IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines
7.2 Fix Pack 6

Troubleshooting Guide

Note

Before using this information and the product it supports, read the information in “Notices” on page 33.

This edition applies to version 7.2.0.6 of IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines (product number 5724-L92) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Troubleshooting basics

To troubleshoot a problem, gather information about the problem for IBM® Software Support, use logging data, and consult the lists of identified problems and workarounds.

For general troubleshooting information, see the *IBM Tivoli Monitoring Troubleshooting Guide*. For other problem-solving options, see "Support information" on page 29.

You can resolve some problems by ensuring that your system matches the system requirements. The most up-to-date requirements are in the Software product compatibility reports (http://publib.boulder.ibm.com/infocenter/prodguid/v1r0/clarity/index.html).

The following activities can help you find a solution to the problem you are having:

- "Gathering product information for IBM Software Support" on page 1
- "Using logging" on page 2
- "Consulting the lists of identified problems and workarounds" on page 2

Gathering product information for IBM Software Support

Before contacting IBM Software Support about a problem you are experiencing with this product, gather the information shown in Table 1 on page 1.

Table 1. Information to gather before contacting IBM Software Support

Information type	Description
Log files	Collect trace log files from failing systems. Most logs are located in a logs subdirectory on the host computer. See "Principal trace log files" on page 3 for lists of all trace log files and their locations. For general information about the IBM Tivoli® Monitoring environment, see the *Tivoli Enterprise Portal User's Guide*.
Linux Kernel-based Virtual Machines information	Version number and patch level
Operating system	Operating system version number and patch level
Messages	Messages and other information displayed on the screen
Version numbers for IBM Tivoli Monitoring	Version number of the following members of the monitoring environment: • IBM Tivoli Monitoring. Also provide the patch level, if available. • Linux Kernel-based Virtual Machines agent
Screen captures	Screen captures of incorrect output, if any
(UNIX systems only) Core dump files	If the system stops on UNIX systems, collect the core dump file from the *install_dir*/bin directory, where *install_dir* is the directory where you installed the monitoring agent.

You can use the pdcollect tool to collect the most commonly used information from a system. This tool gathers log files, configuration information, version information, and other data. For more information about using this tool, see "pdcollect tool" in the *IBM Tivoli Monitoring Troubleshooting Guide*.

For information about working with IBM Software Support, see IBM Support Portal Service Requests and PMRs (http://www.ibm.com/support/entry/portal/Open_service_request/Software/Software_support_(general)).

Using logging

Logging is the primary troubleshooting feature in the monitoring agent. *Logging* refers to the text messages and trace data that is generated by the agent. Messages and trace data are sent to a file.

Trace data captures transient information about the current operating environment when a component or application fails to operate as designed. IBM Software Support personnel use the captured trace information to determine the source of an error or unexpected condition. See "Trace logging" on page 2 for more information.

Consulting the lists of identified problems and workarounds

Known problems are organized into types such as those in the following list to make them easier to locate:

- Installation, configuration, uninstallation
- Remote deployment
- Agent
- Workspace
- Situation
- Take Action commands
- Discovery Library Adapter
- Tivoli Common Reporting

See "Problems and workarounds" on page 15 for information about symptoms and detailed workarounds for these types of problems.

For general troubleshooting information, see the *IBM Tivoli Monitoring Troubleshooting Guide*.

Trace logging

Trace logs are used to capture information about the operating environment when component software fails to operate as designed.

The principal log type is the RAS (Reliability, Availability, and Serviceability) trace log. These logs are in the English language only. The RAS trace log mechanism is available for all components of IBM Tivoli Monitoring. Most logs are in a logs subdirectory on the host computer. See the following information to learn how to configure and use trace logging:

- "Overview of log file management" on page 3
- "Principal trace log files" on page 3
- "Examples: Using trace logs" on page 9
- "RAS trace parameters" on page 10
- "Dynamic modification of trace settings" on page 12
- "Setting trace parameters for the Tivoli Enterprise Console server" on page 15

Note: The documentation refers to the RAS facility in IBM Tivoli Monitoring as "RAS1."

IBM Software Support personnel use the information captured by trace logging to trace a problem to its source or to determine why an error occurred. All components in the IBM Tivoli Monitoring environment have a default tracing level. The tracing level can be changed on a per-component level to adjust the type of trace information collected, the degree of trace detail, the number of trace logs to be kept, and the amount of disk space used for tracing.

Overview of log file management

Knowing the naming conventions for log files helps you to find the files.

Agent log file naming conventions

Table 2 on page 4 provides the names, locations, and descriptions of IBM Tivoli Monitoring general RAS1 log files. The log file names for the Linux Kernel-based Virtual Machines agent adhere to the following naming convention:

Windows systems

hostname_productcode_instance-name_program_HEXtimestamp-nn.log

Linux® and UNIX systems

hostname_productcode_instance-name_program_HEXtimestamp-nn.log

Where:

hostname
Host name of the computer where the monitoring component is running.

productcode
Two-character product code. For IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines, the product code is v1.

instance-name
Instance name of the agent.

program
Name of the program being run.

HEXtimestamp
Hexadecimal time stamp representing the time at which the program started.

nn
Rolling log suffix.

Principal trace log files

Trace log files are located on various systems.

Table 2 on page 4 contains locations, file names, and descriptions of trace logs that can help determine the source of problems with agents.

Table 2. Trace log files for troubleshooting agents		
System where log is located	**File name and path**	**Description**

On the Tivoli Enterprise Monitoring Server	• **Windows:** The
	`IBM Tivoli Monitoring timestamp.log`
	file in the
	`install_dir\InstallITM`
	path
	• **UNIX:** The
	`candle_installation.log`
	file in the
	`install_dir/logs`
	path
	• **Linux:** The `candle_installation.log` file in the `install_dir/logs` path
		Provides details about products that are installed.
		Note: Trace logging is enabled by default. A configuration step is not required to enable this tracing.

On the Tivoli Enterprise Monitoring Server	The
	`Warehouse_Configuration.log`
	file is in the following location on Windows systems:
	`install_dir\InstallITM`
		Provides details about the configuration of data warehousing for historical reporting.

Table 2. Trace log files for troubleshooting agents (continued)

System where log is located	File name and path	Description

On the Tivoli Enterprise Monitoring Server	The name of the RAS log file is as follows:
	• **Windows:**
	`install_dir\logs\hostname_ms_timestamp-nn.log`
	• **UNIX:**
	`install_dir/logs/hostname_ms_timestamp-nn.log`
	• **Linux:**
	`install_dir/logs/hostname_ms_timestamp-nn.log`
	Note: File names for RAS1 logs include a hexadecimal time stamp.
	Also on UNIX systems, a log with a decimal time stamp is provided:
	`hostname_v1_timestamp.log`
	and
	`hostname_v1_timestamp.pidnnnnn`
	in the *install_dir*/logs path, where *nnnnn* is the process ID number.
		Traces activity on the monitoring server.

Table 2. Trace log files for troubleshooting agents (continued)

System where log is located	File name and path	Description

On the Tivoli Enterprise Portal Server	The name of the RAS log file is as follows:
	• **Windows:**
	`install_dir\logs\hostname_cq_HEXtimestamp-nn.log`
	• **UNIX:**
	`install_dir/logs/hostname_cq_HEXtimestamp-nn.log`
	• **Linux:**
	`install_dir /logs/hostname_cq_HEXtimestamp-nn.log`
	Note: File names for RAS1 logs include a hexadecimal time stamp.
	Also on UNIX systems, a log with a decimal time stamp is provided:
	`hostname_v1_timestamp.log`
	and
	`hostname_v1_timestamp.pidnnnnn`
	in the *install_dir*/logs path, where *nnnnn* is the process ID number.
		Traces activity on the portal server.

On the Tivoli Enterprise Portal Server	The `teps_odbc.log` file is located in the following path:
	• **Windows:**
	`install_dir\InstallITM`
	• **UNIX:**
	`install_dir/logs`
	• **Linux:**
	`install_dir/logs`
		When you enable historical reporting, this log file traces the status of the warehouse proxy agent.

Table 2. Trace log files for troubleshooting agents (continued)

System where log is located	File name and path	Description
On the computer that hosts the monitoring agent	The RAS1 log files are as follows: • **UNIX:** *hostname*_v1_*instance_name*_kv1agent_HEXtimestamp-nn.log in the *install_dir*/logs directory • **Linux:** *hostname*_v1_*instance_name*_kv1agent_HEXtimestamp-nn.log in the *install_dir*/logs directory These logs are in the following directories: • **UNIX:** *install_dir*/logs • **Linux:** *install_dir*/logs On Linux systems, the following additional logs are provided: – *hostname*_v1_timestamp.log – *hostname*_v1_timestamp.pidnnnnn in the *install_dir*/logs path, where *nnnnn* is the process ID number	Traces activity of the monitoring agent.

Table 2. Trace log files for troubleshooting agents (continued)		
System where log is located	**File name and path**	**Description**

On the computer that hosts the monitoring agent	The agent operations log files are as follows:
	`instance_hostname_V1.LG0`
	is the current log created when the agent is started.
	`instance_hostname_V1.LG1`
	is the backup of the previous log.
	These logs are in the following directory depending on the operating system that you are using:
	Linux:
	`install_dir/logs`
	UNIX:
	`install_dir/logs`
		Shows whether the agent could connect to the monitoring server. Shows which situations are started and stopped, and shows other events while the agent is running. A new version of this file is generated every time the agent is restarted.
		IBM Tivoli Monitoring generates one backup copy of the `*.LG0` file with the tag `.LG1`. View the `.LG1` tag to learn the following details regarding the *previous* monitoring session:
		• Status of connectivity with the monitoring server
		• Situations that were running
		• The success or failure status of Take Action commands

On the computer that hosts the monitoring agent	The Take Action command log files are as follows: • *host*_v1_*instance*_*takeactioncommand*.log The logs are in the following directories: • **UNIX:** *install_dir*/logs • **Linux:** *install_dir*/logs	Traces activity each time a Take Action command runs. For example, when a hypothetical **start_command** Take Action command runs, IBM Tivoli Monitoring generates a `start_command.log` file.
On the computer that hosts the monitoring agent	The Take Action command log files are as follows: • kv1_data_provider_actions_*instance_n*.log The logs are in the following directories: • **UNIX**: *install_dir*/logs • **Linux**: *install_dir*/logs	Traces activity each time a Take Action command runs. All predefined Take Action commands are logged into this file.

Table 2. Trace log files for troubleshooting agents (continued)		
System where log is located	**File name and path**	**Description**
Definitions of variables: • *timestamp* is a time stamp with a format that includes year (y), month (m), day (d), hour (h), and minute (m), as follows: **yyyymmdd hhmm** • *HEXtimestamp* is a hexadecimal representation of the time at which the process was started. • *install_dir* represents the directory path where you installed the IBM Tivoli Monitoring component. *install_dir* can represent a path on the computer that hosts the monitoring system, the monitoring agent, or the portal. • *instance* refers to the name of the database instance that you are monitoring. • *instance_name* refers to the name of the agent instance. • *hostname* refers to the name of the computer on which the IBM Tivoli Monitoringcomponent runs. • *nn* represents the circular sequence in which logs are rotated. this value includes a range from 1 - 5, by default. The first is always retained because it includes configuration parameters. • *productcode* specifies the product code, for example, um for Universal Agent or nt for Windows systems.		

For more information about the complete set of trace logs that are maintained on the monitoring server, see the *IBM Tivoli Monitoring Installation and Setup Guide*.

Examples: Using trace logs

You can open trace logs in a text editor to learn some basic facts about your IBM Tivoli Monitoring environment.

IBM Software Support applies specialized knowledge to analyze trace logs to determine the source of problems. The following examples are from the Tivoli Enterprise Monitoring Server log.

Example one

This excerpt shows the typical log for a failed connection between a monitoring agent and a monitoring server with the host name **server1a**:

```
(Thursday, August 11, 2005, 08:21:30-{94C}kdcl0cl.c,105,"KDCL0_ClientLookup")
status=1c020006,
 "location server unavailable", ncs/KDC1_STC_SERVER_UNAVAILABLE
(Thursday, August 11, 2005, 08:21:35-{94C}kraarreg.cpp,1157,"LookupProxy") Unable to
connect to
 broker at ip.pipe:: status=0, "success", ncs/KDC1_STC_OK
(Thursday, August 11, 2005, 08:21:35-{94C}kraarreg.cpp,1402,"FindProxyUsingLocalLookup")
Unable
 to find running CMS on CT_CMSLIST <IP.PIPE:#server1a>
```

Example two

The following excerpts from the trace log *for the monitoring server* show the status of an agent, identified here as "Remote node." The name of the computer where the agent is running is **SERVER5B**:

```
(42C039F9.0000-6A4:kpxreqhb.cpp,649,"HeartbeatInserter") Remote node SERVER5B:V1 is ON-LINE.
. . .
(42C3079B.0000-6A4:kpxreqhb.cpp,644,"HeartbeatInserter") Remote node SERVER5B:V1 is OFF-
LINE.
```

See the following key points about the preceding excerpts:

- The monitoring server appends the V1 product code to the server name to form a unique name (SERVER5B:V1) for this instance of the IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines. By using this unique name, you can distinguish multiple monitoring products that might be running on SERVER5B.

- The log shows when the agent started (ON-LINE) and later stopped (OFF-LINE) in the environment.
- For the sake of brevity, an ellipsis (...) represents the series of trace log entries that were generated while the agent was running.
- Between the ON-LINE and OFF-LINE log entries, the agent was communicating with the monitoring server.
- The ON-LINE and OFF-LINE log entries are always available in the trace log. All trace levels that are described in "Setting RAS trace parameters by using the GUI" on page 10 provide these entries.

On Windows systems, you can use the following alternate method to view trace logs:

1. In the Windows **Start** menu, click **Program Files** > **IBM Tivoli Monitoring** > **Manage Tivoli Enterprise Monitoring Services**. The **Manage Tivoli Enterprise Monitoring Services** window is displayed.
2. Right-click a component and click **Advanced** > **View Trace Log** in the menu. For example, if you want to view the trace log for the agent, right-click the name of that agent in the window. You can also use the viewer to access remote logs.

RAS trace parameters

Pinpoint a problem by setting detailed tracing of individual components of the monitoring agent and modules

See "Overview of log file management" on page 3 to ensure that you understand log rolling and can reference the correct log files when you manage log file generation.

Setting RAS trace parameters by using the GUI

On Windows systems, you can use the graphical user interface to set trace options.

About this task

The IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines uses RAS1 tracing and generates the logs described in Table 2 on page 4. The default RAS1 trace level is ERROR.

Procedure

1. Open the **Manage Tivoli Enterprise Monitoring Services** window.
2. Select **Advanced** > **Edit Trace Parms**. The **Tivoli Enterprise Monitoring Server Trace Parameters** window is displayed.
3. Select a new trace setting in the pull-down menu in the **Enter RAS1 Filters** field or type a valid string.
   - General error tracing. KBB_RAS1=ERROR
   - Intensive error tracing. KBB_RAS1=ERROR (UNIT:kv1 ALL)
   - Maximum error tracing. KBB_RAS1=ERROR (UNIT:kv1 ALL) (UNIT:kra ALL)

   Note: As this example shows, you can set multiple RAS tracing options in a single statement.
4. Modify the value for Maximum Log Size Per File (MB) to change the log file size (changes LIMIT value).
5. Modify the value for Maximum Number of Log Files Per Session to change the number of log files per startup of a program (changes COUNT value).
6. Modify the value for Maximum Number of Log Files Total to change the number of log files for all startups of a program (changes MAXFILES value).
7. Optional: Click Y (Yes) in the **KDC_DEBUG Setting** menu to log information that can help you diagnose communications and connectivity problems between the monitoring agent and the monitoring server. The **KDC_DEBUG** setting and the **Maximum error tracing** setting can generate a large amount of trace logging. Use these settings only temporarily, while you are troubleshooting problems. Otherwise, the logs can occupy excessive amounts of hard disk space.

8. Click **OK**. You see a message reporting a restart of the monitoring agent so that your changes take effect.

What to do next

Monitor the size of the `logs` directory. Default behavior can generate a total of 45 - 60 MB for each agent that is running on a computer. For example, each database instance that you monitor can generate 45 - 60 MB of log data. See the "Procedure" section to learn how to adjust file size and numbers of log files to prevent logging activity from occupying too much disk space.

Regularly prune log files other than the RAS1 log files in the `logs` directory. Unlike the RAS1 log files that are pruned automatically, other log types can grow indefinitely, for example, the logs in Table 2 on page 4 that include a process ID number (PID).

Use collector trace logs as an additional source of troubleshooting information.

Note: The **KDC_DEBUG** setting and the **Maximum error tracing** setting can generate a large amount of trace logging. Use these settings only temporarily while you are troubleshooting problems. Otherwise, the logs can occupy excessive amounts of hard disk space.

Manually setting RAS trace parameters

You can manually edit the RAS1 trace logging parameters.

About this task

Agents use RAS1 tracing and generate the logs described in Table 2 on page 4. The default RAS1 trace level is ERROR.

Procedure

1. Open the trace options file:
   - **UNIX systems:**

     install_dir /config/v1_*instance name*.config
2. Edit the line that begins with **KBB_RAS1=** to set trace logging preferences. For example, if you want detailed trace logging, set the **Maximum Tracing** option: `KBB_RAS1=ERROR (UNIT:kv1 ALL) (UNIT:kra ALL)`
3. Edit the line that begins with **KBB_RAS1_LOG=** to manage the generation of log files:
   - **MAXFILES**: The total number of files that are to be kept for all startups of a specific program. When this value is exceeded, the oldest log files are discarded. The default value is 9.
   - **LIMIT**: The maximum size, in megabytes (MB) of a RAS1 log file. The default value is 5.
   - IBM Software Support might guide you to modify the following parameters:
     - **COUNT**: The number of log files to keep in the rolling cycle of one program startup. The default is 3.
     - **PRESERVE**: The number of files that are not to be reused in the rolling cycle of one program startup. The default value is 1.

     Note: The **KBB_RAS1_LOG** parameter also provides for the specification of the log file directory, log file name, and the inventory control file directory and name. Do not modify these values or log information can be lost.
4. Restart the monitoring agent so that your changes take effect.

What to do next

Monitor the size of the `logs` directory. Default behavior can generate a total of 45 - 60 MB for each agent that is running on a computer. For example, each database instance that you monitor can generate 45 - 60 MB of log data. See the "Procedure" section to learn how to adjust file size and numbers of log files to prevent logging activity from occupying too much disk space.

Regularly prune log files other than the RAS1 log files in the `logs` directory. Unlike the RAS1 log files that are pruned automatically, other log types can grow indefinitely, for example, the logs in Table 2 on page 4 that include a process ID number (PID).

Use collector trace logs as an additional source of troubleshooting information.

Note: The **KDC_DEBUG** setting and the **Maximum error tracing** setting can generate a large amount of trace logging. Use these settings only temporarily while you are troubleshooting problems. Otherwise, the logs can occupy excessive amounts of hard disk space.

Dynamic modification of trace settings

You can dynamically modify the trace settings for an IBM Tivoli Monitoring component, such as, Tivoli Enterprise Monitoring Server, Tivoli Enterprise Portal Server, most monitoring agents, and other components. You can access these components, except for a few monitoring agents, from the tracing utility.

Dynamic modification of the trace settings is the most efficient method, because you can do it without restarting the component. Settings take effect immediately. Modifications by this method are not persistent.

Note: When the component is restarted, the trace settings are read again from the `.env` file. Dynamically modifying these settings does not change the settings in the `.env` files. To modify these trace settings permanently, modify them in the `.env` files.

ras1

Run this command to modify the trace settings for a Tivoli Monitoring component.

The syntax is as follows:

```
ras1 set|list (UNIT|COMP: class_name ANY|ALL|Detail|ERROR|Flow|INPUT|Metrics|OUTPUT|STATE)
{(UNIT|COMP: class_name ANY|ALL|Detail|ERROR|Flow|INPUT|Metrics|OUTPUT|STATE)}
```

You can specify more than one component class to which to apply the trace settings.

Command options

`set`
Turns on or off tracing depending upon the value of its parameters. If the parameter is **ANY**, it turns it off. All other parameters turn on tracing based on the specified type or level.

`list`
Displays the default level and type of tracing that is set by default.

Parameters

The parameters that determine the component classes to which to apply the trace settings are as follows:

`COMP:` *class_name*
Modifies the trace setting for the name of the component class, as specified by *class_name* , for example, `COMP:KDH`. The output contains trace for the specified class.

`UNIT:` *class_name*
Modifies the trace setting for any unit that starts with the specified *class_name* value, for example, `UNIT: kra`. The output contains trace for any unit that begins with the specified filter pattern.

The parameters that determine the trace level and type are as follows:

`ALL`
Displays all trace levels, including every trace point defined for the component. This setting might result in a large amount of trace, so specify other parameters to exclude unwanted trace. You might require the **ALL** parameter to isolate a problem, which is the equivalent to setting "`Error Detail Flow State Input Output Metrics`".

ANY
Turns off tracing.

Detail
Displays detailed information about each function.

When entered with the `list` option, the trace is tagged with `Det`.

ERROR
Logs internal error conditions.

When entered with the `list` option, the trace is tagged with ER. The output can also be tagged with EVERYE+EVERYU+ER.

Flow
Displays control flow data for each function entry and exit.

When entered with the `list` option, the trace is tagged with `Fl`.

INPUT
Displays input data for each function.

When entered with the `list` option, the trace is tagged with `IN`.

Metrics
Displays metrics on each function.

When entered with the `list` option, the trace is tagged with `ME`.

OUTPUT
Displays output data for each function.

When entered with the `list` option, the trace is tagged with `OUT`.

State
Displays the status for each function.

When entered with the `list` option, the trace is tagged with `St`.

Example

If you enter `ras1 set (COMP:KDH ALL) (COMP:ACF1 ALL) (COMP:KDE ALL)`, the trace utility turns on all levels of tracing for all the files and functions for which KDH, ACF1, and KDE are the classes.

```
kbbcre1.c, 400, May 29 2007, 12:54:43, 1.1, *
kbbcrn1.c, 400, May 29 2007, 12:54:42, 1.1, *
kdhb1de.c, 400, May 29 2007, 12:59:34, 1.1, KDH
kdh0med.c, 400, May 29 2007, 12:59:24, 1.1, KDH
kdhsrej.c, 400, May 29 2007, 13:00:06, 1.5, KDH
kdhb1fh.c, 400, May 29 2007, 12:59:33, 1.1, KDH
kdhb1oe.c, 400, May 29 2007, 12:59:38, 1.2, KDH
kdhs1ns.c, 400, May 29 2007, 13:00:08, 1.3, KDH
kbbacdl.c, 400, May 29 2007, 12:54:27, 1.2, ACF1
kbbaclc.c, 400, May 29 2007, 12:54:27, 1.4, ACF1
kbbac1i.c, 400, May 29 2007, 12:54:28, 1.11, ACF1
vkdhsfcn.c, 400, May 29 2007, 13:00:11, 1.1, KDH
kdhserq.c, 400, May 29 2007, 12:59:53, 1.1, KDH
kdhb1pr.c, 400, May 29 2007, 12:59:39, 1.1, KDH
kdhsgnh.c, 400, May 29 2007, 12:59:49, 1.1, KDH
kdh0uts.c, 400, May 29 2007, 12:59:23, 1.1, KDH
kdhsrsp.c, 400, May 29 2007, 13:00:13, 1.2, KDH
kdhs1rp.c, 400, May 29 2007, 13:00:12, 1.1, KDH
kdhscsv.c, 400, May 29 2007, 12:59:58, 1.9, KDH
kdebbac.c, 400, May 29 2007, 12:56:50, 1.10, KDE
...
```

Turning on tracing

To use the tracing utility, you must use a local logon credential for the computer. This tracing method uses the IBM Tivoli Monitoring Service Console. Access the Service Console by using a web browser.

About this task

When you start the Service Console, information is displayed about the components that are currently running on that computer. For example, these components are listed as follows:

- Tivoli Enterprise Portal Server: `cnp`
- Monitoring Agent for Windows OS: `nt`
- Tivoli Enterprise Monitoring Server: `ms`

After you log on, you can type a question mark (?) to display a list of the supported commands. Use the **`ras1`** command to modify trace settings. If you type this command in the field provided in the Service Console window and click **Submit**, the help for this command is displayed.

Procedure

1. Open a web browser and enter the URL to access the Service Console.

   ```
   http://hostname:1920
   ```

   where *hostname* is the IP address or host name of the computer on which the IBM Tivoli Monitoring component is running.
2. Click the hyperlink associated with the component for which you want to modify its trace settings.

   Note: In the previous view, if you want to modify tracing for the Tivoli Enterprise Monitoring Server, select **IBM Tivoli Monitoring Service Console** under **Service Point: system.*your host name*_ms**.
3. Enter a user ID and password to access the system. This ID is any valid user that has access to the system.
4. Enter the command to turn on the required level of trace for the specified component classes or units.

   ```
   ras1 set (UNIT|COMP: class_name ALL|Flow|ERROR|Detail|INPUT|Metrics|OUTPUT|STATE)
   {(UNIT|COMP: class_name ALL|Flow|ERROR|Detail|INPUT|Metrics|OUTPUT|STATE)}
   ```

   For example, to turn on the control flow trace for the KDE, the command is:

   ```
   ras1 (COMP:KDE Flow)
   ```

Turning off tracing

You can use the IBM Tivoli Monitoring Service Console to run the **`ras1`** command and dynamically turn off tracing.

Procedure

1. Open a web browser and enter the URL to access the Service Console.

   ```
   http://hostname:1920
   ```

   where *hostname* is the IP address or host name of the computer on which the IBM Tivoli Monitoring component is running.
2. Click the hyperlink associated with the component for which you want to modify its trace settings.
3. Enter a user ID and password to access the system. This ID is any valid user that has access to the system.
4. Enter the command to turn off the required level of trace for the specified component classes or units.

```
ras1 set (UNIT|COMP: class_name ANY)
{(UNIT|COMP: class_name ANY)}
```

For example, to turn off tracing for the kbbcrcd class of the Windows OS agent, the command is:

```
ras1 set (UNIT:kbbcrcd ANY)
```

Setting trace parameters for the Tivoli Enterprise Console server

In addition to the trace information captured by IBM Tivoli Monitoring, you can also collect additional trace information for the Tivoli Enterprise Console components that gather event server metrics.

About this task

To collect this information, modify the `.tec_diag_config` file on the Tivoli Enterprise Console event server. Use the steps in the following procedure to modify the event server trace parameters.

Procedure

1. Open the `$BINDIR/TME/TEC/.tec_diag_config` file in an ASCII editor.
2. Locate the entries that configure trace logging for the agent components on the event server. Two entries are included, one for `tec_reception` and one for `tec_rule`:

```
# to debug Agent Utils
tec_reception Agent_Utils  error   /tmp/tec_reception
SP
# to debug Agent Utils
tec_rule Agent_Utils   error   /tmp/tec_rule
```

3. To gather additional trace information, modify these entries to specify a trace level of `trace2`:

```
# to debug Agent Utils
tec_reception Agent_Utils   trace2  /tmp/tec_reception
SP
# to debug Agent Utils
tec_rule Agent_Utils       trace2  /tmp/tec_rule
```

4. In addition, modify the `Highest_level` entries for `tec_rule` and `tec_reception`:

```
tec_reception Highest_level  trace2
SP
tec_rule Highest_level trace2
```

Problems and workarounds

The known problems and workarounds are organized into types of problems that might occur with an agent, for example installation and configuration problems and workspace problems.

You can resolve some problems by ensuring that your system matches system requirements. The most up-to-date requirements are in the Software product compatibility reports (http://publib.boulder.ibm.com/infocenter/prodguid/v1r0/clarity/index.html).

For general troubleshooting information, see the *IBM Tivoli Monitoring Troubleshooting Guide*.

Installation and configuration troubleshooting

Problems can occur during installation, configuration, and uninstallation of the agent.

See Table 3 on page 16 and Table 4 on page 17 for information about these problems and solutions.

Table 3. Problems and solutions for installation and configuration

Problem	Solution
(*UNIX only*) During a command-line installation, you choose to install a component that is currently installed, and you see the following warning: `WARNING - you are about to install the SAME version of "component_name"` where *component_name* is the name of the component that you are attempting to install. **Note:** This problem affects UNIX command-line installations. If you monitor only Windows environments, you see this problem if you choose to install a product component (for example, a monitoring server) on a UNIX system.	You must exit and restart the installation process. You cannot return to the list where you selected components to install. When you run the installer again, do not attempt to install any component that is currently installed.
A message similar to "`Unable to find running CMS on CT_CMSLIST`" in the log file is displayed.	If a message similar to "`Unable to find running CMS on CT_CMSLIST`" is displayed in the log file, the agent cannot connect to the monitoring server. Confirm the following points: • Do multiple network interface cards (NICs) exist on the system? • If multiple NICs exist on the system, find out which one is configured for the monitoring server. Ensure that you specify the correct host name and port settings for communication in the IBM Tivoli Monitoring environment.
The system is experiencing high CPU usage.	**Agent process:** View the memory usage of the KV1CMA process. If CPU usage seems to be excessive, restart the monitoring agent. **Network cards:** The network card configurations can decrease the performance of a system. Each stream of packets that a network card receives (assuming that it is a broadcast or destined for the under-performing system) must generate a CPU interrupt and transfer the data through the I/O bus. If the network card in question is a bus-mastering card, work can be offloaded and a data transfer between memory and the network card can continue without using CPU processing power. Bus-mastering cards are 32-bit and are based on PCI or EISA bus architectures.

Table 4. General problems and solutions for uninstallation	
Problem	**Solution**
The way to remove inactive managed systems (systems whose status is OFFLINE) from the Navigator tree in the portal is not obvious.	Use the following steps to remove, but not uninstall, an offline managed system from the Navigator tree: 1. Click the **Enterprise** icon in the Navigator tree. 2. Right-click, and then click **Workspace** > **Managed System Status**. 3. Right-click the offline managed system, and select **Clear offline entry**. To uninstall the monitoring agent, use the procedure described in the *IBM Tivoli Monitoring Installation and Setup Guide*.

Table 4. General problems and solutions for uninstallation (continued)

Problem	Solution
IBM Tivoli Monitoring might not be able to generate a unique name for monitoring components because of the truncation of names that the product automatically generates.	If the agent supports multiple instances, IBM Tivoli Monitoring automatically creates a name for each monitoring component by concatenating the subsystem name, host name, and product code separated by colons (*subsystem_name*:*hostname*:KV1). **Note:** When you monitor a multinode system, such as a database, IBM Tivoli Monitoring adds a subsystem name to the concatenated name, typically a database instance name. The length of the name that IBM Tivoli Monitoring generates is limited to 32 characters. Truncation can result in multiple components having the same 32-character name. If this problem happens, shorten the *hostname* portion of the name as follows: 1. Open the configuration file for the monitoring agent, which is located in the following path: • **On UNIX and Linux:** `itm_home/config/`*`product_code`*`.ini` and *`product_code`*`.config`. For example, the file names for the Monitoring Agent for UNIX OS is `ux.ini` and `ux.config`. 2. Find the line that begins with `CTIRA_HOSTNAME=`. 3. Type a new name for host name that is a unique, shorter name for the host computer. The final concatenated name including the subsystem name, new host name, and KV1, cannot be longer than 32 characters. **Note:** You must ensure that the resulting name is unique with respect to any existing monitoring component that was previously registered with the Tivoli Enterprise Monitoring Server. 4. Save the file. 5. Restart the agent.

Table 4. General problems and solutions for uninstallation (continued)

Problem	Solution
Installation on RHEL Linux 64-bit systems uses **install.sh**. Running this script fails with a `runGSkit failure: Return error code: 99.`	GSkit is called by **install.sh** and fails when runGSkit calls `verifyInstall`. Review the *InstallDirectory*`/logs/candle_installation.log` file and look for references to `runGSkit`. For example, output similar to the following might be present: `runGSkit: ----- Running command: /opt/IBM/ITM/li6243/gs/bin/private_verifyinstall /opt/IBM/ITM/li6243/gs/bin/gsk7ver: error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory Error: Verify Failed Expected Details of gskit in /opt/IBM/ITM/li6243/gs runGSkit: return code from command is 99 runGSkit: ----- End of running command ----- runGSkit: error Return error code: 99 runGSkit: error GSKit check failure, script: /opt/IBM/ITM/li6243/gs/bin/private_verifyinstall runGSkit: error li6243 - GSK check error, verifyInstall test failed`  In the previous example, the 32-bit version of the `libstdc++.so.5` file is not present. This file comes from the `compat-libstdc++-33-3.2.3-XX.i686.rpm` package, which is not installed on 64-bit RHEL systems by default. When this package is installed, the problem no longer occurs.

After installation, the Linux Kernel-based Virtual Machines agent instance fails to start. The following message appears in the agent log:
`(4CF55620.003F-1:kbbssge.c,52,"BSS1_GetEnv")`
`KBB_SIG1="-asyncoff -syncoff -dumpoff"`
`(4CF55620.0040-1:signalmanager.cpp,170, "startManagerThread")`
`Error starting signal manager thread. Return code = 11;`
`Resource temporarily unavailable.`
`Use the return code and message to investigate the failure. Agent is terminating.`
	The probable cause of the problem is the public domain Korn shell, **pdksh**. Uninstall the **pdksh** shell and install the **ksh** rpm that is included on the Linux installation media.

The software inventory tag for the agent on UNIX and Linux systems is not removed during uninstallation of the agent.	After uninstalling the agent, manually remove the file named *`full name of agent`*`.cmptag` from the `$CANDLEHOME/properties/version/` directory.

Remote deployment troubleshooting

Problems can occur with remote deployment and removal of agent software using the Agent Remote Deploy process.

Table 5 on page 20 contains problems and solutions related to remote deployment.

Table 5. Remote deployment problems and solutions

Problem	Solution
While you are using the remote deployment feature to install the IBM Tivoli Monitoring for Virtual Environments Agent for Linux Kernel-based Virtual Machines, an empty command window is displayed on the target computer. This problem occurs when the target of remote deployment is a Windows computer. (For more information about the remote deployment feature, see the *IBM Tivoli Monitoring Installation and Setup Guide*.)	Do not close or modify this window. It is part of the installation process and is dismissed automatically.
The removal of a monitoring agent fails when you use the remote removal process in the Tivoli Enterprise Portal desktop or browser.	This problem might occur when you attempt the remote removal process immediately after you restart the Tivoli Enterprise Monitoring Server. You must allow time for the monitoring agent to refresh its connection with the Tivoli Enterprise Monitoring Server before you begin the remote removal process.

Agent troubleshooting

A problem can occur with the agent after it has been installed.

Table 6 on page 20 contains problems and solutions that can occur with the agent after it is installed.

Table 6. Agent problems and solutions

Problem	Solution
When using the **itmcmd agent** commands to start or stop this monitoring agent, you receive the following error message: `MKCIIN0201E Specified product is not configured.`	Include the command option **-o** to specify the instance to start or stop. The instance name must match the name used for configuring the agent. For example: `./itmcmd agent -o Test1 start v1` For more information about using the itmcmd commands, see the *IBM Tivoli Monitoring Command Reference*.
In the agent log, you see the message SEVERE: `DataSource.connect: failed to connect to data source ip address`.	Find the hypervisor URI of the host that failed to connect. It is listed in a message just before the `failed to connect` message. On the computer where the agent is installed, enter the command `export LIBVIRT_DEBUG=yes` followed by `virsh -c hypervisor uri that failed`. Review the extra debug messages for symptoms of an underlying problem.

Table 6. Agent problems and solutions (continued)

Problem	**Solution**
After turning on extra libvirt messages by using `export LIBVIRT_DEBUG=yes`, when you execute the command `virsh -c` *`hypervisor uri`*, you see the message: `bash: nc: command not found`.	Install netcat-openbsd on the host of the hypervisor to be monitored.
CPU Model might show truncated string for the CPU make.	Upgrade an agent to the version 7.3 Fix Pack 1 where this attribute is enhanced to show the full name(max 100 Characters) of CPU Model.

Table 6. Agent problems and solutions (continued)	
Problem	**Solution**
A configured and running instance of the monitoring agent is not displayed in the Tivoli Enterprise Portal, but other instances of the monitoring agent on the same system are displayed in the portal.	IBM Tivoli Monitoring products use Remote Procedure Call (RPC) to define and control product behavior. RPC is the mechanism that a client process uses to make a subroutine call (such as GetTimeOfDay or ShutdownServer) to a server process somewhere in the network. Tivoli processes can be configured to use TCP/UDP, TCP/IP, SNA, and SSL as the protocol (or delivery mechanism) for RPCs that you want. IP.PIPE is the name given to Tivoli TCP/IP protocol for RPCs. The RPCs are socket-based operations that use TCP/IP ports to form socket addresses. IP.PIPE implements virtual sockets and multiplexes all virtual socket traffic across a single physical TCP/IP port (visible from the **netstat** command). A Tivoli process derives the physical port for IP.PIPE communications based on the configured, well-known port for the hub Tivoli Enterprise Monitoring Server. (This well-known port or BASE_PORT is configured by using the 'PORT:' keyword on the **KDC_FAMILIES / KDE_TRANSPORT** environment variable and defaults to '1918'.) The physical port allocation method is defined as (BASE_PORT + 4096*N), where N=0 for a Tivoli Enterprise Monitoring Server process and N={1, 2, ..., 15} for another type of monitoring server process. Two architectural limits result as a consequence of the physical port allocation method: • No more than one Tivoli Enterprise Monitoring Server reporting to a specific Tivoli Enterprise Monitoring Server hub can be active on a system image. • No more than 15 IP.PIPE processes can be active on a single system image. A single system image can support any number of Tivoli Enterprise Monitoring Server processes (address spaces) if each Tivoli Enterprise Monitoring Server on that image reports to a different hub. By definition, one Tivoli Enterprise Monitoring Server hub is available per monitoring enterprise, so this architecture limit has been reduced to one Tivoli Enterprise Monitoring Server per system image.

Table 6. Agent problems and solutions (continued)	
Problem	**Solution**
Continued from previous row.	No more than 15 IP.PIPE processes or address spaces can be active on a single system image. With the first limit expressed earlier, this second limitation refers specifically to Tivoli Enterprise Monitoring Agent processes: no more than 15 agents per system image. This limitation can be circumvented (at current maintenance levels, IBM Tivoli Monitoring V6.1, Fix Pack 4 and later) if the Tivoli Enterprise Monitoring Agent process is configured to use the EPHEMERAL IP.PIPE process. (This process is IP.PIPE configured with the 'EPHEMERAL:Y' keyword in the **KDC_FAMILIES / KDE_TRANSPORT** environment variable). The number of ephemeral IP.PIPE connections per system image has no limitation. If ephemeral endpoints are used, the Warehouse Proxy agent is accessible from the Tivoli Enterprise Monitoring Server associated with the agents using ephemeral connections either by running the Warehouse Proxy agent on the same computer or by using the Firewall Gateway feature. (The Firewall Gateway feature relays the Warehouse Proxy agent connection from the Tivoli Enterprise Monitoring Server computer to the Warehouse Proxy agent computer if the Warehouse Proxy agent cannot coexist on the same computer.)
Log data accumulates too rapidly.	Check the RAS trace option settings, which are described in “Setting RAS trace parameters by using the GUI” on page 10. The trace option settings that you can set on the KBB_RAS1= and KDC_DEBUG= lines potentially generate large amounts of data.
Agent connection count to data source is restricted to 6 attempts.	User can change the connection attempts count as per requirement by setting the variable *KV1_DATA_PROVIDER_CONNECTION_RETRY_COUNT=6* with the desired value in the instance environment file. For more information on setting environment variables, refer to **Configuring environment variable** from Monitoring Agent for Linux Kernel-based Virtual Machines Installation and Configuration Guide.

Workspace troubleshooting

Problems can occur with general workspaces and agent-specific workspaces.

Table 7 on page 24 contains problems and solutions related to workspaces.

Table 7. Workspace problems and solutions

Problem	Solution
The process application components are available, but the Availability status shows PROCESS_DATA_NOT_ AVAILABLE.	This problem occurs because the PerfProc performance object is disabled. When this condition exists, IBM Tivoli Monitoring cannot collect performance data for this process. Use the following steps to confirm that this problem exists and to resolve it: 1. In the Windows **Start** menu, click **Run**. 2. Type `perfmon.exe` in the **Open** field of the **Run** window. The **Performance** window is displayed. 3. Click the plus sign (+) in the toolbar. The **Add Counters** window is displayed. 4. Look for **Process** in the **Performance object** menu. 5. Complete one of the following actions: • If you see **Process** in the menu, the PerfProc performance object is enabled and the problem is coming from a different source. You might need to contact IBM Software Support. • If you do not see **Process** in the menu, use the Microsoft utility from the Microsoft.com Operations website to enable the PerfProc performance object. The **Process** performance object becomes visible in the **Performance object** menu of the **Add Counters** windows, and IBM Tivoli Monitoring is able to detect Availability data. 6. Restart the monitoring agent.
The name of the attribute does not display in a bar chart or graph view.	When a chart or graph view that includes the attribute is scaled to a small size, a blank space is displayed instead of a truncated name. To see the name of the attribute, expand the view of the chart until sufficient space is available to display all characters of the attribute name.
At the end of each view, you see the following Historical workspace KFWITM220E error: `Request failed during execution`.	Ensure that you configure all groups that supply data to the view. In the Historical Configuration view, ensure that data collection is started for all groups that supply data to the view.

Table 7. Workspace problems and solutions (continued)

Problem	Solution
You start collection of historical data but the data cannot be seen.	Use the following managing options for historical data collection: • Basic historical data collection populates the Warehouse with raw data. This type of data collection is turned off by default. For information about managing this feature including how to set the interval at which data is collected, see "Managing historical data" in the *IBM Tivoli Monitoring Administrator's Guide*. By setting a more frequent interval for data collection, you reduce the load on the system incurred every time data is uploaded. • Use the Summarization and Pruning agent to collect specific amounts and types of historical data. Historical data is not displayed until the Summarization and Pruning monitoring agent begins collecting the data. By default, this agent begins collection at 2 a.m. daily. At that point, data is visible in the workspace view. For information about how to modify the default collection settings, see "Managing historical data" in the *IBM Tivoli Monitoring Administrator's Guide*.
Historical data collection is unavailable because of incorrect queries in the Tivoli Enterprise Portal.	The Sort By, Group By, and First/Last functions column are not compatible with the historical data collection feature. Use of these advanced functions makes a query ineligible for historical data collection. Even if data collection has started, you cannot use the time span feature if the query for the chart or table includes column functions or advanced query options (Sort By, Group By, First / Last). To ensure support of historical data collection, do not use the Sort By, Group By, or First/Last functions in your queries. For information about the historical data collection function, See "Managing historical data" in the *IBM Tivoli Monitoring Administrator's Guide* or the Tivoli Enterprise Portal online help .
When you use a long process name in the situation, the process name is truncated.	Truncation of process or service names for situations in the Availability table in the portal display is the expected behavior. The maximum name length is 100 bytes.
Regular (non-historical) monitoring data fails to be displayed.	Check the formation of the queries you use to gather data. For example, look for invalid SQL statements.

Table 7. Workspace problems and solutions (continued)	
Problem	**Solution**
Navigator items and workspace titles are labeled with internal names such as `Kxx:KXX0000` instead of the correct names (such as Disk), where XX and xx represent the two-character agent code.	Ensure that application support has been added on the monitoring server, portal server, and portal client. For more information about installing application support, see "Installing and enabling application support" in the *IBM Tivoli Monitoring Installation and Setup Guide*.

Situation troubleshooting

Problems can occur with situations and situation configuration.

Table 8 on page 26 contains problems and solutions for situations.

Table 8. Situation problems and solutions	
Problem	**Solution**
Monitoring activity requires too much disk space.	Check the RAS trace logging settings that are described in “Setting RAS trace parameters by using the GUI” on page 10. For example, trace logs grow rapidly when you apply the `ALL` logging option.
Monitoring activity requires too many system resources.	See the information about disk capacity planning for historical data in the Reference guide for the agent for a description of the performance impact of specific attribute groups. If possible, decrease your use of the attribute groups that require greater system resources.
A formula that uses mathematical operators appears to be incorrect. For example, if you were monitoring a Linux system, the formula that calculates when **Free Memory** falls under 10 percent of **Total Memory** does not work: `LT #'Linux_VM_Stats.Total_Memory' / 10`	This formula is incorrect because situation predicates support only logical operators. Your formulas cannot have mathematical operators. **Note:** The Situation Editor provides alternatives to math operators. In the example, you can select the **% Memory Free** attribute and avoid the need for math operators.
You want to change the appearance of situations when they are displayed in the navigation tree.	1. Right-click an item in the navigation tree. 2. Click **Situations** in the menu. The **Situation Editor** window is displayed. 3. Select the situation that you want to modify. 4. Use the **State** menu to set the status and appearance of the Situation when it triggers. **Note:** The **State** setting is not related to severity settings in the Tivoli Enterprise Console.

Table 8. Situation problems and solutions (continued)	
Problem	**Solution**
When a situation is triggered in the Event Log attribute group, it remains in the Situation Event Console as long as the event ID entry is present in the Event Log workspace. When this event ID entry is removed from the Event Log workspace on the Tivoli Enterprise Portal, the situation is also cleared even if the actual problem that caused the event is not resolved, and the event ID entry is also present in the Windows Event Viewer.	A timeout occurs on the cache of events for the NT Event Log group. Increase the cache time of Event Log collection to meet your requirements by adding the following variable and timeout value to the K*pc*ENV file for the agent (where *pc* is the two-letter product code): `CDP_NT_EVENT_LOG_CACHE_TIMEOUT=3600` This variable determines how long events from the NT Event Log are kept.
The situation for a specific agent is not visible in the Tivoli Enterprise Portal.	Open the Situation Editor. Access the All managed servers view. If the situation is not displayed, confirm that the monitoring server has been seeded for the agent. If not, seed the server, as described in the *IBM Tivoli Monitoring Installation and Setup Guide*.
The monitoring interval is too long.	Access the Situation Editor view for the situation that you want to modify. Check the **Sampling interval** area in the **Formula** tab. Adjust the time interval as required.
The situation did not activate at startup.	Manually recycle the situation as follows: 1. Right-click the situation and select **Stop Situation**. 2. Right-click the situation and select **Start Situation**. **Note:** You can permanently avoid this problem by selecting the **Run at Startup** check box of the Situation Editor view for a specific situation.
The situation is not displayed.	Click the **Action** tab and check whether the situation has an automated corrective action. This action can occur directly or through a policy. The situation might be resolving so quickly that you do not see the event or the update in the graphical user interface.
An Alert event did not occur even though the predicate was correctly specified.	Check the logs, reports, and workspaces.
A situation fires on an unexpected managed object.	Confirm that you distributed and started the situation on the correct managed system.
The product did not distribute the situation to a managed system.	Click the **Distribution** tab and check the distribution settings for the situation.

Table 8. Situation problems and solutions (continued)

Problem	Solution

The situation does not fire.	This problem can be caused when incorrect predicates are present in the formula that defines the situation. For example, the managed object shows a state that normally triggers a monitoring event, but the situation is not true because the wrong attribute is specified in the formula.
	
	In the **Formula** tab, analyze predicates as follows:
	
	1. Click the **fx** icon in the **Formula** area. The **Show formula** window is displayed.
	   a. Confirm the following details in the **Formula** area of the window:
	      • The attributes that you intend to monitor are specified in the formula.
	      • The situations that you intend to monitor are specified in the formula.
	      • The logical operators in the formula match your monitoring goal.
	      • The numeric values in the formula match your monitoring goal.
	   b. (Optional) Select the **Show detailed formula** check box to see the original names of attributes in the application or operating system that you are monitoring.
	   c. Click **OK** to dismiss the **Show formula** window.
	2. (Optional) In the **Formula** area of the **Formula** tab, temporarily assign numeric values that immediately trigger a monitoring event. The triggering of the event confirms that other predicates in the formula are valid.
	
	Note: After you complete this test, you must restore the numeric values to valid levels so that you do not generate excessive monitoring data based on your temporary settings.
	
	For additional information about situations that do not fire, see "Situations are not firing" in the *IBM Tivoli Monitoring Troubleshooting Guide*.

Situation events are not displayed in the Events Console view of the workspace.	Associate the situation with a Navigator item. **Note:** The situation does not need to be displayed in the workspace. It is sufficient that the situation is associated with any Navigator item.

Table 8. Situation problems and solutions (continued)	
Problem	**Solution**
You do not have access to a situation.	**Note:** You must have administrator privileges to complete these steps. 1. Click **Edit** > **Administer Users** to access the **Administer Users** window. 2. In the **Users** area, select the user whose privileges you want to modify. 3. In the **Permissions** tab, **Applications** tab, and **Navigator Views** tab, select the permissions or privileges that correspond to the user role. 4. Click **OK**.
A managed system seems to be offline.	1. Select **Physical View** and click the Enterprise Level of the navigator tree. 2. Click **View** > **Workspace** > **Managed System Status** to see a list of managed systems and their status. 3. If a system is offline, check network connectivity and the status of the specific system or application.

Take Action commands troubleshooting

Problems can occur with Take Action commands.

Table 9 on page 29 contains problems and solutions that can occur with Take Action commands.

When each Take Action command runs, it generates a log file listed in Table 2 on page 4.

Table 9. Take Action commands problems and solutions	
Problem	**Solution**
Take Action commands often require several minutes to complete.	Allow several minutes. If you do not see a message advising you of completion, try to run the command manually.
Situations fail to trigger Take Action commands.	Attempt to manually run the Take Action command in the Tivoli Enterprise Portal. If the Take Action command works, look for configuration problems in the situation. See "Situation troubleshooting" on page 26. If the Take Action command fails, for general information about troubleshooting Take Action commands, see the *IBM Tivoli Monitoring Troubleshooting Guide*.

Support information

If you have a problem with your IBM software, you want to resolve it quickly.

IBM provides the following ways for you to obtain the support you need:

Online

The following websites contain troubleshooting information:

- Go to the IBM Software Support website (http://www.ibm.com/support/entry/portal/software) and follow the instructions.
- Go to the Application Performance Management Wiki (http://www.ibm.com/developerworks/servicemanagement/apm/index.html). Feel free to contribute to this wiki.

IBM Support Assistant
The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, go to the IBM Support Assistant website (http://www.ibm.com/software/support/isa).

Informational, warning, and error messages overview

Messages relay information about how the system or application is performing and can alert you to exceptional conditions when they occur.

Messages are sent to an output destination, such as a file, database, or console screen.

If you receive a warning or error message, you can do one of the following actions:

- Follow the instructions listed in the Detail window of the message if this information is included there.
- Consult the message details listed in this topic to see what action you can take to correct the problem.
- Consult the message log for message ID, text, time, and date of the message, as well as other data you can use to diagnose the problem.

Message format

The message format contains a message ID and text, an explanation, and an operator response.

Agent messages have the following format:

Message ID and text
Explanation
Operator Response

The message ID has the following format:

```
CCC#####severity
```

where:

CCC
Prefix that indicates the component to which the message applies. The following components are used:

K*XX*
Three-character product code for the agent.

####
Number of the message

severity
Severity of the message. Three levels of severity are used:

I
Informational messages provide feedback about something that happened in the product or system that might be important. These messages can provide guidance when you are requesting a specific action from the product.

W
Warning messages call your attention to an exception condition. The condition might not be an error but can cause problems if not resolved.

E

Error messages indicate that an action cannot be completed because of a user or system error. These messages require user response.

The *Text* of the message provides a general statement regarding the problem or condition that occurred. The *Explanation* provides additional information about the message and the possible cause for the condition. The *Operator Response* provides actions to take in response to the condition, particularly for error messages (messages with the "E" suffix).

Note: Many message texts and explanations contain variables, such as the specific name of a server or application. Those variables are represented in this topic as symbols, such as "&1." Actual messages contain values for these variables.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM‘s application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2009. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2009. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information (www.ibm.com/legal/copytrade.shtml).

Intel, Intel logo, and Intel Xeon, are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's user name for purposes of session management, authentication, and single sign-on configuration. These cookies cannot be disabled.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Index