Horizon DaaS application.

- InstanceID: Key to uniquely identify the instance of this class. Set to DesktonehostName_Jboss.
- **ThreadCount**: Total number of threads running during the monitoring sample.
- ThreadGroupCount: Total number of thread groups that exist during the sample time.
- **HeapSize**: Current size of heap memory
- **MaxHeapSize**: Maximum heap memory allowed on the application server.
- Uptime: The length of time the application server has been running in milliseconds.

Calculations

- **Heap size used**: 100*HeapSize/MaxHeapSize. Recommendation is to warn at 85% and then increase the alert priority in 5% increments (that is, 90, 95, 100).

Mitigation

At 85%, schedule a restart of the dtService. At 90% or higher, restart the dtService immediately:

```
$ service dtService restart
```

If the heap memory used increases to high levels often (more than once per week), you should analyze your environment in concert with Horizon DaaS support.

2.2.10 Desktone_RemoteAccessManagerStatistics

Description

Reports on the status of the dtRAM service on a Tenant node. Note that this CIM provider actually runs on a Tenant node, not on the dtRAM itself.

Properties

- InstanceID: Key to uniquely identify the instance of this class. Set to `Desktone_hostname_RemoteAccessManager`.
- **EnabledStatus**: Reports if the dtRAM is enabled or not on this node. Possible values are TRUE or FALSE.
- **AvailableStatus**: Reports the availabiliy of dtRAM service is enabled. Possible value are TRUE or FALSE.

Mitigation

If the dtRAM is enabled, but the AvailableStatus is anything other than TRUE, you should check the dtramd status from the dtRAM machine:

```
dtRAM01# /usr/local/etc/rc.d/dtramd status
dtramd is running as pid 1369
If dtRAM is not running, restart it:
dtRAM01# /usr/local/etc/rc.d/dtramd restart
```

2.2.11 Desktone_ActiveDirectoryStatus

Description

ActiveDirectoryStatus provider is derived from CIM_LogicalElement, and it provides information and status of domain controllers which are added in Horizon DaaS Platform. This provider runs on Service Provider and Tenant appliances.

Properties

- CSCreationClassName [key]: Name of the class used to create the database instance.
- SystemName [key]: Name of the system on which the provider instance is running. Set to host name in our case.
- CreationClassName [key]: Name of the class used to create the provider instance.
- DcAddress [key]: describes the unique domain controller address.
- DomainName: describes the domain name associated with domain controller.
- LdapUri: describes the LDAP Url of current domain controller
- **ResponseTime**: describes the response time in milliseconds for LDAP query from Horizon DaaS appliance. The administrator should monitor this property and alert as required if it is preferred domain controller. Example: 0-15 seconds response time is OK, 15-30 sec is WARN, and >30 secs is CRITICAL.
- LastUpdated: describes the last updated time for this controller
- **IsPreferred**: Indicates whether the domain controller is preferred domain controller or not in Horizon DaaS Platform
- **CommunicationStatus** [derived]: indicates the ability of the Horizon DaaS Platform to communicate with domain controller. 2 – OK, 4 – Lost Communication
- **OperationalStatus** [derived]: indicates the status of domain controller in Horizon DaaS Platform . 2-OK, 13 – Lost communication.
- **Status** [derived, **deprecated**]: indicates the current state of domain controller in Horizon DaaS Platform (OK, Lost Comm)

Mitigation

Make sure that preferred domain controllers are up and running, and verify the latency between appliance and domain controller if response time is high.

Check the required communication ports are open between domain controller and Horizon DaaS appliances.

2.2.12 Desktone_HypervisorManagerStatus

Description

HypervisorManagerStatus provider is derived from CIM_LogicalElement, and it provides information and status of Hypervisor Managers in Horizon DaaS Platform. The Hypervisor Manager is a Horizon DaaS entity which manages the hypervisor hosts. This provider runs on Service Provider appliances only.

Properties

- CSCreationClassName [key]: Name of the class used to create the database instance.
- SystemName [key]: Name of the system on which the provider instance is running. Set to host name in our case.

- CreationClassName [key]: Name of the class used to create the provider instance.
- HostAddress [key]: describes the hypervisor manager host address and version. It is an address of vCenter, ESX, or vCloud host.
- Type: describes the type of hypervisor manager whether it is vCenter/ESX/ vCloud and its product version. Ex: "ESX, 5.1.0"
- **CommunicationStatus** [derived]: indicates the ability of the Horizon DaaS Hypervisor Manager to communicate with Hypervisor Host. 2 – OK, 4 – Lost Communication
- **OperationalStatus** [derived]: indicates the current status of the Horizon DaaS Hypervisor Manager in Horizon DaaS Platform. 2- OK, 13 – Lost communication,
- **Status** [derived, **deprecated**]: indicates the current status of Horizon DaaS Hypervisor Manager in Horizon DaaS Platform (OK, Lost Comm)

Mitigation

- Make sure that discovered host is assigned to resource manager.
- Make sure that Hypervisor host is running and reachable from Service Provider appliance.
- Please verify if there any API compatibility errors in Service Provider or Resource Manager desktone logs.
- Check the required communication ports are open between Horizon DaaS appliances and hypervisor hosts.

2.2.13 Desktone_NTPService

Description

NTPService provider is derived from CIM_Service, and it provides information about NTP daemon which runs on Horizon DaaS appliance. It also reports time synchronization status. The NTPService provider is available on all Horizon DaaS appliances except dtRAM appliance.

Properties

- CSCreationClassName [key, dervied]: Name of the class used to create the database instance.
- SystemName [key, dervied]: Name of the system on which the NTP daemon is running. Set to host name in our case.
- CreationClassName [key, derived]: Name of the class used to create the provider instance.
- name [key, derived]: describes the name of the service. It is “NTPD” in our case.
- **Started**[derived]: Started is a Boolean that indicates whether the NTP Service has been started (TRUE), or stopped (FALSE).
- ServerAddresses: describes the NTP server addresses configured in /etc/ntp.conf. It is a comma separated string of addresses.
- PrimarySource: describes the current NTP source in use for time synchronization.
- **SyncState**: indicates NTP synchronization status. TRUE, if NTP is in sync with time source, otherwise FALSE. The SyncState depends on jitter, condition of peer and reach status.
- **Jitter**: describes the jitter value in milliseconds of selected time source. If there is any problem to get the jitter or no primary source is selected by NTP, it returns 60000 milliseconds in order to alert. Providers marks SyncState property to FALSE if jitter is higher than 1000 milliseconds.
- **OperationalStatus**[derived]: indicates the current status of NTP daemon and time synchronization.

OperationalStatus=2 (OK) -> NTP time is in sync(SyncState =TRUE) and all time sources configured are reachable.

OperationalStatus=5 (Predictive Failure) indicates NTP time is in sync, but one or more configured time servers are not reachable or rejected.

OperationalStatus=6 (ERROR) time source is not in sync or NTP service is down

- **StatusDescriptions** [derived]: describes the OperationalStatus in detail which helps administrator troubleshoot NTP time synchronization.

Mitigation

Make sure that NTP daemon is running. Troubleshoot NTP for time synchronization.

Make sure that there is connectivity between the service provider nodes and the ntp source.

3 WBEM and CIM

The Horizon DaaS management appliances allow monitoring via the standard WBEM (web-based enterprise management) CIM (common information model) interface. You can use any monitoring tool capable of understanding the CIM data model (for example, Tivoli).

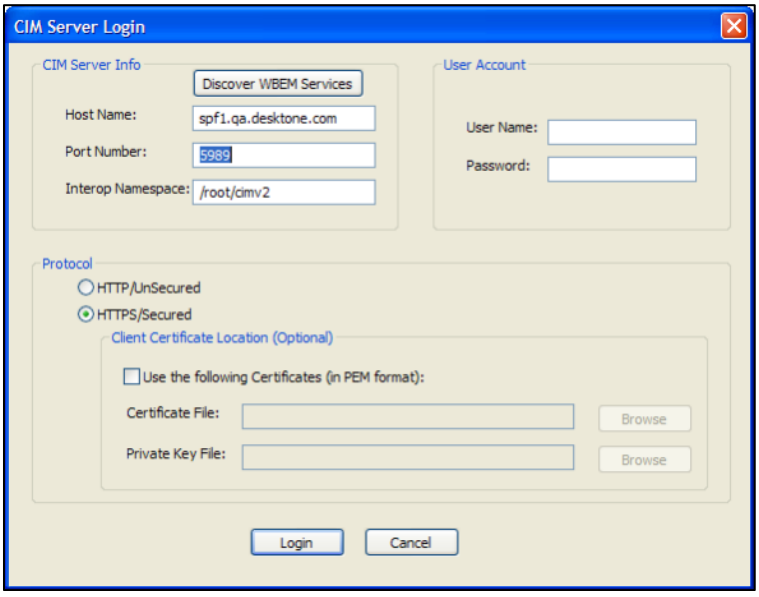
3.1 Connecting to the WBEM/CIM Server of a Horizon DaaS Management Appliance

To login to the WBEM/CIM interface of one of the Horizon DaaS management appliances, you need the following information:

- Host name: The DNS name or IP address of the management appliance
- Port number: 5989
- Namespace: /root/cimv2

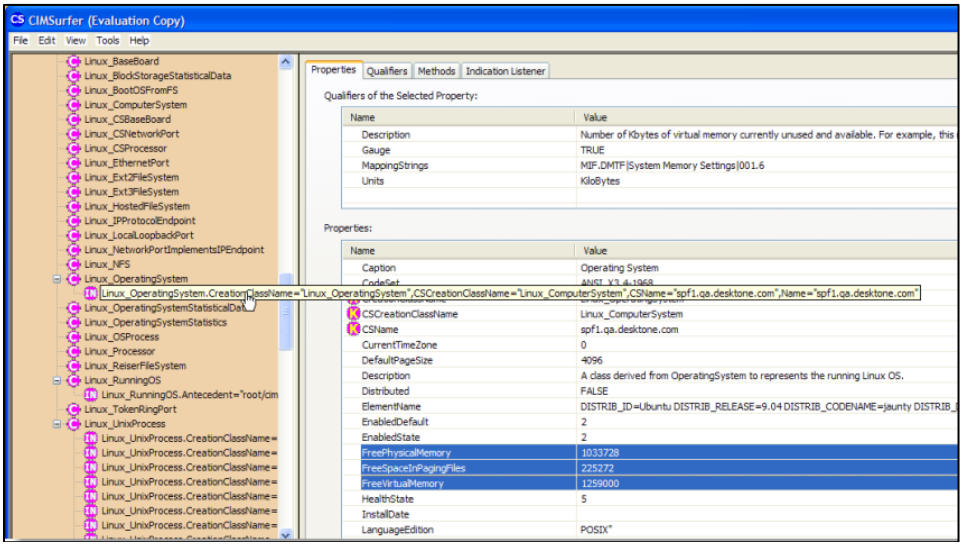
For example, CIMSurfer is a basic browser for CIM information. In practice, you would use a different tool, such as Tivoli, that automatically monitors a number of management appliances and provide alerts based on conditions in the CIM classes of interest. This example also accesses the CIM server without a certificate.

Figure 1: CIM Server Login



The following figure shows the type of information available from the Linux_OperatingSystem class. Using the properties, you can determine the amount or percentage of free memory that is still available.

Figure 2: CIM Classes

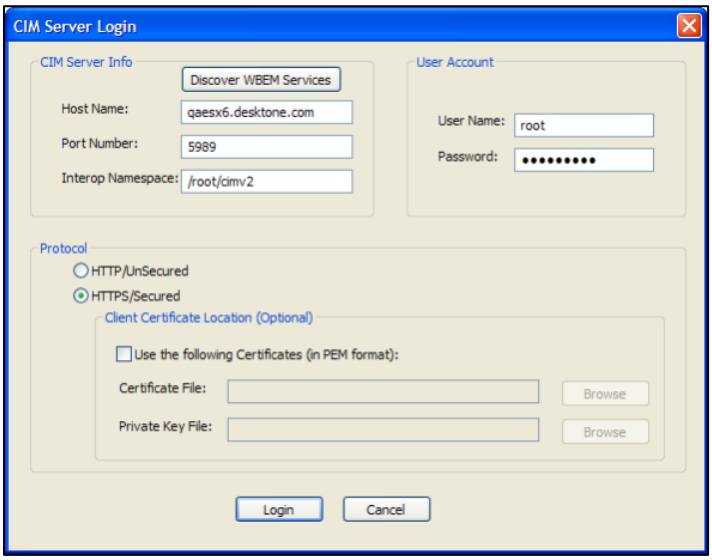


3.2 Using WBEM/CIM to Monitor ESX Hosts

Since the ESX hosts also expose a WBEM/CIM interface, you can also monitor the ESX hosts. Logging in to the ESX is the same as logging into a management appliance, except that user credentials are required.

The login credentials should be the same ones you use to access the host using the VI client:

Figure 3: Logging in to the ESX



The following figure illustrates how the ESX hosts will expose different classes than the Horizon DaaS management appliances. We recommend that you consult VMware to determine which classes are important to monitor.

Figure 4: Classes Exposed by ESX Host