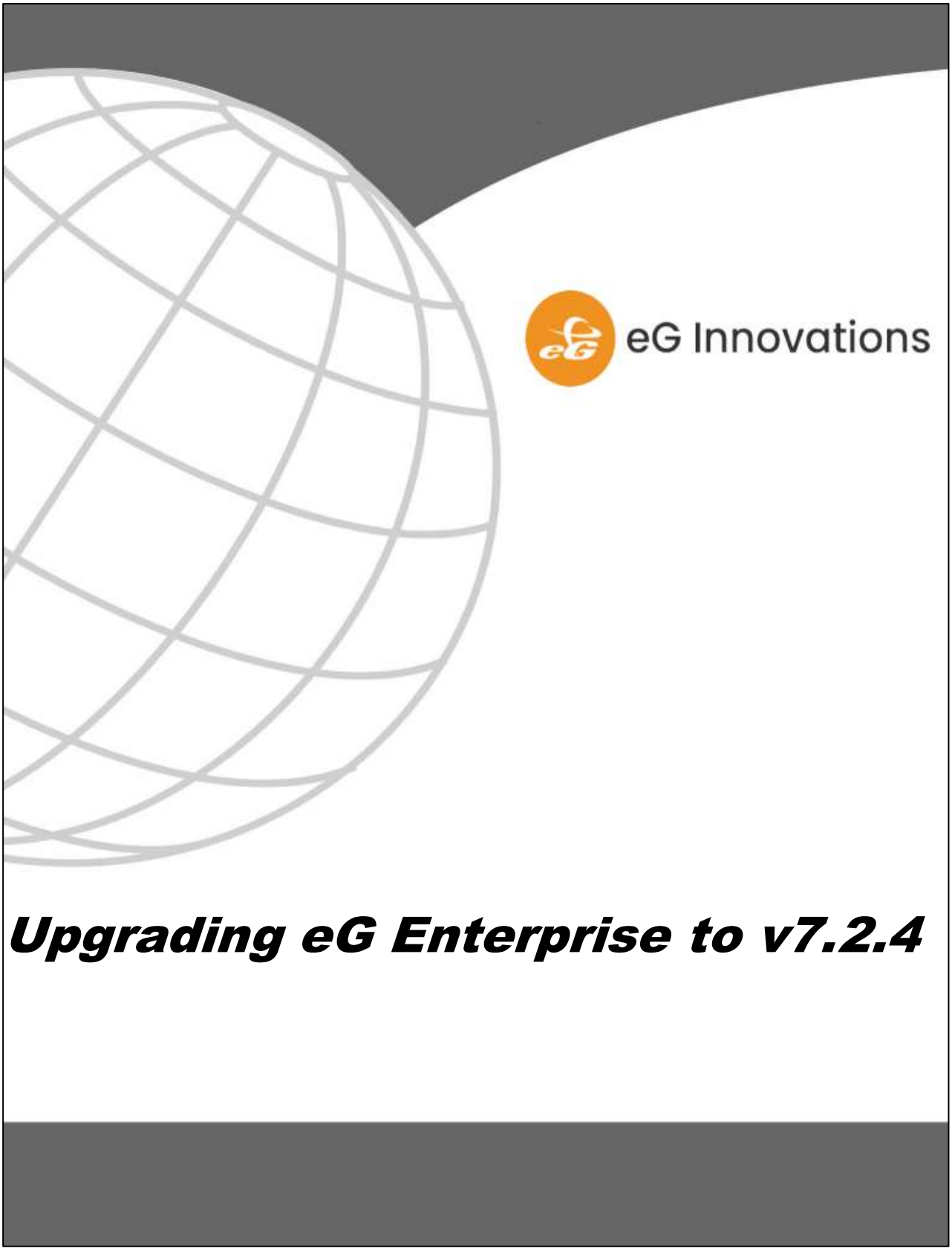

Upgrading eG Enterprise to v7.2.4

Table of Contents

Upgrading eG Enterprise to v7.2.4

Upgrades to the eG manager add new features and enhancements to the eG manager. From time to time, eG Innovations provides its existing customers who are covered under the eG Annual Software Maintenance (EASM) program with upgrades to the latest released version of the eG Enterprise Real-time monitoring and Proactive Infrastructure Triage solution. If you are not covered by EASM, please contact sales@eginnovations.com for details on how you can sign up for this program.

This document provides you with the guidelines for upgrading the eG Enterprise suite from version 7.1.4 (or higher) to v7.2.4.

If the eG manager that is in use in your environment is of a version lower than eG 7.1.4, please contact support@eginnovations.com to obtain a prior upgrade.

The broad steps towards upgrading the manager to version 7.2.4 are as follows:

- Take a backup of the eG Enterprise Suite
- Ensure that the pre-requisites for the upgrade are fulfilled
- Upgrade the eG manager
- Upgrade the eG agents

The sections that follow will discuss each of these steps in great detail.

1.1 Taking a Backup of the eG Enterprise Suite

Prior to performing an upgrade, it is recommended that you take a backup of the eG Enterprise suite (the eG manager and external agent), so that you can always revert to it. The backup and restore procedures vary according to the operating system of the eG manager.

In case of Windows environments, a minimal backup for upgrade purposes should include the following:

- A backup of the eG install directory
- A backup of the eG database

Kindly refer to Chapter 7 of the *eG Installation Guide* for taking a backup of the eG directory and database. An elaborate restoration procedure is also provided here.

1.2 Pre-requisites for Upgrading the eG Manager

Before upgrading the eG manager to v7.2.4, make sure that the following are in place:

- Ensure that the eG manager is of version 7.1.4;
- Some of the older versions of the eG manager use TLS v1. Starting with v7.2.4 however, this protocol is no longer supported. Instead, TLS v1.2 is supported. This is why, if an eG manager that is using TLSv1 is being upgraded to v7.2.4, the upgrade will fail with the following error message:

```
WELCOME TO 7.2.4 UPGRADE

=========================


*********************************************************

 Upgrade process might take several minutes to complete.

 PLEASE DO NOT INTERRUPT THIS PROCESS.

*********************************************************

Checking the Manager Version to start the upgrade

eG manager being upgraded is using TLSv1. eG Enterprise v7.2.4 does not support
this protocol. Please configure your eG manager with TLSv1.2, and then start
the upgrade process.


Press any key to continue . . .
```

To avoid this failure, before attempting the upgrade, do the following:

a. Check whether the eG manager being upgraded is using TLSv1;

b. If so, reconfigure that eG manager with TLSv1.2.

c. Finally, start upgrading the eG manager to v7.2.4.

To confirm the TLS version used by the eG manager, do the following:

a. Login to the system hosting the eG manager.

b. Open the server.xml file in the <EG_INSTALL_DIR>\manager\tomcat\conf\ directory (on Windows; on Unix, this will be /opt/egurkha/manager/tomcat/conf/).

c. In the file, look for the entries highlighted below:

```
<Connector executor="tomcatThreadPool"
protocol="org.apache.coyote.http11.Http11Nio2Protocol"

port="443"

sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
scheme="https" secure="true" SSLEnabled="true"

enableLookups="false" acceptCount="6000" maxConnections="1200"
MaxKeepAliveRequests="1" connectionTimeout="300000"
socket.rxBufSize="131072" socket.txBufSize="131072"
maxHttpHeaderSize="131072" URIEncoding="UTF-8" useSendfile="false"
tcpNoDelay="false" compression="force"
compressibleMimeType="font/woff,font/woff2,text/html,text/xml,text/plain,a
```

```
pplication/x-java-applet,application/octet-
stream,application/xml,text/javascript,text/css,image/png,image/bmp,image/
jpeg,image/gif,application/pdf,application/x-
javascript,application/javascript,application/json,application/x-
shockwave-flash,application/xhtml+xml,application/xml+xhtml"
noCompressionUserAgents="gozilla, traviata" server="eG Manager">

<SSLHostConfig  honorCipherOrder="true"  ciphers="
TLS_RSA_WITH_AES_128_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA
_WITH_AES_256_CBC_SHA " certificateVerification="none" protocols="TLSv1"
sslProtocol="TLS"><Certificate
certificateKeystoreFile="webapps/egmonitor.tpa.ebay.com.pfx"
certificateKeystorePassword="${EgSslPassword}" /></SSLHostConfig>

 </Connector>
```

d. If you find these entries, then you can conclude that the eG manager is using TLS v1.

Once you have confirmed that the TLS version is 1, proceed to reconfigure the eG manager with TLSv1.2. For that, follow the steps below:

a. Stop the eG manager.

b. Login to the system hosting the eG manager.

c. Open the server.xml file in the <EG_INSTALL_DIR>\manager\tomcat\conf\ directory (on Windows; on Unix, this will be /opt/egurkha/manager/tomcat/conf/).

d. In the file, look for the entries highlighted below:

```
<Connector executor="tomcatThreadPool"
protocol="org.apache.coyote.http11.Http11Nio2Protocol"

port="443"

sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
scheme="https" secure="true" SSLEnabled="true"

enableLookups="false" acceptCount="6000" maxConnections="1200"
MaxKeepAliveRequests="1" connectionTimeout="300000"
socket.rxBufSize="131072" socket.txBufSize="131072"
maxHttpHeaderSize="131072" URIEncoding="UTF-8" useSendfile="false"
tcpNoDelay="false" compression="force"
compressibleMimeType="font/woff,font/woff2,text/html,text/xml,text/plain,a
pplication/x-java-applet,application/octet-
stream,application/xml,text/javascript,text/css,image/png,image/bmp,image/
jpeg,image/gif,application/pdf,application/x-
javascript,application/javascript,application/json,application/x-
shockwave-flash,application/xhtml+xml,application/xml+xhtml"
noCompressionUserAgents="gozilla, traviata" server="eG Manager">

<SSLHostConfig  honorCipherOrder="true"  ciphers="
TLS_RSA_WITH_AES_128_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA
_WITH_AES_256_CBC_SHA " certificateVerification="none" protocols="TLSv1"
sslProtocol="TLS"><Certificate
certificateKeystoreFile="webapps/egmonitor.tpa.ebay.com.pfx"
certificateKeystorePassword="${EgSslPassword}" /></SSLHostConfig>
```

```
</Connector>
```

e. Then, replace the **ciphers=** entry highlighted at step 4 above, with the following

```
ciphers="TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,TLS_ECDHE_ECDSA_WITH_AES_25
6_GCM_SHA384,TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,TLS_ECDH_RSA_WITH_AES_
256 GCM SHA384,TLS ECDHE RSA WITH AES 128 GCM SHA256,TLS ECDHE RSA WITH AE
S_256_CBC_SHA384,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_
AES_256_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_
256_CBC_SHA"
```

f. Next, replace the **protocols=** entry highlighted at step 4 above, with the following:

```
protocols="TLSv1.2"
```

g. Finally, save the file and start the eG manager.

h. Then, check whether the eG agents are continuing to report metrics to the eG manager. If they are, then proceed to upgrade the eG manager.

- If the eG manager to be upgraded is SSL-enabled, but the default SSL certificate bundled with the eG manager was not used for SSL-enabling the manager (i.e., a certificate signed by an internal CA or a self-signed certificate was used for SSL-enabling instead), then first check whether/not the certificate **keystore** file is stored in the <EG_INSTALL_DIR>\manager\tomcat\webapps folder (on Windows; on Unix, this will be: opt/egurkha/manager/tomcat/webapps) on that manager. If not, then do the following:

  - Copy the keystore file from its current location to the <EG_INSTALL_DIR>\manager\tomcat\webapps folder (on Windows; on Unix this will be: opt/egurkha/manager/tomcat/webapps).
  - Then, edit the **server.xml** file (in the <CATALINA_HOME>\conf directory) on the eG manager host.
  - If the eG manager to be upgraded is of version 7.1.4 (or above), then change the **certificateKeystoreFile** parameter in the **server.xml** file to "webapps/<keystorefile_name>". For instance, if the name of your keystore file is **eGmanager.bin**, then your specification will be:

    certificateKeystoreFile="webapps/eGmanager.bin"

  - Then, save the file.
  - Restart the eG manager.
  - Check whether the manager is functioning properly.
  - Then, proceed to upgrade the eG manager.

  On the other hand, if the keystore file is already located in the <EG_INSTALL_DIR>\manager\tomcat\webapps folder only, then you do not have perform any of the steps above. You can simply proceed to upgrade the manager.

1.3 Upgrading the eG Manager on Linux

1. To begin the manager upgrade, first, connect to the URL: http://www.eginnovations.com/Upgrade724/
2. Click on the **Manager** folder within. In this folder, you will find sub-folders named after each operating system that is supported by the eG manager. Every operating system-specific folder will contain the upgrade package for upgrading the eG manager on that operating system.
3. If you click on the **Linux** directory, the following files will be listed:

eGupgrade.sh script file

eG_7.2.4.tar.gz file

eg_upgrade.jar

4. For the purpose of upgrading, the eG manager to v7.2.4, you need to download all the files mentioned above to a convenient location on the eG manager host.

Remember to unzip the **eG_7.2.4.tar.gz** file using the **gunzip** command once it is downloaded.

5. Next, grant **execute** permissions for the script file **eGupgrade.sh** using the following command:

   chmod +x eGupgrade.sh

6. Also, provide the user permission to execute the **eG_7.2.4.tar**, the **eg_upgrade.jar**, and the **eGupgrade.sh**, using the following commands:

   chown <eG Installed User>:<Group Name> eG_7.2.4.tar

   chown <eG Installed User>:<Group Name> eGupgrade.sh

   chown <eG Installed User>:<Group Name> eg_upgrade.jar

   For example, **chown john:egurkha eG_7.2.4.tar**.

7. Then, proceed to execute the file **eGupgrade.sh** using the following command:

   ./eGupgrade.sh -<option>, where <option> can be any one of the following:

   - **u** - to upgrade the eG manager
   - **r** - to revert to the previous version of the manager
   - **c** - to commit the manager upgrade changes

An eG install user alone should execute the upgrade.

8. If you do not specify the parameter **-<option>**, the following message will appear, prompting you to provide any one of the above-mentioned parameters:

```
Usage : ./eGupgrade.sh OPTION
•  u ..... Upgrade the eG Manager
•  r ..... Revert the eG Manager to its previous version
•  c ..... Confirm and Commit upgrade changes and enable agent auto-upgrade
```

9. To trigger the manager upgrade process, provide the following command:

   ./eGupgrade.sh -u

   If the user executing the script is **not** the eG install user, then you will receive the following message immediately upon execution, and will not be permitted to proceed with the upgradation:

```
Please login as '<eG install user>' and run this script!
**************************************************************************
```

10. Upon execution, the command will perform the following processing:

```
WELCOME TO 7.2.4 UPGRADE
========================
Starting upgrade of the eG Manager ...
*****************************************************
The upgrade process will take several minutes to complete.
PLEASE DO NOT INTERRUPT THIS PROCESS.
*****************************************************
Stopping the eG Manager ...
*****************************************************************
The eG Manager has been successfully stopped.
*****************************************************************
Stopping the eG Agent ...
*********************************************
The eG agent has been stopped successfully.
*********************************************
Extracting the files required for Upgrade...
Find free space in your eGurkha install directory...
Backup of the eG Manager started ...
Backup of the eG Manager completed
Upgrading the eG Manager configuration ...
Upgrading necessary config files...
Configuration files upgraded successfully!!!
Upgrading the eG database ...
Upgraded the eG database ...
The Lib files has been updated successfully!!
Copying Ic specific Images
updated manifest
```

11. If the manager upgrade completes successfully, you will view the following messages:

```
*************************************************************************
The eG Manager upgrade has been completed successfully!!!
Copy the new 7.2.4 license to /opt/egurkha/bin directory .
Execute the command /opt/egurkha/bin/start_manager to start the eG Manager.

If a Default Agent is present, please start the eG Agent
using the command /opt/egurkha/bin/start_agent.

*****************************************************************
To complete the upgrade process, you will need to reindex the database backend and
add new indexes for faster access.

Analysing Database size and indexes... Please wait...
```

12. When analyzing database size, if the upgrade process finds the size to be above 500 GB, then it will automatically provide you with the following database optimization recommendations.

```
Total size of the database is :

The size of the eG backend database is > 500 GB. Therefore, we recommend that you perform database indexing in 2 phases


Phase 1: For good performance of the eG manager, you must reindex the eG database. To do this in online mode open the file <EG_HOME>/bin/QueryManualExecuteRebuildIndex_ONLINE.sql and execute all the queries in this file manually, one by one. These queries, when executed, reindex the top tables (in terms of size). To reindex in offline mode (more time consumed but efficient), execute all the queries in the file <EG_HOME>/bin/QueryManualExecuteRebuildIndex_OFFLINE.sql.


Phase 2: Please execute the file <EG HOME>/bin/UpgradeIndex.bat to create new indexes in the database that are required to complete the upgrade.
```

On the other hand, if the upgrade process finds database size to be less than 500 GB, it will offer the following recommendations:

```
Total size of the database is :

To complete the upgrade process, you will need to reindex the database backend and add new indexes for faster access. Note that this activity may take up to 4 hours (the actual value could vary depending on the size of the database and the current fragmentation level).

1) Run the script /opt/egurkha/bin/ExecuteIndexes.sh offline to reindex the database.

2) Run the script /opt/egurkha/bin/UpgradeIndex.sh to create new indexes in the database that are required to complete the upgrade.

*******************************************************************************
```

For enhanced database performance**, it is mandatory that you follow the optimization instructions given by the upgrade process**, after you upgrade the eG manager and agents. To understand these database optimization options better, refer to Section 1.20 of this document.

13. If upgrade succeeds, copy the v7.2.4 license to the **/opt/egurkha/bin** directory as indicated by the message above.
14. After successful upgrade, it is recommended that you index the eG database to optimize its performance. The detailed steps for this are available in the Database Optimizations section of this document.
15. Then, start the manager using the **/opt/egurkha/bin/start_manager** command.

- After starting the manager, clear the browser history of the browser that you would be using to connect to the eG management console. Before clearing the history, make sure that you configure the cleanup process to remove all the browsing and download histories, cookies and plugins, and cached images and files, from the beginning of time. Figure 1 indicates how this configuration is to be performed on a Chrome browser.

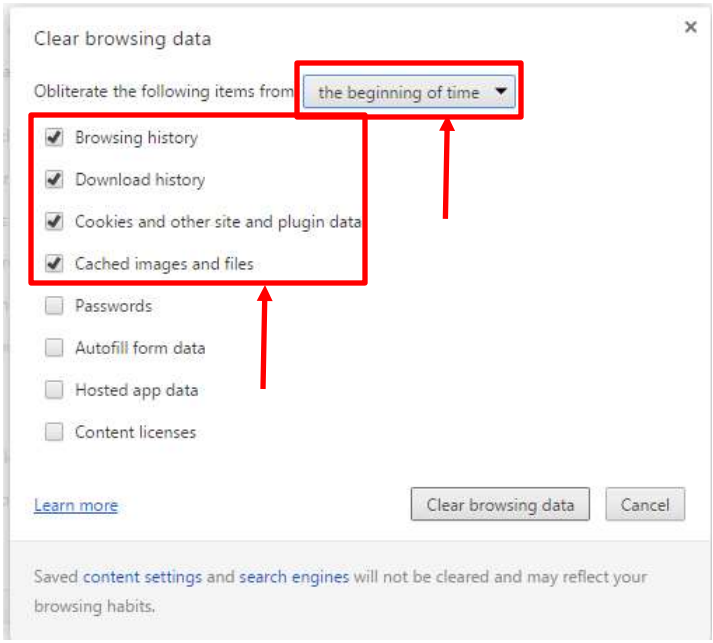

Figure 1: Configuring the browser history cleanup

- When the manager is started, email alerts will once again be sent out for all performance issues that prevailed prior to the upgrade.
- Log into the admin interface of the eG manager and verify whether all the basic configurations (such as segment topologies defined, servers managed/added, sites configured etc., in the previous version) are intact.
- Do not perform any further configurations using the eG administrative interface before committing the manager upgrade.
- Do not use any of the new features in v7.2.4 before committing the upgrade.

16. The manager upgrade process stops the eG agent running on the manager system. Therefore, start the agent using the command: **/opt/egurkha/bin/start_agent**
17. If for some reason you want to revert to the previous version of the manager after the upgrade, then use the command **/eGupgrade.sh –r**. If reverting is successful, the following message will appear:

```
**************************************************************************
Reverting the eG Manager to its original configuration ...
**************************************************************************
Revert process might take several minutes to complete.
PLEASE DO NOT INTERRUPT THIS PROCESS.
**************************************************************************
```

```
Stopping the eG Manager ...
****************************************************************
The eG Manager has been successfully stopped.
****************************************************************
Stopping the eG Agent ...
***********************************************
The eG agent has been stopped successfully.
***********************************************
Reverting database...
************************************************************
Successfully reverted the eG Manager to the previous version!
************************************************************
Please execute the command /opt/egurkha/bin/start_manager
and /opt/egurkha/bin/start_agent to start the eG Manager and eG Agent.
```

18. On the other hand, if you want to proceed with the upgrade, commit the changes made to the eG manager, by executing the command: **./eGupgrade.sh –c**.

19. Upon successful completion of the commit, the following message will appear:

```
****************************************************************
Successfully committed the eG Manager upgrade to version 7.2.4!
```

It is not possible to revert the eG manager, once the manager upgrade is committed.

1.4 Upgrading the eG Manager on Windows

To upgrade the eG manager on Windows, do the following:

1. Go to the URL: http://www.eginnovations.com/Upgrade724/
2. Click on the **Manager** folder within. In this folder, you will find sub-folders named after each operating system that is supported by the eG manager. Every operating system-specific folder will contain the upgrade package for upgrading the eG manager on that operating system. For instance, the contents of the **Windows2008** folder will enable you to upgrade an eG manager on a Windows 2008 server.

   Each of these Windows folders will contain the following files:

   - The batch file **eGupgrade.bat**
   - The jar file **Extract.jar**
   - The **eG_<OS>_eG7.2.4.zip** file
   - The **eg_upgrade.jar** file

3. For the purpose of upgrading the eG manager to v7.2.4, download the aforesaid files from the corresponding Windows folder to a convenient location on your local disk (say, **c:\tmp**):
4. The agent upgrade procedure begins only after the eG manager is upgraded successfully and the changes are committed. Even while the agent upgrade is in progress, the old (i.e., agents of a previous version) eG agents in the target environment, will continue to report measurement data to the upgraded eG manager.

5. Then proceed to execute the upgrade batch file eGupgrade.bat using the following command from the command prompt:

eGupgrade.bat

6. The program will request you to choose from the following options:

```
WELCOME TO 7.2.4 UPGRADE

========================

Enter Your Option :

U - Upgrade  R - Revert  C - Commit E - Exit] ?
```

U - to upgrade the eG manager

R - to revert to the previous version of the manager

C - to commit the manager upgrade changes and to initiate the agent upgrading process

E – to exit the upgrade menu.

To trigger the manager upgrade process, enter **U**.

7. Upon execution, the following messages will appear:

```
WELCOME TO 7.2.4 UPGRADE
=========================
*********************************************************
 Upgrade process might take several minutes to complete.
 PLEASE DO NOT INTERRUPT THIS PROCESS.
*********************************************************
Checking the Manager Version to start the upgrade
Extracting the files required for Upgrade...
Checking free space in your egurkha install directory..........
Starting upgrade of the eG Manager ...
Stopping the eG Agent ...
*********************************
The eGAgentMon service stopped..
*********************************
**********************************
The eGurkhaAgent service stopped...
**********************************
*********************************************
The eG Agent has been successfully stopped.
*********************************************
Stopping the eG Manager...
**********************************
The eGMon service stopped..
**********************************
*****************************************
The eGurkhaTomcat service stopped...
*****************************************
*********************************************
The eG Manager has been successfully stopped.
*********************************************
Backup of the eG Manager started ...
Backup of the eG Manager completed
Upgrading the eG Manager configuration ...
Upgrading necessary config files...
Configuration files upgraded successfully!!!
Upgrading the eG database ...
Upgraded the eG database ...
```

```
*********************************************
MyDashboard Upgrade started.................
*********************************************
MyDashboard Upgrade Skipped as My Dashboard upgrade is not required for this manager
version---->724
Copying IC specific Images
Ic specific image files are copied
```

8. When the manager upgrade completes successfully, you will view the following message:

```
*****************************************************************
The eG Manager upgrade has been completed successfully!!!
Copy the new v7.2.4 license to the C:\eGurkha\bin directory.
Execute the command C:\eGurkha\lib\start_manager to start the eG Manager.

If an External Agent is present, please start the eG Agent
using the command C:\eGurkha\lib\start_agent.
*****************************************************************
*****************************************************************
To complete the upgrade process, you will need to reindex the database backend and
add new indexes for faster access.

Analysing Database size and indexes... Please wait...
```

9. When analyzing database size, if the upgrade process finds the size to be above 500 GB, then it will automatically provide you with the following database optimization recommendations.

```
Total size of the database is :

The size of the eG backend database is > 500 GB. Therefore, we recommend that you
perform database indexing in 2 phases


Phase 1: For good performance of the eG manager, you must reindex the eG database. To
do this in online mode open the file
<EG HOME>/bin/QueryManualExecuteRebuildIndex ONLINE.sql and execute all the queries
in this file manually, one by one. These queries, when executed, reindex the top
tables (in terms of size). To reindex in offline mode (more time consumed but
efficient), execute all the queries in the file
<EG_HOME>/bin/QueryManualExecuteRebuildIndex_OFFLINE.sql.


Phase 2: Please execute the file <EG_HOME>/bin/UpgradeIndex.bat to create new indexes
in the database that are required to complete the upgrade.
```

On the other hand, if the upgrade process finds database size to be less than 500 GB, it will offer the following recommendations:

```
Total size of the database is :


To complete the upgrade process, you will need to reindex the database backend and
add new indexes for faster access. Note that this activity may take up to 4 hours (the
actual value could vary depending on the size of the database and the current
fragmentation level).

1) Run the script <EG_HOME>\bin\ExecuteIndexes.bat offline to reindex the database.

2) Run the script <EG_HOME>\bin\UpgradeIndex.bat to create new indexes in the database
that are required to complete the upgrade.
```

****

For enhanced database performance**, it is mandatory that you follow the optimization instructions given by the upgrade process**, after you upgrade the eG manager and agents. To understand these database optimization options better, refer to Section 1.20 of this document.

10. If upgrade succeeds, copy the v7.2.4 license to the **<EG_INSTALL_DIR>\bin** directory as indicated by the message above.
11. After successful upgrade, it is recommended that you index the eG database to optimize its performance. The detailed steps for this are available in the Database Optimizations section of this document.
12. Then, start the manager using the menu sequence, eG Monitoring Suite -> eG Manager -> Start Manager.

- After starting the manager, clear the browser history of the browser that you would be using to connect to the eG management console. Before clearing the history, make sure that you configure the cleanup process to remove all the browsing and download histories, cookies and plugins, and cached images and files, from the beginning of time. Figure 2 indicates how this configuration is to be performed on a Chrome browser.

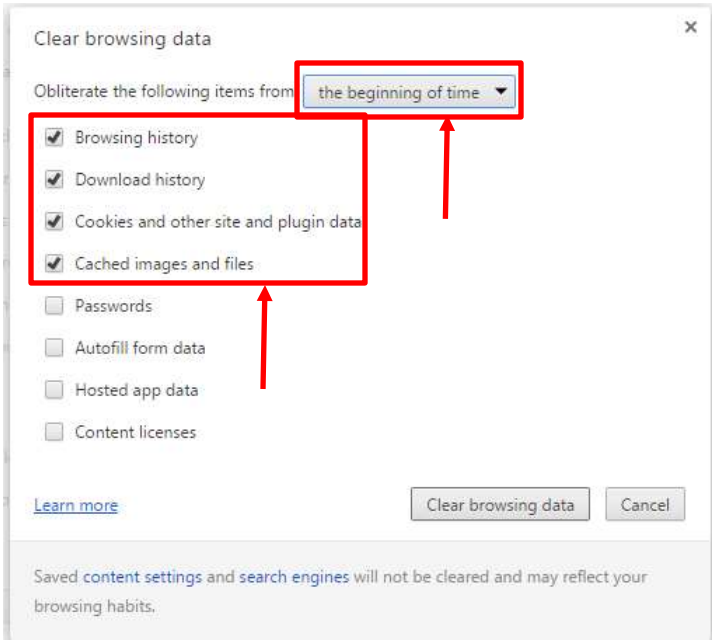

Figure 2: Configuring the browser history cleanup

Note

- When the manager is started, email alerts will once again be sent out for all performance issues that prevailed prior to the upgrade.
- Log into the admin interface of the eG manager and verify whether all the basic configurations (such as segment topologies defined, servers managed/added, sites configured etc., in the previous version) are intact.
- Do not perform any further configurations using the eG administrative interface before committing the manager upgrade.
- Do not use any of the new features in v7.2.4 before committing the upgrade.

13. The manager upgrade process stops the eG agent running on the manager system. Therefore, start the agent using the following menu sequence:

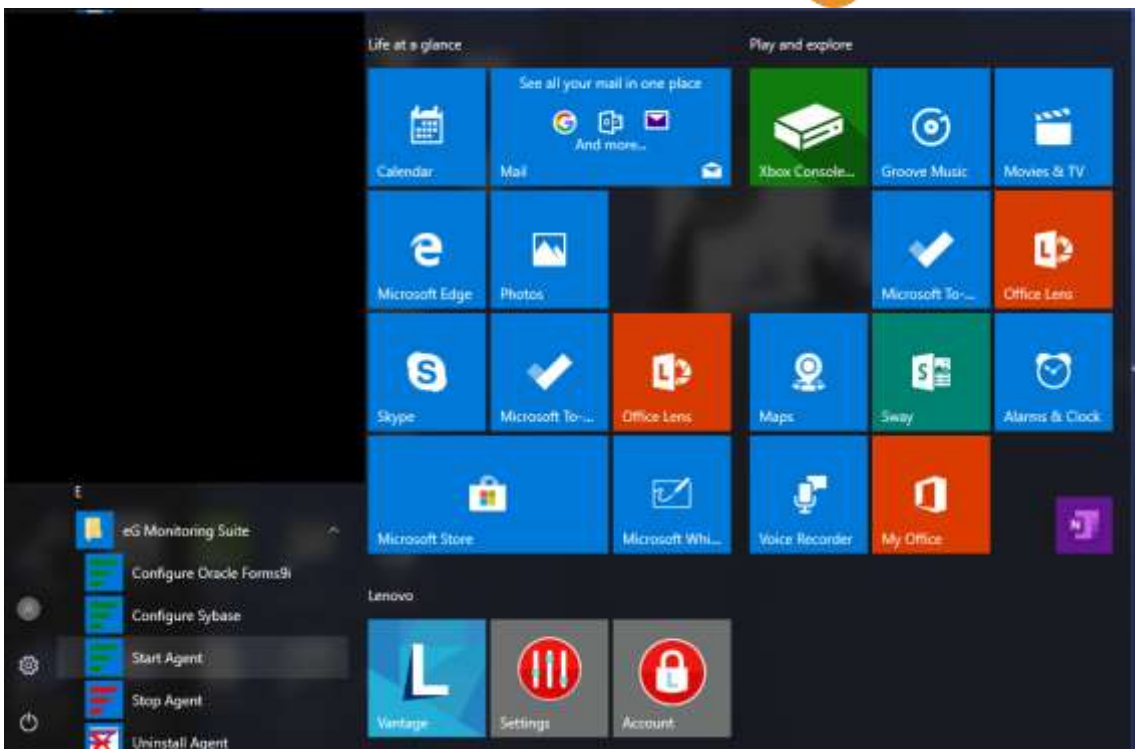

Figure 3: Starting the eG agent

14. If for some reason you want to revert to the previous version of the manager after the upgrade, then execute the command **eGupgrade.bat** once again, but this time select the **R** option.

Note

- If the web server is being monitored on the eG manager host, then, stop the web server **before initiating the revert operation**.
- Make sure that all files and folders that are open on the eG manager host and all operations that you may have started on the host are closed/stopped before attempting the revert.

15. If reverting is successful, the following message will appear:

```
WELCOME TO REVERT OPERATION
===========================
Reverting the eG Manager to its original configuration.
*****************************************************
Revert process might take several minutes to complete.
PLEASE DO NOT INTERRUPT THIS PROCESS.
*****************************************************
Stopping the eG Manager ...
***********************************
The eGMon service stopped..
***********************************

***************************************
The eGurkhaTomcat service stopped...
***************************************
```

```
*********************************************
The eG Manager has been successfully stopped.
*********************************************
Stopping the eG Agent ...
The eGAgentMon service is not started.

More help is available by typing NET HELPMSG 3521.

*******************************
The eGAgentMon service stopped..
*******************************
*******************************
The eGurkhaAgent service stopped...
*******************************
*********************************************
The eG Agent has been successfully stopped.
*********************************************
        1 dir(s) moved.
Reverting Database...

************************************************************
Successfully reverted the eG Manager to the previous version!
Please execute the command C:\eGurkha\lib\start_manager
and C:\eGurkha\lib\start_agent to start the eG Manager and eG Agent.
************************************************************
```

Sometimes, during a revert operation on Windows, you may find that the command prompt hangs, passively waiting for user inputs. In such cases, do the following to resume the revert operation:

- Open the **<EG_INSTALL_DIR>\manager\logs\upgrade_event** file.
- Scan the file for the following entry:

```
Does D:\egurkha\bin specify a file name or directory name on the
target (F=file,D=directory)?
```

- If you find it, then close the file, return to the command prompt, and type **D** therein. The revert operation will then continue without a glitch.

16. On the other hand, if you want to proceed with the upgrade, commit the changes made to the eG manager, by executing the eGupgrade.bat command yet again, and selecting the option C.
17. Upon successful completion of the commit, the following message will appear:

```
WELCOME TO COMMIT OPERATION
===========================

Committing the eG Manager Upgrade Changes ....

Stopping the eG Manager...
***********************************
The eGMon service stopped..
***********************************
***************************************
The eGurkhaTomcat service stopped...
***************************************
*********************************************
The eG Manager has been successfully stopped.
*********************************************
```

```
Committing the eG Database...

****************************************
 Successfully Committed the eG Manager Upgrade.
```

It is not possible to revert the eG manager, once the manager upgrade is committed.

1.5 Troubleshooting the Manager Upgrade

If the manager upgrade fails, then check the **<EG_INSTALL_DIR>\manager\logs\upgrade_event** file (on Windows; on Linux, this will be the **/opt/egurkha/manager/logs/upgrade_event** file) for any errors during the upgrade. Also, check the **<EG_INSTALL_DIR>\tmp\logging** file (on Windows; on Linux, this will be the **/opt/egurkha/tmp/logging** file) or error messages.

For any further clarifications, mail support@eginnovations.com enclosing the above-mentioned files.

1.6 Upgrading the eG Agents

Soon after committing the manager upgrade, do the following:

1. Login to the eG manager host.
2. Go to the URL: http://www.eginnovations.com/Upgrade724/Agent/
3. The agent upgrade patch for each operating system will be available therein. The table below lists the patch files and the operating systems they apply to:

OS	Agent Upgrade Pack
AIX	ega_aix_724_1I.zip
HPUX	ega_hpx_724_1I.zip
Linux	ega_lin_724_1I.zip
Solaris	ega_sol_724_1I.zip
Win2008	ega_win2008_724_1I.zip
Win2012	ega_win2012_724_1I.zip
Win2016	ega_win2016_724_1I.zip
Win2019	ega_win2019_724_1I.zip

4. Download each of the zip files present here to the corresponding OS-specific folders that exist within the **<EG_MANAGER_INSTALL_DIR>\manager\config\tests** directory.

When the eG agents check the manager for the existence of upgrade patches, they will automatically download the patch files and install them.

If you want to enable this capability for specific agents only, first login to the eG manager's administrative interface as user *admin* with password admin. Then, select the **Settings** option from the **Upgrade** menu in the **Agents** tile.

Figure 4 to Figure 8 depict how the auto upgrade capability is enabled for an agent. Using these pages administrators can perform the following tasks:

- Enable the auto upgrade capability for specific agents or all of them, as required
- Specify the frequency with which the agents will check the manager for upgrades
- Select the agent that needs to be upgraded immediately

This page (see Figure 4) appears when the **Settings** option of the **Upgrade** menu in the **Agents** tile is clicked.

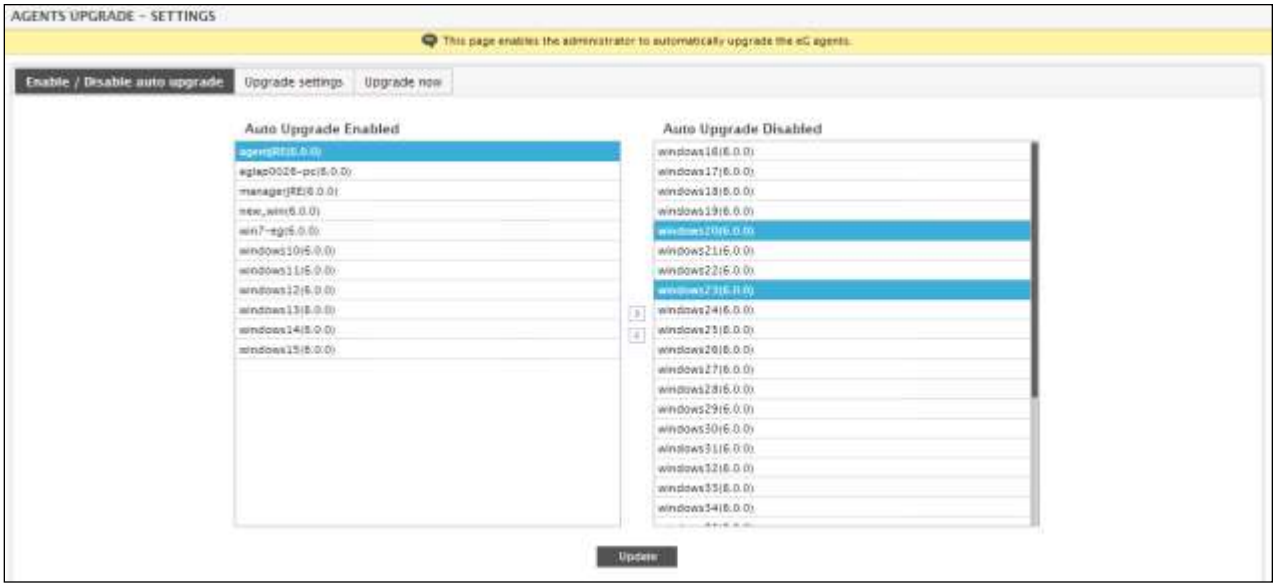

Figure 4: Selecting the agent for which the auto upgrading capability is to be enabled

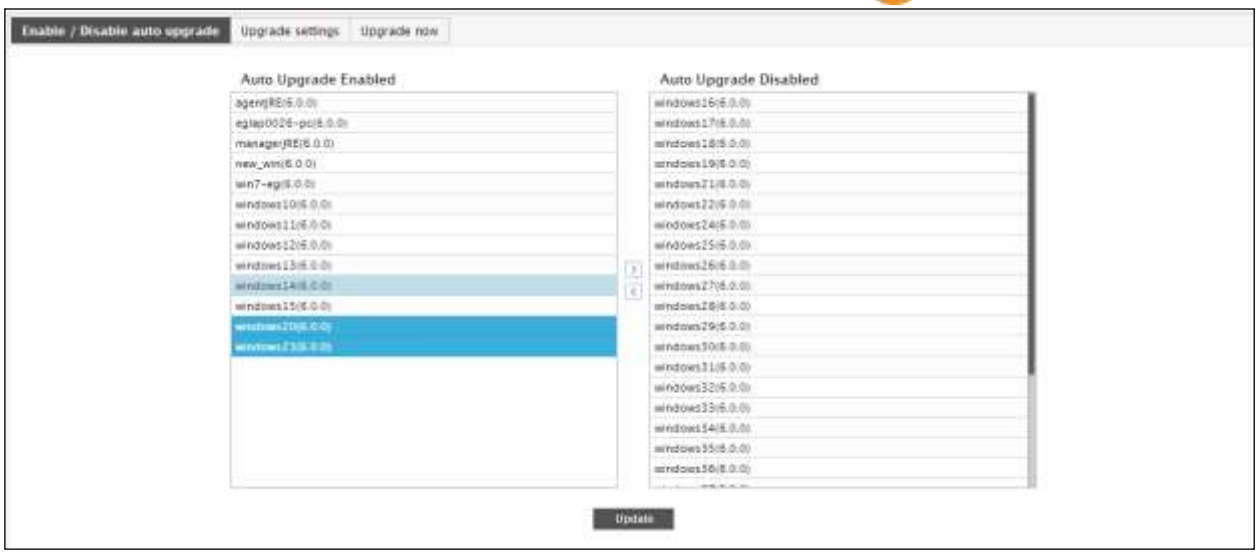

Figure 5: Enabling the auto upgrading capability for an agent

To enable the auto upgrade capability for specific agents, do the following:

1. From the **Auto Upgrade Disabled** list in the **Enable/Disable auto upgrade** tab page (see Figure 4), select the agent(s) for which the auto upgrade capability is to be enabled.
2. Then, click the **<** button to transfer the selection to the **Auto Upgrade Enabled** list (see Figure 5).
3. To disable this capability later, select the agent(s) from the **Auto Upgrade Enabled** list, click the **>** button, and transfer the selection back to the **Auto Upgrade Disabled** list.

To specify the frequency with which the agent should check the manager for upgrades, do the following:

1. Click on the **Upgrade settings** tab page as depicted by Figure 6.
2. Then, from the **How often agents should check for auto upgrade package** list box (see Figure 6), select the time interval at which the agents (for which auto upgrade has been enabled) need to check the manager for upgrades.

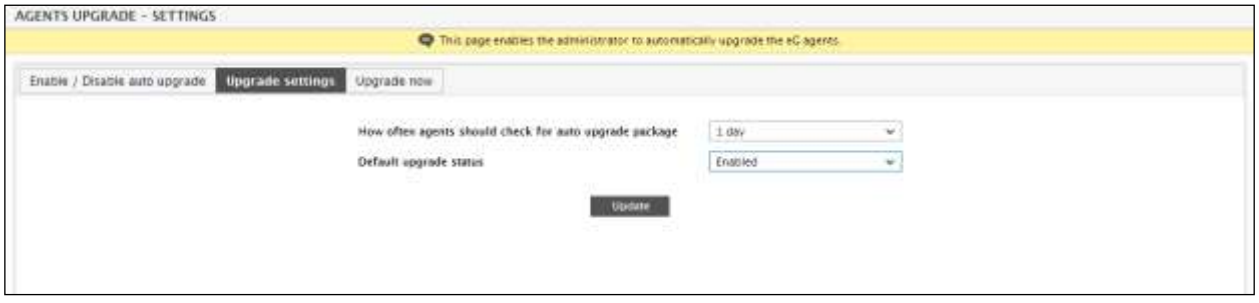

Figure 6: Specifying the upgrade interval

3. By selecting the **Enabled** or **Disabled** option from the **Default upgrade status** list box, administrators can indicate whether auto upgrade is, by default, enabled/disabled for new agents to the eG Enterprise system (see Figure 6).
4. Finally, click the **Update** button (see Figure 6).

To upgrade agents within the next 15 minutes, do the following:

1. Click on the **Upgrade now** tab page as shown by Figure 7.
2. All the agents for which auto-upgrade has been enabled will then appear in the **Auto Upgradeable Agents** list in

Figure 7.

3. From this list, select the agents which need to be auto-upgraded within the next 15 minutes, and click the < button (see Figure 7).

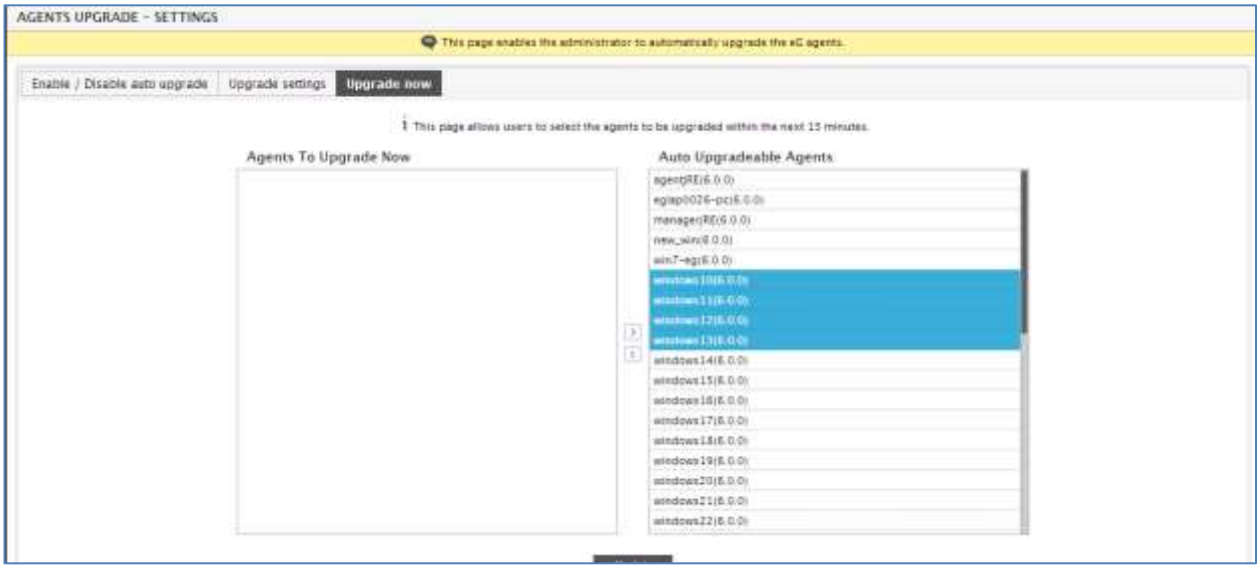

Figure 7: Selecting the agents to be upgraded now

4. The selected agent will then be transferred to the **Agents To Upgrade Now** list (see Figure 8).

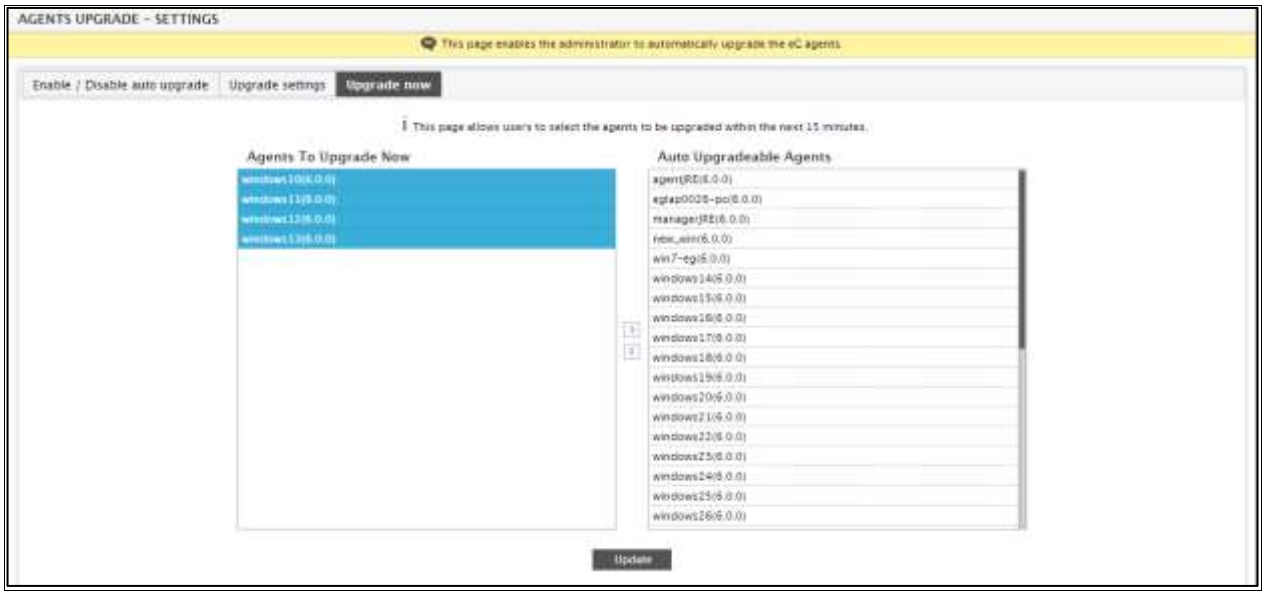

Figure 8: The agents for which Upgrade Now has been enabled

5. To disable the **Upgrade Now** capability later, you can select the agents from the **Agents To Upgrade Now** list and click the > button.

- The **Upgrade now** capability can be enabled for a maximum of 10 agents, simultaneously, to minimize the impact on the eG manager.

Note

- Only the agents that are currently running can receive the upgrade package from the manager.
- After agent upgrade completes, ensure that the **Auto Upgrade** capability is disabled for all the eG agents. This needs to be done to ensure that the eG agents do not periodically check the manager for upgrade patches, thereby consuming considerable resources.

1.6.1 How to Check if the eG Agent Upgrade is Successful?

You can do the following to check if the eG agent has upgraded successfully:

1. Login to the eG admin interface. Follow the Agents -> Upgrade -> Status menu sequence. When Figure 9 appears, select **Version** from the **Filer By** list, select **7.2.4** from the **Filter Criteria** drop-down, and click the **Submit** button. If your eG agents have been successfully upgraded to v7.14, then they will be listed here.

Figure 9: Checking if the eG agents were successfully upgraded to v7.2.4

2. Post the upgrade, you will also be able to reset the agent memory from the eG admin interface itself. For that, follow the Agents -> Status menu sequence, and click the **Agent Xmx Settings** icon corresponding to the agent of interest.

Figure 10: Changing the Xmx configuration of the eG agent from the eG admin interface

1.7 Upgrading the eG Agent's JRE

To ensure proper eG agent operations post the upgrade, **it is mandatory that you upgrade the JRE of the eG agents, after upgrading the eG agents**. The JRE versions you need to upload to are listed in the table below:

Operating System	JRE version
Windows (32-bit)	HotSpot JRE 1.8.0_231
Windows (64-bit)	OpenJRE 17.0.5
Linux (32-bit)	HotSpot JRE 1.8.0_351
Linux (64-bit)	OpenJRE 17.0.5
Solaris (AMD and SPARC)	HotSpot JRE 1.8.0_351
AIX	IBM JRE 1.8.0_341
HP-UX (Itanium)	HotSpot JRE 1.8.0.22-hp-ux

Given below are the steps to be followed to upgrade the JRE used by the eG agent:

1. Typically, the JRE upgrade packages for all operating systems will be available in a **JREUpgrade** folder in the URL **http://www.eginnovations.com/Upgrade724/Agent/**.
2. The table below lists the upgrade packages and the operating systems they apply to:

Operating System	JRE upgrade pack
Windows 2008 / Windows 7 – 32-bit	ega_win2008_725_1I.zip
Windows 2008 / Windows 7 – 64-bit	ega_win2008_x64_725_1I.zip
Windows 2012 / Windows 8 – 32-bit	ega_win2012_725_1I.zip
Windows 2012 / Windows 8 – 64-bit	ega_win2012_x64_725_1I.zip
Windows 2016 / Windows 10 – 32-bit	ega_win2016_725_1I.zip
Windows 2016 / Windows 10 – 64-bit	ega_win2016_x64_725_1I.zip
Windows 2019 – 32-bit	ega_win2019_725_1I.zip
Windows 2019 – 64-bit	ega_win2019_x64_725_1I.zip
Windows 2022 – 32-bit	ega_win2022_725_1I.zip
Windows 2022 – 64-bit	ega_win2022_x64_725_1I.zip
Linux (32-bit)	ega_lin_725_1I.zip
Linux (64-bit)	ega_lin_x64_725_1I.zip
Solaris (AMD)	ega_sol_i86_725_1I.zip
Solaris (SPARC)	ega_sol_725_1I.zip
AIX	ega_aix_725_1I.zip
HP-UX (PA-RISC)	ega_hpx_725_1I.zip
HP-UX (Itanium)	ega_hpx_ita_725_1I.zip

3. To upgrade the JRE used by the eG agent on a particular operating system, first download the 64-bit and 32-bit upgrade packs corresponding to that operating system from the aforesaid URL to the relevant eG agent host. Then, copy the packs to the **<EG_AGENT_INSTALL_DIR>\manager\config\tests\<OperatingSystem>** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**).

Even if you are upgrading the JRE of only a 32-bit agent and not a 64-bit one, you will have to download both the 32-bit and 64-bit upgrade packs corresponding to the operating system and copy the downloaded packs to the **<EG_AGENT_INSTALL_DIR>\manager\config\tests\<OperatingSystem>** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**).

The same applies when you are upgrading the JRE of a 64-bit agent.

4. Next, proceed to enable auto-upgrade for those agents for which JRE is to be upgraded by following the steps discussed hereunder:
   - Login to the eG administrative interface.
   - Select the **Settings** option of the **Upgrade** menu in the **Agents** tile.
   - From the **Auto Upgrade Disabled** list in the **Enable/Disable auto upgrade** tab page, select the agent(s) for which the auto upgrade capability is to be enabled.
   - Then, click the < button to transfer the selection to the **Auto Upgrade Enabled** list.
5. The agents in the **Auto Upgrade Enabled** list will then run the JRE upgrade package that was previously copied to the **<EG_AGENT_INSTALL_DIR>\manager\config\tests\<OperatingSystem>** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**), and automatically upgrade their JRE.

To know whether the JRE was upgraded successfully or not, use the procedure discussed in Section 1.14 of this document.

1.7.1.1 Troubleshooting the JRE Upgrade

To troubleshoot errors that occur during a JRE upgrade on Windows, check the following files for errors:

- The **autoupgrade.log** file in the **<EG_INSTALL_DIR>\agent\logs** directory (on Windows; on Unix, this will be **opt/egurkha/agent/logs**);
- The **JREUpgradeevent** file in the **<EG_INSTALL_DIR>\agent\logs** directory on Windows; on Unix, this will be **opt/egurkha/agent/logs**)

1.8 Upgrade Caveats

- New tests which are enabled by default in the new version will be disabled when the manager is upgraded. Such tests can be enabled after the eG agent is upgraded, using the eG administrative interface.
- If WebLogic or WebSphere-related tests that were running prior to the upgrade have stopped, then redeploy the **egurkha.war** and the **egurkha.ear** files, respectively.
- Typically, the eG install user should run the upgrade script. If in some cases the manager itself is running as root on default ports, the upgrade script can be executed as root user. However, ensure that the manager is

started as root user after the upgrade.

- If the eG agent monitoring the AWS cloud is upgraded to v7.2.4, you will find that the **AWS Virtual Private Cloud** test stops reporting metrics post the upgrade:

  To make sure that this test runs and report metrics, do the following after upgrading the agent:

6. Login to the eG agent host.
7. Connect to the URL: http://www.eginnovations.com/Upgrade724/AWSJar
8. Download the **aws-java-sdk.jar** file in the aforesaid location to any folder on the eG agent host.
9. Copy the jar file to the **<EG_AGENT_INSTALL_DIR>\lib** directory on the host.
10. Restart the eG agent.

- In version 7.1.8, if an eG agent had auto-discovered a Citrix Delivery Controller on the same host as the Citrix Cloud Connector it is installed on, and that delivery controller had then been managed, then upgrading the eG manager to v7.2.4 will automatically unmanage the delivery controller.
- If you had RUM-enabled the *eG Manager* by managing it as a *Real User Monitor* component prior to the upgrade, then after the eG manager is upgraded to v7.2.4, no metrics will be reported for that *Real User Monitor* component.
- Starting from v7.2.4, eG Enterprise will no longer provide monitoring support for *Microsoft Skype for Business Online* components. If you had managed any component of this type prior to the upgrade, then note that that component will stop reporting metrics after the upgrade.
- The *eG Manager License Usage* test mapped to *eG Manager* component will stop reporting metrics after the eG manager is upgraded to v7.2.4. This is because, significant licensing changes have been made in v7.2.4. However, note that once the eG agent (monitoring the *eG Manager* component) is upgraded to v7.2.4, this test will start reporting metrics.
- Before upgrading the eG manager to v7.2.4, if you had created a custom user role (using the eG admin interface), and granted only specific monitoring privileges to that role, then after the manager upgrade, these privileges will be automatically overridden. Instead, all monitoring privileges (even the ones that were not originally granted) will be automatically granted to that role.
- Any change made to the *Virtual Apps* dashboard in the secondary manager prior to the upgrade, will not be reflected in both the primary and secondary managers after the upgrade.
- If you log into a SaaS manager after upgrading it to v7.2.4, you will find that Figure 11 appears informing you

that no eG agent executables exist on the eG manager.

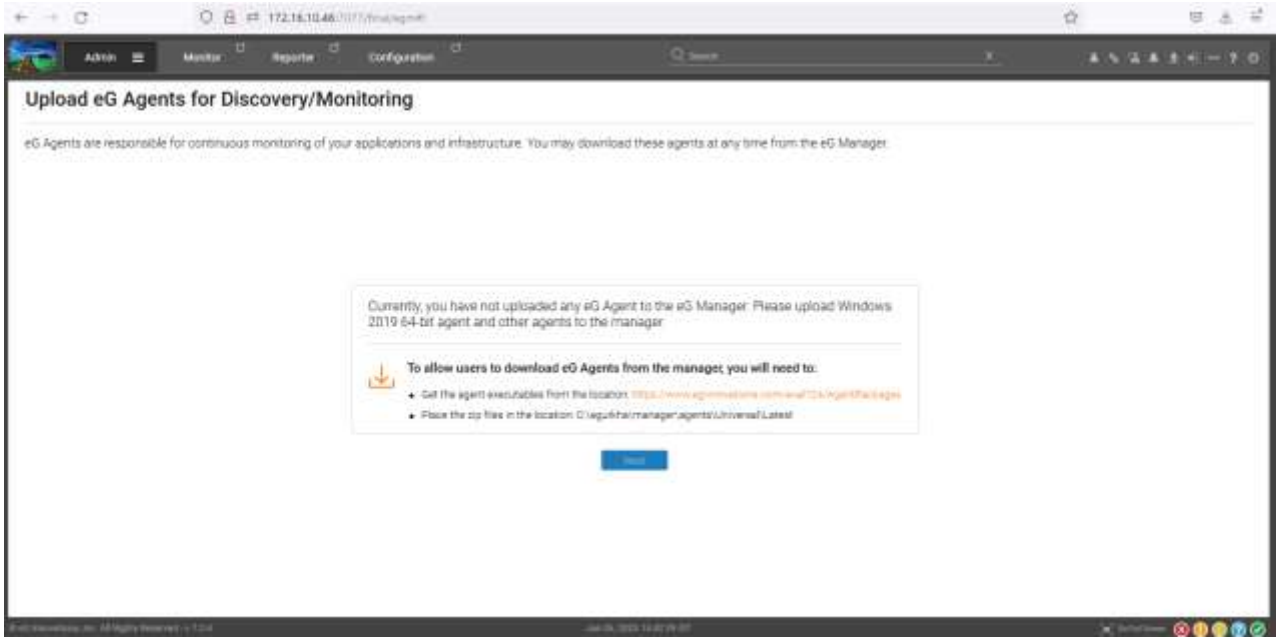

Figure 11: The page that appears once you log into a SaaS manager after upgrading it to v7.2.4

This is because, the upgrade process will automatically remove all agent executables that you may have uploaded to the eG manager prior to the upgrade. To continue using the SaaS manager, you have to do as instructed by Figure 11. In other words, download the agent executables, and then upload them to the <EG_MANAGER_INSTALL_DIR>\manager\agents\Universal\Latest directory (on Windows; on Unix, this will be /opt/egurkha/manager/agents/Universal/Latest) on the eG manager.

1.9 Upgrading the eG Real User Monitor (RUM)

The default RUM collector – i.e., the collector that is bundled with the eG manager – will get automatically upgraded to v7.2.4, once the eG manager is upgraded.

However, if your RUM collector resides on a host other than the eG manager host, then upgrading the eG manager and agent to v7.2.4, does not automatically upgrade the collector. This is why, in such cases, even after upgrading the eG manager and agent, you will find that the eG Real User Monitor continues to report only those metrics and support only those features that were available in the older version. The new RUM metrics that have been added in v7.2.4, and the RUM-related features/enhancements introduced in v7.2.4 will not be available post the manager and agent upgrade.

Please refer to the *Detailed Release Notes for eG Enterprise v7.2.4* document for more information on the new RUM capabilities introduced in v7.2.4.

To avail the benefits of the new RUM capabilities, you will have to upgrade the (non-default) eG RUM Collector.

1.9.1 Upgrading the eG RUM Collector on Windows

For upgrading the collector on Windows, do the following after the eG manager and agent are upgraded:

1. Login to the system hosting the RUM collector.
2. From the browser, connect to the URL: http:/www.eginnovations.com/Upgrade724/RumCollector
3. You will find the following files in this location:
   - The batch file **eGRumUpgrade.bat**
   - The jar file **Extract.jar**
   - The **eG_RumCollectorUpgrade.zip** file
   - The **eg_upgrade.jar** file
4. Download the contents of this URL to any folder on the RUM collector host.
5. Then, go to command prompt, and switch to the folder to which the above-mentioned files are downloaded. Next, proceed to execute the upgrade batch file **eGRumUpgrade.bat** using the following command from the command prompt:

   eGRumUpgrade.bat

6. The program will request you to choose from the following options:

```
WELCOME TO 7.2.4 RUM COLLECTOR UPGRADE

====================================

Enter Your Option :

U - Upgrade  R - Revert  C - Commit E - Exit] ?
```

   U - to upgrade the eG RUM collector

   R - to revert to the previous version of the collector

   C - to commit the upgrade changes

   E – to exit the upgrade menu.

   To trigger the upgrade process, enter **U**.

7. Upon execution, the following messages will appear:

```
WELCOME TO 7.2.4 Rum Collector upgrade
====================================

*****************************************************
 Upgrade process might take several minutes to complete.
 PLEASE DO NOT INTERRUPT THIS PROCESS.
*****************************************************

Extracting the files required for Upgrade...

Starting upgrade of the eG RUM Collector ...
Stopping the eG Rum Collector...
*******************************
The eGRumMon service stopped..
*******************************
*********************************
The egRUM service stopped...
```

```
**********************************
**********************************************
The eG RUM has been successfully stopped.
**********************************************
**************************************
Backup of the eG RumCollector started ...
**************************************

Backup of the eG Rum Collector completed
**************************************
Upgrading the eG Rum Collector configuration ...
**************************************
```

8. When the RUM collector upgrade completes successfully, you will view the following message:

```
*****************************************************************
The eG Rum Collector upgrade has been completed successfully!!!
Execute the command C:\egurkha\bin\start_collector to start the eG Rum Collector.
*****************************************************************
```

9. If for some reason you want to revert to the previous version of the collector after the upgrade, then execute the command **eGRumUpgrade.bat** once again, but this time select the **R** option.

10. Upon a successful revert, the following messages will appear:

```
WELCOME TO REVERT OPERATION
===========================

Reverting the eG Rum Collector to its original configuration.

**************************************************
Revert process might take several minutes to complete.
PLEASE DO NOT INTERRUPT THIS PROCESS.
**************************************************

Stopping the eG Rum Collector ...
**********************************
The eGRumMon service stopped..
**********************************

**************************************
The egRUM service stopped...
**************************************
**********************************************
The eG Rum Collector has been successfully stopped.
**********************************************
        1 dir(s) moved.

*************************************************************
Successfully reverted the eG Rum Collector to the previous version!
Please execute the command C:\egurkha\bin\start_collector
*************************************************************
```

11. On the other hand, if you want to proceed with the upgrade, commit the changes made to the RUM collector by executing the **eGRumUpgrade.bat** command yet again, and selecting the option **C**.

12. Upon successful completion of the commit, the following message will appear:

```
WELCOME TO COMMIT OPERATION
===========================


Committing the RUM Collector Upgrade Changes ....
```

```
Stopping the eG Rum Collector...
*******************************
The eGRumMon service stopped..
*******************************
*********************************
The egRUM service stopped...
*********************************
*********************************************
The eG RUM has been successfully stopped.
Committing the eG RUM Collector...

*************************************
 Successfully Committed the eG RUM Collector Upgrade.
*************************************
```

It is not possible to revert the eG RUM collector, once the upgrade is committed.

1.9.2 Upgrading the eG RUM Collector on Unix

For upgrading the collector on Unix, undeploy the **rumcollector.war** file and redeploy the latest war file on the Apache Tomcat server or the Oracle WebLogic server (on Unix) hosting the collector. The steps to achieve this on an Apache Tomcat server are as follows:

1. Login to the system hosting the eG RUM collector.
2. Stop the eG RUM collector.
3. Then, take a backup of the <CATALINA_HOME>/webapps/rumcollector folder and the <CATALINA_HOME>/webapps/rumcollector.war file.
4. Next, delete the **rumcollector** folder and **rumcollector.war** file from the above-mentioned location.
5. Then, download the latest war file from the URL: https://www.eginnovations.com/eval724/Additional Products/RUM Collector Application WAR/, to any location on the RUM collector host.
6. Redeploy the war file.
7. If you have made any manual changes to the <EG_RUM_COLLECTOR_INSTALL_DIR>/rumcollector/WEB-INF/lib/rum.properties file, then make sure you copy those changes from your backup and paste it in the newly created rum.properties file after redeployment of the war file.
8. Restart the Apache Tomcat server.

On an Oracle WebLogic server, you need to login to the WebLogic administrative console, undeploy the rumcollector.war file, and redeploy it. Before redeploying, make sure you download the war file from https://www.eginnovations.com/eval724/Additional Products/RUM Collector Application WAR/ to any location on the RUM collector host.

1.9.3 Upgrading the eG RUM Collector on Azure App Services

To upgrade the eG RUM collector deployed on Azure App Services, do the following:

1. Login to the Azure portal.

2. Click on the **App Services** option as indicated by Figure 12.

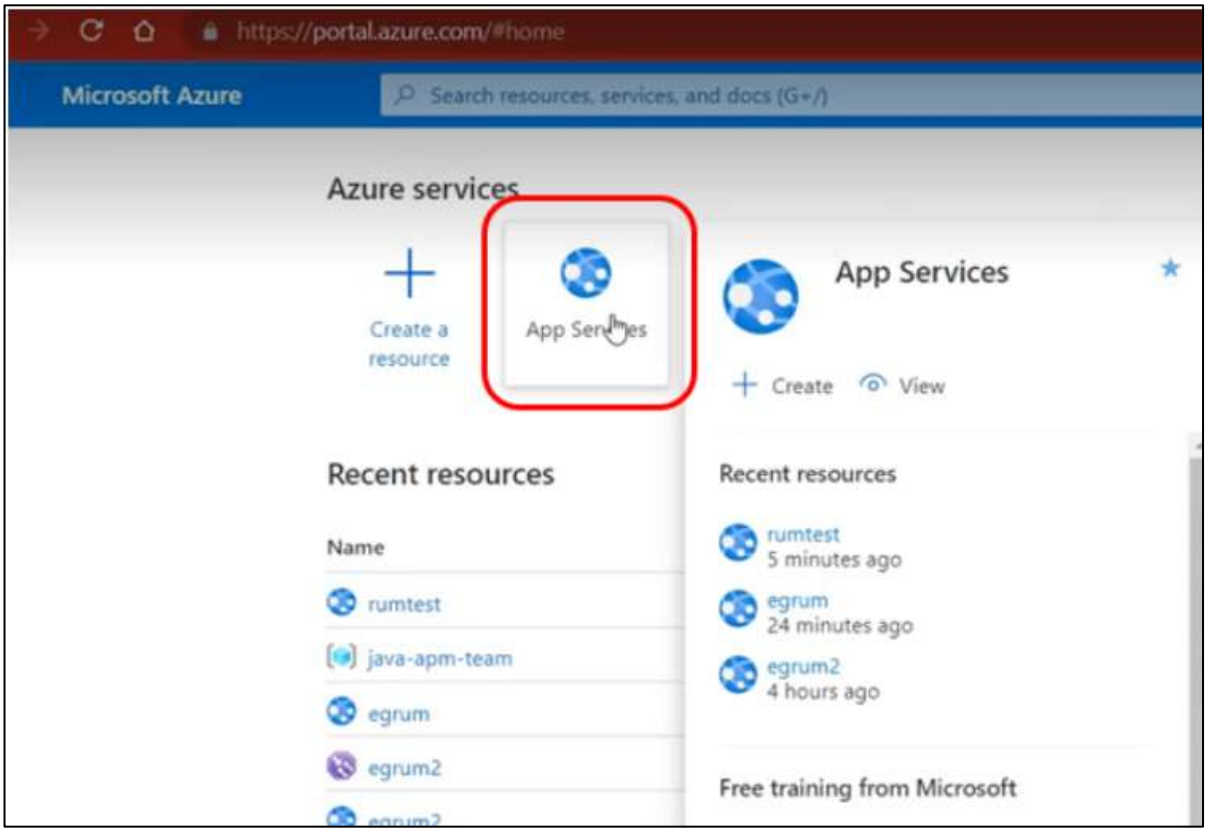

Figure 12: Selecting the App Services option from the Azure portal

3. An **App Services** panel will open listing the applications that are deployed. Pick the eG RUM Collector application from the list by clicking on it. Figure 13 will then appear displaying the properties of the chosen application.

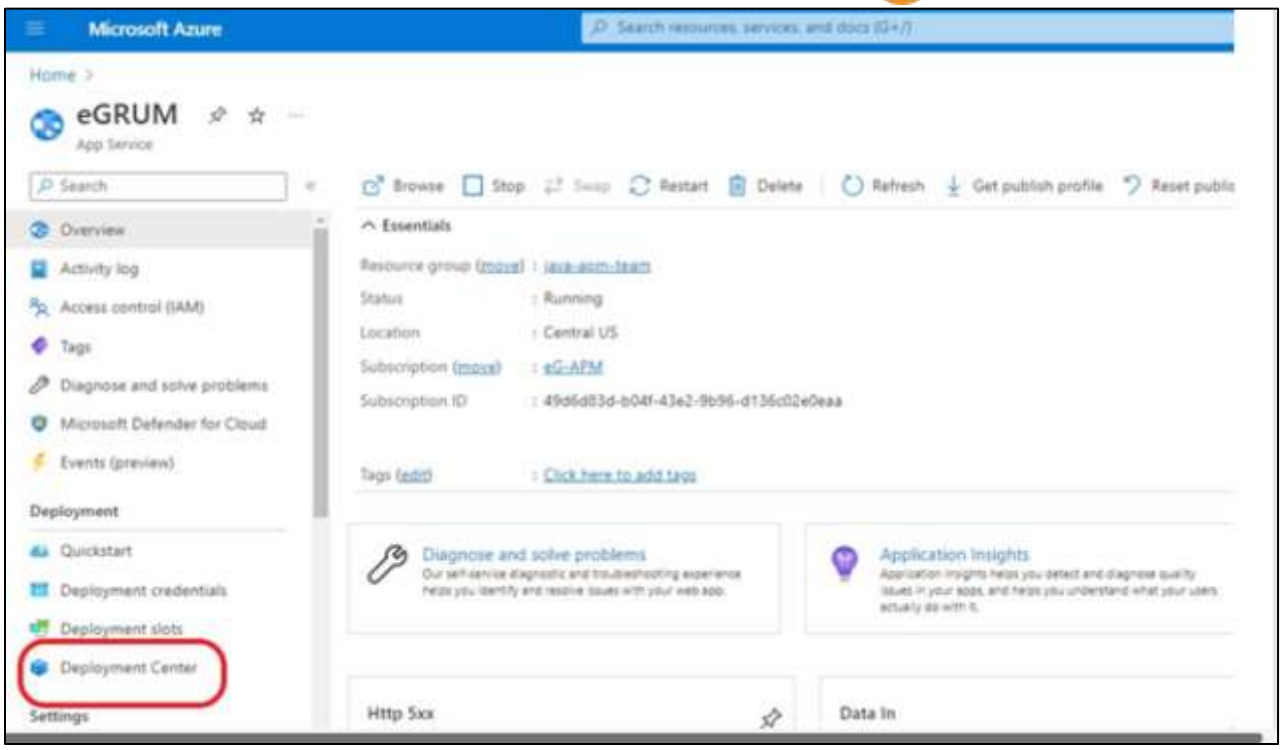

Figure 13: Clicking on the Deployment Center option

4. Then, click on the **Deployment Center** option in the left panel of Figure 13. Figure 14 will then appear.

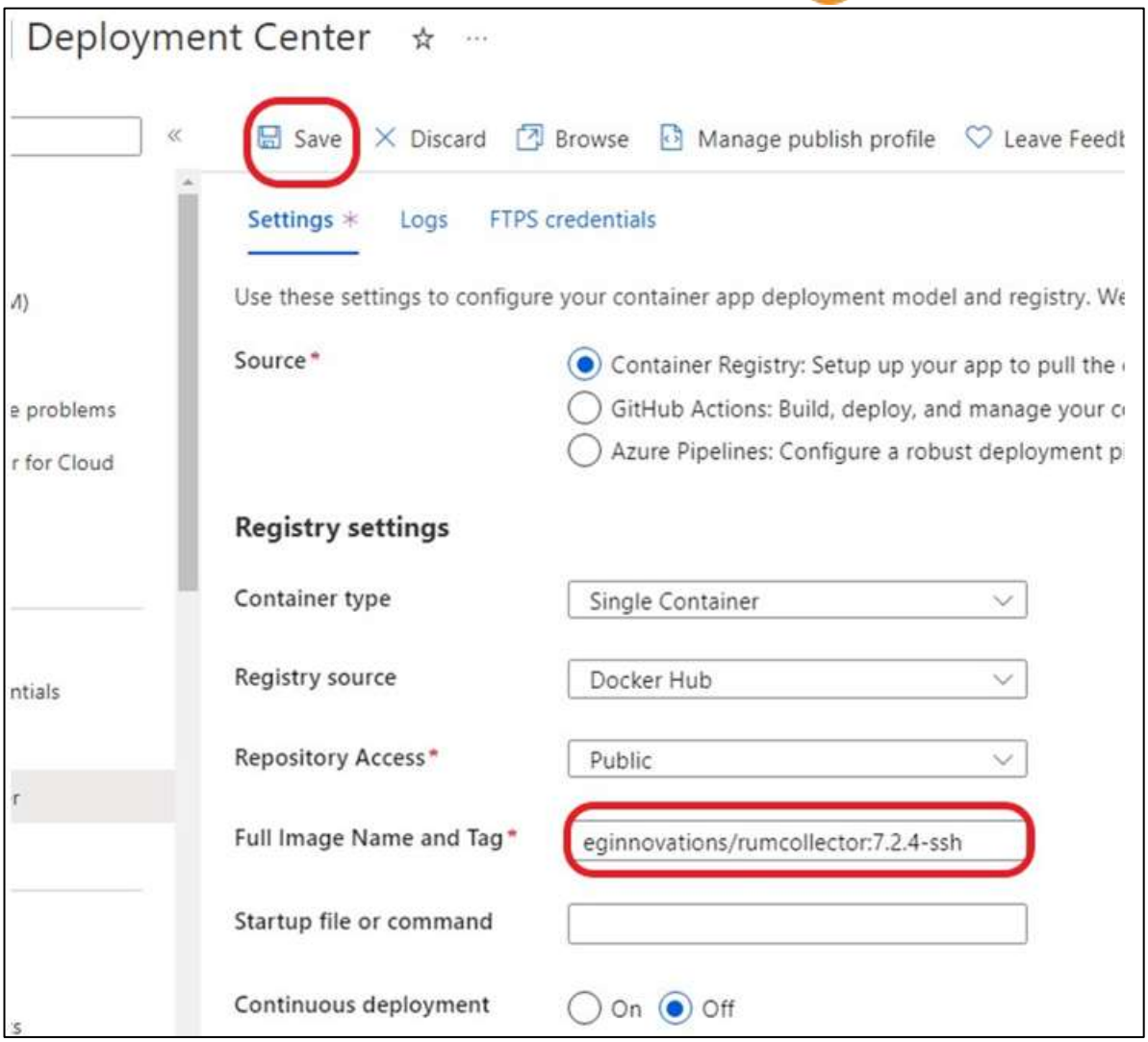

Figure 14: Specifying the path to the eG RUM Collector container image for v7.2.4

5. Against the **Full Image Name and Tag** field in Figure 14, specify the full path to the eG RUM collector container image for v7.2.4. This will be: **eginnovations/rumcollector:7.2.4-ssh**
6. Then, click the **Save** button in Figure 14. Doing so will automatically deploy the updated container image in the Azure App Services. No restarts are required.
7. Once the container configuration is successfully updated in Azure App Services, a message to that effect will appear (as indicated by Figure 15)

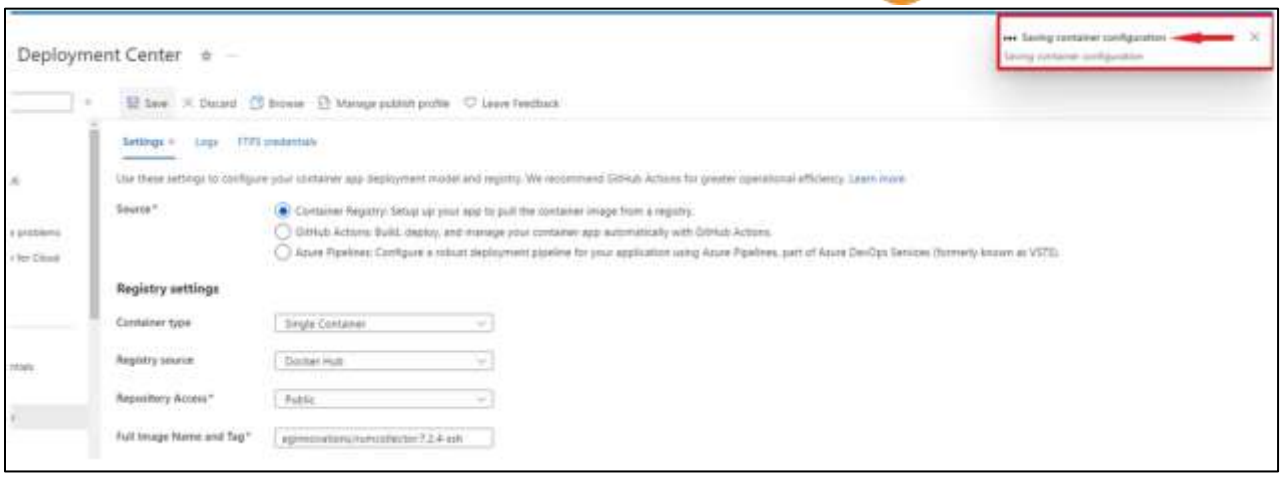

Figure 15: Message indicating that the container configuration has been successfully updated

8. After you see the above message, wait for a couple of minutes for the changes to be reflected on the RUM collector side. Then, proceed to verify whether/not the RUM Collector has been successfully upgraded. For this, key in the URL of the eG RUM Collector application in the address bar of the browser, in the following format: *https://<RUMCollectorAppName>.azurewebsites.net/rumcollector/Welcome.jsp*

9. If the RUM collector has been successfully upgraded, then the above URL should open a Welcome page, with the browser title displaying the latest RUM collector version – i.e., v7.2.4 (see Figure 16).

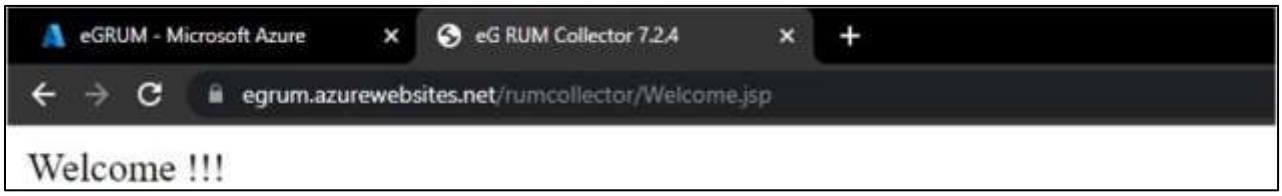

Figure 16: Browser title displaying RUM collector version as 7.2.4

- The RUM collector upgrade process stops and starts the RUM collector. Typically, RUM-enabled web pages load the **egrum.js** script from the RUM collector. As a result, the load time of these web pages may be impacted when a collector upgrade is in progress. The impact however depends upon whether the web pages load the egrum.js script file synchronously or asynchronously. In eG Enterprise v7.1 (and its variants), the RUM code snippet has been changed to ensure that the **egrum.js** script is loaded asynchronously. This has been done so that the non-availability of the **egrum.js** script does not in any way impact web page load time. Prior to v7.1 however, the **egrum.js** script was loaded only synchronously. This means, if the RUM collector being upgraded is of a version before v7.1, then the target web pages would have been instrumented to load the **egrum.js** script synchronously. In this case, during the RUM collector upgrade process, the load time of these web pages may increase. However, if the RUM collector being upgraded is of version 7.1 (or after), then the target web pages would have been configured to load the **egrum.js** script asynchronously. In this case, while the RUM collector is being upgraded, the load time of these web pages will not be impacted.

- Typically, the **egrum.js** cohabits with the eG RUM collector. Sometimes however, to ensure the optimal performance and faster loading of the monitored web pages, you may have copied the **egrum.js** script file to the server hosting the target web site/application or to any 'highly available' location. In this case, after upgrading the eG RUM Collector, make sure you copy the latest **egrum.js** script from the system hosting the upgraded collector to any other location(s) you may have copied it to.

- As already mentioned, in eG Enterprise v7.1 (and its variants), the eG RUM code snippet has been changed to ensure that the **egrum.js** script is loaded asynchronously. This has been done so that the non-availability of the **egrum.js** script does not in any way impact web page load time. Prior to v7.1 however, the **egrum.js** script was loaded only synchronously. This is why, if the RUM collector being upgraded is of a version before v7.1, we recommend that you do the following, after the RUM collector is upgraded to v7.2.4:

    - Login to the eG admin interface.
    - Follow the Infrastructure -> Components -> Add/Modify component menu.
    - Select *Real User Monitor* as the **Component type**, and click the **Edit** button alongside the RUM component of interest to you.
    - When Figure 17 appears, scroll down the image until you view the eG RUM code snippet. Scroll down the snippet until you view the line of code that loads the **egrum.js** script. If that line ends with the string 'async' (as indicated by Figure 17), it implies that the egrum.js script is loaded asynchronously.

Figure 17: Viewing the eG RUM code snippet

- Copy the complete code snippet from Figure 17 and re-insert/re-inject it into each of the web pages you want monitored.

- In v7.2.4, when adding/modifying a Real User Monitor component using the eG admin interface, you can optionally turn on certain new capabilities for the eG Real User Monitor – eg., capturing user name, enabling AJAX correlation etc. (see Figure 18).

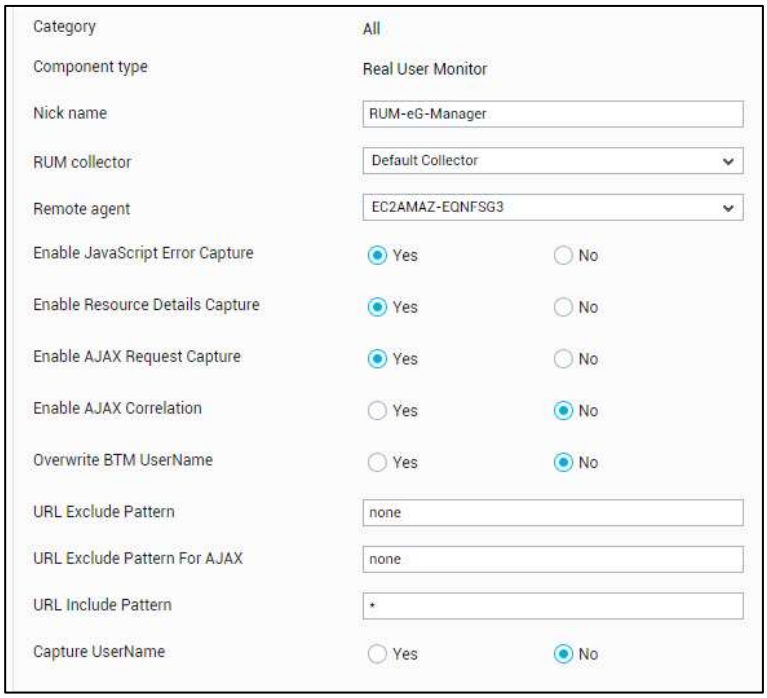

Figure 18: Adding a Real User Monitor

After upgrading the eG manager, agent, and eG RUM collector that is monitoring a Real User Monitor component to v7.2.4, if you modify the configuration of that component by enabling one/more of the new RUM capabilities (as depicted by Figure 18), then the eG RUM code snippet will change accordingly. To ensure that this change is replicated across all pages where this code

snippet is injected, you need to **copy the new code snippet from the eG admin interface and re-inject/re-insert it into each of the monitored web pages.**

- In v7.2.4, the RUM code snippet for Single Page Applications (eg., Generic, Angular etc.) has changed. So, if you are upgrading from v7.1.8, and you had instrumented a Single Page Application (e.g., Angular JS) for Real User Monitoring in v7.1.8, then after upgrading eG Enterprise to v7.2.4, **you will have to copy the new code snippet from the eG admin interface and re-inject/re-insert it into each monitored base, IFrame, AJAX page of the target Single Page Application.** To achieve this, follow the steps below:
  - Login to the eG admin interface.
  - Follow the Infrastructure -> Components -> Add/Modify menu sequence, select *Real User Monitor* from the **Component type** drop-down, **and** click the **Modify** icon corresponding to the Single Page Application (SPA) that is being managed as a *Real User Monitor* component.
  - Copy the new RUM code snippet from the page that appears next.
  - Paste the copied snippet in the *<head>* tag of the **index.html** page of the monitored SPA. If the SPA is served via web server / load balancer, then use the container-based injection technique to re-instrument the SPA. This way, you can once again RUM-enable the base pages, iFrames, and AJAX pages of the target Single Page Application.

  If you had RUM-enabled the virtual pages of the SPA too in v7.1.8, then you would have implemented code-level changes in v7.1.8 to achieve the same. In v7.2.4 however, these code-level changes are bundled into the target SPA. Therefore, after upgrading to eG Enterprise v7.2.4, you will have to remove the changes that you earlier made as they are now redundant. So, once eG is upgraded to v7.2.4, remove the code block you inserted into the **app.component.ts** file (in the <APP_HOME>\src\app directory) of the target SPA to capture virtual page navigations. Similarly, remove the code block you inserted in the **app.modules.ts** file (in the <APP_HOME>\src\app directory) of the target SPA to capture error events.

  Also, note that the RUM code snippet of non-SPA applications has not changed from v7.1.8 to v7.2.4. Therefore, if a non-SPA application was managed as a Real User Monitor component in v7.1.8, then after upgrading to v7.2.4, **you do not have to inject/re-insert code snippet into each of the monitored web pages of the non-SPA applications.**

.

1.10 Upgrading the eG Java BTM

Upgrading the eG manager and agent to v7.2.4, does not automatically upgrade the eG Java Business Transaction Monitor (BTM) as well. This is why, even after upgrading the eG manager and agent, you will find that the eG Java BTM continues to report only those BTM metrics and support only those features that weailable in the older version. The new business transaction metrics that have been added in v7.2.4, and the BTM-related features/enhancements introduced in v7.2.4

will not be available post the manager and agent upgrade.

Please refer to the *Detailed Release Notes for eG Enterprise v7.2.4* document for more information on the new BTM capabilities introduced in v7.2.4.

To avail the benefits of the new monitoring capabilities, you will have to upgrade the eG Java BTM. For upgrading the Java BTM, do the following after the eG manager and agent are upgraded:

1. Login to the BTM-enabled server.
2. Stop the server instance that is BTM-enabled. If multiple instances on the server have been instrumented for BTM, then stop all the BTM-enabled instances.
3. By default, the eg_btm.jar file – i.e., the eG Application Server Agent – will reside in the /opt/egurkha/lib/apm/Java/default folder (on Unix; on Windows, this will be the <EG_AGENT_INSTALL_DIR>\lib\apm\Java\default folder) on the agent host.
4. After the upgrade, a file named eg_btm_new.jar will be newly created in this folder.
5. If all the BTM-enabled server instances are using the eg_btm.jar file in the default path for business transaction monitoring, then follow the steps below to upgrade Java BTM for these instances:
6. Rename the eg_btm.jar file in the default path to eg_btm_old.jar.
7. Rename the eg_btm_new.jar file in the default path to eg_btm.jar.
8. Restart all the instances.
9. On the other hand, if when BTM-enabling server instances, the eg_btm.jar file had been copied to instance-specific folders on the application server, then after upgrading the manager and agent to v7.2.4, do the following:
10. Reconfigure the individual server instances to use the eg_btm.jar file in the default path for business transaction monitoring. For that, first, login to the eG admin interface
11. Follow the Agents -> BTM Profiler Download menu sequence.
12. When Figure 19 appears, look for the BTM-enabled JVM instance in the list that appears. If the instance is being monitored in an agent-based manner, then once you find the instance in the list in Figure 19, click the **Copy JVM Option** icon corresponding to it.

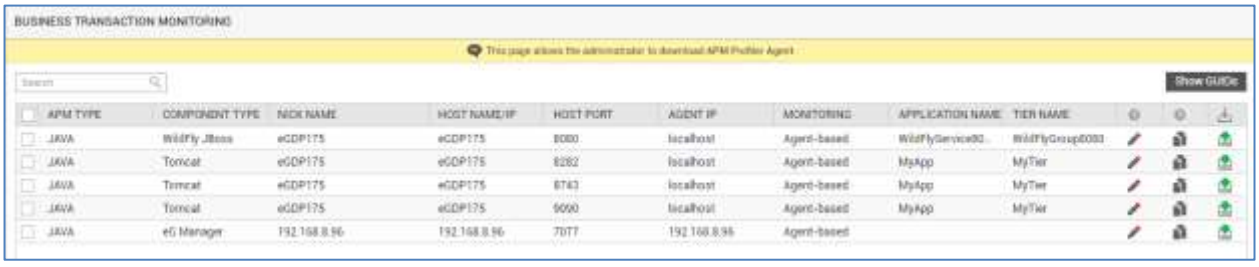

Figure 19: Clicking on the Copy JVM Option icon corresponding to the BTM-enabled component

13. When Figure 19 appears, look for the BTM-enabled JVM instance in the list that appears. If the instance is being

monitored in an agent-based manner, then once you find the instance in the list in Figure 19, click the **Copy JVM Option** icon corresponding to it.

14. Figure 20 then appears displaying the JVM arguments that need to be inserted in the start-up script of the target instance to BTM-enable it. This argument will automatically point to the eg_btm.jar file in the default path. Click on the **Copy** button to copy the displayed JVM arguments. This will copy the arguments to the clip board.

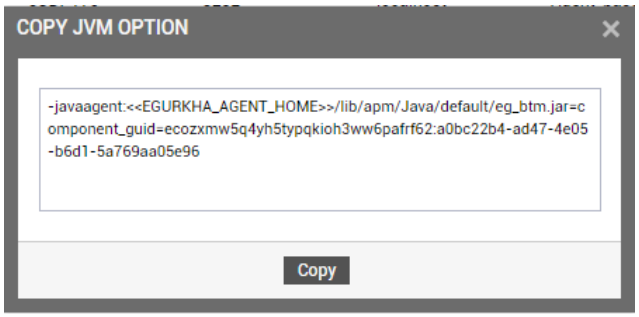

Figure 20: Copying the JVM arguments

15. Then, open the start-up script of the target JVM instance and look for the -javaagent entry therein. Once you find it, select the complete entry and overwrite it by pasting the copied arguments over it. The steps for achieving this differ from one Java/J2EE container to another. Please refer to the Java Business Transaction Monitoring document for the steps relevant to the target JVM instance.
16. Finally, save the changes to the script and restart the instance.
17. Make sure you repeat steps d-g above for every BTM-enabled application/application instance in your environment that is being monitored in an agent-based manner.
18. On the other hand, if the target JVM instance is being monitored in an agentless manner (i.e., using a remote agent), then, once you find the instance in the list in Figure 19, click the **Download** button corresponding to that instance. This will download the profiler-related files, including the eg_btm.jar file and .props files for v7.2.4, to the local host.
19. Once the files are downloaded to the local host, replace the old profiler files on the system hosting the JVM instance with the downloaded files.
20. Finally, restart the target JVM instance.
21. Make sure you repeat steps i-k for every BTM-enabled application/application instance in your environment that is monitored in an agentless manner.

1.11 Upgrading the eG .NET Profiler

If .NET application-level tracing is enabled on a system, then note that the upgrade process will not automatically upgrade the .NET profiler. In other words, if only the eG manager and agent are upgraded to v7.2.4 and not the .NET profiler, then .NET transactions will not be monitored until the profiler is also upgraded. To upgrade the .NET profiler, you have to run the **<<EG_INSTALL_DIR>>\lib\update_profiler.bat** command on the Profiler host, after you stop the target web server for regular maintenance.

Note:

After you upgrade the .NET Profiler to v7.2.4, you may notice sudden and significant spikes in resource consumption by the profiler. This is because in v7.2.4, the .NET Profiler automatically captures asynchronous calls made by the target .NET application, and reports the details of all such calls made as part of detailed diagnostics. This exercise may leave a heavy resource footprint on the profiler, and may also adversely impact the performance of the target

.NET application. To avoid this, after the upgrade, you can choose to disable asynchronous call monitoring. For this, do the following:

- Login to the IIS web server hosting the .NET Profiler.
- Edit the eg_DotnetServer.ini file in the <EG_AGENT_INSTALL_DIR>\agent\config folder.
- Search for the **ENABLE_THREADS_TASKS** parameter in the file. Once you find it, you will see that it is set to 1 by default. This means that asynchronous call monitoring is enabled by default for the profiler.
- To disable this capability, set the **ENABLE_THREADS_TASKS** parameter to 0.
- Then, save the file.
- Restart the IIS web server.

1.12 Upgrading the eG PHP Profiler

Where the eG PHP Profiler is installed, the upgrade process will not automatically upgrade the profiler. In other words, if only the eG manager and agent are upgraded to v7.2.4 and not the PHP profiler, then PHP transactions will not be monitored until the profiler is also upgraded. To upgrade the .NET profiler, after the eG manager and agent are upgraded, you will need to deploy the profiler on the target PHP application once again. For this, follow the steps detailed in the *How to BTM-enable a PHP Web Application?* topic in the *PHP Business Transaction Monitoring* document.

1.13 Determining the Status of the eG Agents

The eG manager is able to determine and report the operational status of all the eG agents in the target environment. The sections that follow will discuss how to view this status information.

If you select the **Agent Status** option from the **Agents** tile, you will be lead to Figure 21, which will provide status information for agents based on the agent types.

To obtain the status of the eG agents of a particular type, the administrator has to first select the type of agent (whether basic, premium, external, or remote) from the **Agent type** list box.

The IP address / host names of the agents of the selected type will then be displayed. A symbol against each agent indicates that the agent has been deployed. A symbol appears against each agent implying that the agent has not been deployed. While the symbol indicates that the agent is running currently, the **Status** column indicates that the eG agent is not running.

Figure 21: Status information for agents

Also, using the **Search** text box, you can find out the status of a particular agent. To know the status of a particular agent, just specify the IP address / host name of that agent in the **Search** text box, and then click the 'magnifying glass' icon next to it. The status of the specified agent will then appear. If the exact IP address / host name of the agent is not

known, then a string or a character that features in the IP / host name of the agent can be provided in this text box (see Figure 22). Multiple search conditions can be specified as a comma-separated list.

Figure 22: Searching for agent status

To know the agents that are currently in a particular state, simply select an **Agent status** (which can be Running/Not Running/All). The default selection here is *All*.

You can even remotely initiate an agent-restart, by simply clicking on the **Restart** icon that corresponds to an agent. To restart all agents, click on the **Restart All Agents** button in Figure 22. Doing so immediately sends out restart requests to all the agents that are currently running and reporting metrics to the eG manager. If an agent is not running currently, then the eG Enterprise system sends out the restart request soon after that agent starts running.

If an administrator needs to be alerted upon login, about agents that are not reporting measures to the manager, then do the following:

1. Open the **eg_services.ini** file in the **<EG_HOME_DIR>/manager/config** directory.
2. In the **[MISC_ARGS]** section, set the **AlertAgentsNotRunning** flag to **Yes** (default is **No**).
3. Once this is done, the next time the administrator logs into the admin interface, a message listing the agents that are not running will be displayed.

Note

An eG agent can be configured to run specific tests once a day or once every few hours. You can configure the eG manager to exclude tests that are infrequently run when it determines whether an agent is running or not. To do this, modify the value of **NotReportingCutoffFactor** in the **[MISC_ARGS]** section of the **eg_services.ini** file. By default, tests running with measure period of greater than 20 minutes are not considered by the eG manager for determining if an agent is running or not.

Also, by default, output logging is disabled for th, e eG agents configured in an environment. The eG Enterprise system allows you to enable output and error logging for a specific agent from the eG administrative interface itself, thereby saving you the trouble of running the **debugon.bat** file to achieve the same. When output logging is enabled, an **agentout.log** file is created in the **<EG_INSTALL_DIR>\agent\logs** directory to which details of the tests run and measures reported by that agent to the manager are recorded. To enable output logging for an agent, set the **Output Logging Enabled?** flag for that agent to **ON**. When you attempt to enable output logging, a message box shown by Figure 23 will appear, requesting your confirmation to enable output logging for that agent.

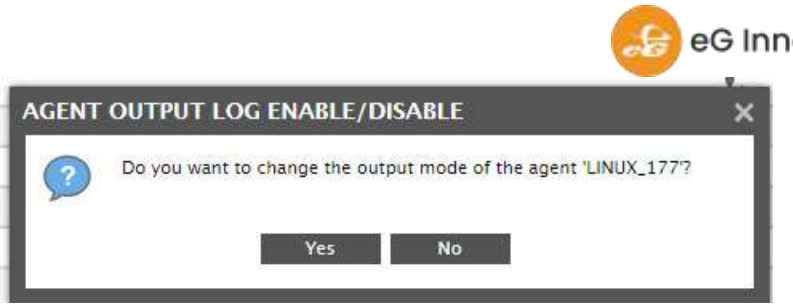

Figure 23: A message box requesting your confirmation to enable output logging

Click on the **OK** button in the message box to enable output logging or the **Cancel** button to disable it. You can then click on the **LOGS** icon that corresponds to an agent in to view both error logs and output logs related to that agent. Clicking on the **LOGS** icon corresponding to that agent will lead you to Figure 24, where the contents of the **error_log** of the corresponding agent can be viewed by default.

Note

If you have turned on output logging for an eG agent using the **AGENTS – STATUS** page, then you should not turn off output logging for that eG agent by manually running the **debugon.bat** file. Likewise, if you have turned on output logging for an eG agent by running the **debugon.bat** file, then you should not turn it off using the **AGENTS – STATUS** page.

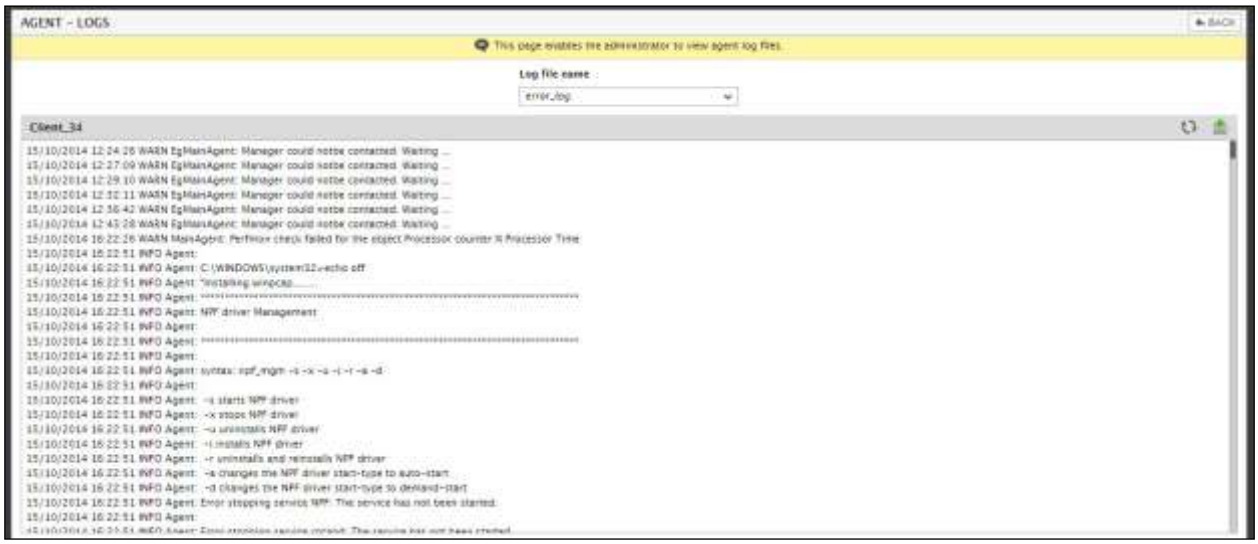

Figure 24: Viewing the error_log of an agent

You can pick any log file from the **Log file name** list to view its contents (see Figure 25).

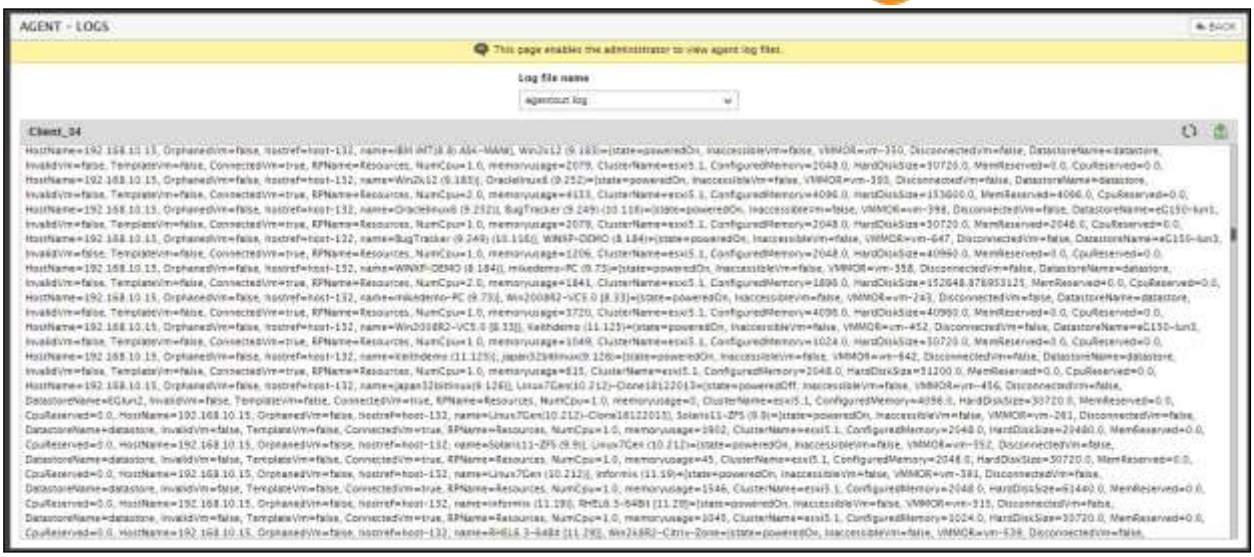

Figure 25: Viewing a different log file

At any given point in time, you can click the **Refresh** button at the right, top corner of the area where the log file contents are displayed to refresh the contents of the log file. This way, you can make sure that the log file you are viewing is up to date.

You can also click on the **Download** button next to the **Refresh** button to download the chosen log file.

Clicking on an agent displayed in Figure 22 will lead the users to an **Agent Information** page (see Figure 26), which provides some agent-related information. This includes:

- The **Agent IP/Nickname**
- An indicator as to whether the auto upgrade capability has been **Enabled** for that agent, or **Disabled**
- The ID of the last upgraded package (if any) (if no upgrading has occurred, then this will be 'None")
- The date and time at which the agent was last upgraded
- The **HostName** of the agent
- The operating system on which the agent is executing
- The current version of the agent
- The date and time at which the agent last updated the manager with configuration changes
- A **Reset** button
- A **Restart Agent** button

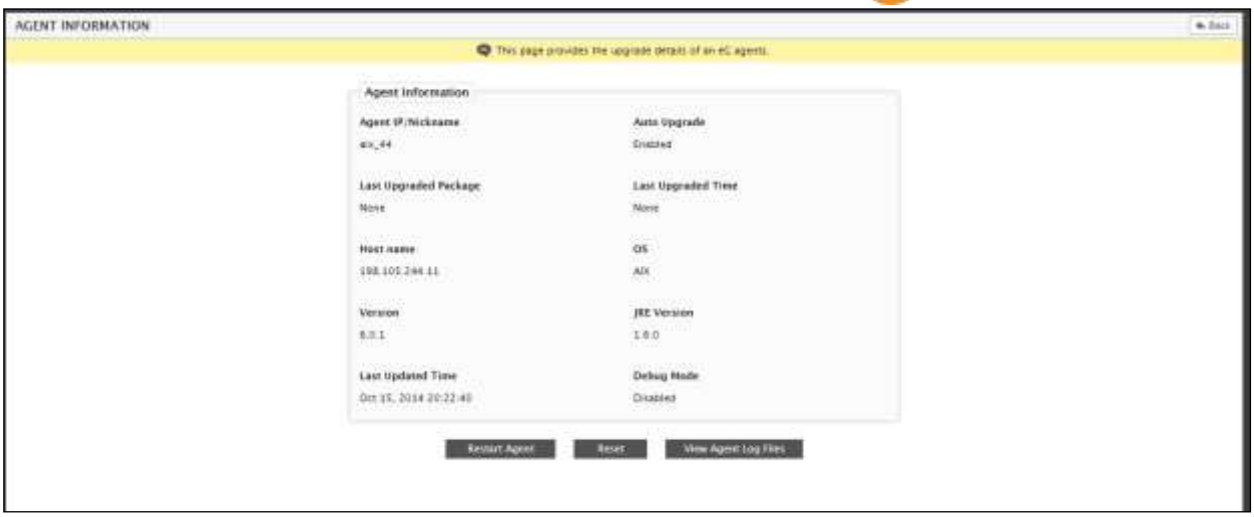

Figure 26: A page displaying the upgrade information of an agent

Once an agent is upgraded, information regarding the upgraded package will be registered with the manager. Figure 26 provides that information. Now, the next time the agent requests for an upgrade, the manager checks whether any newer upgrades are available. If any such upgrade is found, it sends the same to the agent. If for some reason the information pertaining to the last upgrade has to be cleared from the agent's upgrade history, then click on the **Reset** button. This ensures that the details of the last upgrade are lost, and helps the agent download the last upgrade once again from the manager. To restart the agent, click on the **Restart Agent** button in Figure 26. To view agent logs, click on the **View Agent Log Files** button in Figure 26.

Moreover, if the **Agent type** chosen from Figure 27 is **External Agents** or **Remote Agents**, then you will also be able to view the count of hosts assigned (if any) to each external/remote agent. For this, you will have to click on the '+' button that pre-fixes an agent (see Figure 27). Beneath the assigned host count, you can see that Figure 27 also reveals which specific hosts have been assigned to that agent. From a single glance therefore, you can precisely identify the external/remote agents that are been actively utilized, and those that are not.

Figure 27: Viewing the status of external agents

To add more hosts to an external/remote agent, click on the button corresponding to an agent in Figure 27. This will open Figure 28, using which you can assign more hosts to the agent or disassociate existing hosts from it.

Figure 28: Associating/Disassociating hosts from an external agent

1.14 Viewing the Upgrade Status

Clicking on the **Status** option in the **Upgrade** menu of the **Agents** tile will open Figure 29 that reveals the following information indicating the upgrade status of every agent reporting to a manager:

- The IP/hostname of the agent
- The unique package id of the last upgraded package of the agent
- The time of upgrade
- Whether upgrade is currently disabled or enabled for the agent
- The operating system on which the agent executes
- The current version of the agent
- The version of JRE used by the agent

Figure 29: Viewing the upgrade status of all agents

To view the upgrade status selectively, choose a **Filter By** option. By default, **None** (see Figure 29) will be selected in this list box. Besides this, the list box offers the following filtering options:

- To view the upgrade information pertaining to agents of a particular version (see Figure 30), select the **Version** option from the Filter By list box. From the **Filter Criteria** list box that appears next, select a particular version

number, and finally, click the **Submit** button.

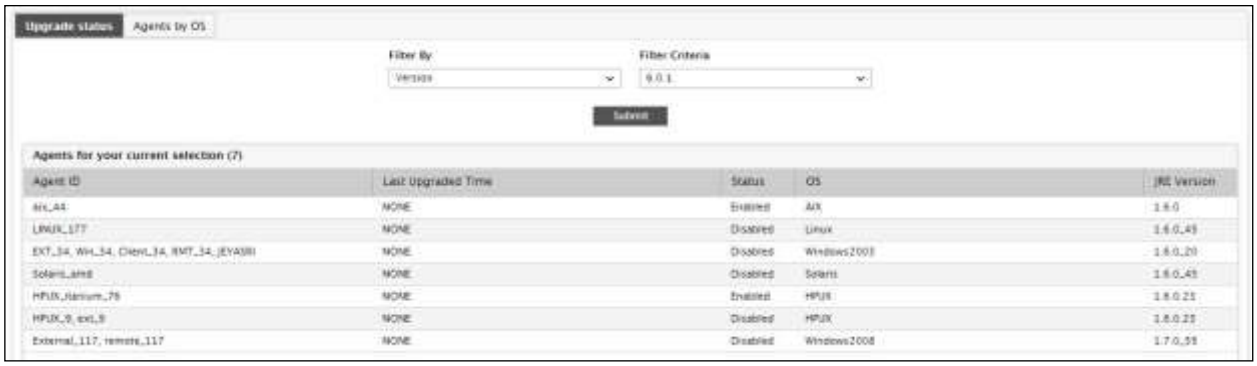

Figure 30: Viewing the upgrade status of agents of a specific version

- To view the upgrade information of agents executing on a specific operating system (see Figure 31), select the **Operating system** option from the Filter By list box. From the **Filter Criteria** list box that appears next, select a particular operating system, and finally, click the **Submit** button.

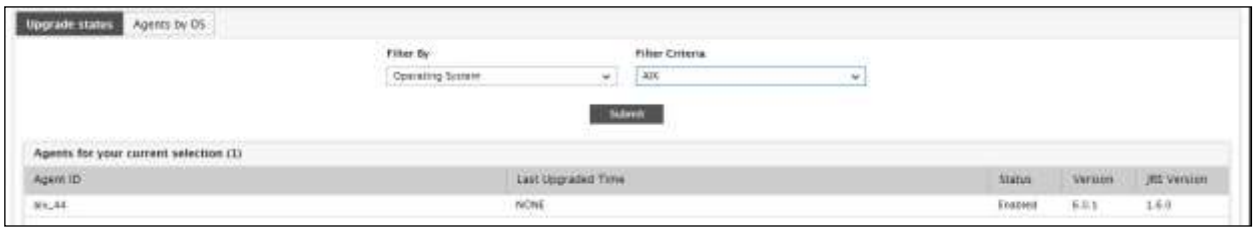

Figure 31: Viewing the upgrade status of agents executing on a particular operating system

- To view the agent upgrade status based on the upgrade setting (i.e. whether enabled/disabled) (see Figure 32), select the **Upgrade setting** option from the **Filter By** list box. From the **Filter Criteria** list box that appears next, select either **Enabled** or **Disabled**, and finally, click the **Submit** button.

Figure 32: Viewing the upgrade status of agents with a specific upgrade setting

- To view the agent upgrade status based on the JRE version, select the **JRE version** option from the **Filter By** list box (see Figure 33). From the **Filter Criteria** list box that appears next, select the JRE version to search for, and click the **Submit** button.

Figure 33: Viewing the upgrade status of agents with a specific JRE version

If you click on the **Agent by OS** tab page in Figure 33 you will also view a brief summary of the number of agents executing on every OS (see Figure 34).

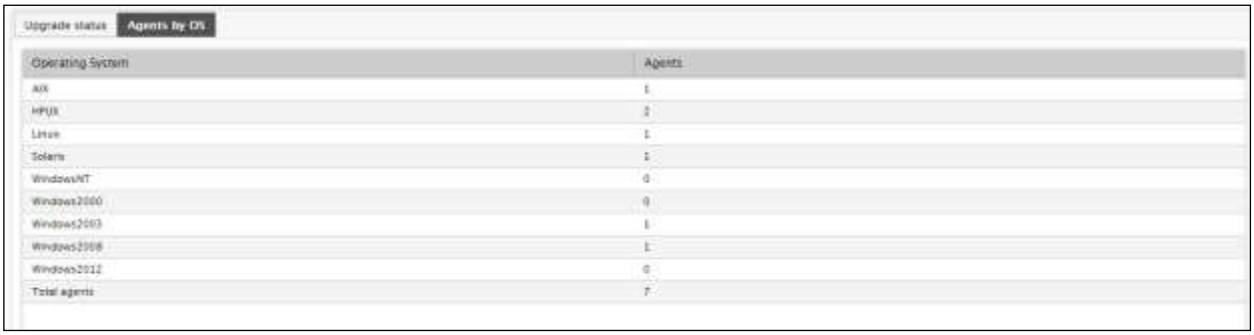

Figure 34: Agent summary by OS

1.15 Advanced Search Options

In environments comprising of a large number of components, it is often very difficult for administrators to remember which agent manages each of the monitored components, and whether upgrade/upgrade now has been enabled for those agents or not. eG Enterprise therefore, provides a single interface using which administrators can search for and view details of agents based on a given component name, component type, and/or IP address(es). In addition, this interface permits the display of agents based on upgrade status (i.e., whether auto-upgrade/Upgrade now has been enabled/not), and allows administrators to instantly enable/disable upgrade for one/more of the listed agents.

To access this user interface, select the **Advanced Search** menu option from the **Upgrade** menu in the **Agents** tile.

Figure 35: The ADVANCED SEARCH page displaying the filter criteria

Figure 35 appears next.

1. First, pick a **Search criteria**.
2. If you want to view the details of agents monitoring specific components, select **Component** from **Search criteria** and specify a search string in the **Component name** text box.
3. Then, click the **Search** button Figure 36. The result set that appears (see Figure 36) displays the details of agents that are monitoring those components which have names that embed the specified string.

Figure 36: Searching based on Component name

4. In the same way, to view the details of all the agents that are monitoring components of a particular type, specify the **Component type**, and click the **Search** button to retrieve the results (see Figure 37).

Figure 37: Searching based on Component type

5. Similarly, you can also view the details of agents of a particular status, by selecting the desired option from the **Status** list. You can thus choose to view information pertaining to agents for which auto upgrade is disabled/enabled, or upgrade now is disabled/enabled (see Figure 38).

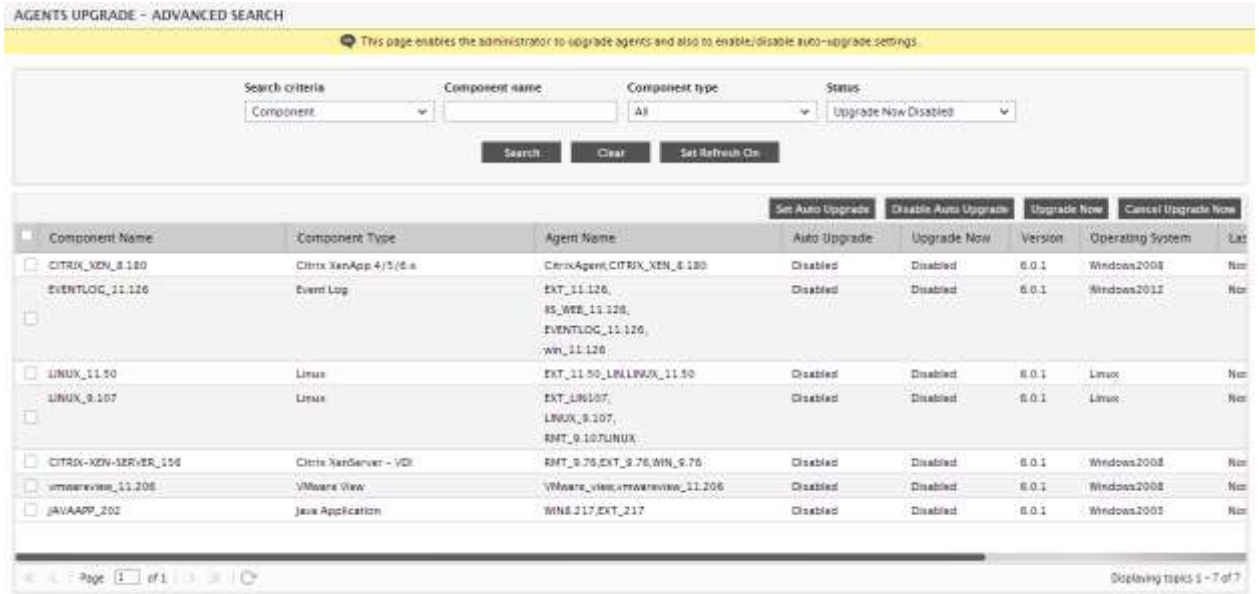

Figure 38: Searching based on Status

6. Alternatively, a combination of search criteria can also be specified as indicated by Figure 39.

Figure 39: Searching based on Component Type and Status

7. Also, instead of filtering your agent-view on the basis of a specific component name, type, or agent status, you can simply provide an IP address or range of IP addresses for which agent information is required. Clicking on the **Search** button then, will display the details of all agents with host/nick names that are associated with the given IP address(es) (see Figure 40 and Figure 41).

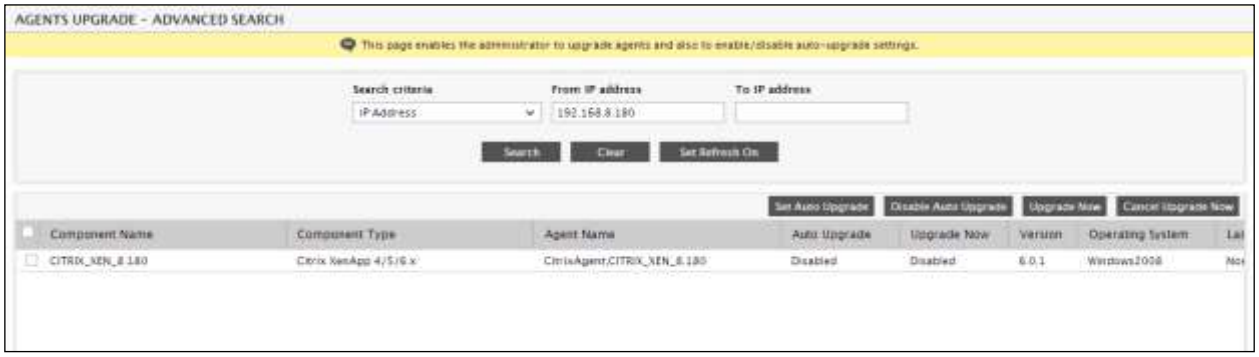

Figure 40: Searching based on a single IP address

Figure 41: Searching based on a range of IP addresses

8. However, regardless of the search criteria specified, the following information is typically retrieved and displayed in the **ADVANCED SEARCH** page:

   - **Component Name**
   - **Component Type**

- **Agent Name** - All the nick names that map to the IP address of the displayed **Component Name**
- Whether **Auto Upgrade** and **Upgrade Now** have been enabled for the agent or not
- The current **Version** of the eG agent
- The **Operating System** on which the eG agent functions
- The ID of the **Last Upgraded Package**
- **Last Upgraded Time** - The time at which the agent was last upgraded

9. By default, the agent information displayed is sorted in the descending order of the contents of the **Component type** column. This is indicated by a down-arrow mark adjacent to the column-heading **Component type** (see Figure 42). To sort the agent details in the ascending order of the component types, click on the **Component type** column heading. Also, you can, if you so desire, sort the agent details on any other column, by clicking on the corresponding column heading. The up-arrow mark will then move to that column (see Figure 43).

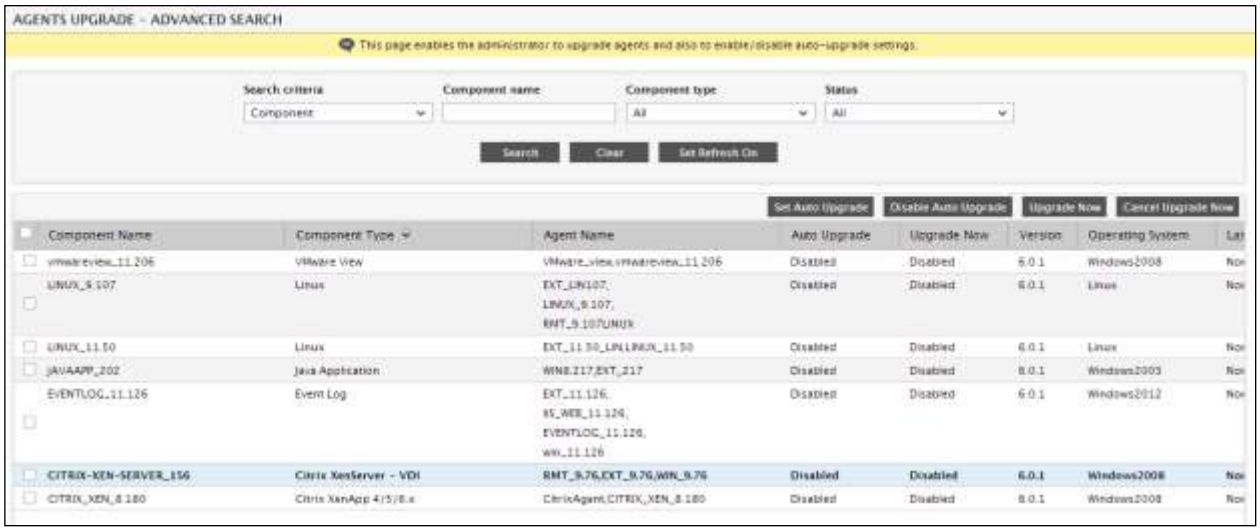

Figure 42: Sorting in the descending order of component types

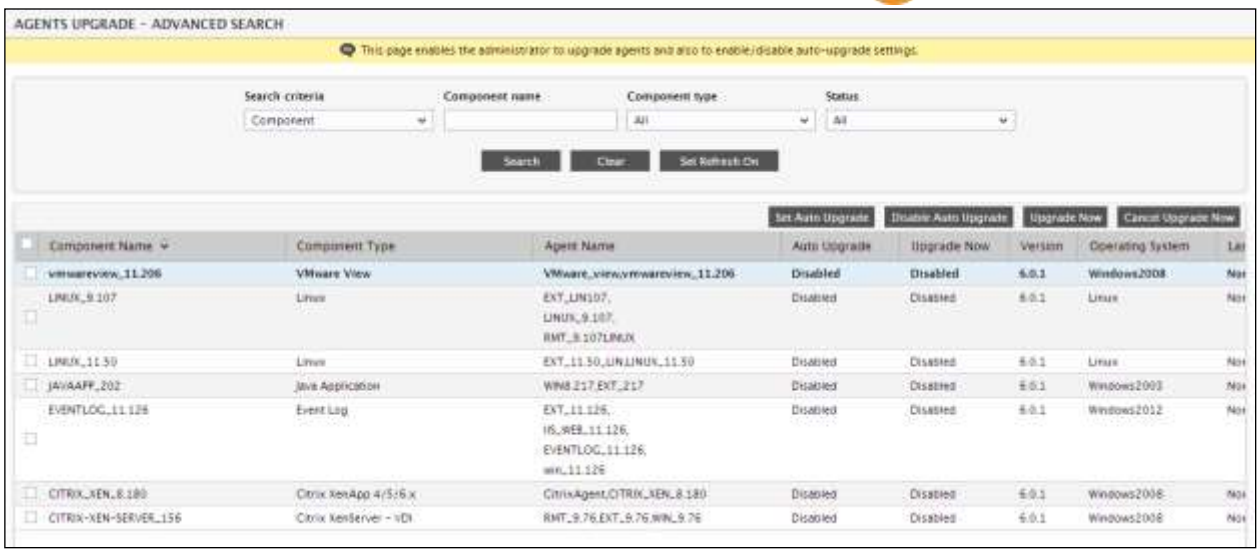

Figure 43: Changing the sort by column

10. At any point in time, you can clear the displayed information by clicking on the **Clear** button in Figure 43.

11. As stated earlier, the **ADVANCED SEARCH** page not only provides agent information, but also allows you to enable/disable auto-upgrade or the 'upgrade now' capabilities of the agents. To enable the auto-upgrade capability of multiple eG agents simultaneously, click on the check boxes that prefix every row of information (in the **ADVANCED SEARCH** page) related to these agents as depicted by Figure 44, and click the **Set Auto Upgrade** button.

Figure 44: Selecting the agents for which auto-upgrade is to be enabled

12. If the auto-upgrade capability was enabled successfully for the chosen agents, then the **Auto Upgrade** column of Figure 45 that appears next, will indicate the same.

Figure 45: Enabling the Auto Upgrade capability

13. Similarly, to disable the auto-upgrade capability, select the check boxes prefixing the corresponding agent details,

and click the **Disable Auto Upgrade** button. The **Auto Upgrade** column will then indicate whether the auto-upgrade capability of those agents was successfully disabled or not.

14. Likewise, you can enable/disable the 'Upgrade now' capability of agents by first selecting the check boxes corresponding to the agent information, and clicking the **Upgrade Now** or **Cancel Upgrade Now** buttons (as the case may be). Figure 46 and Figure 47 indicate the procedure for upgrading a few chosen agents, now (i.e., within the next 15 minutes).

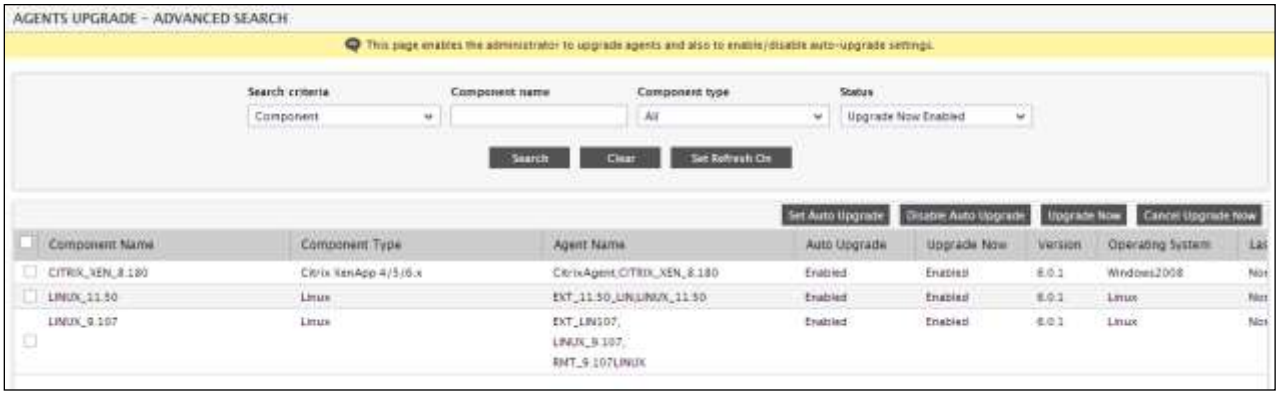

Figure 46: Selecting the agents to be upgraded now

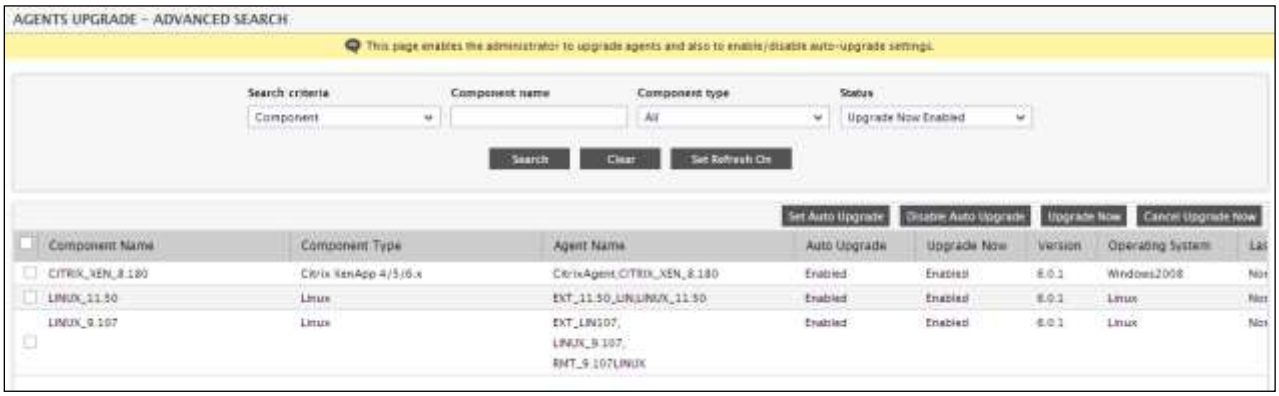

Figure 47: Upgrading the agents within the next 15 minutes

15. By default, the **ADVANCED SEARCH** page does not refresh automatically. Clicking on the **Set Refresh On** button in Figure 48 allows the page to automatically refresh according to a pre-configured refresh period, and also enables administrators to track how long it would be before the next reload occurs (see Figure 48).

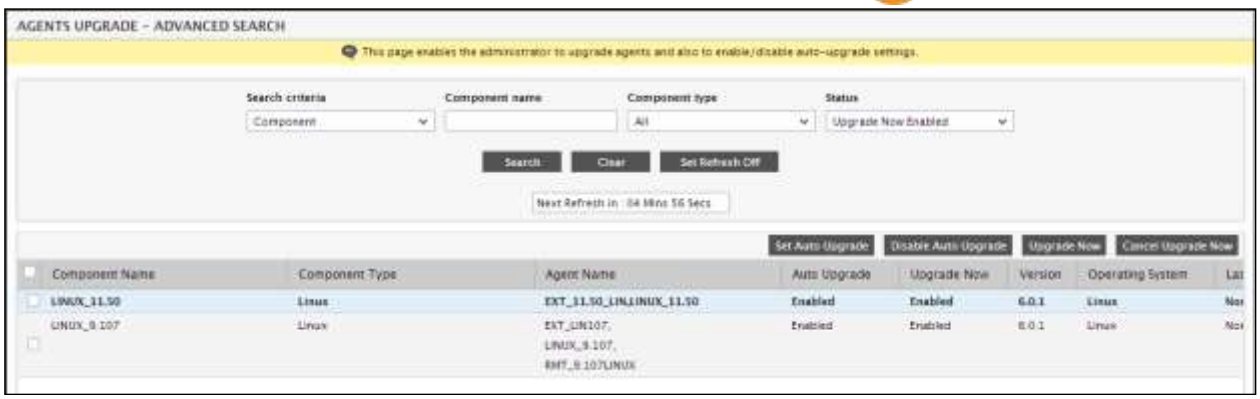

Figure 48: Setting refresh on and tracking time to refresh

Once the **Set Refresh On** button is clicked, the **ADVANCED SEARCH** page refreshes every 5 minutes (i.e., 300 seconds), by default. You can however, modify the refresh period by editing the **eg_ui.ini** file in the **<EG_INSTALL_DIR>\manager\config** directory. The **AutoUpgrade** parameter in the **[REFRESH]** section of this file is set to 300 (seconds) by default. If need be, this default setting can be overridden. To disable the automatic refresh capability of this page, click on the **Set Refresh Off** button in Figure 48.

1.16 Troubleshooting the Agent Upgrade

If the agent auto-upgrade fails, then do the following:

- Check whether the eGAgentmon service is running. The auto-upgrade is performed only by this service. If the agentmon service has been disabled/stopped, the agent auto-upgrade will not happen. Therefore, start this service to initiate the auto-upgrade.
- Take care to commit the manager upgrade (i.e., run the command **eGupgrade.sh –c** in Unix environments, and **eGupgrade.bat** for Windows environments). If not, the agent upgrade will not start.

1.17 Upgrading the eG VM Agent

Only those eG VM Agents that are configured to talk to the eG manager can be upgraded using the procedure mentioned below. Before attempting the upgrade, make sure that you configure the firewall (if any) to allow each eG VM Agent to communicate with the eG manager's TCP port (default: 7077). To upgrade the eG VM Agent, follow the steps below:

1. Login to the eG manager host and create a folder named **VmAgent** in the **<EG_INSTALL_DIR>\manager\config\tests** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**).
2. Connect to the URL: http://www.eginnovations.com/Upgrade724/VMAgent/ - and download the **egvm_win_724_1l.zip** file therein to any location on the manager host.
3. Next, copy the **egvm_win_724_1l.zip** file to the **VmAgent** folder you created in step 1.
4. The next time the eG VM Agent contacts the eG manager, it will check the contents of the **VmAgent** folder, and if any upgrade packs are found in the folder, it will automatically download the pack and upgrade.

Note:

eG VM Agents that are configured to directly talk to a specific remote agent cannot be auto-upgraded. To upgrade such VM agents, you have to uninstall them from the target VMs and then reinstall them.

1.17.1 Troubleshooting the eG VM Agent Upgrade

If the eG VM Agent upgrade fails, check the **<EGVMAGENT_INSTALL_DIR>\logs\VmAutoUpgradelog.log** file for errors and probable reasons for those errors.

1.17.2 Upgrading the JRE of the eG VM Agent

To ensure that the eG VM Agent functions uninterrupted after the upgrade, it is recommended that you upgrade the JRE of the eG VM Agent.

Given below are the steps to be followed to upgrade the JRE used by the eG VM Agent, after upgrading the eG VM Agent:

1. Login to the eG manager host.
2. Connect to the URL: **http://www.eginnovations.com/Upgrade724/Agent/JREUpgrade** - and download the following files to any location on the eG manager host:
   - **egvm_win_725_1l.zip**
   - **egvm_win_x64_725_1l.zip**
3. Next, copy these files to the **VmAgent** folder you created as part of the eG VM Agent upgrade process. Refer to step 1 of Section 1.17 for details. This folder will be available in the **<EG_INSTALL_DIR>\manager\config\tests** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**).
4. The next time the eG VM Agent contacts the eG manager, it will check the contents of the **VmAgent** folder, and if any JRE upgrade packs are found in the folder, it will automatically download the packs and upgrade.

Even if you are upgrading the JRE of only a 32-bit eG VM Agent and not a 64-bit one, you will have to download both the 32-bit and 64-bit JRE upgrade packs for the eG VM Agent and copy the downloaded packs to the **<EG_AGENT_INSTALL_DIR>\manager\config\tests** directory (on Windows; on Unix, this will be **opt/egurkha/manager/config/tests**).

The same applies when you are upgrading the JRE of a 64-bit eG VM Agent.

1.18 Upgrading an eG Agent Installed on a Golden Image

To upgrade an eG agent installed on a golden image to v7.2.4, you need to uninstall the old eG agent on the golden image, and then install version 7.2.4 of the eG agent on it. The steps to achieve this are as follows:

1. Login to the VM hosting the golden image.
2. Uninstall the old eG agent on that VM.
3. Then, go to the URL: http://www.eginnovations.com/eval724/
4. Click on the **Agent** folder within, open the sub-folder that corresponds to your VM's operating system, and download the contents of that sub-folder to any location on the VM.
5. Then, using Windows explorer, open the local folder to which you downloaded the eG agent installable at step 4 above. Run the ***.exe** in that folder to install the eG agent on the VM. **When installing and configuring the eG agent on a golden image, make sure that you do not set a nick name for that agent.**

1.19 How to Upgrade a Redundant Setup?

To upgrade a redundant setup, follow the steps given below:

1. If required, run the backup routine discussed in Section 1.1 separately for each of the managers and their respective databases.
2. With the secondary managers running, first upgrade the primary manager. The upgrade procedure is the same as that which is discussed in Section 1.2.
3. Once upgrading is complete, stop the secondary managers, and then start the primary manager.
4. Finally, upgrade the secondary managers and then start them.
5. Once the managers are started, make sure that the agent upgrade packs are copied to both the managers in the cluster as described in Section 1.6.

Step 5 needs to be followed whether/not you have agents reporting to the secondary manager.

6. The eG agents download the upgrade package from the eG manager to which they report. The agent upgrade procedure is the same as that which is discussed in Section 1.6.
7. The primary manager and the secondary managers need to be up and running for the agent upgrade to function without a glitch.

Both the managers in a redundant setup should be of the same version.

1.20 Database Optimizations

Subsequent to upgrading the eG manager and agents, **it is mandatory to optimize the eG database** to enhance its performance.

As stated already, the eG manager upgrade process automatically analyzes the eG database and computes its total size. In the process, the eG manager upgrade automatically performs the following:

1. Identifies the tables in the eG database that require new indexes or need their existing indexes modified;
2. Compares the size of the tables identified in step 1 above, and picks the top tables in terms of size

You should proceed with the optimization based on the database size reported by the upgrade process. The sub-sections below explain the steps to optimize the database depending upon its size. They also indicate roughly how much time it may take for the process to complete. **Note that the time estimates given below are approximate, and may vary depending upon the actual database size and defragmentation levels.**

1.20.1 Optimizing an eG Database Over 500 GB in Size

If the upgrade process reports that your database is over 500 GB in size, then, after upgrading the eG manager and agents, you **must perform database indexing in 2 phases**, so that the eG database performance improves. These 2 phases are discussed below:

1. In the first phase, you need to defragment and reindex the top tables (in terms of size) identified by the eG manager upgrade process. This activity can be performed in the OFFLINE or ONLINE mode. You can choose the mode by weighing the following factors:
    - The total size of the eG database;
    - Whether/not the eG manager and the eG agent (monitoring the eG database) should be stopped until defragmenting and reindexing is complete;
    - The total time taken for defragmenting and reindexing the top tables

    Let us now take a look at each mode:

    - **OFFLINE:** In the OFFLINE mode, **you should stop the eG manager and agent** (monitoring the eG database backend) before defragmenting/reindexing the top tables.

      To defragment and reindex say, the *top 10* tables of an eG database that is between *5 TB – 10 TB in size*, it will approximately take *10 – 20 hours* (as the case may be). As the database size and number of tables increase, the duration of this activity will also increase.

      For defragmenting/reindexing tables in the OFFLINE mode, follow the steps below:

        - Stop the eG manager/agent.
        - Open the appropriate database client (e.g., Oracle SQL Net Client, Microsoft Enterprise Manager / Query Analyzer)

- Using the client, execute the queries contained within the <EG_HOME>\bin**QueryManualExecuteRebuildIndex_OFFLINE.sql** file (on Windows; on Unix, this will be opt/egurkha/bin/**QueryManualExecuteRebuildIndex_OFFLINE.sql**), one after another.
- Once the top tables are successfully defragmented/reindexed, start the eG manager/agent.

If you do not want the eG manager and agent to remain unavailable until the whole defragmenting/reindexing exercise is completed, you can even distribute the query execution across multiple time windows. This way, you can start the eG manager after one set of queries is executed, and stop it again only before another set is run.

- **ONLINE:** In the ONLINE mode, the top tables can be defragmented and reindexed **while the eG manager and agent are up and running**. In other words, you need not stop the eG manager/agent in this mode. The time taken for online reindexing may be double the time it takes for offline reindexing. You can opt for this mode if you want to avoid eG manager/agent downtime.

  For defragmenting/reindexing tables in the ONLINE mode, follow the steps below:

  - Open the appropriate database client (e.g., Oracle SQL Net Client, Microsoft Enterprise Manager / Query Analyzer)
  - Using the client, execute the queries contained within the <EG_HOME>\bin**QueryManualExecuteRebuildIndex_ONLINE.sql** file (on Windows; on Unix, this will be opt/egurkha/bin/**QueryManualExecuteRebuildIndex_ONLINE.sql**), one after another.

2. In the second phase, you have to create new indexes for all the tables in the eG database. The eG manager **must be stopped** before you begin creating the indexes. The estimated time required to complete this process would be approximately 10 hours for a database of size 10 TB.

   For new index creation, follow the steps below:

   - Stop the eG manager.
   - Execute the file, <EG_HOME>\bin**UpgradeIndex.bat** (on Windows; on Unix, this will be /opt/egurkha/bin/**UpgradeIndex.sh**).
   - Type **C** at the prompt that appears.
   - Doing so will create an **IndexUpgrade.sql** file in the <EG_HOME>\bin directory (on Windows; on Unix, this will be the /opt/egurkha/bin directory). This file will contain the list of new indexes to be created and the queries to be used to create them.
   - Open the appropriate database client (e.g., Oracle SQL Net Client, Microsoft Enterprise Manager / Query Analyzer)
   - Using the client, execute the queries contained within the **IndexUpgrade.sql** file, one after another.
   - After all the required indexes are successfully created, start the eG manager.

1.20.2 Optimizing an eG Database that is Less than 500 GB in Size

On the other hand, if the upgrade process reports that your database is less than 500 GB in size, then do the following:

1. First, reindex the entire eG database. This can be performed in the ONLINE or OFFLINE mode.

   In the OFFLINE mode, **you should stop the eG manager/agent** before defragmenting/reindexing the eG database.

   To reindex in the OFFLINE mode, do the following:

   - Stop the eG manager/agent;

- Open the command prompt and switch to the <EG_HOME>\bin directory (on Windows; on Unix, this will be the /opt/egurkha/bin/ directory).
- Issue the following command at the prompt:

  ExecuteIndexes OFFLINE

- Start the eG manager/agent after the database is fully reindexed.

The time taken for this activity may vary depending upon the database size and current defragmentation level.

If you do not want the eG manager and agent to remain unavailable until the whole defragmenting/reindexing exercise is completed, you can even distribute the query execution across multiple time windows. This way, you can start the eG manager after one set of queries is executed, and stop it again only before another set is run.

In the ONLINE mode, the eG database can be defragmented and reindexed **while the eG manager and agent are up and running**. In other words, you need not stop the eG manager/agent in this mode. The time taken for online reindexing may be double the time it takes for offline reindexing. You can opt for this mode if you want to avoid eG manager/agent downtime.

To reindex the eG database in the ONLINE mode, do the following:

- Open the command prompt and switch to the <EG_HOME>\bin directory (on Windows; on Unix, this will be the /opt/egurkha/bin/ directory).
- Issue the following command at the prompt:

  ExecuteIndexes ONLINE

2. Next, proceed to create new indexes in all the tables of the eG database. The eG manager **must be stopped** before you begin creating the indexes. The time taken for this activity may vary depending upon the database size and current defragmentation level.

   For new index creation, follow the steps below:

   - Stop the eG manager.
   - Execute the file, <EG_HOME>\bin**UpgradeIndex.bat** (on Windows; on Unix, this will be /opt/egurkha/bin/**UpgradeIndex.sh**).
   - Type **C** at the prompt that appears.
   - Doing so will create an **IndexUpgrade.sql** file in the <EG_HOME>\bin directory (on Windows; on Unix, this will be the /opt/egurkha/bin directory). This file will contain the list of new indexes to be created and the queries to be used to create them.
   - Open the appropriate database client (e.g., Oracle SQL Net Client, Microsoft Enterprise Manager / Query Analyzer)
   - Using the client, execute the queries contained within the **IndexUpgrade.sql** file, one after another.
   - After all the required indexes are successfully created, start the eG manager.

1.21 Upgrade FAQ

Given below are a list of frequently asked questions pertaining to the agent and manager upgrade procedures, and their corresponding answers. Refer to this list, whenever in doubt.

a. I have added custom tests and components to my eG Enterprise deployment. Will I need to apply these once again I upgrade?

No. This will not be necessary, as upgrade does not affect the configurations performed in the original version.

b. Will I lose existing historical data when I upgrade the eG Enterprise manager and agents?

No, you will not.

c. Say, I do not have permission to upgrade some agents. Can I continue to run the agents in an older version and have them report to the upgraded eG manager?

Yes, this is possible. However, new tests and new measures added in the latest version will not be available to the old agents.

d. I have created custom reports and favorites. Will I have to create these once I upgrade the eG manager?

If the relevant report options have not been removed in the latest version, then the custom reports and favorites will remain.

e. After the upgrade of the eG Manager, if I need to install agents, should I deploy agents for the newer version?

In order to take advantage of the extended monitoring capabilities, it is recommended that you use an agent that matches in version with the manager. However, this is not mandatory. At the same time, note that new tests and new measures added in the latest version will not be available to the old agents.