Monitoring Microsoft Windows Event Logs

eG Innovations Product Documentation

www.eginnovations.com

eG Total Performance Visibility

Table of Contents

CHAPTER 1: INTRODUCTION 3

CHAPTER 2: HOW DOES EG ENTERPRISE MONITOR EVENT LOGS? 4

2.1 Pre-requisites for monitoring the Eventlog server 4

CHAPTER 3: MONITORING EVENT LOGS 6

3.1 The EventLog Layer 6

3.1.1 Application Event Log Test 7

3.1.2 Event Log Test 14

3.1.3 System Event Log Test 21

3.1.4 Security Log Test 29

ABOUT EG INNOVATIONS 34

Table of Figures

Figure 2.1: Opening the Local Security Policy 4
Figure 2.2: Viewing the Properties of Audit logon events 5
Figure 2.3: Checking the boxes for Success and Failure 5
Figure 3.1: The layer model of an EventLog server 6
Figure 3.2: Figure 1.2: Test executing on the EventLog layer 7
Figure 3.3: The detailed diagnosis of the Application warnings measure 13
Figure 3.4: The detailed diagnosis of the Application information count measure 14
Figure 3.5: The detailed diagnosis of the System errors measure 28
Figure 3.6: The detailed diagnosis of the System information messages measure 28
Figure 3.7: The detailed diagnosis of the System warnings measure 29
Figure 3.8: The detailed diagnosis of the Successful audits measure 33

Chapter 1: Introduction

Many applications record errors and events in various proprietary error logs. These proprietary error logs have different formats and display different user interfaces. Moreover, the system administrator cannot merge the data to provide a complete report. Therefore, the administrator needs to check a variety of sources to diagnose problems.

Event logging in Microsoft Windows NT/Windows 2000 provides a standard, centralized way for applications and the operating system to record important software and hardware events. The event-logging service stores events from various sources in a single collection called an event log. The system administrator can use the event log to help determine what conditions caused the error and the context in which it occurred. By periodically viewing the event log, the system administrator may be able to identify problems (such as a failing hard drive) before they cause damage. This where eG Enterprise helps administrators.

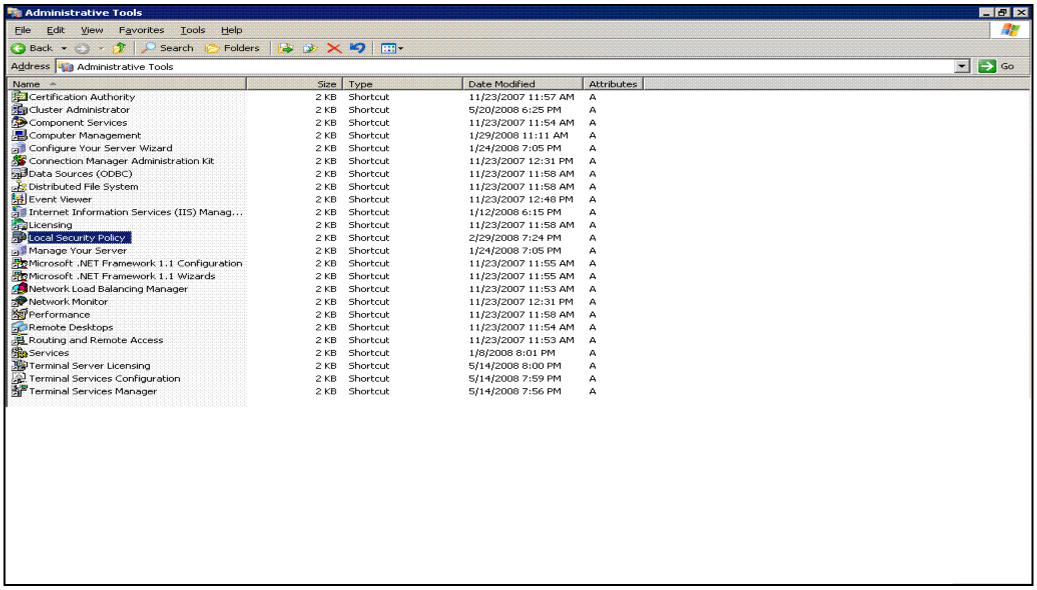
Chapter 2: How does eG Enterprise Monitor Event logs?

eG Enterprise monitors the event logs in an agent based manner. Before starting to monitor the event logs in the server, the following pre-requisites should be fulfilled:

2.1 Pre-requisites for monitoring the Eventlog server

To enable the Security log events do the following steps:

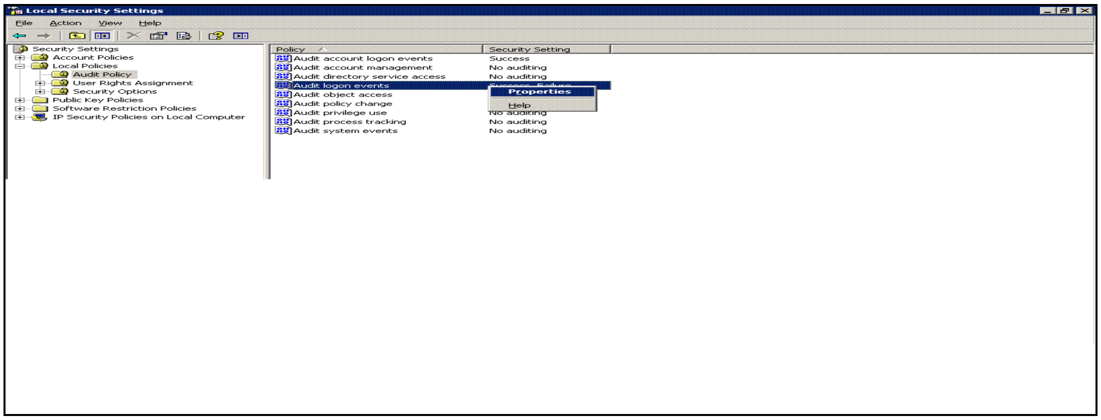
1. Follow the menu sequence Start -> Settings -> Control Panel -> Administrative tools.
2. Now click on the **Local security policy** node in the **Administrative Tools** window.



Name	Size	Type	Date Modified	Attributes
Certification Authority	2 KB	Shortcut	11/23/2007 11:57 AM	A
Cluster Administrator	2 KB	Shortcut	5/20/2008 6:25 PM	A
Component Services	2 KB	Shortcut	11/23/2007 11:54 AM	A
Computer Management	2 KB	Shortcut	1/29/2008 11:11 AM	A
Configure Your Server Wizard	2 KB	Shortcut	1/24/2008 7:05 PM	A
Connection Manager Administration Kit	2 KB	Shortcut	11/23/2007 12:31 PM	A
Data Sources (ODBC)	2 KB	Shortcut	11/23/2007 11:58 AM	A
Distributed File System	2 KB	Shortcut	11/23/2007 11:58 AM	A
Event Viewer	2 KB	Shortcut	11/23/2007 12:48 PM	A
Internet Information Services (IIS) Manag...	2 KB	Shortcut	1/12/2008 6:15 PM	A
Licensing	2 KB	Shortcut	11/23/2007 11:58 AM	A
Local Security Policy	2 KB	Shortcut	2/29/2008 7:24 PM	A
Manage Your Server	2 KB	Shortcut	1/24/2008 7:05 PM	A
Microsoft .NET Framework 1.1 Configuration	2 KB	Shortcut	11/23/2007 11:55 AM	A
Microsoft .NET Framework 1.1 Wizards	2 KB	Shortcut	11/23/2007 11:55 AM	A
Network Load Balancing Manager	2 KB	Shortcut	11/23/2007 11:53 AM	A
Network Monitor	2 KB	Shortcut	11/23/2007 12:31 PM	A
Performance	2 KB	Shortcut	11/23/2007 11:58 AM	A
Remote Desktops	2 KB	Shortcut	11/23/2007 11:54 AM	A
Routing and Remote Access	2 KB	Shortcut	11/23/2007 11:53 AM	A
Services	2 KB	Shortcut	1/8/2008 8:01 PM	A
Terminal Server Licensing	2 KB	Shortcut	5/14/2008 8:00 PM	A
Terminal Services Configuration	2 KB	Shortcut	5/14/2008 7:59 PM	A
Terminal Services Manager	2 KB	Shortcut	5/14/2008 7:56 PM	A

Figure 2.1: Opening the Local Security Policy

3. When the **Local Security Settings** window opens, expand the **Local Policies** node in the tree-structure in the left panel of the window, and click on the **Audit Policy** sub-node.
4. From the list of audit policies displayed in the right panel, select **Audit logon events** and right-click on it to choose **Properties**.

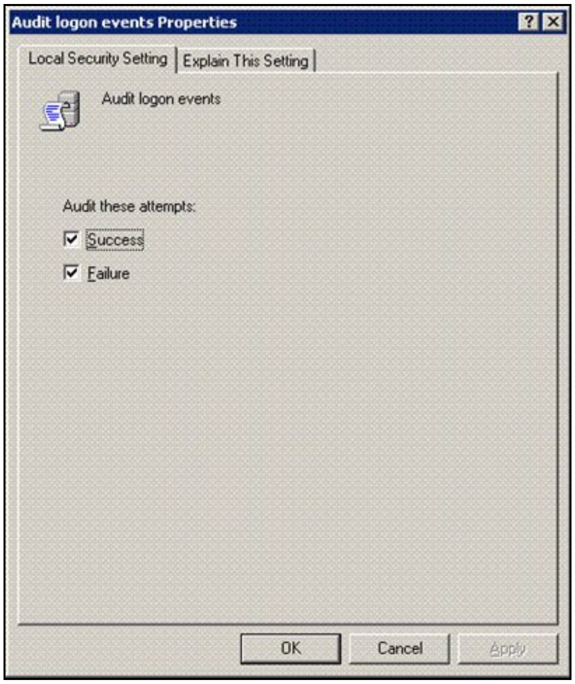


Policy	Security Setting
Audit account logon events	Success
Audit account management	No auditing
Audit directory service access	No auditing
Audit logon events	Success, Failure
Audit object access	
Audit policy change	
Audit privilege use	No auditing
Audit process tracking	No auditing
Audit system events	No auditing

Figure 2.2: Viewing the Properties of Audit logon events

5. You will see two check boxes for **Success** and **Failure** in the **Audit logon events Properties** window.

6. Select both the check boxes and click on **Apply** and then click on **OK** to register the changes.



Audit these attempts:

☑ Success

☑ Failure

Figure 2.3: Checking the boxes for Success and Failure

Chapter 3: Monitoring Event Logs

The eG Enterprise suite provides for a special server type named Event Log. The layer model of an Event Log server is as depicted below:

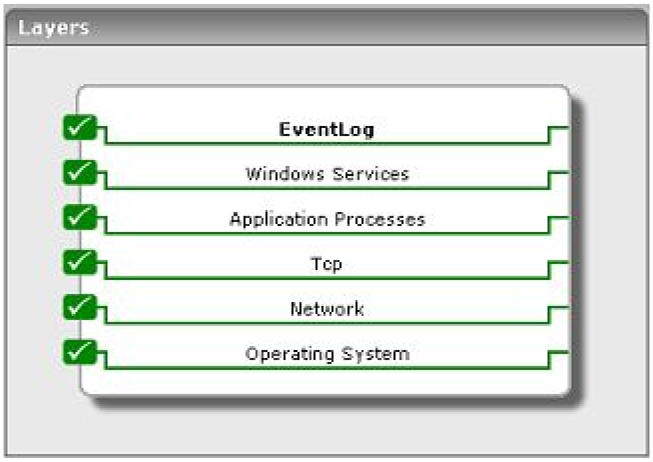


Figure 3.1: The layer model of an EventLog server

The 5 layers at the bottom of Figure 3.1 have been dealt with extensively in the *Monitoring Unix and Windows Servers* document. The following section will throw light on the **EventLog** layer, which is the upper most layer.

3.1 The EventLog Layer

This layer monitors the system, application, and security logs on the Windows host, and reports the number of errors/warnings/general information events that have occurred on the host.

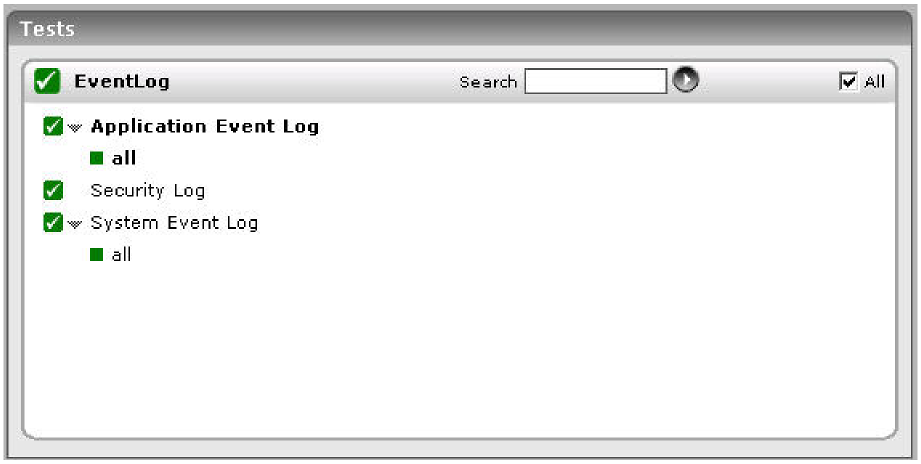


Figure 3.2: Figure 1.2: Test executing on the EventLog layer

3.1.1 Application Event Log Test

This test reports the statistical information about the application events generated by the target system.

Target of the test : Any host system

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Filter configured.

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed
Host	The IP address of the host for which this test is to be configured.
Port	Refers to the port used by the EventLog Service. Here it is null.
LogType	Refers to the type of event logs to be monitored. The default value is *application*.
Policy Based Filter	Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options: • Manually specify the event sources, IDs, and descriptions in the Filter text area, or, • Select a specification from the predefined filter policies listed in the Filter box

Parameter	Description
	For explicit, manual specification of the filter conditions, select the **No** option against the Policy Based Filter field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **Yes** option against the Policy Based Filter field.
Filter	If the Policy based Filter flag is set to **No**, then a Filter text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the FILTER text area takes the value, OS_events:all:Browse,Print:all:none:all:none. Here: • *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI; • *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify none. • Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use all to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded. • In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The all in our example represents that all the event IDs need to be considered while monitoring. • Similarly, the *none* (following all in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying all makes sure that all the event IDs are excluded from monitoring. • The *all* which follows implies that all events, regardless of description, need to

Parameter	Description
	be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - _desc*_, or desc, or _*desc*_,or _desc*_, or _desc1*desc2_, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. • In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: _desc*_, or desc, or _*desc*,or desc*_, or _desc1*desc2_, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring. By default, the Filter parameter contains the value: _all:all:none:all:none:all:none_. Multiple filters are to be separated by semi-colons (;). **Note:** The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window. On the other hand, if the Policy Based Filter flag is set to **Yes**, then a Filter list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format: *{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}* To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the Filter list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link

Parameter	Description
	appears just above the test configuration section, once the **Yes** option is chosen against Policy Based Filter. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one (refer to Adding a New Policy). The changed policy or the new policy can then be associated with the test by selecting the policy name from the Filter list box in this page.
UseWMI	The eG agent can either use **WMI** to extract event log statistics or directly parse the event logs using **event log API**s. If the UseWMI flag is **Yes**, then **WMI** is used. If not, the **event log API**s are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the **WinMgmt** process to shoot up. On such systems, set the UseWMI parameter value to **No**. On the other hand, **when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the UseWMI flag should always be set to 'Yes'**.
Stateless Alerts	Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the **EventLog** test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the Stateless Alerts flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.
Events During Restart	By default, the Events During Restart flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.
DDForInformation	eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the DDForInformation and DDForWarning flags

Parameter	Description
	have been made available in this page. By default, both these flags are set to **Yes**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the DDForInformation flag to **No**.
DDForWarning	To ensure that the test does not generate and store detailed measures for warning events, set the DDForWarning flag to **No**.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against DD Frequency.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enabled/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

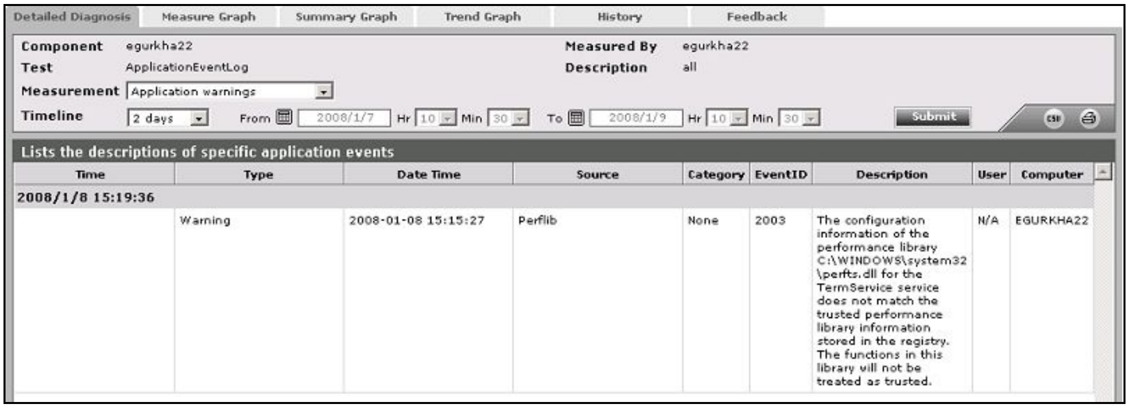
Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Application errors	This refers to the number of application error events that were generated.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of problems like loss of functionality or data in one or more applications.

Measurement	Description	Measurement Unit	Interpretation
			Please check the Application Logs in the Event Log Viewer for more details.
Application information count	This refers to the number of application information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed by one or more applications. Please check the Application Logs in the Event Log Viewer for more details.
Application warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates application problems that may not have an immediate impact, but may cause future problems in one or more applications. Please check the Application Logs in the Event Log Viewer for more details.
Application critical errors	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that an application or a component cannot automatically recover from. This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in one or more applications. The detailed diagnosis of this measure describes all the critical application events that were generated during the last measurement period. Please check the Application Logs in the Event Log Viewer for more details.
Application verbose	Indicates the number of	Number	Verbose logging provides more details

Measurement	Description	Measurement Unit	Interpretation
	verbose events that were generated when the test was last executed.		in the log entry, which will enable you to troubleshoot issues better. This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the Application Logs in the Event Log Viewer for more details.

The detailed diagnosis of the *Application warnings* measure, if enabled, describes all the application warnings that were generated during the last measurement period.



Component egurkha22

Test ApplicationEventLog

Measurement Application warnings

Timeline 2 days From 2008/1/7 Hr 10 Min 30 To 2008/1/9 Hr 10 Min 30

Measured By egurkha22

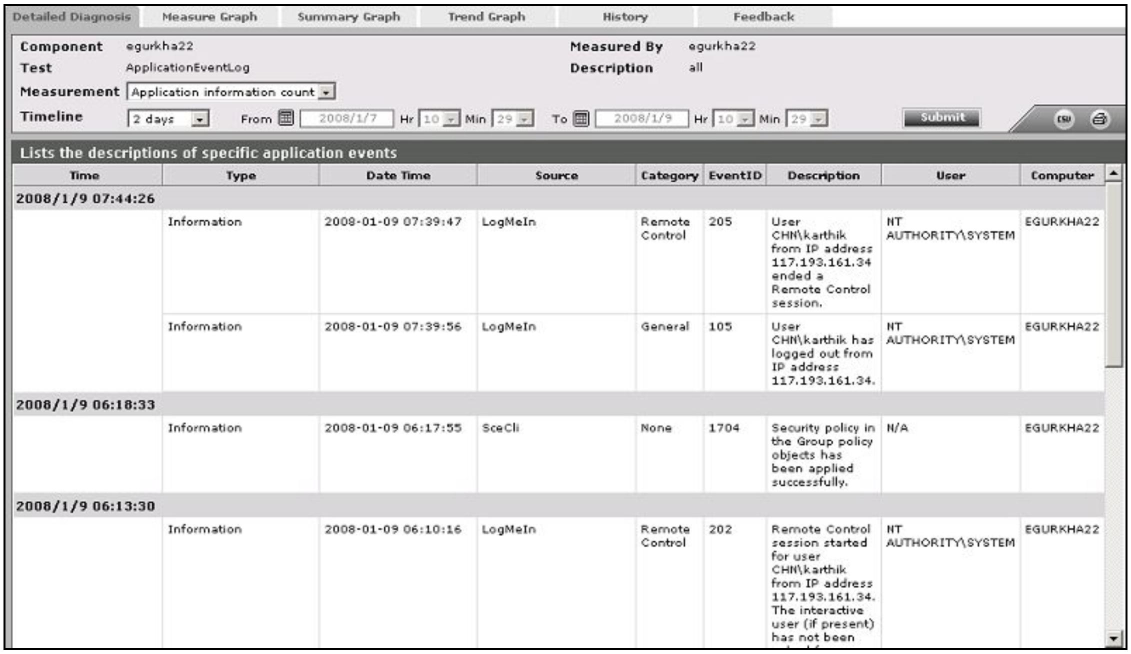
Description all

Lists the descriptions of specific application events

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/8 15:19:36								
	Warning	2008-01-08 15:15:27	Perflib	None	2003	The configuration information of the performance library C:\WINDOWS\system32\perfts.dll for the TermService service does not match the trusted performance library information stored in the registry. The functions in this library vill not be treated as trusted.	N/A	EGURKHA22

Figure 3.3: The detailed diagnosis of the Application warnings measure

The detailed diagnosis of the *Application information count* measure, if enabled, describes all the general information events that were generated during the last measurement period.



Component egurkha22

Test ApplicationEventLog

Measurement Application information count

Timeline 2 days From 2008/1/7 Hr 10 Min 29 To 2008/1/9 Hr 10 Min 29

Measured By egurkha22

Description all

Lists the descriptions of specific application events

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/9 07:44:26								
	Information	2008-01-09 07:39:47	LogMeIn	Remote Control	205	User CHN\karthik from IP address 117.193.161.34 ended a Remote Control session.	NT AUTHORITY\SYSTEM	EGURKHA22
	Information	2008-01-09 07:39:56	LogMeIn	General	105	User CHN\karthik has logged out from IP address 117.193.161.34.	NT AUTHORITY\SYSTEM	EGURKHA22
2008/1/9 06:18:33								
	Information	2008-01-09 06:17:55	SceCli	None	1704	Security policy in the Group policy objects has been applied successfully.	N/A	EGURKHA22
2008/1/9 06:13:30								
	Information	2008-01-09 06:10:16	LogMeIn	Remote Control	202	Remote Control session started for user CHN\karthik from IP address 117.193.161.34. The interactive user (if present) has not been	NT AUTHORITY\SYSTEM	EGURKHA22

Figure 3.4: The detailed diagnosis of the Application information count measure

1. The filter policy for the **Application EventLog** test, **Application Events** test, **System Events** test, and **System EventLog** test typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is expressed by the eG Enterprise system in the following format:

```
{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}
```

2. On the other hand, the filter policy for the **Security Log** test comprises of a specific set of event sources, event ids, and users to be monitored. This specification is expressed by the eG Enterprise system in the following format:

```
{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_ included}: {event_ IDs_ to_ be_ excluded}: {users_ to_ be_ included}: {users_ to_ be_ excluded}
```

3.1.2 Event Log Test

This test reports the statistical information about the events generated by various applications and windows services and drivers in the target system. This test is disabled by default. This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick *Eventlog* as the **Component type**, set

Performance as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **<** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : Any host system

Agent deploying the test : An internal agent

Outputs of the test : One set of results for server being monitored.

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed
Host	The IP address of the host for which this test is to be configured.
Port	Refers to the port used by the EventLog Service. Here it is null.
EventHost	This is the same as the Host.
EventSrc	Enter the specific events to be monitored in the EventSrc text box. The name of the event source can be obtained from the Event Viewer window that appears on following the menu sequence: *Start -> Programs -> Administrative Tools -> Event Viewer* (If the Programs menu does not contain the Administrative Tools option, then check *Start->Settings ->Control Panel* for the same). The value that appears in the Source column of this window should be used to specify the EventSrc parameter. By default, "*All*" will be displayed against EventSrc indicating that all events will be monitored by default. While specifying multiple events, make sure that they are separated by commas (,). **Note:** The EventSrc specified should be exactly the same as that which appears in the Event Viewer window.
Excludedsrc	If specific events are to be excluded from monitoring, then specify the events to be excluded in the Excludedsrc text box, as a comma-separated list.
UseWMI	The eG agent can either use **WMI** to extract event log statistics or directly parse the event logs using **event log API**s. If the UseWMI flag is **Yes**, then **WMI** is used. If not, the **event log API**s are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the **WinMgmt** process to shoot up. On such systems, set the UseWMI parameter value to **No**. On the other hand, **when monitoring systems that are operating on any other flavor of Windows (say,**

Parameter	Description
	Windows 2003/XP/2008/7/Vista/12), the UseWMI flag should always be set to 'Yes'.
Stateless Alerts	Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the **EventLog** test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the Stateless Alerts flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against DD Frequency.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enabled/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Application errors	This refers to the number of application error events that were generated.	Number	A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems. An increasing trend or high value indicates the existence of problems like loss of functionality or data in one or more applications. Please check the Application Logs in the Event Log Viewer for more details.
Application information messages	This refers to the number of application information events generated when the test was last executed.	Number	A change in the value of this measure may indicate infrequent but successful operations performed by one or more applications. Please check the Application Logs in the Event Log Viewer for more details.
Application warnings	This refers to the number of warnings that were generated when the test was last executed.	Number	A high value of this measure indicates application problems that may not have an immediate impact, but may cause future problems in one or more applications. Please check the Application Logs in the Event Log Viewer for more details.
Application critical errors	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that an application or a component cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates that the system is in a healthy state and all applications are running smoothly without any potential problems.

Measurement	Description	Measurement Unit	Interpretation
			An increasing trend or high value indicates the existence of fatal/irrepairable problems in one or more applications. The detailed diagnosis of this measure describes all the critical application events that were generated during the last measurement period. Please check the Application Logs in the Event Log Viewer for more details.
Application verbose	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the Application Logs in the Event Log Viewer for more details.
System errors	This refers to the number of system error events generated during the last execution of the test.	Number	A very low value (zero) indicates that the system is in healthy state and all Windows services and low level drivers are running without any potential problems. An increasing trend or a high value indicates the existence of problems such as loss of functionality or data in one or more Windows services and low level drivers. Please check the System Logs in the Event Log Viewer for more details.
System information	This refers to the number	Number	A change in value of this measure may

Measurement	Description	Measurement Unit	Interpretation
messages	of service-related and driver-related information events that were generated during the test's last execution.		indicate infrequent but successful operations performed by one or more applications. Please check the System Logs in the Event Log Viewer for more details.
System warnings	This refers to the number of service-related and driver-related warnings generated in the during the test's last execution.	Number	A high value of this measure indicates problems that may not have an immediate impact, but may cause future problems in one or more Windows servers and low level drivers. Please check the System Logs in the Event Log Viewer for more details.
System critical errors	Indicates the number of critical events that were generated when the test was last executed.	Number	A critical event is one that a system cannot automatically recover from. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates that the system is in a healthy state and is running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in the system. The detailed diagnosis of this measure describes all the critical system events that were generated during the last measurement period. Please check the System Logs in the Event Log Viewer for more details.
System verbose	Indicates the number of verbose events that were generated when the test	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better.

Measurement	Description	Measurement Unit	Interpretation
	was last executed.		This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the System Logs in the Event Log Viewer for more details.

Note:

The Stateless Alerting capability is currently available for the following tests alone, by default:

- EventLog test
- Application EventLog test
- System EventLog test
- Application Events test
- System Events test
- Security Log test
- Account Management Events test

If need be, you can enable the **stateless alerting** capability for other tests. To achieve this, follow the steps given below:

- Login to the eG manager host.
- Edit the **eg_specs.ini** file in the <EG_INSTALL_DIR>\manager\config directory.
- Locate the test for which the **Stateless Alarms** flag has to be enabled.
- Insert the entry, **-statelessAlerts yes**, into the test specification as depicted below:

  ```
  EventLogTest::$hostName:$portNo=$hostName, -auto, -host $hostName -port $portNo -
  eventhost $hostIp -eventsrc all -excludedSrc none -useWmi yes -statelessAlerts yes
  -ddFreq 1:1 -rptName $hostName, 300
  ```

- Finally, save the file.

If need be, you can change the status of the **statelessAlerts** flag by reconfiguring the test in the eG administrative interface.

Once the **stateless alerting capability** is enabled for a test (as discussed above), you will find that everytime the test reports a problem, the eG manager does the following:

- Closes the alarm that pre-exists for that problem;
- Sends out a normal alert indicating the closure of the old problem;
- Opens a new alarm and assigns a new alarm ID to it;
- Sends out a fresh email alert to the configured users, intimating them of the new issue.

In a redundant manager setup, the secondary manager automatically downloads the updated **eg_ specs.ini** file from the primary manager, and determines whether the stateless alerting capability has been enabled for any of the tests reporting metrics to it. If so, everytime a threshold violation is detected by such a test, the secondary manager will perform the tasks discussed above for the problem reported by that test. Similarly, the primary manager will check whether the stateless alert flag has been switched on for any of the tests reporting to it, and if so, will automatically perform the above-mentioned tasks whenever those tests report a deviation from the norm.

Note:

- Since alerts will be closed after every measurement period, alarm escalation will no longer be relevant for tests that have **statelessAlerts** set to **yes**.
- For tests with **statelessAlerts** set to **yes**, **statelessAlerts** will apply for all measurements of that test (i.e., it will not be possible to only have one of the measurements with stateless alerts and others without).
- If **statelessAlerts** is set to **yes** for a test, an alarm will be opened during one measurement period (if a threshold violation happens) and will be closed prior to the next measurement period. This way, if a threshold violation happens in successive measurement periods, there will be one alarm per measurement period. This will reflect in all the corresponding places in the eG Enterprise system. For example, multiple alerts in successive measurement periods will result in multiple trouble tickets being opened (one for each measurement period). Likewise, the alarm history will also show alarms being opened during a measurement period and closed during the next measurement period.

3.1.3 System Event Log Test

This test reports the statistical information about the system events generated by the target system.

Target of the test : Any host system

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Filter configured.

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed
Host	The IP address of the host for which this test is to be configured.
Port	Refers to the port used by the EventLog Service. Here it is null.
LogType	Refers to the type of event logs to be monitored. The default value is *application*.
Policy Based Filter	Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options: • Manually specify the event sources, IDs, and descriptions in the Filter text area, or, • Select a specification from the predefined filter policies listed in the Filter box For explicit, manual specification of the filter conditions, select the **No** option against the Policy Based Filter field. This is the default selection. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **Yes** option against the Policy Based Filter field.
Filter	If the Policy based Filter flag is set to **No**, then a Filter text area will appear, wherein you will have to specify the event sources, event IDs, and event descriptions to be monitored. This specification should be of the following format: *{Displayname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}*. For example, assume that the FILTER text area takes the value, OS_events:all:Browse,Print:all:none:all:none. Here: • *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI; • *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify none.

Parameter	Description
	• Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, _Browse_ and _Print_ have been excluded from monitoring. Alternatively, you can use all to indicate that all the event sources have to be excluded from monitoring, or _none_ to denote that none of the event sources need be excluded. • In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The all in our example represents that all the event IDs need to be considered while monitoring. • Similarly, the _none_ (following all in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying all makes sure that all the event IDs are excluded from monitoring. • The _all_ which follows implies that all events, regardless of description, need to be included for monitoring. To exclude all events, use none. On the other hand, if you provide a comma-separated list of event descriptions, then the events with the specified descriptions will alone be monitored. Event descriptions can be of any of the following forms - _desc*_, or desc, or _*desc*_,or _desc*_, or _desc1*desc2_, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. • In the same way, you can also provide a comma-separated list of event descriptions to be excluded from monitoring. Here again, the specification can be of any of the following forms: _desc*_, or desc, or _*desc*,or desc*_, or _desc1*desc2_, etc. desc here refers to any string that forms part of the description. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. In our example however, none is specified, indicating that no event descriptions are to be excluded from monitoring. If you use all instead, it would mean that all event descriptions are to be excluded from monitoring. By default, the Filter parameter contains the value: _all:all:none:all:none:all:none_.

Parameter	Description
	Multiple filters are to be separated by semi-colons (;). **Note:** The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window. On the other hand, if the Policy Based Filter flag is set to **Yes**, then a Filter list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and event descriptions to be monitored. This specification is built into the policy in the following format: *{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{event_descriptions_to_be_included}:{event_descriptions_to_be_excluded}* To monitor a specific combination of event sources, event IDs, and event descriptions, you can choose the corresponding filter policy from the Filter list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **Yes** option is chosen against Policy Based Filter. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one (refer to Adding a New Policy). The changed policy or the new policy can then be associated with the test by selecting the policy name from the Filter list box in this page.
UseWMI	The eG agent can either use **WMI** to extract event log statistics or directly parse the event logs using **event log APIs**. If the UseWMI flag is **Yes**, then **WMI** is used. If not, the **event log APIs** are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the **WinMgmt** process to shoot up. On such systems, set the UseWMI parameter value to **No**. On the other hand, **when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the UseWMI flag should always be set to 'Yes'**.
Stateless Alerts	Typically, the eG manager generates email alerts only when the state of a specific measurement changes. A state change typically occurs only when the threshold of a measure is violated a configured number of times within a specified time window. While this ensured that the eG manager raised alarms only when the problem was severe enough, in some cases, it may cause one/more problems to go unnoticed, just because they did not result in a state change. For example, take the case of the

Parameter	Description
	EventLog test. When this test captures an error event for the very first time, the eG manager will send out a **CRITICAL** email alert with the details of the error event to configured recipients. Now, the next time the test runs, if a different error event is captured, the eG manager will keep the state of the measure as **CRITICAL**, but will not send out the details of this error event to the user; thus, the second issue will remain hidden from the user. To make sure that administrators do not miss/overlook critical issues, the eG Enterprise monitoring solution provides the **stateless alerting** capability. To enable this capability for this test, set the Stateless Alerts flag to **Yes**. This will ensure that email alerts are generated for this test, regardless of whether or not the state of the measures reported by this test changes.
Events During Restart	By default, the Events During Restart flag is set to **Yes**. This ensures that whenever the agent is stopped and later started, the events that might have occurred during the period of non-availability of the agent are included in the number of events reported by the agent. Setting the flag to **No** ensures that the agent, when restarted, ignores the events that occurred during the time it was not available.
DDForInformation	eG Enterprise also provides you with options to restrict the amount of storage required for event log tests. Towards this end, the DDForInformation and DDForWarning flags have been made available in this page. By default, both these flags are set to **Yes**, indicating that by default, the test generates detailed diagnostic measures for information events and warning events. If you do not want the test to generate and store detailed measures for information events, set the DDForInformation flag to **No**.
DDForWarning	To ensure that the test does not generate and store detailed measures for warning events, set the DDForWarning flag to **No**.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against DD Frequency.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enabled/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled:

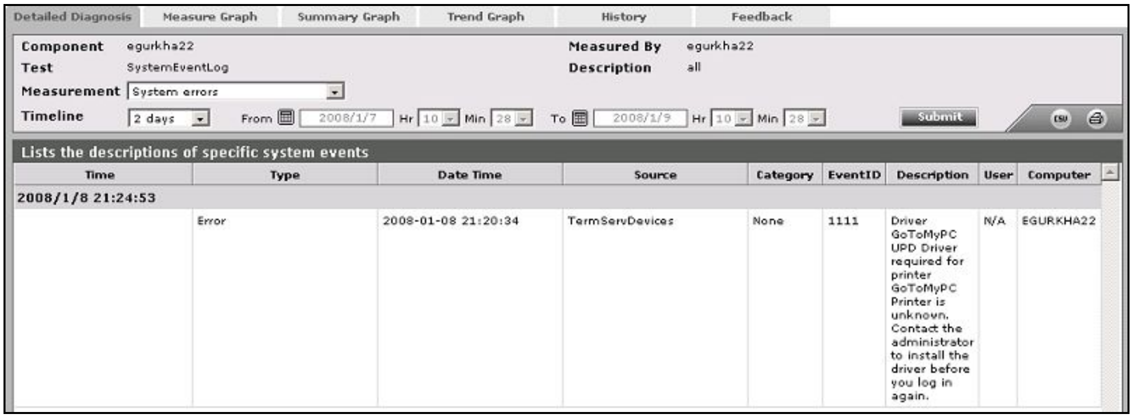
Parameter	Description
	- The eG manager license should allow the detailed diagnosis capability - Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
System Errors	This refers to the number of system error events generated during the last execution of the test.	Number	A very low value (zero) indicates that the system is in healthy state and all Windows services and low level drivers are running without any potential problems. An increasing trend or a high value indicates the existence of problems such as loss of functionality or data in one or more Windows services and low level drivers. Please check the System Logs in the Event Log Viewer for more details.
System information messages	This refers to the number of service-related and driver-related information events that were generated during the test's last execution.	Number	A change in value of this measure may indicate infrequent but successful operations performed by one or more applications. Please check the System Logs in the Event Log Viewer for more details.
System warnings	This refers to the number of service-related and driver-related warnings generated in the during the test's last execution.	Number	A high value of this measure indicates problems that may not have an immediate impact, but may cause future problems in one or more Windows servers and low level drivers. Please check the System Logs in the Event Log Viewer for more details.
System critical errors	Indicates the number of critical events that were	Number	A critical event is one that a system cannot automatically recover from.

Measurement	Description	Measurement Unit	Interpretation
	generated when the test was last executed.		**This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. A very low value (zero) indicates that the system is in a healthy state and is running smoothly without any potential problems. An increasing trend or high value indicates the existence of fatal/irrepairable problems in the system. The detailed diagnosis of this measure describes all the critical system events that were generated during the last measurement period. Please check the System Logs in the Event Log Viewer for more details.
System verbose	Indicates the number of verbose events that were generated when the test was last executed.	Number	Verbose logging provides more details in the log entry, which will enable you to troubleshoot issues better. **This measure is applicable only for Windows 2008/Windows Vista/Windows 7 systems**. The detailed diagnosis of this measure describes all the verbose events that were generated during the last measurement period. Please check the System Logs in the Event Log Viewer for more details.

The detailed diagnosis of the *System errors* measure, provides detailed descriptions of the system errors that occurred on the host during the last measurement period.



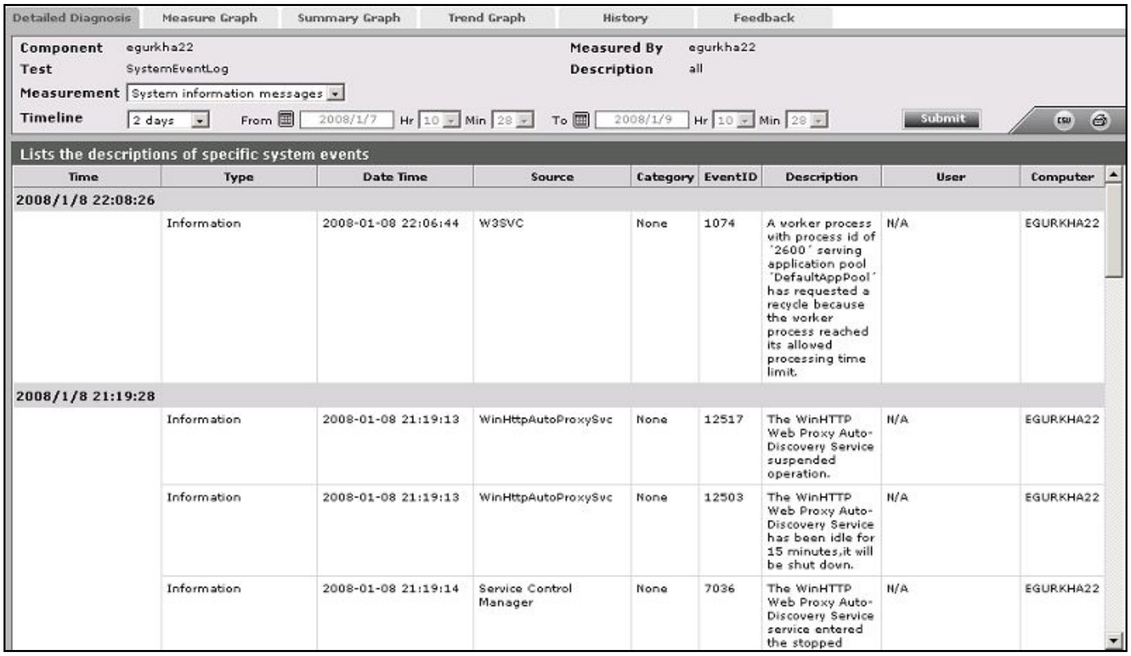
Component egurkha22
Test SystemEventLog
Measurement System errors
Timeline 2 days From 2008/1/7 Hr 10 Min 28 To 2008/1/9 Hr 10 Min 28
Measured By egurkha22
Description all

Lists the descriptions of specific system events

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/8 21:24:53								
	Error	2008-01-08 21:20:34	TermServDevices	None	1111	Driver GoToMyPC UPD Driver required for printer GoToMyPC Printer is unknown. Contact the administrator to install the driver before you log in again.	N/A	EGURKHA22

Figure 3.5: The detailed diagnosis of the System errors measure

The detailed diagnosis of the *System information messages* measure, provides detailed descriptions of the information events that occurred on the host during the last measurement period.



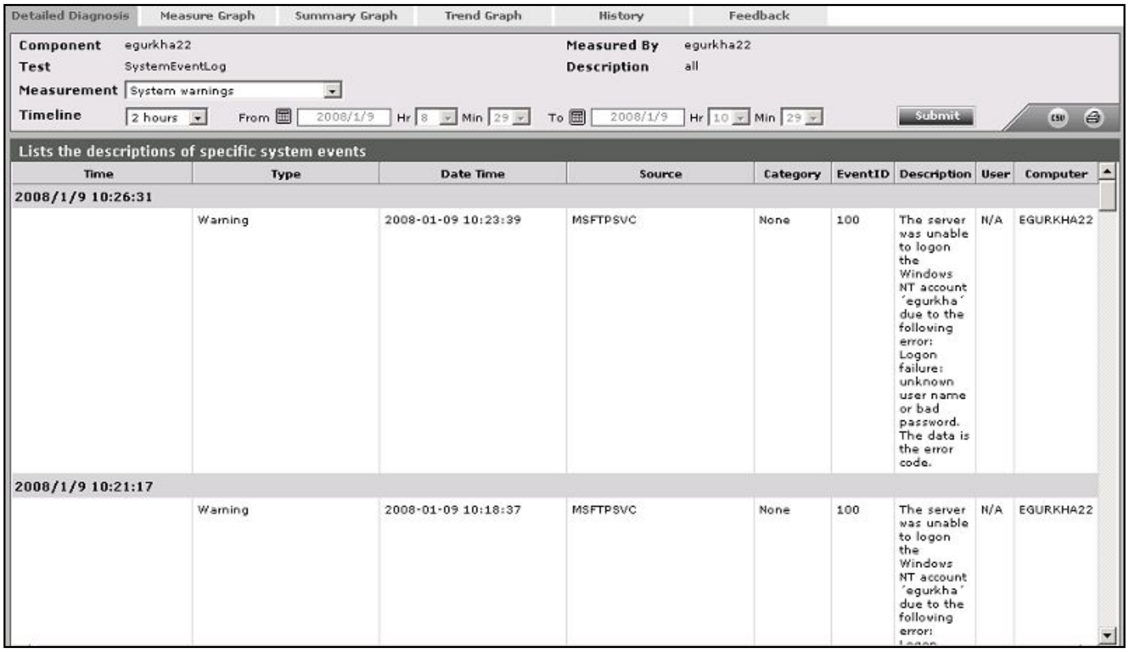
Component egurkha22
Test SystemEventLog
Measurement System information messages
Timeline 2 days From 2008/1/7 Hr 10 Min 28 To 2008/1/9 Hr 10 Min 28
Measured By egurkha22
Description all

Lists the descriptions of specific system events

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/8 22:08:26								
	Information	2008-01-08 22:06:44	W3SVC	None	1074	A worker process with process id of '2600' serving application pool 'DefaultAppPool' has requested a recycle because the worker process reached its allowed processing time limit.	N/A	EGURKHA22
2008/1/8 21:19:28								
	Information	2008-01-08 21:19:13	WinHttpAutoProxySvc	None	12517	The WinHTTP Web Proxy Auto-Discovery Service suspended operation.	N/A	EGURKHA22
	Information	2008-01-08 21:19:13	WinHttpAutoProxySvc	None	12503	The WinHTTP Web Proxy Auto-Discovery Service has been idle for 15 minutes,it will be shut down.	N/A	EGURKHA22
	Information	2008-01-08 21:19:14	Service Control Manager	None	7036	The WinHTTP Web Proxy Auto-Discovery Service service entered the stopped	N/A	EGURKHA22

Figure 3.6: The detailed diagnosis of the System information messages measure

The detailed diagnosis of the *System warnings* measure, provides detailed descriptions of the warning events that occurred on the host during the last measurement period.



Component egurkha22
Test SystemEventLog
Measurement System warnings
Timeline 2 hours From 2008/1/9 Hr 8 Min 29 To 2008/1/9 Hr 10 Min 29

Measured By egurkha22
Description all

Lists the descriptions of specific system events

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/9 10:26:31								
	Warning	2008-01-09 10:23:39	MSFTPSVC	None	100	The server was unable to logon the Windows NT account ´egurkha´ due to the following error: Logon failure: unknown user name or bad password. The data is the error code.	N/A	EGURKHA22
2008/1/9 10:21:17								
	Warning	2008-01-09 10:18:37	MSFTPSVC	None	100	The server was unable to logon the Windows NT account ´egurkha´ due to the following error:	N/A	EGURKHA22

Figure 3.7: The detailed diagnosis of the System warnings measure

3.1.4 Security Log Test

The Security Log test reports statistics relating to the Windows security log audits. **Note that this test will not work on Windows Vista**.

Target of the test : Any host system

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the server being monitored.

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed
Host	The IP address of the host for which this test is to be configured.
Port	Refers to the port used by the EventLog Service. Here it is *null*.
LogType	Refers to the type of event logs to be monitored. The default value is *security*.
SuccessEventsinDD	By default, this parameter displays *none*, indicating that by default none of the successful log audits will be reflected in the detailed diagnosis. If you set this parameter to, say 10, then the test will display only the 10 most recent successful log

Parameter	Description
	audits in the detailed diagnosis page. Setting this parameter to all, on the other hand will make sure that all successful log audits are listed in the detailed diagnosis.
FailureEventsinDD	By default, this parameter displays *all*, indicating that by default all the failed log audits will be reflected in the detailed diagnosis. If you set this parameter to, say 10, then the test will display only the 10 most recent log audits that failed, in the detailed diagnosis page. Setting this parameter to none, on the other hand will make sure that none of the failed log audits are listed in the detailed diagnosis.
UseWMI	The eG agent can either use **WMI** to extract event log statistics or directly parse the event logs using **event log API**s. If the UseWMI flag is **Yes**, then **WMI** is used. If not, the **event log API**s are used. This option is provided because on some Windows NT/2000 systems (especially ones with service pack 3 or lower), the use of WMI access to event logs can cause the CPU usage of the **WinMgmt** process to shoot up. On such systems, set the UseWMI parameter value to **No**. On the other hand, **when monitoring systems that are operating on any other flavor of Windows (say, Windows 2003/XP/2008/7/Vista/12), the UseWMI flag should always be set to 'Yes'**.
Policy Based Filter	Using this page, administrators can configure the event sources, event IDs, and event descriptions to be monitored by this test. In order to enable administrators to easily and accurately provide this specification, this page provides the following options: • Manually specify the event sources, IDs, and users in the Filter text area, or, • Select a specification from the predefined filter policies listed in the Filter box For explicit, manual specification of the filter conditions, select the **No** option against the Policy Based Filter field. To choose from the list of pre-configured filter policies, or to create a new filter policy and then associate the same with the test, select the **Yes** option against the Policy Based Filter field. This is the default selection.
Filter	If the Policy based Filter flag is set to **No**, then a Filter text area will appear, wherein you will have to specify the event sources, event IDs, and event users to be monitored. This specification should be of the following format: {Displayname}:{event_sources_ to_be_included}:{event_sources_to_be_excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{users_to_be_included}:{ users_to_be_excluded}. For example, assume that the Filter text area takes the value, *OS_ events:all:Browse,Print:all:none:all:none*. Here: • *OS_events* is the display name that will appear as a descriptor of the test in the monitor UI;

Parameter	Description
	• *all* indicates that all the event sources need to be considered while monitoring. To monitor specific event sources, provide the source names as a comma-separated list. To ensure that none of the event sources are monitored, specify *none*. • Next, to ensure that specific event sources are excluded from monitoring, provide a comma-separated list of source names. Accordingly, in our example, *Browse* and *Print* have been excluded from monitoring. Alternatively, you can use *all* to indicate that all the event sources have to be excluded from monitoring, or *none* to denote that none of the event sources need be excluded. • In the same manner, you can provide a comma-separated list of event IDs that require monitoring. The *all* in our example represents that all the event IDs need to be considered while monitoring. • Similarly, the *none* (following *all* in our example) is indicative of the fact that none of the event IDs need to be excluded from monitoring. On the other hand, if you want to instruct the eG Enterprise system to ignore a few event IDs during monitoring, then provide the IDs as a comma-separated list. Likewise, specifying all makes sure that all the event IDs are excluded from monitoring. • In the same way, you can also ensure that events generated by specific users on the target host are alone tracked by providing a comma-separated list of users to be monitored – for example, *john,elvis*. In our example however, *all* is specified, indicating that *all* users need be monitored. • You can similarly indicate if specific users need to be excluded from monitoring. In our example however, *none* is provided to ensure that no users are excluded from monitoring. **Note:** The event sources and event IDs specified here should be exactly the same as that which appears in the Event Viewer window. On the other hand, if the Policy Based Filter flag is set to **Yes**, then a Filter list box will appear, displaying the filter policies that pre-exist in the eG Enterprise system. A filter policy typically comprises of a specific set of event sources, event IDs, and users to be monitored. This specification is built into the policy in the following format: *{Policyname}:{event_sources_to_be_included}:{event_sources_to_be_*

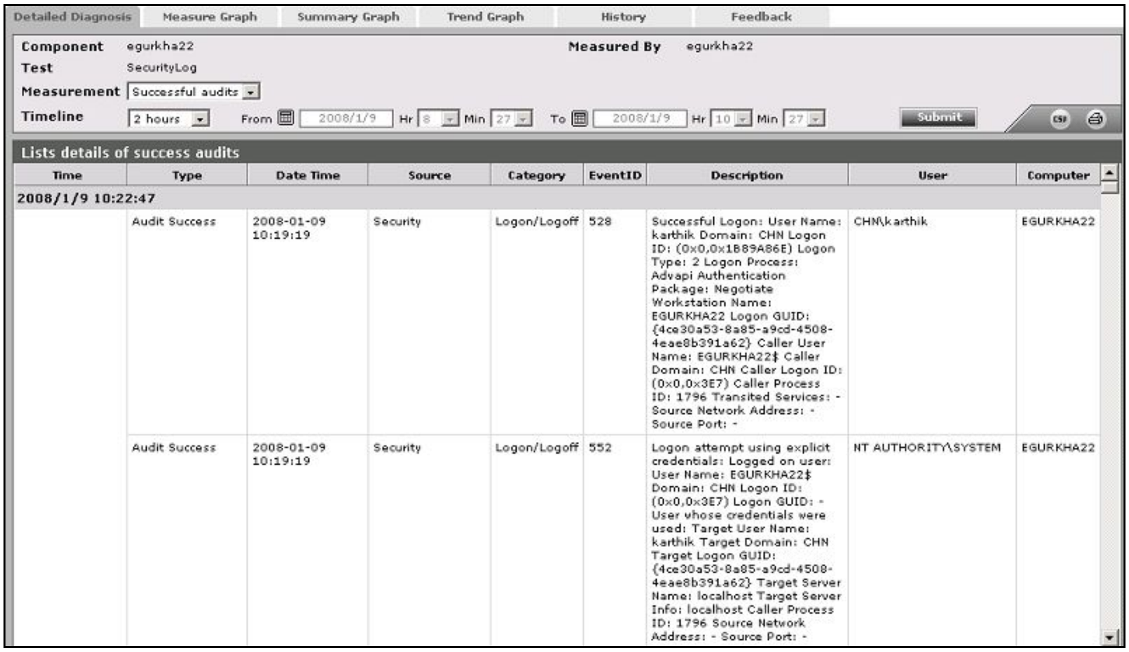
Parameter	Description
	excluded}:{event_IDs_to_be_included}:{event_IDs_to_be_excluded}:{users_to_be_included}:{users_to_be_excluded} To monitor a specific combination of event sources, event IDs, and users, you can choose the corresponding filter policy from the Filter list box. Multiple filter policies can be so selected. Alternatively, you can modify any of the existing policies to suit your needs, or create a new filter policy. To facilitate this, a **Click here** link appears just above the test configuration section, once the **Yes** option is chosen against Policy Based Filter. Clicking on the **Click here** link leads you to a page where you can modify the existing policies or create a new one. The changed policy or the new policy can then be associated with the test by selecting the policy name from the Filter list box in this page.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against DD Frequency.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enabled/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Successful audits	Indicates the number of successful audits of windows security logs.	Number	The detailed diagnosis of this measure, if enabled, provides the details of the successful log audits.
Failure audits	Indicates the number of	Number	The detailed diagnosis of this

Measurement	Description	Measurement Unit	Interpretation
	windows security log audits that failed.		measure, if enabled, provides the details of the failed log audits.

The detailed diagnosis of the *Successful audits* measure, if enabled, provides the details of the successful log audits.



Component egurkha22

Measured By egurkha22

Test SecurityLog

Measurement Successful audits

Timeline 2 hours From 2008/1/9 Hr 8 Min 27 To 2008/1/9 Hr 10 Min 27

Lists details of success audits

Time	Type	Date Time	Source	Category	EventID	Description	User	Computer
2008/1/9 10:22:47								
	Audit Success	2008-01-09 10:19:19	Security	Logon/Logoff	528	Successful Logon: User Name: karthik Domain: CHN Logon ID: (0x0,0x1B89A86E) Logon Type: 2 Logon Process: Advapi Authentication Package: Negotiate Workstation Name: EGURKHA22 Logon GUID: {4ce30a53-8a85-a9cd-4508-4eae8b391a62} Caller User Name: EGURKHA22$ Caller Domain: CHN Caller Logon ID: (0x0,0x3E7) Caller Process ID: 1796 Transited Services: - Source Network Address: - Source Port: -	CHN\karthik	EGURKHA22
	Audit Success	2008-01-09 10:19:19	Security	Logon/Logoff	552	Logon attempt using explicit credentials: Logged on user: User Name: EGURKHA22$ Domain: CHN Logon ID: (0x0,0x3E7) Logon GUID: - User vhose credentials were used: Target User Name: karthik Target Domain: CHN Target Logon GUID: {4ce30a53-8a85-a9cd-4508-4eae8b391a62} Target Server Name: localhost Target Server Info: localhost Caller Process ID: 1796 Source Network Address: - Source Port: -	NT AUTHORITY\SYSTEM	EGURKHA22

Figure 3.8: The detailed diagnosis of the Successful audits measure

About eG Innovations

eG Innovations provides intelligent performance management solutions that automate and dramatically accelerate the discovery, diagnosis, and resolution of IT performance issues in on-premises, cloud and hybrid environments. Where traditional monitoring tools often fail to provide insight into the performance drivers of business services and user experience, eG Innovations provides total performance visibility across every layer and every tier of the IT infrastructure that supports the business service chain. From desktops to applications, from servers to network and storage, from virtualization to cloud, eG Innovations helps companies proactively discover, instantly diagnose, and rapidly resolve even the most challenging performance and user experience issues.

eG Innovations is dedicated to helping businesses across the globe transform IT service delivery into a competitive advantage and a center for productivity, growth and profit. Many of the world's largest businesses use eG Enterprise to enhance IT service performance, increase operational efficiency, ensure IT effectiveness and deliver on the ROI promise of transformational IT investments across physical, virtual and cloud environments.

To learn more visit www.eginnovations.com.

Contact Us

For support queries, email support@eginnovations.com.

To contact eG Innovations sales team, email sales@eginnovations.com.