Additional Tests offered by eG Enterprise

eG Innovations Product Documentation

www.eginnovations.com

Table of Contents

Chapter 1: Additional Tests

The eG Enterprise suite provides for a few in-built tests that can be associated with any existing server type or new server type that is added using the **Integration Console** utility.

Note:

The tests discussed in this document will not be available for any of the existing (i.e. built-in) server types. If need be, you can associate one/more of these tests to an existing server-type/layer using the licensed **eG Integration Console** component.

1.1 Application Traps Test

This test reports the number of SNMP trap messages sent on account of errors in the transactions of various applications.

Target of the test : An SNMP trap

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the application listens.
SourceAddress	Specify a comma-separated list of IP addresses or address patterns of the hosts sending the traps. For example, *10.0.0.1,192.168.10.**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
OIDValue	Provide a comma-separated list of OID and value pairs returned by the traps. The values are to be expressed in the form, *DisplayName:OID-OIDValue*. For example, assume that the following OIDs are to be considered by this test: *.1.3.6.1.4.1.9156.1.1.2 and .1.3.6.1.4.1.9156.1.1.3*. The values of these OIDs are as given hereunder:

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><th>OID</th><th>Value</th></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.2</td><td>Host_system</td></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.3</td><td>NETWORK</td></tr>
</table>

In this case the oidvalue parameter can be configured as Trap1:*.1.3.6.1.4.1.9156.1.1.2-Host_system*,Trap2:*.1.3.6.1.4.1.9156.1.1.3-Network*, where Trap1 and Trap2 are the display names that appear as descriptors of this test in the monitor interface.
The test considers a configured OID for monitoring only when the actual value of the OID matches with its configured value. For instance, in the example above, if the value of OID *.1.3.6.1.4.1.9156.1.1.2* is found to be Host and not Host_system, then the test ignores OID *.1.3.6.1.4.1.9156.1.1.2* while monitoring.
An * can be used in the OID/value patterns to denote any number of leading or trailing characters (as the case may be). For example, to monitor all the OIDs that return values which begin with the letter 'F', set this parameter to Failed:*-F*.</td></tr>
<tr><td>ShowOID</td><td>Selecting the **True** option against ShowOID will ensure that the detailed diagnosis of this test shows the OID strings along with their corresponding values. If you select **False**, then the values alone will appear in the detailed diagnosis page, and not the OIDs.</td></tr>
<tr><td>Detailed Diagnosis</td><td>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 enable/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.</td></tr>
</table>

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
SNMP traps received	Indicates the number of	Number	The detailed diagnosis of this

Measurement	Description	Measurement Unit	Interpretation
	trap messages sent since the last measurement period.		measure, if enabled, provides the host from which an SNMP trap originated, the time at which the trap was sent, and the details of the trap.

1.2 Alert Log Test

This test monitors multiple alert log files for different patterns.

Target of the test : Any host

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every AlertFile and SearchPattern combination.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the server listens.
AlertFile	Specify the path to the log file to be monitored. For e.g., */user/john/new_john.log*. Multiple log file paths can be provided as a comma-separated list - eg., */user/john/critical_egurkha.log,/tmp/log/major.log*. Also, instead of a specific log file path, the path to the directory containing log files can be provided - eg., */user/logs*. This ensures that eG Enterprise monitors the most recent log files in the specified directory. Specific log file name patterns can also be specified. For example, to monitor the latest log files with names containing the strings 'dblogs' and 'applogs', the parameter specification can be, */tmp/db/*dblogs*,/tmp/app/*applogs**. Here, '*' indicates leading/trailing characters (as the case may be). In this case, the eG agent first enumerates all the log files in the specified path that match the given pattern, and then picks only the latest log file from the result set for monitoring. Your AlertFile specification can also be of the following format: Name@logfilepath_or_pattern. Here, Name represents the display name of the path being configured. Accordingly, the parameter specification for the 'dblogs' and 'applogs' example discussed above can be:

Parameter	Description
	*dblogs@/tmp/db/*dblogs*,applogs@/tmp/app/*applogs**. In this case, the display names 'dblogs' and 'applogs' will alone be displayed as descriptors of this test. **Note:** If your AlertFile specification consists of file patterns that include wildcard characters (eg., */tmp/db/*dblogs*,/tmp/app/*applogs**), then such configurations will only be supported in the ANSI format, and not the UTF format. Every time this test is executed, the eG agent verifies the following: a. Whether any changes have occurred in the size and/or timestamp of the log files that were monitoring during the last measurement period; b. Whether any new log files (that match the alertfile specification) have been newly added since the last measurement period; If a few lines have been added to a log file that was monitored previously, then the eG agent monitors the additions to that log file, and then proceeds to monitor newer log files (if any). If an older log file has been overwritten, then, the eG agent monitors this log file completely, and then proceeds to monitor the newer log files (if any).
SearchPattern	Enter the specific patterns of alerts to be monitored. The pattern should be in the following format: <PatternName>:<Pattern>, where <PatternName> is the pattern name that will be displayed in the monitor interface and <Pattern> is an expression of the form - **expr* or expr or *expr or expr**, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, say you specify ORA:ORA-* in the SearchPattern text box. This indicates that "ORA" is the pattern name to be displayed in the monitor interface. "ORA-*" indicates that the test will monitor only those lines in the alert log which start with the term "ORA-". Similarly, if your pattern specification reads: *offline:*offline*, then it means that the pattern name is offline and that the test will monitor those lines in the alert log which end with the term offline. A single pattern may also be of the form e1+e2, where + signifies an OR condition. That is, the <PatternName> is matched if either e1 is true or e2 is true. Multiple search patterns can be specified as a comma-separated list. For example: *ORA:ORA-*,offline:*offline*,online:*online* If the AlertFile specification is of the format *Name@logfilepath*, then the descriptor for this test in the eG monitor interface will be of the format: *Name:PatternName*. On the other hand, if the AlertFile specification consists only of a comma-separated list of log file paths, then the descriptors will be of the format: *LogFilePath:PatternName*.

Parameter	Description
	If you want all the messages in a log file to be monitored, then your specification would be: *<PatternName>:**.
Lines	Specify two numbers in the format x:y. This means that when a line in the alert file matches a particular pattern, then x lines before the matched line and y lines after the matched line will be reported in the detail diagnosis output (in addition to the matched line). The default value here is 0:0. Multiple entries can be provided as a comma-separated list. If you give 1:1 as the value for Lines, then this value will be applied to all the patterns specified in the SearchPattern field. If you give 0:0,1:1,2:1 as the value for Lines and if the corresponding value in the SearchPattern filed is like *ORA:ORA-*,offline:*offline*,online:*online* then: 0:0 will be applied to ORA:ORA-* pattern 1:1 will be applied to offline:*offline* pattern 2:1 will be applied to online:*online pattern
Exclude Pattern	Provide a comma-separated list of patterns to be excluded from monitoring in the Exclude Pattern text box. For example **critical*, *exception**. By default, this parameter is set to '*none*'. Alternately, you can also specify the path to a specific log file from which patterns are to be excluded. For this, your exclude pattern should be of the following format: *Name@ Pattern name*. For e.g., if you wish to exclude "critical" patterns from */user/john/new_john.log,* then your specification should be */user/john/new_ john.log@critical*. Multiple patterns can also be excluded from different log files by providing them as a comma-separated list - e.g., */user/john/critical_ egurkha.log@critical,/tmp/log/major.log@exception*.
ExcludeFiles	**Note that, this parameter is applicable only when the AlertFile parameter is specified with the path to the directory containing log files**. Provide a comma-separated list of file formats to be excluded from monitoring in the ExcludeFiles text box. By default, this parameter is set to **.gz,*.tar,*.zip* indicating that the files of the mentioned formats will be excluded from monitoring by the test. However, you can add more file formats to the default list as follows: **.gz,*.tar,*.zip, *.cab, *.7z, *.rar*.
UniqueMatch	By default, the UniqueMatch parameter is set to **False**, indicating that, by default, the test checks every line in the log file for the existence of each of the configured SearchPatterns. By setting this parameter to **True**, you can instruct the test to ignore a line and move to the next as soon as a match for one of the configured patterns is found in that line. For example, assume that *Pattern1:*fatal*,Pattern2:*error** is the

Parameter	Description
	SearchPattern that has been configured. If UniqueMatch is set to **False**, then the test will read every line in the log file completely to check for the existence of messages embedding the strings 'fatal' and 'error'. If both the patterns are detected in the same line, then the number of matches will be incremented by 2. On the other hand, if UniqueMatch is set to **True**, then the test will read a line only until a match for one of the configured patterns is found and not both. This means that even if the strings 'fatal' and 'error' follow one another in the same line, the test will consider only the first match and not the next. The match count in this case will therefore be incremented by only 1.
RotatingFile	This flag governs the display of descriptors for this test in the eG monitoring console. If this flag is set to **True** and the AlertFile text box contains the full path to a specific (log/text) file, then, the descriptors of this test will be displayed in the following format: _Directory_containing_monitored_file:<SearchPattern>_. For instance, if the AlertFile parameter is set to _c:\eGurkha\logs\syslog.txt_, and RotatingFile is set to **True**, then, your descriptor will be of the following format: _c:\eGurkha\logs:<SearchPattern>_. On the other hand, if the RotatingFile flag had been set to **False**, then the descriptors will be of the following format: _<FileName>:<SearchPattern>_ - i.e., _syslog.txt:<SearchPattern>_ in the case of the example above. If this flag is set to **True** and the AlertFile parameter is set to the directory containing log files, then, the descriptors of this test will be displayed in the format: _Configured_directory_path:<SearchPattern>_. For instance, if the AlertFile parameter is set to _c:\eGurkha\logs_, and RotatingFile is set to **True**, then, your descriptor will be: _c:\eGurkha\logs:<SearchPattern>_. On the other hand, if the RrotatingFile parameter had been set to **False**, then the descriptors will be of the following format: _Configured_directory:<SearchPattern>_ - i.e., _logs:<SearchPattern>_ in the case of the example above. If this flag is set to true and the AlertFile parameter is set to a specific file pattern, then, the descriptors of this test will be of the following format: _<FilePattern>:<SearchPattern>_. For instance, if the AlertFile parameter is set to _c:\eGurkha\logs\*sys*_, and rotatingfile is set to **True**, then, your descriptor will be: _*sys*:<SearchPattern>_. In this case, the descriptor format will not change even if the RotatingFile flag status is changed .
CaseSensitive	This flag is set to **No** by default. This indicates that the test functions in a 'case-insensitive' manner by default. This implies that, by default, the test ignores the case of your AlertFile and SearchPattern specifications. If this flag is set to **Yes** on the other hand, then the test will function in a 'case-sensitive' manner. In this case therefore, for the test to work, even the case of your AlertFile and SearchPattern specifications should match with the actuals.

Parameter	Description
RotatingFile	By default, this flag is set to **False**. Set this flag to **True** if you want the test to support the 'roll over' capability of the specified AlertFile. A roll over typically occurs when the timestamp of a file changes or when the log file size crosses a pre-determined threshold. When a log file rolls over, the errors/warnings that pre-exist in that file will be automatically copied to a new file, and all errors/warnings that are captured subsequently will be logged in the original/old file. For instance, say, errors and warnings were originally logged to a file named *error_log*. When a roll over occurs, the content of the file *error_log* will be copied to a file named *error_log.1*, and all new errors/warnings will be logged in *error_log*. In such a scenario, since the RolloverFile flag is set to **False** by default, the test by default scans only *error_log.1* for new log entries and ignores *error_log*. On the other hand, if the flag is set to **True**, then the test will scan both *error_log* and *error_log.1* for new entries. If you want this test to support the 'roll over' capability described above, the following conditions need to be fulfilled: • The AlertFile parameter has to be configured only with the name and/or path of one/more alert files. File patterns or directory specifications should not be specified in the AlertFile text box. • The roll over file name should be of the format: "*<alertfile>.1*", and this file must be in the same directory as the AlertFile. By default, this flag is set to **false**. Set this flag to **true** if log files do not 'roll over' in your environment, but get overwritten instead. In such environments typically, new error/warning messages that are captured will be written into the log file that pre-exists and will replace the original contents of that log file; unlike when 'roll over' is enabled, no new log files are created for new entries in this case. If the **OVERWRITTENFILE** flag is set to **true**, then the test will scan the new entries in the log file for matching patterns. However, if the flag is set to **false**, then the test will ignore the new entries. **Note:** If your AlertFile specification consists of file patterns that include wildcard characters (eg.,/tmp/db/*dblogs*,/tmp/app/*applogs*), then such configurations will only be supported in the ANSI format, and not the UTF format.
EncodeFormat	By default, this is set to **none**, indicating that no encoding format applies by default. However, if the test has to use a specific encoding format for reading from the specified AlertFile , then you will have to provide a valid encoding format here - eg.,

Parameter	Description
	UTF-8, UTF-16, etc. Where multiple log files are being monitored, you will have to provide a comma-separated list of encoding formats – one each for every log file monitored. Make sure that your encoding format specification follows the same sequence as your AlertFile specification. In other words, the first encoding format should apply to the first alert file, and so on. For instance, say that your alertfile specification is as follows:*D:\logs\report.log,E:\logs\error.log, C:\logs\warn_log*. Assume that while UTF-8 needs to be used for reading from report.log, UTF-16 is to be used for reading from warn_log . No encoding format need be applied to *error.log*. In this case, your EncodeFormatspecification will be: UTF-8,none,UTF-16.
UseUTF8	If UTF-8 encoding is to be used for reading the specified log file, then, set the UseUTF8 flag to **True**. By default, this flag is set to **False**. If multiple log files are being monitored, then, for each file, you will have to indicate whether UTF-8 encoding is to be used for reading that file or not. For instance, assume that the AlertFile parameter is set to *dblogs@/tmp/db/dblogs.log,applogs@/tmp/app/applogs.log*. Now, to instruct the test to use UTF-8 encoding for reading the 'dblogs' log file and not to use the UTF-8 encoding while reading the 'applogs' log file, your UseUTF8 setting should be as follows: **True,False**. Note that the number of values provided against the UseUTF8 parameter should be equal to the number of log files being monitored. Also, note that if the AlertFile being monitored has BOM, then the test will automatically use UTF-8 encoding to read that file, even if the UseUTF8 flag is set to **False**. **Note:** If your AlertFile specification consists of file patterns that include wildcard characters (eg.,*/tmp/db/*dblogs*,/tmp/app/*applogs**), then the files that match such patterns will only support the ANSI format, and not the UTF format, even if the UTF-8 parameter is set to **true** for such patterns.
UseUTF16	If UTF-16 encoding is to be used for reading the specified log file, then, set the UseUTF16 flag to **true**. By default, this flag is set to **False**. If multiple log files are being monitored, then, for each file, you will have to indicate whether UTF-16 encoding is to be used for reading that file or not. For instance, assume that the AlertFile parameter is set to *dblogs@/tmp/db/dblogs.log,applogs@/tmp/app/applogs.log*. Now, to instruct the test to use UTF-16 encoding for reading the 'dblogs' log file and not to use the UTF-16 encoding while reading the 'applogs' log file, your UseUTF8 setting should be as follows: **true,false**. Note that the number of values provided against the UseUTF16 parameter should be equal to the number of log files being monitored. **Note:** If your AlertFile specification consists of file patterns that include wildcard characters (eg.,*/tmp/db/*dblogs*,/tmp/app/*applogs**), then the files that match such patterns

Parameter	Description
	will only support the ANSI format, and not the UTF format, even if the UTF-16 parameter is set to **true** for such patterns.
Use Sudo	**This parameter is applicable to Unix environments only.** By default, the eG agent does not require any special permissions to parse and read messages from the log file to be monitored. This is why, the Use Sudo parameter is set to **No** by default. In some highly-secure Unix environments however, the eG agentinstall user may not have the permission to read the log file to be monitored. In such environments, you will have to follow the steps below to ensure that the test is able to read the log file and report metrics: • Edit the **SUDOERS** file on the target host and append an entry of the following format to it: <eG_agent_install_user> ALL=(ALL) NOPASSWD: <Log_file_with_path> • For instance, if the eG agent install user is eguser, and the log file to be monitored is */usr/bin/logs/procs.log*, then the entry in the **SUDOERS** file should be: eguser ALL=(ALL) NOPASSWD: /usr/bin/logs/procs.log • Finally, save the file. • Then, when configuring this test using the eG admin interface, set the Use Sudo parameter to **Yes**. Once this is done, then every time the test runs, it will check whether the eG agent install user has the necessary permissions to read the log file. If the user does not have the permissions, then the test runs the **sudo** command to change the permissions of the user, so that the eG agent is able to read from the log file.
Sudo Path	**This parameter is relevant only when the Use Sudo parameter is set to 'Yes'**. By default, the Sudo Path is set to *none*. This implies that the **sudo** command is in its default location - i.e., in the */usr/bin* or */usr/sbin* folder of the target host. In this case, once the Use Sudo flag is set to **Yes**, the eG agent automatically runs the **sudo** command from its default location to allow access to the configured log file. However, if the **sudo** command is available in a different location in your environment, you will have to explicitly specify the full path to the **sudo** command in the Sudo Path text box to enable the eG agent to run the **sudo** command.
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

Parameter	Description
	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 enable/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 abnormalfrequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Recent errors	Indicates the number of errors that were added to the alert log when the test was last executed.	Number	The value of this measure is a clear indicator of the number of “new” alerts that have come into the alert log of the monitored database. The detailed diagnosis of this measure, if enabled, provides the detailed descriptions of the errors of the configured patterns.

1.3 Device CPU Usage

This test provides CPU usage statistics by polling the NetSNMP MIB.

Target of the test : Any host

Agent deploying the test : An external/remote agent

Outputs of the test : One set of results for every router 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.
SNMPPort	The port at which the monitored target exposes its SNMP MIB; The default value is 161.
SNMPVersion	By default, the eG agent supports SNMP version 1. Accordingly, the default selection in the SNMPversion list is **v1**. However, if a different SNMP framework is in use in your environment, say SNMP **v2** or **v3**, then select the corresponding option from this list.
SNMPCommunity	The SNMP community name that the test uses to communicate with the firewall. This parameter is specific to SNMP **v1** and **v2** only. Therefore, if the SNMPVersion chosen is **v3**, then this parameter will not appear.
Username	This parameter appears only when **v3** is selected as the SNMPversion. SNMP version 3 (SNMPv3) is an extensible SNMP Framework which supplements the SNMPv2 Framework, by additionally supporting message security, access control, and remote SNMP configuration capabilities. To extract performance statistics from the MIB using the highly secure SNMP v3 protocol, the eG agent has to be configured with the required access privileges – in other words, the eG agent should connect to the MIB using the credentials of a user with access permissions to be MIB. Therefore, specify the name of such a user against this parameter.
Context	This parameter appears only when v3 is selected as the SNMPVersion. An SNMP context is a collection of management information accessible by an SNMP entity. An item of management information may exist in more than one context and an SNMP entity potentially has access to many contexts. A context is identified by the SNMPEngineID value of the entity hosting the management information (also called a contextEngineID) and a context name that identifies the specific context (also called a contextName). If the Username provided is associated with a context name, then the eG agent will be able to poll the MIB and collect metrics only if it is configured with the context name as well. In such cases therefore, specify the context name of the Username in the Context text box. By default, this parameter is set to *none*.
AuthPass	Specify the password that corresponds to the above-mentioned Username. This parameter once again appears only if the SNMPversion selected is **v3**.
Confirm Password	Confirm the AuthPass by retyping it here.
AuthType	This parameter too appears only if **v3** is selected as the SNMPversion. From the Authtype list box, choose the authentication algorithm using which SNMP v3 converts the specified username and password into a 32-bit format to ensure security of SNMP

Parameter	Description
	transactions. You can choose between the following options: • **MD5** – Message Digest Algorithm • **SHA** – Secure Hash Algorithm
EncryptFlag	This flag appears only when **v3** is selected as the SNMPversion. By default, the eG agent does not encrypt SNMP requests. Accordingly, the this flag is set to **No** by default. To ensure that SNMP requests sent by the eG agent are encrypted, select the **Yes** option.
EncryptType	If this EncryptFlag is set to **Yes**, then you will have to mention the encryption type by selecting an option from the EncryptType list. SNMP v3 supports the following encryption types: • **DES** – Data Encryption Standard • **AES** – Advanced Encryption Standard
EncryptPassword	Specify the encryption password here.
Confirm Password	Confirm the encryption password by retyping it here.
Timeout	Specify the duration (in seconds) within which the SNMP query executed by this test should time out in this text box. The default is 10 seconds.
Data Over TCP	By default, in an IT environment, all data transmission occurs over UDP. Some environments however, may be specifically configured to offload a fraction of the data traffic – for instance, certain types of data traffic or traffic pertaining to specific components – to other protocols like TCP, so as to prevent UDP overloads. In such environments, you can instruct the eG agent to conduct the SNMP data traffic related to the monitored target over TCP (and not UDP). For this, set this flag to **Yes**. By default, this flag is set to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total CPU usage	Indicates the total CPU usage of the server.	Percent	A high value could signify a CPU bottleneck. The CPU utilization may be high because a few processes are consuming a lot of CPU, or because there are too many processes

Measurement	Description	Measurement Unit	Interpretation
			contending for a limited resource. Check the currently running processes to see the exact cause of the problem.
User CPU	Indicates the percentage of CPU that is being used for user processes.	Percent	An unusually high value indicates a problem and may be due to too many user tasks executing simultaneously.
System CPU	Indicates the percentage of CPU that is being used for system processes.	Percent	An unusually high value indicates a problem and may be due to too many system-level tasks executing simultaneously.
Nice CPU	Indicates the percentage of CPU being used by Nice processes (i.e., processes that do not have the default priority).	Percent	
Idle CPU	Indicates the percentage of time that the server is idle.	Percent	

1.4 Directory Test

This test monitors one or more directories on a server.

Target of the test : Any host

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The IP address oh the host for which the test is to be configured.
Port	The port on which the specified host listens.
TargetDirs	Specify a comma-separated list of directory names to be monitored

Parameter	Description
Recursive	This flag indicates if the test must check the target directories recursively or not. If this flag is set to **True**, then all the sub-directories of each target directory are also checked.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total files	Indicates the total number of files in a target directory.	Number	
Total sub directories	Indicates the total number of sub-directories in a target directory.	Number	
Modified files	Indicates the number of files in the target directory that were modified in the last measurement period.	Number	
Directory size	Indicates the total size of all the files in the target directory.	MB	If the value of this measure is found to be alarmingly high, then ensure that unnecessary files occupying large amounts of directory space are immediately identified and removed. This is essential in order to ensure optimum use of the available disk space.

1.5 Directory Updates Test

This test monitors specific directories for files that are older than a configured duration.

Target of the test : Any host

Agent deploying the test : An external/remote agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.

Parameter	Description
Host	The IP address of the host for which this test is to be configured.
Port	The port at which the specified host listens.
Directory List	This text box takes a comma separated list of directory paths that are to be monitored. For example, if you want to monitor a directory called temp in the C drive, then you need to specify, *c:\temp*. If you would like to monitor a directory named root which is a sub-directory of temp, then your specification should be: *c:\temp\root*. To monitor both the temp and root directories in our example, specify the following in the Directory List text box: *c:\temp,c:\temp\root*. Alternatively, your specification can also be of the following format: *DisplayName@DirPath*. For instance, to monitor the *c:\temp* directory, your specification can be: *Temp@c:\temp*. In this case, the DisplayName Temp will appear as the descriptor of the test. You can also monitor multiple directories using the same format. For instance, to monitor the temp and root directories in the C drive, your specification can be: *Temp@c:\temp,Root@c:\root*. In this case, Temp and Root will be the descriptors of the test.
Hours Older	This test reports the number of old files in the configured directories. In the Hours Older text box therefore, you need to specify how old the files in the specified directory have to be, so that they are considered for monitoring by this test. For example, if the Directory List contains *c:\temp*, and the Hours Older text box contains the value 2, then the test will report the number of files in the temp directory that were last modified over (i.e., greater than) 2 hours before. For every directory specification in the Directory List, you can specify a corresponding value in the Hours Older text box - i.e., if 3 directories are configured in the Directory List, then the Hours Older can also contain a comma-separated list of 3 values - say, *2,3,4*. In this case, the test will report the following: • For the first directory in the Directory List, the test will report the number of files in the directory that were last modified over 2 hours ago. • For the second directory in the Directory List, the test will report the number of files in the directory that were last modified over 3 hours before. • For the third directory in the Directory List, the test will report the number of files in the directory that were last modified over 4 hours ago. Alternatively, you can also specify a single value in the Hours Older text box. This value will automatically apply to all the directories configured in the Directory List. In other words, the number of values that you specify in the Hours Older text box should either be 1 or should be equal to the number of directories configured in the Directory List.

Parameter	Description
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 enable/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
Number of old files	Indicates the number of old files in this directory.	Number	In the event that the host runs out of space, you might want to check the value of this measure to figure out if there are too many old files. If so, then you can use the detailed diagnosis of this test to identify the old files, determine whether you still need the files, and if found useless, remove the files so as to make space in the directory.

1.6 Device Disk Usage

This test provides disk usage statistics by polling the NetSNMP MIB.

Target of the test : Any host

Agent deploying the test : An external/remote agent

Outputs of the test : One set of results for every router 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.
SNMPPort	The port at which the monitored target exposes its SNMP MIB; The default value is 161.
SNMPVersion	By default, the eG agent supports SNMP version 1. Accordingly, the default selection in the SNMPversion list is **v1**. However, if a different SNMP framework is in use in your environment, say SNMP **v2** or **v3**, then select the corresponding option from this list.
SNMPCommunity	The SNMP community name that the test uses to communicate with the firewall. This parameter is specific to SNMP **v1** and **v2** only. Therefore, if the SNMPVersion chosen is **v3**, then this parameter will not appear.
UserName	This parameter appears only when **v3** is selected as the SNMPVersion. SNMP version 3 (SNMPv3) is an extensible SNMP Framework which supplements the SNMPv2 Framework, by additionally supporting message security, access control, and remote SNMP configuration capabilities. To extract performance statistics from the MIB using the highly secure SNMP v3 protocol, the eG agent has to be configured with the required access privileges – in other words, the eG agent should connect to the MIB using the credentials of a user with access permissions to be MIB. Therefore, specify the name of such a user against this parameter.
Context	This parameter appears only when v3 is selected as the SNMPVersion. An SNMP context is a collection of management information accessible by an SNMP entity. An item of management information may exist in more than one context and an SNMP entity potentially has access to many contexts. A context is identified by the SNMPEngineID value of the entity hosting the management information (also called a contextEngineID) and a context name that identifies the specific context (also called a contextName). If the Username provided is associated with a context name, then the eG agent will be able to poll the MIB and collect metrics only if it is configured with the context name as well. In such cases therefore, specify the context name of the Username in the Context text box. By default, this parameter is set to *none*.
AuthPass	Specify the password that corresponds to the above-mentioned Username. This parameter once again appears only if the SNMPversion selected is **v3**.
Confirm Password	Confirm the AuthPass by retyping it here.
AuthType	This parameter too appears only if **v3** is selected as the SNMPversion. From the Authtype list box, choose the authentication algorithm using which SNMP v3 converts the specified username and password into a 32-bit format to ensure security of SNMP

Parameter	Description
	transactions. You can choose between the following options: • **MD5** – Message Digest Algorithm • **SHA** – Secure Hash Algorithm
EncryptFlag	This flag appears only when **v3** is selected as the SNMPversion. By default, the eG agent does not encrypt SNMP requests. Accordingly, the this flag is set to **No** by default. To ensure that SNMP requests sent by the eG agent are encrypted, select the **Yes** option.
EncryptType	If this EncryptFlag is set to **Yes**, then you will have to mention the encryption type by selecting an option from the EncryptType list. SNMP v3 supports the following encryption types: • **DES** – Data Encryption Standard • **AES** – Advanced Encryption Standard
EncryptPassword	Specify the encryption password here.
Confirm Password	Confirm the encryption password by retyping it here.
Timeout	Specify the duration (in seconds) within which the SNMP query executed by this test should time out in this text box. The default is 10 seconds.
Data Over TCP	By default, in an IT environment, all data transmission occurs over UDP. Some environments however, may be specifically configured to offload a fraction of the data traffic – for instance, certain types of data traffic or traffic pertaining to specific components – to other protocols like TCP, so as to prevent UDP overloads. In such environments, you can instruct the eG agent to conduct the SNMP data traffic related to the monitored target over TCP (and not UDP). For this, set this flag to **Yes**. By default, this flag is set to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total size	Indicates the total size of each disk/partition.	MB	
Free space	Indicates the available space on the disk.	MB	Ideally, the value of this measure should be high.

Measurement	Description	Measurement Unit	Interpretation
Used space	Indicates the used space on the disk.	MB	
Percent usage	Indicates the percentage of space used on disk.	Percent	A value close to 100% is a cause for concern, as it indicates that the disk is running out of space.
Inodes used	Indicates the percentage of inodes used on disk.	Percent	

1.7 Exception Log Test

This test reports general statistics pertaining to the log files in a host.

Target of the test : Any host

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	Host name/IP address of the server for which the test is to be configured
PortNo	The port on which the specified server listens for HTTP requests
LogFile	The name of the log file to be monitored.
LogDir	The full path to the specified log file.
EmptyFile	Enter either **true** or **false**. The entry **true** instructs the eG Enterprise suite to monitor even empty log files. The entry **false** instructs the eG Enterprise suite to ignore empty log files during monitoring. By default, this text box will hold the value **false**.
HighPattern	In order to track critical exceptions logged in the log file, you need to specify the pattern of such exceptions, here. For eg., if critical exception logs contain the string "Error", then your pattern specification could be *Error*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
LowPattern	To monitor minor exceptions logged in the log file, the pattern of the minor exceptions

Parameter	Description
	has to be specified in this text box. For eg., if minor exception logs contain the string "Low", then the pattern specification could be *Low*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
MediumPattern	For monitoring the medium exceptions in the log file, the pattern of these exceptions needs to be defined in this text box. For eg., if medium exception logs contain the string "Warning", then the pattern specification could be *Warning*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total exceptions	Indicates the total number of exceptions logged in the log file	Number	A high value of this measure indicates the need to analyze the exceptions, ascertain their severity, and take corrective action if required.
High exceptions	Indicates the number of critical exceptions that have been logged in the log file	Number	System performance will suffer much on the occurrence of critical exceptions. Such exceptions will have to be fixed with immediate effect.
Medium exceptions	Indicates the number of not-very-critical exceptions logged in the log file	Number	Medium exceptions might not have an immediate impact on the system performance, but, in the long run, they could grow to be fatal. Such exceptions need not be looked into immediately, but will have to be fixed soon enough.
Low exceptions	Indicates the number of very minor exceptions in the log file	Number	Low exceptions are very negligible in nature and can be ignored.

Note:

If a log file to be monitored is not found or is empty, then the errcount will be 0.

1.8 Error Log Test

This test reports general statistics pertaining to the log files in a host.

Target of the test : Any host

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	Host name/IP address of the server for which the test is to be configured.
PortNo	The port on which the specified server listens for HTTP requests.
LogFile	The name of the log file to be monitored.
LogDir	The full path to the specified log file.
EmptyFile	Enter either **true** or **false**. The entry **true** instructs the eG Enterprise suite to monitor even empty log files. The entry **false** instructs the eG Enterprise suite to ignore empty log files during monitoring. By default, this text box will hold the value **false**.
ErrPattern	In order to track the errors logged in a log file, you need to specify the pattern for the error logs in this text box. For eg., if the error logs contain the string "Error", then your pattern specification could be *Error*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Exceptions	Indicates the total number of errors logged in the log file	Number	A high value of this measure indicates an urgent need to identify the root-cause of the errors and take corrective action.

Note:

If a log file to be monitored is not found or is empty, then the errcount will be 0.

1.9 File Size Test

The FileSize test monitors the file size of each of the files specified as parameters to the test.

Target of the test : Any host

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every file configured.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The IP address oh the host for which the test is to be configured.
Port	The port on which the specified host listens.
Files	Specify a comma separated list of file reference and file path combinations - e.g., *agentlog:c:\eg\agent\logs\agentout.log,managerlog:c:\eg\manager\logs\error_log*.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current size	The current size of the file in Kilobytes	KB	Alerts can be generated when a file exceeds a pre-defined maximum size.

1.10 Large File Test

Some systems in a target environment could be hosting files of large sizes; a few of these files might not be of any use to either the user or the system (eg., *.tmp). In order to locate these files and remove them so as to conserve disk space, the LargeFileTest comes in handy. This test reveals the number of files in a specific directory that are of or above a configured size. If such large-sized files exist, then the detailed diagnosis of this test, when enabled, provides the names of the large files and their respective sizes.

Target of the test : A host system

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Directory configured.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Directories	Specify a comma-separated list of directories to be searched and file sizes, in the following format:*{FULL_PATH_TO_DIR}@{FILE_SIZE}*. For example, to check whether the directory *c:\documents\important* consists of files that are of size 2 MB or above, specify the following in the Directories text box: *c:\documents\important@2*. Similarly, multiple *{DIR}@{FILE_SIZE}* combinations can be provided as a comma-separated list. For example: *c:\documents\important@2*,*c:\letters\business@1*. In case of Unix environments, this will be:*/opt/docs@2,/opt/bin@3*.
Recursive	Set the Recursive flag to **Yes** to ensure that the test searches even the sub-directories within the configured Directories for the files. By setting this flag to **No**, you can instruct the test to search for the files in the parent directory only.
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 enable/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
Largefiles count	Indicates the number of files of or above a configured size in this directory.	Number	The detailed diagnosis of this test, if enabled, provides the names of the large files and their respective sizes.

1.11 Message Log Test

This test reports general statistics pertaining to the log files in a host.

Target of the test : Any host

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	Host name/IP address of the server for which the test is to be configured.
PortNo	The port on which the specified server listens for HTTP requests.
LogFile	The name of the log file to be monitored.
LogDir	The full path to the specified log file.
EmptyFile	Enter either **true** or **false**. The entry **true** instructs the eG Enterprise suite to monitor even empty log files. The entry **false** instructs the eG Enterprise suite to ignore empty log files during monitoring. By default, this text box will hold the value **false**.
HighPattern	In order to track critical exceptions logged in the log file, you need to specify the pattern of such exceptions, here. For eg., if critical exception logs contain the string "Error", then your pattern specification could be *Error*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
LowPattern	To monitor minor exceptions logged in the log file, the pattern of the minor exceptions has to be specified in this text box. For eg., if minor exception logs contain the string "Low", then the pattern specification could be *Low*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.
MediumPattern	For monitoring the medium exceptions in the log file, the pattern of these exceptions needs to be defined in this text box. For eg., if medium exception logs contain the string "Warning", then the pattern specification could be *Warning*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of exceptions	Indicates the total number of exceptions logged in the log file	Number	A high value of this measure indicates the need to analyze the exceptions, ascertain their severity, and take corrective action if required.
High exception count	Indicates the number of critical exceptions that have been logged in the log file	Number	System performance will suffer much on the occurrence of critical exceptions. Such exceptions will have to be fixed with immediate effect.
Medium exception count	Indicates the number of not-very-critical exceptions logged in the log file	Number	Medium exceptions might not have an immediate impact on the system performance, but, in the long run, they could grow to be fatal. Such exceptions need not be looked into immediately, but will have to be fixed soon enough.
Low exception count	Indicates the number of very minor exceptions in the log file	Number	Low exceptions are very negligible in nature and can be ignored.

Note:

If a log file to be monitored is not found or is empty, then the errcount will be 0.

1.12 Device Memory Usage

This test provides memory statistics by polling the NetSNMP MIB.

Target of the test : Any host

Agent deploying the test : An external/remote agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.

Parameter	Description
Host	The IP address of the host for which this test is to be configured.
SNMPPort	The port at which the monitored target exposes its SNMP MIB; The default value is 161.
SNMPVersion	By default, the eG agent supports SNMP version 1. Accordingly, the default selection in the SNMPversion list is **v1**. However, if a different SNMP framework is in use in your environment, say SNMP **v2** or **v3**, then select the corresponding option from this list.
SNMPCommunity	The SNMP community name that the test uses to communicate with the firewall. This parameter is specific to SNMP **v1** and **v2** only. Therefore, if the SNMPVersion chosen is **v3**, then this parameter will not appear.
UserName	This parameter appears only when **v3** is selected as the SNMPVersion. SNMP version 3 (SNMPv3) is an extensible SNMP Framework which supplements the SNMPv2 Framework, by additionally supporting message security, access control, and remote SNMP configuration capabilities. To extract performance statistics from the MIB using the highly secure SNMP v3 protocol, the eG agent has to be configured with the required access privileges – in other words, the eG agent should connect to the MIB using the credentials of a user with access permissions to be MIB. Therefore, specify the name of such a user against this parameter.
Context	This parameter appears only when v3 is selected as the SNMPVersion. An SNMP context is a collection of management information accessible by an SNMP entity. An item of management information may exist in more than one context and an SNMP entity potentially has access to many contexts. A context is identified by the SNMPEngineID value of the entity hosting the management information (also called a contextEngineID) and a context name that identifies the specific context (also called a contextName). If the Username provided is associated with a context name, then the eG agent will be able to poll the MIB and collect metrics only if it is configured with the context name as well. In such cases therefore, specify the context name of the Username in the Context text box. By default, this parameter is set to *none*.
AuthPass	Specify the password that corresponds to the above-mentioned Username. This parameter once again appears only if the SNMPversion selected is **v3**.
Confirm Password	Confirm the AuthPass by retyping it here.
AuthType	This parameter too appears only if **v3** is selected as the SNMPversion. From the Authtype list box, choose the authentication algorithm using which SNMP v3 converts the specified username and password into a 32-bit format to ensure security of SNMP transactions. You can choose between the following options: • **MD5** – Message Digest Algorithm

Parameter	Description
	- **SHA** – Secure Hash Algorithm
EncryptFlag	This flag appears only when **v3** is selected as the SNMPversion. By default, the eG agent does not encrypt SNMP requests. Accordingly, the this flag is set to **No** by default. To ensure that SNMP requests sent by the eG agent are encrypted, select the **Yes** option.
EncryptType	If this EncryptFlag is set to **Yes**, then you will have to mention the encryption type by selecting an option from the EncryptType list. SNMP v3 supports the following encryption types: - **DES** – Data Encryption Standard - **AES** – Advanced Encryption Standard
EncryptPassword	Specify the encryption password here.
Confirm Password	Confirm the encryption password by retyping it here.
Timeout	Specify the duration (in seconds) within which the SNMP query executed by this test should time out in this text box. The default is 10 seconds.
Data Over TCP	By default, in an IT environment, all data transmission occurs over UDP. Some environments however, may be specifically configured to offload a fraction of the data traffic – for instance, certain types of data traffic or traffic pertaining to specific components – to other protocols like TCP, so as to prevent UDP overloads. In such environments, you can instruct the eG agent to conduct the SNMP data traffic related to the monitored target over TCP (and not UDP). For this, set this flag to **Yes**. By default, this flag is set to **No**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total swap	Indicates the total amount of swap space configured for this host.	MB	
Available swap	Indicates the amount of swap space currently unused or available.	MB	
Swap availability	Indicates the percentage of the unused or available	Percent	A very low value indicates that the swap space configured may not be

Measurement	Description	Measurement Unit	Interpretation
	swap memory.		sufficient. A value close to 100% may imply that the swap space configured may be too large.
Real memory	Indicates the total amount of real/physical memory installed on this host.	MB	
Available real memory	Indicates the amount of real/physical memory currently unused or available.	MB	
Free memory	Indicates the total amount of memory free or available for use on this host.	MB	A very low value of free memory is also an indication of high memory utilization on a host.
Shared memory	Indicates the total amount of real or virtual memory currently allocated for use as shared memory.	MB	
Buffer memory	Indicates the total amount of real or virtual memory currently allocated for use as memory buffers.	MB	
Cached memory	Indicates the total amount of real or virtual memory currently allocated for use as cached memory.	MB	

1.13 Network Traps Test

This test reports the count of SNMP trap messages sent on account of errors in the transactions between the network devices.

Target of the test : Any host

Agent deploying the test : An internal agent

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

Configurable parameters for the test

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Test Period</td><td>How often should the test be executed.</td></tr>
<tr><td>Host</td><td>The host for which the test is to be configured.</td></tr>
<tr><td>Port</td><td>The port at which the application listens.</td></tr>
<tr><td>SourceAddress</td><td>Specify a comma-separated list of IP addresses or address patterns of the hosts sending the traps. For example, 10.0.0.1,192.168.10.*. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.</td></tr>
<tr><td>OIDValue</td><td>Provide a comma-separated list of OID and value pairs returned by the traps. The values are to be expressed in the form, DisplayName:OID-OIDValue. For example, assume that the following OIDs are to be considered by this test: .1.3.6.1.4.1.9156.1.1.2 and .1.3.6.1.4.1.9156.1.1.3. The values of these OIDs are as given hereunder:
<table>
<tr><th>OID</th><th>Value</th></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.2</td><td>Host_system</td></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.3</td><td>NETWORK</td></tr>
</table>
In this case the oidvalue parameter can be configured as Trap1:.1.3.6.1.4.1.9156.1.1.2-Host_system,Trap2:.1.3.6.1.4.1.9156.1.1.3-Network, where Trap1 and Trap2 are the display names that appear as descriptors of this test in the monitor interface.
The test considers a configured OID for monitoring only when the actual value of the OID matches with its configured value. For instance, in the example above, if the value of OID .1.3.6.1.4.1.9156.1.1.2 is found to be Host and not Host_system, then the test ignores OID .1.3.6.1.4.1.9156.1.1.2 while monitoring.
An * can be used in the OID/value patterns to denote any number of leading or trailing characters (as the case may be). For example, to monitor all the OIDs that return values which begin with the letter 'F', set this parameter to Failed:*-F*.</td></tr>
<tr><td>ShowOID</td><td>Selecting the True option against ShowOID will ensure that the detailed diagnosis of this test shows the OID strings along with their corresponding values. If you select False, then the values alone will appear in the detailed diagnosis page, and not the OIDs.</td></tr>
<tr><td>Detailed Diagnosis</td><td>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</td></tr>
</table>

Parameter	Description
	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 enable/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
SNMP traps received	Indicates the number of trap messages sent since the last measurement period.	Number	The detailed diagnosis of this measure, if enabled, provides the host from which an SNMP trap originated, the time at which the trap was sent, and the details of the trap.

1.14 Old Files Test

This test tracks the age of the files within a specified directory on the system.

Target of the test : Any host

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The IP address oh the host for which the test is to be configured.
Port	The port on which the specified host listens.
TargetDirs	Specify a comma-separated list of directory names to be monitored.

Parameter	Description
Recursive	This flag indicates if the test must check the target directories recursively or not. If this flag is set to **True**, then all the sub-directories of each target directory are also checked.
MaxAge	This test will report the number of files that are older than the duration (in minutes) specified in the MaxAge text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total files	The total number of files in the directory being monitored.	Number	
Total old files	The total number of old files - i.e. the files for which last modified time was smaller than the current time.	Number	

1.15 Process Activity Test

The ProcessActivity test reports statistics related to the number and size of processes executing on a system. This test works on Solaris, Linux, HPUX, and AIX platforms only.

Target of the test : Solaris, Linux, AIX, and HPUX systems

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the every process pattern configured .

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the application listens.
Process	Enter a comma separated list of processNames:processPattern pairs which identify the process(es) executing on the server under consideration. processName is a string

Parameter	Description
	that will be used for display purposes only. processPattern is an expression of the form - *expr* or expr or *expr or expr* or *expr1*expr2*... or expr1*expr2, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, the Process parameter can contain the following value: Java:*java*. Here, Java is the pattern name that will be displayed in the eG monitor interface as the info (descriptor) of the ProcActivityTest. The Java pattern in our example will monitor those processes, the names of which embed the string 'java'.
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 enable/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
Current processes	Indicates the number of processes currently running.	Number	
Processes added	Indicates the number of processes added during the last measurement period.	Number	
Processes removed	Indicates the number of processes that were abnormally terminated/completed during the last measurement period.	Number	
Virtual size	Indicates the total size of the process in virtual memory.	MB	

Measurement	Description	Measurement Unit	Interpretation
Resident size	Indicates the resident size of the process. This denotes the size taken up by the process in the RAM, i.e., real address space.	MB	Virtual size is always greater than or equal to the resident size of the process. This measure will not be available for AIX platforms.

1.16 Process Details Test

This test is used to monitor the memory leaks (if any) in any Windows application or process. This test is particularly useful in development and staging environments, where memory leaks with applications can be detected early and recoding done to overcome the leaks.

Target of the test : Any host

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
PortNo	The port on which the specified host listens.
ProcessName	The name of the Windows application / process to be monitored. Multiple applications can be specified as a comma-separated list.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current handles	Indicates the total number of file handles that are currently owned by each thread in the process.	Number	If there is a consistent increase in the value of this measure over time, then it is a clear indicator of a memory leak in the process.
Private memory	Indicates the resources	KB	If there is a consistent increase in the

Measurement	Description	Measurement Unit	Interpretation
	(handles, physical RAM, the paging file, system resources, etc.) that the process has allocated that cannot be shared with other processes.		value of this measure over time, then it is a clear indicator of a memory leak in the process.
Pool paged memory usage	Indicates the memory in the paged pool. A paged pool is an area of system memory for objects that can be written to the disk, but which must remain in the physical memory.	KB	If there is a consistent increase in the value of this measure over time, then it is a clear indicator of a memory leak in the process.
Pool non-paged memory usage	Indicates the memory in the non-paged pool. A non-paged pool is an area of system memory for objects that cannot be written to the disk, but which must remain in the physical memory as long as they are allocated.	KB	If there is a consistent increase in the value of this measure over time, then it is a clear indicator of a memory leak in the process.

1.17 Process Pools Test

This test reports a variety of CPU and memory statistics pertaining to every process in a process tree, starting from the root-process to its leaves (i.e. it reports measures related to both parent and child processes).

Target of the test : Any host

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.
Process	Enter a comma separated list of *names:pattern* pairs which identify the process(es) associated with the server being considered. *processName* is a string that will be used for display purposes only. *processPattern* is an expression of the form - **expr** or *expr* or **expr or expr** or **expr1*expr2*...* or *expr1*expr2*, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. For example, for an iPlanet application server (Nas_server), there are three processes named kcs, kjs, and kxs associated with the application server. For this server type, in the Process text box, enter "*kcsProcess:*kcs*, kjsProcess:*kjs*, kxsProcess:*kxs**, where * denotes zero or more characters. Other special characters such as slashes (\\) can also be used while defining the process pattern. For example, if a server's root directory is */home/egurkha/apache* and the server executable named httpd exists in the bin directory, then, the process pattern is "**/home/egurkha/apache/bin/httpd**". To determine the process pattern to use for your application, on Windows environments, look for the process name(s) in the **Task Manager -> Processes** selection. To determine the process pattern to use on Unix environments, use the ps command (e.g., the command "ps -e -o pid,args" can be used to determine the processes running on the target system; from this, choose the processes of interest to you).
PIDFile	Enter a comma separated list of process *names:paths* to pid files that contain the process ids of the processes that need to be monitored. processName is a string that will be used for display purposes only. For example, this text box could contain, *WebServer:/tmp/pid_file1, Apache:/tmp/pid_file2*, where pid_file1 and pid_file2 are the files containing the process ids. Note that each pid file can contain only one pid.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Processes running	Number of instances of a process(es) currently executing on a host	Number	This value indicates if too many or too few processes corresponding to an application are executing on the host.
CPU usage	Percentage of CPU used by executing process(es) corresponding to the	Percent	A very high value could indicate that processes corresponding to the specified pattern are consuming

Measurement	Description	Measurement Unit	Interpretation
	pattern specified		excessive CPU resources.
Memory usage	For one or more processes corresponding to a specified set of patterns, this value represents the ratio of the resident set size of the processes to the physical memory of the host system, expressed as a percentage.	Percent	A sudden increase in memory utilization for a process(es) may be indicative of memory leaks in the application.

Note:

If a log file to be monitored is not found or is empty, then the errcount will be 0.

1.18 SQL Response Test

The responsiveness of a database to SQL queries is not only indicative of the health of the database server, but also the efficiency of the queries. A well-tuned database is one that quickly responds to SQL queries, and a well-built SQL query is one that succeeds in retrieving the desired results from the database and that too, in record time. The SQLResponseTest monitors SQL queries from start to finish, and reports the status of the query execution and its responsiveness. This way, administrators are proactively notified of failed queries and queries that take too long to execute, so that root-cause diagnosis is instantly initialized.

Target of the test : A database server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the application listens.

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>JDBC Driver</td><td>Specify the JDBC driver that is used to access the database. The table below lists the JDBC drivers that correspond to some of the most popular database servers that are monitored by eG Enterprise. Refer to this table whenever in need.
<table>
<tr><th>Database</th><th>Driver</th></tr>
<tr><td>Oracle</td><td>oracle.jdbc.driver.OracleDriver</td></tr>
<tr><td>MS SQL</td><td>net.sourceforge.jtds.jdbc.Driver</td></tr>
<tr><td>Informix</td><td>com.informix.jdbc.IfxDriver</td></tr>
<tr><td>Sybase</td><td>com.sybase.jdbc2.jdbc.SybDriver</td></tr>
<tr><td>MySql</td><td>org.gjt.mm.mysql.Driver</td></tr>
</table>
</td></tr>
<tr><td>Connection URL</td><td>Specify the JDBC URL for the database. The URL format is JDBC driver specific. The table below lists the JDBC URLs for some of the most popular database servers that are monitored by eG Enterprise. While configuring this test for any of the database servers in this table, you can specify a URL of the corresponding format.
<table>
<tr><th>Database</th><th>Driver</th></tr>
<tr><td>Database</td><td>URL Format</td></tr>
<tr><td>Oracle</td><td>jdbc:oracle:thin:@{host}:{port}:{instance}</td></tr>
<tr><td>MS SQL</td><td>jdbc:jtds:sqlserver://{host}:{port}/{database}</td></tr>
<tr><td>Informix</td><td>jdbc:informix-sqli://{host}:{port}/{database}:informixserver={instance}</td></tr>
<tr><td>Sybase</td><td>jdbc:sybase:Tds:{host}:{port}/{database}</td></tr>
</table>
If the target database is not in the above list, then follow the steps given below:
<ul>
<li>Download the JDBC driver of the new database from the database vendor.</li>
<li>Copy the relevant java package files (jar or zip) into the {EG_AGENT_INSTALL_DIR}\lib directory (on Windows; on Unix, this will be the opt/egurkha/lib directory).</li>
<li>If a Unix agent is executing this test, then simply proceed to restart the eG agent. In case of a Windows agent however, edit the <b>debugoff.bat</b> file in the {EG_AGENT_INSTALL_DIR}\lib directory to manually set the Classpath value. Then, execute debugoff.bat so that the agent service is reinstalled on Windows with the new classpath settings.</li>
<li>Next, login to the eG administrative interface and configure this test with the JDBC Driver and Connection URL that corresponds to the new database.</li>
</ul>
</td></tr>
</table>

Parameter	Description
User	The name of the User who is vested with the privilege to execute the configured query.
Password	The password of the specified User.
Confirm Password	Confirm the password by retyping it in the Confirm Password text box.
Query	Specify the query to be executed and monitored.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Query status	Indicates whether the configured query has been successfully executed.	Boolean	The value of 1 indicates successful execution, and 0 indicates failure. In case of query failure, you can use the detailed diagnosis of this measure, if enabled, to view the errors that caused the query to fail; troubleshooting thus becomes easier.
Query time	Indicates the time taken to execute the query and retrieve results.	Secs	An abnormally high value is a cause for concern, and warrants further investigation.

1.19 SSL Certificate Test

All SSL web servers are configured with security certificates. During the SSL protocol handshake with clients, a server exchanges this certificate with the clients. An SSL certificate includes information about the server/domain to which the certificate is licensed, the issuing authority, and a validity period for the certificate. Beyond the validity period, the SSL certificate becomes invalid, and clients' SSL connections to the web server would fail. To avoid such a situation, it is essential that web server administrators are alerted in advance about the potential expiry of the SSL certificates on their web site. The **SSL Certificate** test monitors the validity period for SSL certificates of different web sites.

Target of the test : A Web server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every Target configured.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the application listens.
Timeout	Provide the duration (in seconds) beyond which the test times out.
Targets	Provide a comma-separated list of *{HostIP/Name}:{Port)* pairs, which represent the web sites to be monitored. For example, *192.168.10.7:443,192.168.10.8:443*. The test connects to each IP/port pair and checks for validity of the certificate associated with this target. One set of metrics is reported for each target. The descriptor represents the common name (CN) value of the SSL certificate

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
SSL certificate validity	Represents the validity of the SSL certificate in days.	Days	As this value approaches close to 0, an alert is generated to proactively inform the administrator that the SSL certificate is nearing expiry. A value of 0 indicates that the SSL certificate has expired.

1.20 Stratus Hardware Traps Test

This test monitors the status of various hardware elements present in the Stratus server using SNMP traps.

Target of the test : An SNMP trap

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every OID value monitored.

Configurable parameters for the test

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Test Period</td><td>How often should the test be executed.</td></tr>
<tr><td>Host</td><td>The host for which the test is to be configured.</td></tr>
<tr><td>Port</td><td>The port at which the application listens.</td></tr>
<tr><td>SourceAddress</td><td>Specify a comma-separated list of IP addresses or address patterns of the hosts sending the traps. For example, *10.0.0.1,192.168.10.**. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters.</td></tr>
<tr><td>OIDValue</td><td>Provide a comma-separated list of OID and value pairs returned by the traps. The values are to be expressed in the form, *DisplayName:OID-OIDValue*. For example, assume that the following OIDs are to be considered by this test: *.1.3.6.1.4.1.9156.1.1.2 and .1.3.6.1.4.1.9156.1.1.3*. The values of these OIDs are as given hereunder:
<table>
<tr><th>OID</th><th>Value</th></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.2</td><td>Host_system</td></tr>
<tr><td>.1.3.6.1.4.1.9156.1.1.3</td><td>NETWORK</td></tr>
</table>

In this case the OIDvalue parameter can be configured as Trap1:*.1.3.6.1.4.1.9156.1.1.2-Host_system*,Trap2:*.1.3.6.1.4.1.9156.1.1.3-Network*, where Trap1 and Trap2 are the display names that appear as descriptors of this test in the monitor interface.
The test considers a configured OID for monitoring only when the actual value of the OID matches with its configured value. For instance, in the example above, if the value of OID *.1.3.6.1.4.1.9156.1.1.2* is found to be Host and not Host_system, then the test ignores OID *.1.3.6.1.4.1.9156.1.1.2* while monitoring.
An * can be used in the OID/value patterns to denote any number of leading or trailing characters (as the case may be). For example, to monitor all the OIDs that return values which begin with the letter 'F', set this parameter to *Failed:*-F**.</td></tr>
<tr><td>ShowOID</td><td>Selecting the **True** option against ShowOID will ensure that the detailed diagnosis of this test shows the OID strings along with their corresponding values. If you select **False**, then the values alone will appear in the detailed diagnosis page, and not the OIDs.</td></tr>
<tr><td>Detailed Diagnosis</td><td>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</td></tr>
</table>

Parameter	Description
	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 enable/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
Empty	Indicates that a slot in the system is in an "empty" state.	Boolean	For a slot, this state indicates that the slot is empty, physically not present, or electrically inaccessible. If the empty device causes the system to be go into simplex mode, the device is no longer fault tolerant. In some cases this state represents both a slot and a device. For instance, an instance of an SRA_DIMM in the Empty state means that a slot exists for the DIMM, but that the slot is empty. DIMMs, CPU Boards, IO Boards and Processors are represented by such WMI objects. Sensors go to this state instead of the"Not Present" state when they are not present. Empty devices are generally enumerable.
Not present	Indicates that a device in the system is in a "not present" state.	Boolean	This state indicates that a device is either physically not present or electrically inaccessible. For instance, pulling the power cord on a CPU board makes the DIMMs and Processors on the board go to this state. When a WMI object goes to this state, it is generally not enumerable. Thus, this

Measurement	Description	Measurement Unit	Interpretation
			state only appears in state change events.
Removed	Indicates that a device in the system is in a "removed" state. Usually, this is a final state but it can be a transient state.	Boolean	Usually, this state indicates that a device was intentionally removed from service. When intentionally removed from service, the device remains in this state. Only some devices go to this state when removed from services; other devices go to other offline states. Some devices pass through this state as they are brought online.
Dumping	Indicates that a device is in a "Dumping" state. This is a transient state.	Boolean	This state indicates a device is in the process of writing a dump to a file.
Diagnostics passed	Indicates that a device is in a "Diagnostic Passed" state. This is a transient state and the device should change to "online" state when it is brought online.	Boolean	This state indicates that a device has just completed its diagnostics tests.
Initialising	Indicates that a device is in a "Initialising" state. This is a transient state and the device should change to "online" state when it is brought online.	Boolean	This state indicates that a device is in the process of initializing.
Syncing	Indicates that a device is in a "synching" state. This is a transient state and the device should change to "online" state when it is brought online.	Boolean	This state indicates that a device is synchronizing itself with its partners. For instance, when a CPU is brought up, it synchronizes its memory and its processor state with that of its partners.
Offline	Indicates that a device is in a "offline" state.	Boolean	This state indicates that a device is offline. Only some devices can go to this state while other devices go into

Measurement	Description	Measurement Unit	Interpretation
			the "Removed From Service" state.
Firmware update complete	Indicates that a device's firmware update procedure has completed.	Boolean	
Diagnostics	Indicates that a device is running diagnostics.	Boolean	
Online	Indicates that a device is in a "online" state.	Boolean	This state indicates that the device is online, but not configured for redundancy. For instance, a working NIC that is not part of a team will be in this state. Although the online state does not indicate whether a device is safe-to-pull or not, on a properly configured system such devices can be assumed safe-to-pull.
Simplex	Indicates that a device is in a "Simplex" state.	Boolean	This state indicates that a device is online, configured for redundancy, and is not safe-to-pull. When applied to a port, indicates that the port is configured for redundancy, and that whatever is connected to the port is not safe-to-pull.
Duplex	Indicates that a device is in a "Duplex" state.	Boolean	This state indicates that a device is online, configured for redundancy, and is safe-to-pull. When applied to a port, indicates that the port is configured for redundancy, and that whatever is connected to the port is safe-to-pull.
Shot	Indicates that a device is in a "Shot" state. This is a transient state and the device should either transit to "broken" or "online" state after diagnostic is done.	Boolean	This state indicates that a device experienced a problem and will soon move to either an online state or the broken state.
Broken	Indicates that a device is in a "Broken" state.	Boolean	This state Indicates that a device is

Measurement	Description	Measurement Unit	Interpretation
			broken. In the case of a port, this state may mean that the port is inoperative or that that which attaches to the port is inoperative. There are several reasons that a device could be broken but usually points to hardware errors. Contact your service providers for service checks. In the case where the device is a port, it usually indicates that there is nothing attached to the port, or when whatever should be attached to the port is not responding. For example, a NIC port will be in this state when it cannot detect link.

1.21 TCP Connection Test

This test reports various statistics pertaining to TCP connections to and from a host, from an external perspective.

Target of the test : Any host

Agent deploying the test : An external agent

Outputs of the test : One set of results for every configured port name.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	Host name of the server for which the test is to be configured.
PortNo	Enter the port to which the specified host listens.
TargetPorts	Specify either a comma-separated list of port numbers that are to be tested (eg., 80,7077,1521), or a comma-separated list of *port name:port number* pairs that are to be tested (eg., smtp:25,mssql:1433). In the latter case, the port name will be displayed in the monitor interface. Alternatively, this parameter can take a comma-separated list of *port name:IP address:port number* pairs that are to be tested, so as to enable the

Parameter	Description
	test to try and connect to Tcp ports on multiple IP addresses. For example, *mysql:192.168.0.102:1433,egwebsite:209.15.165.127:80*.
IsPassive	If the value chosen is **Yes**, then the server under consideration is a passive server in a cluster. No alerts will be generated if the server is not running. Measures will be reported as “Not applicable’ by the agent if the server is not up.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Availability	Whether the TCP connection is available	Percent	An availability problem can be caused by different factors – e.g., the server process may not be up, a network problem may exist, or there could be a configuration problem with the DNS server.
Response time	Time taken (in seconds) by the server to respond to a request.	Secs	An increase in response time can be caused by several factors such as a server bottleneck, a configuration problem with the DNS server, a network problem, etc.

1.22 WebLogic Log Patterns Test

This test monitors an application log and reports measures such as the total number of responses that have been logged and average response time of every log file entry pattern that has been configured.

Target of the test : A WebLogic server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.

Parameter	Description
Host	The host for which the test is to be configured.
Port	The port at which the application listens.
AbsoluteFileName	Specify the full path to the log file to be monitored.
RecordPattern	The records in the log file that need to be considered for monitoring will have to be provided in the RecordPattern text box. The pattern configuration should be in the following format: *{f0}sep1{f1}sep2{f2}*, where {f0}, {f1}, and {f2} represent the indexes of the first, second, and third fields (respectively) of the records logged in the log file, and sep1 and sep2 are the separators after {f0} and {f1} respectively. A separator can be a combination of any number of characters. For example, take the case of a log file with the following entry: *eg_sample_appln_jsp :;TIME:2005-01-01 00:06:26.904;Thread_ID:ExecuteThread: '48' for queue: default';Duration:233* To ensure that the above record is considered for monitoring, the record pattern will have to be specified as follows: *{f0};{f1};{f2};{f3}:{f4},* where {f0} represents the first field of the record, which is followed by the separator ';', and so on.
SearchPattern	Of the records that match the configured RecordPattern, the eG agent will search for and monitor only those records which match the string patterns specified in the SearchPattern text box. To help you understand how to configure a SearchPattern, let us take the example of the following search pattern: Info1:any,f0:*eg_sample_appln_jsp *,count(*),avg(f4). • Here, *Info1* is just a display name that will be displayed in the eG monitor interface as a descriptor of this test. • Use the term **ALL** or **Any** to instruct the eG Enterprise system to consider only those records that fulfill the condition that follows, for monitoring. The condition is: f0:*eg_sample_appln_jsp*. This indicates that for a record to be considered for monitoring, the first field (i.e. the field with index 0) of the record should embed the string *eg_sample_appln_jsp.* • COUNT(*) returns the number of records that fulfill the configured criteria. • *AVG(f5)* returns the average of the values of all the fields with index 5 (i.e. the sixth field), in the records that match the configured criteria. According to this specification, the eG Enterprise system, while taking a count and while calculating the average, will consider only those records where the first field

Parameter	Description
	embeds the string *eg_sample_appln_jsp*. Similarly, multiple search patterns can be provided separated by "#&".

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Calls	Indicates the number of account calls that are being made during a period of time.	Number	A high value of this measure indicates a heavy workload on the server.
Avg response time	Indicates the average response time for account calls.	Secs	A dramatic increase in this value may be indicative of poor responsiveness of the server.
Note: If any of the measures of this test returns the value -5, then such a measure will not be displayed in the monitor interface. On the other hand, if all the measures of this test return the value -5, then all the measures will appear in the monitor interface, but the value displayed for each measure will be "Not Available".			

1.23 WebLogic Log Requests Test

This test monitors a web server access log and reports measures such as the number of requests that have been logged, the number of successful responses, the number of failed responses, etc., for every pattern that has been configured.

Target of the test : A WebLogic server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every search pattern being configured.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	The host for which the test is to be configured.
Port	The port at which the application listens.

Parameter	**Description**
AbsoluteFileName	Specify the full path to the log file to be monitored.
RecordPattern	The records in the log file that need to be considered for monitoring will have to be provided in the RecordPattern text box. The pattern configuration should be in the following format: *{f0}sep1{f1}sep2{f2}*, where {f0}, {f1}, and {f2} represent the indexes of the first, second, and third fields (respectively) of the records logged in the log file, and sep1 and sep2 are the separators after {f0} and {f1} respectively. A separator can be a combination of any number of characters. For example, take the case of a log file with the following entry: *192.168.10.7 - - [12/Nov/1998:09:40:40 -0500] "POST /soap/servlet/helloworld HTTP/1.1" 200 3834* To ensure that the above record is considered for monitoring, the record pattern will have to be specified as follows: *{f0}- -{f1}"{f2}"{f3} {f4}*, where {f0} represents the first field of the record, which is followed by the separator '- -', and so on.
SearchPattern	Of the records that match the configured RecordPattern, the eG agent will search for and monitor only those records which match the string patterns specified in the SearchPattern text box. To help you understand how to configure a SearchPattern, let us take the example of the following search pattern: _IP1:ALL,F0:192.168.10.7*,F3: 200*,COUNT(*),AVG(F4)_. • *Here,* IP1 *is just a display name that will be displayed in the eG monitor interface as a descriptor of this test.* • The term ALL instructs the eG Enterprise system to consider only those records that fulfill all the conditions that follow. Alternatively, the key word Any can be used, which implies that the eG Enterprise system, while monitoring, will consider even those records that fulfill either of the conditions that follow. The conditions are: • F0: 192.168.10.7* indicates that for a record to be considered for monitoring, the first field (i.e. the field with index 0) of the record should begin with the IP 192.168.10.1. Alternatively, the condition can be configured as _F0:192.168.10.7*+192.168.10.8*+192.168.10.9*_, where '+' denotes an 'OR' operator. This configuration indicates that for a record to be considered for monitoring, the first field of the record should begin with any of the three values configured - i.e. 192.168.10.7, 192.168.10.8, or 192.168.10.9. • F3: 200* indicates that for a record to be considered for monitoring, the fourth field (i.e. the field with index 3) of the record should begin with the number 200.

Parameter	Description
	Alternatively, the condition can be configured as F3:200*+300*+400*, where '+' denotes an 'OR' operator. This configuration indicates that for a record to be considered for monitoring, the fourth field of the record should begin with any of the three values configured - i.e. 200, 300, or 400. • COUNT(*) returns the number of records that fulfill the configured criteria. • AVG(F4) returns the average of the values of all the fields with index 4 (i.e. the fifth field), in the records that match the configured criteria. According to this specification, the eG Enterprise system, while taking a count and while calculating the average, will consider only those records where the first field starts with '192.168.10.1' and the fourth field starts with '200'. The number ‘200’ indicates a successful response. Therefore, this specification will report the metrics pertaining to only the successful responses for the IP patterns defined within the descriptor IP1 (i.e. 192.168.10.7*). However, the test's configuration becomes complete only if the failure statistics are also extracted for IP1. Therefore, you will have to provide another search pattern for the descriptor IP1, so that the failure information is collected. The format of this pattern should be: IP1_FAIL: ALL,f0:192.168.10.7*,!f3:200*,COUNT(*),AVG(f4). Note that the descriptor names are the same, but the one meant for monitoring the failure cases, has been tagged as _FAIL. The specification !f3:200 indicates that the records with the number ‘200’ (in the fourth field) should NOT be considered for monitoring. ‘!’ is a NOT operator. Since ‘200’ represents a success state, !200 ensures that only the failed responses for IP1 are considered for monitoring. The complete SearchPattern will hence be: IP1:ALL,f0:192.168.10.7*,f3:200*,COUNT(*),AVG(f4)#& IP1_ FAIL:ALL,f0:192.168.10.7*,!f3:200*,COUNT(*),AVG(f4), where #& is the separator. In the monitor interface however, the descriptor IP1 alone will appear, but when clicked, will display both the success and failure statistics for the pattern 192.168.10.7*. Therefore, it is imperative that the WLLogReqTest be configured in such a way that it tracks both the success and failure cases for every IP pattern configured for monitoring. Otherwise, the test will not function as desired. This implies that if an IP pattern IP2 is configured for monitoring successful responses, then an IP2_FAIL should follow to monitor the failed responses. Similarly, multiple patterns can be configured for monitoring, separated by ‘#&’.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total requests	Indicates the number of account calls that are being made during a period of time.	Number	A high value of this measure indicates a heavy workload on the server.
Successes	Indicates the number of successful responses.	Number	Low value of this measure indicates less number of successful responses from the server.
Avg success bytes	Indicates the number of bytes of successful responses	Bytes	A high value of this measure indicates a high rate of successful responses.
Failures	Indicates the number of failed responses.	Number	
Avg fail bytes	Indicates the number of bytes of failed responses.	Bytes	A high value of this measure indicates a high failure rate.
Avg bytes sent	Indicates the size (in bytes) of responses sent by the server.	Bytes	
Note: If any of the measures of this test returns the value -5, then such a measure will not be displayed in the monitor interface. On the other hand, if all the measures of this test return the value -5, then all the measures will appear in the monitor interface, but the value displayed for each measure will be "Not Available".			

1.24 Windows Interrupts Test

This test reports how busy the system processor was while handling hardware device interrupts.

Target of the test : A Windows host

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the host 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	The port at which the specified host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Interrupt time	Indicates the percentage of time spent by the processor for receiving and servicing the hardware interrupts during the last polling interval.	Percent	This is an indirect indicator of the activity of devices that generate interrupts such as system Clocks, the mouse device drivers, data communication lines, network interface cards and other peripheral devices. In general, a very high value of this measure might indicate that a disk or network adapter needs upgrading or replacing.

1.25 WebLogic Log Responses Test

This test monitors an application log and reports measures such as the total number of responses that have been logged and average response time of every log file entry pattern that has been configured.

Target of the test : A WebLogic server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every search pattern being configured.

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.

Parameter	Description
Host	The host for which the test is to be configured.
Port	The port at which the server listens.
AbsoluteFileName	Specify the full path to the log file to be monitored.
RecordPattern	The records in the log file that need to be considered for monitoring will have to be provided in the RecordPattern text box. The pattern configuration should be in the following format: *{f0}sep1{f1}sep2{f2}*, where {f0}, {f1}, and {f2} represent the indexes of the first, second, and third fields (respectively) of the records logged in the log file, and sep1 and sep2 are the separators after {f0} and {f1} respectively. A separator can be a combination of any number of characters. For example, take the case of a log file with the following entries: *2486:Sampleappln:LoginUser->Time Taken for:LOGIN_CHECK; is:155* *2530:Sampleappln:LoginUser->Time Taken for:AVAIL_CHECK; is:252* To ensure that the above records are considered for monitoring, the record pattern will have to be specified as follows: *{f0}:{f1}:{f2}->{f3}:{f4}:{f5},* where {f0} represents the first field of the record, which is followed by the separator ':', and so on.
SearchPattern	Of the records that match the configured RecordPattern, the eG agent will search for and monitor only those records which match the string patterns specified in the SearchPattern text box. To help you understand how to configure a SearchPattern, let us take the example of the following search pattern: Info1:ANY,f4:!LOGIN_CHECK*,COUNT(*),AVG(f5). • Here, *Info1* is just a display name that will be displayed in the eG monitor interface as a descriptor of this test. • Use the term **ALL** or **Any** to instruct the eG Enterprise system to consider only those records that fulfill the condition that follows, for monitoring. The condition is: f4:!LOGIN_CHECK*. This indicates that for a record to be considered for monitoring, the fifth field (i.e. the field with index 4) of the record should 'not' begin with the string *LOGIN_CHECK*. The '!' symbol is the 'not' operator. • COUNT(*) returns the number of records that fulfill the configured criteria. • AVG(f5) returns the average of the values of all the fields with index 5 (i.e. the sixth field), in the records that match the configured criteria. According to this specification, the eG Enterprise system, while taking a count and while calculating the average, will consider only those records where the fifth field does

Parameter	Description
	not begin with 'LOGIN_CHECK'. Similarly, multiple search patterns can be provided separated by "#&". For example, *Info1:ANY,f4:!LOGIN_CHECK*,COUNT(*),AVG(f5)#&Info2:ALL,f4:AVAIL_CHECK*,COUNT(*),AVG(f5).*

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Calls	Indicates the number of account calls that are being made during a period of time.	Number	A high value of this measure indicates a heavy workload on the server.
Avg response time	Indicates the average response time for account calls.	Secs	A dramatic increase in this value may be indicative of poor responsiveness of the server.
Note: If any of the measures of this test returns the value -5, then such a measure will not be displayed in the monitor interface. On the other hand, if all the measures of this test return the value -5, then all the measures will appear in the monitor interface, but the value displayed for each measure will be "Not Available".			

1.26 Windows Memory Stats Test

This test reports details about the physical memory of the system.

Target of the test : A Windows host

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the host 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	The port at which the specified host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Committed memory in use	Indicates the committed bytes as a percentage of the Commit Limit.	Percent	In the event that the host runs out of space, you might want to check the value of this measure to figure out if there are too many old files. If so, then you can use the detailed diagnosis of this test to identify the old files, determine whether you still need the files, and if found useless, remove the files so as to make space in the directory.
Pool nonpaged failures	Indicates the number of times allocations have failed from non paged pool.	Number	Generally, a non-zero value indicates a shortage of physical memory.
Pool paged failures	Indicates the number of times allocations have failed from paged pool.	Number	A non-zero value indicates a shortage of physical memory.
Copy read hits	Indicates the percentage of copy read calls satisfied by reads from the Cache out of all read calls.	Percent	Any value over 80% is excellent.

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.