Performance Sentry VM Installation and User Guide

Copyright Information

Table of Contents

System Requirements

Thank you for your interest in Performance Sentry VM! Below is a list of the system requirements for Performance Sentry VM. If you have any questions regarding these requirements, please visit the Demand Technology Software website http://demandtech.com or send email to support@demandtech.com

- VMware vCenter Server 4.0 or later
- VMware ESX Host 4.0 or later
- Windows Server 2008 or later, Windows 7 or later
- Microsoft .NET Framework 3.5

For those customers running the Performance Sentry Collection Service who want to collect the VMware performance data supplied by Performance Sentry VM you will need, at minimum, version 3.1.4.4 or later. However, version 4.1.1.10 or later is recommended for more robust support for VMware performance data.

Operational Overview

Performance Sentry VM works as a "provider" of VMware performance metrics in Windows. It retrieves performance metrics directly from the ESX or ESXi host for both the host and all of the guests (virtual machines) running under that same host.

Those metrics are then formatted to the Windows object/counter performance data structure and made available for consumption by any Windows management software such as Windows Perfmon and Microsoft System Center Operations Manager (SCOM). Of course we would prefer that you use Demand Technology Software's Performance Sentry for Windows, which offers many advantages over other frameworks.

View the document **Performance Sentry VM Objects** for a detailed description of the latest supported VMware performance objects and counters.

Performance Sentry VM runs as a Windows service that runs continuously in the background, starting and stopping with Windows startup and shutdown. It writes messages to its own application text log and to the system level Application Event Log. By default, only error, warning and a small set of information messages are written to the Application Event Log. Logging can be controlled by parameters in the file named **SentryVMParameters.xml.** See the section in this document titled **Controlling Performance Sentry VM Messaging** for additional information.

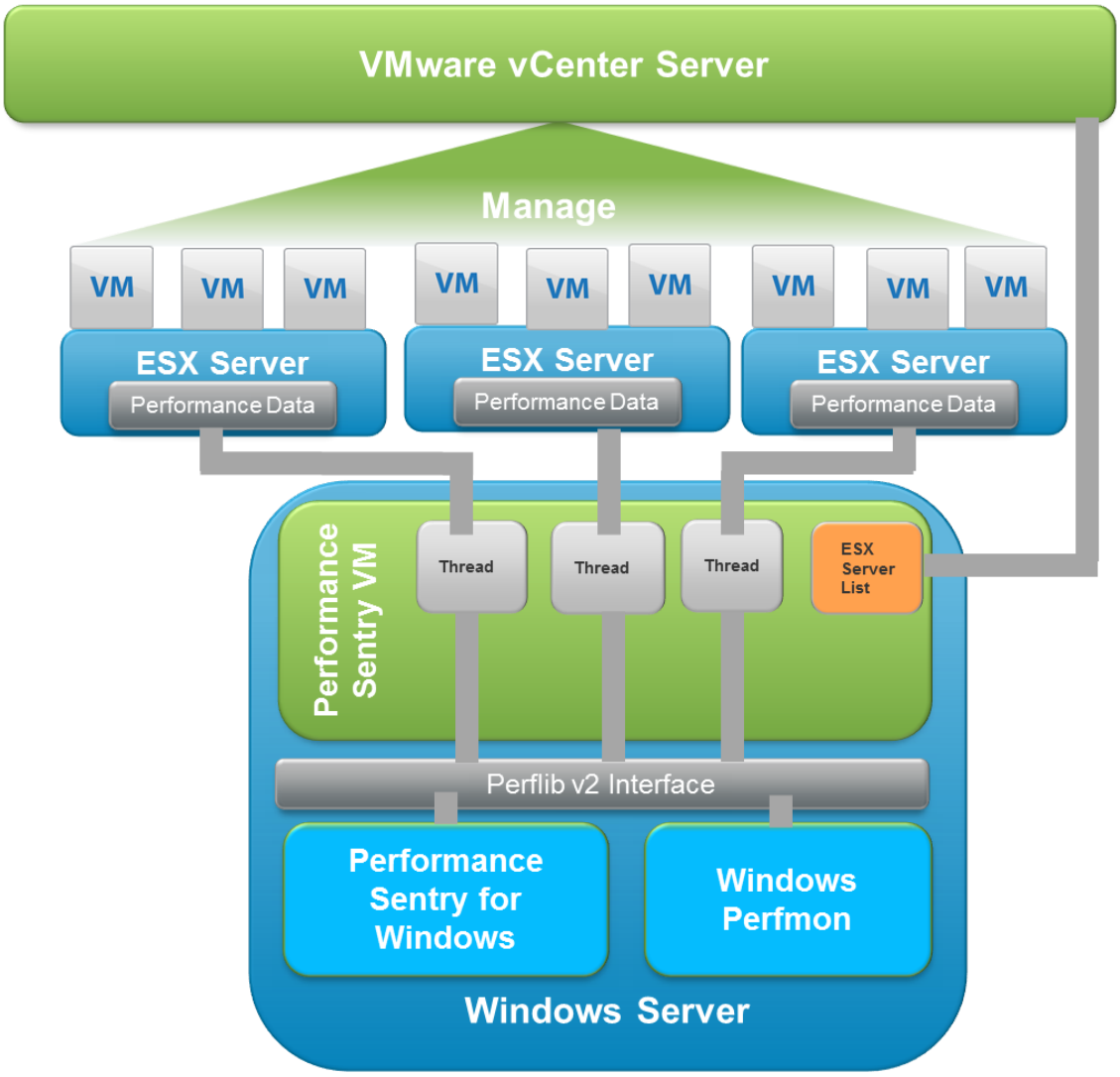
At initialization, Performance Sentry VM connects to the target VMware vCenter Server using stored encrypted credentials to query the names of the ESX servers under vCenter control. Performance Sentry VM then directly connects to each ESX server using stored encrypted credentials and queries *all* of the available host and guest performance metrics.

VMware performance data is obtained directly from the ESX Host system from three Providers named Host System, Virtual Machine and Resource Pool.[1] The VMware data is organized in groups (e.g. CPU, Memory, Disk) and each group has a set of possible counters. For example, there are Memory counters in the Memory group in each of the Providers. For more information on these providers and performance and tuning whitepapers, visit VMware's website: http:///www.vmware.com

[1] In the VMware context "A performance provider is any entity on the system that generates utilization or consumption information." Reference: vSphere Web Services SDK Programming Guide, available at VMware.com.

Performance Sentry VM collects data from these providers and acts in turn as a Provider to the Microsoft Windows PerflibV2 facility.

The values from the VMware Providers are not changed or recalculated in any way and are simply formatted to the Windows object/counter performance data structure and made available to any Windows performance management framework. A diagram of this architecture appears below:



Performance Sentry VM connects to the VMware vCenter Server to retrieve a list of ESX servers under its management. Sentry VM then connects directly to each ESX server using separate collection threads. Host and Virtual Machine (guest) performance data is retrieved from each ESX server at 20 second intervals (the ESX Server refresh rate). The data is made available to the Windows Perflib v2 interface, which then allows Performance Sentry for Windows, Perfmon, or any Windows performance management API consumer to gather the VMware performance metrics.

A host and guest machine filtering capability has been implemented to control the volume of data presented to consumers. Filtering is described in the section titled **Limit Monitoring to Specific Machines** in this document.

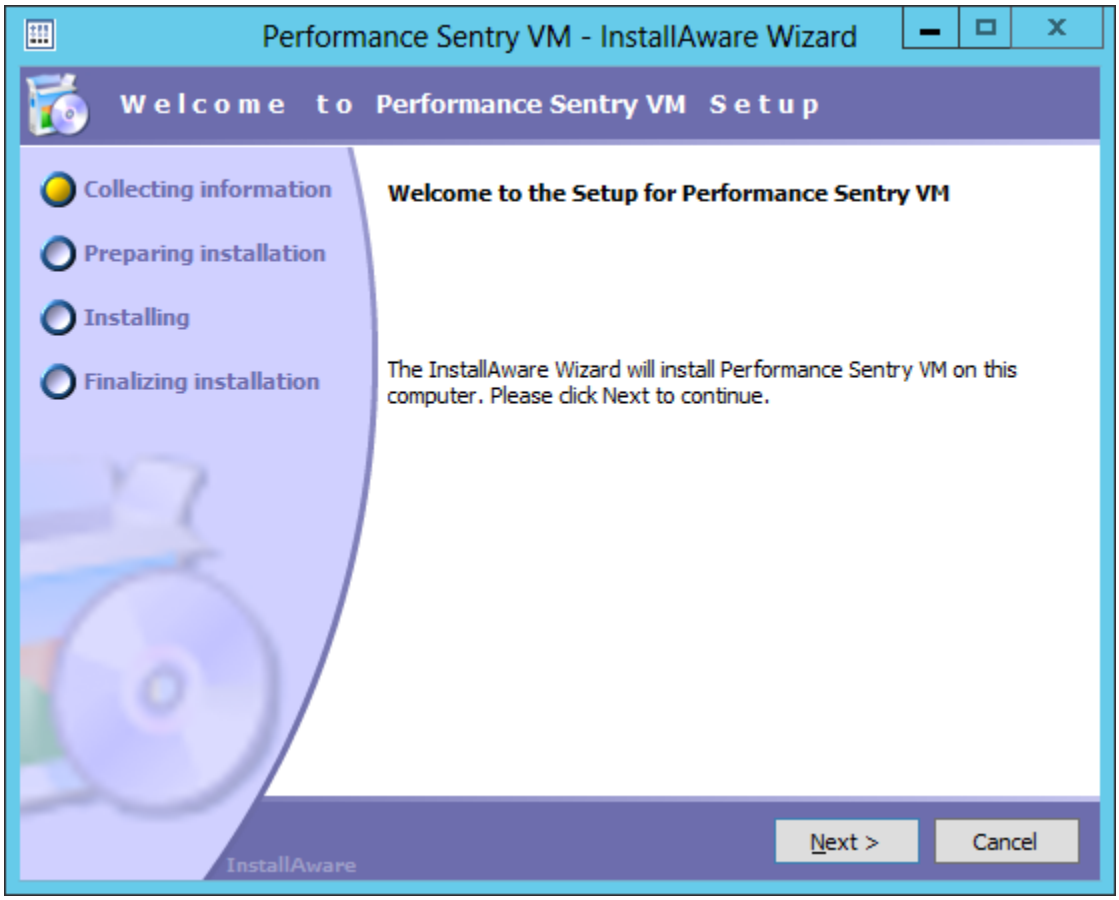
For customers using Performance Sentry for Windows, a sample Data Collection Set (DCS) definition file (with a file extension of .dcsx), is provided to gather VMware performance data objects, along and a small, default set of Windows performance data objects. The file name is 'VMware Collection Set.dcsx' and can be viewed in the Performance Sentry Administration application or found in the Performance Sentry Collection Service installation folder. For an existing data collection set definition in use at your installation, VMware objects and counters can be added to that collection set using the Performance Sentry Administration DCS Editor facility.

Installation Instructions

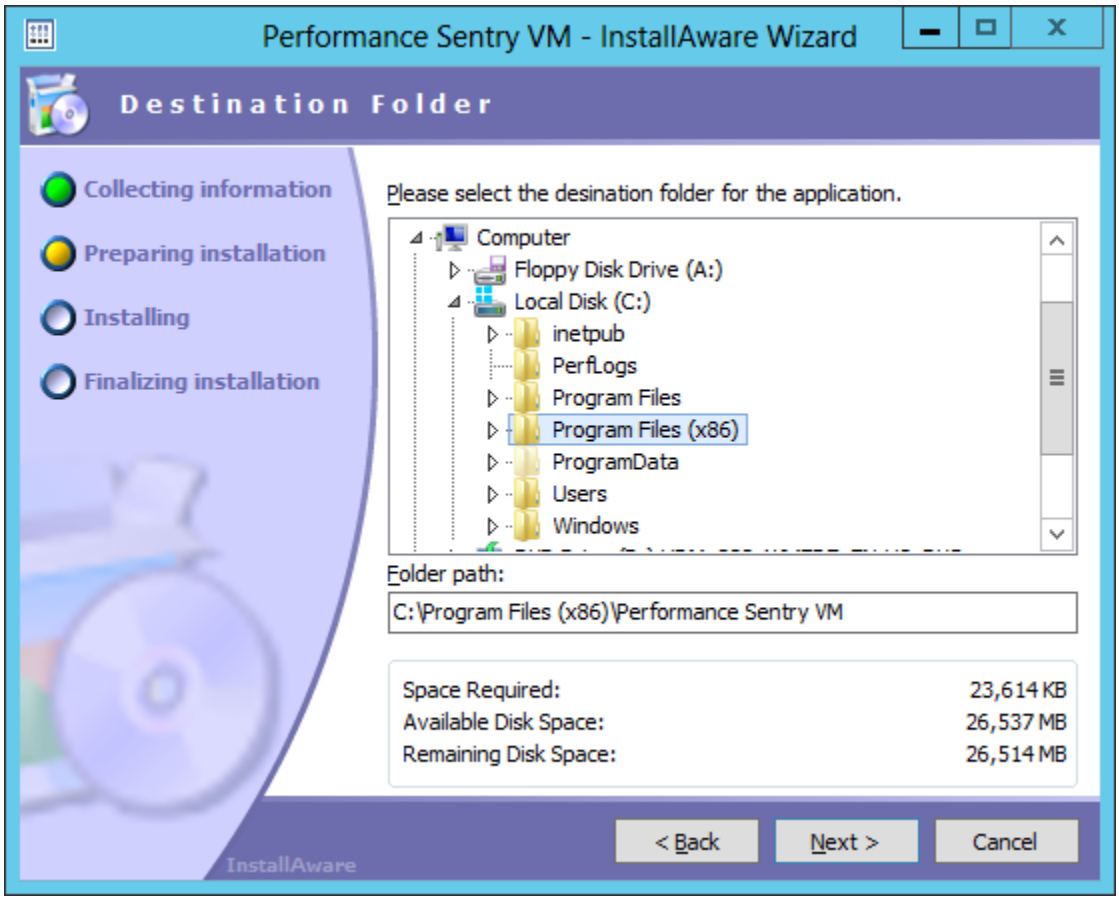
Before launching the install package, you will need a VMware vCenter User Name and Password with read access and the IP Address of the VMware vCenter Server to which Performance Sentry VM will connect.

You will also need a User Name and Password to access each ESX host managed by the vCenter Server. This User Name requires a minimum of read access for retrieving VMware performance data from each ESX server.

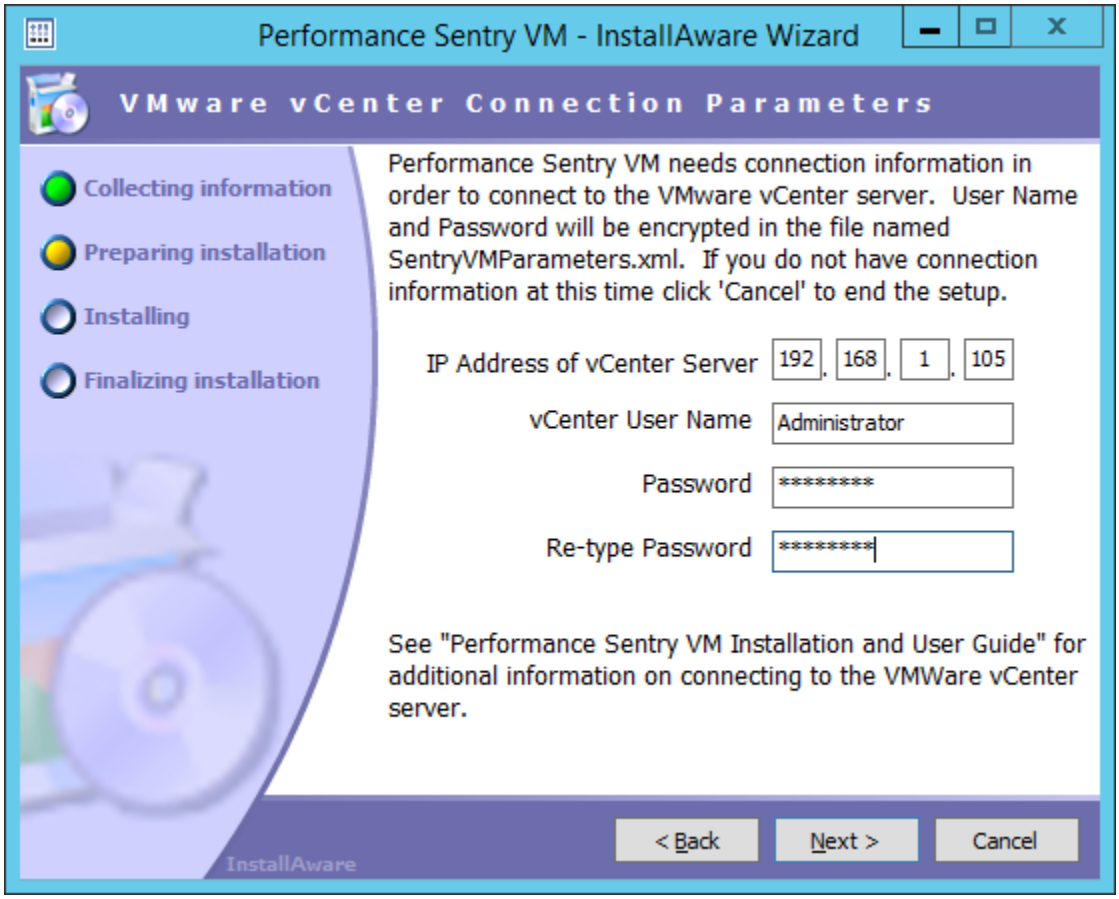
Step 1 – Launch the setup by running **Performance Sentry VM.exe**



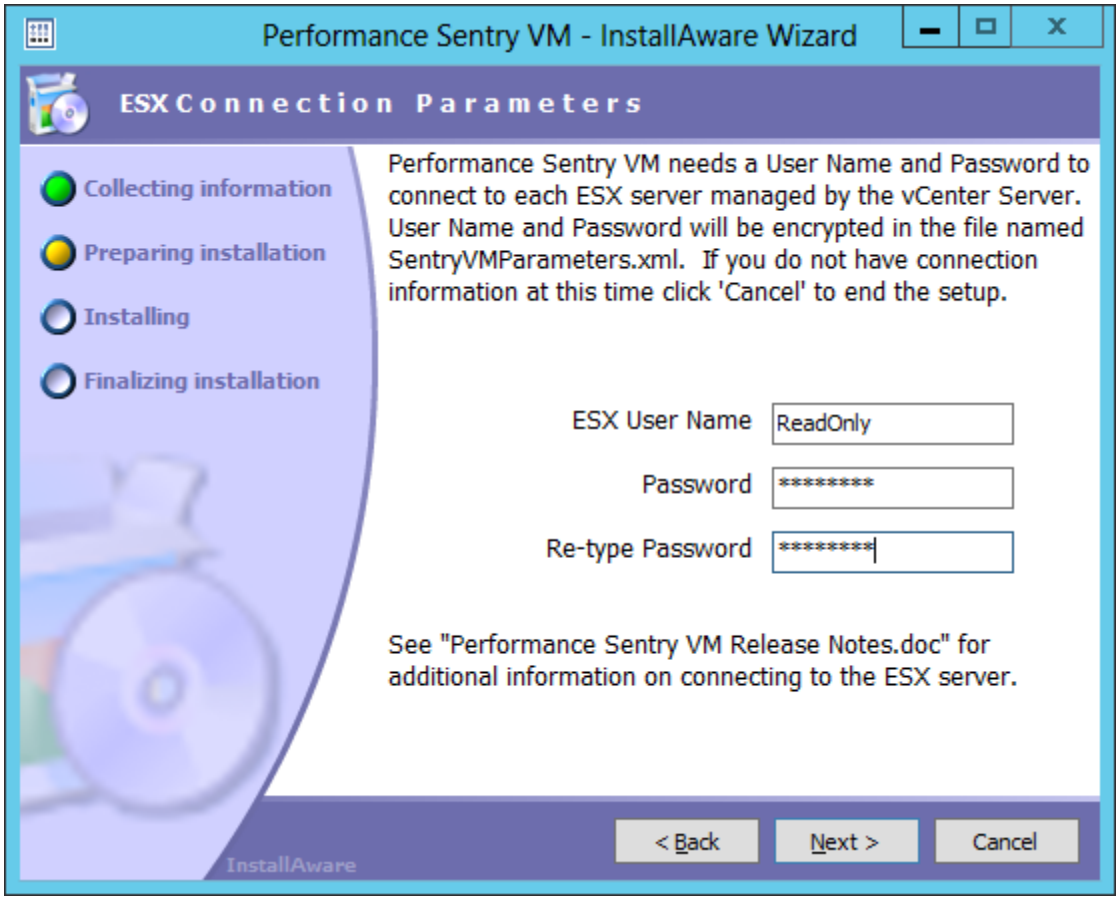
Step 2 – Select the destination folder. The default destination folder is C:\Program Files (x86)\Performance Sentry VM



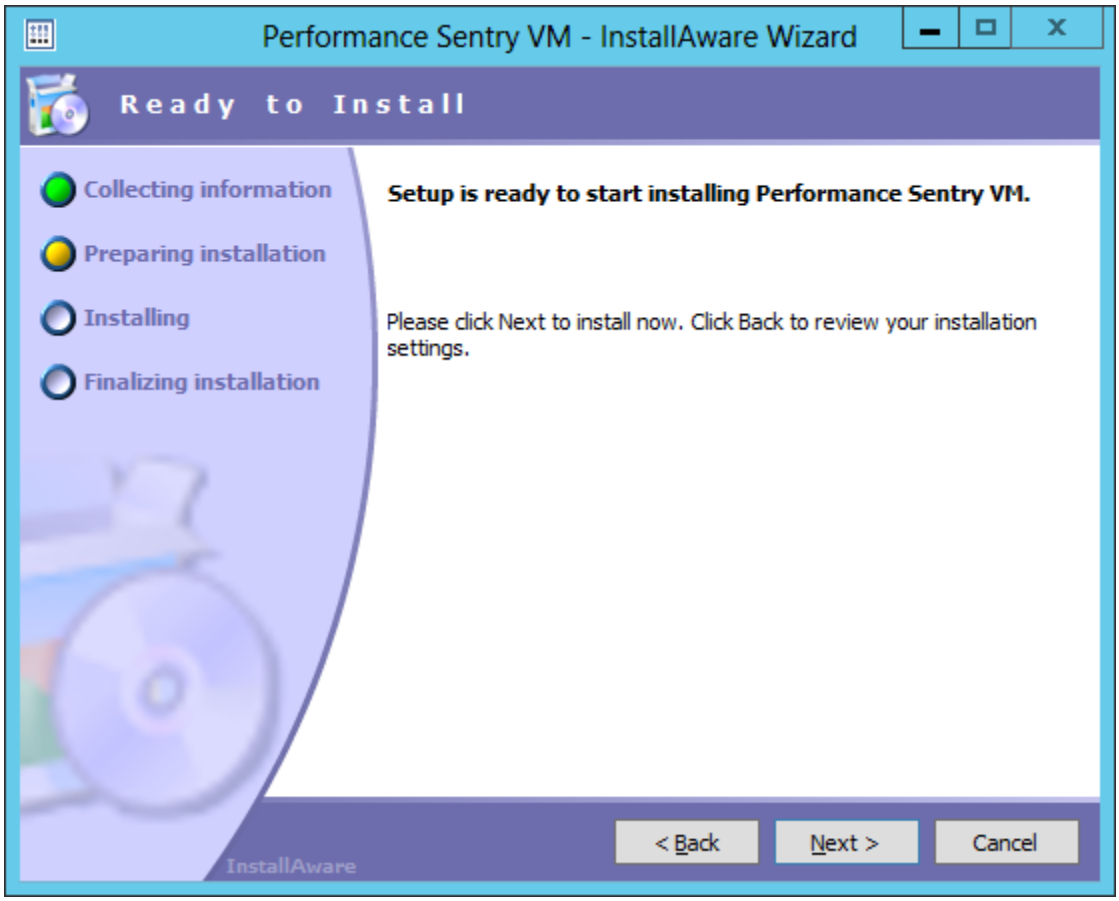
Step 3 – Enter the IP Address of the vCenter Server along with the User Name and Password with read access to vCenter. These credentials will be encrypted and stored in the Performance Sentry VM parameter file.



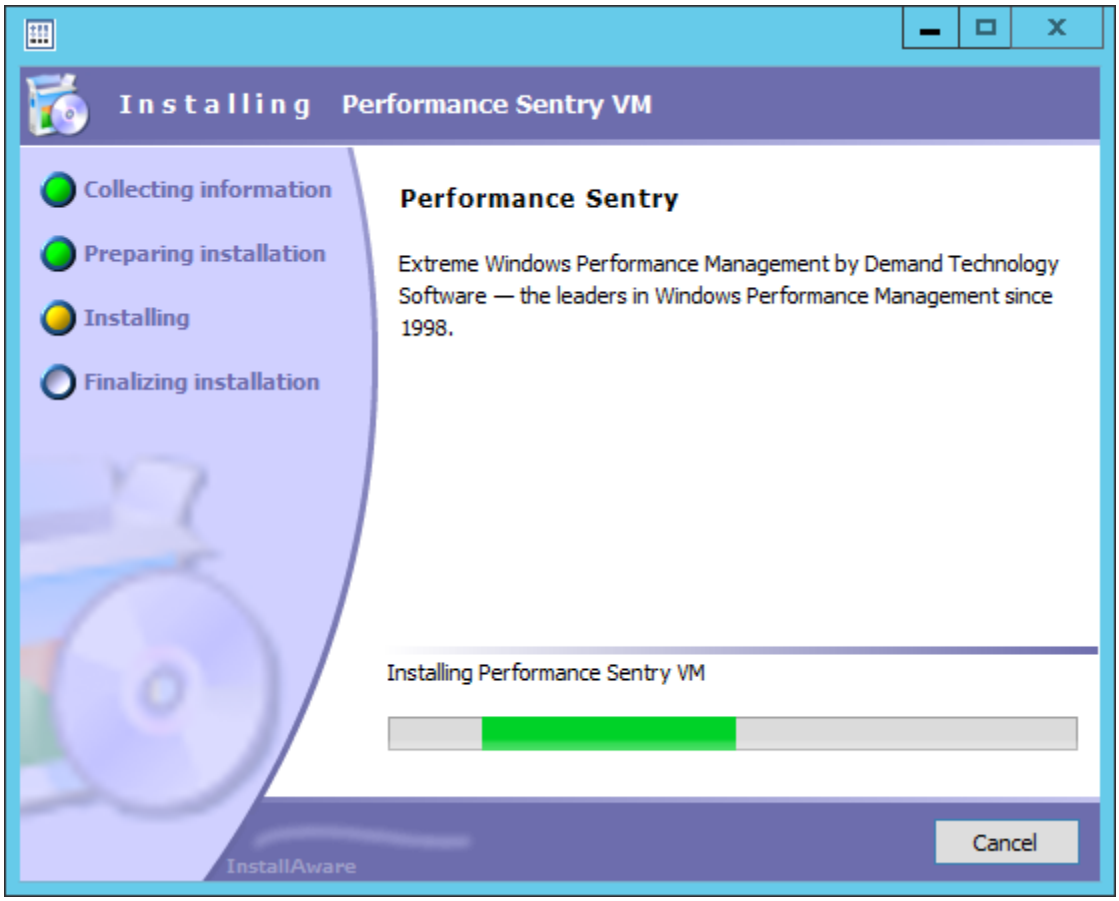
Step 4 – Enter the ESX Server User Name and Password. These credentials will be used to directly connect to every ESX server managed by the vCenter Server indicated in the previous screen. These encrypted credentials will be stored in the Performance Sentry VM parameter file.



Step 5 – Setup is now ready to begin installation. Click **Next** to begin installing the necessary files and settings.



The **Installing Performance Sentry VM** dialog box will indicate the installation progress:



When the setup is finished you will see the following completion dialog box:

Check the **Run Performance Monitor now checkbox** (if it is not already checked) and click **Finish** to launch Microsoft Performance and Reliability (perfmon).

With Performance Sentry VM running you will have access to over 350 counters from 24 different VMware 'host' and 'guest' objects. For a complete listing of those objects and counters please see the document **Sentry VM Objects** that is included in your download and found in the installation folder.

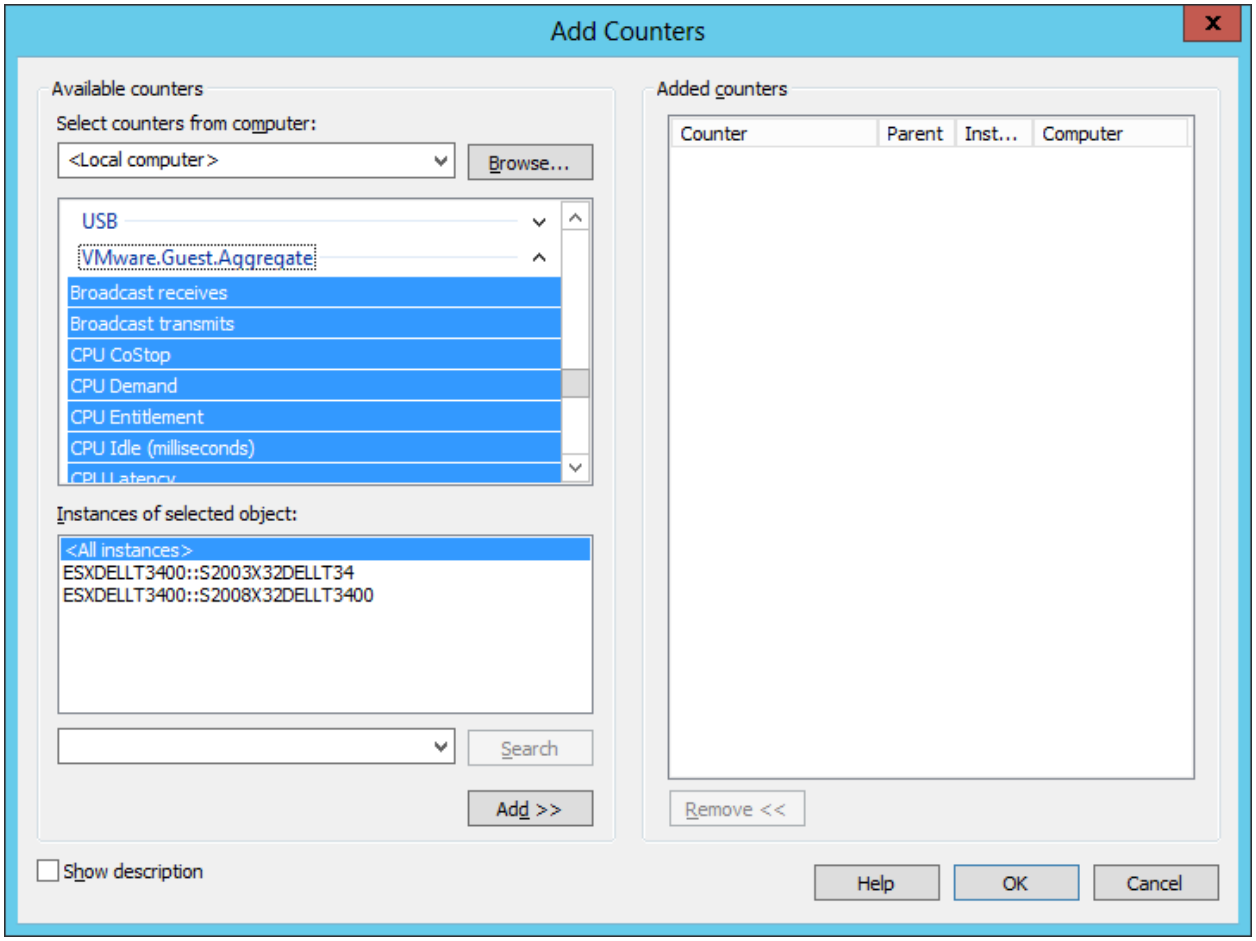
You can read more about displaying Performance Sentry VM performance data in the next section titled **Viewing Performance Sentry VM Data through Perfmon**

Because Performance Sentry VM supports the standard Microsoft Windows performance monitoring interface, VMware performance data can be viewed and collected by products like **Microsoft System Center Operations Manager (SCOM), perfmon** and of course Demand Technology Software's **Performance Sentry Collection Service,** as well as most other Windows performance monitoring tools. Additionally, Demand Technology Software provides a Microsoft System Center Operations Manager Management Pack (MP). See the

section titled **Viewing Performance Sentry VM Data through SCOM** which follows after the next section.

Viewing Performance Sentry VM Data through Perfmon

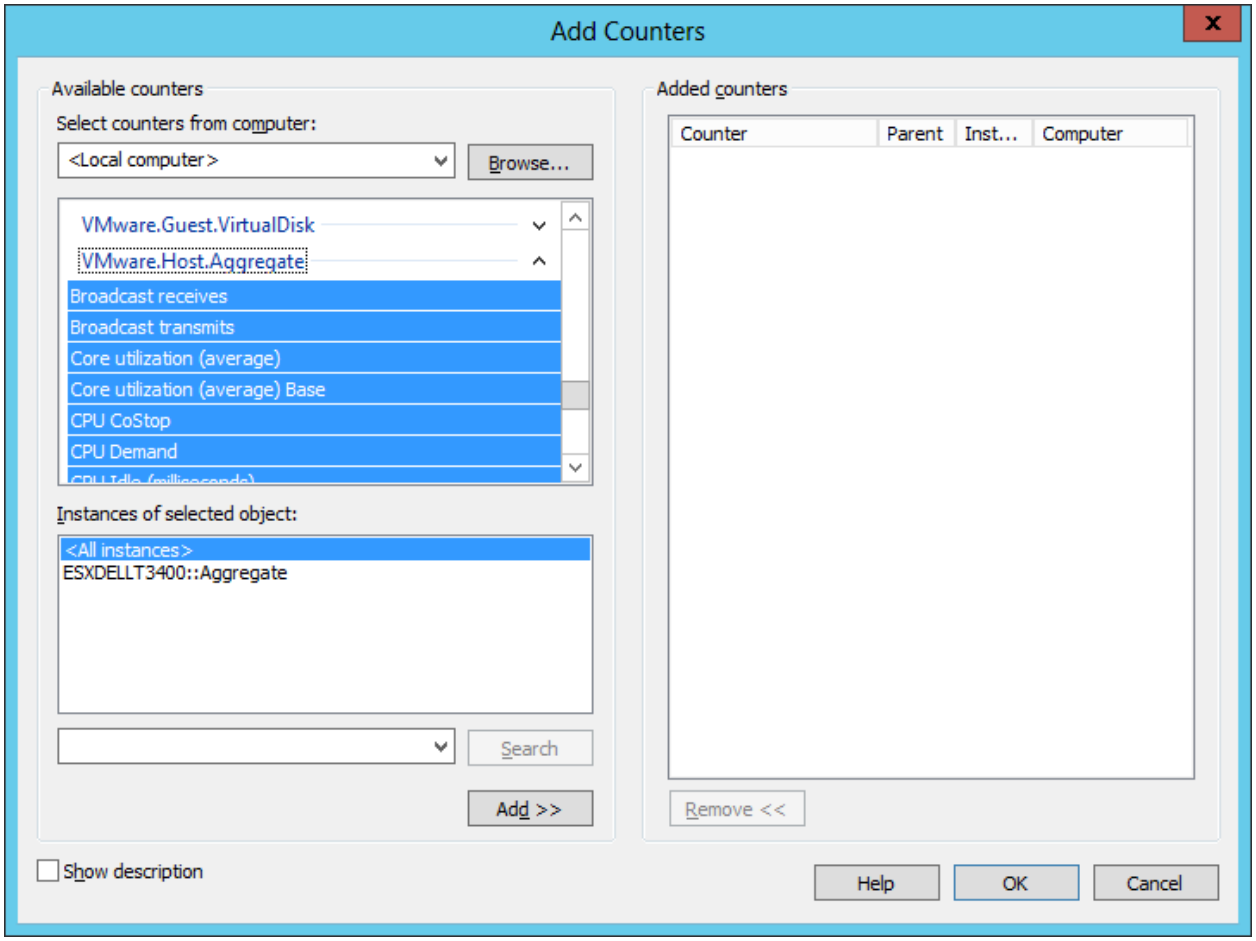
If you wish to view the VMware performance data provided by Performance Sentry VM in perfmon you can add the VMware objects and counters through the **Perfmon Add Counters** function. Here is a screenshot of the perfmon **Add Counters** dialog box showing the **VMware.Guest.Aggregate** object and some of its associated counters:



VMware.Guest.Aggregate object in the 'Add Counters' dialog box in perfmon

You will notice that the Instances of the object are in the form of HOSTNAME::GUESTNAME. For more information on the Object/Counter format, see the document **Sentry VM Objects** under **Instance Name conventions**.

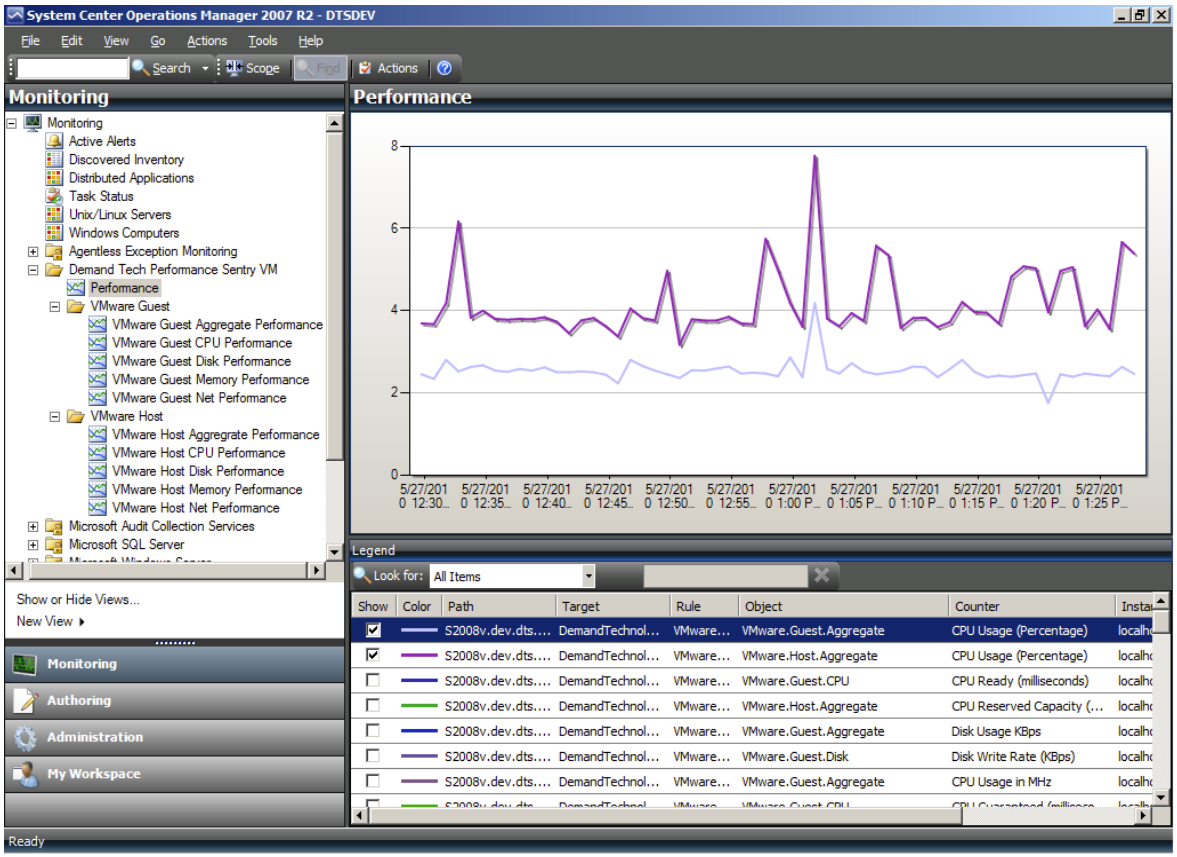
Below is another screenshot of the perfmon **Add Counters** window showing the **VMware.Host.Aggregate** object and some of its associated counters:



VMware.Host.Aggregate object in the 'Add Counters' dialog box in perfmon

Viewing Performance Sentry VM Data through SCOM

Demand Technology Software has a free Performance Sentry VM Management Pack (MP) to monitor important VMware performance metrics made available through Performance Sentry VM.



VMware Guest vs. Host Aggregate CPU viewed through SCOM

Limit Monitoring to Specific Machines

Version 2.0 of Performance Sentry VM processes VMware performance data from all ESX servers and VM guests running under the control of one VMware vCenter Server. Therefore it may be necessary to limit processing of ESX servers and VM guests, using the **SentryVMMachineList.xml** file.

Each time that Performance Sentry VM is started, it creates a file named **SentryVMMachineList.SAMPLE.xml**. This file is a list of all ESX servers and all virtual machines running under the control of the VMware vCenter Server to which Performance Sentry VM is connected. This is a sample file that can be edited and saved as **SentryVMMachineList.xml** using notepad, or an XML editor, to specify only the ESX hosts and virtual machines for which VMware performance data is to be processed.

Only those machines in the **include** section of the SentryVMMachineList.xml file will be processed by Performance Sentry VM. Simply edit the file and remove the names of the unnecessary hosts and unnecessary guests. You may also simplify the host and guest specification by using wildcards (*) for trailing characters. Only a single trailing wildcard can be specified.

For example:

```
<hosts>
  <host>PRODESX40A</host>
  <host>PRODESX40B</host>
  <host>PRODESX40C</host>
  <host>PRODESX55A</host>
  <host>PRODESX55B</host>
  <host>PRODESX55C</host>
</hosts>
```

Can be represented by:

```
<hosts>
  <host>PRODESX40*</host>
  <host>PRODESX55*</host>
</hosts>
```

And similarly with guests:

```
<guests>
  <guest>PRODS2008*</guest>
  <guest>PRODS2012*</guest>
</guests>
```

If there are host names specified in the **hosts** section, then only the guests (or virtual machines) running under those hosts will be processed. You can further limit the guests by specifying them in the **guests** section, or you can process

performance data for all guests running under the specified hosts by using the wildcard:

```
<guests>
    <guest>*</guest>
 </guests>
```

Similarly, you can limit collection to only specific guests in the entire vCenter by using a wildcard in the **host** section and listing the guests in the **guests** section.

For example:

```
  <hosts>
    <host>*</host>
  </hosts>
  <guests>
    <guest>PRODS2008*</guest>
    <guest>PRODS2012*</guest>
  </guests>
```

Performance Sentry VM can run on any Windows Server 2008 or newer, or Windows 7, or newer computer having network access to a VMware vCenter Server. Performance Sentry VM can run inside a Windows virtual machine within vCenter, or on a Windows machine external to vCenter.

There is no need to install and run Performance Sentry VM on more than a single guest within each vCenter because Sentry VM collects VMware performance data from all hosts and all guests running under that vCenter. The typical enterprise would have a single Windows virtual machine running Performance Sentry VM for each vCenter server.

However, some customers have found it advantageous to install multiple instances of Performance Sentry VM within vCenter. Each instance is then used to monitor all ESX hosts assigned to a specific cluster. This is perfectly acceptable as Performance Sentry is licensed by the total number of hosts being monitored, not the number instances of Performance Sentry VM running in the enterprise.

Modifying vCenter Server and ESX Server Credentials

The vCenter Server and ESX Server credentials you entered in the installation dialogs are encrypted and stored in the Performance Sentry VM parameter file named **SentryVMParameters.xml** located in the installation folder.

These credentials are used to connect to the vCenter Server and to directly connect to every ESX server managed by the vCenter Server. You can view credentials under the tags of <vCenterCredential> and <ESXHostCredential> respectively in the **SentryVMParameters.xml.** Even though you can view them, you cannot modify them directly. To modify the credentials, or to add additional ESX server credentials, you must use the **vmcmd** command line interface.

vmcmd is found in the installation folder and has the following options:

```
addvCenterCredential [host] [user] [pwd]
removevCenterCredential
addESXHostCredential [host|*] [user] [pwd]
removeESXHostCredential [host|*]
```

you can view these options for yourself by opening a command window and changing folders to Performance Sentry VM and keying in the following command: `vmcmd /?`

Note that "*" in the 'add' and 'remove' options for ESXHostCredential represents a 'default' login for all hosts for which there is no ESXHostCredential entry in the parameter file. In other words, "*" contains the default credentials for all ESX hosts.

To modify the vCenter credentials, key in the command line:

```
vmcmd -addvCenterCredential [host] [user] [pwd]
```

where 'host' is the IP address of the vCenter Server or the server name on which vCenter is running and 'user' is a user name that has read capability.

To modify an ESX Server credential, key in the command line:

```
vmcmd -addESXHostCredential [host|*] [user] [pwd]
```

where 'host' is the specific ESX server name you wish to modify and 'user' a user name with read capability.

To add a new ESX Server credential, key in the command line:

```
vmcmd -addESXHostCredential [host|*] [user] [pwd]
```

where 'host' is the specific ESX server name you wish to add and 'user' a user name with read capability on that host.

Controlling Performance Sentry VM Connection to vCenter

Performance Sentry VM connects to VMware vCenter Server at startup and at subsequent intervals to retrieve a list of ESX servers under its control. You may control the delay time between connection attempts through parameters in the **SentryVMParameters.xml** parameter file located in the installation folder. The three parameters listed below control Performance Sentry VM connection to vCenter:

```
<vCenterQueryDelaySecs>3600</vCenterQueryDelaySecs>
<vCenterReconnectDelaySecs>300</vCenterReconnectDelaySecs>
<vCenterQueryRetryLimit>3</vCenterQueryRetryLimit>
```

vCenterQueryDelaySecs specifies the number of seconds that Performance Sentry VM waits to reconnect with vCenter to determine whether new ESX servers have been brought online since the last connection with vCenter. If a new ESX server is discovered, then Performance Sentry VM will create a new thread and attempt to connect to that server. The default delay interval is 3600 seconds or one hour.

vCenterReconnectDelaySecs specifies the number of seconds that Performance Sentry VM waits before attempting to connect to vCenter after a failed connection attempt. The default is 300 seconds or 5 minutes.

vCenterQueryRetryLimit specifies the number of times that Performance Sentry VM attempts to reconnect after a failed attempt to vCenter. After reaching the retry limit, Performance Sentry VM will wait for vCenterQueryDelaySecs until the next attempt to connect to vCenter.

Controlling Performance Sentry VM Messaging

By default, Performance Sentry VM writes messages to its own log file named **<computername>.SentryVM.LOG** located in the installation folder and to the Application Event Log. You may control logging by modifying the following default parameters in the **SentryVMParameters.xml** parameter file:

```
<emitAppEventlog>true</emitAppEventlog>
<emitAppTextlog>true</emitAppTextlog>
<outFolder></outFolder>
<maxTextLogKB>4096</maxTextLogKB>
<maxTextLogDays>30</maxTextLogDays>
<logLevel>INFO</logLevel>
```

Change 'true' to 'false' in their respective entries if you do not wish to write to the system level application event log or the application level text log (<computername>.SentryVM.LOG). You can write to a different folder by specifying the full path name in the outFolder parameter.

maxTextLogKB and maxTextLogDays control the point at which the current log file is closed and renamed **computername>.SentryVM.OLD.LOG** and a new log is opened.

logLevel controls the quantity of messages that are written to each of the logs. By default, 'INFO' is specified which results in informational messages indicating the status of the Performance Sentry VM service as well as any error or warning messages. There are five levels of logging; each level provides more information than the one listed above it. The levels are:

ERROR – critical errors only
WARN – warning errors, plus critical errors
INFO – informational, warning and critical errors
APPINFO – application information messages plus all 'INFO' messages
VINFO – verbose information used primarily for diagnostic purposes

Most customers will find that **INFO** level messages will suit their needs.

Contact Us

For additional information, visit Demand Technology's website at http://demandtech.com/

Contact support@demandtech.com if you have any questions regarding this document or Performance Sentry VM.

Again, thank you for your interest in Performance Sentry VM!