Contents

Preface ...................................................................................................................... vii
  Audience and Scope. ............................................................................................... vii
  Conventions Used in this Guide .............................................................................. vii

1 Introduction ............................................................................................................. 1
  The Quest Software Vintela Integrated Architecture. ........................................... 1
    Vintela Management Extensions ........................................................................... 1
  Why VMX? .................................................................................................................. 2
    Core Design ............................................................................................................. 3
    User Interface Design ............................................................................................ 3
    Client Technology ................................................................................................... 3

2 Hardware and Software Inventory ....................................................................... 5
  Using Resource Explorer to View Inventory Data ................................................... 5
  Hardware Inventory .................................................................................................. 6
    How to Collect Other Inventory Attributes ......................................................... 7
    Viewing Hardware Inventory ................................................................................. 8
  Software Inventory ................................................................................................... 9
    Viewing Software Inventory ................................................................................. 9
  Configuring File and Inventory Collections .......................................................... 10
    VMX File Collections ............................................................................................. 11
    VMX Inventory Collections .................................................................................... 13
    Viewing Software Inventory Details ..................................................................... 14

3 Advanced Inventory Collection .......................................................................... 17
  Adding Non-Computer Related Hardware Classes ............................................... 17
  Verifying MOF-based Extended Hardware Inventory ............................................. 18
  Extending Software Inventory with Scripts ........................................................... 19
    Script Output Defaults ............................................................................................ 19

4 Queries and Collections ....................................................................................... 21

5 Software Distribution ............................................................................................ 23
  Important Software Distribution Considerations ............................................... 23
  Distributing Software Using the Wizard ............................................................... 23
    Identify the Package ............................................................................................... 25
    Specify the Source Directory location .................................................................. 26
    Select the Software Distribution Method ............................................................ 27
      RPM Program Properties ..................................................................................... 28
      Solaris Program Properties .................................................................................. 29
      AIX Program Properties ....................................................................................... 31
      HP-UX Program Properties ................................................................................ 32
      OS X Program Properties .................................................................................... 32
      General Unix Program Properties .................................................................... 34
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Software Updates</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Important Considerations About Building Scripts</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Implementing a Software Patch on a Vintela Management Extensions Client</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Verifying the Software Patch</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>Troubleshooting Patch Implementation</td>
<td>64</td>
</tr>
<tr>
<td>7</td>
<td>Software Metering</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>Adding a Software Metering Rule</td>
<td>65</td>
</tr>
<tr>
<td>8</td>
<td>Remote Tools</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>Important Remote Tools Considerations</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>Remote Tools Security Rights</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>Restricting Access to the VMX Remote Tool Extensions</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>Configuring Remote Tools</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>Building the Command Line</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>VAS-Enabled Systems</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>The Remote Tool Types</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td>Using the Remote Tools Manager</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>Add Tool</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>Edit Tool</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>Delete Tool</td>
<td>77</td>
</tr>
<tr>
<td>9</td>
<td>Reports</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>Working with Reports</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>Troubleshooting Vintela Management Extensions Reports</td>
<td>86</td>
</tr>
</tbody>
</table>
Preface

Audience and Scope

This guide is intended for Windows® and Unix system administrators who will be using Vintela Management Extensions from Quest Software for the first time. By following the instructions presented in this guide, a system administrator will be able to manage Unix systems with the Microsoft® Systems Management Server 2003 console.

Note To access the VMX Installation Guide, navigate to the docs\pdf directory of the distribution media. By following the instructions presented in that guide, a system administrator will be able to install and configure new or existing Unix systems from within the Microsoft® Systems Management Server 2003 console.

Conventions Used in this Guide

The following notation conventions are used throughout this guide:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code (courier)</td>
<td>Highlights commands, coding, or class names. For example, “indclass is the name of a CIM indication class.”</td>
</tr>
<tr>
<td>Code italic (courier italic)</td>
<td>Highlights code variables (or place holders). Such as: “Select key, prop from c where e.”</td>
</tr>
<tr>
<td>SansSerif (Arial)</td>
<td>Highlights file names and paths</td>
</tr>
<tr>
<td>SansSerif italics (Arial italic)</td>
<td>Highlights file name and path variables</td>
</tr>
<tr>
<td>Italic</td>
<td>Highlights publication titles and emphasises important or new terms</td>
</tr>
<tr>
<td>Bold (Arial or Arial Black)</td>
<td>Interface elements (menus, fields, tabs, etc.) For example, “Select the Next button.” Selecting a menu item is indicated as follows: “Choose All Tasks</td>
</tr>
</tbody>
</table>

Quest Software products support a number of different implementations of Unix that include Solaris®, HP-UX®, Linux®, Macintosh OS X®, and AIX®. To refer to all of these platforms, the term “Unix” will be used for conciseness and consistency.
Introduction

The Quest Software Vintela Integrated Architecture

Quest Software has a unique combination of innovation, partnerships, alliances, and experience that has allowed the company to develop the core technologies that deliver true cross-platform interoperability of non-Windows systems from the Microsoft® tools and services preferred by enterprise managers. Branded as the Vintela Integration Architecture from Quest Software, this comprehensive architectural approach addresses the challenges associated with implementing and using different operating systems within an enterprise IT environment.

Through the Vintela Integration Architecture, Quest implements standards (both formal and de-facto) to bring uniformity to disparate systems. This technology strategy allows organizations to leverage their existing investment in Microsoft products and technologies by adding the ability to integrate and manage Unix, Linux, Java, and Macintosh environments through Microsoft systems and interfaces. It delivers a single point of administration without requiring duplicate tools and repetitive tasks.

Utilizing existing Microsoft Windows management tools and services, the Vintela Integration Architecture provides a set of coordinating technologies for Unix that reflect and compliment the design synergies of the Windows architecture. At the same time--to the Unix administrator--Quest Software solutions honor both the spirit and standards of the Unix experience. As a result, the Vintela Integration Architecture provides the ideal way to address the challenges organizations face when running Windows, Unix, and Linux in their IT infrastructure.

The Vintela Integration Architecture is based upon proven industry standards and protocols (such as WBEM, CIM, LDAP, and Kerberos) to power Quest Software solutions that address compelling customer needs in mixed OS environments. The Vintela Integration Architecture from Quest Software is the Holy Grail for pure integration and interoperability between Windows and non-Window environments.

Founded on the Vintela Integration Architecture architecture, Quest Software provides an ever-growing family of products that increase the ease of integration and interoperability between Microsoft and Unix environments. It's an idea whose time has come. Quest solutions allow enterprise organizations to leverage their Microsoft infrastructure investments across all platforms while consolidating their resource expertise around a single set of tools. These products fall under three general categories:

▲ Identity Management (see: [www.vintela.com/company/ident_mng.php](http://www.vintela.com/company/ident_mng.php) for more information)
▲ Systems Management (see: [www.vintela.com/company/sys_mng.php](http://www.vintela.com/company/sys_mng.php) for more information)
▲ Developer and Open Source (see: [www.vintela.com/company/open_source.php](http://www.vintela.com/company/open_source.php) for more information)

Vintela Management Extensions

Vintela Management Extensions from Quest Software (or VMX) is the Vintela Integration Architecture component for integrating Unix systems with the Microsoft Systems Management Server (SMS) 2003. VMX provides an administration console snap-in and client that gives IT administrators the ability to manage Unix, Linux, and Macintosh OS X systems from within the SMS console. They now have a single point of management to consolidate and standardize change and configuration management tasks for non-Windows systems. All of the Vintela Integration Architecture components can work together to provide
enhanced functionality. VMX has now integrated the Quest Software Vintela Authentication Services (VAS) technology so that VMX and SMS can work with the Unix systems that have been joined to Active Directory domains. The integration of VAS with VMX provides the following benefits:

▲ **Enhanced Security.** If the system is joined to Active Directory, and SMS is configured to publish data into AD, then security will be enhanced because the Management Point certificates will be obtained from Active Directory instead of HTTP.

▲ **Software Metering.** Software Metering allows you to determine how your organization uses software programs and helps ensure software license compliance. VMX, with VAS installed, lets the client determine if the user running the metered program is a VAS-enabled user and, if so, reports the user's domain.

▲ **Remote Tools.** VMX, with VAS installed, will not require users to enter their passwords when launching Remote Tools.

---

**Note** If you purchase and install the full version of VAS, then any programs run by VAS-enabled users have the complete DOMAIN/username.

For customers that do not have the full version of VAS already installed, VMX comes with a limited version of VAS, called vas-host which provides the capability to join Unix systems to Active Directory. VMX can then use the Active Directory identity for the Unix system and the VAS LDAP capabilities to query Active Directory for information relating the Unix systems and SMS.

It’s important to note that vas-host only provides a subset of the full VAS feature set: only those options that are directly relevant to joining Unix systems to Active Directory. For the user and group management features, you must use the full version of the VAS client. VMX is compatible with both VAS versions - if the full VAS client is already installed, it will not attempt to install the vas-host packages. (See VMX Integration with VAS on page 99 for more information on using vas-host.)

For more information on VAS, the Vintela Integration Architecture, and other Vintela Integration Architecture components (VGP, VSJ, and VSM) please visit www.vintela.com.

---

**Why VMX?**

Vintela Management Extensions (VMX) solves the problem of administering a mixed-platform environment by seamlessly extending SMS management support to non-Windows resources. VMX enables SMS to become a single integrated tool for managing all computer resources in your enterprise, Windows and non-Windows alike.

Enterprise-wide management of resources on both Windows and Unix platforms can be problematic. Microsoft Systems Management Server (SMS) provides excellent Windows resource management, but until VMX it did not provide the same capability to Unix resources. In a mixed-platform environment, administrators had to rely on tools other than SMS for all non-Windows platform management even though the administrative tasks are essentially the same.

VMX eliminates this problem and enables SMS to become the single integrated tool for managing all computer resources in your enterprise, Windows and non-Windows alike. All key VMX features including Hardware/Software Inventory, Software Distribution, and Remote Tools are seamlessly integrated into the SMS Administrator Console.
Core Design

The core design objective for VMX is to seamlessly provide Unix systems management capability through SMS. Seamless means that in all possible situations, VMX is designed to look, feel, and act as if it were written by the SMS development team. VMX even uses the same database and tables so all SMS features work as expected, even while managing Unix resources.

It is vital to understand that managing Unix systems with VMX and SMS is the same as managing Windows systems. Discovery, Hardware and Software Inventory, and Software Distribution all function the same as they do in SMS. VMX does not just use SMS data, nor does it replace the standard SMS Administrator Console and other standard tools. VMX seamlessly integrates Unix systems management into SMS by extending existing functionality and by adding Unix-specific functionality that acts the way you would expect SMS to act. VMX also adds the ability to launch Unix configuration and management tools from the Administration Console.

User Interface Design

In order to achieve seamless integration with the SMS user interface, the VMX interface is written as an MMC snap-in extension. It is specifically designed to transparently plug into the SMS Administrative console. shortcut menu items, wizards, property pages, and other components necessary to provide Unix management through SMS are simply integrated into the existing menus and property sheets.

The SMS Administrative Console is based on the Microsoft Management Console (MMC), as are many of the Windows management tools. Examples include the Event Viewer, Performance Monitor, Active Directory management, etc.

MMC is designed to be extensible. Snap-ins can be combined together in the MMC workspace to provide greater tool integration and flexibility for the administrator. There are many other benefits of using MMC which will not be covered here.

Client Technology

SMS utilizes the Windows Management Instrumentation (WMI), the Microsoft WBEM implementation for Windows. VMX uses OpenWBEM and VMX Providers that give Unix resources equivalent functionality.

WBEM is an industry initiative to provide management of systems, networks, users and applications across multiple vendor environments. WBEM simplifies system management by providing better access to both software and hardware data that is readable by WBEM-compliant applications.

VMX uses the same CIM Schema as Microsoft WMI (and so by extension uses the same CIM schema as SMS). Managed object data is presented in the SMS Resource Explorer, is used to define queries and collections, and is used in reports. VMX over time will add schema extensions specific to Unix properties. Even though these extensions are Unix-specific, they will integrate seamlessly into SMS.

“The CIM is a hierarchical, object oriented architecture that makes it comparatively straightforward to track and depict the often complex interdependencies and associations among different managed objects. Such interdependencies may include those between logical network connections and underlying physical devices, or those of an e-commerce transaction and the web and database servers on which it depends.” (CIM Tutorial, CIM Overview, DMTF and WBEM Solutions)
For more information see the VMX White Paper at: http://www.vintela.com/products/vmx/
VMX enables SMS administrators to seamlessly access hardware inventory for Windows, Unix, and Macintosh OS-X systems within SMS. SMS automatically displays system information for these resources within the SMS Resource Explorer and standard SMS reports.

VMX furthermore integrates software inventory using the Add/Remove Programs Hardware provider in the same manner that Windows-based systems employ.

- VMX enables SMS to discover and view information about a managed Unix-based resource. With VMX, SMS administrators can discover and manage non-Windows resources in their environment.
- VMX extends the same granular inventory search capabilities used for Windows systems to discover non-Windows resources using SMS.
- VMX allows administrators to use the same highly flexible and fully extensible Web reporting engine to gather information on non-Windows systems. SMS supports more than 120 default reports. No additional customization is required to gather information about Unix, Linux and Macintosh OS X systems.

Using Resource Explorer to View Inventory Data

Resource Explorer is a tool in the SMS Administrator console that displays the collected inventory data. When you start the Resource Explorer, it opens a window that displays the information collected by hardware and software inventory. In VMX the Resource Explorer displays hardware and software inventory information for Unix, Linux, and Macintosh OS X systems without any modification required.

To launch the Resource Explorer, follow these steps:

1. Navigate to a collection containing the client.
2. In the details pane, right-click the client whose information you want to view.

3. Choose All Tasks | Start Resource Explorer.

Hardware Inventory

Hardware inventory is the process of gathering hardware-related information from current clients in an SMS site. The information gathered includes processor type, network card, amount of memory, and disk information. VMX Client and Advanced Client hardware inventory work in similar fashion. You do not have to make any modifications to your queries. VMX uses the same hardware classes, database, infrastructure, and interface that SMS utilizes for managing Windows-based systems.

In addition, the Hardware Inventory client honors the policy settings in much the same fashion that the Windows Advanced Client does. The following similarities are notable:

1. Hardware inventory collection is optional. It honors the Client Agents settings.
2. The hardware agent honors the same collection schedule.
3. The hardware agent uses the same sms_def.mof file as its class and settings resource.

How to Collect Other Inventory Attributes

Perhaps you want to collect the serial numbers of your computer systems. The Serial Number attribute is not by default reported in the ComputerSystem class. But, you can obtain that data by going out to the command line and entering the following:

```
/opt/vintela/vmx/bin/owexecwql localhost:owipc /root/cimv2 "select SerialNumber from Vintela_computerSystem"
```

Got response: 1 CIMInstances
CIM Instance 1:
INSTANCE OF Vintela_ComputerSystem
{
    SerialNumber ="70431501060280";
};

The above prints the Serial Number to the screen. You can put it into a text file and do advanced collections, as well. (See Advanced Inventory Collection on page 17.)
Viewing Hardware Inventory

Follow these steps to view Hardware Inventory:

1. Start the Resource Explorer (See Using Resource Explorer to View Inventory Data on page 5.)
2. Expand the Hardware node.

**Note** If you have just installed VMX, your Hardware group may not have had enough time to populate as shown in the screen shot below. If that is the case you, need to run discovery and force inventory, as follows:
1. At the Linux console enter: `/opt/vintela/vmx/bin/vmxclienttool --run-discovery`.
2. Then enter: `/opt/vintela/vmx/bin/vmxclienttool --run-hardware-inventory`.
3. Click F5 on the Hardware node in the Resource Explorer and wait a minute or two.

3. Click **Operating System** to display the operating system information:

![Resource Explorer](image)

4. Right-click the operating system listed in the right-hand pane.
5. Choose **Properties** to display the Operating System Properties for this machine:
Repeat steps 4 through 6 to see more hardware inventory by selecting Services, Disk Drives, Partitions, Logical Disk, and so forth.

Software Inventory

Software Inventory is the process of gathering software information from current site clients in an SMS site. The information gathered can include operating system, installed programs, and any files you want to look for. With VMX, Add/Remove Programs software inventory information for Unix-based resources is found in the same place as it is for Windows resources. Thus, existing native packages for operating systems such as RPM for Linux show up under Add/Remove Programs in the SMS Resource Explorer.

Viewing Software Inventory

Follow these steps to view installed software on a VMX Client machine:

1. Start the Resource Explorer (See Using Resource Explorer to View Inventory Data on page 5.)
2. Expand the Hardware group.
3. Click Add/Remove Programs to display the programs installed on this machine:
Note
It is common for a much larger number of software objects to display from a Linux system than from a Windows system.

Configuring File and Inventory Collections

In VMX, File and Inventory Collections work just like they do in SMS, with a few additional features. You use software inventory to collect files from the clients and store them at the primary site server that the clients are assigned to. After you create the file collection rule and propagate it to the clients, the files are collected each time software inventory runs. You must specify the files you want to collect. When you do, you can use wildcard characters or specify multiple variations of a file, such as Status*.doc.

To configure a File or Inventory Collection, you must open the Software Inventory Client Agent Properties dialog, as follows:

1. From the Systems Management Server, navigate to Site Database | Site Hierarchy.
2. Expand a Site Code - Site Name node.
3. Expand Site Settings.
4. Select Client Agents.
5. In the details pane, right-click Software Inventory Client Agent and click Properties from the shortcut menu.
VMX File Collections

To configure a collection of non-Windows files, follow these steps:

1. Select the **VMX File Collection** tab with the Quest icon:
2. To specify files to collect, click the **New** icon (the “starburst” icon) to open the Collected File Properties dialog where you can add a new entry to the Files to be collected list:

![Collected File Properties dialog](image1)

3. Enter the name of a file or file type you want to collect from clients. For example, if you want to collect security file logs from VMX Clients, enter `secure`. You can enter exact file names, or you can use wildcards (such as `*.zip`). For example if you want to collect `secure.1`, `secure.2`, and `secure.3`, enter `secure.*`.

4. Click the **Set** button display the Path Properties dialog where you can select a full file path or a system environment variable to search:

![Path Properties dialog](image2)

By default, all hard disks on the SMS clients are scanned for files to collect. If you want to scan a particular folder or folder tree, click the **Set** button to change the default.
Note  When SMS sends a large volume of collected files across the network, network performance can suffer. To minimize this problem, you can restrict the path so that you collect only copies of the files from the desired folder tree, or schedule software inventory when network traffic is lightest.

i  Select All client hard disks to scan all hard disks on the SMS clients for the files to collect.
ii  Select Path name to specify a single folder or folder tree. For example the secure files are in the /var/log path.
iii  Select Search subdirectories to indicate that you want to search sub folders.
iv  Click OK to close the Path Properties dialog.

5.  Select Only collect files modified after the following date and enter a date in the field provided if you desire to limit the number of files you collect.

6.  Select Case sensitive file matching, if you desire.

7.  Select an amount in the Maximum size (KB) field for the files you want to collect. This is the maximum size allowed for collected files, in kilobytes (KB), that SMS collects from a client during a software inventory cycle. If the total size of the files collected by this rule exceeds this value, none of the files are collected.

   Collected files from each client in a site hierarchy can generate quite a bit of network traffic and require extensive storage space, especially if you choose a larger value for the combined size of the files. Test the traffic that would be generated on your network before you enable a larger combined file size.

8.  Click OK.

Next time the inventory cycles you can view the results with the Resource Explorer. (See Using Resource Explorer to View Inventory Data on page 5 for instructions on how to launch the Resource Explorer.)

If you wish to force this process, use the vmxclienttool. First force the policy update and then force file collection, by entering the following commands at the command line:

```
# /opt/vintela/vmx/bin/vmxclienttool --run-policy-update
# /opt/vintela/vmx/bin/vmxclienttool --run-file-collection full
```

For more information about using the vmxclienttool, see Using the VMX Client Tool on page 94.

**VMX Inventory Collections**

When you enable software inventory for a site, use this tab to specify the file types to inventory and how to report the inventory in the Resource Explorer.

To inventory non-Windows file types, follow the steps above to open the Software Inventory Client Agent Properties dialog:
Select the VMX Inventory Collections tab with the Quest icon from the Software Inventory Client Agent Properties dialog.

The procedure for using this tab is similar to the VMX File Collection tab, as explained above. However, there are two additional settings on this page that enable you to specify the level of inventory file information that you want reported in the Resource Explorer. For SMS to report information about inventoried files, you must select at least one of the following:

- **File details**: displays all scanned files, including unknown files, in the Resource Explorer under File Details.
- **Product details**: Resource Explorer lists software products only when manufacturer information is present.

**Note**  If you want Product Details for the Macintosh, you must not only select **Product details**, you must also edit the configuration file. On the command line of each client, go to: `/etc/opt/vintela/vmx` and edit the `vmx.conf` file. Change the line that says: “vmx.package_cache=disabled” to “vmx.package_cache=enabled.”

### Viewing Software Inventory Details

To view Software Inventory Details, follow these steps:

1. Start the Resource Explorer (See Using Resource Explorer to View Inventory Data on page 5.)
2. Expand **Software** to display the Software Inventory options:
   - △ Collected Files
   - △ File Details
3. Expand **File Details** to display information about all scanned files, including unknown files.
4. Expand **Product Details** to lists software products when manufacturer information is present.
Advanced Inventory Collection

You can use Advanced Inventory Collection to extend system-specific hardware inventory to meet additional requirements. VMX supports the Management Object Format, MOF-based text files for additional, custom hardware classes, and script-based extensions for the Add/Remove Programs hardware inventory class.

VMX Clients do not support architectures other than the SYSTEM architecture. Architectures other than the system architecture are not system specific and are therefore supported by traditional SMS Advanced Clients. For background information on the traditional Advanced Inventory Collection mechanisms, refer to the Microsoft Systems Management Server 2003 Operations Guide, page 67.

Adding Non-Computer Related Hardware Classes

The classic example of non-hardware and software related data associated with a system involves user data such as employee number, name, department, and cube number. Such data are valuable when running reports, but are not readily available to the VMX Client. However, this data can be supplied to the VMX Client using files containing MOF formatted text.

To extend the Hardware Inventory, follow these steps:

1. Add the new class and property reporting settings to the sms_def.mof file located in the \SMS\inboxes\clifiles.src\hinv folder on the site server:

   A new class definition for our example problem, called “SMS_UserInfo”, could look like the following:

   ```
   [SMS_Report (TRUE),
   SMS_Group_Name ("UserInfo"),
   SMS_Class_ID ("MICROSOFT|UserInfo|1.0") ]
   ```

   ```
   class SMS_UserInfo : SMS_Class_Template
   {
   [SMS_Report (TRUE), key]
   uint16 EmpNum;
   [SMS_Report (TRUE), key]
   string FullName;
   [SMS_Report (TRUE) ]
   string Department;
   [SMS_Report (TRUE) ]
   string Emp Position;
   [SMS_Report (TRUE) ]
   uint16 CubeNumber;
   }
   ```
Note  Do not make these changes to sms_def.mof while the SMS Administrator Console is running.

2. Create a file with a .mof extension on the VMX Client in the /var/opt/vintela/vmx/noidmofs directory (you may need to create the directory). The file should contain the class data that is specific to that machine.

Continuing our “UserInfo” example, the .mof file might take the following form: (Note the quotation marks surrounding the string data)

```mof
instance of SMS_UserInfo
{
  EmpNum = 100;
  FullName = "George Washington";
  Department = "State";
  EmpPosition = "President";
  CubeNumber = 001;
};

instance of SMS_UserInfo
{
  EmpNum = 101;
  FullName = "Hannibal Hamlin";
  Department = "State";
  EmpPosition = "Vice President";
  CubeNumber = 002;
};
```

Note  It is not necessary to compile the .mof file (as with the Windows client). The VMX Client Hardware Inventory Provider can use the text file.

This data will then be fully available from within SMS for the host system after the next policy and hardware inventory cycles complete.

**Verifying MOF-based Extended Hardware Inventory**

The additional hardware data shows up in the client's Resource Explorer in the same fashion as traditional hardware data once the next policy / hardware cycle completes. The policy update is needed to get the new sms_def.mof settings. The OpenWBEM log file defaults to the syslog. It is configurable in the /etc/opt/vintela/vmx/openwbem/openwbem.conf file and contains data describing any errors that have occurred during the hardware inventory collection process. To expedite these cycles, you can use the vmxclienttool to force these cycles to run, and for additional information on the use of vmxclienttool see Using the VMX Client Tool on page 94.
Extending Software Inventory with Scripts

Script-based software inventory uses a user-supplied shell script to add software inventory data into the SMS database. The data provided by the script displays in the SMS Administrator Console Resource Explorer under the Add/Remove Programs class.

The following instructions describe how to integrate the script output into the VMX Client Software Inventory Provider.

1. Create and validate the script using your preferred editor.

   **Note** You must test the shell script to ensure that it runs and produces the correct output.

2. Place the shell script (or scripts) in the `/var/opt/vintela/vmx/sinv_scripts` directory:

   As part of the inventory process, the VMX Client will run any scripts in this directory and parse the output into instances of the AddRemovePrograms class which it will include in its report to SMS.

Script Output Defaults

The VMX Client Inventory Provider expects a specific field order with pipe ('|') delimited output, as follows:

   DisplayName|InstallDate|ProdID|Publisher|Version

If a field's data is not available, a delimiter value is still required in the output. For example, if the DisplayName, ProdID, Publisher, and Version are known, but the InstallDate is not, the output would like be this:

   Vim - Vi iMproved||Vim-sparc|Bram Moolenar|6.3

   **Note** The delimiter count remains at 4.

The ProdID field is the key attribute in the AddRemoveSoftware CIM Class. It is required. A record reflecting the software package will not appear without it. Ensure that the ProdID value is unique for each system, perhaps by combining uname output with program name data. By default, the ProdID value is not displayed in the Resource Explorer.

You can configure the VMX Client to accept different delimiter(s), or a different field order and count by editing the `vmx.conf` file found in `/etc/opt/vintela/vmx/vmx.conf`

1. To change the default “pipe” delimiter to a comma, for example, add or edit the following line in the `vmx.conf` file:

   ```
   vmx.software_script_delim=,
   ```

2. To change the field order the VMX Client expects to receive from the script, add or edit the following line in the `vmx.conf` file:

   ```
   vmx.software_script_fields=ProdID,InstallDate,Publisher,DisplayName, Version
   ```

3. To leave a field out of the list, simply omit it, as shown in the following example (the ProdID field cannot be omitted):

   ```
   DisplayName,ProdID,Publisher,Version
   ```
Queries and Collections

You use queries and collections to manage the resources on your network. You can create default Unix-based queries and collections by running the `installQueries.bat` and `installCollections.bat` files found in `C:\Program Files\Vintela\VMX\MMC Snap-Ins`, preferably from the Site Server.

The resulting queries and collections are based on the following SQL statement, and serve as a basis for any Queries or Collections that you may want to add:

```
select Name, SMSAssignedSites, addresses, subnets,
       OperatingSystemNameAndVersion, ResourceDomainORWorkgroup,
       LastLogonUserDomain, LastLogonUserName, SMSUniqueIdentifier, resources,
       resource type, NetbiosName from SMS_R_System
```

The following lists the limiting clauses for each Query:

- All Unix Systems: where ClientVersion like '%VMX%
- All AIX Systems: where OperatingSystemNameAndVersion like '%AIX%
- All HP-UX Systems: where OperatingSystemNameAndVersion like '%HP-UX%
- All Macintosh OS-X Systems: where OperatingSystemNameAndVersion like '%Darwin%
- All Solaris Systems: where OperatingSystemNameAndVersion like '%Solaris%
- All Linux Systems: where OperatingSystemNameAndVersion like '%Linux%
- All Redhat Linux Systems: where OperatingSystemNameAndVersion like '%RedHat%
- All Suse Linux Systems: where OperatingSystemNameAndVersion like '%Suse%

The following lists the Collections that get created and the limiting clauses associated with each:

- All Unix Systems: where client version like '%VMX%
- All AIX Systems: where OperatingSystemNameAndVersion like '%AIX%
- All HP-UX Systems: where OperatingSystemNameAndVersion like '%HP-UX%
- All Linux Systems: where OperatingSystemNameAndVersion like '%Linux%
- All Macintosh OS-X Systems: where OperatingSystemNameAndVersion like '%Darwin%
- All Solaris Systems: where OperatingSystemNameAndVersion like '%Solaris%

If you need additional queries or collections beyond what these batch files provide, perhaps relating to versions of Operating Systems, simply use the SMS Administrator console to define new ones. The process of creating queries and collections is exactly the same as creating additional ones for Windows resources.
Software Distribution

Vintela Management Extensions software distribution to a managed Unix resource works just like the Windows Software Distribution. This section discusses the following two ways to distribute software to Unix/Linux resources:

- Distributing Software Using the Wizard
- Distributing Software Manually

Important Software Distribution Considerations

- Distributing an Existing Package.
  The Vintela Management Extensions Software Distribution Wizard does not allow you to distribute an existing package or use an existing advertisement. To distribute an existing package, you must either use the SMS Distribute Software Wizard instead of the Vintela Management Extensions Software Distribution Wizard, or manually create a new Advertisement. If you want to use an existing Advertisement, then you must re-run the Advertisement.

- Use of Quotes
  If you select a file name containing spaces, you must put it in quotes. For example if you are doing a general distribution with the shell script, it should look something like this: `vmx:script “/script name.sh” arg1 arg2 arg3`.
  You must also put quotes around script arguments which contain spaces if you want them interpreted as a single argument. For example: `vmx:script /bin/sh foo_script.sh “arg one”`.

- Set Permissions
  If you write a script to install files which are downloaded by the Vintela Management Extensions Client as part of software distribution, you should explicitly set the permissions on any file and not rely on the permissions that the Vintela Management Extensions Client sets on the files.

- Enable BITS
  Please see the VMX Installation Guide Planning and Deployment Guidelines for more information about BITS.

Distributing Software Using the Wizard

Vintela Management Extensions adds a software distribution wizard to the SMS console. It works just like the SMS Distribution Software Wizard and is launched from the same places.

Use the Vintela Management Extensions Software Distribution Wizard to configure the necessary settings for software distribution.

The Wizard steps you through the process of configuring software distribution packages, programs, and advertisements. Each step mirrors the SMS Distribution Software Wizard. Only when configuring Unix-specific distribution parameters, does the process differ.

The Wizard leads you through the following sequence:
1. Identifying the Package
2. Specifying the Source Directory location
3. Selecting the software distribution method, used to create a program associated with the package, such as Linux, Solaris, AIX, HP-UX, OS X or General Unix
4. Selecting the Distribution Points
5. Identifying the Program
6. Creating an Advertisement
7. Setting the date to Assign the Program

To use the Vintela Management Extensions Software Distribution Wizard follow this step-by-step tutorial to distribute chkrootkit-0.43-0.fdr.4.rh80.i386.rpm to the Vintela Management Extensions Client.

If you have reason to believe your system was involved in a security incident and might be running a sniffer, you can use chkrootkit to find out if your system was compromised. chkrootkit is a tool that detects and cleans up rootkits. RootKits are programs used by hackers to compromise Unix system roots. A rootkit is a collection of modified program sources or binaries which replace an entire set of system binaries.

Note: The sample files that you need for the following exercises are in the \eval\rpms folder of the installation CD-ROM.

1. In the SMS Administrator Console, navigate to and right-click the collection to which you wish to distribute software.
2. Choose All Tasks | Distribute Software to Unix Systems to start the Vintela Management Extensions Software Distribution Wizard.
3. Click Next.
Identify the Package

1. Enter *chkrootkit* in the Name field.
2. Enter *0.43-0* in the Version field.
3. Enter *English* in the Language field.
4. Click *Next*. 
Specify the Source Directory location

1. Click **Local drive on site server** to choose the Source Directory location where you will access the files.
2. Click **Browse** to navigate to the drive and select the eval\rpm folder.
3. Click **OK** to select the path and close the Browse for Folders dialog.
4. Click **Next**.
Select the Software Distribution Method

**Note** Depending on which software distribution method you select, a dialog displays asking you to enter Program Properties for that method. The next few sub-sections discuss the Program Properties pages for each method. Once you enter the Program Properties for that method, the Vintela Management Extensions Software Distribution Wizard proceeds to the Distribution Points dialog described in Select the Distribution Points on page 34.

To continue this step-by-step tutorial, follow these steps:

1. Select the **Linux (rpm)** software distribution method.
2. Click **Next**.
1. Choose **Install** as the Installation Type. (If the RPM you want to distribute is a software upgrade, select the **Upgrade** Installation Type to upgrade the client's software.)

2. Click **Browse**.

3. Navigate to and select **chkrootkit-0.43-0.fdr.4.rh80.i386.rpm**.

   Note the following with regards to browsing for RPM files:
   
   ▲ When specifying RPMs, the File Open dialog uses the location of the source files you previously specified in the Source Directory dialog; no other filenames are allowed. If you have not specified the source directory where the .rpm files are located, then you will not be able to launch the dialog to collect RPM filenames.
   
   ▲ There is no Add or Delete option for the RPM filenames. Click Browse to open the Select RPM files for Program dialog and select the file or files you want to add or delete. You can select one or more file names.
   
   ▲ Vintela Management Extensions strips the RPM file names of the path since that is not applicable to delivery and installation. SMS copies the files specified in the package to a directory on the distribution point. The Vintela Management Extensions Client then retrieves those into a working directory on the client. The RPM command is then executed against those files.

4. Click **Open** to select the RPMs files for the program.

5. Click **Options** to display the RPM Installation Options dialog where you can fine-tune the RPM installation:
The following are some special notes about the Vintela Management Extensions implementation of RPM Options.

- The RPM Installation Options dialog allows you to select all RPM Options except the `-h` and the `--percent`. Those options are used to present information to you when the RPM is being installed and have no value in Vintela Management Extensions. If you enter them in the Command Line field on the General tab, they will be correctly maintained, but the RPM Options page does not provide the option for setting or un-setting them.

- The `-vv` option provides the same data as the `-v` option, plus some other information. Therefore, if you select both the `-v` and the `-vv` options, Vintela Management Extensions automatically removes the `-v` option and uses the `-vv` option.

- The RPM information collected by the RPM Property page and the RPM Options page contain everything needed to build the complete command line. This data is written to the Command Line field on the General tab. Vintela Management Extensions synchronizes the data among the SMS General tab, the Command Line, and the RPM tab. The data is identical.

For more information on the RPM Installation Options, refer to *Understanding the RPM Command Line Options* on page 59.

6. Select **Install Documentation** and **Output additional information**.
7. Click **OK** to select the RPM Installation Options.
8. Click **Next** to continue.

To continue the tutorial, proceed to *Selecting the Distribution Points* on page 43 to select the Distribution Points.

### Solaris Program Properties

If you select the Solaris (pkgadd) software distribution method, the Wizard opens the Solaris Software Distribution Program `-pkgadd` dialog:
1. Click **Browse** to navigate to and select the package file.

2. Enter all or the names of the specific package(s) to install.

3. Click the **Options** button to display the Solaris Add Package Options dialog:

Select the package options and click **OK**.
**Note** For more information on the package options, click the **Help** button.

The default installation admin file: `/var/sadm/install/admin/default`, is designed to favor user interaction when conflicts or settings issues arise. Quest therefore recommends that you include in your distribution an admin file with options consistent with the non-interactive nature of the Vintela Management Extensions Client. Alternatively, you can reference an existing admin file on the target machine. If you choose to reference an existing admin file, make sure the path to the file is consistent across the collection. Otherwise, when the pkgadd system prompts for input, the deployment will fail by timing out (in 30 minutes).

For more information about the admin file, refer to the man page under the Solaris command, by entering `man -s 4 admin` at the command line on the Solaris machine.

4. Click **Next** to continue.

To continue the tutorial, proceed to **Selecting the Distribution Points** on page 43 to select the Distribution Points.

**AIX Program Properties**

If you select the AIX (installp) software distribution method, the Wizard opens the AIX Software Distribution - ‘installp’ dialog:

1. Click **Browse** to navigate to and select the package file.
2. Enter a specific package name or “all” to install all packages in that file or directory.
3. Select the package options (-a or -c).
4. Click **Next** to continue.
Note For more information on the package options, click the Help button.

To continue the tutorial, proceed to Selecting the Distribution Points on page 43 to select the Distribution Points.

**HP-UX Program Properties**

If you select the HP-UX (swinstall) software distribution method, the Wizard opens the HP-UX Software Distribution - 'swinstall' dialog:

1. Click Browse to navigate to and select the Depot Name.
2. Select Variables (-x) and enter the variable name, if desired. The -x option allows you to override the default value of session options. Enter the desired “option=value” in the field provided. For example, by default the autoreboot option is set to false, but if you have a package which requires a reboot, Select Variables and enter autoreboot=true in the field. You may specify multiple -x options. For a complete list of the defaults, options, and possible values you can specify, see the HP-UX documentation for the swinstall command.
3. Click Next to continue.

To continue the tutorial, proceed to Selecting the Distribution Points on page 43 to select the Distribution Points.

**OS X Program Properties**

If you select the OS X software distribution method, the Wizard opens the OS X Program Properties dialog:
1. Select **Disk Image File (-i)** and click **Browse** to navigate to and select the Disk Image File.

2. Select **Specify a package within the disk image file (-p)** to specify package(s).

   **Note** You do not need to select any packages; you can just send a .dmg and ignore this checkbox. But, if you want to install some package directory files from a Disk Image, enter the name(s) in this text field to build a “Shell Glob.” This is because it is not possible to open a dmg image on any other system besides a Mac. For example, if you have a dmg with package files for three different applications called **foo**, **bar**, and **baz** but you only want to install **bar** and **baz**, the entry in this text field would look like this: `{bar, baz} *`

3. Click **Browse** to navigate to and select the package directory file.

   **Note** You can only browse to the package if you have package directory files that are NOT in a disk image.

4. Select **Language** and enter the language name in the field.

5. Select **Target Volume** and enter the target volume in the field.

6. Select other desired package options (-q, -v, or -o).

7. Click **Next** to continue.

   **Note** For more information on the package options, click the **Help** button.

To continue the tutorial, proceed to **Selecting the Distribution Points** on page 43 to select the Distribution Points.
General Unix Program Properties

If you select the General Unix software distribution method, the Wizard opens a dialog asking you for the path to the script file:

1. Click **Browse** to navigate to and select the script file.
2. Click **Open** or double click to return to the Wizard.
3. Click **Next** to continue.

Select the Distribution Points

Once you complete the program properties for the software distribution method of your choice as described in the previous sections, the Wizard displays the Distribution Points dialog:
1. Select **LAB2**, the distribution point.

2. Click **Next** to display the Program Identification dialog:

   ![Program Identification Dialog]

   **Identify the Program**

   1. Enter **chkrootkit** in the Name field.

   2. Enter a comment in the Comment field, if desired.
3. Click **Next** to display the Advertise a Program dialog:

![VMX Software Distribution Wizard](image)

**Create an Advertisement**

From the Advertise a program dialog, follow these steps:

1. Select **Yes**, to advertise a program from this package.
2. Click **Next**.

At this point, the Wizard takes different paths depending on where you launched it. The next three screen shots show you the options you may be presented with. However, if you are following the step-by-step tutorial, skip these three dialogs and go to the Advertisement Name dialog described in **Specify an Advertisement Name** on page 38.

For example, if you start the Vintela Management Extensions Software Distribution Wizard from a package in the SMS Administrator Console, the Wizard is set to use the selected package. However, if you launch the Vintela Management Extensions Software Distribution Wizard from any node besides a Collection, it opens the Advertisement Target dialog:
This dialog prompts you to specify the collection that you want to receive the advertisement.

1. Click **Browse** to select a collection to receive the advertisement.
2. Click **Next** to display the Advertisement Name dialog described in Specify an Advertisement Name on page 38.

If you launch the Vintela Management Extensions Software Distribution Wizard from a system, it opens a different Advertisement Target dialog, shown below, that prompts you to choose an existing collection that contains that resource or create a new collection:
To demonstrate how Vintela Management Extensions allows you to create a new collection from a single system “on the fly,” choose to create a new collection.

Click **Next** to open the New Collection dialog:

![New Collection dialog]

Click **Next** to display the Advertisement Name dialog described in the next section.

### Specify an Advertisement Name

If you are following the step-by-step tutorial, you launched Vintela Management Extensions Software Distribution Wizard from a specific collection. Therefore, the Wizard knew which collection you wanted, so it skipped the above three dialogs. You are now at the Advertisement Name dialog:
1. Enter a comment in the Comment field, if desired.
2. Click **Next** to display the Advertise to Subcollection dialog:

3. Click **Next** to display the Advertisement Schedule dialog:
4. Click **Next** to accept the defaults set by the Wizard.

5. Click **Next** to accept the defaults set by the Wizard.
Chapter 5 – Software Distribution

Distributing Software Manually

You can manually create packages, programs and advertisements just like you can in Windows.

In the following example, you manually create a package, program, and advertisement for gc-6.2-0.fdr.3.rh80.i386.rpm and gc-devel-6.2-0.fdr.3.rh80.i386.rpm, the libraries and header files for gc development.

gc is the Boehm-Demers-Weiser conservative garbage collector that you can use as a garbage collecting replacement for C malloc or C++ new. It allows you to allocate memory basically as you normally would, without explicitly de-allocating memory that is no longer useful. The collector automatically recycles memory when it determines that it can no longer be otherwise accessed.

Creating a Package

1. In the SMS Administrator Console, right-click the Packages group.
2. Choose New | Package to display the Package Properties dialog:

6. Click Finish.
   The Wizard automatically creates the package, program, and advertisement for the software distribution.
3. In the General tab:
   i  Enter `garbCol` (for “garbage collector”) in the Name field.
   ii Enter `6.2-0` in the Version field.
   iii Enter `English` in the Language field

4. In the Data Source tab:
   i  Select **This package contains source files**.
   ii  Click **Set** to display the Set Source Directory dialog:
5. In the Set Source Directory dialog, do these steps:
   i. Select Local drive on site server.
   ii. Click Browse to locate the source directory.
   iii. Navigate to the CD-ROM drive and select the eval\rpm folder where the gc-6.2-0.fdr.3.rh80.i386.rpm is located.
   iv. Click OK to close the Browse for folders dialog.
   v. Click OK to close the Set Source Directory dialog.

6. Click OK to save the package properties you just configured, close the Package Properties dialog, and create the package.

Selecting the Distribution Points

1. In the SMS System Administrator Console, expand the Packages group.
2. Expand the garbCol package.
3. Right-click Distribution Points.
4. Select New | Distribution Points from the shortcut menu to start the New Distribution Points Wizard:
5. Click **Next** to display the Copy Package dialog:

6. Select LAB2 in the Distribution points list.

7. Click **Finish**.
   
   SMS assigns the package to the new distribution point.
Creating a Software Program

For a fuller understanding of program properties see Understanding the RPM Command Line Options on page 59.

1. In the SMS System Administrator Console, expand the Packages group.
2. Expand the garbCol package.
3. Right-click the Programs group and choose New | Program to display the Program Properties dialog:

![Program Properties dialog](image)

4. In the General tab:
   i. Enter gc (for “garbage collector”) in the Name field.
   ii. Enter VMX in the Command Line field.

**Note** To be able to leave the General tab, you must enter something in the Command field. VMX replaces whatever you enter here with the proper installation commands from operating system tab that you choose next.

5. In the Requirements Tab you can change the Maximum allowed run time to the desired number of minutes.
Note For large package distribution that may take several hours you can configure the time out value for program run times on client computers. This avoids timing out when distributions take a long time. You can configure the timeout value after you have created the program using the VMX Software Distribution Wizard or when you manually create a software program.

If you have created an Advertisement using the VMX Software Distribution Wizard and then determine that you need to re-set the **Maximum allowed run time**, you can access the Program Properties, as follows: navigate to **Packages**, expand the package you want to reconfigure. Then, select **Programs** and double click the name of the program. Open the **Requirement** tab.

If you do not specify a **Maximum run time option** on the Program Properties dialog, SMS will provide a default value of 12 hours.

For more information, click the **Help** button to access Microsoft's explanation of the **Maximum allowed run time** feature.

6. In the RPM tab:
i Select This SMS program is an RPM.
ii Select Install from the Installation Type group.
iii Click Browse and VMX displays the directory you specified when you created the package.
iv Select gc-6.2-0.fdr.3.rh80.i386.rpm and click Open.
v Click the Options button to open the RPM Installation Options dialog:
Note  Vintela Management Extensions provides access to advanced RPM functionality. For more information on the package options, click the Help button or refer to Understanding the RPM Command Line Options on page 59, for more information about program properties. For the purposes of this example, you do not need to set any additional RPM options.

vi  Click OK to close the RPM Installation Options dialog.

vii Click OK to close the Program Properties dialog and return to the Administrator Console.

Now go back to step #3 above, re-open the Program Properties dialog, and create another software program named gcLib for the libraries and header files for the gc development, the gc-devel-6.2-0.fdr.3.rh80.i386.rpm.

Make gcLib dependent on gc. Do that after you complete the RPM tab in step #5 above. At that point, select the Advanced tab from the Program Properties dialog:

From the Advanced tab, follow these steps:

i  Select Run another program first.

ii Choose garbCol as the package name.

iii Choose gc as the program to run first.

When you create the Advertisement in the next step of this tutorial (in Creating an Advertisement below), you only need to advertise gcLib as it will automatically call gc.

7. In the AIX Tab:
i Select This SMS program is an AIX package.
ii Select Browse to navigate to and select the package file.
iii Select Directory and enter the directory name in the field.
iv Select the package options (-a or -c) and click OK.

Note For more information on the package options, click the Help button.
8. In the Solaris tab:

i Select **This SMS program is a Solaris package**.

ii Click **Browse** to navigate to and select the package file.

iii Click the **Options** button to display the Solaris Add Package Option dialog:

iv Select the package options and click **OK**.
Chapter 5 – Software Distribution

Note For more information on the package options, click the Help button.

v Click OK.

9. In the OS X tab:

i Select **This SMS program is an OS X package**.

ii Select **Disk Image File (-i)**.

iii Click **Browse** to navigate to and select the disk image file.

iv Select **Specify a package within the disk image file (-p)**.

v Click **Browse** to navigate to and select the package directory file.

vi Select **Language** and enter the language name in the field.

vii Select **Target Volume** and enter the target volume in the field.

viii Select other desired package options (-q, -v, or -o) and click **OK**.

Note For more information on the package options, click the Help button.
10. In the HP-UX tab:

i. Check **This SMS program is an HP-UX depot**.

ii. Click **Browse** to navigate to and select the depot name.

iii. Click **Variables (-x)** and enter the variable name.

Note: For more information on the `-x` option, click the **Help** button.

iv. Click **OK**.
11. In the General Unix tab:

   ![Program Properties window]

   i. Select **This SMS Program is a Vintela Management Extensions General Unix Distribution**.
   
   ii. Click **Browse** to navigate to and select the script file.
   
   iii. Click **Open** or double click to select.
   
   iv. Click **OK**.

12. Click **OK** to close the Program Properties dialog and create the program.
    SMS copies the program and related files to the distribution point you selected.
Creating an Advertisement

1. In the SMS Administrator Console, right-click the Advertisements group.
2. Choose New | Advertisement to display the Advertisement Properties dialog:

3. In the General tab:
   i  Enter colGarb (for “Collect Garbage”) in the Name field.
   ii Select garbCol from the Package drop-down list.
   iii Select gcLib from the Program drop-down list.
   iv Click the Browse button next to the Collection field.
   v Select All Linux Systems.
   vi Click OK to close the Browse dialog.
4. In the Schedule Tab, create the schedule as follows:
   i Click the New Schedule button (the “starburst” icon), to assign a new schedule.
   ii Click Apply to use the default schedule.
   iii Click OK to complete advertisement.
   iv Click OK to close the Advertisement Properties dialog.

   SMS creates the advertisement and sends the appropriate distribution policy to the Management Point. In this example scenario, gcLib is dependent on gc, so you only need to advertise gcLib. It calls the gc program automatically.
Verifying Software Distribution

After distributing software or patches to your client, you may want to verify that the procedure was successful. This section explores some ways to do that.

First, you can check the System Status, as follows:

1. Expand **System Status** node of the SMS Administrator Console.
2. Click **Advertisement Status** to see the status of the advertisements.
   Advertisement Status provides a summary report of the total number of clients that have received and run an advertisement.
   To get details on a specific advertisement, follow these steps:
   i. Expand **Advertisement Status**.
   ii. Click the advertisement name listed under Advertisement Status.
      The SMS Administrator Console lists the site servers in the right-hand pane.
   iii. Right click a site server to open the shortcut menu:

   iv. Choose **Show Messages | All** to list all the status messages for the specified site server:
3. **Click Package Status** to see the status of the packages.

   Package Status provides a summary report of the health of packages and distribution points in your site. In many organizations, administrators distribute multiple packages concurrently to multiple destinations. Package Status allows you to monitor when packages arrive at distribution points. A package must arrive at a distribution point before a client can access, install, or run an advertisement. All dates in Package Status are displayed in the time zone of the computer running the Administrator console.

   Package Status provides three levels of status information details:
   - ▲ Summary status for all packages in all sites
   - ▲ Details for a specific package in all sites
   - ▲ Details for a specific package in a single site

   To get details on a specific package, follow these steps:
   i  Expand **Package Status**.
   ii  Click the package name listed under Package Status to list the sites servers in the right-hand pane.
   iii  Right click a site server to open the shortcut menu.
   iv  Choose **Show Messages | All** to display the Status Messages: Set Viewing Period dialog:

   For a complete list of VMX Status Messages, refer to Vintela Management Extensions Client Status Message on page 87.
Set the viewing period and click **OK** to display the SMS Status Message Viewer:

4. **Click Status Message Queries.**

The SMS Administrator Console opens a list of Message Status queries that are useful. For example, you will find All Status Messages for a Specific Advertisement at a Specific Site or All Status Messages for a Specific Package at a Specific Site helpful.
You can also check Reports to verify that the software distributed, as follows:

1. Expand **Reporting** node of the SMS Administrator Console.
2. Click **Reports**.
   
The SMS Administrator Console opens a list of Reports that are useful. For example, you will find Advertisement status messages for a particular client and advertisement helpful.

### Verifying Software Distribution on a Linux System

In addition to using the System Status, you can verify that the software was distributed by checking on the Vintela Management Extensions Client for the distributed software.

---

**Note** SMS can take some time to distribute the software. Be sure to let sufficient time lapse after the software distribution process.

---

In this example we will verify that the chkrootkit software was distributed using the Vintela Management Extensions Software Distribution Wizard.

1. In the SMS Administrator Console, click **Packages**.

---

**Note** The SMS Administrator lists the names of the packages in the right-hand window along with their Package ID. The Package ID indicates that it was installed by means of the SMS process.

---

2. Establish a remote session with PuTTY, as follows:

---

**Note** You must have an X Server running. For more information about running remote tools, see [Remote Tools](#) on page 69.

---

i. In the SMS Administrator Console, expand **Collections**.
ii. Select the **All Linux Systems** group.
iii. Right-click on the Vintela Management Extensions client machine to open its shortcut menu.
iv. Choose **All Tasks | Remote Unix Tools** to view the remote Unix tools.
v. Click **PuTTY ssh shell** from the shortcut menu.
vi. Enter the username and password at the prompt and PuTTY logs into the remote system and starts the terminal session.

---

**Note** Your keystrokes will not display as you type the password.

---

3. At the command prompt, enter `rpm -q chkrootkit`.
Understanding the RPM Command Line Options

This section explains the Vintela Management Extensions extensions to the SMS Program properties.

Vintela Management Extensions has added a page in the program properties sheets for each software program type it supports. For example, there is an RPM tab for the Linux packages.

You can enter the RPM command line in the General tab provided by SMS and Vintela Management Extensions will recognize it as long as you precede the command with “Vintela Management Extensions.”

A valid line would look like:

```
vmlx:rpm -i --nodeps a.rpm C.rpm d.rpm
```

You can use multiple RPM files names. For example:

```
vmlx:rpm -i a.rpm b.rpm c.rpm d.rpm
```

You must locate the files in the location specified with the SMS Package. The Vintela Management Extensions MMC Snap-in extension does not move or relocate files.

You can use all standard combination of the RPM option flags. For example, you can enter -i or --install to specify an RPM installation.

Vintela Management Extensions does not store the entered string. It parses it and stores the flags and values separately. It then recomposes the string when it needs to represent it, such as in the Command Line field in the General tab.

---

**Note** If you entered both -i and --install or if you entered a single option several times, Vintela Management Extensions interprets and stores these as a single value, type=install. When Vintela Management Extensions reconstructs the command line, instead of having both the -i and --install, it has only -i. Vintela Management Extensions uses the shortest representation of a command line whenever possible because there is a limit of 255 characters for the RPM command line.

It is not necessary for you to understand or use the RPM command line. Vintela Management Extensions builds it for you when you use the RPM tab.

For Vintela Management Extensions to recognize a command line in the SMS Program, it must have the following:

1. The Command Line field on the General tab must start with vmlx:rpm.
2. You must check **This SMS program is an RPM** on the RPM tab.

In order to switch from the General tab to another tab you must type something into the Command Line field, you cannot leave it blank. If you enter **vmlx:rpm** then Vintela Management Extensions will check **This SMS program is an RPM** automatically. But you can enter anything and then check that box later.

The RPM tab allows you to specify that the SMS program is an RPM, whether it is for Install, Upgrade, or Freshen.

You can browse for the RPM file names in the directory specified in the package. No other filenames are allowed.

Click the **RPM Options** button to set the RPM options, except the -h and the --percent. Those options are used to present information to you when the RPM is being installed and have no value in Vintela
Management Extensions. If you enter them in the command line on the General tab, they will be correctly maintained, but the RPM Options page does not provide the option for setting or unsetting them.

The `-vv` option provides the same data as the `-v` option, plus some other information. Therefore, if you select both the `-v` and the `-vv` options, Vintela Management Extensions automatically removes the `-v` option and uses the `-vv` option.

The RPM information collected by the RPM Property page and the RPM Options page contain everything needed to build the complete command line. This data is written to the Command Line on the General tab. Vintela Management Extensions synchronizes the data among the SMS General tab, the command line, and the RPM tab; the data is identical. The tabs with the Quest icons have Windows controls for building the RPM command line, and the General tab allows you to manually enter the command line. Regardless of how you enter or modify the data, these two sets of data are 100% in sync. They have exactly the same meaning, but with two different visual presentations. The Vintela Management Extensions pages operate directly on the data that underlies the General tab by SMS.

**Note** If you have not specified the source files in the package, then you will not be able to launch the dialog to collect RPM filenames.

When specifying RPMs, the File Open dialog uses the location of the source files you specified in the package. The files must be in the directory specified by the package.

There is no Add or Delete option for the RPM filename. Open the RPM Filename Collection dialog and select the file or files you want to add or delete. You can select one or more file names in the File Open dialog.

Vintela Management Extensions strips the RPM flies names of the path since that is not applicable to delivery and installation. SMS copies the files specified in the package to a directory on the distribution point. The Vintela Management Extensions Client then retrieves those into a working directory on the client. The RPM command is then executed against those files.
Software Updates

Vintela Management Extensions does not have full patch management. If you have patches available, download them and distribute them by means of the software distribution methods described in Software Distribution on page 23. Most often you will want to distribute a script file using the General Unix Software Distribution method described in General Unix Program Properties on page 34.

Important Considerations About Building Scripts

If you build a script file on a Windows system, it will create Windows end-of-line characters unless your text editor has an option to save Unix end-of-line characters. VMX tries to correct for this. When VMX sees #! in the first line of a script file, it automatically replaces the Windows end-of-line characters with Unix end-of-line characters.

In some cases, patches or software files start with #! but include binary data. In that case, you must prepare a script that calls the file with the binary data.

The following demonstration uses sample script files from the scripts folder on the distribution media.

Implementing a Software Patch on a Vintela Management Extensions Client

This section demonstrates how to successfully deploy patches for a Unix system using the VMX Client Installation Wizard.

The system selected to perform this demonstration is a Sparc-based Solaris 9 system. The patch is the point patch #112661-06 obtained from the Sun site http://sunsolve.sun.com.

Point Patches are designed to fix or solve a specific bug on your system or the software installed on it. Usually they replace or add files to the system. The purpose of point patch #112661-06 is to prevent the IIM port from being hijacked.

Note: Before you begin to deploy patches you must have a working SMS 2003 environment, properly configured with installed Vintela Management Extensions components. You must also have a Vintela Management Extensions Client running.

To deploy the Solaris Cluster Patch on a Vintela Management Extensions Client, follow these steps:

1. Download the patch from the Sun site and copy it to a place on the site server where you will be able to distribute the software. For the sake of this demo the file is named: 112661-06.zip.

Note: You must create a shell script named patchadd.sh and place it in the same place you copy the zip file.
The content of this script file is as follows:

```
#!/bin/sh
# The above line must be present; it tells Vintela Management Extensions to
# strip the
# Windows end-of-line characters.
# This script will install the patch 112661-06 on a Sparc based
Solaris 9
#
echo "Patch Installation in progress..."

echo "Unpacking ZIP file..."
/usr/bin/unzip 112661-06.zip

echo "Moving patch folder to /tmp..."
mv 112661-06 /tmp

echo "Changing current directory to /tmp..."
cd /tmp

echo "Installing patch..."
/usr/sbin/patchadd 112661-06
```

**Note**  Even though the package content is downloaded to `/var/opt/vintela/vmx/packagecache`, the `patchadd` command will not install the software from that directory. For that reason, move the patch folder to the `/tmp` directory and install it from there.

2. Start the SMS Administrator Console and navigate to the **All Solaris Systems Collection**, right-click the group and from the shortcut menu select **All Tasks | Distribute Software to Unix Systems**.
3. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
4. Enter **patch-112661-06** in the Name field of the Package Identification dialog.
5. Enter **9** in the Version field.
6. Enter **Sun Microsystems** in the Publisher field.
7. Enter **English** in the Language field.
8. Enter **Solaris Software Patching using SMS** in the Comments field.
9. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
10. Select **Local Drive on Site Server** on the Source Directory dialog.
11. Click **Browse** to navigate to the drive and select the directory where you placed the zip file content.
12. Click **OK** to select the path and close the Browse for Folders dialog.
13. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
14. Select **General Unix** as the software distribution method for the Sun Point Patch at the Select Software Distribution Method dialog.
15. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
16. Click **Browse** to navigate to and select the script named **patchadd.sh**.
17. Click **Open** or double click to return to the Wizard.
18. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
19. Select the Distribution Points where you want to copy the patches in the Distribution Points dialog.
20. Click **Next** to continue the Vintela Management Extensions Software Distribution Wizard.
21. Enter **patch-112661-06** in the Name field of the Program Identification dialog.
22. Enter **This script will install patch 112661-06 on a Sparc based Solaris 9 system** in the Comment field.
23. Click **Next** to continue to continue the Vintela Management Extensions Software Distribution Wizard.
24. Select **Yes** at the Advertise a Program dialog.
25. Click **Next** to continue to continue the Vintela Management Extensions Software Distribution Wizard.
26. Click **Next** to accept the defaults set by the Wizard on the Advertisement Name dialog and continue.
27. Click **Next** to continue to accept the defaults set by the Wizard on the Advertise to Sub-collections dialog and continue.
28. Click **Next** to accept the defaults set by the Wizard on the Advertisement Schedule dialog and continue.
29. Click **Next** to accept the defaults set by the Wizard on the Assign Program dialog.
30. Click **Finish** to complete the task.

The Vintela Management Extensions Software Distribution Wizard automatically creates the package, program, and advertisement for the software distribution.

**Verifying the Software Patch**

Once you deploy the patch using the Vintela Management Extensions Software Distribution Wizard, you will want to verify that the package and advertisement were created and that no errors were generated in the SMS Status System.

Follow these steps to make sure the patch was distributed to the client:

1. Look at the System Status to verify that the **SMS_Distribution_Manager** and **SMS_Replication_Manager** did not report errors.
2. Verify that every Distribution Point System you selected received a copy of the patches. The Wizard creates a folder containing the software and scripts with a name like `C:\SMSPKGC$\<PKG-ID>`.
3. On the client side, use `vmxclienttool` to verify the client is receiving policies from the Management Point. (See Using the VMX Client Tool on page 94 for more information on how to use `vmxclienttool`.)
Troubleshooting Patch Implementation

There are different ways to confirm that a patch has been installed successfully or not depending on which method you used to install the patch. In our case, we can confirm the installation using the following command:

```
# showrev -p | grep 112661-06
Patch: 112661-06 Obsoletes: 113971-01 Requires: Incompatibles:
    Packages: SUNWlccom, SUNWiiimu, SUNWiiimr, SUNWxi18n,
              SUNWxim, SUNWxi18x, SUNWxinx, SUNWj3irt
#```

The output shows the patch recently installed on the system along with other useful information. If you do not receive the previous output, consult the log file created in `/var/sadm/patch/112661-06`.

For more information refer to the README file that comes with the zip file or in the Sun site [http://sunsolve.sun.com](http://sunsolve.sun.com).
Software Metering

In Vintela Management Extensions, Software Metering works just like it does in SMS with a few exceptions. By using software metering data, you can determine how your organization uses software programs to help ensure software license compliance. You can combine software metering program usage data with other SMS data such as software inventory, product compliance, and hardware inventory to create comprehensive reports.

SMS 2003 software metering monitors and collects software usage data on clients. Data collection is based on software metering rules that are configured by the System Administrator. The Software Metering Client Agent runs on the client. The agent accepts software metering rules from the SMS site server and records program usage as specified in the rules. Program usage data from individual clients is forwarded to the client's assigned SMS site for processing according to the prescribed metering collection schedule. The site then summarizes the data and propagates it to its parent site. The central site contains program usage data from all clients within the hierarchy that are assigned to sites that have software metering enabled.

The process of configuring the Software Metering Client Agent for Vintela Management Extensions is exactly like it is for SMS. See Chapter 8, *Software Metering*, in the *Microsoft Systems Management Server 2003 Operating Guide* for more information.

Adding a Software Metering Rule

Set the rules for metering Vintela Management Extensions Clients in the Software Metering Rule Properties dialog, as follows:

1. Expand **Site Database** on the Administrator Console.
2. Right-click **Software Metering Rules** and choose **New | Software Metering Rule**.
3. Complete the information on the General tab as you would normally do for SMS with the exceptions noted below:

**Program name**: The name of the software program, up to 256 characters. This field is required.

**File name**: The software program’s executable file name, up to 256 characters. File Name is the name that launches the program; it must be an appropriate executable file. Some programs are used as placeholders to launch other programs, so you should specify the name of the program that
ultimately executes the program itself. Remember, in Software Metering, the file name is case sensitive. And, with Vintela Management Extensions, do NOT use Browse, to select Unix files.

**Note**
In Vintela Management Extensions, File Name is mandatory and not optional as it is for Windows applications.

Unix files do not have the .exe file type extension. So, when you click **OK**, a warning dialog displays:

![Software Metering Rule Properties window](image)

**Original file name**: Vintela Management Extensions ignores this field; leave it blank when dealing with Vintela Management Extensions Client Software Metering.

**Version**: The version of the software program. You can use up to 48 characters. Alpha numeric, ?, *, and PERIOD characters are allowed. The asterisk and question mark characters are wildcard character entries. Use the wildcard character “*” for a string and “?” to represent a character. Vintela Management Extensions checks the file version slightly differently than Windows does. For Vintela Management Extensions, if the executable’s version information is not available, the rule’s File Version will be ignored for purposes of matching the rule. In the Window’s client, if you leave the Version field blank, software metering will match the rule only if the version in the program header is blank.

**Language**: The language of the software program. NOTE: If the language for an SMS metering rule is set to anything besides “Any,” the metered data will not be reported by SMS. So you must set the language for Vintela Management Extensions metering rules to “Any”.

**Comment**: Administrator comments, up to 256 characters. This field is optional.

**Site Code**: The site code that the software metering rule will apply to.

4. Select **This rule applies to the specified site and all its child sites** to indicate whether the software metering rule will apply to all lower level sites, in addition to the site that is specified in site code.
5. Click **OK** to close the Software Metering Rule Properties dialog. The Vintela Management Extensions Client will receive the software metering rules in the policy.
Remote Tools

VMX Remote Tools Extensions allows you to remotely access Unix and Linux systems. This gives you the ability to run programs and utilities on your VMX Client systems from within the SMS Administrator Console.

**Note** You must have an X Server running in order to launch some X based tools. A Windows-based X Server enables the UI to show the Unix tools on your Windows desktop. Some choices might be StarNet X-Win32 X Window Server, a platform-independent, network-transparent windowing system that provides a client/server interface between display hardware and the desktop environments. Or, you might use Cygwin/XFree86, an open-source implementation of the X Window system. There is also a Hummingbird product. However, Quest cannot provide support for any of these third-party products.

VMX allows you to remotely access Unix and Linux systems using any of the following methods:

- Opening a standard, ssh-based, interactive shell window.
- Launching specific tools such as the RedHat Log Viewer or the Top system monitor.
- Using the remote control client VNC (Virtual Network Computer).

Each of the above options relies on an SSH (Secure SHell) server that is configured and running on the target system. VMX recommends the open-source SSH server OpenSSH. For information regarding obtaining and installing the proper SSH server software, go to Vintela Resource Central at [http://www.vintela.com/resources](http://www.vintela.com/resources), or to the OpenSSH website at [www.openssh.org](http://www.openssh.org).

**Important Remote Tools Considerations**

The following are some special notes pertaining to accessing Remote Tools on non-Windows environments:

- VMX Remote Tools does not provide the ability to remotely control a Unix or Linux system like you can with a Windows system. VMX allows you to launch Unix/Linux tools located on the VMX Client from a Windows machine. You can add or remove remote tools to customize the UI for your own environment.
- You can now configure VAS-enabled Remote tools. If you have a kerberos-enabled sshd, PuTTY will be integrated with the windows ssapi to enable better VAS integration. See VAS-Enabled Systems on page 73 for more information.
- You can now configure the Virtual Network Computer (VNC) Software for remote access to the Macintosh Desktop GUI, see Configuring Remote Access for the Macintosh on page 82.
- Remote Tools X forwarding is required on SSH. If SSH on the system you are connecting to by means of Remote Tools doesn’t allow X forwarding, the remote tools will not work for that system.
Remote Tools Security Rights

VMX users must have the correct security rights in order to use Remote Tools. VMX honors and applies standard SMS security rights without any changes to the normal functionality. You can assign specific permissions to Users and Groups, just as you would with SMS 2003.

You set user rights by means of the Security Rights node of the SMS Administrator Console:

1. In the SMS Administrator Console, right-click the Security Rights node.
2. Choose New | Class Security Right, as shown in the following graphic:

The Security Rights Properties dialog displays:
Select the User name.
4. Select the Collection Class.
5. Check the Use Remote Tools Permissions.
6. Click OK.
   SMS adds the new name, class, and permissions for that user to the list in the right-hand detail pane.

Restricting Access to the VMX Remote Tool Extensions

You can disable VMX remote tools by using standard SMS access control. Simply, edit the Class Security Right or the Instance Security Right properties for the specific user. For example,

1. From the SMS Console, click Security Rights to list the User Name Accounts in the right-hand window.
2. Double Click the User Name Account of the Collection Class that has Use Remote Tools permissions.
3. Scroll down in the Permission window to locate and un-select Use Remote Tools.
4. Click OK.


Configuring Remote Tools

The Remote Tool Manager allows the options that show up in the VMX client-based systems options to be fully customized.
To start the Remote Tool Manager, follow these steps:

1. In the SMS Administrator Console, right-click the **Tools** node.
2. Choose **All Tasks | VMX Remote Tools Configuration** to display the Remote Tool Manager dialog:

   ![Remote Tool Manager](image)

   The step-by-step directions for this procedure continue at: **Add Tool** on page 75. However, the following information will help you in building the command line in that dialog.

### Building the Command Line

The commands you built with the Remote Tools Manager apply to all instances of the SMS console, not just the instance where the options are configured. It is therefore important when building commands to remember to build them so that they will run on all machines where the console is installed. To this end, VMX has supplied some substitution variables, described in the following section.

Use the following variables when building the command line. The VMX Console Extension software replaces these variables with the following data:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%PUTTY%</td>
<td>The path to PuTTY.exe installed with the VMX Console Extensions. The installer allows this location to be customized, so the location could be different depending on the options chosen at installation. (VMX gets this location from the Windows Registry.)</td>
</tr>
</tbody>
</table>
The Remote Tools commands have the following basic form (other options can be added):

```
%PLINK% -ssh -X -l %USERNAME% %IPADDRESS% <command>
```

After the VMX Console Extension software replaces these variables, this command line will look similar to the following example:

```
C:\path\PLINK.exe -ssh -X -l johnUser 192.168.5.10 redhat-config-users
```

This command directs PLINK to use the SSH protocol to connect to the target machine. The `-X` option enables X11 forwarding (must be enabled on the target machine). The `-l` option tells PLINK to use “johnUser” for the username, the IP address of the target machine is 192.168.5.10 (taken from the selected computer object), and the command to issue on the client is “redhat-config-users.”

**VAS-Enabled Systems**

When connecting to the selected computer, VMX, by default, prompts for the desired username and password (so long as the `%USERNAME%` substitution variable appears in the command line). VAS enables true Single-Sign-On (SSO) capabilities throughout the entire enterprise so that it can make a connection without prompting. VMX Remote Tools takes advantage of this feature.

The ssh client provided by VMX is the popular ssh client, PuTTY. The version of PuTTY supplied by VMX is Microsoft SSPI-enabled. SSPI (with Kerberos) is the standard used to provide the SSO functionality. With VAS, the target computer is joined to the same Active Directory domain to which the Window's client belongs. This enables the VMX PuTTY client to use the credentials obtained by the current user at login time, to obtain new credentials for the target system. The GSSAPI-enabled ssh server process running on the target computer can examine those credentials and authenticate the user without prompting.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>%PLINK%</code></td>
<td>The path to the PLINK.exe installed with the VMX Console Extensions. The installer allows this location to be customized, so the location could be different depending on the options chosen at installation. (VMX gets this location form the Windows Registry.)</td>
</tr>
<tr>
<td><code>%IPADDRESS%</code></td>
<td>The IP address of the remote system. This is the IP address of the target (VMX Client) machine from which you will run the tool. VMX retrieves the IP address of the client machine you select.</td>
</tr>
<tr>
<td><code>%HOSTNAME%</code></td>
<td>The Host name of the remote system. This is the host name of the target (VMX Client) machine from which you will run the tool. VMX retrieves the hostname of the client machine you select.</td>
</tr>
<tr>
<td><code>%USERNAME%</code></td>
<td>If included, a dialog displays when the connection is attempted that prompts for a username. VMX populates the dialog with the username of the user that is currently logged, but you can change it to whatever you’d like. This is useful in cases where your Windows username is different from your Unix or Linux username.</td>
</tr>
<tr>
<td><code>%VMX_MMC_DIR%</code></td>
<td>If this variable is included, the location where the VMX MMC Console Extensions were installed will be inserted in the command line.</td>
</tr>
</tbody>
</table>
The command line option for PLINK that enables the SSPI-based connection to occur is `-c`. This option simply tells PLINK to use the current username when connecting as PLINK will try the sspi connection whenever the supplied username matches the current user.

The command line for a SSO connection to the selected system will therefore look similar to the following: `%PLINK% -c -ssh -X %HOSTNAME% <command>`. Use `%HOSTNAME%` not `%IPADDRESS%` for Active Directory to obtain the correct SSPI “ticket”.

**The Remote Tool Types**

PuTTY.exe, PLINK.exe, and “VNC Executable.exe” are the executables VMX uses to connect to the target systems. They each have their unique characteristics, as follows:

PuTTY is a client program for SSH, Telnet, and Rlogin network protocols. Using PuTTY you can run a command shell session on another computer. PuTTY implements the client end of that session: the end at which the session is displayed, rather than the end at which it runs. PuTTY allows you to remotely issue commands to a Linux or Unix shell.

A typical PuTTY command line has the following form:

```
%PUTTY% -ssh -X -l %USERNAME% %IPADDRESS%
```

PLINK (PuTTY Link) is a command-line connection tool similar to the Unix ssh. It is generally used to call specific, X-Windows based commands or applications. Unlike Putty, PLINK is not used to open an interactive command shell (unless you call a X-Windows based terminal emulator like xterm).

The following screen shot shows PLINK’s options:

You can use any of the above command line options together with the substitution variables in building your tool's command.
Note VMX highly recommends the use of the SSH protocol over the alternatives.

VNC (Virtual Network Computing) software makes it possible to view and fully-interact with one computer from any other computer or mobile device anywhere on the Internet in much the same fashion that the Windows-based remote control works. VNC is not delivered by VMX.

```
%VMX_MMC_DIR%ssh_vnc.vbs %PLINK% %IPADDRESS% "C:\VNC Directory" "VNC Executable.exe"
```

The `ssh_vnc.vbs` is a script authored by Quest that allows the VNC-based communication to occur through SSH.

For more information regarding setting up VNC on Macintosh clients and servers, see Configuring Remote Tools on page 71.

Using the Remote Tools Manager

The following sections describe how to use the Remote Tools Manager to administer your remote tool options.

Add Tool

Continuing from the steps outlined in Configuring Remote Tools on page 71, to add a remote tool to the VMX Client, follow these steps:

1. Click Add from the Remote Tool Tool Manager screen.
Note To see sample input for this dialog, see edit tool on page 76:

2. Enter the title of the tool in the Tool Name text field. This text displays on the shortcut menu when you right-click a VMX Client and select Remote Tools.
3. Enter a brief description of the tool in the Description field.
4. Enter the command line that executes the remote tool in the Command Line field. For a detailed explanation of the command line, see Building the Command Line on page 72.
5. Click OK to add the new remote tool to the Remote Tool shortcut menu.

Note The first time you open the shortcut menu for a system, there is a delay while it creates the default remote tools configuration. It writes the Remote Tools configuration to the SMS Site Control File (SCF) upon first use.

Edit Tool
To edit the configuration of a remote tool, click Edit from the Remote Tool Manager screen. The Remote Tool Configuration dialog displays:

The Remote Tools Configuration screen has the following text fields:

- **Tool Name**: The title of the tool.
- **Description**: A brief description of the tool.
- **Sub-Menu**: Currently, this field is not in use.
Command Line: The command line that executes the remote tool.

For a detailed explanation of the command line, see Building the Command Line on page 72.

Delete Tool

To delete a remote tool from the VMX Client, select the tool and click Delete from the Remote Tool Manager dialog.

Running Remote Tools

To launch Remote Unix Tools, follow these steps:

1. Start your X Server if you plan to launch some X based tools.
   A Windows-based X Server enables the UI to show the Unix tools on your Windows desktop. (See the note on the first page of this chapter for more information about X Servers.)
2. Expand Collections in the SMS Administrator Console.
3. Click the collection containing a VMX Client based system.
4. Right-click the desired VMX Client machine listed in the right-hand detail pane to open its shortcut menu.
5. Choose All Tasks | Remote Unix Tools to display the remote tools that are currently available:
Example Configurations

The following sections explain how to run the following Remote Tools:

- Running the Red Hat Services Configuration Tool tool
- Launching a PuTTY Session
- Starting the Top Process Monitor

Red Hat Services Configuration Tool

To run the default Red Hat Services Configuration tool, follow these steps:

1. Continuing from the steps above, choose Red Hat Services Configuration to display the Enter Username dialog:

   ![Enter Username dialog]

   Enter the username you want to connect to the remote system with. Your Windows username has been added for convenience. If this is the same as the Linux system username, simply press Enter.

   **Username**

   [root]

   ![Click OK]

2. Enter the username.
3. Enter the password to open the Red Hat Services Configuration report:

![Password Input](image)

### Launching a PuTTY Session

One of the remote tools VMX provides access to is PuTTY. To show you how to launch and run Remote Tools the example below establishes a remote session with PuTTY, which opens a command line window where you can execute system administration commands, run monitoring utilities, or perform other tasks.

PuTTY is a Telnet/SSH client for Windows systems that lets you establish a remote session with a Unix/Linux resource.

To establish a remote session with a VMX Client using PuTTY, follow these steps:

1. Expand **Collections** in the SMS Administrator Console.
2. Click the **All Linux Systems** group.
3. Right-click the VMX client machine, and open the shortcut menu.
4. Choose **All Tasks | Remote Unix Tools | PuTTY ssh shell**.
5. Enter the user name in the Enter Username dialog to connect to the remote system.
The Terminal Command Line window opens and prompts you for a password:

6. Enter the password at the command prompt. (Your keystrokes will not display as you type the password.)
PuTTY logs in to the remote system and starts the terminal session:

You are now connected to the remote system and can execute system administration commands, run monitoring utilities, or perform other tasks.

**Starting the Top Process Monitor**

To demonstrate how Remote Tools work, this section explains how to launch the Top Process Monitor from the command window.

The Top Process Monitor shows useful information about the current system state, such as:

- Uptime
- Connected users
- Running processes information, such as:
  - Total, sleeping, running, zombie
  - Process ID and name
  - Process owner
  - Memory and CPU time
- System memory usage
- System CPU usage
- System swap file usage (total, used, free, and cached)

Follow these steps to start the Top Process Monitor:
1. Enter `top` at the command prompt:

![Top Command Output]

2. For more information on Top, enter `?` for a list of Top commands.

3. Press any key to continue.

4. At a Linux command prompt, enter `man top` to display the Top documentation.
   - Enter `q` to quit the Top Process Monitor
   - Enter `exit` to end your remote terminal session.

**Troubleshooting Remote Tools**

Generally speaking, if you are experiencing problems running or configuring remote tools, make sure that you have entered in the command line correctly, there is network connectivity and that the tool is actually installed on the target Unix system.

If that doesn’t work, here are some other issues you may have with Remote Tools:

- **X Server Not Installed or Running on the Console Machine.**
  You must have an X Server installed and running on the console.

- **SSH Not Running on VMX Client.**
  Only Linux systems (Red Hat and Suse) and Solaris 9 have SSH installed on them by default. You must install SSH on AIX, HP-UX, OS X and older versions of Solaris clients manually.

- **Remote Tool Application Not Installed on VMX Client.**
  Install and test it on the client.

- **Firewall or Network Problems.**
  Check with your System Administrator.
Configuring Remote Access for the Macintosh

VNC (Virtual Network Computing) software makes it possible to view and fully interact with one computer from any other computer or mobile device anywhere on the Internet. VNC software is cross-platform, allowing remote control between different types of computers.

Go to: [http://www.redstonesoftware.com/vnc.html](http://www.redstonesoftware.com/vnc.html) and download a full featured VNC server that provides remote access to the GUI, keyboard, and mouse using any VNC client. Then set up the server side and the client sides, as follows:

To set up VNC Server on the remote machine, follow these steps:

1. On the **General** tab:
   - Display Number: Set to 0 if you are using port 5900 or 1 if you are using port 5901.
   - Port: Set to 5900 or 5901.
   - Password: Enter the VNC Password.
   - Display Name: (this is automatically entered for you.)

2. On the **System** tab: Leave all the defaults.

3. On the **Sharing** tab: Leave all the defaults, except: select **Only allow local connections (require SSH)**.
   
   Quest strongly recommends choosing this option for your Macintosh systems. The SSH-based connections option allows you to obtain maximum security with the VNC software. SSH is the secure connection method used by the VMX Remote Tools when connecting to your Macintosh system from the SMS Console. Configuring VNC to accept only SSH connections protects your Macintosh from unauthorized access through VNC.

4. On the **Startup** tab: Leave all the defaults and click the **Start Server** button.

To set up the VNC Viewer on the local client machine, go to: [http://www.realvnc.com/](http://www.realvnc.com/) and download the VNC client (viewer) for Windows and install it to the C:\VNC directory (or another directory of your choice). Then use the following steps to create a new remote tool called VNC Viewer. The Viewer uses a script to connect to the remote machine and forwards ports 5900 and 5901 to it. It connects to those ports on the local machine with the VNC Viewer:

1. Using the VMX Remote Tool Manager, open the VNC Viewer tool for editing:
   - i. On the SMS Administrator's Console, navigate to **Tools | All Tasks | VMX Remote Tools Configuration** to display the VMX Remote Tool Manager dialog.
   - ii. Select **VNC Viewer over SSH** and click **Edit** to display the Remote Tool Configuration dialog.

   The default Command Line looks like this:
   ```
   %VMX_MMC_DIR%ssh_vnc.vbs %PLINK% %IPADDRESS% "C:\VNC Directory" "VNCExecutable.exe"
   ```

2. Change the second to last parameter on the Command Line from the default of “C:\VNC Directory” to the actual location of the VNC Viewer directory.

3. Change the last parameter on the Command Line from the default of “VNC Executable.exe” to the actual name of the executable on the local machine.

To launch the VNC Viewer, follow these steps:

1. In SMS Administrator's Console, find the client machine that has the VNC Server running on it; right click it to open the shortcut menu.

2. Choose **All Tasks | Remote Unix Tools | VNC Viewer over SSH** to display the Enter Username dialog.
3. Enter the username to connect to the remote system.

4. Enter root password for the machine to which are trying to connect at the PLink Password dialog.

5. Enter **127.0.0.1:0** in the Connection Details field of the VNC Viewer Connection Details dialog. This is the local IP address for the Mac. (Note: The “:0” is the default if you are using Port 5900. Enter **127.0.0.1:1** if you configured your VNC Server to use Port 5901.)

6. Enter the VNC server password that you configured when you set up the VNC server in the VNC Server Password dialog.

You now have remote control access.

An error displays for the following conditions:

- △ If the SSH connection fails.
- △ If the VNC connection fails
- △ If a VNC session is already running; multiple VNC Viewer sessions are NOT allowed because the SSH port forwarding will only work with one host at a time.
Reports

Vintela Management Extensions uses existing SMS data and data structures so you can immediately generate reports about your Unix resources without having to create any new or different ones. Of course, you can create new reports, if you desire.

Access Unix resource reports just like Windows resource reports. Since Vintela Management Extensions uses the same data for Unix resources as SMS does for Windows resources, you do not need to rewrite reports or discard existing ones; there is no additional report configuration or setup necessary; no additional tasks required. For example, viewing the Count Processor Speeds report automatically shows all Unix resources, together with Windows resources.

**Working with Reports**

1. In the SMS Administrator Console, expand **Reporting**.
2. Click **Reports**.
3. Locate, then right-click a report name in the report list.
4. Choose **All Tasks | Run | [Resource Name]**.
5. Enter the User name and Password to login to SMS Reporting.
   SMS displays the View SMS Reports page for this report in your web browser:

6. Enter the host name of the Vintela Management Extensions Client machine in the Computer Name field.
7. Click **Display** to display the report for the Vintela Management Extensions Client machine:
Troubleshooting Vintela Management Extensions Reports

Reports works just like it does in SMS. The only difference is that the data in the fields may not look the same as it does for Windows. Please refer to Chapter 11, Creating Reports in the Microsoft Systems Management Server 2003 Operations Guide for any troubleshooting issues you might encounter.
System Status

The SMS System Status assists you in monitoring and troubleshooting your SMS sites. System Status contains status summaries for Advertisement Status, Package Status, Site Status, and Status Message Queries.

Using System Status with Vintela Management Extensions is the same as for SMS. For more information, see Chapter 14, Using the SMS Status System in the Microsoft Systems Management Server 2003 Operations Guide.

Vintela Management Extensions Client Status Message

Vintela Management Extensions does not update all status message queries. The following lists the Message #, the WMI Class Name, and the Descriptions of the Status Messages that Vintela Management Extensions uses:

Table 2: Vintela Management Extensions Client Status Messages

<table>
<thead>
<tr>
<th>Status Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>110800: SMS_RemoteClientInstalled</td>
<td>The SMS Advanced client was installed.</td>
</tr>
<tr>
<td>10019: SoftDistOfferRejectedExpiredEvent</td>
<td>Advertisement was rejected because it has expired.</td>
</tr>
<tr>
<td>10002: SoftDistOfferReceivedEvent</td>
<td>Advertisement was received.</td>
</tr>
<tr>
<td>10025: SoftDistDownloadFailedEvent</td>
<td>The download of the package files failed.</td>
</tr>
<tr>
<td>Event Code</td>
<td>Event Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>10053: SoftDistAdvertDownloadFailedEvent</td>
<td>The advertisement failed due to the package download's failure.</td>
</tr>
<tr>
<td>10050: SoftDistErrorInsufficientCacheEvent</td>
<td>The cache directory did not have enough free space in which to store the package.</td>
</tr>
<tr>
<td>10051: SoftDistErrorNoContent</td>
<td>The content for the package could not be located.</td>
</tr>
<tr>
<td>10037: SoftDistProgramWaitingForAnotherProgram</td>
<td>The program is waiting for another program to finish.</td>
</tr>
<tr>
<td>10006: SoftDistProgramErrorEvent</td>
<td>The program for the advertisement failed.</td>
</tr>
<tr>
<td>10023: SoftDistDownloadStartedEvent</td>
<td>Content download for the package has started.</td>
</tr>
<tr>
<td>10024: SoftDistDownloadCompletedEvent</td>
<td>Content download for the package has completed.</td>
</tr>
<tr>
<td>10004: SoftDistProgramUnableToExecuteEvent</td>
<td>The program for the advertisement could not be executed.</td>
</tr>
<tr>
<td>10005: SoftDistProgramStartedEvent</td>
<td>The program for the advertisement has been started.</td>
</tr>
<tr>
<td>10070: SoftDistProgramExceededTime</td>
<td>The program for the advertisement did not finish in the allotted time and was cancelled.</td>
</tr>
<tr>
<td>10008: SoftDistProgramCompletedSuccessfullyEvent</td>
<td>The program for the advertisement completed successfully.</td>
</tr>
<tr>
<td>10900: SWMtr_Status_WMIFailure</td>
<td>Error Occurred Examining Current Metering CIM Instances.</td>
</tr>
<tr>
<td>10902: SWMtr_Status_Send_UsageReport_Failed</td>
<td>Upload of Meter report failed.</td>
</tr>
<tr>
<td>10605: CLIMSG_INV_INFO_GENERIC_COLLECTED_SUCCESSFULLY</td>
<td>Software inventory collected successfully</td>
</tr>
<tr>
<td>10651: CLIMSG_FILECOLL_WARNING_FILECOL_MAXIMUM</td>
<td>Maximums reached in file collection.</td>
</tr>
<tr>
<td>10652: CLIMSG_FILECOLL_WARNING_FILECOL_MAXIMUM</td>
<td>Error in execution of file collection processing.</td>
</tr>
<tr>
<td>10907: SWMtr_Status_Received_MeterRule</td>
<td>Software Metering Rule successfully received.</td>
</tr>
</tbody>
</table>

### Using System Status

For a step-by-step example of using System Status to verify software distribution, refer to [Verifying Software Distribution](#) on page 55.
Vintela Management Extensions uses the regular SMS Backup and Recovery tools to help you quickly and reliably complete tasks that are essential to your overall backup and recovery strategy. Backing up a site is automated in SMS 2003 by the integrated Backup SMS Site Server task. Use the SMS backup task to regularly back up data such as SMS files, registry keys, and configuration information from your site server, and from your site database server or provider server if necessary. SMS 2003 also provides recovery and repair tools. Vintela Management Extensions does not store any additional information outside of the SMS database and Site Control File. Thus, as long as you use the normal SMS procedures there is nothing else to do for Vintela Management Extensions.


The VMX Client

Each Unix system under SMS management runs the VMX Client software. It provides all of the data and functionality necessary to manage the Unix system.

The VMX Client consists of two primary components: OpenWBEM and Feature Providers. Although Feature Providers run under the auspices of OpenWBEM, they are treated as a single separate component for the purposes of this discussion.

OpenWBEM

Web Based Enterprise Management (WBEM) is an industry initiative to provide management of systems, networks, users, and applications across multiple vendor environments. WBEM simplifies system management, providing better access to both software and hardware data that is readable by WBEM-compliant applications. OpenWBEM is an enterprise-grade, open-source implementation of WBEM, and is a core component of the VMX Client. It is a clean, well-designed, solid, open source, Quest-maintained implementation of a CIM Object Manager (or CIMOM) that complies with the DMTF-developed WBEM standard. DMFT stands for Distributed Management Task Force. (See: www.dmtf.org for more information.) Quest Software is the primary author / maintainer of the project and owns many of the related copyrights.

There are several things that make OpenWBEM a key component in the VMX Client, and the following sections discuss these in detail. But as the core of the VMX Client, here's the bottom line:

- OpenWBEM is a stable, mature, CIMOM based on WBEM standards that has been implemented to deliver the data and control exactly as you would expect.
- OpenWBEM is no risk to use: Quest owns the copyrights to and is the principle maintainer of the software. Licensing is covered by the BSD Style License; information on it can be found at: http://www.openwbem.org/license.html.
- OpenWBEM carries with it full Quest customer support: If you have an issue with the VMX Client implementation of OpenWBEM, Quest will solve it.

The next sections discuss the following about OpenWBEM:

- OpenWBEM as Open Source Software
- OpenWBEM Maturity
- OpenWBEM Flexibility

OpenWBEM As Open Source Software

OpenWBEM is open source software. It won Best Open Source Project at the January 2002 Linux-World Expo. One of the reasons it is a stable, mature client is because the open source nature of the software allows for a wide community of users and reviewers. Many developers from different organizations and companies work with the source and as they find issues or create new functionality, their contributions are added to the project. This makes OpenWBEM much more robust, stable, and solid than proprietary WBEM offerings.
Several commercial organizations have either adopted it or are in the process of adopting it as the basis for the agent technology portion of their management system. Moreover, OpenWBEM is licensed under the BSD Style License. Also, Quest maintains OpenWBEM and therefore incurs no risk in using it as the core component of the VMX Client.

**OpenWBEM Maturity**

OpenWBEM has been in existence since late 2000, and because of the open source nature of the software and the dedicated and talented developers working on it, it has the most functionality of any similar system. Here is a partial list of its major functionality:

- CIM operations of HTTP and HTTPS (see Listeners in Table 3 below)
- Extensible authentication (see Security/Authentication in Table 3 below)
- Fast scalable hierarchical repository with indexing
- Extensible Provider interface
- WQL Level 2 support for server and client
- Access control lists
- SLP integration

For more information on OpenWBEM functionality, see: [www.openwbem.org](http://www.openwbem.org).

**OpenWBEM Flexibility**

There are several things that make OpenWBEM the most flexible WBEM-based management software. First it has been ported to a number of Unix and Linux system.

Linux Platforms:

- **Distributions:**
  - RedHat
  - Debian
  - SUSE
- **Architectures:**
  - i386
  - IBM s390
  - MIPS

Other Platforms:

- UnixWare 7
- Solaris 8, 9, and 10
- Macintosh OS X
- AIX
- HP-UX

Second, OpenWBEM is designed for extensibility and contains several pluggable components, including listener/communication, authentication, logger, and scheduler modules. This means that OpenWBEM can easily accommodate new technologies and changing needs.
These are the extensible and pluggable OpenWBEM components:

**Table 3: Pluggable OpenWBEM Components**

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listeners</td>
<td>OpenWBEM currently provides standards-based listeners (HTTP and HTTPS), and can support other proprietary communications protocols when necessary and when greater efficiency is desired.</td>
</tr>
<tr>
<td>Security/Authentication</td>
<td>OpenWBEM provides authentication through pluggable authentication modules. Some of the authentication modules are: Access Control Lists (ACLs) and Public Key Infrastructure (PKI). OpenWBEM also supports Kerberos by means of the Quest Software Vintela Authentication System (VAS).</td>
</tr>
<tr>
<td>Dynamic CIM Schema</td>
<td>OpenWBEM supports dynamic CIM Schema: that is, a custom schema can be designed and installed into the OpenWBEM CIMOM; it will recognize and begin to operate on it without restarting or recompiling the system.</td>
</tr>
<tr>
<td>Providers</td>
<td>Providers are an integral and powerful part of OpenWBEM's robustness and flexibility. A provider can represent nearly anything to a management system, from simply providing hardware inventory data to an OpenWBEM repository to controlling different parts of a system. OpenWBEM providers also are dynamically loadable. Also, anyone can write their own providers using a variety of languages, including C, C++, and Perl.</td>
</tr>
</tbody>
</table>

**Feature Providers**

This section discusses the following providers that VMX adds to OpenWBEM to create the VMX Client.

- Hardware and Software Inventory Providers
- Software Distribution Providers
- SMS Poller Providers

**Hardware and Software Inventory Providers**

The VMX Client contains Hardware and Software Inventory Providers that collect hardware, file, and product installation data. The providers take care of the system-specific functions necessary to collect the data. There is typically one provider for each hardware CIM class, including the Add/Remove Programs class. For example, when you launch a Resource Browser on a managed Unix system, the data collected by each Hardware Inventory Provider display in the Resource Browser, just like it does for a Windows-based resource.
Software Distribution Providers

The Software Distribution Providers deploy programs and packages to the Unix systems on which they reside. This includes support for native installation mechanisms, such as the Redhat Package Manager, the Solaris Packaging system, as well as a general, script-based delivery mechanism. The native versions all have corresponding options in the VMX Software Distribution Wizard in the administrator's console. The scripted-based version uses the General Unix software distribution method in the wizard which requires you to enter the name of a script file. It runs the script file along with any supporting files such as tarballs and properly installs the desired software.

SMS Poller Provider (MP Communication Provider)

The SMS Poller Provider is a special Provider that enables VMX Client communication with your SMS Management Points. It is the part of the VMX Client that makes OpenWBEM act like an SMS Advanced Client.

Each managed Unix Resource's VMX Client communicates with your Management Point by using the Client's Poller Provider. It communicates with the VMX Management Point Extension to seamlessly provide data and control to SMS.

Using the VMX Client Tool

Quest has provided vmxclienttool (the VMX Client Tool) to help you perform maintenance tasks. You can use it to force policy update, run hardware inventory and to run software distribution before its regularly scheduled process time.

vmxclienttool requires root access. It authenticates a user based on file system permissions and does not require a password. Anyone with an euid of root may use the vmxclienttool. If you use sudo or another similar tool, you can also run vmxclienttool.

Locate vmxclienttool in the following path on the VMX Client: /opt/vintela/vmx/bin/

You can run vmxclienttool without parameters to list its options, as follows:

```
# /opt/vintela/vmx/bin/vmxclienttool
```

The vmxclienttool options are as listed, as shown it the following screen shot:
Warning: be aware that if you use `vmxclienttool` to force a software distribution to a VMX Client before the advertisement is scheduled, you may get messages indicating that the software is already installed when the VMX Client runs the software distribution at the regularly scheduled time.

### Stopping and Restarting the VMX Daemon

When troubleshooting the VMX Client, it might be necessary for you to stop and restart the VMX daemon in order to pick up configuration changes. Use `vintela-vmx` (the VMX init script) to stop, start, restart, and check the status of the VMX Client:

```
[root@cslin1 bin]# /etc/init.d/vintela-vmx
Usage: vintela-vmx {[re]start|stop|reload|force-reload|status}
[root@cslin1 bin]#
```
The vintela-vmx options are defined, as follows:

**Table 4: vintela-vmx Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vintela-vmx start</td>
<td>Start the service.</td>
</tr>
<tr>
<td>vintela-vmx stop</td>
<td>Stop the service.</td>
</tr>
<tr>
<td>vintela-vmx restart</td>
<td>Stop and restart the service if the service is already running, otherwise</td>
</tr>
<tr>
<td></td>
<td>start the service.</td>
</tr>
<tr>
<td>vintela-vmx reload</td>
<td>Reload the service configuration without actually stopping and</td>
</tr>
<tr>
<td></td>
<td>restarting the service.</td>
</tr>
<tr>
<td>vintela-vmx force reload</td>
<td>Reloads the service configuration if the service supports this, otherwise</td>
</tr>
<tr>
<td></td>
<td>stops the service, if it is running, and then restarts it.</td>
</tr>
<tr>
<td>vintela-vmx status</td>
<td>Displays the current service status: running or not running; enabled or</td>
</tr>
<tr>
<td></td>
<td>not enabled; or, dead.</td>
</tr>
</tbody>
</table>

The following lists the vintela-vmx locations for each operating system that VMX supports:

**Table 5: vintela-vmx Locations**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux and Solaris</td>
<td>/etc/init.d/vintela-vmx</td>
</tr>
<tr>
<td>HP-UX</td>
<td>/sbin/init.d/vintela-vmx</td>
</tr>
<tr>
<td>AIX</td>
<td>/opt/vintela/vmx/sbin/vintela-vmx</td>
</tr>
<tr>
<td>Macintosh</td>
<td>/Library/StartupItems/VintelaVMX/VintelaVMX</td>
</tr>
</tbody>
</table>

The following are examples of using the vintela-vmx with each of the supported operating systems:

```
/etc/init.d/vintela-vmx stop
/sbin/init.d/vintela-vmx start
/opt/vintela/vmx/sbin/vintela-vmx restart
/Library/StartupItems/VintelaVMX/VintelaVMX status
```

**VMX Policy Spy Tool**

VMX includes a troubleshooting tool called the VMX Policy Spy that allows you to view the policies on the VMX Client. This is not a tool that views the Policies on the SMS side; it doesn't look at the SMS database. It looks at what actually got down to the VMX Client and what the OpenWBEM CIMOM thinks it should be doing.
Note This tool is just a policy viewer. You cannot affect any changes using this tool. It allows you to verify the policy on the client.

To access the VMX Policy Spy tool, follow these steps:

1. Right click an appropriate client to open the shortcut menu.

2. Choose All Tasks | VMX Policy Spy to display the VMX Policy Spy dialog.
3. Edit the Host Name or SSH port fields, if necessary.
4. Enter in the User Name.
5. Select **Use: Sudo**, if necessary.

**Note** To run the Policy Spy the user needs to have root privileges. *sudo* is a tool that gives users root privileges. So the user can either use “root” for their username or they can use another username with the sudo option. But, you need to configure the sudo functionality on the Unix system for this to work properly.

6. Enter the Password.
7. Click the **Show Policies** button.
   The “Retrieving Policies” message displays while it gathers the policy information.
8. Select **List** to view the details of all the policies.
   The Tree view lists the policies by instance category in the upper window.
9. From the Tree view, expand a policy and click the policy ID number to view the detail of the selected policy in the lower window.
10. From the Action menu, select **Export to file...** to open the Save As dialog to copy the details of all the policies out to a file.

**Note** The Export to file option always exports all the policy information no matter what view you are in. This way, to simplify troubleshooting, you can send the policy information to Support for analysis.
VMX Integration with VAS

The Quest Software Vintela Integrated Architecture component for integrating Unix systems with Active Directory is Vintela Authentication Services from Quest Software or VAS. VAS extends the identity management capabilities of Microsoft Active Directory to non-Windows systems. It allows administrators to join Unix systems to Active Directory domains to store, access and easily manage information.

For customers that do not have the full version of VAS already installed, VMX automatically installs a limited version of VAS, called vas-host which allows you to join Unix systems to Active Directory. VMX can then use the Active Directory identity for the Unix system and the VAS LDAP capabilities to query Active Directory for information relating to the Unix systems and SMS.

It's important to note that vas-host only provides a subset of the full VAS feature set: only those options that are directly relevant to joining Unix systems to Active Directory. For the user and group management features, you must use the full version of the VAS client. VMX is compatible with both VAS versions — if the full VAS client is already installed, it will not attempt to install the vas-host packages.

Like the full version of VAS, vas-host has a command line tool, vastool, that provides an option called join that adds a Unix system to Active Directory.

To run vastool join, the root user must enter the following at a shell prompt:

```
# /opt/vas/bin/vastool -u matt join example.com
```

Where matt is the username of an Active Directory user with sufficient administrative privileges to create a computer object in Active Directory (normally a user who is a member of the Domain Admins group), and example.com is the name of the Active Directory domain to which you are joining the computer. When prompted for the user's password, enter it on the command line. The results of vastool join display.

For more information about using the vastool command, please see VASTOOL MAN PAGE on page 101.
VASTOOL MAN PAGE

vastool

Name
vastool – A command line administration tool for use with vascd.

Synopsis
vastool_command [vastool_command arguments]

Description
vastool is a command line program that allows you to configure vascd; access information stored in Active Directory; and to store information in Active Directory. vastool is located at /opt/vas/bin/vastool. It has been designed to be script-friendly and to allow administrators to easily manage computer objects and other information stored in Active Directory from Unix/Linux workstations.

To run vastool, you must specify the options for vastool, a command to run, and the options for that specific command. The following is a list of supported vastool commands and a brief description of each command’s purpose. A more detailed explanation of each command will follow later.

configure—Update VAS configuration files.
flush—Flush the vascd cache.
info—Provides information about the hosts Active Directory configuration.
join—Adds a computer object to Active Directory, and starts vascd.
kinit—Performs kinit functions—obtains Kerberos ticket(s) for service(s).
klist—Performs klist functions—lists the Kerberos tickets stored in the calling user’s credentials cache.
kdestroy—Performs kdestroy functions—destroys all existing tickets in the calling user’s credentials cache.
ktlist—List Kerberos keys contained in a Kerberos v5 keytab.
realms—Detect the realms (domains) on your network and the servers providing LDAP and Kerberos services for those realms.
service—Manage service accounts in Active Directory.
timesync—Query and synchronize system time with Active Directory or other specified time server.
unconfigure—Update PAM, NSS, and other configuration files to not use the VAS components.
unjoin—Configure the system to not use the VAS client for authentication and for NSS, and then removes the computer object from Active Directory.
**vastool Options**

These are the options you can pass to vastool. They must be specified before the command name.

**-h [command]** — If no command is specified, it shows the vastool usage and a list of available commands. If a command is specified, it shows the usage for that vastool command.

**-v** — Print out the vastool version and exit.

**-u** — principal Sets the principal name to authenticate as when the vastool command needs to access Active Directory. If the caller has root access, "host/" can be specified and vastool will authenticate as the computer object that vastool is running on.

If -u is not used, then vastool will authenticate as the calling user, and will attempt to reuse Kerberos tickets from the user’s credentials cache. If -u is specified, then no existing credentials cache will be used, and new tickets obtained will not be saved to disk.

**-p password** — This is option is used to specify a password when loading users. All new users will have their passwords set to the specified password.

**-w password** — This option allows you to pass in a password on the command line. Please note that this is a security hole in a production environment. This option should only be used in testing environments.

**-s** — This option will cause vastool to not prompt for any initial passwords, but instead read them from stdin. This allows you to use some vastool commands in a non-interactive mode.

**-f** — By default, vastool will delete whatever tickets are obtained during it’s operation if an alternate username is specified using the -u option. -f will cause vastool to not flush the ticket cache, but keep them in a disk based ticket cache for the calling user.

**-d** — This option will enable debugging output to stderr. This is useful when troubleshooting problems.

**-glevel** — This allows you to override the default debugging level that vastool uses when the -d option is specified. By default, the debug level is 2. It can be set anywhere from 1 to 5, with a higher level producing more debug output.

**vastool Command Synopsis**

The following is a detailed description of all the available vastool commands. Their usage descriptions, a detailed explanation of their purpose, how they work, and examples are included for each command.

**vastool configure**

vastool configure can be used to modify the VAS Kerberos realm configuration. This command must be run as root.

```
vastool [vastool options] configure realm realm name [server...] |
extra-realm realm name server... | computer-name name [-g] [service...]
```

Note that these commands are for advanced usage only. The vastool join command will automatically perform these steps for you.

Configuring the realm will modify `/etc/opt/vas/vas.conf` to use the given realm name as your default realm. If a list of server names is passed in, these servers will be stored as the servers for the given realm. In Active Directory terms, the realm will be the domain name of the domain of which this computer is a member.
vastool configure extra-realm can also be used to configure other domains if you need to support multiple servers in your Active Directory tree. This will add information for these realms, but it will not make the new realm the default realm.

If you are configuring a computer whose hostname will not always match up with the name of the computer object in Active Directory that represents your computer, you must use vastool configure computer-name to specify the name of the computer as it appears in Active Directory. Otherwise, your computer will not be able to communicate with Active Directory. When using vastool join you can specify this name with the -n option.

Following are examples of configuring the example.com realm, configuring the extra-realm example.com with a specific server, and using a special name for the host computer.

```
vastool configure realm example.com
vastool configure extra-realm sub.example.com server.sub.example.com
vastool configure computer-name mycomputer
```

**vastool flush**

vastool flush can be used to clear the vascd cache. This command must be run as root.

```
vastool [vastool options] flush keytab | statedir
```

Flushing the keytab will delete the VAS keytab file. Flushing the statedir will delete all of the VAS state information. Flushing the keytab and statedir will not recreate their respective information, and is only used during uninstallation of the vas client.

If you do not specify an argument to vastool flush, it does nothing.

**vastool info**

vastool info is used to collect information about the host’s Active Directory configuration.

```
vastool [vastool options] info site | domain | forest-root-dn | root-dn | id | servers [domain]
```

The info command provides information about the host’s Active Directory Site to which this host belongs to.

The domain option returns the name of the Active Directory Domain to which the host machine is joined.

The root-dn option returns the forest root of the domain to which you are joined. If you were joined to example.com, root-dn would return DC=example,DC=com. The forest-root-dn will return the DN of the Forest Root domain.

The id option shows the Active Directory location of the computer object.

The servers option will show all of the available servers for a given domain which are in the Unix host’s Active Directory site.

Following is an example of how to use vastool info to determine the host machine’s default site.

```
vastool info site
```
**vastool join**

vastool join is a convenient command that wraps all of the necessary steps to configure the VAS client on a computer into one. It configures your Active Directory domain and creates a computer object in Active Directory. This command must be run as root.

```

vastool join will internally call vastool configure realm to configure the domain, vastool create to create a computer object in Active Directory for the computer, and vgptool apply to license VAS using any existing group policy objects. The vascd client daemon will then be started. For more information on each of these steps, see their respective sections in this document.

The `–c` option will allow you to specify a container where your new computer object will be created. If that is not specified, then the computer object will be created in the default computers container. The `–n` option allows you to specify a different name for the computer object than what vastool would generate from your hostname. If `–n` is used, then a special parameter will be set in `/etc/opt/vas/vas.conf` to denote to all the VAS components which name should be used for the computer object. You can also set this name by running vastool configure computer-name. The computer name specified with the `–n` option should not be in the "host/FQDN" form—it should just be a name which is an alternative to the system’s hostname. If the name specified does not have a ‘.’ character in it, then the domain name will be appended to the specified computer name to create a FQDN.

Following the options, you must specify your Active Directory domain, which will act as your Kerberos realm. The services for this domain will be automatically detected through DNS and LDAP lookups. If you do not intend to use DNS, or it has not been configured to support Active Directory SRV records, you must specify a space separated list of domain controllers for the domain you are joining after the domain argument.

If a computer object already exists in the directory for the computer name you are trying to use, an error will be reported. To override the existing computer object, use the `–f` option. In this case, the computer object’s authentication key will be reset. Any other systems authenticating as that computer object will no longer be able to authenticate after the authentication key is reset.

If VGP (Vintela Group Policy) is installed on the host, vastool will use vgptool to apply all configured Group Policy settings after the join is successful. This may include automatic licensing of VAS. You can disable this behavior by using the `–l` option. For more information about using VGP with VAS, see the VGP Administrators Guide.

The following is an example of vastool join using all of the defaults, then an example of joining a computer with a name other than it’s hostname into a non default container in an environment where DNS is not properly configured.

```
vastool -u admin join example.com server.example.com
vastool -u admin join -c "OU=Testlab,DC=example,DC=com" -n test server example.com
server.example.com
```
**vastool kinit**

vastool kinit can be used to obtain Kerberos tickets.

```
vastool [vastool options] kinit [service...]```

If no arguments are specified, then the Kerberos TGT is obtained if it is not in the user’s ticket cache. If services are specified, then those tickets will be obtained. If the `-u` vastool option was not specified, then these tickets will be stored in the user’s ticket cache.

vastool kinit can be used to debug problems with Kerberos authentication. For example, to test if vascd can authenticate to Active Directory, you would run as root:

```
vastool -u host/ kinit```

Using the vastool `-s` option, you can use the vastool kinit command as an authentication API from scripts and other programs which do not use PAM or the VAS API. This can be done by running:

```
vastool -u jdoe -s kinit```

and then writing jdoe’s password to stdin of the vastool process. The exit code of the vastool process will be 0 on a successful authentication, and non-zero if authentication failed.

If you see any error messages, then vascd could not authenticate to Active Directory.

**vastool klist**

vastool klist can be used to list all of the tickets currently in the calling user’s Kerberos ticket cache.

```
vastool klist [-f] [-v] [-c cache]```

The tickets in the user’s ticket cache are printed to stdout. They will show the name of the service each ticket is for, the time the ticket was issued, and the time the ticket will expire. The ticket cache will be stored as a file owned by the user with permissions of 0600 at `$HOME/.krb5cc` or in `/tmp/krb5cc_{user’s UID}`.

The `-f` option will show the flags that apply to each ticket. The `-v` will list more details about each ticket.

**vastool kdestroy**

vastool kdestroy will destroy all of the tickets that are in the calling user’s Kerberos ticket cache.

```
vastool [vastool options] kdestroy```

A user’s Kerberos ticket cache is a file owned by the user with permissions of 0600 that will be either at `$HOME/.krb5cc` or in `/tmp/krb5cc_{user’s UID}`. Normally, the user’s Kerberos TGT is stored there along with any other tickets that have been obtained. These tickets can all be cleared with vastool kdestroy.

**vastool ktlist**

vastool ktlist can be used to list all of the keys in a Kerberos keytab file. The ability to directly manipulate the keys is not provided, since that ability is provided through the vastool join and vastool service commands.

```
vastool ktlist -v keytab```

Each key in the given keytab is printed to stdout. This command is provided mainly for troubleshooting purposes. If no keytab is listed, then the default keytab configured in the vas.conf file will be listed.

The `-v` option will show extra information for each key, including the date and actual key value.
**vastool realms**

**vastool realms** can be used to query the network for the Active Directory domains and also will detect the domain controllers on the network. This information can be stored in the realms cache. To do this you must be root.

```
vastool [vastool options] realms find root | srvs | domains [realm] |
        site names | srvs | local | subnet | cache list | update |
update-realm realm | flush [realm] | toconf
```

**vastool realms find** will detect various settings, services, and domains on your network. **realms find root** will detect the forest root. **realms find domains** will detect all of the domains in the entire forest. **realms find srvs** will detect the services for a given domain.

**vastool realms site names** will detect all of the configured Active Directory sites in your forest.

**vastool realms site srvs** will list all of the domain controllers and which site they are in. **vastool realms site local** will detect what site to which the local VAS client should belong. **vastool realms site subnet** will determine the subnet of your client.

**vastool realms cache list** will print out the service and domain information that is stored in the realms cache. This cache is used to decrease the amount of DNS traffic that must be performed by the VAS client. **vastool realms cache update** will detect all of the domains and services in the entire forest and store that information in the realms cache. **cache update-realm** will detect the services and update the cache for only the given realm.

**vastool realms flush** will clear all of the entries from the realms cache, and reload it. If a realm name is specified, then only the entries for the given realm will be reloaded.

When using VAS with MIT Kerberos compatible applications, you will need to symlink `/etc/krb5.conf` to `/etc/opt/vas/vas.conf`. Before doing that though, you need to make sure that all of the realm information in the VAS realms cache is stored in `/etc/opt/vas/vas.conf`. This can be done with **vastool realms toconf**. This will make a realms entry for each realm VAS knows about in the [realms] section. This way the MIT-compatible applications will be able to work by reusing the VAS configuration information.

**vastool service**

```
vastool [vastool options] service create [-c container] name spn... |
        delete name | list name
```

**vastool service** can be used to create and delete service accounts in Active Directory. An Active Directory service account is a user account which is intended to be used by services running on Unix hosts. When a service account is created, a random password is generated for the account and a Kerberos keytab is created for the service.

Each service has a User Principal Name (UPN), and an optional set of Service Principal Names (SPN’s). The UPN is typically named `service/host@domain`, where service matches the type of service running—for example, `http/` or `ftp/`. The keytab file created for the service will be named `service.keytab` and will be created in the VAS configuration directory at `/etc/opt/vas`. The default permissions on the keytab file will be 0600 and the file will be owned by root. You should update the ownership of the file so that the corresponding service has the rights to read from the keytab file.
To create a service account, you must run `vastool service create name` as root where name is the service account name. By default, the service account will be created in the default computers container.

You can override this location by using the `-c` option to specify an alternate OU to create the service account in. If you specify service/ as the principal name, then the hostname of the machine the command is run will be used to build a complete service principal name. You must supply the username and password of an Active Directory user that has permissions to create users. You can add an optional list of other servicePrincipalName’s to the account. An example of create a service account for an SQL server is:

```
vastool -u admin service create sql/
```

To delete a service account, run `vastool service delete name`. The account in Active Directory will be deleted, and the keytab file for the service will be deleted. For example, to delete the sql service account, run:

```
vastool -u admin service delete sql/
```

You can list the service principals associated with a Service account with `vastool service list service`. To list the principals associated with the sql service account, do the following:

```
vastool -u admin service list sql/
```

**vastool timesync**

`vastool timesync` can be used to query the time on the Active Directory server and synchronized the system clock with the Active Directory server. In order to set the system clock this command must be run as root.

```
vastool [vastool options] timesync [-q] [server]
```

Running `vastool timesync` without specifying a time server will automatically use the Active Directory server that was configured using the `vastool join` or the `vastool configure realm` command.

Use the `-q` option to query the server’s time without setting the system clock.

**vastool unconfigure**

`vastool unconfigure` can be used to remove the VAS configuration from the NSS and PAM subsystems. This command must be run as root.

```
vastool [vastool options] unconfigure nss | pam [service...]
```

**vastool unjoin**

`vastool unjoin` is a convenient command that wraps all of the steps to remove your computer object from Active Directory and to remove the VAS configuration from the NSS and PAM subsystems in one step. This command must be run as root.

```
vastool [vastool options] unjoin [-n computer name]
```

`vastool unjoin` will first prompt you for your administrative password to delete the computer object for your machine in Active Directory. As part of this process, `vascd` will be stopped.
If you used the -n option in the vastool join process, then you need to specify that same name that you used in the join in the vastool unjoin process.

The following are examples of unjoining the machine where the computer object is named after the system hostname, and unjoining the machine when the computer object name does not match the hostname.

```
  vastool -u admin unjoin
  vastool -u admin unjoin -n computer name
```

See Also

vascd(1)

Authors

Copyright 2005 Quest Software, Inc. All Rights Reserved
Index

A
Advanced Inventory Collection, 17

C
CIM, 3
CIM (Common Information Model), 1
Collected Files, 14
Collections, 21
Command Line Substitution Variables, 72

D
Distributed Management Task Force (DMTF), 91

F
File Collections, 10
File Details, 15
File details, 14
Formatting Conventions, vii

H
Hardware Inventory, 5

I
Integrating Unix systems with Active Directory, 99
Inventory Collections, 10, 13

J
Joining Unix systems to Active Directory, 99

L
Last Software Scan, 15

M
Maximum run time option, 46
MOF Files, 17
MOF-based Extended Hardware Inventory, 18

N
Non-Computer Related Hardware Classes, 17

O
OpenWBEM, 3
OpenWBEM log file, 18

P
Product Detail, 15
Product details, 14

Q
Queries, 21

R
Remote Tools Manager, using, 75
Remote Tools, running, 77
Resource Explorer, 5, 18

S
Script-based software inventory, 19
sms_def.mof, 18
Software Inventory, 5, 9
Standards, Quest, 1
status messages in VMX Client, 87
V
VAS integration, 2
vastool join, example of usage, 99
Vintela Integrated Architecture, 1
VMX Client Status Messages, 87
VMX Inventory Collections tab, 14
VNC (Virtual Network Computing), 75

W
WBEM, 3
WBEM (Web-Based Enterprise Management), 1
Windows Management Instrumentation (WMI), 3
WMI (Windows Management Instrumentation), 3

X
X Server, 77