Monitoring Microsoft Exchange 2013/2016

eG Innovations Product Documentation

www.eginnovations.com

Total Performance Visibility

Table of Contents

Table of Figures

Chapter 1: Introduction

Microsoft Exchange Server 2013/2016 brings a new rich set of technologies, features, and services to the Exchange Server product line. Its goal is to support people and organizations as their work habits evolve from a communication focus to a collaboration focus. At the same time, Exchange Server 2013/2016 helps lower the total cost of ownership whether you deploy Exchange 2013/2016 on-premises or provision your mailboxes in the cloud.

For Exchange Server 2013/2016, the new architecture consolidates the number of server roles from four to two: the Client Access Server (CAS) role and the Mailbox Server (MS) role. The Client Access server is a thin, stateless server that serves as a proxy for client connections to the Mailbox server. The Mailbox server handles the processing for all client connections to the active mailbox database.

One of the key functions of these server roles is to support the smooth flow of mails through the transport pipeline. The transport pipeline is a collection of services, connections, components, and queues that work together to route all messages to the categorizer in the Transport service on a Mailbox server inside the organization.

The transport pipeline consists of the following services:

- *Front End Transport service on Client Access servers* This service acts as a stateless proxy for all inbound and (optionally) outbound external SMTP traffic for the Exchange 2013/2016 organization. The Front End Transport service doesn't inspect message content, doesn't communicate with the Mailbox Transport service on Mailbox servers, and doesn't queue any messages locally.
- *Transport service on Mailbox servers* This service is virtually identical to the Hub Transport server role in previous versions of Exchange. The Transport service handles all SMTP mail flow for the organization, performs message categorization, and performs message content inspection. Unlike previous versions of Exchange, the Transport service never communicates directly with mailbox databases. That task is now handled by the Mailbox Transport service. The Transport service routes messages between the Mailbox Transport service, the Transport service, the Front End Transport service, and (depending on your configuration) the Transport service on Edge Transport servers.
- *Mailbox Transport service on Mailbox servers* This service consists of two separate services: the Mailbox Transport Submission service and Mailbox Transport Delivery service. The Mailbox Transport Delivery service receives SMTP messages from the Transport service on the local

Mailbox server or on other Mailbox servers, and connects to the local mailbox database using an Exchange remote procedure call (RPC) to deliver the message. The Mailbox Transport Submission service connects to the local mailbox database using RPC to retrieve messages, and submits the messages over SMTP to the Transport service on the local Mailbox server, or on other Mailbox servers. The Mailbox Transport Submission service has access to the same routing topology information as the Transport service. Like the Front End Transport service, the Mailbox Transport service also doesn't queue any messages locally.

- *Transport service on Edge Transport servers* This service is very similar to the Transport service on Mailbox servers. If you have an Edge Transport server installed in the perimeter network, all mail coming from the Internet or going to the Internet flows through the Transport service Edge Transport server. This service is described in more detail later in this topic.

The following figure shows the relationships among the components in the Exchange 2013/2016 transport pipeline.

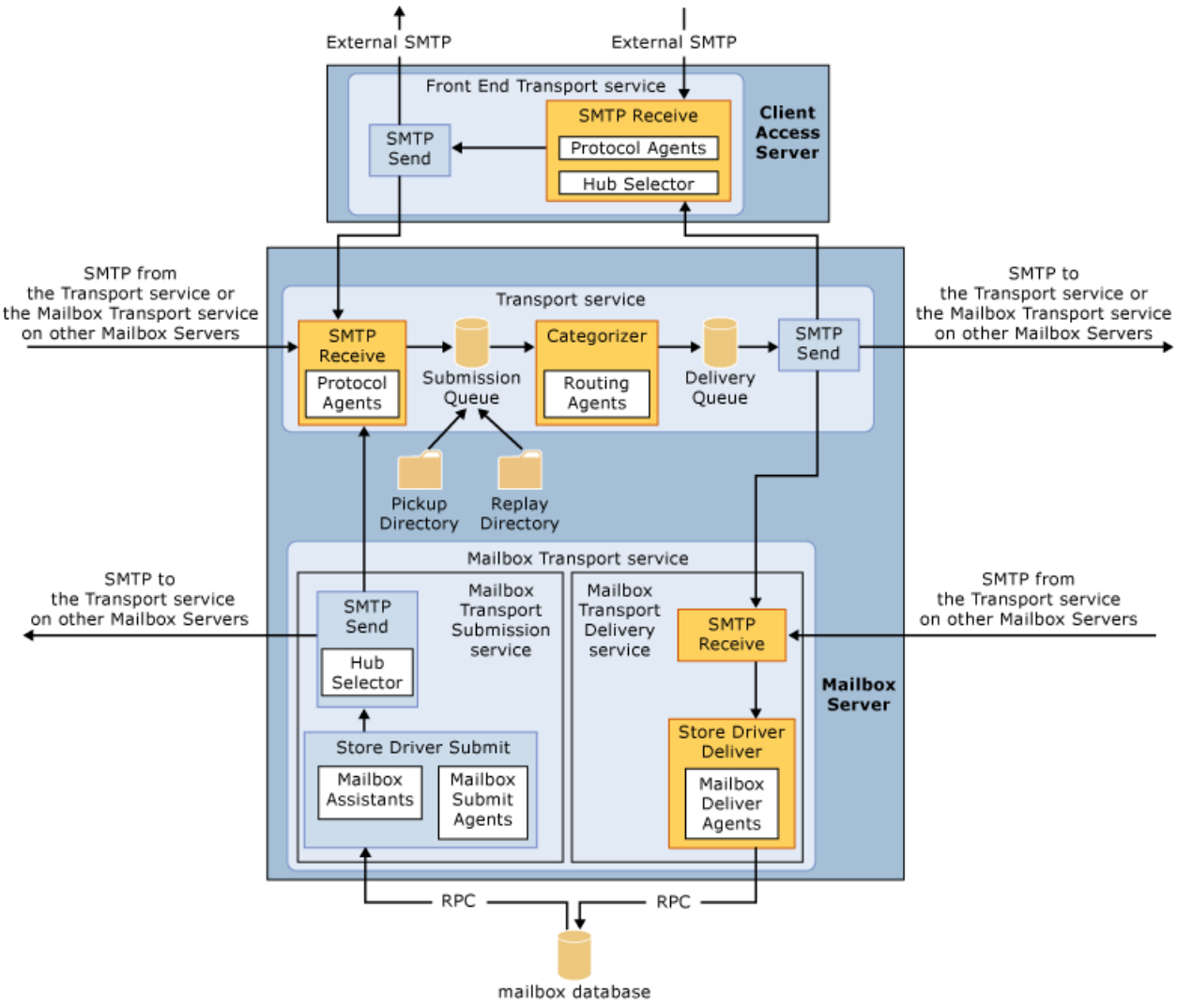

Figure 1.1: Overview of transport pipeline

Errors/delays anywhere in the transport pipeline – be it with the CAS, with the Mailbox server, with the transport agents, with the Send/Receive connectors, with the anti-malware/spam engines, with the transport queues, with Exchange search or indexing – can adversely impact mail flow and ultimately, the timely delivery of mails to the designated destination. In an era where time is money, delays in email delivery often translates into loss of revenue, reputation, and the escalation of support costs.

To avoid this, administrators should continuously monitor every aspect of the mail flow and capture even the smallest of deviations from the norm.

Chapter 2: Administering the eG Manager to monitor the Microsoft Exchange 2013/2016

1. Log into the eG administrative interface.
2. eG Enterprise cannot automatically discover Microsoft Exchange 2013/2016 server. You need to manually add the server using the **COMPONENTS** page (see Figure 2.1) that appears when the Infrastructure -> Components -> Add/Modify menu sequence is followed. Remember that components manually added are managed automatically.

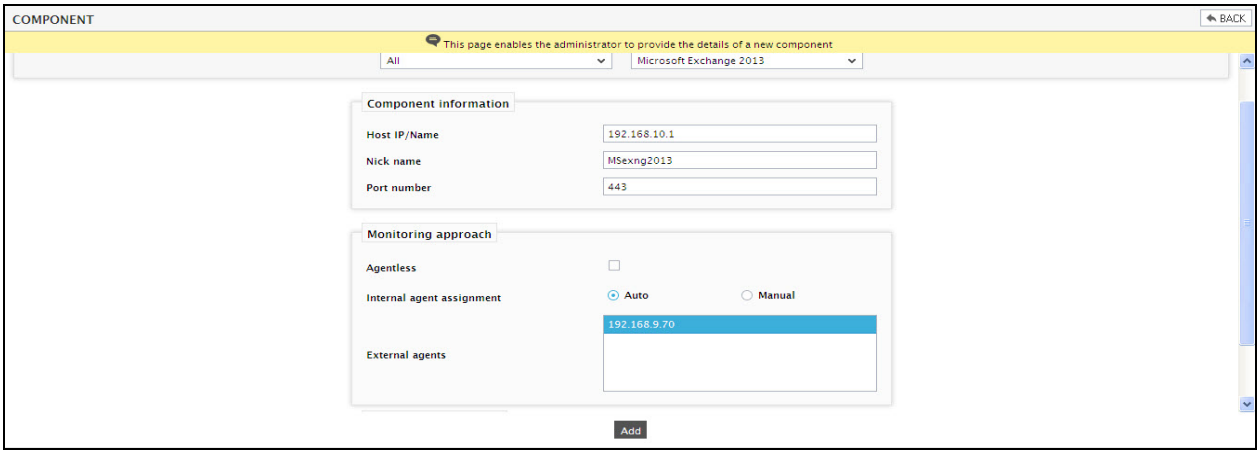

Figure 2.1: Adding a Microsoft Exchange 2013/2016 server

3. Specify the Host IP and the Nick name of the Microsoft Exchange 2013/2016 server in Figure 2.1. Then click the Add button to register the changes.
4. When you attempt to sign out, a list of unconfigured tests will appear as shown in Figure 2.2.

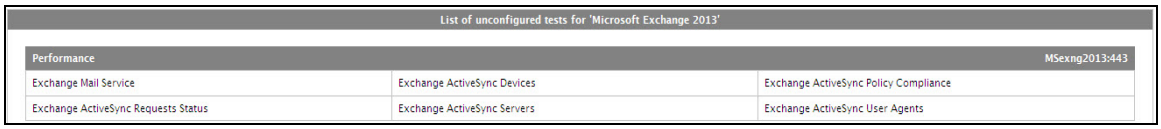

Figure 2.2: List of Unconfigured tests to be configured for the Microsoft Exchange 2013/2016 server

5. Click on the **Exchange ActiveSync Devices** test to configure it. To know how to configure this test, click here.
6. Once all the tests are configured, signout of the eG administrative interface.

Chapter 3: Monitoring Microsoft Exchange 2013/2016

eG Enterprise provides an integrated Microsoft Exchange 2013/2016 model, which monitors each of the core components of the transport pipeline and those that are in the periphery (eg., anti-malware/spam engine, Exchange Search and Content indexing engine, etc.), and proactively alerts administrators to current/potential bottlenecks to mail delivery.

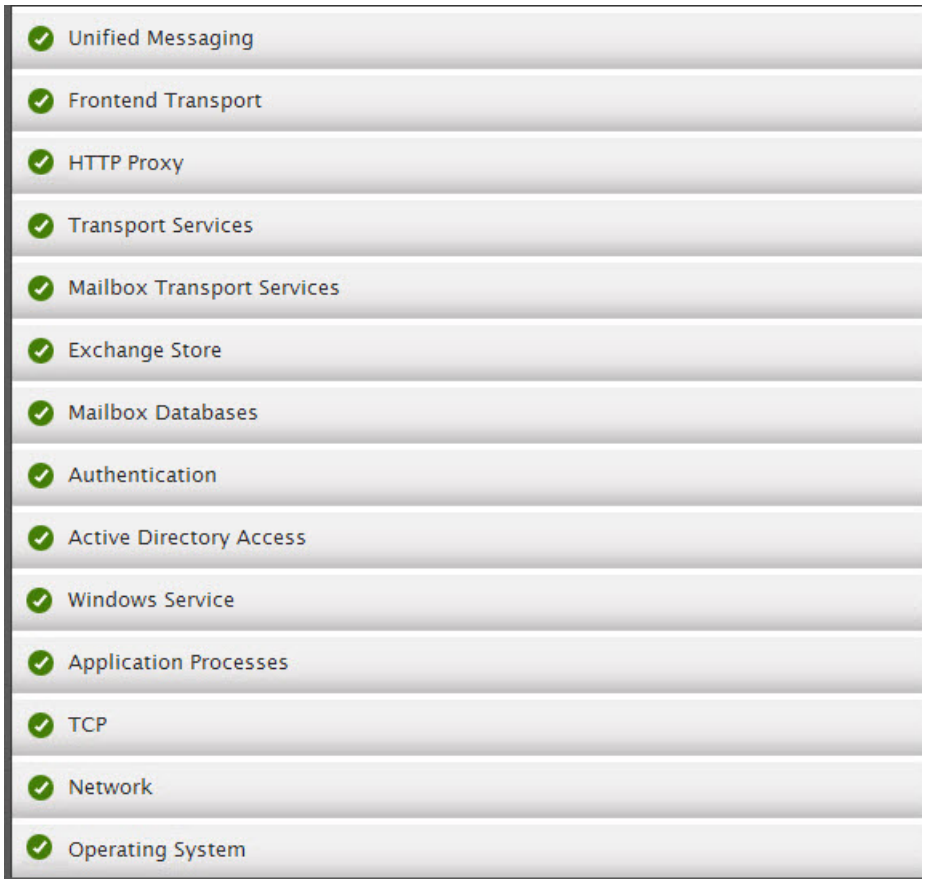

Figure 3.1: Layer model of Microsoft Exchange 2013/2016 server

Each layer of Figure 3.1 is mapped to a series of tests. These tests use Windows perfmon or Exchange management shell commands to pull out a wealth of performance information related to the Exchange 2013/2016 server. Using these performance statistics, Exchange administrators can find quick and accurate answers to the following performance queries:

Since the 5 layers at the bottom of the layer model depicted by Figure 3.1 have already been discussed in the Monitoring Unix and Windows Servers document, the sections that follow will discuss all layers above the **Windows Service** layer only.

3.1 The Managed Availability Layer

Note:

This layer will appear only when the Managed Availability process logs information, warning, or error events in the event log.

Using the **Managed Availability Monitoring Log** test mapped to this layer, you can promptly capture errors that the Manage Availability engine of Exchange cannot self-heal.

3.1.1 Managed Availability Monitoring Log

Managed availability, also known as Active Monitoring or Local Active Monitoring, is the integration of built-in monitoring and recovery actions with the Exchange high availability platform. It's designed to detect and recover from problems as soon as they occur and are discovered by the system. Unlike previous external monitoring solutions and techniques for Exchange, managed availability doesn't try to identify or communicate the root cause of an issue. It's instead focused on recovery aspects that address three key areas of the user experience:

- *Availability* Can users access the service?
- *Latency* How is the experience for users?
- *Errors* Are users able to accomplish what they want?

Managed availability is an internal process that runs on every Exchange 2013/2016 server. It polls and analyzes hundreds of health metrics every second. If something is found to be wrong, most of the time it will be fixed automatically. But there will always be issues that managed availability won't be able to fix on its own. In those cases, managed availability will escalate the issue to an administrator by means of event logging. Using the **Managed Availability Monitoring Log** test, administrators can scan the event logs for information, warning, or error messages logged by the Managed Availability process, and capture critical errors/warnings that Exchange cannot self-heal using the Managed Availability engine.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.

3. **PORT** – The port at which the host listens.
4. **LOG TYPE** – By default, Microsoft-Exchange-ManagedAvailability/Monitoringwill be set as the **LOG TYPE**.
5. **DD FREQUENCY**- Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Alert value:	Indicates the type of event that was captured and logged by the managed availability process in the event log during the last measurement period.		The values that this measure can take and their corresponding numeric values are as follows: Measure Value / Numeric Value Information – 0 Warning – 1 Error – 2 **Note:** By default, this measure reports one of the **Measure Value**s listed in the table above. In the graph of this measure however, the alert status is indicated by the corresponding numeric equivalents only.
Total alerts:	Indicates the total number of errors that the Managed Availability process logged in the event logs.	Number	A high value is a cause for concern.

3.2 The Active Directory Access Layer

Many client processes/services on the Exchange server interact with Active Directory to obtain useful recipient and configuration iformation. The tests mapped to this layer monitor these interactions to bring problems to light.

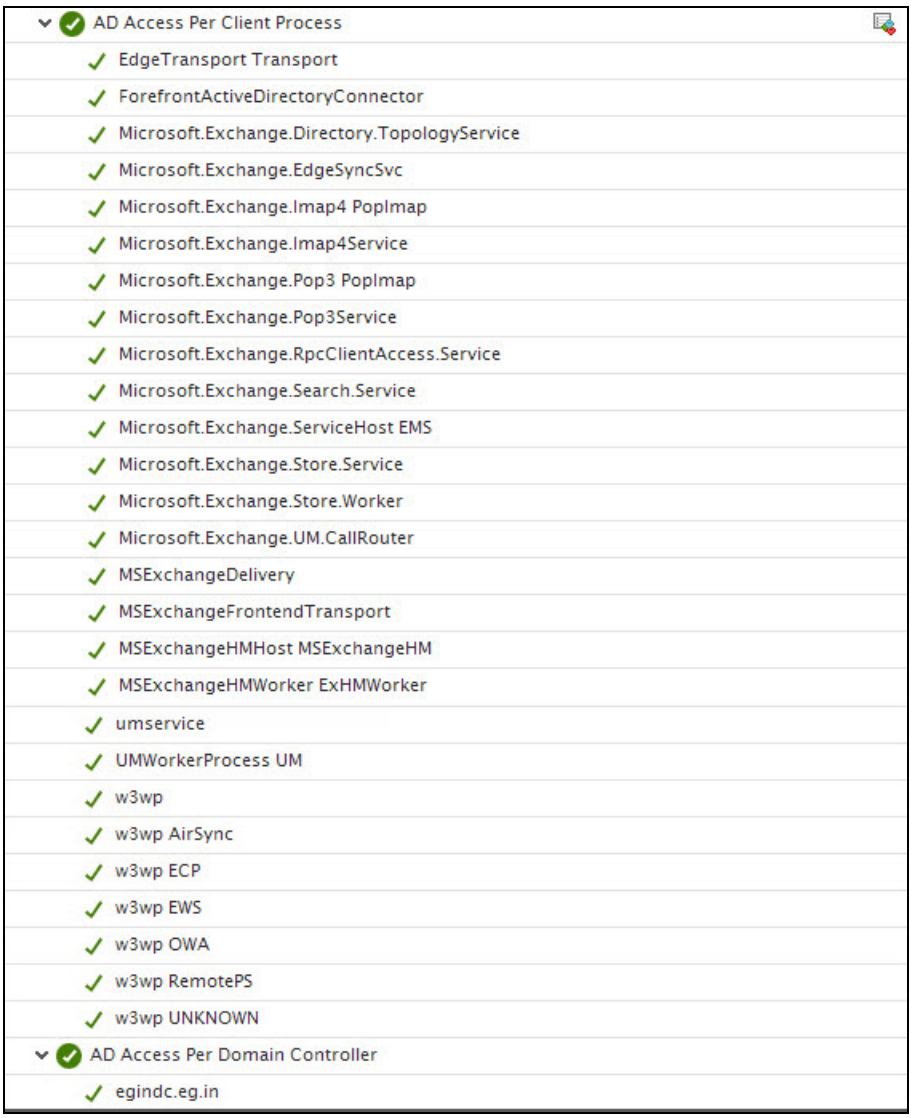

Figure 3.2: The tests mapped to the Active Directory Access layer

3.2.1 AD Access Per Client Process Test

Many client processes/services on the Exchange server interact with Active Directory to obtain useful recipient and configuration information. For example, you have the **Microsoft.Exchange.EdgeSyncSvc** process that keeps the recipient and configuration information up to date when an Edge Server is subscribed to the same AD site as the Mailbox

server. You also have the **ADTopologyService** that locates Active Directory domain controllers and global catalog servers and provides Active Directory topology information to other Exchange Server services.

When the communication between any of these processes/services and AD slow down, user experience with Exchange will certainly be impacted adversely. If a user then complains that Exchange is slow, administrators will have to instantly figure out which process's interactions with AD are abnormal and where is the bottleneck – in running LDAP search queries? In processing LDAP read requests? In processing LDAP write requests? The **AD Access Per Client Process** test provides administrators with these insights. This test auto- discovers the critical client processes/services running on the Exchange server and reports how quickly every process services the LDAP search/read/write requests it receives. In the process, the test accurately pinpoints where the bottleneck is.

Target of the test : An Exchange 2013/2016 Server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each client process/service on the Exchange server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.
4. **DD FREQUENCY** - The **DD FREQUENCY** refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
5. **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
LDAP read calls rate:	Indicates the rate at which Depth 0 LDAP search calls are made by this process.	Calls/Sec	Depth 0 calls refer to search queries that search only the base DN. Compare the value of this measure across processes to identify which process is imposing the maximum read request load on the AD server. Use the detailed diagnosis of this measure to know which instances of a given process are currently communicating with the AD server, and the health of the interactions of each instance.
LDAP read time:	Indicates the time taken by this process to execute LDAP read requests and return a response.	Secs	The average value for this measure should be less than 0.05 seconds. Spikes (Maximum) should not be higher than 0.1 seconds. Compare the value of this measure across processes to identify which process is processing read requests slowly. Once the process is identified, compare the LDAP read time of that process with the value of the LDAP search time and LDAP write time measures to know

Measurement	Description	Measurement Unit	Interpretation
			where exactly the slowdown occurred – when processing search requests, read requests, or write requests?
LDAP search calls rate:	Indicates the rate at which Depth 1 and 2 LDAP search calls were made by this process.	Calls/Sec	Depth 1 and 2 calls refer to search queries that search 1 and 2 levels below the base DN. Compare the value of this measure across processes to identify which process is imposing the maximum search request load on the AD server.
LDAP search time:	Indicates the time taken by this process to send run an LDAP search query on AD and receive a response.	Secs	The average value for this measure should be less than 0.05 seconds. Spikes (Maximum) should not be higher than 0.1 seconds. Compare the value of this measure across domain controllers to identify which process's queries are being processed very slowly by AD. Once the process is identified, compare the LDAP search time of that process with the value of the LDAP read time and LDAP write time measures to know where exactly the slowdown occurred – when processing search requests, read requests, or write requests?
LDAP timeout errors rate:	Indicates the number of LDAP operations made	Errors/Sec	

Measurement	Description	Measurement Unit	Interpretation
	per second by this process because of an exceeded timeout.		
LDAP write calls rate:	Indicates the rate at which this process made Add/Modify/Delete calls to AD.	Calls/Sec	Compare the value of this measure across processes to know which process made the maximum number of add/modify/delete calls to AD and contributed the most to its workload.
LDAP write time:	Indicates the time taken by this process to send an add/modify/delete request to AD and receive a response.	Secs	A consistent increase in this value could indicate a bottleneck when *adding/modifying/deleting* objects in AD. Compare the value of this measure across processes to identify which process's write requests are being processed very slowly by AD. Once the process is identified, compare the LDAP write time of that process with the value of the LDAP search time and LDAP read time measures to know where exactly the slowdown occurred – when processing search requests, read requests, or write requests?
Long running LDAP operations:	Indicates the number of LDAP operations made by this process per minute that took longer than the specified threshold (default	Operations/Minute	*MaxQueryDuration* is an LDAP administration limit that represents the maximum time a domain controller should spend on a single search. When this limit is reached, the domain controller

Measurement	Description	Measurement Unit	Interpretation
	threshold is 15 seconds).		returns a "timeLimitExceeded" error. By comparing the value of this measure across processes, you can identify the process that is responsible for triggering the maximum number of long-running queries.
Outstanding requests:	Indicates the current number of pending LDAP searches for this process.	Number	Compare the value of this measure across processes to identify that process with the maximum number of pending search requests. The reason for this anomaly should be investigated and the source of the processing bottleneck should be cleared for optimal performance of Exchange 2013/2016.

Use the detailed diagnosis of the *LDAP read calls rate* measure to know which instances of a given process are currently communicating with the AD server, and the health of the interactions of each instance. From this, you can identify that instance which is taking too long to perform LDAP operations on the AD server, resulting in a slowdown.

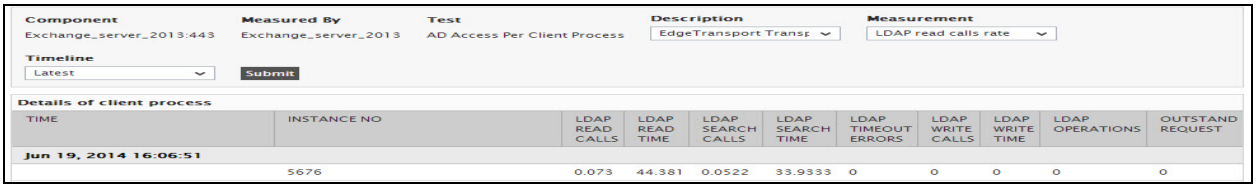

Figure 3.3: The detailed diagnosis of the LDAP read calls rate measure

3.2.2 AD Access Per Domain Controller Test

Active Directory servers must be available for Exchange 2013/2016 to function correctly. Microsoft Exchange Server 2013/2016 stores all configuration and recipient information in the Active Directory directory service database. When a computer running Exchange 2013/2016 requires information

about recipients and information about the configuration of the Exchange organization, it runs LDAP queries on the Active Directory to access the information. For this purpose, as soon as Exchange 2013/2016 starts, it binds randomly with a domain controller and global catalog server in its own site. If any of these domain controllers process the LDAP queries slowly, Exchange 2013/2016 server roles may not have the recipient/configuration information they require on time; this in turn may significantly slowdown mission-critical operations of the Exchange server such as authentication, mail delivery, etc.

If this is to be avoided, administrators should keep a close watch on the LDAP queries to each domain controller in the same site as Exchange 2013/2016, measure the time taken by each controller to process the queries, and swoop down on domain controllers that are experiencing serious processing bottlenecks. For this purpose, administrators can use the **AD Access Per Domain Controller** test. This test auto-discovers the domain controllers used by Exchange 2013/2016, and reports how quickly every controller processes the LDAP requests it receives. In the process, the test accurately pinpoints the slow controllers.

Target of the test : An Exchange 2013/2016 Server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each domain controller

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
LDAP read calls rate:	Indicates the rate at which Depth 0 LDAP search calls are made to this domain controller.	Calls/Sec	Depth 0 calls refer to search queries that search only the base DN. Compare the value of this measure across domain controllers to identify which domain controller is the busiest in terms of the frequency of such calls.

Measurement	Description	Measurement Unit	Interpretation
LDAP read time:	Indicates the time taken by this domain controller to execute LDAP read requests and return a response.	Secs	The average value for this measure should be less than 50 milliseconds. Spikes (Maximum) should not be higher than 100 milliseconds. Compare the value of this measure across domain controllers to identify which controller is processing read requests slowly.
LDAP search calls rate:	Indicates the rate at which Depth 1 and 2 LDAP search calls were made to this domain controller.	Calls/Sec	Depth 1 and 2 calls refer to search queries that search 1 and 2 levels below the base DN. Compare the value of this measure across domain controllers to identify which domain controller is the busiest in terms of the frequency of such calls.
LDAP search time:	Indicates the time taken by this domain controller to process LDAP search queries and return a response.	Secs	The average value for this measure should be less than 50 milliseconds. Spikes (Maximum) should not be higher than 100 milliseconds. Compare the value of this measure across domain controllers to identify which controller is processing search queries slowly.
Time limit exceeded LDAP searches:	Indicates the number of LDAP searches to this domain controller that executed beyond a configured duration in the last minute.	Number	**MaxQueryDuration** is an LDAP administration limit that represents the maximum time a domain controller should spend on a single search. When this limit is reached, the domain controller returns a "*timeLimitExceeded*" error.

Measurement	Description	Measurement Unit	Interpretation
			By comparing the value of this measure across domain controllers, you can identify that controller with the maximum number of long-running queries. Such controllers could be experiencing serious processing bottlenecks.
Timed out LDAP searches:	Indicates the number of LDAP searches to this domain controller that timed out in the last minute.	Number	Ideally, the value of this measure should be low. A high value indicates that too many LDAP searches timed out.
Outstanding requests:	Indicates the current number of pending LDAP operations to this domain controller.	Number	Compare the value of this measure across domain controllers to identify that controller with the highest number of pending operation. The reason for this anomaly should be investigated and the source of the processing bottleneck should be cleared for optimal performance of Exchange 2013/2016.

3.3 The Authentication Layer

With the help of the **Exchange Authentication** test mapped to this layer, administrators can instantly detect bottlenecks in authentication.

Figure 3.4: The tests mapped to the Authentication layer

3.3.1 Exchange Authentication Test

Authentication is the process by which a client and a server verify their identities for transmitting data. In Microsoft Exchange, authentication is used to determine whether a user or client that wants to communicate with the Exchange server is who or what it says it is.

When you install Exchange server and the Client Access server role, virtual directories are configured for several services. These include Outlook Web App, the Availability service, Unified Messaging, and Microsoft Exchange ActiveSync. By default, each virtual directory is configured to use an authentication method. The failure of an authentication method naturally results in denial of access to the corresponding service. This is why, when a user/client attempting to connect to Exchange to avail of a particular service complains of a delay or a denial of access, administrators should immediately check the authentication process to look for failures/latencies. The **Exchange Authentication** test helps with this. For every authentication method that is configured, this test tracks the authentication requests and captures latencies and rejections. This way, the test helps administrators proactively identify probable bottlenecks in authentication and the methods affected.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every authentication method configured

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current authentication request latency:	Indicates the latency of current authentication requests to this authentication method.	Secs	A low value is desired for this measure. Compare the value of this measure across authentication methods to know which method is the most latent.
Outstanding	Indicates the number of	Number	A consistent increase in the value

Measurement	Description	Measurement Unit	Interpretation
authentication requests:	authentication requests to this method that are pending processing.		of this measure is indicative of an authentication bottleneck. Compare the value of this measure across authentication methods to know which method is taking too long to process requests.
Rejected authentication requests:	Indicates the number of rejected authentication requests to this method.	Number	The value 0 is desired for this measure. Further investigation is required if a non- zero value is reported by this measure.

3.4 The Mailbox Databases Layer

The tests mapped to this layer focus on the health of mailbox databases and mailboxes, and promptly alerts administrators to:

- Abnormal growth in mailbox size and probable quota violations
- Unmounted mailbox databases
- Indexing bottlenecks
- Latencies in database replication
- Problems in Exchange Search
- Factors impeding smooth mail flow

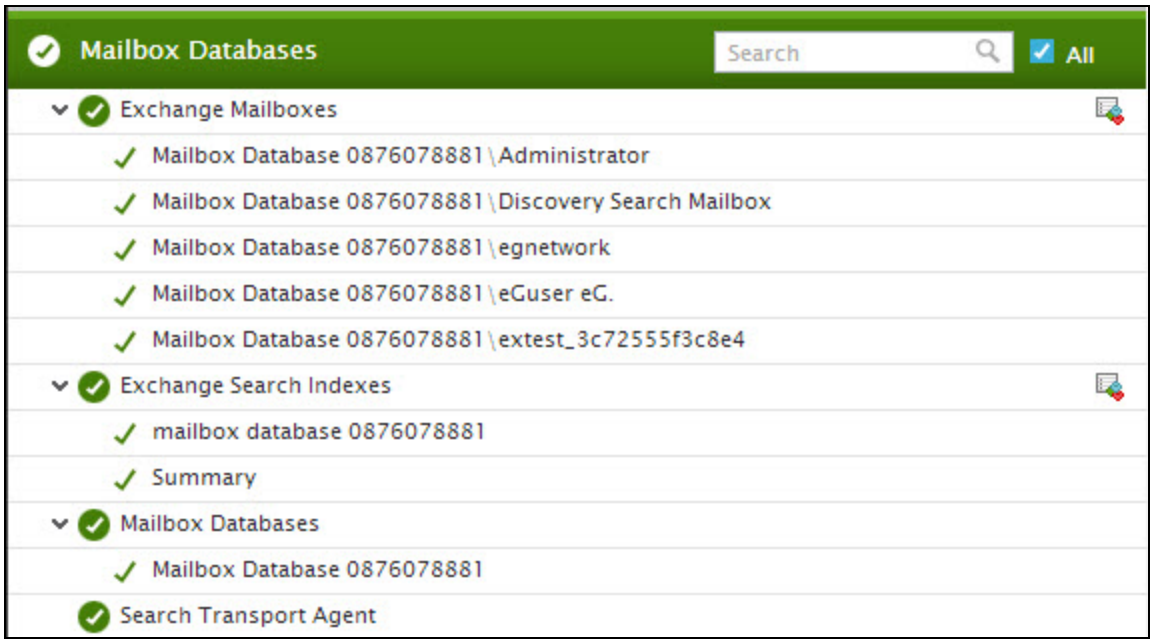

Figure 3.5: The tests mapped to the Mailbox Databases layer

3.4.1 Exchange Mailboxes Test

The Exchange mailbox is one of the core components of an Exchange infrastructure. The non-availability of the mailbox or improper quota setting for the mailbox can often result in important emails been returned as 'undelivered' or can prohibit users from sending out business-critical mails This in turn can negatively impact user productivity and diminish user confidence in the messaging system. To avert this, administrators should periodically check the availability and the size of each mailbox, promptly detect sudden/prolonged breaks in mailbox availability or the abnormal growth in mailbox size, and resolve the discovered problems before users are affected. This is where the **Exchange Mailboxes t**est helps. This test auto-discovers all the mailboxes on the Exchange server. For each mailbox, the test reports whether/not that mailbox is accessible, and if so, reveals the number and size of items in that mailbox, the quota configuration of that mailbox, and how close the current mailbox size is to its configured quota. This way, the test pinpoints those mailboxes that are disconnected or are growing in size abnormally.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each mailbox on the Exchange 2013/2016 server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed

2. **HOST -** The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH -** The Exchange Management Shell is a command-line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap-in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to none by default.
5. **DD FREQUENCY** - The *DD FREQUENCY* refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
6. 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
Item count:	Indicates the number of items in this mailbox.	Number	Compare the value of this measure across mailboxes to identify that mailbox with the maximum number of items. Since the likelihood of such mailboxes to grow out their quota is very high, administrators

Measurement	Description	Measurement Unit	Interpretation
			may want to closely track the variations in the item count of these mailboxes. In the event of abnormal growth, administrators can remove obsolete items from these mailboxes to reduce the mailbox size. You can use the detailed diagnosis of this measure to know when a mailbox was last logged into and logged out of.
Total item size:	Indicates the total size of this mailbox.	MB	Compare the value of this measure across mailboxes to identify that mailbox that is of the maximum size. Since the likelihood of such mailboxes to grow out their quota is very high, administrators may want to closely and continuously track the variations in the size of these mailboxes. In the event of abnormal growth therefore, administrators can remove obsolete items from these mailboxes to reduce the mailbox size.
Dumpster item count:	Indicates the total number of items in this mailbox's dumpster.	Number	The Recoverable Items folder (known in earlier versions of Exchange as the dumpster) exists to protect from accidental or malicious deletions and to facilitate discovery efforts commonly undertaken before or during litigation or investigations.

Measurement	Description	Measurement Unit	Interpretation
			By comparing the value of this measure across mailboxes, administrators can identify which mailbox's dumpster has the maximum number of items. As the dumpster too contributes to the mailbox size, the unnecessary accumulation of messages in the dumpster can also cause mailbox size to increase. In the event of abnormal growth in the size of a mailbox therefore, administrators may want to check the dumpster to see if any messages have to be permanently removed from the dumpster and remove such messages so as to minimize the size.
Dumpster item size:	Indicates the total size of the items in this mailbox's dumpster.	MB	By comparing the value of this measure across mailboxes, administrators can identify which mailbox's dumpster is of the maximum size. As the dumpster too contributes to the mailbox size, the unnecessary accumulation of messages in the dumpster can also cause mailbox size to increase. In the event of abnormal growth in the size of a mailbox therefore, administrators may want to check the dumpster to

Measurement	Description	Measurement Unit	Interpretation
			see if any messages have to be permanently removed from the dumpster and remove such messages so as to minimize the size.
Is mailbox quarantined?	Indicates whether/not this mailbox is quarantined.		Mailboxes are quarantined when they affect the availability of the mailbox database. Essentially, quarantining is designed to detect clients that are taking up too much of the Store's attention because something is going wrong. MAPI clients like Outlook use multiple threads within the Store process when they connect to mailboxes. If one or more of these threads "freeze" for some reason, they can cause the Store to consume more CPU than it should in an attempt to service the thread. The problem might be caused by corrupt mailbox data or a software bug in either the client or Store process or some other reason such as network failure. Quarantining is performed by a background thread that runs every two hours within the Store to check the number of crashes experienced by mailboxes. If a mailbox exceeds the crash threshold it is deemed to be a threat to the overall stability of the Store and is therefore put into

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>quarantine.

The values that this measure can report and their corresponding numeric values are listed hereunder:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>0</td></tr><tr><td>False</td><td>1</td></tr></table>
Note:

By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the status of the quarantining is indicated by the corresponding numeric equivalents only.</td></tr>
<tr><td>Is malbox disconnected?:</td><td>Indicates whether/not this mailbox is disconnected.</td><td></td><td>The values that this measure can report and their corresponding numeric values are listed hereunder:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>0</td></tr><tr><td>False</td><td>1</td></tr></table>
Note:

By default, this measure reports one of the Measure Values listed</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			in the table above. In the graph of this measure however, the status of the mailbox connection is indicated by the corresponding numeric equivalents only.
Warning quota size:	Indicates the warning quota configuration for this mailbox.	MB	Storage quotas let you control the size of mailboxes and manage the growth of mailbox databases. As part of the storage quota configuration, administrators can specify the maximum storage limit before a warning is issued to the user. The value of this measure signifies this storage limit. If the mailbox size reaches or exceeds the value of this measure, Exchange sends a warning message to the user.
Send quota size:	Indicates the prohibit send limit configured for this mailbox.	MB	If the mailbox reaches the prohibit send limit – i.e., the value of this measure - then Exchange prevents the user from sending new messages and displays a descriptive error message.
Send receive quota size:	Indicates the prohibit send and receive limit configured for this mailbox.	MB	If the mailbox size reaches or exceeds the value reported by this measure, Exchange prevents the mailbox user from sending new messages and will not deliver any new messages to the mailbox. Any messages sent to the mailbox are returned to the sender with a descriptive error message.

Measurement	Description	Measurement Unit	Interpretation
Warning quota percentage:	Indicates what percentage of the warning quota configured is currently consumed by this mailbox.	Percent	A value close to 100% indicates that the mailbox is growing rapidly and is about to reach the warning quota configured. If the warning limit is reached, then Exchange will send a warning message to the user. This is the first sign of the abnormal growth of a mailbox. To prevent the warning message, you can either alter the warning quota configuration or remove stale/obsolete items from the mailbox to reduce its size.
Send quota percentage:	Indicates what percentage of the prohibit send limit configured that is currently consumed by this mailbox.	Percent	A value close to 100% indicates that the mailbox is growing rapidly and is about to reach the prohibit send limit configured. If this limit is reached, then Exchange will prevent the user from sending new messages and display a descriptive error message. To ensure that mailbox users are always able to send mails from their mailbox, you can either alter the prohibit send limit configuration or remove stale/obsolete items from the mailbox to reduce its size so that it does not grow beyond the prohibit send limit configuration.
Send receive quota percentage:	Indicates what percentage of the prohibit send and receive limit configured that is currently	Percent	A value close to 100% indicates that the mailbox is growing rapidly and is about to reach the prohibit send and receive limit configured. If this limit is reached, then Exchange

Measurement	Description	Measurement Unit	Interpretation
	consumed by this mailbox.		will prevent the user from sending new messages and will not deliver any new messages to the mailbox. To ensure that mailbox users are always able to send and receive mails using their mailbox, you can either alter the prohibit send and receive limit configuration or remove stale/obsolete items from the mailbox to reduce its size so that it does not grow beyond the prohibit send and receive limit configuration.

You can use the detailed diagnosis of the *Item count* measure to know when a mailbox was last logged into and logged out of. From this, you can infer when the mailbox was last accessed and the duration of access.

Figure 3.6: The detailed diagnosis of the Item count measure

3.4.2 Mailbox Databases Test

A mailbox database consists of one/more mailboxes. Very often, administrators may have to quickly identify which database has the maximum free space in it, which database is running out of space, and which mailbox has too many mailboxes already, so that they know where more mailboxes/users can be created and where no new mailboxes can be added. The **Mailbox Databases** test assists administrators in this exercise. This test auto-discovers the mailbox databases on the Exchange server. For each mailbox database so discovered, the test reports the current database size, the number of mailboxes in that database, and the mount status of the database. Using this information, administrators can accurately identify the database with the maximum space, the database with very few mailboxes, and that which is mounted and ready for use, so that they are able to determine where more mailboxes can be created.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each mailbox database on the Exchange 2013/2016 server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH -** The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap- in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.
5. **DD FREQUENCY**- Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
6. 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
Database size:	Indicates the current size of this mailbox database.	MB	Compare the value of this measure across databases to know which databases are of a very small size. Such databases are candidates for the inclusion of more mailboxes. To know the full path to the .edb file that corresponds to a mailbox database and the server on which it is mounted, use the detailed diagnosis of this measure.
White space space:	Indicates the size of the white space in this database.	MB	Whenever emails/mailboxes are deleted, or any other cleanup functionality takes place, database space is freed for use, The free space so created is called white space. The whitespace is eventually used by new data, so the mailbox database size does not grow until it has to because it has used the whitespace. Administrators can compare the value of this measure across databases to know which database has the maximum white space. When administrators plan to create more mailboxes, such databases can be considered for use first, before attempting to expand database space or delete mailboxes to make more space.
Is mailbox database mounted?:	Indicates the mount status of this database.		The values that this measure can report and their corresponding numeric values are listed hereunder:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Mounted</td><td>1</td></tr>
<tr><td>Dismounted</td><td>0</td></tr>
</table>
Note:

By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the mount status is indicated by the corresponding numeric equivalents only.
</td></tr>
<tr><td>Mailbox count:</td><td>Indicates the number of mailboxes in this database.</td><td>Number</td><td>Compare the value of this measure across databases to know which database has the least number of mailboxes. This database can be configured with more mailboxes, if and when the need arises.

Use the detailed diagnosis of this measure to know which mailboxes are stored in this database, the email ID of each mailbox, and which features – i.e., ActiveSync, OWA, POP3, IMAP4, MAPI, and OMA - are enabled for every mailbox and which ones are disabled.</td></tr>
<tr><td>Is backup in progress?</td><td>Indicates whether/not this database is in the process of being backed up.</td><td></td><td>The values that this measure can report and their corresponding numeric values are listed hereunder:</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Yes</td><td>0</td></tr>
<tr><td>No</td><td>1</td></tr>
</table>
Note:

By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the backup status is indicated by the corresponding numeric equivalents only.
</td></tr>
<tr><td>Replication type:</td><td>Indicates the current replication status of this database.</td><td></td><td>The values that this measure can report and their corresponding numeric values are listed hereunder:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Remote</td><td>0</td></tr>
<tr><td>None</td><td>1</td></tr>
</table>
Note:

By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the database replication status is indicated by the corresponding numeric equivalents only.
</td></tr>
</table>

Using the detailed diagnosis of the Database size measure you can know the the full path to the .edb file that corresponds to a mailbox database and the server on which it is mounted.

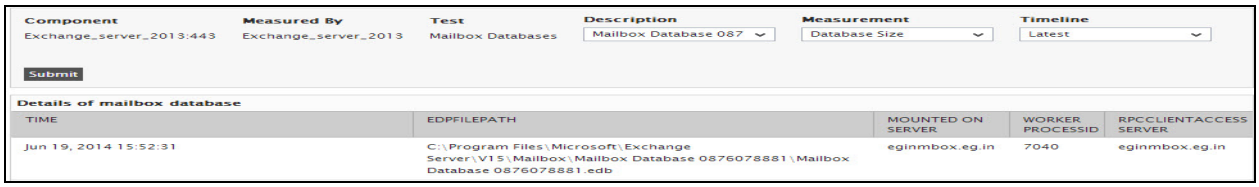

Figure 3.7: Figure 1.7: The detailed diagnosis of the Database size measure

3.4.3 Search Transport Agent Test

Content indexing is a built-in feature which enables fast searches and lookups through mailboxes and public folders stored in the Exchange Server. To assure Exchange users of a high-quality search experience, administrators should make sure that all documents are indexed quickly and in an error-free manner. For this, administrators can use the **Search Transport Agent** test. This test, at periodic intervals, runs quality checks on the indexing process, reports the count of documents that were skipped when indexing or failed, and reveals bottlenecks in indexing.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Document processing rate:	Indicates the rate at which the pipeline has processed documents.	Processing/Sec	A consistent and marked drop in this value is indicative of an indexing bottleneck.
Document failure rate:	Indicates the rate at which the documents failed processing.	Failure/Sec	A steady and significant increase in this value could indicate serious issues in indexing that need to be looked into.

Measurement	Description	Measurement Unit	Interpretation
Document skip rate:	Indicates the rate at which processing was skipped for documents.	Skips/Sec	

3.4.4 Exchange Search Indexes Test

With increasing mailbox sizes and increasing amounts of data being stored in mailboxes in the form of messages and attachments, it's crucial for users to be able to quickly search and locate the messages they need. Exchange Search indexes mailboxes and supported attachments in Exchange mailboxes to enable fast searches and lookups through mailboxes and public folders stored in the Exchange Server. To perform content indexing, Exchange Search uses the Microsoft Search Foundation. This serves as the common underlying content indexing engine in Exchange and SharePoint. If this content indexing engine is slow in processing messages, then index creation will take too long, resulting in either the failure of search queries or significant delays in the execution of queries. To ensure that user experience with Exchange Search remains top notch and user productivity improves, administrators will have to periodically check how fast the Microsoft Search Foundation processes messages in each mailbox database, promptly capture processing delays, accurately identify the mailbox database contributing to the delay, and rapidly initiate remedial action. The **Exchange Search Indexes** test assists administrators in this endeavor. At pre-configured intervals, this test monitors how well the content indexing engine processes messages in every mailbox database, and proactively alerts administrators to a potential slow down in content indexing. In the process, the test also points you to the mailbox database where the slowdown could have originated, and thus aids troubleshooting efforts.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each mailbox database on the Exchange 2013/2016 server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Mailboxes to be crawled:	Indicates the number of mailboxes in this database that are yet to be crawled for content.	Number	A low value is desired for this measure. A high value or a consistent rise in the value of this measure is a cause for concern, as it indicates an indexing bottleneck. Compare the value of this measure across databases to know which mailbox database has the maximum number of mailboxes that are pending processing by the content indexing engine. This database could be contributing to the indexing bottleneck.
Items processed rate:	Indicates the rate at which items in this mailbox database are processed.	Number	A high value is desired for this measure. A consistent drop in this value could indicate a potential indexing bottleneck. Under such circumstances, you may want to compare the value of this measure across mailbox databases to identify that database which could be contributing to the slowdown.
Age of last notification processed:	Indicates the number of seconds between notification and processing for the most recent item in this database.	Secs	A high value indicates that index updation is taking too long. Typically, indexes are updated based on notifications from the mailbox database as new messages arrive. If the time lapse between when a notification is sent out by the database to when processing is begun by the content indexing engine is very high, it hints at a lethargic content indexing engine. Compare the value of this measure across databases to know where index updation is taking the longest.

Measurement	Description	Measurement Unit	Interpretation
Items scheduled for reprocessing:	Indicates the number of items in this database that have been scheduled for reprocessing.	Number	Ideally, the value of this measure should be low. A high value is indicative of frequent processing failures. Compare the value of this measure across databases to know where the engine experienced the maximum number of failures.

3.4.5 Mailbox Databases Replication Test

To protect Exchange Server 2013/2016 mailbox databases and the data they contain, Mailbox servers and databases can be configured for high availability and site resilience. A DAG (Database Availability Group) is the base component of the high availability and site resilience framework built into Exchange 2013/2016. A DAG is a group of up to 16 Mailbox servers that host a set of databases and provides automatic, database-level recovery from failures that affect individual databases, networks, or servers. Once a DAG is created, administrators can create upto 16 copies of an Exchange 2013/2016 mailbox database on multiple Mailbox servers within this DAG. While one of the mailbox copies is set as the active copy, the other copies can be set as passive copies. Each DAG member hosting a copy of a given mailbox database participates in a process of continuous replication to keep the copies consistent. Database replication occurs between Exchange Server 2013/2016 DAG members using two different methods:

- *File Mode replication* – each transaction log is fully written (a 1MB log file) and then copied from the DAG member hosting the active database copy to each DAG member that host a passive database copy of that database.

  The other DAG members then replay the transaction log file into their own passive copy of the database to update it.

- *Block mode replication* – In this case, each database transaction is written to the log buffer on the active server and also sent to the log buffer of DAG members hosting passive copies of the database. As the log buffer becomes full, the member of the DAG is then able to build their own transaction log file for replay into their passive database copy.

Latencies in replication can cause the active and passive mailbox database copies to be out-of-sync, resulting in inconsistencies in mailbox data in the event of a failure. In order to avert such anomalies, Exchange administrators should keep a close watch on the database replication activity, spot potential delays in replication, identify where the replication process is stalling, and clear the bottleneck quickly, so that there is no loss of data when a server/database failure occurs. To achieve

this, administrators can use the **Mailbox Databases Replication** test. This test auto-discovers the mailbox databases on the Mailbox server, and for each database, reports the replication mode, the number of log files that are pending copying, inspection, and replay, and the health of the database copies and content index. This way, the test instantly captures replication bottlenecks, the source of the bottleneck – copying? inspection? replaying? - and the abnormal state of database copies. In addition, the test also reports how each database uses the disk space in the Mailbox server, thus pinpointing those databases that are consuming too much space and could hence be candidates for migration to other servers in the DAG. In the process, the test turns the spot light on a potential space crunch in the server that could cause replication to fail.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the *Microsoft Exchange 2013/2016* as the **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the each ESE database on the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH -** The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap- in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.
5. 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
Continuous replication:	Indicates the mode of continuous replication in which this database is currently running.	Number	This measure reports the value 1 if the database is running in continuous replication - block mode. This measure reports the value 0 if the database is running in continuous replication - file mode.
Generation of the last log file number:	Indicates the log generation number of the last log file which has been copied to this database.	Number	When a log file reaches a certain size, it's renamed with the next generation sequence number, and a new log file is created. This measure reports the sequence number of the last log file that was copied to a database for inspection and replay.
Generation of last log file copied notification:	Indicates the log generation number of the last log file copied to this database, which the copier knows about.	Number	Compare the value of this measure with that of the Generation of the last log file number measure to check for discrepancies. If it exists, it could indicate a problem in copying. Further investigation may be required to determine the reason for this anomaly.
Copy queue length:	Indicates the number of log generations for this database that	Number	A high value for this measure could indicate a delay in copying, and may warrant an investigation.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>are waiting to be both copied and inspected successfully.</td><td></td><td></td></tr>
<tr><td>Generation of last log file inspected:</td><td>Indicates the log generation number of the last log file related to this database that was inspected successfully.</td><td>Number</td><td>By comparing the value of this measure with that of the Generation of last log file copied notification measure, you can figure out if the sequence number of the last log file that was inspected is way behind that of the log file that was copied. If so, it could indicate that inspection is taking too long a time.</td></tr>
<tr><td>Log copy rate:</td><td>Indicates the number of bytes of logged data related to this database that was copied per second.</td><td>KB/Sec</td><td>A consistent drop in this value could indicate that log files are being copied slowly. This in turn can impact how quickly database replication is carried out.</td></tr>
<tr><td>Is log copy falling behind?:</td><td>Indicates whether or not log copying and inspection are able to keep up with log generation for this database.</td><td></td><td>If log copying and inspection are lagging behind, then this measure will return the value True. If copying and inspection are able to keep up with log generation, this measure will return the value False.

The numeric values that correspond to the above- mentioned measure values are as follows:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>True</td><td>1</td></tr><tr><td>False</td><td>0</td></tr></table>
If this measure reports the value False, it</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			is a cause for concern as it could indicate one fo the following: • Log generation rate is high; • Log copying and inspection is very slow Log generation rates can increase owing to: • Corruption of the database copy; • The presence of a number of messages in the database that are of a large size. • Many mailbox moves **Note:** By default, this measure reports one of the **Measure Value**s listed in the table above. In the graph of this measure however, the corresponding numeric values are only plotted.
Is log replay falling behind:	Indicates whether/not log replay is able to keep up with log copying and inspection for this database.		If log copying replay is lagging behind, then this measure will return the value *True*. If replay is able to keep up with log copying and inspection, this measure will return the value False. The numeric values that correspond to the above- mentioned measure values are as follows:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>True</td><td>1</td></tr>
<tr><td>False</td><td>0</td></tr>
</table>
If this measure reports the value False, it is an indicator that log replay is too slow. One of the reasons for this is the configuration of a high replay lag time. Replay lag time is a property of a mailbox database copy that specifies the amount of time, in minutes, to delay log replay for the database copy. If a high value is set for this property, then a delay in log replaying becomes inevitable. To speed up log replaying, reduce the value of this properly.

Note:

By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the corresponding numeric values are only plotted.</td></tr>
<tr><td>Generation of last log file replayed:</td><td>Indicates the log generation number of the last log file related to this database that was replayed.</td><td>Number</td><td>By comparing the value of this measure with that of the Generation of last log file inspected measure will throw light on the gap between the inspected and the replayed logs, thus revealing how many logs are pending replay. This way, administrators can figure out if the replication process is spending too much time on log replaying.</td></tr>
<tr><td>Replay lag:</td><td>Indicates the</td><td>Percent</td><td>This measure is a good indicator of the</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>percentage of actual lag in replay of the log files related to this database, relative to the configured lag.</td><td></td><td>amount of lag a database copy with replay lag configured is actually currently realizing. Replay lag time is a property of a mailbox database copy that specifies the amount of time, in minutes, to delay log replay for the database copy.

A high value for this measure is a cause for concern as it indicates that log file replaying is delayed beyond the permitted limit. This implies that a high replay lag configuration is not the reason for replaying to slow down. The real reasons for the delay should hence be investigated and determined.</td></tr>
<tr><td>Replay queue length:</td><td>Indicates the the number of log generations pertaining to this database that are waiting to be replayed.</td><td>Number</td><td>A high value for this measure could indicate a delay in replaying and may warrant an investigation.</td></tr>
<tr><td>Status:</td><td>Indicates the health and status of this database copy.</td><td></td><td>The values that this measure can take and their corresponding numeric values are as follows:

<table><tr><th>Measure Value</th><th>Numeri-c Value</th></tr><tr><td>ServiceDown</td><td>0</td></tr><tr><td>Suspended</td><td>1</td></tr><tr><td>ActivationSuspended</td><td>2</td></tr><tr><td>Failed</td><td>3</td></tr></table></td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>FailedAndSuspended</td><td>4</td></tr>
<tr><td>DisconnectedAndHealthy</td><td>5</td></tr>
<tr><td>DisconnectedAndResynchronizing</td><td>6</td></tr>
<tr><td>Dismounted</td><td>7</td></tr>
<tr><td>Dismounting</td><td>8</td></tr>
<tr><td>Resynchronizing</td><td>9</td></tr>
<tr><td>SinglePageRestore</td><td>10</td></tr>
<tr><td>Seeding</td><td>11</td></tr>
<tr><td>SeedingSource</td><td>12</td></tr>
<tr><td>Initializing</td><td>13</td></tr>
<tr><td>Mounting</td><td>14</td></tr>
<tr><td>Mounted</td><td>15</td></tr>
<tr><td>Healthy</td><td>16</td></tr>
</table>
Note:
By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the database status is indicated by the corresponding numeric equivalents only.
</td></tr>
<tr><td>Content index state:</td><td>Indicates the current state of the content index of this database.</td><td></td><td>The values that this measure can take and their corresponding numeric values are as follows:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>ServiceDown</td><td>0</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeri-c Value</th></tr>
<tr><td>Suspended</td><td>1</td></tr>
<tr><td>ActivationSuspended</td><td>2</td></tr>
<tr><td>Failed</td><td>3</td></tr>
<tr><td>FailedAndSuspended</td><td>4</td></tr>
<tr><td>DisconnectedAndHealthy</td><td>5</td></tr>
<tr><td>Dis-connectedAndResynchronizing</td><td>6</td></tr>
<tr><td>Dismounted</td><td>7</td></tr>
<tr><td>Dismounting</td><td>8</td></tr>
<tr><td>Resynchronizing</td><td>9</td></tr>
<tr><td>SinglePageRestore</td><td>10</td></tr>
<tr><td>Seeding</td><td>11</td></tr>
<tr><td>SeedingSource</td><td>12</td></tr>
<tr><td>Initializing</td><td>13</td></tr>
<tr><td>Mounting</td><td>14</td></tr>
<tr><td>Mounted</td><td>15</td></tr>
<tr><td>Healthy</td><td>16</td></tr>
</table>
Note:
By default, this measure reports one of the Measure Values listed in the table above. In the graph of this measure however, the content index state is indicated by the corresponding numeric equivalents only.
</td></tr>
<tr><td>Percentage of disk free space:</td><td>Indicates the percentage of free space in this mailbox database.</td><td>Percent</td><td>A high value is desired for this measure. If this value grows dangerously close to 0, it indicates depletion of disk space. Compare the value of this measure across databases to know which database does not have enough free</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			space. You may want to allocate more space to this database.
Disk free space:	Indicates the amount of space unused in this database.	MB	Ideally, the value of this measure should be high. A steady decrease in this value indicates depletion of disk space. Compare the value of this measure across databases to know which database does not have enough free space. You may want to allocate more space to this database.
Disk total space:	Indicates the total disk capacity of this database.	MB	

3.4.6 Mail Flow Local Connectivity Test

The true measure of the efficiency of the Exchange 2013/2016 server lies in the success of mail flow and in the speed with which mails flow. Using the **Mail Flow Local Connectivity** test, administrators can test the mail flow and evaluate how efficient the Exchange 2013/2016 server is, and in the process, capture probable latencies in mail flow rapidly.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the *Microsoft Exchange 2013/2016* as the **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Exchange 2013/2016 server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

4. **XCHGEXTENSIONSHELLPATH -** The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap- in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.
5. **EXECUTION TIMEOUT** – Specify the duration (in seconds) for which this test will wait for a response from the server; beyond this duration, the test will timeout.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Status:	Indicates the current status of the mail flow.		The value 1 for this measure indicates mail flow is successful and the value 0 for this measure indicates failure. The numeric values that correspond to these measure values are as follows: **Measure Value** / **Numeric Value** Information / 0 Warning / 1 **Note:** By default, this measure reports one of the **Measure Value**s listed in the table above. In the graph of this measure however, the alert status is indicated by the corresponding numeric equivalents only.
Message flow latency:	Indicates the time taken to deliver a message.	Secs	A low value is desired for this measure. A high value is indicative of a delivery bottleneck.

3.4.7 POP3 Service Performance Test

The POP3 service is an e-mail service that retrieves e-mail. Administrators can use the POP3 service to store and manage e-mail accounts on the mail server. When the POP3 service is installed on the mail server, users can connect to the mail server and retrieve and download e-mail to their local computer using an e-mail client that supports the POP3 protocol (such as Microsoft Outlook). If the POP3 service responds to client requests slowly, it will take a long time for users to download messages from the server, leaving them frustrated. To avoid this, administrators have to continuously track the load on the POP3 service, measure how well the service handles the client requests it receives, proactively spot probable service slowdowns, isolate the source of the slowdown, and pre-emptively fix it. This is where the **POP3 Service Performance** test helps. This test monitors the user's experience with the POP3 service and accurately pinpoints where and why the quality of the experience suffered – is it because the POP3 service was overloaded with requests? Is it because LDAP interactions took too long? or is it because RPC calls to the Mailbox server took a very long time? Precise identification of the problem source enables administrators to fix the problem in record time!

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active SSL connections:	Indicates the number of SSL or TLS connections that are currently open on the POP3 service.	Number	This is a good indicator of the current load on the POP3 service. A steady, yet significant rise in this value could hint at a potential overload condition.
Average command processing rate:	Indicates the time taken	Secs	A low value is desired for this

Measurement	Description	Measurement Unit	Interpretation
	by the service to process the last 1024 commands from the client.		measure. A high value is indicative of a processing bottleneck.
Average LDAP latency:	Indicates the average time it takes for an LDAP call to return results from a domain controller, averaged over LDAP calls in the last minute.	Secs	In the event of a slowdown in the POP3 service, administrators can compare the value of this measure with that of the Average RPC latency measure to know what could have caused the slowdown – poor responsiveness of the domain controller? or a slowdown in RPC calls to the Mailbox server?
Average RPC latency:	Indicates the average time it takes for a remote procedure call to return results from a Mailbox server, averaged over RPC calls in the last minute.	Secs	This value should be below .05 seconds at all times. A slowdown in RPC packet processing can adversely impact the user experience. In the event of a slowdown in the POP3 service, administrators can compare the value of this measure with that of the Average LDAP latency measure to know what could have caused the slowdown – poor responsiveness of the domain controller? or a slowdown in RPC calls to the Mailbox server?
Currently opened POP3 connections:	Indicates the number of connections that are currently open on the POP3 service.	Number	This measure is a good indicator of the current workload of the POP3 service.
Connection rate:	Indicates the rate at which instant messages	Msgs/Sec	These measures are good indicators of the load on the IM

Measurement	Description	Measurement Unit	Interpretation
	were received by the server.		service.

3.4.8 Replication Health Test

Replication technology in Microsoft Exchange Server enables high availability for Exchange Server databases. To maintain data integrity and prevent data loss in times of a database failure, each aspect of this mission-critical replication process – be it the availability of the active copy of the database, the replay and replication of changes to the passive copies, the underlying cluster service – have to function properly. This why, administrators need to keep tabs on the health of each of these aspects, promptly capture abnormalities, and resolve them before they affect the high availability of the Exchange databases. This is where the **Replication Health** test helps. This test checks all aspects of replication and replay and reports on the health of each aspect.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the *Microsoft Exchange 2013/2016* as the **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each aspect of replication

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command-line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap-in (exshell.psc1) for script execution. This is why, the

XCHGEXTENSIONSHELLPATH is set to *none* by default.

5. 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
Status:	Indicates whether/not this aspect of replication is in good health.		If the value of this measure is Success, it indicates that this replication aspect is in good health currently. If the value of this measure is Failure, it indicates problems in this replication aspect. The numeric values that correspond to these measure values are as follows: Measure Value: Numeric Value Success: 1 Failure: 0 If the status reported by this measure is Failure, you can use the detailed diagnosis of this test to determine the reason for the failure. **Note:**

Measurement	Description	Measurement Unit	Interpretation
			Typically, this measure reports the **Measure Value**s listed in the table above to indicate status of each replication- related activity that is monitored. However, in the graph of this measure, the **Numeric value**s are used to represent replication health.

3.4.9 Exchange Search Monitor Test

Exchange Search is an important tool that significantly improves user productivity by enabling users to quickly locate critical email messages – a task that would otherwise take hours in a mailbox that is cluttered with thousands of messages! If this Exchange Search capability is not enabled on an Exchange server or is found to take too much time, it will result in many dissatisfied and unproductive users. To assure users of a high-quality user experience with their Exchange mailboxes, administrators must continuously track the status and performance of the Exchange Search feature and proactively spot anomalies. This is exactly what the **Exchange Search Monitor** test does. This test reports whether/not the Exchange Search feature is currently enabled on a configured mailbox; if enabled, the test further reports how long Exchange Search takes to complete search queries on that mailbox. In the process, the test sheds light on the unavailability of the Exchange Search capability on a mailbox and inconsistencies in its performance.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each mailbox in every mailbox database on the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.

3. **PORT** – The port at which the host listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **MAILBOXNAME**– Specify the name of the mailbox to be monitored by this test.
6. **INDEXINGTIMEOUTINSECONDS** – Specify the duration (in seconds) for which this test will wait for a response from the server. If the server does not respond beyond this duration, the test will timeout. By default, this duration is 60 seconds.
7. 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
Status:	Indicates the current status of Exchange Search on this mailbox.		The values that this measure can report and their corresponding numeric values are listed below:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>True</td><td>0</td></tr>
<tr><td>False</td><td>1</td></tr>
</table>
If the status reported by this measure is False, you can use the detailed diagnosis of this test to know what errors.

Note:

Typically, this measure reports the Measure Values listed in the table above to indicate status of each replication- related activity that is monitored. However, in the graph of this measure, the Numeric values are used to represent replication health.
</td></tr>
<tr><td>Search time:</td><td>Indicates the time taken to complete the search queries on this mailbox.</td><td>Secs</td><td>A low value is desired for this measure. A sudden/gradual increase in the value of this measure is a cause of concern as this may be due to the unavailability of the Exchange Search capability or the inconsistencies in the performance of the mailbox.</td></tr>
</table>

3.4.10 Exchange DAG Health Summary Test

A database availability group (DAG) is the base component of the Mailbox server high availability and site resilience framework built into Microsoft Exchange Server 2013/2016. A DAG is a group of up to 16 Mailbox servers that hosts a set of databases and provides automatic database-level recovery from failures that affect individual servers or databases.

After the DAG is created, Mailbox servers can be added to the DAG. When the first server is added to the DAG, a cluster is formed for use by the DAG. DAGs make use of Windows failover clustering technology, such as the cluster heartbeat, cluster networks, and the cluster database (for storing data that changes, such as database state changes from active to passive or vice versa, or from mounted to dismounted and vice versa). As each subsequent server is added to the DAG, it's joined to the underlying cluster, the cluster's quorum model is automatically adjusted by Exchange, and the server is added to the DAG object in Active Directory.

After Mailbox servers are added to a DAG, you can configure a variety of DAG properties, such as whether to use network encryption or network compression for database replication within the DAG. You can also configure DAG networks and create additional DAG networks.

After you add members to a DAG and configure the DAG, the active mailbox databases on each server can be replicated to the other DAG members. Each DAG member hosting a copy of a given mailbox database participates in a process of continuous replication to keep the copies consistent. Database replication occurs between Exchange Server 2013/2016 DAG members using two different methods:

- *File Mode replication* – each transaction log is fully written (a 1MB log file) and then copied from the DAG member hosting the active database copy to each DAG member that hosts a passive database copy of that database. The other DAG members then replay the transaction log file into their own passive copy of the database to update it.
- *Block mode replication* – as each database transaction is written to the log buffer on the active server and also sent to the log buffer of DAG members hosting passive copies of the database. As the log buffer becomes full member of the DAG is then able to build their own transaction log file for replay into their passive database copy.

Regardless of the replication mode, data consistency and integrity can be maintained only if the active and passive copies of the database are in sync and in good health at all times, the indexes are properly built, and the copy and replay queues are healthy. To ensure this, administrators must periodically run health checks on the database copies, indexes, and queues in the DAG. This can be achieved using the **Exchange DAG Health Summary** test. This test monitors the state of the database copies, queues, and indexes in a DAG, and reports the count of unhealthy database copies, queues, and indexes. This indicates how healthy/reliable the DAG is.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Microsoft Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Preference:	Indicates the activation preference number configured for the DAG.	Number	When creating a mailbox data copy, you can specify the activation preference number, which is used as part of Active Manager's best copy selection process. It's also used to redistribute active mailbox databases throughout the DAG when using the RedistributeActiveDatabases.ps1 script. The value for the activation preference is a number equal to or greater than one, where one is at the top of the preference order. The position number cannot be larger than the number of mailbox database copies.
Total copies:	Indicates the total number of mailbox database copies in the DAG.	Number	
Healthy copies:	Indicates the number of database copies in the DAG that are healthy.	Number	A mailbox database copy is in a Healthy state it implies that it is successfully copying and replaying log files, or it has successfully copied and replayed all available log files.

Measurement	Description	Measurement Unit	Interpretation
			A high value is desired for this measure
Unhealthy copies:	Indicates the number of database copies in the DAG that are in an unhealthy state.	Number	A mailbox database copy is said to be unhealthy, if it is in the Failed, Suspended, or the Failed and Suspended state. The mailbox database copy is in a Failed state because it isn't suspended, and it isn't able to copy or replay log files. While in a Failed state and not suspended, the system will periodically check whether the problem that caused the copy status to change to Failed has been resolved. After the system has detected that the problem is resolved, and barring no other issues, the copy status will automatically change to Healthy. The mailbox database copy is in a Suspended state as a result of an administrator manually suspending the database copy by running the Suspend-MailboxDatabaseCopy cmdlet. The Failed and Suspended states is set simultaneously by the system because a failure was detected, and because resolution of the failure explicitly requires administrator intervention. An example is if the system detects unrecoverable divergence between the active mailbox database and a database copy. Unlike

Measurement	Description	Measurement Unit	Interpretation
			the Failed state, the system won't periodically check whether the problem has been resolved, and automatically recover. Instead, an administrator must intervene to resolve the underlying cause of the failure before the database copy can be transitioned to a healthy state.
Healthy queues:	Indicates the number of healthy queues in the DAG.	Number	A high value is ideal for this measure.
Unhealthy queues:	Indicates the number of unhealthy queues in the DAG.	Number	A low value is desired for this measure.
Lagged queues:	Indicates the number of lagged queues in the DAG.	Number	A lagged database copy is a passive database copy in a database availability group that has a delayed log replay time configured. Normally a passive database copy will replay the transaction log data into the database immediately, so that the passive database copy is as up to date as possible. With a lagged database copy the administrator sets a delay on the log replay, so that the database copy “lags” behind the others in terms of the latest database changes. This lag interval specifies the amount of time between when a transaction log file is generated and when it is replayed into the passive database copy. The default lag interval

Measurement	Description	Measurement Unit	Interpretation
			is 0 and the maximum lag interval is 14 days. If the value of this measure is very high, it implies that many database copy lags are occurring. You may want to consider increasing the lag interval to minimize the queue count.
Healthy indexes:	Indicates the number of healthy indexes in the DAG.	Number	
Unhealthy indexes:	Indicates the number of unhealthy indexes in the DAG.	Number	A low value is desired for this measure.

3.4.11 Exchange DAG Health Details Test

By tracking the status of the mailbox database copies, administrators can receive early warnings of data inconsistencies that are likely to creep into your DAG. By also understanding how the mailbox database copies have been configured, administrators can gauge how the current configuration will impact fail-over and the speed with which data replication occurs in the DAG, and figure out if configuration changes are warranted to ensure high availability and zero data loss. The **Exchange DAG Health Details** test provides the mailbox database copy-level insights that will enable administrators take such decisions. The test keeps an eye on the status of each mailbox database copy and alerts administrators to abnormalities in status. The test also reports the activation preference configured for every mailbox database copy and reveals how the lag time configurations per mailbox database copy are affecting the copy and replay queue lengths. Using this information, administrators can determine if changing the lag time configuration and activation preference will help enhance the DAG performance.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every mailbox database copy in the DAG being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Activation preference:</td><td>Indicates the activation preference number configured for this mailbox database copy.</td><td>Number</td><td>When creating a mailbox data copy, you can specify the activation preference number, which is used as part of Active Manager's best copy selection process. It's also used to redistribute active mailbox databases throughout the DAG when using the RedistributeActiveDatabases.ps1 script. The value for the activation preference is a number equal to or greater than one, where one is at the top of the preference order. The position number cannot be larger than the number of mailbox database copies.</td></tr>
<tr><td>Status:</td><td>Indicates the current status of this mailbox database copy.</td><td></td><td>The values that this measure can report, their description, and their corresponding numeric values are detailed in the table below:
<table>
<tr><th>Measure Value</th><th>Description</th><th>Numeric Value</th></tr>
<tr><td>Healthy</td><td>The mailbox database copy is successfully copying and replaying log files, or it has successfully</td><td>0</td></tr>
</table>
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation

Measure Value	Description	Numeric Value
	copied and replayed all available log files.	
Mounted	The active copy is online and accepting client connections. Only the active copy of the mailbox database copy can have a copy status of Mounted.	1
FailedandSuspended	The Failed and Suspended states have been set simultaneously by the system because a failure was detected, and because	2

Measurement	Description	Measurement Unit	Interpretation

Measure Value	Description	Numeric Value
	resolution of the failure explicitly requires administrator intervention.	
Failed	The mailbox database copy is in a Failed state because it isn't suspended, and it isn't able to copy or replay log files. While in a Failed state and not suspended, the system will periodically check whether the problem that caused the copy status to change to Failed has been	3

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Description</th><th>Numeric Value</th></tr>
<tr><td></td><td>resolved. After the system has detected that the problem is resolved, and barring no other issues, the copy status will automatically change to Healthy.</td><td></td></tr>
<tr><td>ServiceDown</td><td>The Microsoft Exchange Replication service isn't available or running on the server that hosts the mailbox database copy.</td><td>4</td></tr>
</table>
Note:

By default, the test reports the Measure Values listed in the table above to indicate the current state of a mailbox database copy.
</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			In the graph of this measure however, the same is represented using the numeric equivalents only.
Copy queue:	Indicates the length of the copy queue of this mailbox database copy.	Number	The copy queue length signifies the number of logs still to be copied to the passive mailbox database copy. Ideally, a passive mailbox database copy should not have a copy queue length thatis more than 10 logs. A consistent rise in the value of this measure therefore could indicate slowness in copying logs to the passive copy.
Replay queue:	Indicates the length of the replay queue of this database copy.	Number	A steady increase in the value of this measure could indicate a replication bottleneck, as it implies that log files are not getting replayed into the database copy rapidly. This could be owing to a high Replay lag time setting. Replay lag time is a property of a mailbox database copy that specifies the amount of time, in minutes, to delay log replay for the database copy. The replay lag timer starts when a log file has been replicated to the passive copy and has successfully passed inspection. By delaying the replay of logs to the database copy, you have the capability to recover the database to a specific point in time in the past. A mailbox database copy configured with a replay lag time greater than 0 is referred to as a lagged mailbox database copy, or simply, a lagged copy. To reduce the length of the replay queue, you may want to consider reducing the Replay lag time specification of the database copy.

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>A steady increase in the value of this measure could indicate that log files that have been replayed into the database copy and are not truncated quickly enough. This could be because of a high Truncation lag time setting for the database copy.

Truncation lag time is a property of a mailbox database copy that specifies the amount of time, in minutes, to delay log deletion for the database copy after the log file has been replayed into the database copy. The truncation lag timer starts when a log file has been replicated to the passive copy, successfully passed inspection, and has been successfully replayed into the copy of the database. By delaying the truncation of log files from the database copy, you have the capability to recover from failures that affect the log files for the active copy of the database.

You may want to reduce this setting to minimize the copy queue length.</td></tr>
<tr><td>Is replay lagged?:</td><td>Indicates whether/not replay is lagged for this database copy.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table>
Note:

By default, the test reports the Measure Values listed in the table above to indicate</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>whether/not replay was lagged for the database copy. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Is truncation lagged?</td><td>Indicates whether/not truncation is lagged for this database copy.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table>
Note:
By default, the test reports the Measure Values listed in the table above to indicate whether/not truncation was lagged for the database copy. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Content index:</td><td>Indicates the content index status of this mailbox database copy.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Healthy</td><td>0</td></tr><tr><td>Mounted</td><td>1</td></tr><tr><td>FailedAndSuspended</td><td>2</td></tr><tr><td>Crawling</td><td>3</td></tr><tr><td>Failed</td><td>4</td></tr></table>
Note:</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			By default, the test reports the **Measure Value**s listed in the table above to indicate content index status for the database copy. In the graph of this measure however, the same is represented using the numeric equivalents only.

3.4.12 Exchange DAG Member Health Status Test

In order to ensure that all DAG members are rightly configured to ensure continuous data replication between the database copies and instant fail-over in the event of anomalies, administrators must periodically monitor each DAG member for the status of all critical services related to these activities. This can be achieved using the **Exchange DAG Member Health Status** test. This test proactively monitors the continuous replication and the continuous replication pipeline, the availability of Active Manager, and the health and status of the underlying cluster service, quorum, and network components. In the process, the test points administrators to those services/specifications that may not be up and running or may be improper/incorrect on the DAG member, thereby impeding replication or fail-over.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Cluster service:	Verifies that the Cluster service is running and		The values that this measure can take and their corresponding

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td>reachable on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the cluster service status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Replay service:</td><td>Verifies that the Microsoft Exchange Replication service is running and reachable on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Yes</td><td>1</td></tr><tr><td>No</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the replay service status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Active manager:</td><td>Verifies that the instance of Active Manager running on the specified DAG member, or if no DAG member is specified, the local server, is in a valid role (primary, secondary, or stand-alone).</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the active manager status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Tasks RPC listener:</td><td>Verifies that the TCP log copy listener is running and reachable on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the listener status. In the graph of this measure</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>TCP listener</td><td>Verifies that the TCP log copy listener is running and reachable on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the listener status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>DAG members up:</td><td>Verifies that all DAG members are available, running, and reachable.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the member</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Cluster network:</td><td>Verifies that all cluster-managed networks on the specified DAG member, or if no DAG member is specified, the local server, are available.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the cluster network status. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Quorum group:</td><td>Verifies that the default cluster group (quorum group) is in a healthy and online state.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>Measure Values listed in the table above to indicate the quorum group state. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>File share quorum:</td><td>Verifies that the witness server and witness directory and share configured for the DAG are reachable.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the file share quorum state. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Database copy suspended:</td><td>Checks whether any mailbox database copies are in a state of Suspended on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>By default, the test reports the Measure Values listed in the table above to indicate the suspension state of the DAG member. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Database initializing:</td><td>Checks whether any mailbox database copies are in a state of Initializing on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate the initialization of the DAG member. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
<tr><td>Database disconnected:</td><td>Checks whether any mailbox database copies are in a state of Disconnected on the specified DAG member, or if no DAG member is specified, on the local server.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Passed</td><td>1</td></tr>
<tr><td>Failed</td><td>0</td></tr>
</table>
Note:

By default, the test reports the Measure Values listed in the table above to indicate the connection state of the database copies of the DAG member. In the graph of this measure however, the same is represented using the numeric equivalents only.
</td></tr>
<tr><td>Database log copy keeping up:</td><td>Verifies that log copying and inspection by the passive copies of databases on the specified DAG member, or if no DAG member is specified, on the local server, are able to keep up with log generation activity on the active copy.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:

<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Passed</td><td>1</td></tr>
<tr><td>Failed</td><td>0</td></tr>
</table>
Note:

By default, the test reports the Measure Values listed in the table above to indicate whether/not database log copying is able to keep up with log generation. In the graph of this measure however, the same is represented using the numeric equivalents only.
</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td>Database log relay keeping up:</td><td>Verifies that replay activity for the passive copies of databases on the specified DAG member, or if no DAG member is specified, on the local server, is able to keep up with log copying and inspection activity.</td><td></td><td>The values that this measure can take and their corresponding numeric values have been discussed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Passed</td><td>1</td></tr><tr><td>Failed</td><td>0</td></tr></table>Note:
By default, the test reports the Measure Values listed in the table above to indicate whether/not replay activity is able to keep up with log copying and inspection. In the graph of this measure however, the same is represented using the numeric equivalents only.</td></tr>
</table>

3.5 The Exchange Store Layer

The tests mapped to this layer monitors the overall health and efficiency of the ESE database engine and the Exchange information store.

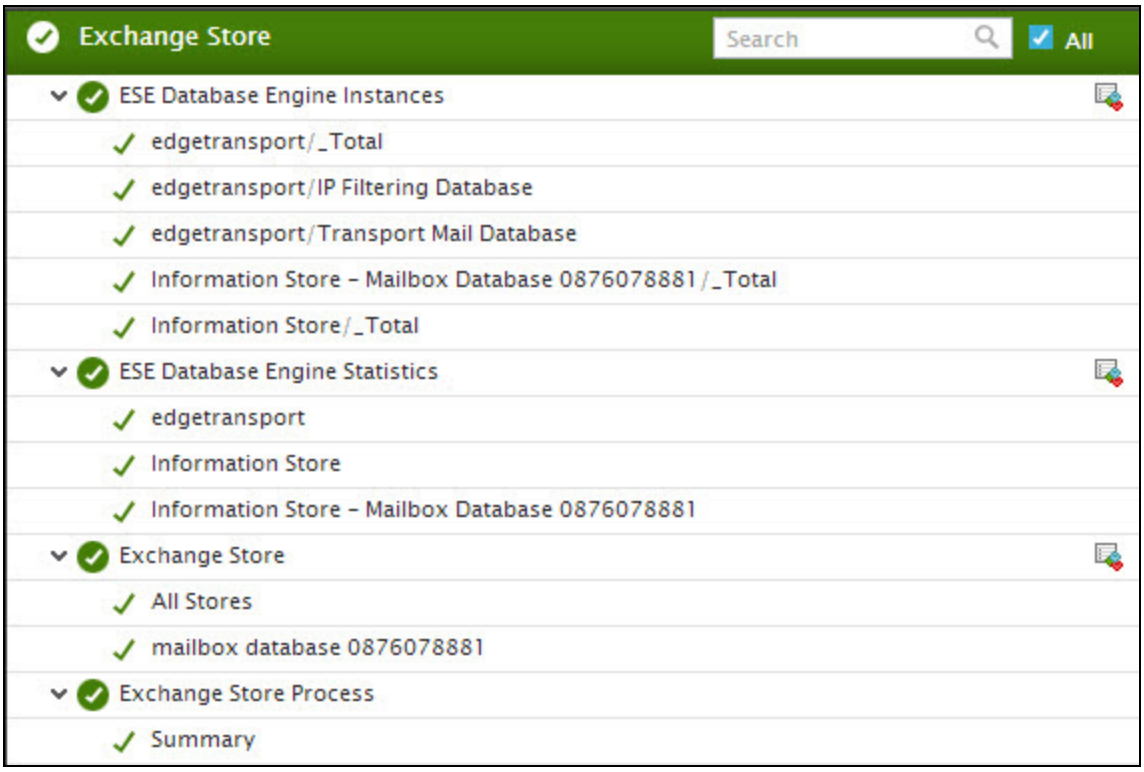

Figure 3.8: The tests mapped to the Exchange Store layer

3.5.1 ESE Database Engine Instances Test

The database engine used in Exchange Server is the Extensible Storage Engine (ESE). The Extensible Storage Engine (ESE) is an advanced indexed and sequential access method (ISAM) storage technology. ESE enables applications to store and retrieve data from tables using indexed or sequential cursor navigation.

The ESE database looks like a single file to Windows. Internally however, the database is a collection of 32KB pages arranged in a balanced B+ tree structure. In this database structure, all data is stored in leaves. At the root-level there are only pointers to internal pages. The internal pages contain pointers to the leaves, and the leaves are linked. ESE databases are organized into groups called instances.

Transactions to the ESE databases are processed (i.e., created) in server memory – in particular, the ESE cache, the log buffers, and the version store. The ESE cache helps reduce I/O operations. The version store is tied to the ESE cache and keeps track of transactions to the database while they are created. When transactions are created, they are stored in a particular log buffer. The log buffer represents a particular log file that belongs to a specific ESE database. Once the log buffer fills up with transactions, the entire log buffer is written to the log file, the log file is closed, and a new one is created. If a transaction fails for any reason, then the ESE database, once mounted, reads a checkpoint file to identify which log file is the checkpoint log,

replays that log file, writes all completed transactions in that log file that have not already been written to the database files, and reverses any incomplete transactiions.

Typically, the performance of an ESE database instance can degrade due to a lot of factors. Ineffective cache usage, poor I/O processing, high transaction load, and improper log buffer sizing are to name a few. Since a user's experience with his/her Exchange mailbox relies upon the error-free functioning of the ESE database instances in the backend, an Exchange administrator should keep an eye on the performance of every database instance, accurately identify those instances and databases that are performing poorly, and rapidly isolate the reasons for the same, so that the road-blocks can be removed quickly and the desired performance levels can be ensured. This is where the **ESE Database Engine Instances** test helps. This test auto-discovers the ESE database instances, and reports the following for every instance:

- How well each database instance is using its cache;
- How quickly an ESE database instance processes I/O requests;
- How swiftly an ESE database instance recovers from failures;
- Whether/not the log buffers are adequately sized;

This way, this test accurately pinpoints that database instance, the performance of which is bottlenecked, leads administrators to where the bottleneck lies, and thus hastens remedial action.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each ESE database instance on the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Database cache hit	Indicates the percentage	Percent	A high value is indicative of optimal

Measurement	Description	Measurement Unit	Interpretation
ratio:	of database requests to this database instance that were fulfilled by the database cache without incurring disk input/output activity.		cache usage, which translates into minimal or no direct disk accesses and high database performance. A low value on the other hand is indicative of poor cache usage. This in turn implies that direct disk accesses are high, thus escalating processing overheads and degrading database performance. You may want to determine the reason for the poor cache usage and fix it to make sure that database performance does not suffer too long. One of the common reasons for this anomaly is the insufficient cache size. In the absence of adequate memory, the Exchange server will not be able to provide adequate memory for caching objects, resulting in a high percentage of cache misses.
Database cache size:	Indicates the amount of system memory, in megabytes (MB), used by the database cache manager associated with this database instance to hold commonly used information from the database files to prevent file operations.	MB	Typically, the Exchange server reserves 25% of the total RAM for caching purposes. If the value of this measure grows closer to this allocated size, it could indicate that the cache would soon run out of memory. If this happens, then the cache will not be able to accommodate more frequently-referenced items. In the long run, this may result in a high rate of cache misses, which in turn can cause direct disk I/O to increase.

Measurement	Description	Measurement Unit	Interpretation
			To avoid this, consider increasing the cache size.
Database defragmentation task:	Indicates the number of database defragmentation tasks currently pending for this database instance.	Number	If this measure reports a very high value, it could denote that many defragmentation tasks are yet to be executed on the database; this in turn implies that the database is still largely fragmented. This will delay data retrieval from the database. Compare the value of this measure across database instance to know which instance is the most fragmented.
Average database read latency:	Indicates the amount of time this database instance took to perform a read operation.	Secs	A low value is desired for this measure. A consistent increase in this value is indicative of a bottleneck when reading from the database. Compare the value of this measure across database instances to know which instance is the slowest when reading and why.
Average database write latency:	Indicates the amount of time this database instance took to perform a write operation.	Secs	A low value is desired for this measure. A consistent increase in this value is indicative of a bottleneck when writing to this database instance. Compare the value of this measure across instances to know which instance is the slowest when writing and why.
Average log read latency:	Indicates the average time this database	Secs	A high value is a cause for concern as it impacts how quickly

Measurement	Description	Measurement Unit	Interpretation
	instance takes to read from a log file.		transaction recovery is performed in the event of a transaction failure. If the transaction log file is read slowly, then it will delay the replay of the unwritten transactions and the writing of the transactions to the database file. This in turn will significantly slow down transaction recovery.
Average log write latency:	Indicates the average time this database instance takes to write data from a log file.	Secs	A high value is a cause for concern as it impacts how quickly transaction recovery is performed in the event of a transaction failure. If the contents of the transaction log file are not written to the database file quickly, transaction recovery will be delayed.
Database read operations rate:	Indicates the rate at which this database instance completes read operations.	Reads/Sec	A consistent drop in this value can indicate a reduction in the number of read requests. It can also indicate a reading bottleneck. In the case of the latter, compare the value of this measure across database instances to know which instance is the slowest when processing read requests.
Database write operations rate:	Indicates the rate at which this database instance completes write operations.	Writes/Sec	A consistent drop in this value can indicate a reduction in the number of write requests. It can also indicate a writing bottleneck. In the case of the latter, compare the value of this measure across database instances to know which

Measurement	Description	Measurement Unit	Interpretation
			instance is the slowest when processing write requests.
Log read operations rate:	Indicates the rate at which this database instance completed log file read operations.	Reads/Sec	A consistent drop in the value of this measure is indicative of a slowdown when reading from a log file.
Log write operations rate:	Indicates the rate at which this database instance completed log write operations.	Writes/Sec	
Log record waits:	Indicates the number of log records that cannot be added to the log buffers of this database instance because the log buffers are full.	Records/Sec	This measure should be as close to zero as possible. If it is not, it might indicate that the size of the log buffer might be a bottleneck. Increasing the memory may solve this problem.
Log thread waits:	Indicates the number of threads waiting in this database instance for their data to be written to the log buffer so that the update of the database can be completed.	Number	This measure should be as low as possible. A high value for this measure may indicate that the log buffer might be a bottleneck. Increasing the memory may solve this problem.
Transaction log files:	Indicates the amount of work, expressed in terms of the number of log files, that needs to be redone or undone to the database files of this database instance if the process fails.	Number	The value of this measure should be below 500 at all times. For a healthy server, this measure should report a value between 20 and 30 for each database instance. If this measure increases continually, this indicates either a long- running transaction, (which will impact the version store), or a

Measurement	Description	Measurement Unit	Interpretation
			bottleneck involving the database disks. In this case, in the event of a failure, transaction recovery will take a considerably long time. Under such circumstances, you may want to consider decreasing the checkpoint depth, so that transactions are written to the database quickly.
Session in use:	Indicates the number of sessions to this database instance that are currently open for use by client threads.	Number	
Session used:	Indicates the percentage of sessions to this database instance that are currently open for use by client threads.	Percent	A high value is desired for this measure. A very low value could indicate that too many open sessions are idle.
Database tables cache hit ratio:	Indicates the percentage of tables in this database instance that were opened using the cached schema information.	Percent	A significantly low value indicates that the Exchange server is not having enough free memory. Increasing the memory may solve this problem.
Database tables opened:	Indicates the number of tables in this database instance that were opened per second.	Opens/sec	
Version buckets allocated:	Indicates the total number of version buckets allocated to this database instance.	Number	The "version buckets" are the message queue database transactions that are kept in memory. All changes that are made to the message queue

Measurement	Description	Measurement Unit	Interpretation
			database stay in memory until those changes can be committed to transaction log files. Factors that can increase the version buckets may be virus issues, integrity of the message queue database, or hard drive performance. The default maximum version bucket count is 16,384. If version buckets reach 70% of maximum, the server is at risk of running out of the version store.

3.5.2 ESE Database Engine Statistics Test

If a user experiences slow downs when accessing his/her Exchange mailbox, nine out of 10 times the reason would be the underlying ESE database. Various factors affect database performance. An under- sized / poorly configured database cache, latency in I/O processing, delay in reading/replaying log files, a session overload, can all cause an ESE database's performance to degrade. To ensure that user productivity is not impacted by poor database performance, administrators should closely monitor each aspect of the performance of every ESE database on the Exchange server, capture anomalies before users notice, and resolve them before they affect the user experience with his/her mailbox. This is where the **ESE Database Engine Statistics** test helps. This test auto-discovers the ESE databases on the Mailbox server, and for each database so discovered, reports the following:

- How effectively is the database using its cache?
- Is the cache sized right?
- How quickly is the database processing read/write requests to it?
- Are log files read and replayed rapidly?
- Are too many sessions to the database idle?

Using the insights provided by this test, administrators can accurately identify slow/overloaded databases, pinpoint why the database performance is bottlenecked, and clear the bottleneck well before users notice and complain.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the each ESE database on the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Pages in database cache being compressed:	Indicates the percentage of pages in the cache of this database that are currently compressed.	Percent	In order to conserve the memory resources of the Exchange Mailbox server, the 'Back Pressure' feature of the server automatically performs 'message dehydration' if the server's memory usage crosses a configured threshold (by default, this is 94%). Message dehydration is the act of removing unnecessary elements of queued messages – i.e., compressing messages - that are cached in memory. Typically, complete messages are cached in memory for enhanced performance. Removal of the MIME content of

Measurement	Description	Measurement Unit	Interpretation
			queued messages from memory reduces the memory that is used. A high value for this measure therefore indicates that messages in cache have been compressed to a large extent, thus freeing up a significant amount of system memory. On the other hand, a high value for this measure can also increase the latency of mailbox operations, because, in the absence of complete messages in the cache, the server will be forced to read the messages directly from the message queue database. To avoid this, the mailbox server has to be sized with adequate memory and the memory usage threshold of the Backprssure feature should also be set high enough.
Page requests fulfilled by database cache:	Indicates the percentage of page requests to this database that were fulfilled by the database cache without causing a file operation.	Percent	If this percentage is too low, it could be because the database cache size may be too small. You may want to consider increasing the cache size, so as to reduce the processing overheads that may be incurred on account of direct database accesses.

Measurement	Description	Measurement Unit	Interpretation
Database cache size:	Indicates the amount of system memory, in megabytes, used by the database cache manager to hold commonly used information from the database file(s) to prevent file operations.	MB	If the database cache size seems to be too small for optimal performance, and there is very little available memory on the system, an increase of memory in the system may increase performance. If there is a large amount of available memory on the system, and the database cache size is not growing beyond a certain point, the database cache size may be capped at an artificially low limit. An increase in this limit may increase performance.
Effective database cache size:	Indicates the amount of system memory, in megabytes, that, hypothetically, would be used by this database cache manager if all used dehydrated/compressed database cache buffers were rehydrated/uncompressed.	MB	
Resident database cache size:	Indicates the amount of system memory (in megabytes) used by this database cache that is currently part of the working set of the process.	MB	If the value of this measure is ever significantly smaller than that of the Database cache size measure then the operating system has chosen to reclaim that system memory for use in other parts of the system. The database cache will recover

Measurement	Description	Measurement Unit	Interpretation
			from this event but if this is a common occurrence then it can lead to significant performance problems.
Database cache page faults rate:	Indicates the rate of page faults that cannot be serviced because there are no pages available for allocation from the database cache.	Stalls/Sec	If the value of this measure is nonzero most of the time, the clean threshold may be too low.
Average read operation latency per database:	Indicates the average time taken by this database to perform read operations.	Secs	A low value is desired for this measure. A consistent increase in this value is indicative of a bottleneck when reading from the database. Compare the value of this measure across databases to know which database is the slowest when reading and why.
Rate of database read operations completed:	Indicates the rate at which this database completes read operations.	Reads/Sec	A consistent drop in this value can indicate a reduction in the number of read requests. It can also indicate a reading bottleneck. In the case of the latter, compare the value of this measure across databases to know which database is the slowest when processing read requests.
Average write operation latency per database:	Indicates the average time taken by this database to perform write operations.	Secs	A high value is a cause for concern as it impacts how quickly transaction recovery is performed in the event of a

Measurement	Description	Measurement Unit	Interpretation
			transaction failure. If the contents of the transaction log file are not written to the database file quickly, transaction recovery will be delayed.
Rate of database write operations completed:	Indicates the rate at which this database completes write operations.	Writes/Sec	A consistent drop in this value can indicate a reduction in the number of write requests. It can also indicate a writing bottleneck. In the case of the latter, compare the value of this measure across databases to know which database is the slowest when processing write requests.
Average read operation latency per log file:	Indicates the average time taken by this database to read a log file.	Secs	A high value is a cause for concern as it impacts how quickly transaction recovery is performed in the event of a transaction failure. If the transaction log file is read slowly, then it will delay the replay of the unwritten transactions and the writing of the transactions to the database file. This in turn will significantly slow down transaction recovery.
Rate of log file read operations completed:	Indicates the rate at which this database completes log file read operations.	Reads/Sec	A consistent drop in the value of this measure is indicative of a slowdown when reading from a log file.
Average write operation latency	Indicates the average time taken by this database to	Secs	A high value is a cause for concern as it impacts how

Measurement	Description	Measurement Unit	Interpretation
per log file:	write from a log file.		quickly transaction recovery is performed in the event of a transaction failure. If the contents of the transaction log file are not written to the database file quickly, transaction recovery will be delayed.
Rate of log file write operations completed:	Indicates the rate at which this database completes log file write operations.	Reads/Sec	A consistent drop in the value of this measure is indicative of a slowdown when writing from a log file.
Database sessions used by client threads:	Indicates the percentage of database sessions currently open for use by client threads.	Percent	A high value is desired for this measure. A very low value could indicate that too many open sessions are idle.
Database sessions in use:	Indicates the number of database sessions currently open for use by client threads.	Number	

3.5.3 Exchange Store Test

A mailbox database in the Exchange Store is said to be in good health if:

- The mailbox database processes RPC requests from clients quickly;
- The message load on the mailbox database is commensurate to its processing ability;
- The mailbox database does not consist of any quarantined mailboxes
- The mailbox database is well-maintained by configuring maintenance schedules

This implies that a slowdown in processing RPC requests, a delay in email delivery/submission, or one/more corrupted mailboxes in the database, can severely hamper mailbox database health. This in turn can significantly impact the user experience with the mailbox server. If this is to be avoided, administrators should keep a watchful eye on all aspects of performance of every mailbox database on the Exchange server, so that they can proactively capture current/potential failures/delays. This is where the **Exchange Store** test helps. For every mailbox database on the Exchange server, this test

measures the load on the database, the processing ability of the database, and the overall health and upkeep of the database. In the process, the test accurately pinpoints those mailbox databases where something is wrong and provides pointers to what could be wrong!

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each mailbox database

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. 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
Percentage of RPC requests:	Indicates the percent of MAPI RPCs that are currently in progress in this database as compared to the maximum allowed.	Percent	If the value of this measure keeps growing close to 100% at a rapid pace, it could indicate increased RPC load or a bottleneck in one or more resources. If the value touches 100%, then the mailbox database will deny new connections. To resolve this problem, determine the resources that are creating a bottleneck, and then try to mitigate the problem. Possible bottlenecks are disk reads or writes, processor time, available memory, and network configuration.
Active mailboxes:	Indicates the number of active mailboxes in this database.	Number	Use the detailed diagnosis of this measure to know the names of the active mailboxes and the number of items in each mailbox.
Database maintenance rate:	Indicates the rate at which database level maintenances are processed by this database.	Maintenances/Sec	A high value is desired for this measure. Compare the value of this measure across databases to know which database is processing database- level maintenance tasks slowly.
Full- refresh rate of lazy indexes:	Indicates the number of lazy indexes being full refreshed per second by this database.	Refresh/Sec	
Incremental refresh rate of	Indicates the number	Refresh/Sec	

Measurement	Description	Measurement Unit	Interpretation
lazy indexes:	of lazy indexes being incrementally refreshed per second by this database.		
Invalidated lazy indexes:	Indicates the rate at which lazy indexes are being invalidated by this database due to the version incompatibility.	Invalidation/Sec	A low value is desired for this measure.
Lazy indexes creation rate:	Indicates the rate at which this database creates lazy indexes.	Created/Sec	
Lazy indexes deletion rate:	Indicates the rate at which this database deletes lazy indexes.	Deleted/Sec	
Mailbox maintenance items:	Indicates the number of mailbox maintenance items in this database.	Number	
Mailbox maintenance rate:	Indicates the rate at which this database processes mailbox level maintenances.	Maintenances/Sec	A high value is desired for this measure. Compare the value of this measure across databases to know which database is processing mailbox- level maintenance tasks slowly.
Mailboxes with maintenance items:	Indicates the number of mailboxes with maintenance items.	Number	
Message delivery rate:	Indicates the number of messages delivered to this database per second.	Msgs/Sec	

Measurement	Description	Measurement Unit	Interpretation
Message submission rate:	Indicates the rate at which messages were submitted by this database for delivery.	Msgs/Sec	Ideally, the value of this measure should be high. Compare the value of this measure across databases to in which database mail delivery is bottlenecked.
Active background tasks:	Indicates the number of background tasks currently executing in this database.	Number	
Active WLM tables under maintenance:	Indicates the number of active WLM LogicalIndex maintenance tables under maintenance in this database.	Number	
Mailboxes with WLM tables under maintenance:	Indicates the number of mailboxes in this database that are marked for WLM LogicalIndex maintenance table maintenance.	Number	
Currently processing maintenance tasks:	Indicates the number of maintenance tasks that are currently processed by this database.	Number	This is a good indicator of the maintenance workload on the database.
Maintenance tasks scheduled:	Indicates the number of LogicalIndex maintenance tasks scheduled for this database.	Number	This is a good indicator of the potential maintenance workload on the database.
Rate of property	Indicates the rate at	Messages/Sec	Property promotion refers to the

Measurement	Description	Measurement Unit	Interpretation
promoted messages:	which properties were promoted for messages in this database.		process of extracting values from properties of a message and writing those values to corresponding columns on the database where the document is stored. When the message property changes, the changes can be automatically written back to the database.
Property promotions rate:	Indicates the rate at notepwhich this database promoted message properties.	Promotions/Sec	Ideally, the value of this measure should be high. A low value is indicative of a bottleneck when performing property promotions.
Quarantined mailboxes:	Indicates the number of mailboxes in this database that are quarantined.	Number	Ideally, the value of this measure should be 0. A non-zero value is indicative of a quarantined mailbox in the database. Quarantining is designed to detect clients that are taking up too much of the Store's attention because something is going wrong. MAPI clients like Outlook use multiple threads within the Store process when they connect to mailboxes. If one or more of these threads "freeze" for some reason, they can cause the Store to consume more CPU than it should in an attempt to service the thread. The problem might be caused by corrupt mailbox data or a software bug in either the client or Store process or some other reason such as network failure. In any case, the

Measurement	Description	Measurement Unit	Interpretation
			freezing of threads or their abnormal termination is bad news! Quarantining is performed by a background thread that runs every two hours within the Store to check the number of crashes experienced by mailboxes. If a mailbox exceeds the crash threshold it is deemed to be a threat to the overall stability of the Store and is therefore put into quarantine. Compare the value of this measure across databases to know which database has the maximum number of quarantined mailboxes. If this value is very high, it could be because a very small crash threshold has been set for that database. If so, you may want to change this crash threshold, so that fewer mailboxes are quarantined.
RPC average latency:	Indicates the amount of time spent by this database in RPC request processing.	Secs	This value should be below 0.05 seconds at all times. A slowdown in RPC packet processing can adversely impact the user experience.
RPC operations rate:	Indicates the rate at which this database processes RPC operations.	Operations/Sec	Generally, spikes in RPC requests that do not increase RPC operations rate indicate that there are bottlenecks preventing the

Measurement	Description	Measurement Unit	Interpretation
			store from fulfilling the requests in a timely manner. It is relatively simple to identify where the bottlenecks are occurring with regards to RPC requests and RPC operations rate. If the client experiences delays, but the RPC requests are zero and the RPC operations rate is low, the performance problem is happening before Exchange processes the requests (that is, before the Microsoft Exchange Information Store service actually gets the incoming requests). All other combinations point to a problem either while Exchange processes the requests or after Exchange processes those requests.
RPC packets rate:	Indicates the rate at which RPC packets are processed by this database.	Packets/Sec	A consistent drop in this value could indicate a slowdown in RPC request processing.
Pending pool RPC async notification calls:	Indicates the total number of async notification calls pending in all RPC context handle pools of this database.	Number	Async MAPI Notifications use Asynchronous RPC to receive notifications from the Exchange Server. This allows MAPI to park a request with the Exchange Server and not have to wait for the next remote operation to get notifications.
Active RPC context handle	Indicates the number of active RPC context	Number	

Measurement	Description	Measurement Unit	Interpretation
pools:	handle pools of this database.		
RPC requests:	Indicates the number of MAPI RPC requests currently in progress in this database.	Number	This is a good indicator of the current RPC workload of the database.

The detailed diagnosis of the *Active Mailboxes* measure reveals the names of the active mailboxes on a particular mailbox database and the number of items in each mailbox.

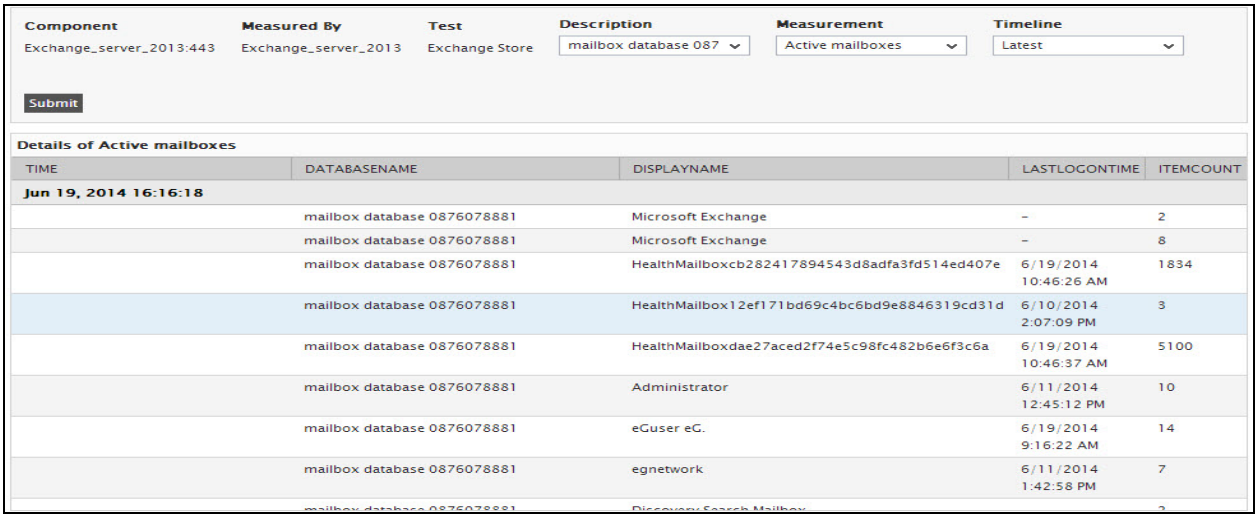

Figure 3.9: The detailed diagnosis of the Active mailboxes measure

3.5.4 Exchange Store Process Test

The **Exchange Store** test discussed previously enables administrators to identify the specific mailbox databases that are experiencing processing bottlenecks. In the event of a slowdown, this test will be useful for isolating those databases that could be contributing to the slowdown. Apart from this, what would further ease the troubleshooting pains of administrators in such situations is knowing what type of database operations performed by which client protocols/types are causing these databases to slow down. Could any client protocol/type be overloading the database with RPC requests, causing it to slow down? Could any client type/protocol be generating way too many log files in the database allowing little free space for speedy processing of requests? Are the RPC requests from any client type/protocol taking too much time being processed? Are requests from any client type frequently triggering LDAP searches, thus increasing the request processing time? Were

too many database operations (i.e., message creation, deletion, updations, etc.) performed by a client type/protocol too often, resulting in a slowdown? Using the **Exchange Store Process** test, administrators can find quick and accurate answers to all these questions. This test auto-discovers the client types/protocols communicating with the Exchange store, reports the RPC request load, operational load, and log file load that each client type/protocol imposed on the store, and measures the impact of this load on the processing time, growth, and overall performance of the store. In the process, the test leads administrators to those client types/protocols that could be affecting database performance and how.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each client type/protocol using the Exchange store or across all client types/protocols discovered

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.
4. **REPORT BY SUMMARY** –By default, this flag is set to **true**. This implies that the test will report metrics for only a Summary descriptor, which reveals the overall Exchange store health across all client types/protocols that communicate with the store. If you want the test to report metrics for each client type/protocol that uses the store, then set this flag to **false**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Admin RPC requests:	Indicates the number of admin requests from this client type/protocol that are currently being processed by the information store. For the Summary descriptor, this measure will report the total number of admin	Number	This is a good indicator of the current admin RPC request load on the Exchange store. Compare the value of this measure across client types/protocols to know which client type is currently overloading the store.

Measurement	Description	Measurement Unit	Interpretation
	requests currently being processed by the store, regardless of the client type/protocol that is making the requests.		
Administrative RPC request rate:	Indicates the rate at which admin requests from this client type/protocol are processed by the store. For the Summary descriptor, this measure will report the aggregate rate at which admin requests are processed by the store, regardless of the client type/protocol that is making the requests.	Requests/Sec	A consistent drop in this value could indicate a bottleneck in request processing.
LDAP search rate:	Indicates the rate at which LDAP searches were performed while processing requests for this client type/protocol. For the Summary descriptor, this measure will report the aggregate rate at which LDAP searches were performed by the store.	Searches/Sec	Frequent LDAP searches when processing requests can delay request processing. You can compare the value of this measure across client types/protocols to know which client type/protocol is triggering the maximum LDAP searches.
Jet log record data:	Indicates the rate at which log data is	KB/Sec	A consistent increase in the value of these measures could indicate a steady increase in database size

Measurement	Description	Measurement Unit	Interpretation
	generated when processing requests for this client type/protocol. For the **Summary** descriptor, this measure will report the aggregate rate at which log data is generated in the store when processing requests from client types/protocols.		owing to log file generation. Compare the value of these measures across client types/protocols to know which client type/protocol is triggering the generation of too many log files, and is eroding database space.
Jet log record rate:	Indicates the rate at which database log records are generated while processing requests for this client type/protocol. For the Summary descriptor, this measure will report the aggregate rate at which database log records are generated in the store when processing requests from client types/protocols.	Records/Sec	
Jet pages modified rate:	Indicates the rate at which database pages are modified while processing requests for this client type/protocol. For the Summary descriptor,	Modified/Sec	These measure are good indicators of the type of workload imposed by a client on the store and how well the store handles the workload.

Measurement	Description	Measurement Unit	Interpretation
	this measure will report the aggregate rate at which database log records are modified in the store when processing requests from client types/protocols.		
Jet pages pre-read rate:	Indicates the rate at which database pages are preread from disk while processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which database were pre-read from the store when processing requests from client types/protocols.	Preread/Sec	
Jet pages read rate:	Indicates the rate at which the database pages are read from disk while processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which database pages are read from the store when processing	Read/Sec	

Measurement	Description	Measurement Unit	Interpretation
	requests from client types/protocols.		
Jet pages referenced rate:	Indicates the rate at which database pages are referenced while processing requests for this client. For the Summary descriptor, this measure will report the aggregate rate at which database pages are referenced in the store when processing requests from client types/protocols.	Referenced/sec	
Jet pages remodified rate:	Indicates the rate at which the database pages are remodified while processing requests for this client. For the Summary descriptor, this measure will report the aggregate rate at which database pages are modified in the store when processing requests from client types/protocols.	Remodified/Sec	
Full-refresh rate of lazy indexes:	Indicates the number of lazy indexes being full refreshed per second when	Refresh/Sec	

Measurement	Description	Measurement Unit	Interpretation
	processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which lazy indexes are full refreshed when processing requests across all client types/protocols.		
Incremental refresh rate of lazy indexes:	Indicates the number of lazy indexes being incrementally refreshed per second when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which lazy indexes are incrementally refreshed when processing requests across all client types/protocols.	Refresh/Sec	
Lazy indexes creation rate:	Indicates the rate at which lazy indexes are created when processing requests for this client type. For the Summary descriptor, this	Created/Sec	

Measurement	Description	Measurement Unit	Interpretation
	measure will report the aggregate rate at which lazy indexes are created when processing requests across all client types/protocols.		
Lazy indexes deletion rate:	Indicates the rate at which lazy indexes were deleted when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which lazy indexes are deleted when processing requests across all client types/protocols.	Deleted/Sec	
Message creation rate:	Indicates the rate at which messages were created in the store when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which messages are created when processing requests across all client	Created/Sec	These measures are good indicators of the operational/transactional load imposed by a client on the store. Compare the value of these measures across client types to identify that client which is imposing the maximum workload on the store.

Measurement	Description	Measurement Unit	Interpretation
	types/protocols.		

Measurement	Description	Measurement Unit	Interpretation
Message deletion rate:	Indicates the rate at which messages were deleted from the store when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which messages are deleted when processing requests across all client types/protocols.	Deleted/Sec	
Messages opened rate:	Indicates the rate at which messages were opened in the store when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which messages are opened when processing requests across all client types/protocols.	Opened/Sec	
Messages updated rate:	Indicates the rate at which messages were updated in the store when processing requests for this client type.	Updated/Sec	

Measurement	Description	Measurement Unit	Interpretation
Property promotions rate:	Indicates the rate at which message properties were promoted when processing requests for this client type. For the Summary descriptor, this measure will report the aggregate rate at which message properties were promoted when processing requests across all client types/protocols.	Promotions/Sec	Property promotion refers to the process of extracting values from properties of a message and writing those values to corresponding columns on the database where the document is stored. When the message property changes, the changes can be automatically written back to the database.
RPC average latency:	Indicates the amount of time spent by the store in processing RPC requests from this client type. For the Summary descriptor, this measure will report the total time spent by the store in processing requests from all client types/protocols.	Secs	This value should be below 0.05 seconds at all times. A slowdown in RPC packet processing can adversely impact the user experience.
RPC data received rate:	Indicates the rate at which data is received from RPC clients when processing requests for this client type/ptocols. For the Summary descriptor,	KB/Sec	

Measurement	Description	Measurement Unit	Interpretation
	this measure will report the aggregate rate at which data is received from RPC clients when processing requests across all client types/protocols.		
RPC data sent rate:	Indicates the rate at which data is sent to RPC clients. For the Summary descriptor, this measure will report the aggregate rate at which data is sent to RPC clients when processing requests across all client types/protocols.	KB/Sec	
RPC operations rate:	Indicates the rate at which RPC operations were processed for this client type. For the Summary descriptor, this measure will report the aggregate rate at which RPC operations were processed across all client types/protocols.	Operations/Sec	Generally, spikes in RPC requests that do not increase RPC operations rate indicate that there are bottlenecks preventing the store from fulfilling the requests in a timely manner. It is relatively simple to identify where the bottlenecks are occurring with regards to RPC requests and RPC operations rate. If the client experiences delays, but the RPC requests are zero and the RPC operations rate is low, the performance problem is happening before Exchange processes the requests (that is, before the Microsoft Exchange Information

Measurement	Description	Measurement Unit	Interpretation
			Store service actually gets the incoming requests). All other combinations point to a problem either while Exchange processes the requests or after Exchange processes those requests.
RPC packets rate:	Indicates the rate at which RPC packets are processed by the store when processing requests for this client type. For the Summary descriptor, this measure will report the RPC packets were processed when servicing requests from all client types/protocols.	Packets/Sec	A consistent drop in this value could indicate a slowdown in RPC request processing.
RPC requests:	Indicates the number of MAPI RPC requests currently in progress in the store for this client type. For the Summary descriptor, this measure will report the total number of MAPI RPC requests currently in progress for all client types/protocols.	Number	This is a good indicator of the RPC workload currently imposed on the store by a client.

3.6 The Mailbox Transport Services Layer

The Mailbox Transport Service consists of two separate services: the Mailbox Transport Submission service and Mailbox Transport Delivery service. The Mailbox Transport Delivery service receives

SMTP messages from the Transport service on the local Mailbox server or on other Mailbox servers, and connects to the local mailbox database using an Exchange remote procedure call (RPC) to deliver the message. The Mailbox Transport Submission service connects to the local mailbox database using RPC to retrieve messages, and submits the messages over SMTP to the Transport service on the local Mailbox server, or on other Mailbox servers.

The tests mapped to this layer monitor the core components and functions of the Mailbox Transport Submission and the Mailbox Transport Delivery services and reports abnormalities.

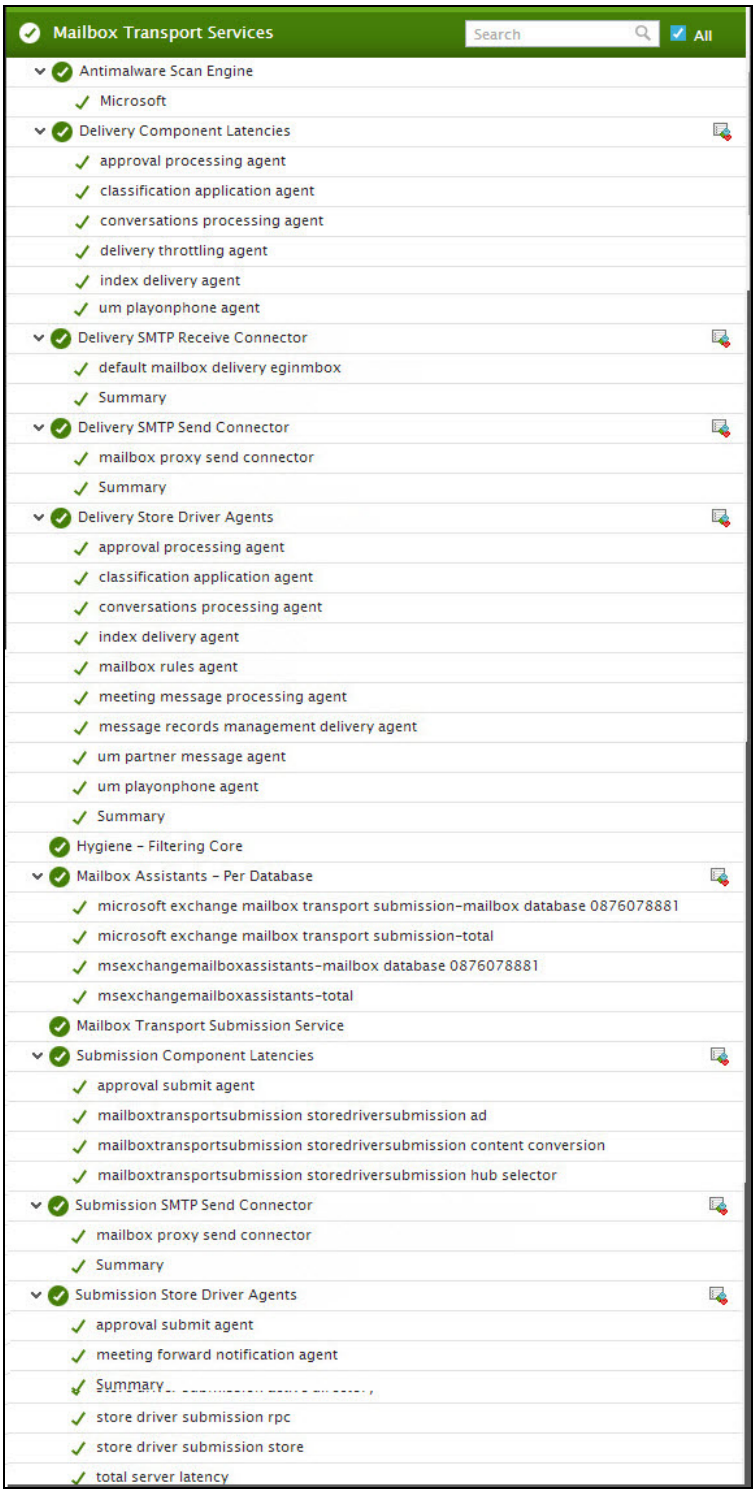

Figure 3.10: The tests mapped to the Mailbox Transport Services layer

3.6.1 Antimalware Scan Engine Test

Malware is comprised of viruses and spyware. Viruses infect other programs and data, and they spread throughout your computer looking for programs to infect. Spyware refers to malware that gathers your personal information, such as sign-in information and personal data, and sends it back to its author. The Microsoft Exchange Server 2013/2016 anti-malware protection feature helps combat malware in your email messaging environment.

There are several anti-malware protection options in Exchange 2013/2016:

- **Built-in anti-malware protection in Exchange 2013/2016** You can use the built-in Exchange on-premises anti-malware protection feature in order to help you combat malware. This basic anti-malware protection can be turned off, replaced, or paired with a cloud-based service (such as Microsoft Exchange Online Protection or Microsoft Forefront Online Protection for Exchange) to provide a layered defense.
- **Cloud-hosted anti-malware protection** You can elect to purchase the Microsoft Forefront Online Protection for Exchange (FOPE) hosted email filtering service or the next version of this service, Exchange Online Protection (EOP). The service leverages partnerships with several best of breed anti-malware engines, thereby providing efficient, cost effective, multi-layered anti-malware protection.

Regardless of which option you choose, you need to ensure that the anti-malware engine functions in an error-free manner and is able to protect your critical email communication from harm. This is why you need the **Antimalware Scan Engine** test from eG. This test auto-discovers the anti-malware scan engines in use in your environment, and for each engine, captures errors in engine functioning and reveals how quickly the engine scans messages and detects malware. In the process, the test sheds light on an engine's incapacities.

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for envery anti-malware scan engine deployed on the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Engine errors:	Indicates the number of engine errors in the last number.	Number	Ideally, the value of this measure should be 0. A non-zero value is indicative an unhealthy engine.
Items processed rate:	Indicates the rate at which this engine processes items.	Processed/Sec	A consistent drop in this rate could indicate a processing bottleneck on the engine.
Malware items detected:	Indicates the number of items detected by this engine as containing malware. .	Number	A high value could indicate an infestation.
Average malware scan time per item:	Indicates the average time that this engine took per item to scan for malware.	Secs	Ideally, the value of this measure should be low. A high value could indicate slowness when scanning, and warrants further investigation.

3.6.2 Delivery Component Latencies Test

One/more latent delivery components can stall email delivery to user mailboxes for long hours. In mission-critical environments, such unprecedented time lags in email delivery can increase user frustration, impact productivity, affect revenues, and escalate costs. To avoid this, administrators should closely monitor how much time each component engaged in email delivery is spending on processing the mail messages, and where the bottleneck is. The **Delivery Component Latencies** test helps with this. For each component engaged in email delivery, this test reports the maximum time that component took to process email messages 90%, 95%, and 99% of the time. This way, the test points to the highly latent components that could be disrupting email delivery.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each delivery component engaged in email delivery

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
90th percentile message latency:	Indicates the maximum time spent by this delivery component in processing the delivery of emails 90% of the time.	Secs	By comparing the values of these measures across delivery components, you can identify where email delivery is bottlenecked.
95th percentile message latency:	Indicates the maximum time spent by this delivery component in processing the delivery of emails 90% of the time.	Secs	
99th percentile message latency:	Indicates the maximum time spent by this delivery component in processing the delivery of emails 99% of the time.	Secs	

3.6.3 Delivery SMTP Receive Connector Test

Receive connectors control the flow of inbound messages to your Exchange organization. They are configured on computers running Microsoft Exchange Server 2013/2016 with the Transport service, or in the Front End service on a Client Access server.

Each Receive connector listens for inbound connections that match the settings of the Receive connector. A Receive connector listens for connections that are received through a particular local IP address and port, and from a specified IP address range.

When you install a Mailbox server running the Transport service, two Receive connectors are created. No additional Receive connectors are needed for typical operation, and in most cases the default Receive connectors don't require a configuration change. These connectors are the following:

- *Default <server name>* Accepts connections from Mailbox servers running the Transport service and from Edge servers.
- *Client Proxy <server name>* Accepts connections from front-end servers. Typically, messages are sent to a front-end server over SMTP.

When mail delivery slows down or when too many mails are returned undelivered, administrators should be able to rapidly identify the connector responsible for this, so that they can figure out how such connectors can be reconfigured to avoid the slowness or the message rejections. To determine this, administrators can use the **Delivery SMTP Receive Connector** test. This test auto-discovers the default and user-configured SMTP receive connectors on the Exchange server. For each discovered connector, the test reports the incoming load on the connector, the rate at which each connector processed the load, and the count of mails rejected by the connector. In the process, the test points to overloaded connectors, slow connectors, and the ones that rejected a vast majority of emails, so that administrators can fine-tune the problematic connectors and minimize the anomalies.

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each SMTP receive connector

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data received rate:	Indicates the rate at which email data was received by this connector.	KB/Sec	In the event of a slowdown, you can compare the value of this measure across connectors to know which

Measurement	Description	Measurement Unit	Interpretation
			connector is processing emails slowly.
Inbound connections to receive connector:	Indicates the current number of inbound connections to this connector.	Number	Compare the value of this measure across connectors to know which connector is being used the most. Overloaded connectors can thus be identified. You may want to configure additional connectors to ensure that load is uniformly balanced across connectors.
Messages received by receive connector:	Indicates the rate at which messages are received by this connector.	Msgs/Sec	
Messages rejected due to size restriction:	Indicates the number of messages that were rejected by this connection due to size restriction.	Number	Ideally, this value should be 0. A non- zero value implies that one/more messages have been rejected owing to violation of size limits. You can apply limits to messages that move through the Microsoft Exchange Server 2013/2016 organization. You can restrict the total size of a message or the size of the individual components of a message, such as the message header, the message attachments, and the number of recipients. You can apply limits globally for the whole Exchange organization, or specifically to a connector or user object. If limits applied to a connector are causing too many messages to be rejected, identifying the connector with the

Measurement	Description	Measurement Unit	Interpretation
			maximum rejections will lead administrators to that connector for which message size restrictions have either to be lifted or fine-tuned.

3.6.4 Delivery SMTP Send Connector Test

In Microsoft Exchange Server 2013/2016, a Send connector controls the flow of outbound messages to the receiving server. They are configured on Mailbox servers running the Transport service. Most commonly, you configure a Send connector to send outbound email messages to a smart host or directly to their recipient, using DNS.

The mailbox delivery Send connector exists in the Mailbox Transport service on every Mailbox server. This connector is implicitly created, invisible, and requires no management. The mailbox delivery Send connector is used to relay messages to the Transport service and the Mailbox Transport service on other Mailbox servers in the organization.

Problems in a delivery send connector therefore can obstruct the flow of messages to other mailbox servers, preventing critical business communication from reaching their destination and causing palpable revenue losses. If this is to be avoided, administrators should keep an eye on how each delivery send connector is processing the outbound messages, proactively detect potential slowdowns in mail delivery, and promptly capture message failures caused due to connector errors. This is where the **Delivery SMTP Send Connector** test helps. This test monitors each delivery send connector and reports the load on the connector, how quickly the connector processes its load, and how many errors of which type the connector encountered. This way, the test points to probable bottlenecks to email delivery and the connector responsible for them.

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each delivery send connector

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data sent rate:	Indicates the rate at which email data was sent by this connector.	KB/Sec	In the event of a slowdown, you can compare the value of this measure across connectors to know which connector is processing emails slowly.
Outbound connections from send connector:	Indicates the current number of outbound connections from this connector.	Number	Compare the value of this measure across connectors to know which connector is being used the most. Overloaded connectors can thus be identified.
Messages sent by send connector:	Indicates the rate at which messages are sent by this connector.	Msgs/Sec	A consistent drop in this value could indicate a bottleneck in message delivery.
Connection failures encountered by send connector:	Indicates the number of connection failures encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
DNS errors encountered by this send connector:	Indicates the number of DNS errors encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
Protocol errors encountered by send connector:	Indicates the number of protocol errors encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
Socket errors encountered by send connector:	Indicates the number of socket errors encountered by this send send connector.	Number	Ideally, the value of this measure should be 0.

3.6.5 Delivery Store Driver Agents Test

The Exchange Store Driver is a core transport component which lives the Mailbox server role. It is responsible for:

- Retrieving messages from the mailbox server that have been submitted by end-users, running the Hub Selector process (in order to select the best Transport service which could be local or another server), and forwarding the message to the Default Receive connector in the Transport service.
- Receiving processed messages from the Transport service and placing the message in the users Inbox, using RPC;

In addition to the above, the store driver also serves as an extensibility platform for both mail submission & delivery. Store Driver currently hosts a number of agents that extend the functionality of Exchange. Examples include such agents as Inbox Rules, Conversations, meeting forward notifications, etc. If any of these extended functionalities fail, then administrators should be instantly alerted to the failure along with information on which agent provided the functionality. The **Delivery Store Driver Agents** test does just that! This test auto-discovers the delivery agents hosted by the store driver and promptly notifies administrators if any of these agents fail too often. Problem-prone agents can thus be identified.

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each auto-discovery store driver agent on the Mailbox server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Store driver delivery agent	Indicates the percentage of time in the	Percent	If any agent fails almost 100% of the time, such an agent can be

Measurement	Description	Measurement Unit	Interpretation
failure:	last 5 minutes this delivery agent failed.		marked as problem- prone. The functionality provided by such agents can be further analyzed to identify failure points.

3.6.6 ygiene - Filtering Core Test

Messaging hygiene refers to the antivirus and antispam framework built into Microsoft Exchange Server.

Exchange 2013/2016 comes out of the box with basic built-in anti-malware protection designed to help organizations combat viruses and spyware in their e-mail messaging environment. This anti-malware feature scans emails in the transport pipeline for viruses, spyware, and malware in real-time, and deletes the messages and attachments found to be infected, so as to shield the mailbox from harm.

If this anti-malware filter takes too long to scan emails or experiences frequent crashes/failures, it will not only delay the flow of emails through the transport pipeline, but will also expose the Exchange environment to malicious virus attacks. To ensure that the Exchange environment stays healthy and protected against such unscrupulous attacks and unnecessary delays, administrators will have to keep a close watch on how the anti-malware filter functions. This is exactly what the **Hygiene – Filter Core** test does. This test tracks the requests to the anti-malware engine, monitors how quickly and efficiently the engine processes the scanning requests it receives, and in the process, proactively alerts administrators to potential delays and errors in filtering.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average scan time:	Indicates the time taken to scan requests.	Secs	A high value could indicate a bottleneck in scanning.
Average classification time per request:	Indicates the time taken to classify one scan request.	Secs	An unusually high value could indicate that request classification is taking longer than expected.
Crashed scan processes:	Indicates the number of scan processes that crashed in the last hour.	Number	Ideally, the value of this measure should be 0. A high value is a cause for concern as it indicates frequent scan crashes.
Running scan processes:	Indicates the number of scan processes currently running.	Number	This is a good indicator of the current workload of the anti-malware filter.
Scan requests error:	Indicates what percentage of scan requests submitted in the last minute encountered errors that prevented the processing of those scan requests.	Percent	This includes scan requests rejected, fatal errors and errors while processing. Ideally, this measure should report the value 0. A high value indicates that many scan requests have encountered errors and were hence not processes. This is a cause for concern and warrants an investigation.
Timed out scan requests:	Indicates the number of scan requests that timed out in the last minute.	Number	
Average wait time for scanned requests:	Indicates the average time for which a scan request waits in the internal queue.	Secs	A high value is indicative of a processing slowdown.
Scan requests	Indicates the number of	Processed/Sec	Ideally, the value of this measure

Measurement	Description	Measurement Unit	Interpretation
processed rate:	scan requests processed per second.		should be high. A consistent drop in this value could indicate a processing slowdown.
Scan requests in request queue:	Indicates the number of scan requests that are currently in the internal queue.	Number	
Scan requests submitted rate:	Indicates the number of scan requests submitted per second, including requests accepted and rejected by the scanning system.	Submitted/Sec	

3.6.7 Mailbox Assistants – Per Database Test

The Microsoft Exchange Mailbox Assistants service performs background processing of mailboxes in the Exchange store. It provides functionality for Calendar Attendant, Resource Booking Attendant, Out of Office Assistant, and Managed Folder Mailbox Assistant.

The Exchange Assistants can be either event-based Assistants or time-based Assistants. The event-based Assistants start to process mailboxes on the occurrence of an event, such as on a change of Out-of-Office (OOF) information in one or more mailboxes. The time-based Assistants process the mailboxes periodically. Each time-based Assistant deploys an Assistants Driver that periodically checks whether the current time is within a specified time window. When the current time reaches the specified time window, the Assistants Driver invokes the corresponding time-based Assistant. The time-based Assistant then obtains a list of mailboxes from the database and starts to process them.

Latencies in background processing can adversely impact a user's experience with his/her Exchange mailbox. If this is to be avoided, administrators should keep an eye on the activities of every assistant on each of the mailboxes it processes, isolate potential processing slowdowns, and identify the mailboxes that will be affected. This is where the **Mailbox Assistants – Per Database** test helps.

This test auto-discovers the mailbox assistants at work and the mailbox databases they are working on. For each assistant, the test reports the time taken by that assistant to process events and mailboxes, and thus reveals bottlenecks in processing.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every mailbox assistant per mailbox database

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap- in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.
5. **EXECUTION TIMEOUT** – Specify the duration (in seconds) for which this test will wait for a response from the server; beyond this duration, the test will timeout.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average event processing time:	Indicates the average time taken by this assistant for processing events chosen as interesting.	Secs	Ideally, the value of this measure should be less than 2 minutes (i.e., 120 seconds). A very high value indicates that there are many events in queue with long waiting times. This in turn indicates a processing bottleneck.
Average mailbox processing time:	Indicates the average processing time of	Secs	A low value is desired for this measure, as high values are

Measurement	Description	Measurement Unit	Interpretation
	mailboxes for time-based assistants.		indicative of delays in processing.
Time elapsed since the last databases update:	Indicates the time since the last attempt made by this assistant to update the list of databases.	Minutes	
Time elapsed since the last event poll:	Indicates the time elapsed (in seconds) since the last event was polled by this assistant for this database.	Secs	
Time elapsed since the last event poll attempt:	Indicates the time elapsed (in seconds) since the last attempt made by this assistant to poll events for this database.	Secs	
Number of events in queue:	Indicates the current number of events in the in-memory queue of this database waiting to be processed by this assistant.	Number	Ideally, the value of this measure should be low at all times. High values may indicate a performance bottleneck.
Number of mailboxes processed:	Indicates the number of mailboxes processed by this assistant.	Number	
Rate of mailboxes processed:	Indicates the rate at which this assistant processed mailboxes.	Processed/Sec	Ideally, the value of this measure should be high at all times. Low values may indicate a performance bottleneck.

3.6.8 Mailbox Transport Submission Service Test

The Mailbox Transport Submission service connects to the local mailbox database using RPC to retrieve messages, and submits the messages over SMTP to the Transport service on the local Mailbox server, or on other Mailbox servers. If this service fails to submit messages to the transport service, then messages will not be delivered to their destinations, resulting in loss of critical business communication and related revenues. To avoid this, administrators can use the **Mailbox Transport Submission Service** test to monitor the submissions made by mailbox transport submission service and promptly capture permanent/temporary submission failures.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Failed submissions rate:	Indicates the rate at which submissions failed.	Submissions/Sec	A high value for this measure is a cause for concern as it indicates frequent submission failures.
Permanent failed submissions:	Indicates the percentage of permanent submission failures.	Percent	A very low value is desired for this measure.
Successful submissions rate:	Indicates the rate at which submissions succeeded.	Submissions/Sec	A high value is desired for this measure.
Temporary submission rate:	Indicates the number of submissions per second that	Submissions/Sec	Ideally, the value of this measure should be low.

Measurement	Description	Measurement Unit	Interpretation
	experienced temporary failures.		

3.6.9 Submission Store Driver Agents Test

When a user attempts to send an email to a mailbox on the same Mailbox server or another Mailbox server, the Mailbox Transport Submission service on the source Mailbox server uses the Store Driver to connect to the mailbox database using RPC and retrieves the e-mail to be delivered. Many agents run within the store driver to facilitate email pickup and delivery. If any of these agents fail, it can result in the failure of or unprecedented delays in email delivery. If this is to be averted, administrators should track the status of each of the agents running within the store driver, promptly capture agent failures, and revive the agent before its failure impacts email delivery. For this, administrators can use the **Submission Store Driver Agents** test. This test monitors every agent running within the submission store driver and reports the percentage of failures encountered by the agent during the last 5 minutes. Besides alerting administrators to agent failures, this test also pinpoints error-prone agents.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each store driver submission agent

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Store driver submission agent failure:	Indicates the percentage of failures encountered by this agent in the last 5 minutes.	Percent	A value close to 100% is a cause for concern as it indicates continuous failure of the submission agent.

3.6.10 Submission Component Latencies Test

Submission is the process of putting messages into the Submission queue. Submission happens in three ways:

- From SMTP Receive through a Receive connector.
- Through the Pickup directory or the Replay directory. These directories exist on Mailbox servers and Edge Transport servers. Correctly formatted message files that are copied into the Pickup directory or the Replay directory are put directly into the Submission queue.
- Through a transport agent.

Regardless of how submission is performed, a slow down anywhere during the submission process can halt or delay the transmission of email messages to a mailbox. To ensure the uninterrupted delivery of emails, the components engaged in submission should be closely monitored and latencies experienced by components should be promptly detected. This is where the **Submission Component Latencies** test helps. This test auto-discovers all components engaged in mail submission. Then, at configured intervals, this test reports how long it took each component to process email messages, 90%, 95%, and 99% of the time. In the process, the test reveals latencies in message submission and accurately points to the components responsible for the same.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each submission component

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
90th percentile message latency:	Indicates the maximum time taken by this component to process messages 90% of the	Secs	If the value of these measures are consistently high for one/more submission components, it could indicate bottlenecks in mail

Measurement	Description	Measurement Unit	Interpretation
	time.		submission. The reason for the same has to be ascertained and resolved, so that there is an uninterrupted flow of emails end-to-end.
95th percentile message latency:	Indicates the maximum time taken by this component to process messages 95% of the time.	Secs	
99th percentile message latency:	Indicates the maximum time taken by this component to process messages 99% of the time.	Secs	

3.6.11 Submission SMTP Send Connector Test

The Mailbox Transport Submission service connects to the local mailbox database using RPC to retrieve messages, and submits the messages over SMTP to the Transport service on the local Mailbox server, or on other Mailbox servers. To select the best Transport service (which could be local or on another server) to submit the messages to and to actually forward the messages to the Default Receive Connector on the chosen Transport Service, the Mailbox Transport Submission services relies on Send Connectors. If one/more of these send connectors fail due to errors or operate at a lethargic pace, it is bound to severely impact the selection and submission of emails to the Transport service. If ignored, this will aggravate and adversely impact the timely delivery of mails to the destination mailboxes. To prevent this, administrators should monitor each SMTP send connector used the Mailbox Submission Service and proactively identify those send connectors that are either latent or error- prone. The **Submission SMTP Send Connector** test helps administrators achieve this. This test monitors every send connector used by the Mailbox Transport Submission service and reports the throughput of and errors encountered by each send connector. This way, the test accurately pinpoints send connectors that could be experiencing processing bottlenecks and the ones that are error-prone.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each submission send connector

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data sent rate:	Indicates the rate at which this send connector sends data.	KB/Sec	A consistent drop in the value of this measure could indicate a bottleneck when submitting messages. Compare the value of this measure across send connectors to know which connector is slowing down mail submission.
Outbound connections from send connector:	Indicates the number of outbound connections from this SMTP send connector.	Number	Compare the value of this measure across connectors to identify the busiest connector.
Messages sent by send connector:	Indicates the rate at which messages are sent by this send connector.	Msgs/Sec	A consistent decrease in the value of this measure is indicative of a slowdown when submitting messages. To know which send connector is the slowest, compare the value of this measure across connectors.
Connection failures encountered by connector:	Indicates the number of connection failures encountered by this send connector.	Number	Ideally, the value of this measure should be 0. A high value indicates serious connection problems, which need to be investigated right away. Compare the value of this measure across connectors to identify the connector with the maximum connection failures.

Measurement	Description	Measurement Unit	Interpretation
DNS errors encountered by connector:	Indicates the number of DNS errors encountered by this send connector.	Number	Ideally, the value of this measure should be 0. A high value indicates serious issues in name space resolution, which need to be investigated right away. Compare the value of this measure across connectors to identify the connector with the maximum number of DNS errors.
Protocol errors encountered by connector:	Indicates the number of protocol errors encountered by this send connector.	Number	Ideally, the value of this measure should be 0. A high value indicates serious protocol issues, which need to be investigated right away. Compare the value of this measure across connectors to identify the connector with the maximum number of protocol errors.
Socket errors encountered by connector:	Indicates the number of socket errors encountered by this send connector.	Number	Ideally, the value of this measure should be 0. A high value indicates serious socket issues, which need to be investigated right away. Compare the value of this measure across connectors to identify the connector with the maximum number of socket errors.

3.6.12 Classification Scan Engine Test

Data loss prevention (DLP) is an important issue for enterprise message systems because of the extensive use of email for business critical communication that includes sensitive data. In order to enforce compliance requirements for such data, and manage its use in email, without hindering the productivity of workers, DLP features make managing sensitive data easier than ever before.

DLP policies are simple packages that contain sets of conditions, which are made up of transport rules, actions, and exceptions that you create in the Exchange Administration Center (EAC) and then activate to filter email messages. One important feature of transport rules is a new approach to

classifying sensitive information that can be incorporated into mail flow processing. This new DLP feature involves a Classification Engine that performs deep content analysis through keyword matches, dictionary matches, regular expression evaluation, and other content examination to detect content that violates organizational DLP policies.

The Classification engine is also in charge of handling importing of new classification rules packages. These new classification rules packages allow administrators and independent service vendors to create packages to manage specific content. These customer packages are XML files that can be imported via the Exchange command shell. These packages will need to be encrypted to be imported into Exchange 2013/2016. The Microsoft Classification Engine is in charge of decrypting the packages.

Errors in the operations and delays in the loading/content processing of the classification engine can severely hamper the execution of transport rules and the detection of sensitive content in emails. If these problems are allowed to persist, classified information may reach the wrong hands, resulting in organizational mayhem. To avert this, you need to run the **Classification Scan Engine** test at periodic intervals, check for errors in the engine's operations, track the time taken by the engine to load and to scan the content, and capture errors and slow downs proactively.

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Classification engine errors:	Indicates the number of Classification engine errors in the last minute.	Number	Ideally, the value of this measure should be 0. A non-zero value is indicative of engine errors and will warrant immediate investigation.
Average engine	Indicates the average	Secs	A low value is desired for this

Measurement	Description	Measurement Unit	Interpretation
load time per load:	time taken by the engine to load.		measure. A consistent rise in this value is indicative of a bottleneck when loading.
Items scanned for classiciation rate:	Indicates the rate at which the content was scanned for DLP policy violations.	Processed/Sec	A steady drop in the value of this measure is indicative of a processing bottleneck on the engine.
Detected classified items:	Indicates the number of items that have been detected as classified.	Number	
Average classification scan time per item:	Indicates the time taken by the engine to scan the content and detect classified items.	Secs	A steady increase in the value of this measure is indicative of a processing bottleneck on the engine.

3.7 The Transport Services Layer

The Transport service performs email routing within the organization, and between the Front End transport service and the Mailbox Transport service. Using the tests mapped to this layer, administrators can proactively detect latencies in the operations of this service, analyze the impact of these latencies on transport components such as transport queues, and isolate the probable cause for these latencies – could it be owing to slowness experienced by the receive/send SMTP connectors? Could it be because of ineffective transport rules? Or could it be due to inefficient transport agents?

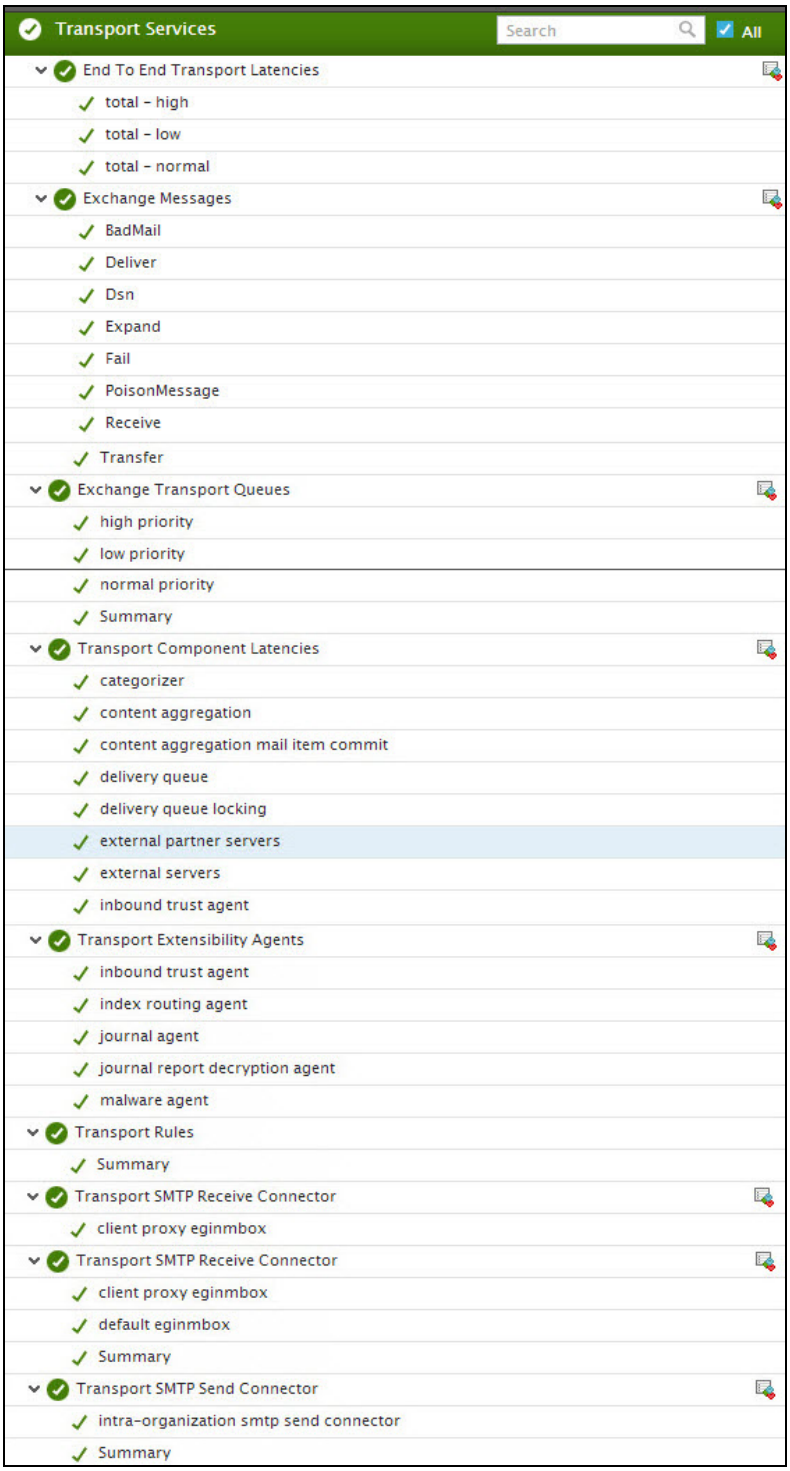

Figure 3.11: The tests mapped to the Transport Services layer

3.7.1 End to End Transport Latencies Test

Periodically, administrators need to monitor how long end-to-end message flow takes, so that probable slowdowns in mail delivery can be proactively detected and resolved, before users complain. To achieve this, administrators can use the **End to End Transport Latencies** test. This test monitors end-to-end mail flow, classifies latencies as high, medium, and low, and for each such classification, reports the maximum time taken by the server for transporting mails from one end to another, 90%, 95%, and 99% of the time. This way, the test leads administrators to current/probable bottlenecks in mail flow.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results each for high, medium, and low latencies

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
90th percentile message latency:	Indicates the maximum time taken for end-to-end email message flow 90% of the time.	Secs	If the value of these measures are consistently high, it could indicate bottlenecks in email flow. The reason for the same has to be ascertained and resolved, so that there is an uninterrupted flow of emails end-to-end. High latencies can be triggered by one or more of the following reasons: • A networking issue exists. • The Transport component of the

Measurement	Description	Measurement Unit	Interpretation
			Microsoft Exchange 2013/2016 is under a heavy load. • Some messages have been sent to large distribution groups. • A Transport Agent issue exists. For example, the antivirus agent or Rights Management Services agent may be experiencing an issue.
95th percentile message latency:	Indicates the maximum time taken for end-to-end email message flow 95% of the time.	Secs	
99th percentile message latency:	Indicates the maximum time taken for end-to-end email message flow 99% of the time.	Secs	

3.7.2 Exchange Transport Queues Test

A queue is a temporary holding location for messages that are waiting to enter the next stage of processing or delivery to a destination. Each queue represents a logical set of messages that the Exchange server processes in a specific order. In Microsoft Exchange Server 2013/2016, queues hold messages before, during and after delivery. Exchange 2013/2016 supports different types of queues, namely: Submission queue, Unreachable queue, Poison message queue, Delivery queue, Shadow queue, and Safety Net. By closely tracking the length of these queues and the speed with which the messages are processed by the queues, administrators can quickly detect bottlenecks in message processing and can identify the exact queue type where the bottleneck lies. To achieve this, administrators can use the **Exchange Transport Queues** test.

This test monitors the queues on the exchange server and discovers the priority of messages enqueued – whether, high, low, or normal. The message priority is typically assigned by the sender in Microsoft Outlook when the sender creates and sends the message. This priority setting affects the transmission of messages from a delivery queue to the destination messaging server. High priority messages are transmitted to their destinations before Normal priority messages, and Normal priority messages are transmitted to their destinations before Low priority messages.

Once the priority is discovered, this test then reports how many messages of each priority are enqueued in the various queues and how quickly the queues are processing messages of every priority. This not only enables administrators to proactively isolate potential message processing bottlenecks, but also helps them precisely identify the following:

- Where the bottleneck originated – i.e., in which type of queue
- What type of messages were being processed when the slowdown occurred – high priority? normal priority? Or low priority messages?

This in turn will enable administrators to define specific SLAs for the delivery time of the messages of each priority.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for messages of every priority

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Available job slots in categorizer:	Indicates the percentage of available job slots in categorizer for messages of this priority.	Percent	The categorizer in the Transport service on a Mailbox server picks up one message at a time from the Submission queue and performs recipient resolution, routing resolution, and content conversion on the message before putting the message in a delivery queue. Additionally, mail flow rules that are defined by the organization are applied. A high value is desired for this measure. A low value or a consistent drop in this value – particularly for high priority

Measurement	Description	Measurement Unit	Interpretation
			messages - is a cause of concern, as it indicates that the categorizer is very busy and may not have enough slots for subsequent messages. This will delay mail delivery, even for high priority messages.
External items queued for delivery in all queues:	Indicates the number of external items of this priority that are queued for delivery in all queues.	Number	A low value is desired for this measure. A high value indicates that many external messages are in queue awaiting delivery, probably owing to a bottleneck in mail flow. Compare the value of this measure across priorities to determine the priority of the largest number of external messages pending delivery.
External items in largest delivery queue:	Indicates the number of external items of this priority in the largest delivery queue.	Number	By comparing the value of this measure across priorities, you can determine the priority of the maximum number of external items in the largest delivery queue. If too many external items in the largest delivery queue are of a high priority, it indicates a serious bottleneck in mail flow which is causing even high priority messages to remain undelivered for long time periods.
External retry items in remote delivery queue:	Indicates the number of external items of this priority set for "retry" in	Number	Ideally, the value of this measure should be very low. A high value or a steady increase in this value

Measurement	Description	Measurement Unit	Interpretation
	the remote delivery queues.		could indicate that many messages are failing and are hence being retried. You can compare the value of this measure across priorities to know the priority of the maximum number of external messages set for retry.
Internal items queued for delivery in all queues:	Indicates the number of internal items of this priority that are queued for delivery in all queues.	Number	A low value is desired for this measure. A high value indicates that many internal messages are in queue awaiting delivery, probably owing to a bottleneck in mail flow. Compare the value of this measure across priorities to determine the priority of the largest number of internal messages pending delivery.
Internal items in largest delivery queue:	Indicates the number of internal items of this priority in the largest delivery queue.	Number	By comparing the value of this measure across priorities, you can determine the priority of the maximum number of internal items in the largest delivery queue. If too many internal items in the largest delivery queue are of a high priority, it indicates a serious bottleneck in mail flow which is causing even high priority messages to remain undelivered for long time periods.
Internal retry items in remote delivery queue:	Indicates the number of internal items of this priority set for "retry" in the remote delivery	Number	Ideally, the value of this measure should be very low. A high value or a steady increase in this value could indicate that many

Measurement	Description	Measurement Unit	Interpretation
	queues.		messages are failing and are hence being retried. You can compare the value of this measure across priorities to know the priority of the maximum number of internal messages set for retry.
Messages delivered rate:	Indicates the rate at which messages of this priority completed delivery.	Msgs/Sec	Consistent drops in the value of this measure is indicative of a slowdown in message delivery. Compare the value of this measure across priorities to know the messages of which priority were delivered most slowly. If a low value is reported for even high priority messages, it is serious issue as it indicates that the slowdown is impacting the delivery of high priority messages as well.
Messages completing categorization:	Indicates the percentage of messages of this priority that have completed categorization since the last measurement period.	Percent	A high value is desired for this measure. A low value is indicative of a bottleneck in categorization.
Messages deferred categorization:	Indicates the percentage of messages of this priority that are being deferred for client- side processing during categorization.	Percent	Exchange and Outlook work together to process messages through all applicable rules. Exchange first executes all the rule processing that is possible on the server and then, if further client-side processing is necessary, it creates a special message in the

Measurement	Description	Measurement Unit	Interpretation
			Deferred Actions folder. These messages are called deferred action messages (DAMs) and basically tell Outlook that it has to complete processing for a message. Outlook reads and executes the DAMs as Exchange creates them. Any DAMs that are accumulated when Outlook is offline are cleared the next time the client initializes. Since deferring can delay message delivery, a low value is desired for this measure.
Messages queued for delivery:	Indicates the number of messages of this priority that were queued to be delivered during the last measurement period.	Number	If the value of this measure only increases consistently and does not drop, it can only indicate that messages are not being delivered as fast as they are being queued for delivery. This indicates a delivery bottleneck.
Messages queued rate:	Indicates the rate at which messages of this priority are queued for delivery.	Msgs/Sec	
Messages submitted rate:	Indicates the rate at which messages of this priority are enqueued in the submission queue.	Msgs/Sec	The Submission queue is used by the categorizer to gather all messages that have to be resolved, routed, and processed by transport agents on the transport server. All messages that are received by a transport server enter processing in the

Measurement	Description	Measurement Unit	Interpretation
			Submission queue. On Mailbox servers, messages are submitted through a Receive connector, the Pickup or Replay directories, or the Mailbox Transport Submission service. On Edge Transport servers, messages are typically submitted through a Receive connector, but the Pickup and Replay directories are also available. If the value of this measure increases consistently but does not drop as quickly, it could indicate that categorizer is unable to keep pace with the message load on the submission queue. The bottleneck in this case is the categorizer.
Retry items in retry mailbox queue:	Indicates the number of items of this priority in retry in the retry mailbox queues.	Number	At any given point in time, the value of this measure should not exceed 100. If too many messages are in the retry mailbox queue, it indicates frequent message delivery failures. This is not a sign of good health and will have to be looked into.
Retry items in non-SMTP queues:	Indicates the number of messages of this priority that are in a retry state in the non-Simple Mail Transfer Protocol (SMTP) gateway delivery queues.	Number	Non-SMTP destinations also use delivery queues if the destination is serviced by a Delivery Agent connector. If too many messages are set for 'retry' in the non-SMTP delivery queues, it could indicate that too

Measurement	Description	Measurement Unit	Interpretation
			many messages could not be delivered to non- SMTP destinations.
Items in submission queue:	Indicates the number of messages of this priority in the submission queue.	Number	At any given point in time, the value of this measure should not exceed 100. If sustained high values are occurring, investigate Active Directory and Mailbox servers for bottlenecks or performance-related issues.
Items in unreachable queue:	Indicates the number of messages of this priority in the unreachable queue.	Number	The Unreachable queue contains messages that can't be routed to their destinations. These unreachable queues should not contain more than 100 messages at any given point in time. If this measure consistently reports a value over 100, it could imply that the routing path for delivery was altered by configuration changes.
Items in poison queue:	Indicates the number of messages of this priority in the poison queue.	Number	The poison message queue is a special queue that's used to isolate messages that are determined to be harmful to the Exchange 2013/2016 system after a transport server or service failure. Ideally, the value of this measure should be 0 at all time. Non-zero values could indicate the existence of messages that are genuinely harmful in their content and format.

Measurement	Description	Measurement Unit	Interpretation
			Alternatively, these messages may also be the results of a poorly written agent that has caused the Exchange server to fail when it processed the supposedly bad messages.

3.7.3 Exchange Queue Statistics Test

A queue is a temporary holding location for messages that are waiting to enter the next stage of processing or delivery to a destination. Each queue represents a logical set of messages that the Exchange server processes in a specific order. In Microsoft Exchange Server 2013, queues hold messages before, during and after delivery. Each queue in the Exchange Server may be in different states as mentioned in the table below:

Measure Value	Description
Active	The queue is actively transmitting messages.
Connecting	The queue is in the process of connecting to the next hop.
Ready	The queue recently transmitted messages, but the queue is now empty.
Retry	The last automatic or manual connection attempt failed, and the queue is waiting to retry the connection.
Suspended	The queue has been manually suspended by an administrator to prevent message delivery. New messages can enter the queue, and messages that are in the act of being transmitted to the next hop will finish delivery and leave the queue. Otherwise, messages won't leave the queue until the queue is manually resumed by an administrator.

Though the queues are accurate indicators of the health of mail flow, it is necessary to identify the state of the queue that is disrupting the flow of mails. Administrators should therefore, continuously track the messages flowing in and out of the queues in each state, so that they can promptly capture bottlenecks in mail flow, before end-users complain. This is exactly what the **Exchange Queue Statistics** test does.

This test automatically discovers the current state of all the queues on the Exchange system, reports the number of messages found in the queues for each state, and also reveals whether/not every

queue in that particular state is able to process and send out messages as quickly as it receives them. In the process, the test points to processing bottlenecks, the exact state of the queue in which the bottleneck has occurred, and which stage of mail flow that bottleneck affects.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each Exchange queue

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. 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
Message count:	Indicates the number of messages currently found in this queue.	Number	A consistent increase in the value of this measure could indicate that the queue is receiving messages, but is sending very few out. This could indicate a processing bottleneck. You may want to compare the value of this measure across queues to accurately identify the queue that consists of the maximum number of messages pending processing. From this, you can determine where exactly processing is bottlenecked.
Incoming message rate:	Indicates the rate at which this queue is receiving messages.	Msgs/Sec	
Outgoing message rate:	Indicates the rate at which this queue is receiving messages.	Msgs/Sec	
Slow drain status:	Indicates the number of messages that are entering the queues in this state faster than they are leaving the queues in this state.	Number	The drain rate is computed by subtracting the value of the Incoming message rate measure from the value of Outgoing message rate measure. If the resultant value is greater than 0, then this measure reports the value Fast. This means that the messages are leaving the queue faster than they are entering the queue. This could imply a bottleneck at the previous hop.

Measurement	Description	Measurement Unit	Interpretation
			If the resultant value is equal to 0, then this measure will report the value Balanced. This means that messages are leaving the queue as fast as they are entering the queue. This is also the value you willl see when the queue is inactive. If the resultant value is lesser than 0, then this measure will report the value Slow. This means that messages are entering the queue faster than they are leaving the queue. At a basic level, if this measure reports the value Fast or Balanced for a queue, it indicates that the queue is healthy and is efficiently draining. On the other hand, the value Slow for a queue indicates that the queue is not efficiently draining. However, before jumping to conclusions, you also need to consider the values of the Incoming message rate, Outgoing message rate, and Message count measures. For example, a queue for which the Drain status is Slow, Message count is high, Incoming message rate is high, and Outgoing message rate is low, then it's obvious that the queue is not draining properly. However, a queue with Drain status as Slow, but also reports very small values for Incoming message rate, Outgoing message rate, and Message count, does not indicate a problem with the queue.

Measurement	Description	Measurement Unit	Interpretation
Balanced drain status:	Indicates the current drain status of this queue.		
Fast drain status:	Indicates the number of messages that are leaving the queues in this state faster than they are entering the queues in this state.	Number	
Retry message count:	Indicates the number of retry messages in this queue.	Kbytes	In Microsoft Exchange Server 2013/2016, messages that cannot be successfully delivered are subject to a Retry, by means of which a renewed connection attempt is made with the destination. If the length of the retry queue is increasing, it implies that replied messages are unable to reach to next hop. The reason for the bottleneck has to be investigated, isolated, and resolved.
Locked message count:	Indicates the count of locked messages in queue.	Number	

3.7.4 Exchange Messages Test

This test tracks the flow of messages through an Exchange organization, and reports the number and size of messages that pertain to every key event type handled by the Exchange server. These types include the following:

Type	Description
SEND	A message sent by Simple Mail Transfer Protocol (SMTP) to a different server.
RECEIVE	A message received and committed to the database.

SUBMIT	A message submitted by an Exchange 2007/2010 computer that has the Mailbox server role installed to an Exchange 2007/2010 computer that has the Hub Transport server role or Edge Transport server role installed.
POISON	A message added to the poison message queue or removed from the poison message queue.
FAIL	Message delivery failed

Whenever a user complaints of not being able to send or receive mails, the metrics reported by this test and the detailed diagnosis information provided therein will enable administrators to accurately determine the current status of the email sent by the user.

If need be, administrators can configure this test to additionally report the total number of messages on the Exchange server and their total size, regardless of event type. Apart from the event types discussed above, this total will also include messages that belong to the following event types:

Type	Description
BADMAIL	A message submitted by the Pickup directory or the Replay directory that cannot be delivered or returned
DELIVER	A message delivered to a mailbox
DEFER	A message for which delivery was delayed
DSN	A message for which a delivery status notification (DSN) was generated
EXPAND	A distribution group was expanded
FAIL	Message delivery failed
REDIRECT	A message redirected to an alternative recipient after an Active Directory directory service lookup
RESOLVE	A message for which recipients were resolved to a different e-mail address after an Active Directory lookup
TRANSFER	Recipients were moved to a forked message because of content conversion, message recipient limits, or agents

Exchange administrators can use this total to accurately assess the overall message traffic on the server and the ability of the server to handle the inflow/outflow of messages.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each of the following event types: SEND, RECEIVE, FAIL, POISON, SUBMIT

Configurable parameters for the test

Parameter	Description
Test Period	How often should the test be executed.
Host	Indicates the IP address of the Mailbox server.
Port	The port number through which the server communicates.
XChgExtensionShellPath	The Exchange Management Shell is a command-line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap-in (exshell.psc1) for script execution. This is why, the XChgExtensionShellPath is set to *none* by default.
AllEvents	By default, this flag is set to **No**, indicating that this test will report metrics for only the following event types by default: SEND, RECEIVE, SUBMIT, FAIL, POISON. If you want the test to additionally report metrics across all event types – i.e., support an additional All descriptor, which will report the total number of emails handled by the server and their total size – then, set this flag to **Yes**.
DDForReceiveMessage	In large, highly active Exchange environments, hundreds of emails may be received by the Exchange server within a short period of time. In such environments, the frequent collection of detailed diagnosis of the received emails may increase the processing overheads of the eG agent, and may even choke the eG database. To avoid this, the DDForReceiveMessage flag is set to **No** by default; this implies that the test will not provide the detailed diagnosis for the RECEIVE descriptor – i.e., for the received messages – by default. To view detailed diagnosis for these messages as well, set this flag to **Yes**.
DDForSendMessage	In large, highly active Exchange environments, hundreds of emails may be sent by the Exchange server within a short period of time. In such environments, the frequent collection of detailed diagnosis information related to the sent emails may increase the processing overheads of the eG agent, and may even choke the eG database. To avoid this, the DDForSendMessage flag is set to **No** by default; this implies that the test will not provide the detailed diagnosis for the SEND descriptor – i.e., for the sent messages – by default. To view detailed diagnosis for these messages as well, set this flag to **Yes**.
DDForSubmitMessage	In large, highly active Exchange environments, hundreds of emails may be

Parameter	Description
	submitted to the transport pipeline within a short period of time. In such environments, the frequent collection of detailed diagnosis information related to the submitted emails may increase the processing overheads of the eG agent, and may even choke the eG database. To avoid this, the DDForSubmitMessage flag is set to **No** by default; this implies that the test will not provide the detailed diagnosis for the SUBMIT descriptor – i.e., for the sent messages – by default. To view detailed diagnosis for these messages as well, set this flag to **Yes**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of emails	Indicates the number of emails of this event type detected during this measurement period	Number	By default, this measure provides detailed diagnosis for the FAIL and POISON messages only. Using the detailed diagnosis of these descriptors, you can view the complete details of the failed and poison messages. Optionally, users can turn on detailed diagnosis generation for the RECEIVE, SEND, and SUBMIT messages as well, so as to view the complete details of such messages. The All descriptor, even if displayed, will not provide detailed diagnosis information.
Total Traffic	Indicates the total size of messages of this event type, during the last measurement period.	MB	Since the value of this measure includes the size of attachments, an unusually high value could indicate that one/more messages carry large attachments. A high value could also indicate the availability of a large number of messages of a particular type.
Internal e-mails	Indicates the total number of email messages of this event type sent/received	Number	This measure is applicable only for SEND and RECEIVE event types.

Measurement	Description	Measurement Unit	Interpretation
	within the organization during the last measurement period.		
Internal e-mail size	Indicates the total size of email messages of this event type sent/received within the organization during the last measurement period.	MB	This measure is applicable only for SEND and RECEIVE event types.
External e-mails	Indicates the total number of messages of this event type sent/received from an external domain during the last measurement period.	Number	This measure is applicable only for SEND and RECEIVE event types.
External e-mail size	Indicates the total size of messages of this event type sent/received from an external domain during the last measurement period.	MB	This measure is applicable only for SEND and RECEIVE event types.

3.7.5 Transport Rules Test

Using transport rules, you can look for specific conditions in messages that pass through your organization and take action on them. Transport rules let you apply messaging policies to email messages, secure messages, protect messaging systems, and prevent information leakage.

The basic workflow for transport rules is as follows:

- You use the Exchange admin center (EAC),the Shell, or a DLP policy to create a transport rule. After you create your rule, it is stored in Active Directory.
- As messages go through the transport pipeline, the Transport rules agent is invoked. The Transport rules agent is a special Transport agent that processes the Transport rules you create.
- The Transport rules agent evaluates the message, and if the message fits the conditions you specify in a transport rule, it takes the specified action on that message based on the mode of the rule.

If these transport rules are not configured properly, then the rules agent will report errors during evaluation, causing messages to be deferred. This in turn will disrupt mail flow and adversely impact

end-user experience with the mail server. To ensure uninterrupted mail flow, administrators will have to capture these errors promptly and reconfigure transport rules rapidly. For this, administrators can use the **Transport Rules** test. This test monitors transport rules and instantly notifies administrators if messages are deferred owing to evaluation errors. This way, the test prompts administrators to quickly reconfigure/remove/disable transport rules so as to avert message deference.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Messages deferred due to rule evaluation errors:	Indicates the number of messages that were deferred due to errors in rule evaluation.	Number	A high value indicates that one/more transport rules are poorly configured are hence causing many errors during evaluation. You need to identify such transport rules and reconfigure them, remove them, or disable them to reduce the value of this measure drastically.
Messages deferred rate:	Indicates the rate at which messages were deferred owing to errors in rule evaluation.	Msgs/Sec	A consistent increase in this value is a cause for concern, as it indicates that transport rules are throwing errors during evaluation causing frequent deference of messages.

3.7.6 Transport Component Latencies Test

Transport bottlenecks are one of the common reasons for poor user experience with the Exchange server. Frequent / prolonged breaks in email message flow end- to- end often results in delays/failures in the delivery of business- critical emails, thus impacting revenues, affecting user productivity, and escalating support costs. To ensure the speedy delivery of mails, latencies in mail flow should be proactively spotted and resolved. The **Transport Component Latencies** test helps with this. This test auto- discovers all transport components engaged in mail flow. Then, at configured intervals, this test reports how long it took each component to process email messages, 90%, 95%, and 99% of the time. In the process, the test reveals latencies in message processing and accurately points to the transport components responsible for the same.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each transport component engaged in mail flow

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
90th percentile message latency:	Indicates the maximum time taken by this transport component to process messages 90% of the time.	Secs	If the value of these measures are consistently high for one/more transport components, it could indicate bottlenecks in email flow. The reason for the same has to be ascertained and resolved, so that there is an uninterrupted flow of emails end-to-end. High latencies can be triggered by one or more of the following reasons:

Measurement	Description	Measurement Unit	Interpretation
			• A networking issue exists. • The Transport component of the Microsoft Exchange 2013/2016 is under a heavy load. • Some messages have been sent to large distribution groups. • A Transport Agent issue exists. For example, the antivirus agent or Rights Management Services agent may be experiencing an issue.
95th percentile message latency:	Indicates the maximum time taken by this transport component to process messages 95% of the time.	Secs	
99th percentile message latency:	Indicates the maximum time taken by this transport component to process messages 99% of the time.	Secs	

3.7.7 Transport Extensibility Agents Test

Transport agents let you install custom software that is created by Microsoft, by third-party vendors, or by your organization, on an Exchange server. This software can then process email messages that pass through the transport pipeline. If users complain of delays when receiving or sending mails over the Exchange server, it could be owing to processing bottlenecks with any transport agent. To confirm this, administrators can use the **Transport Extensibility Agents** test. For every transport agent, this test reports the time taken by the agent to process email messages and the time for which the agent was utilizing the CPU. This clearly pinpoints agents where processing is bottlenecked and those that are hogging CPU resources. If mail flow slows down, such agents can be held responsible for it.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each transport agent

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average agent processing time:	Indicates the time taken by this agent to process email messages.	Secs	Ideally, the value of this measure should be low. A high value or a consistent increase in the value of this measure is indicative of a processing bottleneck with the agent.
Average CPU time taken for asynchronous invocation of agent:	Indicates the total time spent in CPU by the synchronous part of asynchronous agents.	Secs	This measure does not give the CPU usage for work that is outside the first synchronous invoke. By comparing the value of this measure across agents, you can determine which agent's asynchronous invocations are hogging CPU.
Average CPU time taken for synchronous invocation of agent:	Indicates the total time spent in CPU by synchronous invocations of this agent.	Secs	By comparing the value of this measure across agents, you can determine which agent's synchronous invocations are hogging CPU.

3.7.8 Transport SMTP Receive Connector Test

Exchange 2013/2016 servers running the Transport service require Receive connectors to receive messages from the Internet, from email clients, and from other email servers. If a mailbox user complains that he/she is unable to receive certain messages or is receiving messages slowly, very often the source of such problems would be the receive connectors. Administrators should hence be able to figure out instantly which receive connector could probably be contributing to this problem condition and why – could it be because the receive connector is getting choked owing to an overload? Or is the connector rejecting too many messages owing to a size limit violation? The **Transport SMTP Receive Connector** test provides accurate answers to these questions. This test auto-discovers the SMTP Receive Connectors and for each connector reports the load on that

connector and the number of messages rejected by that connector. This way, the test points to connectors that are overloaded and those that are rejecting too many messages.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each Transport SMTP receive connector

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data received rate:	Indicates the rate at which data was received by this connector.	KB/sec	A high value over a period is indicative of an excessive use of the SMTP server. Under such circumstances, compare the value of this measure across receive connectors to know which connector is contributing to the heavy load on the SMTP server.
Inbound connections to the receive connector:	Indicates the number of inbound connections to this receive connector.	Number	The value of this measure indicates the connection load imposed by the send connectors of the Front End Transport service and the Transport service on the same Exhange 2013/2016 server, and the Transport and Mailbox Transport services on external Mailbox servers. In the event of a connection overload, you may want to compare the value of this measure across receive

Measurement	Description	Measurement Unit	Interpretation
			connectors, to know to which receive connector the send connectors are connecting often.
Messages received by receive connector:	Indicates the rate at which messages are received by this receive connector.	Msgs/Sec	Compare the value of this measure across receive connectors to know which receive connector is receiving a steady flow of messages. In the event of an overload, this measure will lead you to the exact connector that could be experiencing the overload.
Messages rejected due to size restriction:	Indicates the number of messages rejected by this connector because of a size restriction.	Number	If the message quota of a receive connector is set too low, then you will have too many messages to the connector getting rejected. To avoid this, you may want to reconfigure the message size quota of the connector with the highest rejection quotient. To identify the connector, compare the value of this measure across receive connectors.

3.7.9 Transport SMTP Send Connector

In Microsoft Exchange Server 2013/2016, a Send connector controls the flow of outbound messages to the receiving server. Exchange 2013/2016 Mailbox servers running the Transport service require Send connectors to deliver messages to the next hop on the way to their destination. Slow and error-prone connectors can therefore obstruct the timely delivery of mails to the end point. This is why, it is imperative that administrators identify such problematic connectors quickly, clear the bottleneck rapidly, and ensure that email delivery is not impeded in any way. This is where the **Transport SMTP Send Connector** test helps. This test auto-discovers the send connectors of the Transport service, and for each connector, reports the message load on that connector, reveals how well the connector handles the load, captures errors/failures experienced by the connector, and thus points administrators to those connectors that are unable to handle the load or are error-prone. In the process, the test enables administrators to figure out where email delivery is bottlenecked.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each Transport SMTP send connector

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Data sent rate:	Indicates the rate at which data was sent by this connector.	KB/sec	A high value over a period is indicative of an excessive use of the SMTP server. Under such circumstances, compare the value of this measure across send connectors to know which connector is contributing to the heavy load on the SMTP server.
Outbound connections from the send connector:	Indicates the number of outbound connections from this send connector.	Number	The value of this measure indicates the connection load imposed by the send connectors of the Transport service on the next hop en-route to the destination. In the event of a connection overload, you may want to compare the value of this measure across send connectors, to know which send connectors have established the maximum number of connections.
Messages sent by send connector:	Indicates the rate at which messages are sent by this send	Msgs/Sec	Compare the value of this measure across send connectors to know which send connectors are

Measurement	Description	Measurement Unit	Interpretation
	connector.		processing messages very slowly. In the event of a slowdown, this measure will lead you to the exact connector that could be contributing to the slowdown.
Connection failures encountered by send connector:	Indicates the number of connection failures currently encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
DNS errors encountered by send connector:	Indicates the number of DNS errors currently encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
Protocol errors encountered by send connector:	Indicates the number of protocol errors currently encountered by this send connector.	Number	Ideally, the value of this measure should be 0.
Socket errors encountered by the send connector:	Indicates the number of socket errors currently experienced by this send connector.	Number	Ideally, the value of this measure should be 0.

3.8 The HTTP Proxy Layer

The tests mapped to this layer monitor the availability, responsiveness, and overall performance of the IIS web server on which Exchange 2013/2016 operates. In addition, these tests:

- Measure the efficiency of the proxy functionality of the CAS server,
- Monitor cache usage and assess its impact on the proxy functionality;
- Pinpoint application pools with poor processing power

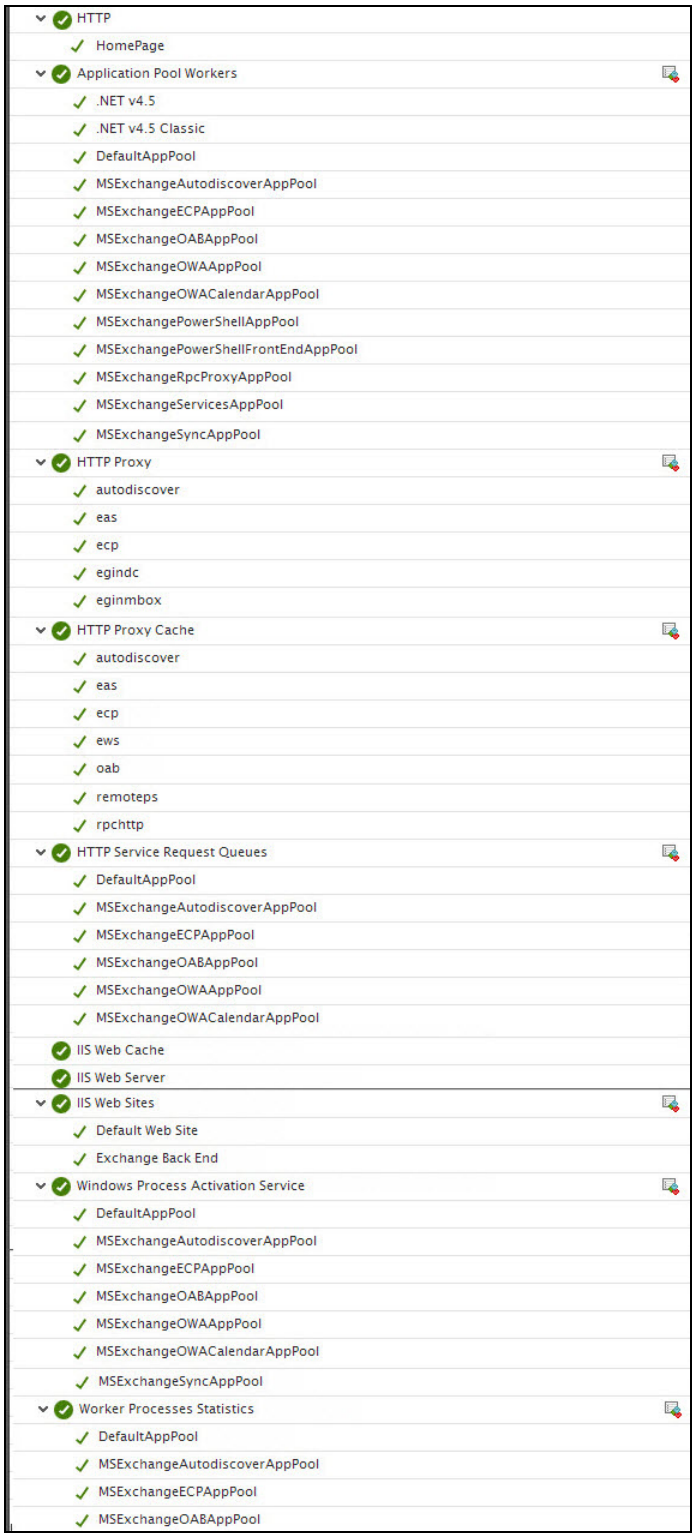

Figure 3.12: The tests mapped to the HTTP Proxy layer

3.8.1 HTTP Proxy Test

The Client Access Server manages client connections through redirection and proxy functionality. The Client Access server authenticates client connections and, in most cases, will proxy a request to the Mailbox server that houses the currently active copy of the database that contains the user's mailbox. In some cases, the Client Access server might redirect the request to a more suitable Client Access server, either in a different location or running a more recent version of Exchange Server.

Many CAS protocols / services are proxy enabled – for eg., Autodiscover, Outlook Web App, Exchange ActiveSync, EAS, ECP, etc.

If a user accessing a proxy-enabled service on CAS complains of a slowdown, then Exchange administrators should be able to identify the exact service that is being affected, and where the CAS processing is bottlenecked – during authentication? When proxying? When making MailboxServerLocator calls? When connecting to the Mailbox server? The **HTTP Proxy** test provides answers to these questions. This test auto-discovers the proxy-enabled services on CAS and reports the time spent by CAS at various stages of processing the requests to each service. In the process, the test accurately pinpoints the latent service and what is causing the latency.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every proxy-enabled service on CAS

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average latency of all AD requests:	Indicates the average latency of AD requests made by CAS for this proxy-enabled service.	Secs	Compare the value of this measure across services to know which service experienced the maximum latency when interacting with

Measurement	Description	Measurement Unit	Interpretation
			Active Directory.
Average latency spent authenticating CAS requests:	Indicates the average time spent by CAS when authenticating requests to this service over the last 200 samples.	Secs	By comparing the value of this measure across services, you can pinpoint that service for which CAS took too much time to perform authentication.
Average latency of CAS processing time:	Indicates the time spent by CAS processing requests to this service.	Secs	This processing time does not include proxying time. Compare the value of this measure across services to identify that service for which processing was delayed.
Average latency of proxy requests:	Indicates the time taken by CAS for proxying requests to this service.	Kbytes	Compare the value of this measure across services to know which service was delivered slowly to users owing to bottlenecks in proxying.
Average latency to resolve tenants:	Indicates the average time required to resolve tenants over the last 200 samples. This includes Global Locator Service (GLS) lookups. (Applicable For Exchange Online Only)	Secs	Compare the value of this measure across services to know which service was delivered slowly to users owing to bottlenecks in tenant resolution.
Data received rate:	Indicates the rate at which data was received by CAS for this service.	KB/Sec	
Data sent rate:	Indicates the rate at which data was sent by CAS for this service.	KB/Sec	

Measurement	Description	Measurement Unit	Interpretation
Connectivity related errors:	Indicates the percentage of connectivity failures that occurred between CAS and the Mailbox server when processing requests for this service.	Percent	A high percentage is a cause for concern. In the event of a slowdown in proxying, a very high value for this measure could indicate that connectivity failures are reason for the delay in proxying.
Average latency of MailboxServerLocator web service calls:	Indicates the time taken by CAS to make MailboxServerLocator calls for this service.	Secs	Compare the value of this measure across services to know which service was delivered slowly to users owing to bottlenecks in MailboxServerLocator calls.
MailboxServerLocator calls made rate:	Indicates the rate at which CAS made MailboxServerLocator calls for this service.	Calls/Sec	
MailboxServerLocator calls failure rate:	Indicates the percentage of failed MailboxServerLocator calls over the last 200 requests for this service.	Percent	By comparing the value of this measure across services to know which service suffered the most owing to MailboxServerLocator call failures.
Latency of MailboxServerLocator last call:	Indicates the latency of the last MailboxServerLocator call for this service.	Secs	
Retried MailboxServerLocator calls:	Indicates the percentage of MailboxServerLocator calls that were retried for this service over the last 200 requests.	Percent	By comparing the value of this measure across services to know which service suffered the most owing to retried MailboxServerLocator calls.

Measurement	Description	Measurement Unit	Interpretation
Concurrent outstanding proxy requests:	Indicates the number if concurrent outstanding proxy requests for this service.	Number	A high value is a cause for concern as it indicates too many proxy requests are pending processing. You can compare the value of this measure across services to know which service has the maximum number of outstanding proxy requests.
Proxy requests processed rate:	Indicates the number of proxy requests processed each second for this service.	Reqs/Sec	A consistent drop in the value of this measure for any service is indicative of bottlenecks in processing proxy requests.
Requests processed rate:	Indicates the number of requests processed each second for this service, which may not involve proxying to a Mailbox server.	Reqs/Sec	

3.8.2 HTTP Service Request Queues Test

By monitoring request queues to every Exchange application pool, administrators can identify those application pools that have too many requests pending and those with a high request rejection rate. This is exactly what the **HTTP Service Request Queues** test does. This test auto-discovers the application pools and reports the length of request queues and rejection rate of requests in queue for each application pool. This way, the test sheds light on those application pools that suffer from processing pains.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every application pool on Exchange

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed

2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Requests arrival rate:	Indicates the rate at which requests are arriving in the request queue of this application pool.	Reqs/Sec	A high rate could indicate a probable request overload on an application pool.
Requests in queue:	Indicates the current number of requests in the queue of this application pool.	Number	If this value increases consistently, it could indicate that the application pool is not processing requests quickly. Compare the value of this measure across pools to identify the pool with a processing bottleneck.
Requests rejected from queue:	Indicates the number of requests in this application pool's queue that were rejected.	Number	A non-zero value is desired for this measure.
Cache hit rate:	Indicates the rate of cache hits from the queue of this application pool.	Hits/sec	
Request rejection rate:	Indicates the rate at which this application pool rejected requests in the queue.	Reqs/Sec	Compare the value of this measure across pools to know which pool rejects queued requests frequently.
Maximum queue item age:	Indicates the age of the oldest request in the queue.	Reqs/Sec	

3.8.3 HTTP Proxy Cache Test

How well the proxy-enabled services such as OWA, Autodiscover, etc., proxy requests relies to a large extent on how the Exchange server caches are sized and utilized. Using the **HTTP Proxy Cache** test, administrators can monitor how each of these services use their caches, measures the size of every cache, and reports abnormalities in usage and size. This way, the bottlenecks to service performance are highlighted and a speedy resolution is enabled.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for each proxy-enabled service on the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
AnchorMailbox cache hit ratio:	Indicates the percentage of subscription requests to this service which resulted in AnchorMailbox cache hits since process start.	Percent	An X-AnchorMailbox is an HTTP header that is included in the initial subscription request from a client. It identifies the first mailbox in a group of mailboxes that share affinity with the same Mailbox server. Examples of AnchorMailbox identifiers are: SID, SMTP, Address, Organization name, Domain name, etc. This header is essential to ensure affinity – i.e., to ensure the association of a sequence of request and response messages

Measurement	Description	Measurement Unit	Interpretation
			with a particular Mailbox server. The client triggers the creation of a cookie by including this header in the subscription request. The subscription response then contains the cookie. The client then sends that cookie in subsequent requests to ensure that the request is routed to the right Mailbox server. A very low value for this measure indicates that the AnchorMailbox cache is utilized poorly. This could be owing to insufficient cache size, which typically renders the cache unable to hold many objects for lookup. Ineffective cache usage often leads to significant delays in request processing and adds to the processing overheads.
Size of AnchorMailbox cache:	Indicates the current size of the AnchorMailbox cache of this service.	KB	A consistent rise in the value of this measure indicates that the cache is getting filled up with frequently-accessed objects. To ensure that these objects do not run out of space, you may have to allocate more space to the cache, increase the maximum size limit upto which the cache can grow, or remove unused items from the cache regularly.
AnchorMailbox to MailboxServer cookie hits:	Indicates the percentage of subscription requests to	Percent	A high value is desired for this measure. A low value is indicative of very low cookie hits, which in turn

Measurement	Description	Measurement Unit	Interpretation
	this service which leveraged the BEServer cookie to avoid database lookups since the process was restarted.		implies that many costly database lookups have occurred.
Database GUID MailboxServer cache hit ratio:	Indicates the percentage of requests to this service which resulted in DbGuid->MbxServer cache hits since process start.	Percent	A high value is desired for this measure.
Backend server cache refreshing queue length:	Indicates the length of the back end server cache refreshing queue of this service.	Number	
Back end server cache size:	Indicates the current size of the back end cache of this service.	KB	A consistent rise in the value of this measure indicates that the cache is getting filled up with frequently-accessed objects. To ensure that these objects do not run out of space, you may have to allocate more space to the cache, increase the maximum size limit upto which the cache can grow, or remove unused items from the cache regularly.
FBA module key cache hit ratio:	Indicates the percentage of requests for this service that resulted in FBA module key cache hits.	Percent	Forms-based authentication (FBA) enables a sign- in page for Exchange Server 2013/2016 Outlook Web App that uses a cookie to store a user's encrypted sign-in credentials in the

Measurement	Description	Measurement Unit	Interpretation
			Internet browser. Tracking the use of this cookie enables the Exchange server to monitor the activity of Outlook Web App sessions on public and private computers. If a session is inactive for too long, the server blocks access until the user re-authenticates. A high value is desired for this measure.
Size of FBA module key cache:	Indicates the current size of the FBA module key cache.	KB	A consistent rise in the value of this measure indicates that the cache is getting filled up with frequently-accessed objects. To ensure that these objects do not run out of space, you may have to allocate more space to the cache, increase the maximum size limit upto which the cache can grow, or remove unused items from the cache regularly.
Overall cache effectiveness:	Indicates the percentage of requests that leveraged client cookies or in- memory cache or both to avoid costly lookups since the process was started.	Percent	Ideally, the value of this measure should be over 80%. If not, check the values reported by the AnchorMailbox cache hit ratio, AnchorMailbox to MailboxServer cookie hits,DatabaseGUID to MailboxServer cache hit ratio, and FBA module key cache hit ratio measures to know which cache is poorly utilized and why.

3.9 The Frontend Transport Layer

The Frontend Transport Service acts as a stateless proxy for all inbound and (optionally) outbound external SMTP traffic for the Exchange 2013/2016 organization.

Using the tests mapped to this layer, administrators can promptly isolate probable processing bottlenecks with this service and can precisely pinpoint where the bottleneck lies.

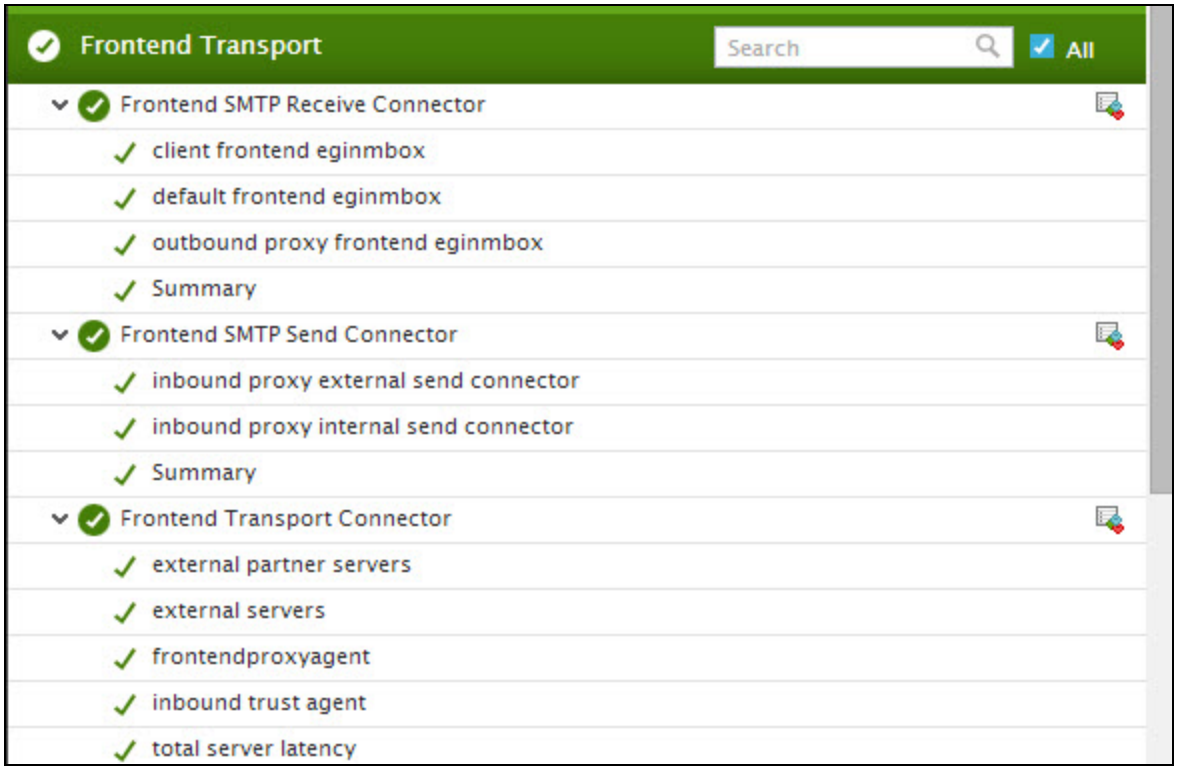

Figure 3.13: The tests mapped to the Frontend Transport Layer

3.9.1 FrontEnd Transport Connector Test

The Front End Transport Service service runs on all Client Access servers and acts as a stateless proxy for all inbound and outbound external SMTP traffic for the Exchange 2013/2016 organization. For outgoing messages, the Transport service on Mailbox servers uses Send connectors to communicate with the Receive connector of a Front End Transport service. The Receive connector then selects a Mailbox server based on the number and type of recipients and the proximity of the AD site and routes the message to that Mailbox server.

For incoming messages, the Receive connector on the Front End Transport service looks for a single healthy Transport service on a Mailbox server to receive the message. Then, via the Send

connector on the Front End Transport service it routes the message to the Transport service on that Mailbox server.

If an FET connector (Receive/Send) takes too long to select the Mailbox server or route messages to a Mailbox server, the delivery of incoming/outgoing emails is bound to be delayed. This is why, administrators need to keep track of how quickly each FET connector processes emails and identify that connector(s) that is highly latent. This is possible using the **FrontEnd Transport Connector** test. For every default and user-configured FET connector, this test reports the maximum time that connector took to proxy emails 90%, 95%, and 99% of the time. This will lead administrators to that connector which is most latent.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results each for every Frontend Transport Connector on the CAS server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
90th percentile message latency:	Indicates the maximum time taken by this connector to proxy emails 90% of the time.	Secs	If the value of these measures is very high for a particular connector, then that connector is deemed to be highly latent. You may want to look into the configuration of that connector and figure out why it is so slow.
95th percentile message latency:	Indicates the maximum time taken by this connector to proxy emails 95% of the time.	Secs	
99th percentile message latency:	Indicates the maximum time taken by this connector to proxy emails 99% of the time.	Secs	

3.9.2 FrontEnd SMTP Receive Connector Test

The FrontEnd SMTP Receive Connector controls the flow of inbound external SMTP traffic. When a message comes into the Exchange 2013/2016 organization from an external domain (say, the internet), the Receive connector looks for a single healthy Transport service on a Mailbox server to receive the message. Likewise, outbound messages to the internet too can routed by the Transport service on the Mailbox server to the Receive connector of the FET for Mailbox selection and delivery.

In the real world, if a single Receive connector is flooded with more messages than it can handle, delivery queues may grow longer, email delivery to internal recipients may slowdown, and critical business correspondence may not reach their destination on time. As a result, enterprises may miss out on lucrative business opportunities and related revenues. To avoid this, administrators should track the load on each FET Receive connector continuously, capture potential overload conditions and isolate the affected connectors, well before the business is impacted. This is where the **FrontEnd SMTP Receive Connector** test helps. For each FET Receive Connector, this test reports the count of recipients and connections handled by that connector. In the process, the test proactively captures those connectors that will probably be overloaded with messages/connections in a short while.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every Frontend SMTP Receive Connector on the target server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of recipients per inbound message received:	Indicates the average number of recipients per inbound message.	Number	Typically, the type and number of recipients determines how an FET connector operates.

Measurement	Description	Measurement Unit	Interpretation
			• For messages with a single mailbox recipient, the FET connector selects a Mailbox server in the target delivery group and gives preference to the Mailbox server based on the proximity of the AD site; • For messages with multiple mailbox recipients, the connector uses the first 20 recipients to select a Mailbox server in the closest delivery group, based on the proximity of the AD site. • If the message has no mailbox recipients, select a random Mailbox server in the local AD site. A high value of this measure is therefore indicative of the average workload of the connector.
Number of new SMTP connections created:	Indicates the rate at which new connections were established by this connector to the SMTP server.	Conns/Sec	A steady increase in this value is indicative of an increase in workload.
Data received rate:	Indicates the rate at which data is received by this connector.	KB/Sec	An above normal value for these measures over a period may indicate that the connector is overloaded.

Measurement	Description	Measurement Unit	Interpretation
Inbound messages received rate:	Indicates the number of messages (sent inbound into the forest) received by this connector every second.	Recv/Sec	
Connections to SMTP server:	Indicates the current number of connections to this connector.	Number	This is a good indicator of how busy the connector currently is.

3.9.3 FrontEnd SMTP Send Connector Test

To manage outbound external SMTP traffic, a FrontEnd SMTP Send Connector can be configured. If email delivery to external domains or to the internet takes too long, it could be because of a processing bottleneck at the FET SMTP Send Connector. By periodically evaluating the efficacy of each FET Send Connector configured, probable connector overloads and potential delivery bottlenecks can be captured early and eliminated. The **FrontEnd SMTP Send Connector** test helps with this. This test keeps track of the load on each connector, measures the message processing ability of the connector, and pinpoints those connectors where processing is bottlenecked.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every Frontend SMTP Send Connector on the CAS server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average recipients per message:	Indicates the average number of recipients handled by this send connector.	Number	Typically, the type and number of recipients determines how an FET connector operates. • For messages with a single mailbox recipient, the FET connector selects a Mailbox server in the target delivery group and gives preference to the Mailbox server based on the proximity of the AD site; • For messages with multiple mailbox recipients, the connector uses the first 20 recipients to select a Mailbox server in the closest delivery group, based on the proximity of the AD site. • If the message has no mailbox recipients, select a random Mailbox server in the local AD site. A high value of this measure is therefore indicative of the average workload of the connector.
Outbound connections established rate:	Indicates the rate at which outbound connections were established by this connector to the external SMTP server.	Conns/Sec	A steady increase in this value is indicative of an increase in workload.
Data sent rate:	Indicates the rate at	KB/Sec	A consistent drop in these values

Measurement	Description	Measurement Unit	Interpretation
	which data is sent by this connector.		could indicate a bottleneck in message transmission.
Messages sent rate:	Indicates the number of messages sent by this connector every second.	Sent/Sec	

3.9.4 Mailbox Folders Test

A mailbox database is a unit of granularity where mailboxes are created and stored. Every mailbox stores email messages, tasks, calendars, and other information pertaining to a specific mailbox owner in pre-configured folders – e.g., Inbox, Deleted Items, Sent Items, etc. – available in his/her mailbox. In addition, a mailbox owner can also create custom folders for storage.

When the size of a folder grows, the size of the corresponding mailbox and the mailbox database also grows. If any mailbox grows beyond control, the Exchange server will no longer be able to send mails from or deliver mails to that mailbox. In business-critical environments, such disruptions to the flow of emails can result in the loss of critical business communication, which in turn can lead to significant loss of revenues and reputation. If this is to be avoided, administrators should not only be able to identify those mailboxes and databases that are growing at a dangerous pace, but should also be able to pick the exact folders in those mailboxes that could be contributing to this abnormal growth. This is where the **Mailbox Folders** test helps.

This test auto-discovers the folders in each mailbox, and for each folder, reports the count and aggregate size of items in that folder and its sub-folders. This way, the test points administrators to the precise folder that is growing abnormally in size and the mailbox and mailbox database to which that folder belongs.

This test is disabled by default. To enable the test, go to the **ENABLE / DISABLE TESTS** page using the menu sequence : Agents -> Tests -> Enable/Disable, pick the *Microsoft Exchange 2013/2016* as the desired **Component type**, set *Performance* as the **Test type**, choose the test from the **DISABLED TESTS** list, and click on the **>>** button to move the test to the **ENABLED TESTS** list. Finally, click the **Update** button.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal/remote agent

Outputs of the test : One set of results for every folder in every mailbox in each mailbox database on the Exchange server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap-in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Items in the folder:	Indicates the number of items currently available in this folder. Note that this measure does not include the items in the sub- folders of this folder.	Number	Compare the value of this measure across folders to know which folder consists of the maximum number of items.
Folder size:	Indicates the current size of this folder (excluding its sub-folders).	MB	Compare the value of this measure across folders to know which folder is of the maximum size. If the mailbox/mailbox database that contains this folder exhausts its configured size limit, you can compare the value of this measure across all folders in a particular mailbox/mailbox database (as the case may be) to identify which

Measurement	Description	Measurement Unit	Interpretation
			folder is impacting the mailbox/mailbox database size. Once the folder is identified, you may want to remove obsolete items from the folder to reduce its size. If the folder appears to be growing in size consistently, you may want to fine- tune the mailbox/mailbox database size limit accordingly.
Items in this folder and its subfolders:	Indicates the total number of items in this folder and its sub-folders (if any).	Number	Compare the value of this measure across folders to know which folder consists of the maximum number of items. You may also want to compare the value of this measure with that of the Items in folder measure to know which is more – items added directly to the folder? Or items in the sub-folders of the parent folder?
Folder and sub-folder size:	Indicates the total size of this folder and its sub-folders.	MB	Compare the value of this measure across folders to know which folder is of the maximum size. If the mailbox/mailbox database that contains this folder exhausts its configured size limit, you can compare the value of this measure across all folders in a particular mailbox/mailbox database (as the case may be) to identify which folder is impacting the mailbox/mailbox database size. Once the folder is identified, you

Measurement	Description	Measurement Unit	Interpretation
			may want to compare the value of this measure with that of the Folder size measure to know what is contributing to the huge folder size – items that are directly available under the parent folder? Or the items in the sub-folders? Based on this comparative analysis, you can then proceed to remove obsolete items from the parent folder or one/more of its sub- folders to reduce its size. If the folder appears to be growing in size consistently, you may want to fine- tune the mailbox/mailbox database size limit accordingly.

3.10 The Unified Messaging Layer

Besides tracking the availability and responsiveness of the Exchange server, the tests mapped to this layer:

- Measure ActiveSync and Outlook Web App performance and reports deviations;
- Monitor the RPC over HTTP connections and reports failures/delays;
- Captures snags in the operations of the Unified Message Call Router;
- Proactively detects potential slowdowns in the delivery of voice mail service to end-users

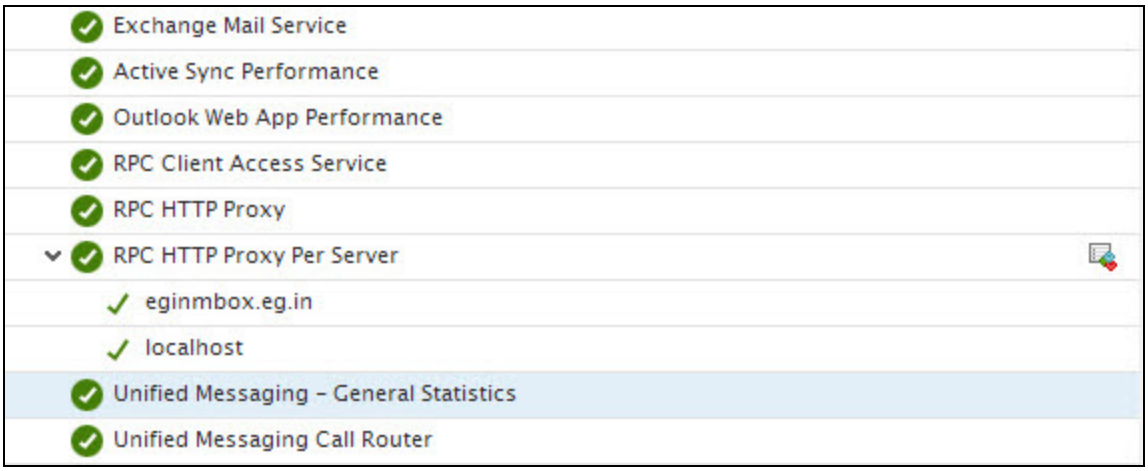

Figure 3.14: The tests mapped to the Unified Messaging Layer

3.10.1 perExchange Mail Service Test

This test monitors the availability and performance of a Microsoft Exchange 2013/2016 mail server from an external perspective. The test mimics a mail client activity by using the Exchange Web Service for sending and receiving mails.

Note:

- For this test to execute smoothly, the external agent executing the test should be in the same domain as the Exchange 2013/2016 server.
- The external agent running this test should be installed on a Windows host that supports **.Net 3.5 Framework**.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An external agent

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

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the server being monitored.
3. **PORT**- The port number of the configured **HOST**.
4. **FROM USER NAME**- Provide any valid email ID in this text box. It is recommended that you create a special mail box and email ID for monitoring purposes, and provide that email ID here.

5. **FROM USER PASSWORD**– Specify the password that corresponds to the **FROM USER NAME** you specified.
6. **CONFIRM PASSWORD**– Confirm the password by retyping it here.
7. **EXCHANGE DOMAIN NAME** - Provide a valid domain in which the target server is running.
8. **WEB SERVICE URL** – To enable the test to connect to Exchange Web Services, you need to provide the **External Web Service URL** here. To know what URL to provide, run the following command from the Exchange server's powershell command prompt:

   *Get-WebServicesVirtualDirectory -server <servername> | select name, *url* | fl*

   For instance, if your Exchange server's name is **Exchange**, then your command will be:

   *Get-WebServicesVirtualDirectory -server Exchange | select name, *url* | fl*

   Upon successful execution of the command, a list of URLs will be displayed. Note down the URL displayed against the label **ExternalUrl** , and enter it against **WEB SERVICE URL**.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Send mail availability:	Indicates the availability of the mail server for receiving the mails sent by the test.	Percent	A value of 0 indicates that the test was not successful in sending a mail. Possible reasons for this could include the mail server being down, the network connection to the server not being available, or the test configuration information being incorrect.
Sent messages:	Indicates the number of messages sent to the mail server.	Number	A value of –1 indicates that the mail server may be down or the configuration information may be incorrect.
Avg time to send messages:	Indicates time taken to send a mail to the mail server.	Secs	A high value of this measure could indicate high network traffic or that the mail server is busy.
Receive mail	Indicates the availability	Percent	The value of 0 indicates that the test

Measurement	Description	Measurement Unit	Interpretation
availability:	of the exchange server for sending mails to the mail client.		was not successful in receiving a mail message from the Exchange server. Possible reasons could be incorrect configuration information.
Received messages:	Indicates the number of messages received by the mail client from the mail server.	Number	The value of 0 indicates that the test was not successful in receiving mail messages from the Exchange server. The possible reasons could be: • The sent messages could be in the message queue of the mail server but not routed to the mail box • Configuration information may be incorrect • Network failure • The mail service may not be running in the user account
Mail received time:	Indicates the time taken by the mail client to receive a mail from the mail server.	Secs	A high value in this measure indicates that the mail server is busy or the network traffic is high.
Avg roundtrip time:	The average of the round trip time (the time lapse between transmission and reception of a message by the server) of all the messages received by the mail server during the last measurement period.	Mins	This is a key measure of quality of the mail service. An increase in roundtrip time may be indicative of a problem with the mail service. Possible reasons could include queuing failures, disk space being full, etc.

Measurement	Description	Measurement Unit	Interpretation
Max roundtrip time:	The high water mark of the round trip time (the time lapse between transmission and reception of a message by the server) of all messages received by the mail server during the last measurement period.	Mins	If the value of the Received messages measure is 1, then the value of this measure will be the same as the Avg roundtrip time measure.

3.10.2 Active Sync Performance Test

Exchange ActiveSync is a Microsoft Exchange synchronization protocol that's optimized to work together with high-latency and low-bandwidth networks. The protocol, based on HTTP and XML, lets mobile phones access an organization's information on a server that's running Microsoft Exchange. Exchange ActiveSync enables mobile phone users to access their email, calendar, contacts, and tasks, and to continue to access this information while they're working offline. In an era where enterprises offer flexible work options to their employees – eg., work-from-home, flex-work, etc. – it is Exchange ActiveSync that helps users to work productively and in an uninterrupted manner, , even when away from the work place. Naturally therefore, if ActiveSync slows down, user productivity is bound to be affected and help desk is sure to be at the receiving end of a barrage of support calls from frustrated users! If such a situation is to be avoided, help desk needs to keep track of how quickly the ActiveSync server processes user requests, proactively detect potential slowdowns, accurately pinpoint the cause of the slowdown, and promptly attend to it. This is where the **Active Sync Performance** test helps.

This test tracks the synchronization and ping requests to the ActiveSync server, checks how quickly ActiveSync services these requests, counts the requests pending processing, and in this way, points administrators to probable processing bottlenecks on the server. The test also measures the time spent by the requests at various stages of processing, and thus precisely pinpoints where the bottleneck lies – did ping/hanging sync take too long? did LDAP calls to the domain controller take too long? was RDP access to the mailbox server slow? were issues in proxying delaying request processing?

Target of the test : An Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Average hang time:	Indicates the average time that elapsed waiting for Ping and Hanging Sync requests to complete.	Secs	Direct Push is the method by which Exchange mailbox data is kept constantly up to date on a mobile device. Mobile devices that support Direct Push issue a long-lived HTTPS request to the Exchange server. If a new email message arrives or any other changes occur within the lifespan of the HTTPS request, the Exchange server issues a response to the device that states that changes have occurred and the device should initiate synchronization with the Exchange server. The device then issues a synchronization request to the server. This is called a 'Ping Sync' request. Newer versions of ActiveSync introduced certain variations to the Direct Push technology. In the new Direct Push functionality, the mobile device sends a long-lived

Measurement	Description	Measurement Unit	Interpretation
			HTTPS request known as a Hanging Sync request to the Exchange server. The Hanging Sync request parks itself on the ActiveSync server. When a new message arrives on the server, the server completes the Sync request and pushes the message directly to the device. This behavior helps reduce bandwidth and may help optimize battery consumption of the mobile device. Ideally, the value of this measure should be low. A high value could indicate that the server is taking too long to respond to Ping and Hanging Sync requests. In the event of a synchronization slowdown, you may want to compare the value of this measure with that of the Average LDAP latency, Average request time, and Average RDP latency measures to know where synchronization is bottlenecked – at the ActiveSync server waiting for Ping/Hanging requests to complete? at the domain controller? or at the mailbox server?
Average LDAP latency:	Indicates the average time that it takes for an LDAP call to return results from a domain controller, averaged	Secs	Ideally, the value of this measure should be low. A high value could indicate that the domain controller is taking too long to respond to

Measurement	Description	Measurement Unit	Interpretation
	over LDAP calls in the last minute.		LDAP calls. This can delay user authentication and slow down synchronization. In the event of a synchronization slowdown therefore, you may want to compare the value of this measure with that of the Average hang time, Average request time, and Average RDP latency measures to know where synchronization is bottlenecked – at the ActiveSync server waiting for Ping/Hanging requests to complete? at the domain controller? or at the mailbox server?
Average request time:	Indicates the average time that elapsed waiting for a sync request to complete. .	Secs	This measure includes Ping Request Time, which can increase the general response time. A high value for this measure could indicate that the ActiveSync server is unable to process requests quickly. To know why, you may want to compare the values of the Average hang time, Average LDAP latency, and Average RDP latency measures. This comparison will reveal where synchronization is bottlenecked – at the ActiveSync server waiting for Ping/Hanging requests to complete? at the domain controller? or at the mailbox server?

Measurement	Description	Measurement Unit	Interpretation
Average RPC latency:	Indicates the average time that it takes for a remote procedure call to return results from a Mailbox server, averaged over RPC calls in the last minute.	Secs	Ideally, the value of this measure should be less than 25 milliseconds – i.e., 0.025 seconds. The latency represents how long it takes from the time the Store.exe process received the packet to the time it returned the packet. An increase in latency may be caused by the the following: • An increase in RPC load • A bottleneck in one or more server resources • Hardware malfunction or configuration error If the ActiveSync server is taking an unusually long time to process sync requests, then, you may want to compare the value of this measure with that of the Average hang time, Average request time, and Average LDAP latency measures to know where synchronization is bottlenecked - at the ActiveSync server waiting for Ping/Hanging requests to complete? at the domain controller? or at the mailbox server?
Current HTTP requests received from ASP .NET:	Indicates the current number of HTTP requests received from the client via ASP .NET.	Number	This is a good indicator of the current workload of Exchange ActiveSync.

Measurement	Description	Measurement Unit	Interpretation
			If the server receives too many sync requests from users, it could result in a situation where server runs out of resources, effectively causing a 'denial of service' (DOS) attack. The worst outcome of such a situation is that the server also becomes unavailable to other users who may not be using EAS protocol to connect. If this is to be avoided, then the variations to the value of this measure should be closely tracked and administrators alerted to sudden/steady increases in this measure value.
Incoming proxy requests:	Indicates the total number of HTTP requests received from ASP.NET and proxied to another Exchange ActiveSync server since the service was restarted.	Number	A computer that is running Exchange 2013/2016 that has the Client Access server role installed can act as a proxy for other Client Access servers within the organization. If a user sends a sync request to the CAS from a mobile device, then that CAS will query the Active Directory directory service to determine the location of the user's mailbox. If the user's mailbox is found to be in a different Active Directory site than the CAS, then this CAS will check whether any other CAS exists in the same AD site as that of the user's mailbox. If such a CAS is found, then the CAS that received the

Measurement	Description	Measurement Unit	Interpretation
			user request will check whether the CAS that is close to the user's mailbox has the **InternalURL** property configured and if the authentication method is Integrated Windows authentication. If so, then the first CAS will proxy the request to the **InternalURL** of the CAS close to the user's mailbox. The *Incoming proxy requests* measure indicates the number of requests to the monitored CAS server that it proxied to another CAS server. The *Outgoing proxy requests* measure indicates the number of requests to another CAS server that were proxied to the monitored server. Both measures are indicators of the workload of the CAS server.
Outgoing proxy requests:	Indicates the total number of HTTP requests received by ASP.NET from another Exchange ActiveSync server since the service was restarted.	Number	
Ping commands pending on the server:	Indicates the number of ping commands that are currently pending on the server.	Number	The **Ping** command is used to request that the server monitor specified folders for changes that would require the client to resynchronize. A consistent rise in this value could indicate a processing bottleneck on the server.
Rate of HTTP	Indicates the rate at	Reqs/Sec	

Measurement	Description	Measurement Unit	Interpretation
requests received from the client:	which HTTP requests were received from the client via ASP .NET.		
Sync commands pending on the server:	Indicates the number of sync commands that are currently pending on the server.	Number	The **Sync** command synchronizes changes in a collection between the client and the server. A consistent rise in this value could indicate a processing bottleneck on the server. You may want to check the value of the Sync commands processed rate measure to corroborate this finding.
Sync commands processed rate:	Indicates the number of sync commands that were processed by second.	Commands/Sec	A consistent dip in this measure is a cause for concern. If the value of the Sync commands pending on the server measure has been increasing steadily, you may want to check the variations to this measure to verify if indeed the server is experiencing a processing slowdown.
Wrong CAS proxy requests:	Indicates the total number of HTTP requests that were received by ASP.NET since the service was restarted that can't be proxied to another Client Access server because the user has to reconfigure the device to point to that	Number	Ideally, the value of this measure should be 0. A non-zero value is indicative of ActiveSync failure due to wrong proxy configuration. One of the reasons that such a problem may occur is when a user mailbox and a user ID are moved to another AD site and then brought back to the original AD site. When this happens, in more cases than one, Active Sync will

Measurement	Description	Measurement Unit	Interpretation
	Client Access server directly.		fail, because the redirect URL would still be referring to the old CAS server.

3.10.3 Exchange ActiveSync Servers Test

Where Exchange ActiveSync is used to synchronize mobile devices with Exchange server mailboxes, Exchange administrators may want to know which devices are connecting to the server at any given point in time, so that accesses by unauthorized devices can be instantly detected and blocked. Administrators may also want to track the usage of mailboxes by mobile devices over time and identify the most and the least effective users, so that access policies can be accordingly drawn. Moreover, when a device user complains of a slowdown when accessing his/her mailbox, administrators may want to take a look at the network traffic generated by every device that is connecting to the server at the time of the slowdown, so that devices that are choking the bandwidth and causing the slowness can be accurately isolated. The **Exchange ActiveSync Servers** test performs all these checks periodically and provides Exchange administrators with actionable information that will enable them to take well-informed and intelligent performance/policy decisions.

This test auto-discovers the devices that are synchronizing with the Exchange mailboxes via ActiveSync, and for each device, reports the number of hits/accesses made by that device and the amount of data transmitted and received by that device. In the process, the test points administrators to the following:

- Devices that are currently connected to the Exchange server; unauthorized devices can thus be quickly captured;
- Devices that are accessing the Exchange server mailboxes frequently and those that seldom use the mailboxes; sizing and policy decisions can be taken based on this observation
- Devices that are consuming excessive bandwidth resources and could hence be contributing to the sluggish quality of the network;

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each device or IP address that is currently accessing the mailboxes on the Exchange server

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** - The port number of the client access server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **LOGFILE NAME** – The Client Access Server is an IIS web server that hosts Exchange-related web pages. This is why, like any other IIS web server, the client access server creates a daily log of its activities – including Exchange ActiveSync- related activities - in the *C:\inetpub\logs\logfiles\W3SVC1* directory by default. To report metrics on ActiveSync, this test parses the client access server's log file, reads the ActiveSync- related errors/warnings/general information messages that were recently logged (i.e., during the last 5 minutes) from the file, and writes them to a *ActiveSynchLog.log* file it creates in the **<EG_ AGENT_INSTALL_DIR>\agent\logs** directory. Then, the test reads the metrics of interest from this log file and reports them to the eG manager. To enable the test to do the above, you need to specify the exact path to the directory that contains the client access server's logs in the **LOGFILENAME** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total hits:	Indicates the current number of hits/accesses to the Exchange mailbox server from this device.	Number	Comparing the value of this measure across devices will help you to identify the device that is constantly accessing the Exchange mailbox server and that which is not using the server as frequently. Based on these usage metrics, administrators can define access

Measurement	Description	Measurement Unit	Interpretation
			policies. Also, this measure serves as a good indicator of the level of device activity on the Exchange server; based on this knowledge, administrators can right-size their Exchange infrastructure – i.e., decide on how much CPU, memory, bandwidth, and disk resources the Exchange server has to be allocated so that it can handle the ActiveSync load.
Data sent:	Indicates the amount of data this device is currently sending to the Exchange mail server.	KB	Compare the value of these measures across the devices to identify the device that is currently generating the maximum amount of network traffic when interacting with its mailbox on the Exchange server. In the event of a slowdown, this comparative analysis will point administrators to that device which is engaged in bandwidth-intensive conversations with the Exchange server, thus causing accesses to slow down. During normal operations on the other hand, administrators can analyze these measures over time to gauge the average network throughput of ActiveSync activities; this can help them decide whether/not more network resources need to be allocated to

Measurement	Description	Measurement Unit	Interpretation
Data received:	Indicates the amount of data currently received by this device from the Exchange mail server.	KB	handle ActiveSync load efficiently.
Average unique devices:	Indicates the number of unique devices currently accessing the ActiveSync server.		

3.10.4 Exchange ActiveSync Requests Status Test

When a mobile device attempts to synchronize with a mailbox on the Exchange server, the server returns an HTTP status code to the device indicating the status of the synchronization attempt. Some of the most critical HTTP status codes for ActiveSync and their interpretations are as follows:

HTTP status code	Description
HTTP_200	Indicates that the device successfully connected to the Exchange server and synchronized with the mailbox on the server.
HTTP_401	Indicates one or all of the following: • The credentials provided to access the server are incorrect; • The user is not enabled for synchronization
HTTP_404	Indicates that an issue exists with the user account
HTTP_404	Indicates that the file requested is not found on the server
HTTP_449	Indicates that the synchronization attempt should be retried
HTTP_500	Indicates one or all of the following: • The Internet Information Service is unavailable. • Windows Integrated Authentication is not enabled on the Exchange Server virtual directory of the server where the mailbox of the user resides. • Synchronization is tried when the mailbox is being moved.
HTTP_502	Indicates an error in the proxy server used to connect to the ActiveSync Server
HTTP_503	Indicates that the ActiveSync service is unavailable

Periodic review of these status codes and the synchronization attempts that resulted in these codes is imperative to understand how error-prone ActiveSync on the Exchange server is, identify the errors that occur frequently, investigate why these errors occur, and easily troubleshoot them. This is where the Exchange ActiveSync Requests Status test helps!

This test automatically discovers the HTTP status codes returned by the Exchange server for ActiveSync accesses. For each status code so discovered, the test reports the number and percentage of accesses that returned that status code. This way, the test points administrators to status codes that were returned most often, thus shedding light on ActiveSync errors that occurred frequently.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each status code returned by the ActiveSync server

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** - The port number of the client access server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command-line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap-in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **LOGFILE NAME** – The Client Access Server is an IIS web server that hosts Exchange-related web pages. This is why, like any other IIS web server, the client access server creates a daily log of its activities – including Exchange ActiveSync-related activities - in the *C:\inetpub\logs\logfiles\W3SVC1* directory by default. To report metrics on ActiveSync, this test parses the client access server's log file, reads the ActiveSync-related errors/warnings/general information messages that were recently logged (i.e., during the last 5 minutes) from the file, and writes them to a *ActiveSynchLog.log* file it creates in the **<EG_AGENT_INSTALL_DIR>\agent\logs** directory. Then, the test reads the metrics of interest from this log file and reports them to the eG manager. To enable the test to do the above, you

need to specify the exact path to the directory that contains the client access server's logs in the **LOGFILENAME** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total hits:	Indicates the current number of hits to the Exchange mailbox server that returned this status code.	Number	Compare the value of these measures across status codes to identify the status code that is returned frequently. High values for the 4xx or 5xx class of status codes is a cause for concern, as they indicate client and server errors respectively. If such status codes are returned often, administrators will have to look up the Microsoft documentation to understand what error condition each code represents and how to resolve it.
Hit ratio:	Indicates the percentage of hits to the Exchange mailbox server that returned this status code.	Percent	

3.10.5 Exchange ActiveSync Devices Test

In environments where ActiveSync is enabled, it is normal for users wielding different types of devices to synchronize their mailbox with their device. In such environments, administrators should pay close attention to the device types that are connected to the Exchange server mailboxes at any given point in time, so that unsupported device types can be detected and the users using such types of devices identified and advised accordingly. It is also essential that administrators study how frequently each of these device types are accessing the Exchange server and monitor the level of activity generated by these device types on the server and on the network. If a device users complains of delays in accessing his/her mailbox, then this visibility will enable administrators to identify those device types to which the slowdown can be attributed. In addition, administrators will also need to know from time-to-time how much load ActiveSync imposes on the Exchange server and the network, across all device types! This aggregated measure will enable administrators to figure out whether/not the Exchange server is sized right to handle the load. To receive such in-depth insights into ActiveSync performance – both at the per-device type level and across all device types – administrators can use the **Exchange ActiveSync Devices** test.

This test auto-discovers the device types currently synchronizing with the Exchange server. For each device type, the test reports the number of ActiveSync accesses made by that device type and the number and size of items transmitted and received by that device type. This way, the test leads administrators to those device types that are utilizing the available network and server resources excessively, thus degrading the experience of some or all device users. Detailed metrics provided by the test also help administrators identify all the users who are using devices of a particular type and pinpoint the exact user who is engaged in a resource-intensive interaction with the Exchange server mailbox. Additionally, the test reports metrics across all device types, thus enabling administrators to measure the current load on the server and the network and assess the ability of the server to handle that load.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each device type that is accessing the Exchange server; an additional All descriptor is also supported, which reports a set of aggregated metrics across all device types

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** - The port number of the client access server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command-line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap-in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **LOGFILE NAME** – The Client Access Server is an IIS web server that hosts Exchange-related web pages. This is why, like any other IIS web server, the client access server creates a daily log of its activities – including Exchange ActiveSync-related activities - in the *C:\inetpub\logs\logfiles\W3SVC1* directory by default. To report metrics on ActiveSync, this test parses the client access server's log file, reads the ActiveSync-related errors/warnings/general information messages that were recently logged (i.e., during the last 5 minutes) from the file, and writes them to a *ActiveSynchLog.log* file it creates in the **<EG_**

AGENT_INSTALL_DIR>\agent\logs directory. Then, the test reads the metrics of interest from this log file and reports them to the eG manager. To enable the test to do the above, you need to specify the exact path to the directory that contains the client access server's logs in the **LOGFILENAME** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total hits:	Indicates the number of hits/accesses made by this device type to the Exchange server mailbox.	Number	Comparing the value of this measure across device types will help administrators identify that device type which is very actively synchronizing with the Exchange mailbox. Using the detailed diagnosis of this measure, administrators can also identify the precise user who is making the maximum number of accesses, which device that user is using, and the details of that device. Based on this information, access policies can be defined. Also, by observing the variations in the value of this measure for the All descriptor, administrators can effectively gauge the typical level of activity on the Exchange server and figure out if the server is sized right to handle this load.
Total items sent:	Indicates the number of items currently sent from this device type to the Exchange server.	Number	These measures indicate how much network traffic and I/O load is generated by each of the device types. By comparing the value of

Measurement	Description	Measurement Unit	Interpretation
			these measures across device types, administrators can easily and accurately identify that device type that is engaged in resource-intensive communication with the Exchange server. In the event of a slowdown, the results of this comparative analysis will lead administrators to that device type that could be contributing to the slowdown. Once the device type is identified, you can use the detailed diagnosis of the Total hits measure to know which user of that device type is actually choking the network/server and what device he/she is currently using.
Total items received:	Indicates the number of items currently received by this device type from the Exchange server.	Number	
Data sent:	Indicates the amount of data currently sent from this device type to the Exchange server.	KB	
Data received:	Indicates the amount of data currently received by this device type from the Exchange server.	KB	

3.10.6 ActiveSync Device Status

Administrators must constantly track the devices connecting to ActiveSync, so that they can proactively identify devices that are unable to sync with user mailboxes, the users using these devices, and the probable reason for the non-sync, much before device users even notice that something is wrong! Likewise, administrators should also be able to zero-in on devices that are connected to ActiveSync, but have been inactive for long time periods, so that they can take efforts to clear out such devices en masse. To isolate such devices, administrators can use the **ActiveSync Device Status** test. This test reports the count of devices that are using ActiveSync without a glitch, those that are having problems using Activesync, and the stale (inactive) devices. Using the detailed diagnosis of this measure, the devices that are operating well, those that are not, and those that are stale can be clearly isolated.

Target of the test : An Exchange 2013/2016 Server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** – The port number of the Exchange server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell which enables you to administer every part of Microsoft Exchange. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. By default, the test auto-discovers the location of the Exchange management shell and thus, automatically loads the Exchange management shell snap -in (exshell.psc1) for script execution. This is why, the **XCHGEXTENSIONSHELLPATH** is set to *none* by default.
5. **INACTIVE DEVICE AGE IN DAYS** – Specify the minimum duration (in days) for which a device should not have synchronized with its mailbox for it to be counted as a stale/inactive device.
6. **SHOW DD FOR OK STATUS DEVICE** – By default, this flag is set to **No**, indicating that detailed metrics will not be available by default for the *Device with OK status* measure reported by this test. To ensure that this test collects detailed metrics for this measure, set this flag to **Yes**.
7. **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
Device with OK status:	Indicates the count of devices that are	Number	**Detailed diagnostics will be available for this measure only**

Measurement	Description	Measurement Unit	Interpretation
	currently able to synchronize with their mailboxes via ActiveSync.		**if the SHOW DD FOR OK STATUS DEVICE flag is set to 'Yes'.** If available, then, you can use the detailed diagnosis of this measure to know which devices are able to connect to ActiveSync and synchronize with their mailboxes, and which users are using these devices.
Device with not OK status:	Indicates the number of devices that are currently unable to synchronize with their Exchange mailboxes via ActiveSync.	Number	Ideally, the value of this measure should be 0. A non-zero value for this measure implies that one/more devices are unable to synchronize with their Exchange mailboxes. To know these devices and their users, use the detailed diagnosis of this measure.
Stale devices:	Indicates the current number of stale devices.	Number	Use the detailed diagnosis of this measure to know which devices are inactive, which users are using such devices, and how long these devices have remained inactive.

3.10.7 Exchange ActiveSync Policy Compliance Test

Exchange ActiveSync mailbox policies let you apply a common set of policy or security settings to a user or group of users. With the help of these policies, Exchange administrators can indicate what specific devices – thus users – connecting to ActiveSyc, can do.

EAS policies are applied to users; each user can have zero policies or one EAS policy at any given time. If you don't explicitly assign a policy to a user, the default policy is applied instead. During the initial sync of a new device (that is, one that has not been synchronized to the server before), the device and server exchange what EAS calls a policy key. Think of the policy key as a GUID or MAC address; it's a unique key that indicates one specific policy. If the device and server keys do not match, the device is required to request the most recent policy and then apply it. The process of

applying a policy to the device is known as provisioning. On most devices, the user will see a dialog box indicating that the server is applying a policy and asking whether to accept it. If the user declines the policy, the server might or might not allow the device to continue to sync to it; the exact behavior depends on whether the default policy on the server allows non-provisioned devices.

Not every device that connects to ActiveSync will implement every setting defined in a policy; some devices may even lie about the policy settings that they implement. Hence, the onus of determining the number of devices that comply with the policy settings and to what extent is the compliance, lies with the administrator. To determine this, administrators can use the **Exchange ActiveSync Policy Compliance** test. This test reports the count and percentage of devices connecting to ActiveSync that are fully compliant, partially compliant, and completely non-compliant with their mailbox policies. This way, the test reveals the degree of compliance to configured policies.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each type of compliance – Compliant, Partially compliant, Not compliant, Unknown

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** - The port number of the client access server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **LOGFILE NAME** – The Client Access Server is an IIS web server that hosts Exchange-related web pages. This is why, like any other IIS web server, the client access server creates a daily log of its activities – including Exchange ActiveSync- related activities - in the *C:\inetpub\logs\logfiles\W3SVC1* directory by default. To report metrics on ActiveSync, this test parses the client access server's log file, reads the ActiveSync- related errors/warnings/general information messages that were recently logged (i.e., during the last

5 minutes) from the file, and writes them to a *ActiveSynchLog.log* file it creates in the **<EG_ AGENT_INSTALL_DIR>\agent\logs** directory. Then, the test reads the metrics of interest from this log file and reports them to the eG manager. To enable the test to do the above, you need to specify the exact path to the directory that contains the client access server's logs in the **LOGFILENAME** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total hits:	Indicates the number of devices currently accessing ActiveSync that are of this compliance type .	Number	Compare the value of this measure across compliance types to know how compliant the maximum number of devices are - fully compliant? partially compliant? non-compliant? or unknown? (i.e., the compliance level cannot be determined)
Hits ratio:	Indicates the percentage of devices currently accessing ActiveSync that are of this compliance type.	Percent	Compare the value of this measure across compliance types to know the degree of compliance of devices accessing ActiveSync - fully compliant? partially compliant? or non- compliant? or unknown? (i.e., the compliance level cannot be determined)

3.10.8 Exchange ActiveSync User Agents Test

Devices communicating with Exchange via ActiveSync identify themselves to Exchange using a 'User Agent' string and a 'User Agent Type' string. For instance, an iPhone may identify itself to Exchange using the user agent string 'Apple-iPhone/xxx.xxx' and the user agent type 'iPhone'. While the user agent string is unique for every device, multiple devices can be of the same user agent type. By tracking the types of user agents that are accessing Exchange via ActiveSync, administrators can determine which type of devices are attempting to synchronize with the Exchange mailboxes. In times of an overload, this information may point administrators to the exact type of devices that could be contributing to the heavy load. To obtain this useful information, administrators can use the **Exchange ActiveSync User Agents** test. For every user agent type, this test reports the total number of user agents of that type that are accessing ActiveSync at any

given point in time. In addition, it also reports the number of unique devices of each type synchronizing with Exchange. This not only indicates the current synchronization load on Exchange, but also helps identify the user agent types (i.e., device types) that could be contributing to the workload.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each user agent type

Configurable parameters for the test

1. **TEST PERIOD** - Indicates how often this test needs to be executed.
2. **HOST** - Indicates the IP address of the Exchange server.
3. **PORT** - The port number of the client access server. By default, this is 443.
4. **XCHGEXTENSIONSHELLPATH** - The Exchange Management Shell is a command- line management interface, built on Windows PowerShell v2, which enables you to administer every part of the Microsoft Exchange Server. This test uses the Exchange management shell to run scripts and collect the desired performance metrics from the Exchange server. To enable the test to load the Exchange management shell snap- in (exshell.psc1) for script execution, you need to specify the full path to the Exchange management shell in the **XCHGEXTENSIONSHELLPATH** text box. For instance, your specification can be, *c:\progra~1\micros~1\exchan~1\v14\bin\exshell.psc1*.
5. **LOGFILE NAME** – The Client Access Server is an IIS web server that hosts Exchange-related web pages. This is why, like any other IIS web server, the client access server creates a daily log of its activities – including Exchange ActiveSync- related activities - in the *C:\inetpub\logs\logfiles\W3SVC1* directory by default. To report metrics on ActiveSync, this test parses the client access server's log file, reads the ActiveSync- related errors/warnings/general information messages that were recently logged (i.e., during the last 5 minutes) from the file, and writes them to a *ActiveSynchLog.log* file it creates in the **<EG_ AGENT_INSTALL_DIR>\agent\logs** directory. Then, the test reads the metrics of interest from this log file and reports them to the eG manager. To enable the test to do the above, you need to specify the exact path to the directory that contains the client access server's logs in the **LOGFILENAME** text box.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total hits:	Indicates the current number of synchronization requests from user agents of this type.	Number	This is a good indicator of the current synchronization load on the Exchange server. You can compare the value of this measure across user agents to know which type of user agents are actually overloading the server.
Unique devices:	Indicates the number of unique devices of this user agent type that are currently accessing ActiveSync.	Number	Compare the value of this measure across user agent types to identify the device type that is significantly impacting the server workload.

3.10.9 Exchange ActiveSync Device Errors Test

In order to enable administrators to quickly troubleshoot current issues with ActiveSync, the eG Exchange Monitor intelligently reads ActiveSync- related errors/warnings/general information messages captured recently (i.e., in the last 5 minutes) from the client access server's log file and writes them to the *ActiveSynchLog.log* file it creates in the **<EG_AGENT_INSTALL_DIR>\agent\logs** directory. At specified intervals, this test scans the *ActiveSynchLog.log* file for configured patterns of errors and reports the number and nature of such errors (if found).

For this test to work, the Exchange ActiveSync User Agents Test should be running and reporting metrics.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every **SEARCHPATTERN** configured

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the server listens

4. **ALERTFILE** – By default, the full path to the *ActiveSynchLog.log* file is set here.

5. **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 you want all the messages in a log file to be monitored, then your specification would be: <*PatternName*>:*.

6. **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

7. **EXCLUDEPATTERN** - Provide a comma- separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example *critical*, *exception**. By default, this parameter is set to 'none'.

8. **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 **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.

9. **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 **ROTATINGFILE** 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 .

10. **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.

11. **ROLLOVERFILE** - 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**.

12. **OVERWRITTENFILE** - 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.

13. **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., 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 **ENCODEFORMAT** specification will be: UTF-8,none,UTF-16.

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.

14. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
15. 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
Recent errors:	Indicates the number of errors that were added to the *ActiveSynchLog.log*	Number	The value of this measure is a clear indicator of the number of "new" errors detected in

Measurement	Description	Measurement Unit	Interpretation
	file when the test was last executed.		ActiveSync. The detailed diagnosis of this measure, if enabled, provides the detailed descriptions of the errors of the configured patterns.

3.10.10 Exchange ActiveSync Device Commands Test

In order to enable administrators to quickly troubleshoot current issues with ActiveSync, the eG Exchange Monitor intelligently reads ActiveSync- related errors/warnings/general information messages captured recently (i.e., in the last 5 minutes) from the client access server's log file and writes them to the *ActiveSynchLog.log* file it creates in the **< EG_ AGENT_ INSTALL_ DIR >\agent\logs** directory. At specified intervals, this test scans the *ActiveSynchLog.log* file for configured patterns of messages and reports the number and nature of messages (if found) matching the configured patterns.

For this test to work, the Exchange ActiveSync User Agents Test should be running and reporting metrics.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for every **SEARCHPATTERN** configured

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the server listens
4. **ALERTFILE** – By default, the full path to the *ActiveSynchLog.log* file is set here.
5. **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 you want all the messages in a log file to be monitored, then your specification would be: *<PatternName>:**.

6. **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

7. **EXCLUDEPATTERN** - Provide a comma- separated list of patterns to be excluded from monitoring in the **EXCLUDEPATTERN** text box. For example **critical*, *exception**. By default, this parameter is set to 'none'.

8. **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 **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.

9. **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 **ROTATINGFILE** 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 .

10. **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.

11. **ROLLOVERFILE** - 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**.

12. **OVERWRITTENFILE** - 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.
13. **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., 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 **ENCODEFORMAT** specification will be: UTF-8,none,UTF-16.

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.

14. **DD FREQUENCY** - Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD FREQUENCY**.
15. 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
New messages:	Indicates the number of messages that were added to the ActiveSynchLog.log file when the test was last executed.	Number	The detailed diagnosis of this measure, if enabled, provides the detailed descriptions of the messages of the configured patterns.

3.10.11 Outlook Web App Performance Test

Microsoft Outlook Web App lets users access their Exchange mailbox from almost any Web browser. In environments that span geographies or in enterprises with a liberal work culture, you will find many users accessing their mailboxes from remote locations world-wide using Outlook Web App. Moreover, in such environments, you will also find user-to-user communication enabled through an integration of Instant Messaging (IM) and OWA. To ensure a high quality user

experience with the Exchange server, administrators of such environments should make sure that remote users do not complain of slowdowns / failures when accessing their mailboxes or when communicating with each other. For this, administrators should track the remote session load on OWA and IM services, gauge how well these services handle the load, proactively detect probable slowdowns in responsiveness, and fix the problem before users register a complaint with help desk. This is where the **Outlook Web App Performance** test helps. This test tracks the requests for OWA and IM services, measures the time taken by the Exchange server to service these requests, and brings even the slightest dips in responsiveness and the smallest of failures to the attention of administrators. This way, the test sheds light on processing bottlenecks that the OWA / IM services may be experiencing.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange server monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.
4. 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
Average response	Indicates the average	Secs	A high value for this measure is

Measurement	Description	Measurement Unit	Interpretation
time:	time that elapsed between the beginning and end of an OEH or ASPX request.		indicative of a bottleneck when processing requests to the OWA service. The detailed diagnosis of this measure, if enabled, reveals the PID of the worker process that is hogging the CPU resources and causing responsiveness to deteriorate.
Average search time:	Indicates the average time that elapsed while waiting for a search to complete.	Secs	A high value indicates a searching bottleneck.
Proxy users:	Indicates the number of users who are currently logged on whose mailbox access is being proxied to another server.	Number	
Active user sessions:	Indicates the number of users who are currently signed in to Outlook Web App.	Number	This is a good measure of the load on OWA.
Failed outlook web app requests rate:	Indicates the rate at which OWA requests failed.	Reqs/Sec	A very low value is desired for this measure. A high value is a cause for concern.
Instant messages received rate:	Indicates the rate at which instant messages were received by the server.	Msgs/Sec	These measures are good indicators of the load on the IM service.
Instant messages sent rate:	Indicates the rate at which instant messages	Msgs/Sec	

Measurement	Description	Measurement Unit	Interpretation
	were sent by the server.		
Percent of instant message sign in failures:	Indicates the percentage of the last 100 users who failed to sign in to the Instant Messaging server.	Percent	A value close to 100% is a cause for concern, as it indicates that almost all sign-in attempts in the recent past have failed. The reasons for the anomaly will have to be investigated.
Instant message sign in failures:	Indicates the number of attempts to sign in to Instant Messaging that have failed since the service was started.	Number	A non-zero value is desired for this measure.
Instant message delivery failures:	Indicates the number of attempts to deliver instant messages that have failed since the service was started.	Number	A non-zero value is desired for this measure.
Instant messages received:	Indicates the number of instant messages that were received since the service was started.	Number	
Instant messages sent:	Indicates the total number of instant messages that were sent since the service was started.	Number	
Active instant messaging users:	Indicates the number of users who are currently signed in to Instant Messaging in Outlook Web App.	Number	This is a good indicator of the current workload of the IM service.
Outlook web app	Indicates the rate at	Sessions/sec	

Measurement	Description	Measurement Unit	Interpretation
user sessions created rate:	which OWA user sessions were created.		
Proxy user request rate:	Indicates the number of proxied requests made per second.	Reqs/Sec	
Timed out requests:	Indicates the number of requests that timed out.	Number	A timeout can occur if there is already one request running from the same user and the new requests from that user could not wait longer than the configured context lock time-out period. The default timeout period setting is 3 seconds. If the value of this measure is very high, it indicates a large number of time outs. You may want to consider resetting the context lock time-out period to reduce this number.
Requests handled rate:	Indicates the rate at which requests were serviced by OWA.	Reqs/Sec	A high value is desired for this measure. A consistent drop in this value is indicative of a processing bottleneck.
Current unique users:	Indicates the number of unique users currently signed in to Outlook Web App.	Number	

The detailed diagnosis of the *Average response time* measure, if enabled, reveals the PID of the worker process that is hogging the CPU resources and causing responsiveness to deteriorate.

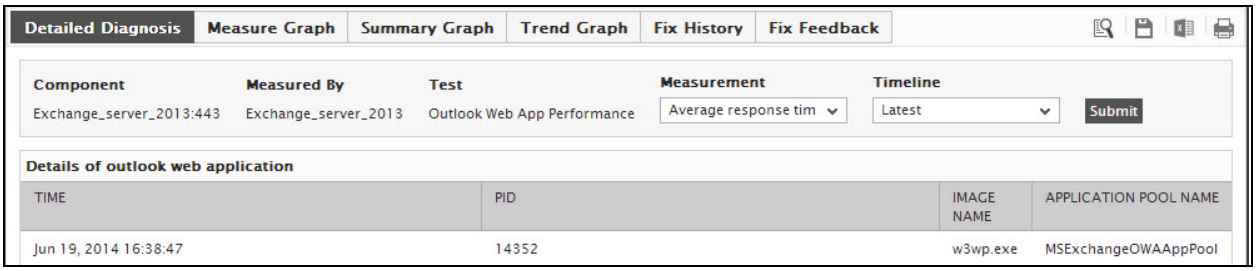

Figure 3.15: The detailed diagnosis of the Average response time measure

3.10.12 RPC Client Access Service Test

In Microsoft Exchange Server 2013/2016, the Outlook Anywhere feature, formerly known as RPC over HTTP, lets clients who use Microsoft Outlook 2013/2016, Outlook 2010, or Outlook 2007 connect to their Exchange servers from outside the corporate network or over the Internet using the RPC over HTTP Windows networking component. The Windows RPC over a component, which Outlook Anywhere clients use to connect, wraps remote procedure calls (RPCs) with an HTTP layer. This allows traffic to traverse network firewalls without requiring RPC ports to be opened. To ensure that the user experience with Exchange remains unaffected, administrators should be able capture RPC connection failures to the server and detect bottlenecks in RPC connections, well before users notice and complain. This is where the **RPC Client Access Service** test helps. By monitoring the RPC connection attempts to the server over HTTP and capturing connection failures and delays promptly, the **RPC Client Access Service** test proactively alerts administrators to real/potential road-blocks to server accesses from Outlook clients.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active users count:	Indicates the number of unique users that have shown some activity in the last 2 minutes.	Number	These measures are a good indicator of the current session load on the Exchange server.
RPCs attempted by users:	Indicates the client-reported number of RPCs attempted by users (since the service wasa started).	Number	
RPCs failed:	Indicates the client-reported number of failed RPCs (since the service was started).	Number	Ideally, this value should be 0. A high value is indicative of frequent RPC failures. The reasons for the same will have to be uncovered.
RPCs succeeded:	Indicates the client-reported number of successful RPCs (since the service was started).	Number	
RPC average latency:	Indicates the latency averaged for the past 1024 packets.	Secs	This value should be below 0.050 seconds at all times. A slowdown in RPC packet processing can adversely impact the user experience.
RFC operations rate:	Indicates the rate at which RPC operations occur, per second.	Operations/Sec	Generally, spikes in RPC requests that do not increase RPC operations/sec indicate that there are bottlenecks preventing the store from fulfilling the requests in a timely manner. It is relatively simple to identify where the bottlenecks are occurring with

Measurement	Description	Measurement Unit	Interpretation
			regards to RPC requests and RPC operations/sec. If the client experiences delays, but the RPC requests are zero and the RPC operations/sec are low, the performance problem is happening before Exchange processes the requests (that is, before the Microsoft Exchange Information Store service actually gets the incoming requests). All other combinations point to a problem either while Exchange processes the requests or after Exchange processes those requests.
RPC packets rate:	Indicates the rate at which RPC packets are processed.	Packets/Sec	A high value is desired for this measure. If this value drops steadily, it could indicate a connection bottleneck.
Current client requests being processed:	Indicates the number of client requests that are currently being processed by the RPC Client Access service.	Number	The Exchange server is configured with a pre-set maximum number of RPC requests that can be handled simultaneously (default is 100). If this value is exceeded, client requests to the server will be rejected. This measure should be below 30 most of the time.
Users connected:	Indicates the number of users who are connected to the service.	Number	This is a good indicator of the current user load on the server.

3.10.13 RPC HTTP Proxy Test

Since all Outlook connectivity in Exchange 2013/2016 takes place over Outlook Anywhere by default, at any given point in time, the Exchange server will be inundated with requests from remote Outlook Anywhere clients. To be able to handle this load, the back-end Exchange server should be sized with adequate server and network resources. To determine the resource requirement, administrators should first figure out how much load is imposed by Outlook Aywhere clients on the server and the network. The **RPC HTTP Proxy** test helps administrators with this. This test reports the number of users currently connected to the Exchange server over Outlook Anywhere, the rate at which requests are sent to the back-end Exchange server, and the bandwidth used by the back-end server when processing and responding to the requests. In the process, the test sheds light on the load imposed by Outlook Anywhere clients on the back-end server and the network.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Exchange 2013/2016 server

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current unique users:	Indicates the number of unique users currently connected to the back-end Exchange server over RPC/HTTP.	Number	This is a good measure of the current user load on the server over Outlook Anywhere.
RPC proxy connection attempts rate:	Indicates the rate at which Outlook Anywhere clients attempted to establish a connection to a back-	Attempts/Sec	

Measurement	Description	Measurement Unit	Interpretation
	end server via RPC/HTTP.		
RPC/HTTP requests rate:	Indicates the rate at which RPC/HTTP requests were sent to the back-end servers.	Reqs/Sec	
Total incoming bandwidth:	Indicates the bandwidth consumed by RPC/HTTP requests received by the Exchange server.	Kbps	These measures are good indicators of the bandwidth consumption of RPC/HTTP traffic. By observing variations to these measures over time, administrators can figure out if the Exchange servers have to be sized with more network resources to handle this traffic.
Total outgoing bandwidth:	Indicates the bandwidth consumed when the Exchange server sent data over RPC/HTTP to the Outlook Anyhwere clients.	Kbps	

3.10.14 RPC HTTP Proxy Per Server Test

Where multiple instances of an Exchange server are running on the same host, administrators can use this test to assess the RPC/HTTP traffic load on the individual instances and to identify the overloaded instance.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for each Exchange server instance running on a host

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Number of connections via RPC/HTTP:	Indicates the number of connections made to this Exchange server instance via RPC/HTTP.	Number	Compare the value of this measure across instances to know which instance has the maximum number of RPC/HTTP connections.

3.10.15 Unified Messaging Call Router Test

The Client Access server runs the Unified Messaging Call Router service. This service receives an incoming call and forwards it to the Mailbox server. Additionally, where users configure their mobile numbers and call forwarding as part of their voice mail settings, this service also receives missed call notifications, which it forwards to the Mailbox server. If this service keeps rejecting inbound calls and notifications, end-user experience with voice mail will suffer, resulting in dissatisfied, unproductive users. To avoid this, administrators should monitor calls to this service and proactively capture call rejections, before it impacts user experience. This is exactly what the **Unified Messaging Call Router** test does. This test keeps track of the inbound calls to the UM Call Router service and brings unusually high rate of call rejections to the attention of administrators, so that potential dips in the quality of the voice mail service can be proactively detected and addressed.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the **HOST** listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Rejected inbound calls:	Indicates the percentage of inbound calls that were rejected by the Microsoft Exchange Unified Messaging Call router service over the last hour.	Percent	Ideally, the value of this measure should be 0. If this measure reports an abnormally high value – say, over 50% - it is a cause for concern. In this case, review the event logs on the Client Access server (CAS) to determine whether the UM objects, such as umipgateway and umhuntgroup, are configured correctly. If the event logs do not contain enough information, you may have to enable UM event logs at the Expert level, and then review the UM trace log files.
Proxied and rejected missed call notifications:	Indicates the percentage of missed call notifications that were proxied to and rejected by the Microsoft Exchange Unified Messaging over the last hour.	Percent	Ideally, the value of this measure should be 0. If this measure reports a non- zero value, it indicates that one/more missed call notifications were rejected by the service. In this case, review the event logs on the Client Access server (CAS) to determine whether the UM objects, such as umipgateway and umhuntgroup, are configured correctly. If the event logs do not contain enough information, you may have to enable UM event logs at the Expert level, and then review the UM trace log files.
Inbound calls received:	Indicates the total number of inbound calls that were	Number	

Measurement	Description	Measurement Unit	Interpretation
	received by the service since startup.		
Inbound calls rejected:	Indicates the total number of inbound calls that were rejected by the Microsoft Exchange Unified Messaging Call Router service since its startup.	Number	

3.10.16 Unified Messaging – General Statistics Test

How good a user's voice mail experience with Exchange is depends upon how quickly Unified Messaging processes incoming calls, resolves the caller IDs, and delivers the service. Frequent call failures, unexpected call disconnects, and high call latencies not only impact the users' voice mail experience, but also provoke users to doubt the efficiency of Unified Messaging. If such doubts are to be avoided and user confidence on Unified Messaging is to be improved, administrators should use the **Unified Messaging – General Statistics** test at regular intervals to check the overall health and performance of Unified Messaging and proactively detect performance bottlenecks. This way, administrators can quickly initiate the required remedial measures and can clear the bottleneck, before end-users complain.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

Outputs of the test : One set of results for the Microsoft Exchange 2013/2016 server being monitored

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Successful caller ID resolutions:	Indicates the percentage of caller IDs that were resolved successfully.	Percent	A high value is desired for this measure.
Successful extension caller ID resolutions:	Indicates the percentage of successful attempts to resolve caller IDs that contained no "@" or "+" signs and were of the same length as the dial plan's extension.	Percent	A high value is desired for this measure.
Average call duration:	Indicates the average duration of calls since the service was started.	Secs	
Average latency:	Indicates the average time from the moment a voice mail event occurs and Unified Messaging receives confirmation from the IP gateway that the message was delivered.	Secs	This average is calculated over the last 50 messages. A consistent rise in this value is indicative of poor Unified Messaging performance.
Calls disconnected:	Indicates the number of calls that were disconnected because they exceeded the UM maximum call length.	Number	The value of this measure includes all types of calls, including fax calls.
Calls disconnected by user failure:	Indicates the total number of calls that disconnected after too many user entry failures.	Number	

Measurement	Description	Measurement Unit	Interpretation
Active calls connected to UM server:	Indicates the number of calls that are currently connected to the UM server.	Number	This is a good indicator of the current workload on the UM server.
Active fax calls connected to UM server:	Indicates the number of fax calls that are currently connected to the UM server.	Number	Voice calls become fax calls after a fax tone is detected. This is a good indicator of the current fax call workload on the UM server.
Subscribers connected to the UM server:	Indicates the number of logged on subscribers who are currently connected to the UM server.	Number	This is a good indicator of the current user load on the UM server.
Active voice mail calls:	Indicates the number of voice mail calls that are currently connected to the Unified Messaging server.	Number	
Delayed calls:	Indicates the number of calls that experienced delays longer than 2 seconds.	Number	Ideally, the value of this measure should be 0. A high value indicates that too many calls are delayed. The reason for the delay should then be investigated and resolved.
User response latency:	Indicates the average response time for the system to respond to a user request.	Secs	A high valu or a steady increase in this value is indicative of the poor responsiveness of the UM server.

3.10.17 Exchange IMAP Test

When you install Microsoft Exchange Server 2013/2016, IMAP4 client connectivity is not enabled. To enable IMAP4 client connectivity, you need to start two IMAP services, the Microsoft Exchange IMAP4 service and the Microsoft Exchange IMAP4 Backend service. When you enable IMAP4,

Exchange 2013/2016 accepts unsecured IMAP4 client communications on port 143 and over port 993 using Secure Sockets Layer (SSL).

The Microsoft Exchange IMAP4 service runs on Exchange 2013/2016 computers that are running the Client Access server role. The Microsoft Exchange IMAP4 Backend service runs on the Exchange 2013/2016 computer that's running the Mailbox server role. In environments where the Client Access and Mailbox roles are running on the same computer, you manage both services on the same computer.

If clients connecting to the Exchange 2013/2016 via IMAP4 complain of slowness in the connections or frequent rejection of connections, administrators may want to figure out where the bottleneck is – is it because of a command processing bottleneck on the server? is it because of slow RPC or LDAP calls to the server? or is it due to a improper server configuration? The **Exchange IMAP** test provides accurate answers for these questions!

This test tracks SSL connections over IMAP4 and periodically reports the count of active, failed, and rejected connections. This way, the test alerts administrators to unusual spikes in rejections and failures. In addition, the test also measures the time taken by the server to process commands and to service LDAP and RPC calls to the IMAP4 service, thus accurately pinpointing the probable reasons for a high degree of latency in the IMAP4 connections.

Target of the test : A Microsoft Exchange 2013/2016 server

Agent deploying the test : An internal agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** – The port at which the host listens.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active SSL connections:	Indicates the the number of SSL or TLS connections that are	Number	This is a good indicator of the current SSL connection load on the service.

Measurement	Description	Measurement Unit	Interpretation
	currently open on the IMAP4 service.		
Average command processing time:	Indicates the average time taken to process commands.	Secs	A consistent increase in the value of this measure is a cause for concern, as it indicates a command processing bottleneck. In the event of a slowdown in request processing by the IMAP4 service, you may want to compare the value of this measure with that of the Average LDAP latency and Average RPC latency measures to determine what is exactly causing the processing delay.
Average LDAP latency:	Indicates the average time taken by an LDAP call to return results from the Mailbox server. .	Secs	A consistent increase in the value of this measure is a cause for concern, as it indicates a slowdown when processing LDAP calls. In the event of a slowdown in request processing by the IMAP4 service, you may want to compare the value of this measure with that of the Average command processing time and Average RPC latency measures to determine what is exactly causing the processing delay.
Average RPC	Indicates the average	Secs	A consistent increase in the

Measurement	Description	Measurement Unit	Interpretation
latency:	time it takes for a remote procedure call to return results from a Mailbox server.		value of this measure is a cause for concern, as it indicates a slowdown when processing RPC requests. In the event of a slowdown in request processing by the IMAP4 service, you may want to compare the value of this measure with that of the Average LDAP latency and Average command processing time measures to determine what is exactly causing the processing delay.
Current connections:	Indicates the total number of connections that are currently open on the IMAP4 service.	Number	This is a good indicator of the current connection load on the IMAP4 service.
Connections rate:	Indicates the rate at which clients connect to the IMAP4 service.	Connections/Sec	
Connections failed:	Indicates the number of connections that have failed since the last measurement period.	Number	Ideally, this value should be 0.
Connections rejected:	Indicates the number of connections that have been rejected since the last measurement period.	Number	Ideally, the value of this measure should be 0. A high value of this measure could indicate that many IMAP4 connections are being rejected by the server. One of the common reasons for this is a very low IMAP4 connection limit

Measurement	Description	Measurement Unit	Interpretation
			setting on the server. If the number of connections exceeds this limit, then subsequent connections will be rejected by the server. You may want to increase the Maximum IMAP4 connections that the server will accept at any point in time, so that no connections are rejected.
Current unauthenticated connections:	Indicates the number of current connections that are not authenticated.	Number	
Login failures:	Indicates the number of LOGIN commands that have failed since the last measurement period.	Number	Ideally, the value of this measure should be 0.
Logout failures:	Indicates the number of LOGOUT commands that have failed since the last measurement period.	Number	Ideally, the value of this measure should be 0.
Logout rate:	Indicates the number of LOGOUT commands per second.	Logouts/Sec	
Login total:	Indicates the total number of LOGIN commands that have been received since the last measurement	Number	

Measurement	Description	Measurement Unit	Interpretation
	period.		
SSL connections:	Indicates the total number of SSL connections to the IMAP4 service since the last measurement period.	Number	This is a good indicator of the total SSL connection load on the IMAP4 service.
Proxy current connections:	Indicates the current number of proxy connections open on the IMAP4 service.	Number	
Proxy connections failed:	Indicates the number of proxy connections to the IMAP4 service that failed since the last measurement.	Number	

Chapter 4: Monitoring the Exchange Client Access Server (CAS) 2013/2016

The Client Access server provides authentication, proxy, and limited redirection services, and offers all the usual client access protocols: HTTP, POP, IMAP, and SMTP.

To monitor the Client Access Server inside-out and bring abnormalities in its operations to the immediate attention of administrators, eG Enterprise provides a dedicated Microsoft Exchange CAS 2013/2016 monitoring model.

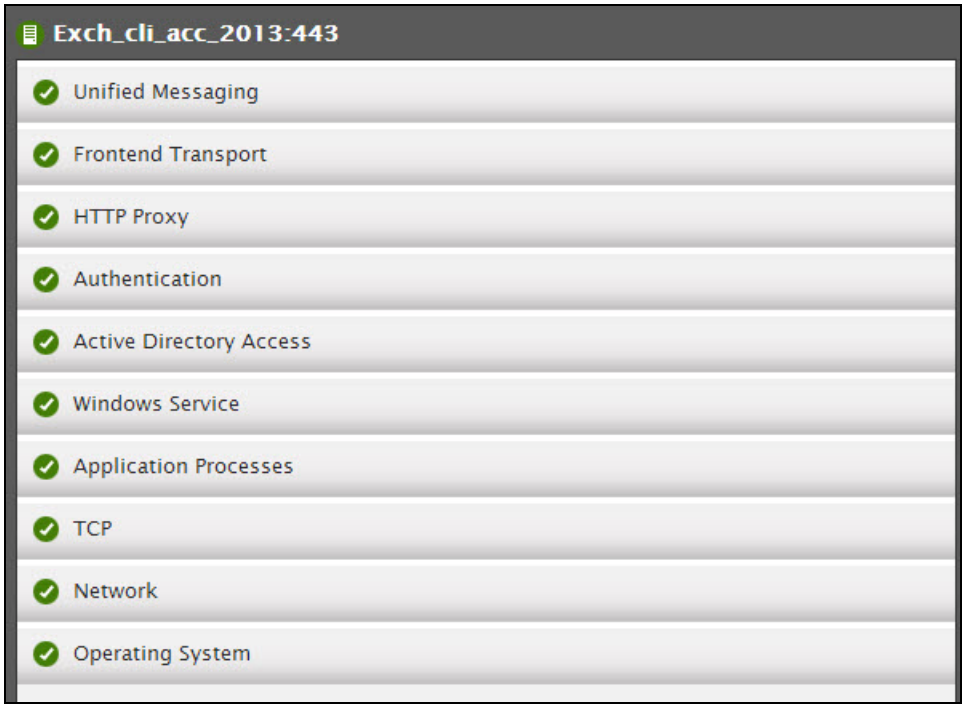

Figure 4.1: Layer model of the Microsoft Exchange CAS 2013/2016

Since each of the layers above and the tests mapped to them have been discussed elaborately in Chapter 1, this chapter will not be repeating the same.

Chapter 5: Monitoring Exchange Mailbox Servers 2013/2016

The Exchange 2013/2016 Mailbox server includes client access protocols, transport services, mailbox databases, and Unified Messaging services (the Client Access server redirects SIP traffic generated from incoming calls to the Mailbox server).

To focus on the health and overall performance of the Mailbox server and capture abnormalities, use the dedicated Microsoft Exchange Mailbox 2013/2016 monitoring model that eG Enterprise provides.

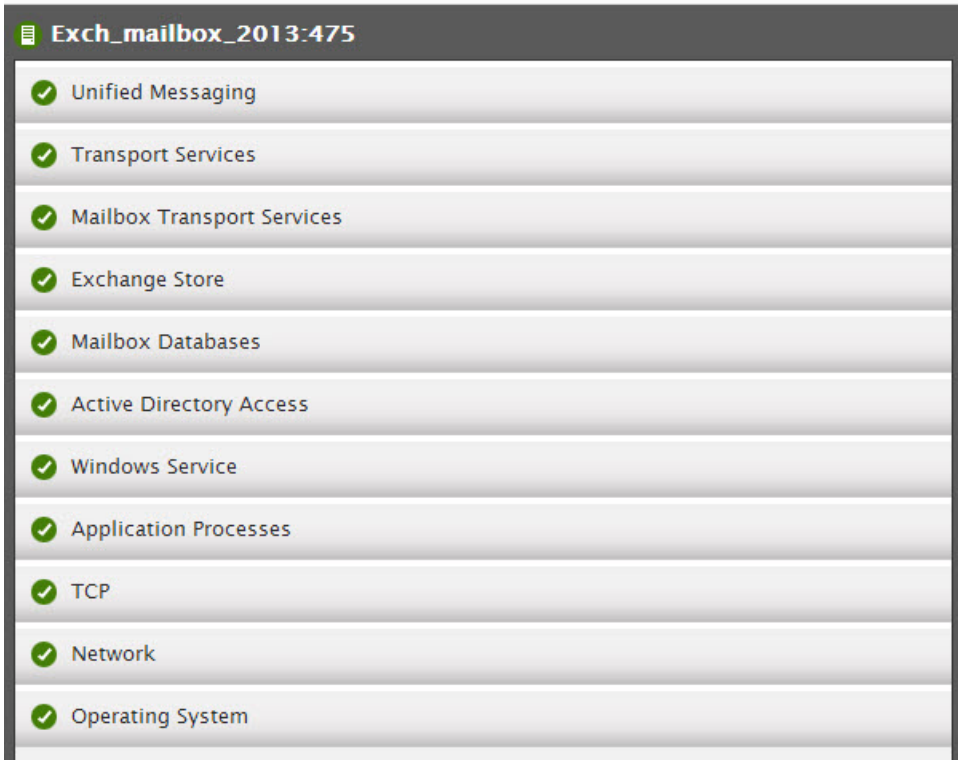

Figure 5.1: Layer model of the Microsoft Exchange Mailbox 2013/2016 server

In addition to the layers shown by Figure 5.1, this model includes an additional **Managed Availability** layer. This layer is positioned above the **Windows Service** layer in Figure 5.1. As all the layers depicted by Figure 5.1, including the **Managed Availability** layer, have been already discussed in Chapter 1 of this document, we will not be repeating the discussion in this chapter.

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.