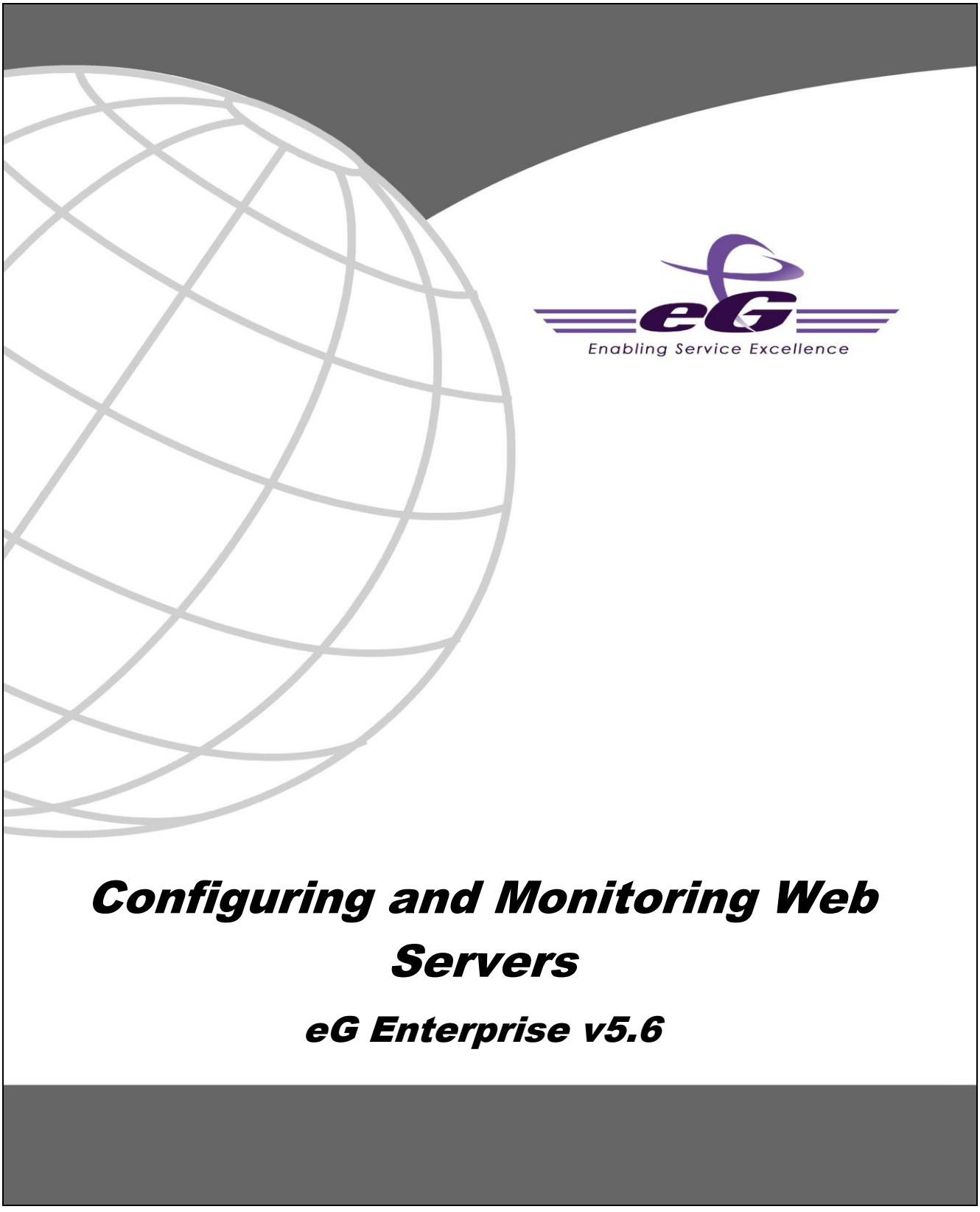

Configuring and Monitoring Web Servers

eG Enterprise v5.6

Restricted Rights Legend

The information contained in this document is confidential and subject to change without notice. No part of this document may be reproduced or disclosed to others without the prior permission of eG Innovations, Inc. eG Innovations, Inc. makes no warranty of any kind with regard to the software and documentation, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Trademarks

Microsoft Windows, Windows NT, Windows 2000, Windows 2003 and Windows 2008 are either registered trademarks or trademarks of Microsoft Corporation in United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Copyright

Table of Contents

Table of Figures

Chapter 1

Configuring and Monitoring an Apache Web Server/IBM HTTP Server

Configuring an Apache web server/IBM HTTP Server to be monitored by an eG agent is an easy two step process.

- Configuring the web server to work with the eG agent
- Configuring the eG manager through its user interface to monitor the web server

1.1 Configuring an Apache/IBM HTTP Web Server on Unix

eG Enterprise's unique web adapter technology enables individual transactions performed by users of a web site to be tracked in real-time without the need for explicit, expensive logging.

The web adapter must be configured for each and every web server that must be monitored by the eG agent. This adapter is part of the eG agent package for Solaris. In case of an Apache/IBM HTTP server on the other hand, a manual procedure needs to be followed in order to configure the web adapter.

To manually configure the eG web adapter on an Apache web server 1.x on Unix, do the following:

1. First, login to the Unix server as the Apache install user.
2. Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following lines to the end of the file:

   LoadModule eg1_module libexec/mod_eg1.so
3. Copy the file **mod_eg1.so** from the **/opt/egurkha/lib** directory to **<APACHE_HOME>/libexec** in the **<APACHE_INSTALL_USER>** directory.
4. Stop and restart the Apache server.

The same procedure applies while configuring the web adapter on an IBM HTTP Server 1.x on Unix.

To configure the web adapter on Apache 1.x on HPUX/AIX servers, the procedure is almost the same as what has been discussed above; however, the only difference is that you will have to append the following lines to the end of the **<APACHE_HOME>/conf/httpd.conf** file:

LoadModule mod_egurkha libexec/mod_egurkha.so

To manually configure the eG web adapter on an Apache web server 2.0 on Unix, do the following:

1. First, login to the Unix server as the Apache install user.
2. Edit the <APACHE_HOME>/conf/httpd.conf file to append the following line:

   LoadModule eg2_module modules/mod_eg2.so
3. Copy the file **mod_eg2.so** from the **/opt/egurkha/lib** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
4. Stop and restart the Apache server.

The same procedure applies for configuring an IBM HTTP Server 2.0 on Unix

Note that you cannot configure the web adapter on Apache web server 2.0 / IBM HTTP Server 2.0 for HPUX.

To manually configure the eG web adapter on an Apache web server 2.2 on Unix, do the following:

1. First, login to the Unix server as the Apache install user.
2. Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following line:

   LoadModule eg2_module modules/mod_eg22.so
3. Copy the file **mod_eg22.so** from the **/opt/egurkha/lib** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
4. Stop and restart the Apache server.

To configure the eG web adapter on Apache 2.2 on HPUX/AIX, follow the same procedure explained above.

To manually configure the eG web adapter on an Apache web server 2.4 on Linux, do the following:

1. First, login to the Unix server as the Apache install user.
2. Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following line:

   LoadModule eg2_module modules/mod_eg24.so
3. Copy the file **mod_eg24.so** from the **/opt/egurkha/lib** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
4. Stop and restart the Apache server.

To configure the eG web adapter on Apache 2.4 on HPUX/AIX, follow the same procedure explained above.

To manually configure the eG web adapter on an IBM HTTP Server 2.x on AIX, do the following:

1. First, login to the AIX server as the IBM HTTP install user.
2. Edit the **<IBM_HTTP_HOME>/conf/httpd.conf** file to append the following line:

   LoadModule eg2_module modules/mod_ibm_eg2.so
3. Copy the file **mod_ibm_eg2.so** from the **/opt/egurkha/lib** directory to **<IBM_HTTP_HOME>/modules** under **<IBM_HTTP_INSTALL_USER>**.
4. Stop and restart the IBM HTTP server.

To manually configure the eG web adapter on an IBM HTTP Server 6.x on Unix (Linux/Solaris/AIX), do the following:

1. First, login to the Unix server as the IBM HTTP install user.
2. Edit the **<IBM_HTTP_HOME>/conf/httpd.conf** file to append the following line:

   LoadModule eg2_module modules/mod_ibm_eg6.so
3. Copy the file **mod_ibm_eg6.so** from the **/opt/egurkha/lib** directory to **<IBM_HTTP_HOME>/modules** under **<IBM_HTTP_INSTALL_USER>**.
4. Stop and restart the IBM HTTP server.

Note that the eG web adapter cannot be configured on an IBM HTTP Server 6.x on HPUX.

1.1.1 Configuring the eG Web Adapter for an Apache Web Server on a 64-bit Linux Operating System

To configure the eG web adapter for an Apache web server on a 64-bit Linux host, do the following:

1. By default, the Java Runtime Environment (JRE) version 1.5 for 32-bit operating systems is bundled as part of the eG agent installable for Linux. To ensure that such an agent works smoothly on a 64-bit Linux operating system, follow the steps given below to change the JRE used by the eG agent, after deploying the standard Linux agent on the 64-bit Linux host:
   - Stop the eG agent.
   - Login to the eG agent host as the eG install user. You will currently be working in the **/opt/egurkha** directory.
   - Move the JRE directory that is used by the eG agent (by default) to another location, say **/tmp**.

     mv jre /tmp
2. If a 64-bit-compatible version of the JRE is already available on the eG agent host, provide a soft link to that directory using the following command:

   ln -s <Full_path_to_the_directory_containing_the_64-bit-compatible_JRE> jre

   For instance, if the 64-bit version of JRE is available in the **/opt/usr/JRE** directory, then the command will be:

   ln -s /opt/usr/JRE jre
3. On the other hand, if a compatible JRE does not pre-exist on the agent system, then download and install the 64-bit version of the JRE from the java.sun.com web site.
4. Then, provide a soft link to the JRE directory using the command indicated by step 4 above.

5. The eG agent for the 64-bit Linux host is bundled with the following shared libraries to be used by the eG web adapter, if configured on the host:
   - mod_eg24.so
   - mod_eg22.so
   - mod_eg2.so
   - libeg_reptr_cat.so
   - libeg_reptr_total.so
   - libeg_reptr_site.so

   These files are available in the **/opt/egurkha/lib/lib64** directory on the host.
6. To enable the eG web adapter for Apache 2.0, following the steps given below:
   - First, login to the Linux host as the Apache install user.
   - Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following line:
   - *LoadModule eg2_module modules/mod_eg2.so*
   - Copy the file **mod_eg2.so** from the **/opt/egurkha/lib/lib64** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
   - Copy the **libeg*.so** files from the **/opt/egurkha/lib/lib64** directory to the **/opt/egurkha/lib** directory.
   - Stop and restart the Apache server.
7. To enable the eG web adapter for Apache 2.2, following the steps given below:
   - First, login to the Linux host as the Apache install user.
   - Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following line:

     LoadModule eg2_module modules/mod_eg22.so
   - Copy the file **mod_eg22.so** from the **/opt/egurkha/lib/lib64** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
   - Copy the **libeg*.so** files from the **/opt/egurkha/lib/lib64** directory to the **/opt/egurkha/lib** directory.
   - Stop and restart the Apache server.
8. To enable the eG web adapter for Apache 2.4, following the steps given below:
   - First, login to the Linux host as the Apache install user.
   - Edit the **<APACHE_HOME>/conf/httpd.conf** file to append the following line:

     LoadModule eg2_module modules/mod_eg24.so
   - Copy the file **mod_eg24.so** from the **/opt/egurkha/lib/lib64** directory to **<APACHE_HOME>/modules** under **<APACHE_INSTALL_USER>**.
   - Copy the **libeg*.so** files from the **/opt/egurkha/lib/lib64** directory to the **/opt/egurkha/lib** directory.
   - Stop and restart the Apache server.
9. Start the eG agent. 0.

1.2 Configuring an Apache/IBM HTTP Web server on Windows Environments

The eG web adapter can be configured on an Apache/IBM HTTP web server on Windows, using a manual configuration process only. The same has been discussed below.

To manually configure the eG web adapter on an Apache web server 1.x on Windows, do the following:

1. First, login to the Windows server.
2. Edit the **<APACHE_HOME>\conf\httpd.conf** file to append the following lines:

   AddModule mod_egurkha.c

   LoadModule mod_egurkha modules/mod_egurkha.dll
3. Copy the file **mod_egurkha.dll** from the **<EG_AGENT_INSTALL_DIR>\lib** directory to **<APACHE_HOME>\modules**.
4. Stop and restart the Apache server.

To manually configure the eG web adapter on an Apache web server 2.0 on Windows, do the following:

1. First, login to the Windows server.
2. Edit the **<APACHE_HOME>\conf\httpd.conf** file to append the following line:

   LoadModule egurkha_module modules/mod_egurkha2_0.dll
3. Copy the file **mod_egurkha2_0.dll** from the **<EG_AGENT_INSTALL_DIR>\lib** directory to **<APACHE_HOME>\modules**.
4. Stop and restart the Apache server.

To manually configure the eG web adapter on an Apache web server 2.2 on Windows, do the following:

1. First, login to the Windows server.
2. Edit the **<APACHE_HOME>\conf\httpd.conf** file to append the following line:

   LoadModule egurkha_module modules/mod_egurkha2_2.dll
3. Copy the file **mod_egurkha2_2.dll** from the **<EG_AGENT_INSTALL_DIR>\lib** directory to **<APACHE_HOME>\modules**.
4. Stop and restart the Apache server.

To manually configure the eG web adapter on an IBM HTTP server 1.x on Windows, do the following:

1. First, login to the Windows server.
2. Edit the **<IBM_HTTPSERVER_HOME>\conf\httpd.conf** file to append the following line:

   LoadModule ibm_egurkha_module modules/ibm_mod_egurkha.dll
3. Copy the file **ibm_mod_egurkha.dll** from the **<EG_AGENT_INSTALL_DIR>\lib** directory to **<IBM_HTTPSERVER_HOME>\modules**.
4. Stop and restart the IBM HTTP server.

To manually configure the eG web adapter on an IBM HTTP web server 2.x on Windows, do the following:

1. First, login to the Windows server.
2. Edit the **<IBM_HTTPSERVER_HOME>\conf\httpd.conf** file to append the following line:

   LoadModule ibm_ egurkha_module modules/ibm_mod_egurkha.dll
3. Copy the file **ibm_mod_egurkha2_0.dll** from the **<EG_AGENT_INSTALL_DIR>\lib** directory to **<IBM_HTTPSERVER_HOME>\modules**.
4. Rename the **<IBM_HTTPSERVER_HOME>\modules\ibm_mod_egurkha2_0.dll** to **ibm_mod_egurkha.dll**
5. Stop and restart the IBM HTTP server.

1.3 Administering the eG Manager to monitor the Apache/IBM HTTP Web Server

To discover a web server automatically, the port on which it is running should be configured. To achieve this, do the following:

1. Log into the eG administrative interface as admin.
2. In the **ADMIN HOME** page that appears, use the following menu sequence: Infrastructure-> Components -> Discover.
3. In the Discovery page that comes up, click the **Change Preferences** button and using the **CHANGE PREFERENCES** page that appears next (see Figure 1.1), configure the port on which the web server is listening. Once done, click on the **Update** button.

Figure 1.1: Configuring the web server port for automatic discovery

4. After updating the changes, begin the discovery process by clicking on the **Start Discovery** button in the **START DISCOVERY** page.

5. Discovered components will then have to be managed manually using the **COMPONENTS - MANAGE/UNMANAGE** page that comes up on following the menu sequence: Infrastructure-> Components -> Manage/Unmanage.

6. In this screen, select **Web** server from the **Component type** drop down list as depicted by Figure 1.2 below.

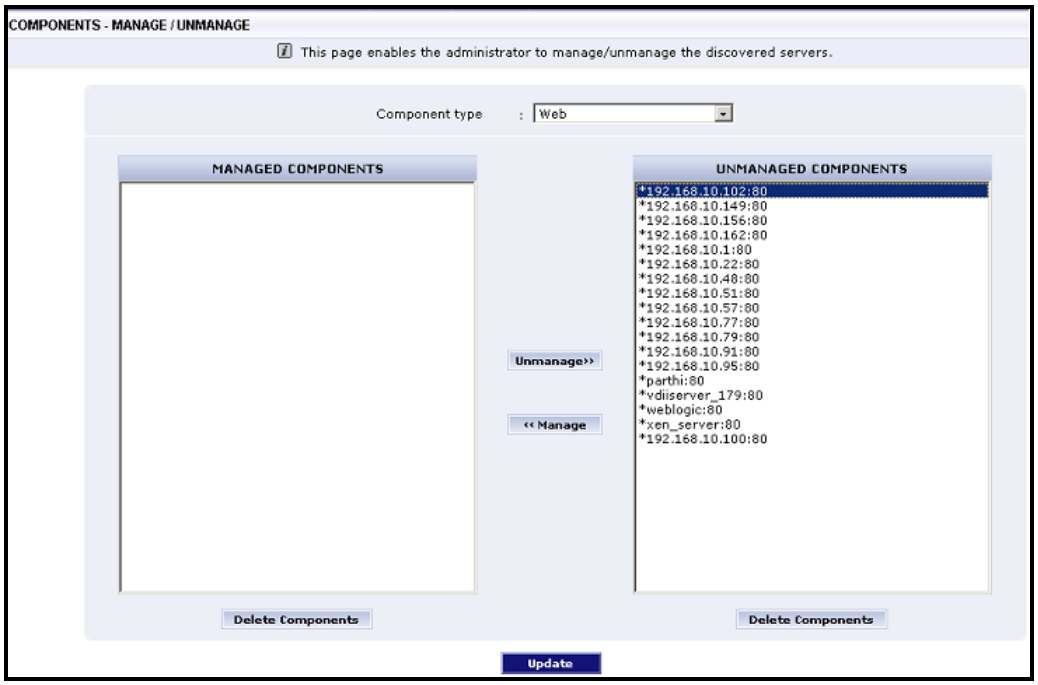

Figure 1.2: Selecting Web server from the drop-down list

7. The host names / IP addresses of the discovered but unmanaged web servers will then be populated in the **UNMANAGED COMPONENTS** list. To manage the web server, select it from the list, click the << **Manage** button and add it to the **MANAGED COMPONENTS** list as shown in Figure 1.3 below.

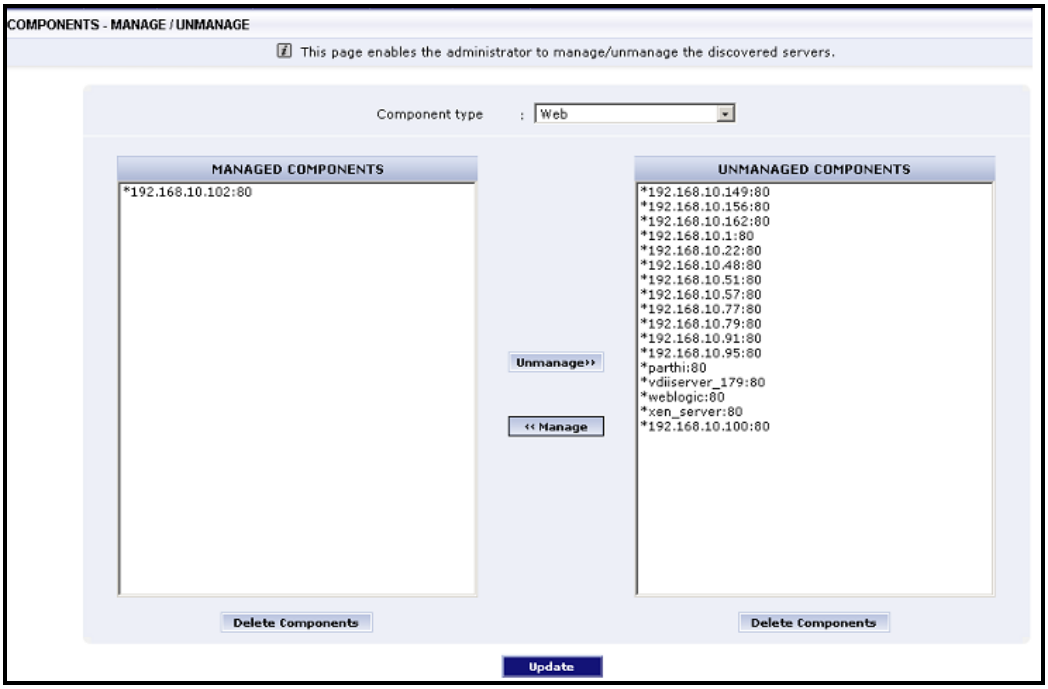

Figure 1.3: Managing the apache web server manually

8. Finally, register the changes by clicking the **Update** button.

9. If the web server is not automatically discovered, manually add it to the environment using the **ADD/MODIFY COMPONENTS** page. Components so added will automatically find a place in the **MANAGED COMPONENTS** list in the **COMPONENTS - MANAGE/UNMANAGE** page above (see Figure 1.3). To access the **ADD/MODIFY COMPONENTS** page, follow the menu sequence: Infrastructure -> Components -> Add/Modify.

10. Here, select **Web** server from the **Component type** drop-down list (see Figure 1.4) and then, click the **Add New Component** button to add a new Apache Web server.

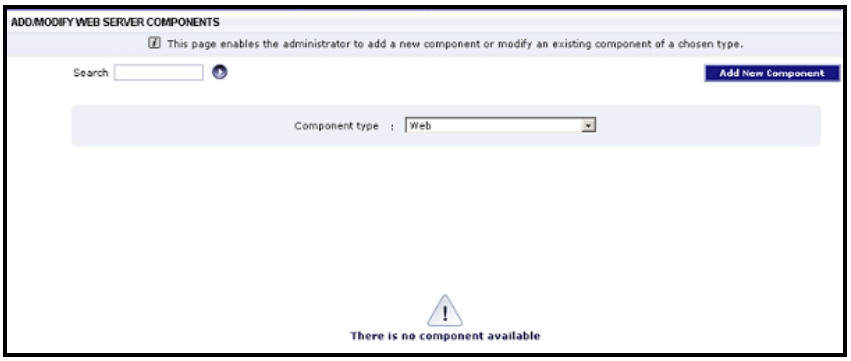

Figure 1.4: Selecting the Web server option from the drop-down list in the ADD/MODIFY page

11. In the **NEW COMPONENT DETAILS** page that appears, provide the details requested as depicted by Figure 1.6. If a valid hostname is specified in 1.6, make sure that this name can be resolved to the corresponding IP address via the DNS server of the target infrastructure. Also, indicate whether the component being added is to be monitored in an **Agentless** manner or not. This option will be available

to you only if the eG license enables the agentless monitoring capability. Moreover, if the **Agentless** option in Figure 1.6 s set to **No**, then an **Internal agent assignment** option will appear. If this is set to **Manual**, then you can associate multiple IPs/nicknames on a host with a single internal agent. The default selection **Auto** indicates that every IP/nick name on a host will be associated with a separate agent. For more details about the **Agentless** and **internal agent assignment** options, refer to the *eG User Manual*. Finally, select the **External Agent** that will monitor the component being added from an external perspective, and click the **Add** button in Figure 1.6 to register the changes.

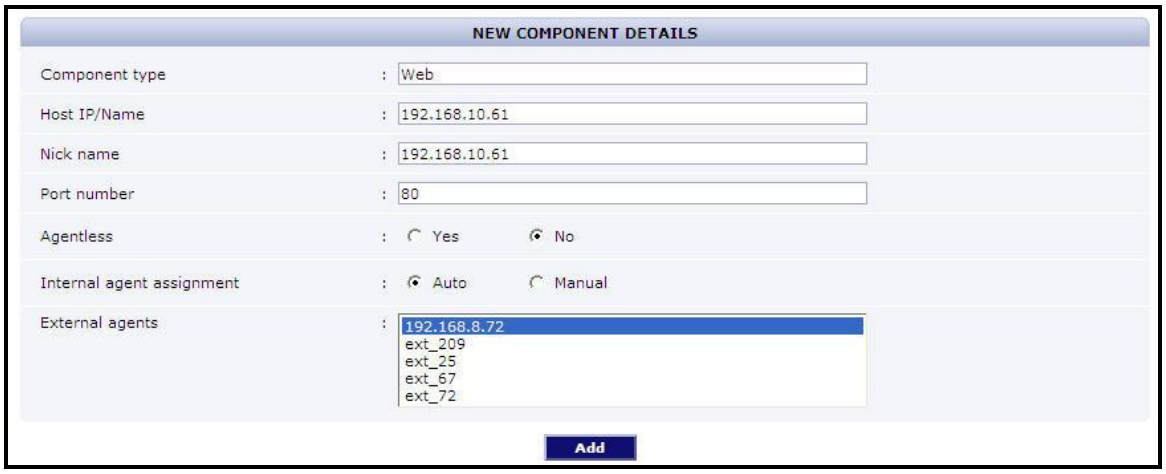

Figure 1.5: Providing the new server details

12. Once the information is successfully updated, you will receive the confirmation via the following screen:

Figure 1.6: Viewing a summary of the details of a new server

13. Then, proceed to configure the services for the web server. A service can be a group of independent components / components that belong to a segment topology / a web site that can be hosted on one or more web servers. In other words, a web site is a subset of a service. The various services that users can avail via a web site are referred to as **transactions**. To configure services, first, follow the menu sequence, Infrastructure ->Services -> Configuration. This will open the **LIST OF SERVICES** page (see Figure 1.7).

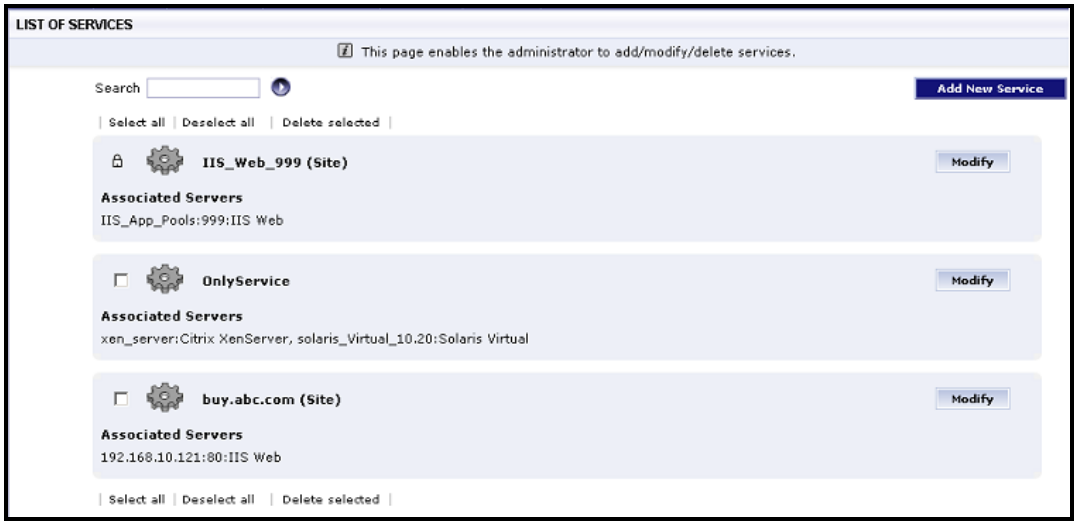

Figure 1.7: A page listing existing web sites

14. Then, click on the **Add New Service** button on the screen. In the screen that appears (see Figure 1.8) the **Name of the service** in the text box provided. Then, from the **Is this service a website** list box, select **Yes**, so as to indicate that the new service is a web site.

Note

While adding a service that is **not** a web site, the **No** option needs to be selected from the **Is this service a website** list box.

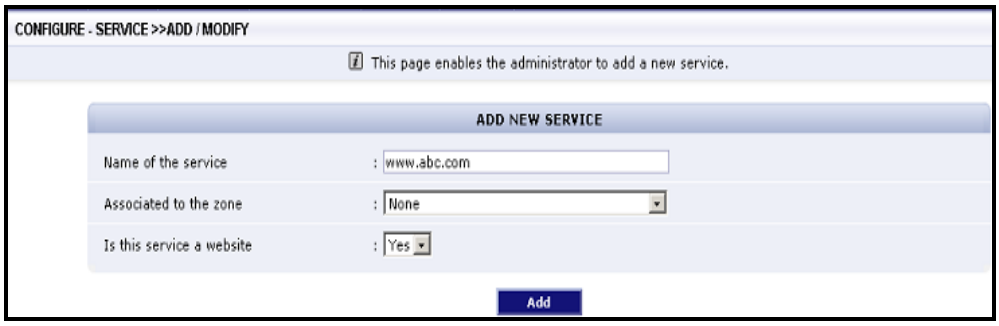

Figure 1.8: Adding a new service

15. Finally, click the **Update** button. Upon clicking, Figure 1.9 will appear, which will help an administrator configure a service that is a web site.

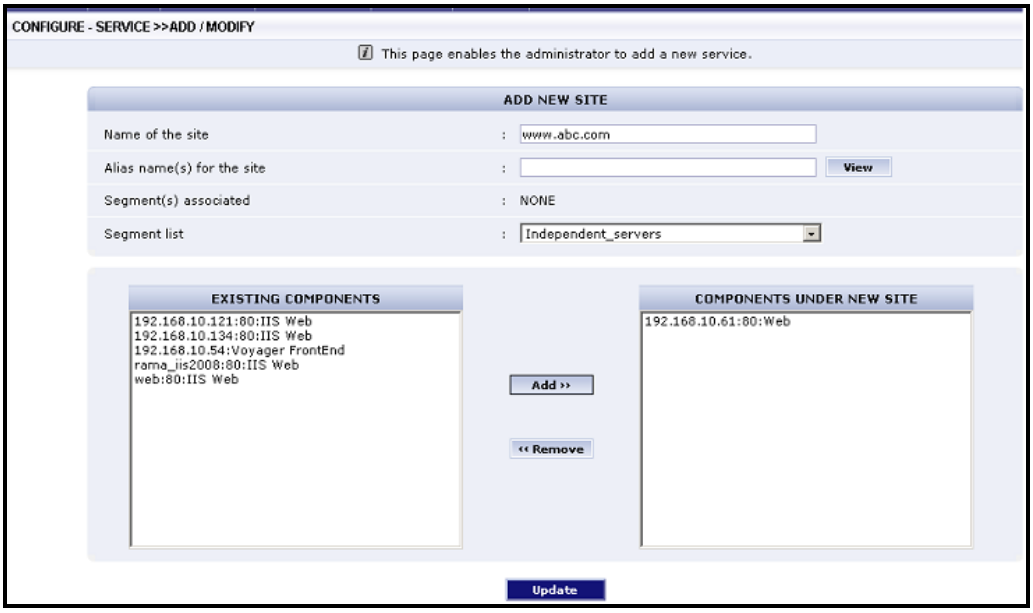

Figure 1.9: Configuring a web site

16. The **Name of the site** box in Figure 1.9 will display the **Name of the service** specified earlier in Figure 1.8. The administrator can change it, if need be.

Note

While adding a service that is **not** a website, the **Name of the site** text box of Figure 1.8 will be replaced by the **Name of the service** text box.

17. A single site can be addressed by various other names in the environment (e.g., www.abc.com may also be accessed as www.abc.com:80, abc.com, us.abc.com, 172.169.10.20 etc.). These names (or IP address:port combinations) can be specified in the **Alias name(s) for the site** text box. To ensure that all requests to a website are captured, it is essential to ensure that all the alias names for a site are specified accurately. The administrators can specify a maximum of six alias names, each of which should be comma separated. While multiple alias names can be specified for a site, in the monitor interface, all the statistics pertaining to this web site will be reported using its site name displayed in the **Name of the site** text box.

Note

Alias names are applicable to web sites alone. Therefore, while configuring a service that is **not** a web site, the **Alias name(s) for the site** text box will not appear.

18. The **Segment list** box contains the list of fully configured segments in the target environment that contain atleast a single web or application server. The site can be associated with any of these segments. The **Independent_servers** option in this list box enables the administrator to associate a site to a single web or application server that does not form a part of the segment topology. By default, a managed web server will be treated as an independent web server by the eG Enterprise system. Therefore, select the **Independent_servers** option from the **Segment list**.

In case of services that are **not** web sites, the **Segment list** list box will list all the fully configured segments in the target environment – not just the segments that contain web / application servers.

19. Once a site is associated with a segment, the user interface lists the web and application servers that form a part of the selected segment (or, if the **Independent_servers** option is chosen, the list of independent web and application servers in the target environment), in the **EXISTING COMPONENTS** list box. The components associated with the site are available in the **COMPONENTS UNDER NEW SITE** list box. The administrator can associate an existing component with the site by selecting the component and then clicking the **Add >>** button. Similarly, an associated component can be removed by clicking the **<< Remove** button.

Unlike web sites that can be associated only with web / application servers, services that are **not** web sites can be associated with any component in the selected segment, or any independent component.

20. Finally, click the **Update** button to register the changes. On doing so, the following screen will pop up:

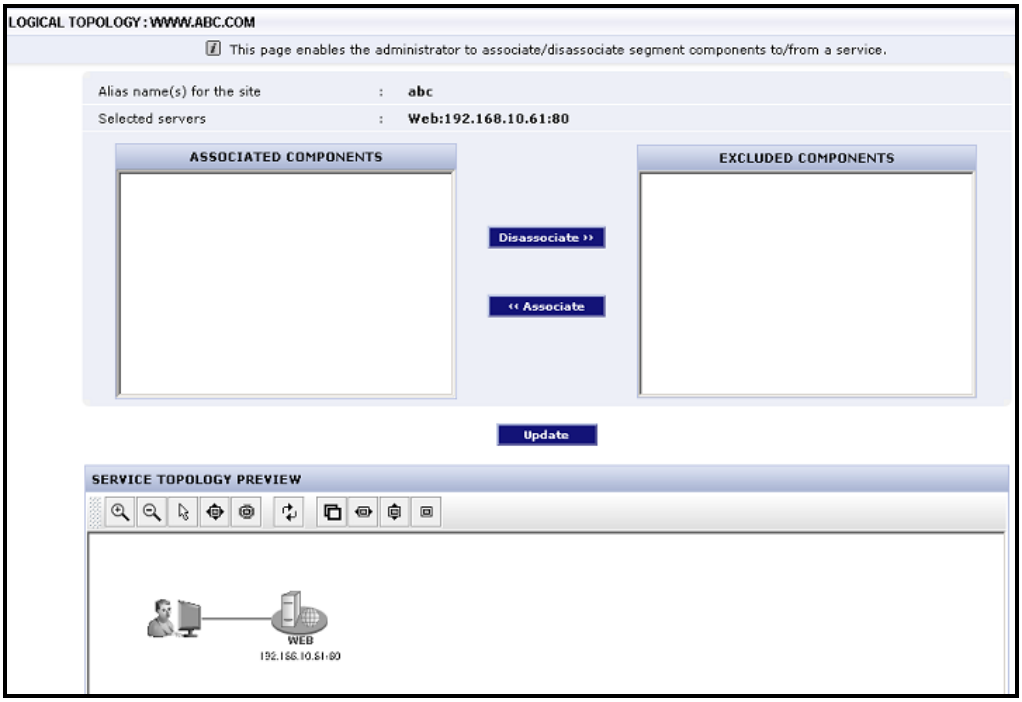

Figure 1.10: Configuring the dependencies of a web site

21. A web site inherits all of the dependencies of the web server(s) with which it is associated. In many instances, a web site may not use all the components that a web server on which it is hosted, is configured to use. To support such a scenario, the eG Enterprise suite allows the administrator to disassociate specific components for a web site. Figure 1.10 indicates the configuration of dependencies for a web site. The components associated with a site are shown in the **ASSOCIATED COMPONENTS** box and the non-related ones in the **EXCLUDED COMPONENTS** box. When a site is added for the first time, all the components that are associated with the corresponding web server(s) are available in the **ASSOCIATED COMPONENTS** list box. The administrator can associate or disassociate the components using the **Associate** or the **Disassociate** buttons. In our example, however, the web server associated with the site is an independent one. Hence, simply click the **Update** button and proceed to configure transactions.

22. Administrators can configure transactions to reflect the key operations performed by users of the web site. For performing this activity, first, open the **LIST OF SITES AND TRANSACTIONS** page (see Figure 1.11) using the following menu sequence: Infrastructure->Services->Transactions.

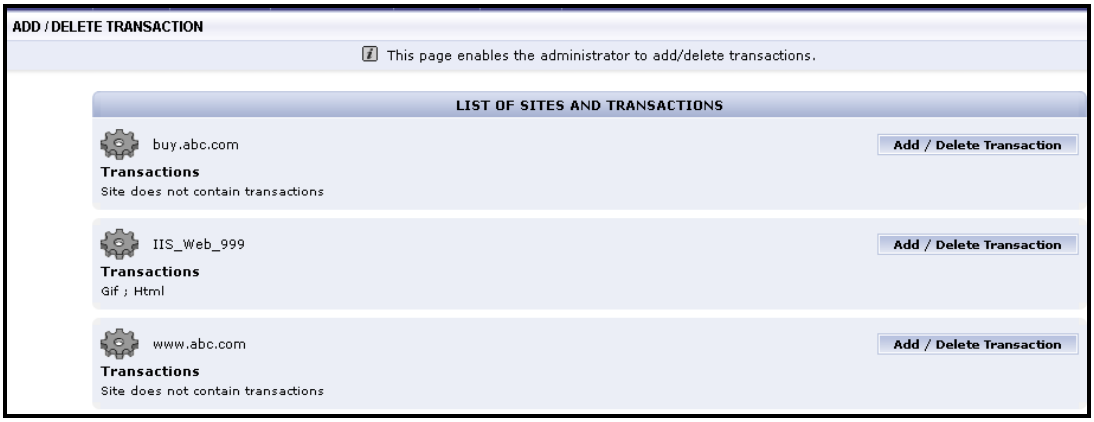

Figure 1.11: A page listing existing transactions of a web site

Note

Transactions can be configured for web sites only. Therefore, Figure 1.12 will not list the services that are not websites.

23. Then, click the **Add/Delete Transaction** button therein, thereby opening the following page:

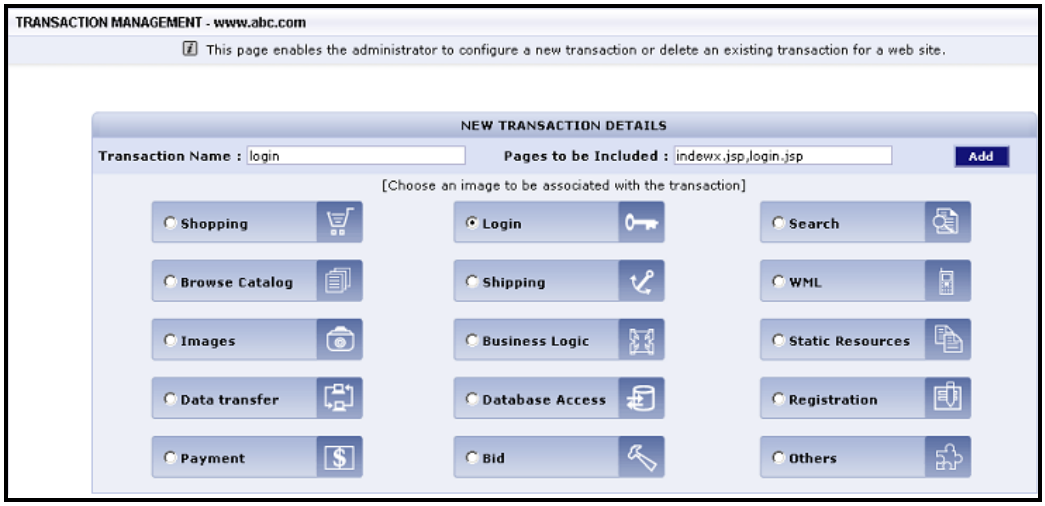

Figure 1.12: Configuring a new transaction

24. Here, specify the **Transaction Name** and the **Pages to be Included** (these are one or more regular expression patterns, where each pattern refers to a set of pages that constitute the transaction).

Example

The transactions of a retail web site could be: login, registration, browsing of the product catalog, searching the catalog, adding to shopping cart, deleting items from the cart, payment, etc. The **PAGES TO BE INCLUDED** for the **Login** transaction could be represented by ***/jsps/Loginform.jsp**.

While mentioning the **PAGES TO BE INCLUDED**, ensure the following:

- The page names should be prefixed by an * (asterisk) or a slash (/). If not, no measurements will be gathered from such pages.

25. You can also associate an image with a transaction, by choosing the same from the list below.
26. Then, to add the transaction, click on the **Add** button. Clicking on **Add** will take you back to the **NEW TRANSACTION DETAILS** page above, where you would be prompted to add another transaction.
27. Now, if you attempt to sign out, the **Processes** test information will flash on the screen, prompting you to configure the test.

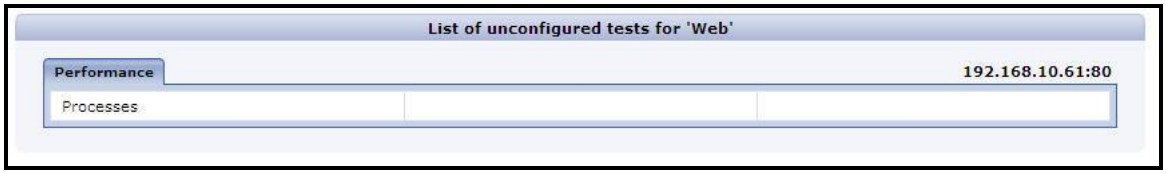

Figure 1.13: Viewing the tests configuration table displaying the Processes test information

28. Clicking on the test name will lead you to the following screen:

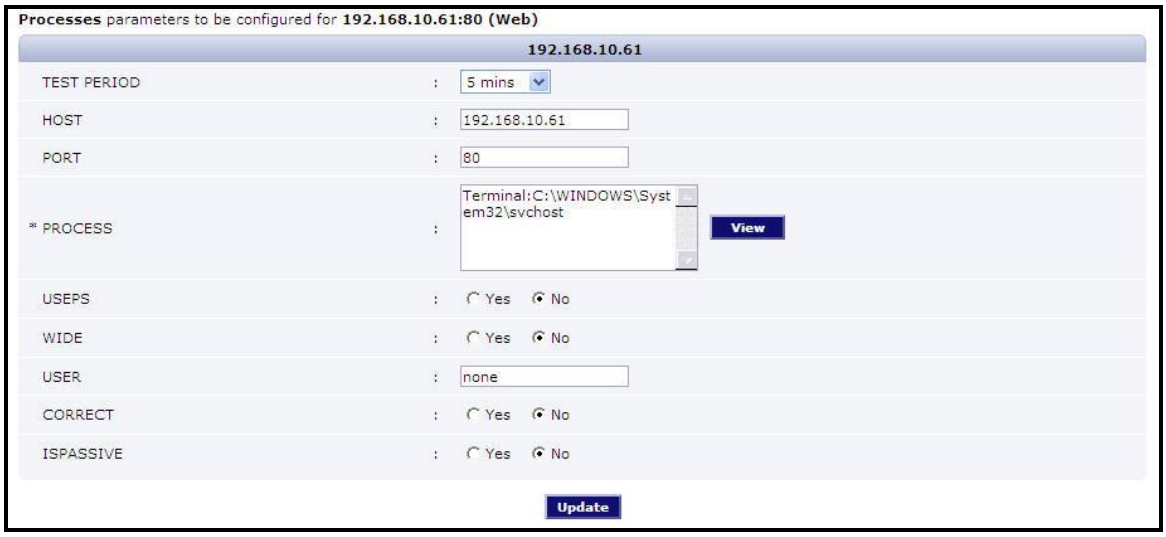

Figure 1.14: Viewing the Processes test configuration screen

29. In Figure 1.14, specify the following:
    - **TEST PERIOD** - How often should the test be executed
    - **HOST** - The host for which the test is to be configured
    - **PORT -** The port to which the specified **HOST** listens
    - **PROCESS** - In the **PROCESS** text box, enter a comma separated list of names:pattern pairs which identify the process(es) associated with the server being considered. processName is a string

that will be used for display purposes only. processPattern is an expression of the form - *expr* or expr or *expr or expr* or *expr1*expr2*... or expr1*expr2, etc. A leading '*' signifies any number of leading characters, while a trailing '*' signifies any number of trailing characters. The pattern(s) used vary from one application to another and must be configured per application. For example, for an iPlanet application server (Nas_server), there are three processes named kcs, kjs, and kxs associated with the application server. For this server type, in the **PROCESS** text box, enter "kcsProcess:*kcs*, kjsProcess:*kjs*, kxsProcess:*kxs*, where * denotes zero or more characters. Other special characters such as slashes (\) can also be used while defining the process pattern. For example, if a server's root directory is /home/egurkha/apache and the server executable named httpd exists in the bin directory, then, the process pattern is "*/home/egurkha/apache/bin/httpd*".

The **PROCESS** parameter supports process patterns containing the ~ character.

To determine the process pattern to use for your application, on Windows environments, look for the process name(s) in the Task Manager -> Processes selection. To determine the process pattern to use on Unix environments, use the ps command (e.g., the command "ps -e -o pid,args" can be used to determine the processes running on the target system; from this, choose the processes of interest to you.)

Also, while monitoring processes on Windows, if the **WIDE** parameter of this test is set to **true**, then your process patterns can include the full path to the process and/or the arguments supported by the process. For instance, your **PROCESSPATTERN** specification can be as follows:

Terminal:C:\WINDOWS\System32\svchost -k DcomLaunch,Remote:C:\WINDOWS\system32\svchost.exe -k netsvcs

Also, note that the **PROCESS** parameter is **case-sensitive** in **Unix environments**.

To save the time and effort involved in such manual process specification, eG Enterprise offers an easy-to-use auto-configure option in the form of a **View** button that is available next to the **PROCESS** text box.

- **USER** - By default, this parameter has a value "none"; this means that the test monitors all processes that match the configured patterns, regardless of the user executing them. If you want the test to monitor the processes for specific users alone, then, on Unix platforms, specify a comma-separated list of users to be monitored in the **USER** text box. For instance: *john,elvis,sydney*

  While monitoring Windows hosts on the other hand, your **USER** configuration should be a comma-separated list of "domain name-user name" pairs, where every pair is expressed in the following format: *Domainname\Username*. For example, to monitor the processes of user *john* and *elvis* who belong to domain *mas*, your **USER** specification should be: *mas\john,mas\elvis*. Also, on a Windows host, you will find system processes running on the following user accounts: *SYSTEM*, *LOCAL SERVICE*, and *NETWORK SERVICE*. While configuring these **USER** accounts, make

sure the *Domainame* is always *NT AUTHORITY*. In this case therefore, your **USER** specification will be: *NT AUTHORITY\SYSTEM,NT AUTHORITY\LOCAL SERVICE,NT AUTHORITY\NETWORK SERVICE*.

If multiple **PROCESS**es are configured for monitoring and multiple **USER**s are also configured, then the test will check whether the first process is run by the first user, the second process by the second user, and so on. For instance, if the **PROCESS**es configured are *java:java.exe,apache:*httpd** and the **USER**s configured are *john,elvis*, then the test will check whether user *john* is running the process *java*, and user *elvis* is running the process *apache*. Similarly, if multiple **PROCESS**es are configured, but a single **USER** alone is configured, then the test will check whether the specified **USER** runs each of the configured **PROCESS**es. However, if you want to check whether a single process, say *java.exe*, is run by multiple users - say, *james* and *jane* - then, you have to do the following:

- Your **USER** specification should be: *james,jane* (if the target host is a Unix host), or *<Domainname>\james,<Domainname>\jane* (if the target host is a Windows host)
- Your **PROCESS** configuration should be: *Process1:java.exe*,*Process2:java.exe*. The number of processes in this case should match the number of users.
- Such a configuration will ensure that the test checks for the *java.exe* process for both the users, *james* and *jane*.

- **CORRECT -** Increased uptime and lower mean time to repair are critical to ensuring that IT infrastructures deliver a high quality of service to users. Towards this end, the eG Enterprise suite embeds an optional auto-correction capability that enables eG agents to automatically correct problems in the environment, as soon as they occur. With this capability, as and when an abnormal situation is detected, an eG agent can initiate corrective actions automatically to resolve the problem. Automatic correction without the need for manual intervention by IT operations staff reduces service downtime and improves operational efficiency. By default, the auto-correction capability is available in the eG Enterprise suite for the Processes running measure of Processes test, and the Service availability measure of WindowsServices test. The eG Enterprise suite includes a default auto-correction script for Processes test.

  When a process that has been configured for monitoring stops, this script automatically executes and starts the process. To enable the auto-correction capability for the Processes test, first, select the **TRUE** option against the **CORRECT** parameter in this page (by default, **FALSE** will be selected here).

- **ALARMTYPE** - Upon selecting the **true** option, three new parameters, namely, **ALARMTYPE**, **USERPARAMS**, and **CORRECTIVESCRIPT** will appear. You can set the corrective script to execute when a specific type of alarm is generated, by selecting an option from the **ALARMTYPE** list box. For example, if the **Critical** option is chosen from the **ALARMTYPE** list box, then the corrective script will run only when a critical alarm for the Processes test is generated. Similarly, if the **Critical/Major** option is chosen, then the corrective script will execute only when the eG Enterprise system generates critical or major alarms for the Processes test. In order to ensure that the corrective script executes regardless of the alarm type, select the **Critical/Major/Minor** option.

- **USERPARAMS** - The user-defined parameters that are to be passed to the corrective script are specified in the **USERPARAMS** text box. One of the following formats can be applied to the **USERPARAMS** specification:

  exec@processName:command: In this specification, *processName* is the display name of the process pattern specified against the PROCESS parameter, and *command* is the command to be executed by the default script when the process(es) represented by the *processName* stops. For example, assume that the **PROCESS** parameter of Processes test has been configured in the following manner: *Apache:*/opt/egurkha/manager/apache/bin/httpd*,Tomcat:*java*tomcat**,

where *Apache* and *Tomcat* are the *processNames* or display names of the configured patterns. If auto-correction is enabled for these processes, then the **USERPARAMS** specification can be as follows:

exec@Apache:/opt/egurkha/manager/apache/bin/apachectl start,Tomcat: /opt/tomcat/bin/catalina.sh start

This indicates that if the processes configured under the *processName "Apache"* stop (i.e. **/opt/egurkha/manager/apache/bin/httpd**), then the script will automatically execute the command *"/opt/egurkha/manager/apache/bin/apachectl start"* to start the processes. Similarly, if the *"Tomcat"* processes (i.e. **java*tomcat**) stop, the script will execute the command *"/opt/tomcat/bin/catalina.sh start"* to start the processes.

command: In this specification, *command* signifies the command to be executed when any of the processes configured for monitoring, stop. Such a format best suits situations where only a single process has been configured for monitoring, or, a single command is capable of starting all the configured processes. For example, assume that the **PROCESS** parameter has been configured to monitor *IISWebSrv:*inetinfo**. Since only one process requires monitoring, the first format need not be used for configuring the **USERPARAMS**. Therefore, simplify specify the command, *"net start World Wide Web Publishing Service"*.

- The **USERPARAMS** specification should be placed within double quotes if this value includes one or more blank spaces (eg.,"*Apache:/opt/egurkha/bin/apachectl start*").
- Note that if a processName configured in the **PROCESS** parameter does not have a corresponding entry in **USERPARAMS** (as discussed in format 1), then the auto-correction capability will not be enabled for these processes.

- **CORRECTIVESCRIPT** - Specify *none* in the **CORRECTIVESCRIPT** text box to use the default auto-correction script. Administrators can build new auto-correction capabilities to address probable issues with other tests, by writing their own corrective scripts. To know how to create custom auto-correction scripts, refer to the *eG User Manual*.
- **WIDE** - **This parameter is valid on Solaris and Windows systems only**.

  On Solaris systems (before v11), if the value of the **WIDE** parameter is **Yes**, the eG agent will use *usr/ucb/ps* instead of */usr/bin/ps* to search for processes executing on the host. In Solaris 11, the eG agent uses the */usr/bin/ps auxwww* command to perform the process search. The */usr/ucb/ps* and the */usr/bin/ps auxwww* commands provide a long output (> 80 characters), whereas */usr/bin/ps* only outputs the first 80 characters of the process path and its arguments. However, some Solaris systems are configured with tightened security, which prevents the *usr/ucb/ps* and/or the */usr/bin/ps auxwww* command to be executed by any and every user to the system - in other words, only pre-designated users will be allowed to execute this command. The **sudo** (*superuser do*) utility (see http://www.gratisoft.us/sudo/) can be used to allow designated users to execute this command. If your system uses **sudo** to restrict access to the commands that return a long output, then set **WIDE** to **Yes** and then specify the value *sudo* for the **KEONIZEDSERVERCMD** parameter. This will ensure that not only does the agent use the */usr/ucb/ps* and/or the */usr/bin/ps auxwww* command (as the case may be) to monitor

processes (like it would do if the **WIDE** parameter were set to **Yes**), but it would also use **sudo** to execute this command.

If the *Processes* test on Solaris 11 fails, then do the following:

- Check whether the **WIDE** parameter is set to **Yes**.
- If so, then make sure that the **KEONIZEDSERVERCMD** parameter is set to **sudo**.
- If the test still fails, then look for the following error in the **error_log** file (that resides in the **/opt/egurkha/agent/logs** directory) on the eG agent host:

  ERROR ProcessTest: ProcessTest failed to execute [sudo: pam_authenticate: Conversation failure]

- The aforesaid error occurs if the *sudo* command prompts for a password at runtime. If you find such an error in the **error_log** file, then, open the **SUDOERS** file on the target host and append an entry of the following format to it:

  Defaults:<eG_Install_Username> !authenticate

  For instance, if *eguser* is the eG install user, then your entry will be: *Defaults:eguser !authenticate*

  This entry will make sure that you are no longer prompted for a password.

- Save the file and restart the eG agent.

On Windows environments, by default, the eG agent uses *perfmon* to search for the processes that match the configured patterns. Accordingly, the **WIDE** parameter is set to **No** by default. Typically, a process definition in Windows includes the *full path to the process*, the *process name*, and *process arguments* (if any). *Perfmon* however scans the system only for *process names* that match the configured patterns – in other words, the process path and arguments are ignored by *perfmon*. This implies that if multiple processes on a Windows host have the same name as specified against **PROCESSPATTERN**, then *perfmon* will only be able to report the overall resource usage across all these processes; it will not provide any pointers to the exact process that is eroding the host's resources. To understand this better, consider the following example. Typically, Windows represents any Java application executing on it as *java.exe*. Say, two Java applications are executing on a Windows host, but from different locations.

If *java.exe* has been configured for monitoring, then by default, *perfmon* will report the availability and average resource usage of both the Java applications executing on the host. If say, one Java application goes down, then *perfmon* will not be able to indicate accurately which of the two Java applications is currently inaccessible. Therefore, to enable administrators to easily differentiate between processes with the same name, and to accurately determine which process is unavailable or resource-hungry, the eG agent should be configured to perform its process searches based on the process path and/or process arguments, and not just on the process name – in other words, the eG agent should be configured **not to use perfmon**.

To achieve this, first, set the **WIDE** parameter to **Yes**. This will instruct the eG agent to not use

perfmon to search for the configured process patterns. Once this is done, then, you can proceed to configure a **PROCESSPATTERN** that includes the *process arguments* and/or the *process* path, in addition to the *process* name. For instance, if both the *Remote Access Connection Manager* service and the *Terminal Services* service on a Windows host, which share the same name – *svchost* - are to be monitored as two different processes, then your **PROCESSPATTERN** specification should be as follows:

Terminal:C:\WINDOWS\System32\svchost -k DcomLaunch,Remote:C:\WINDOWS\system32\svchost.exe -k netsvcs

You can also use wildcard characters, wherever required. For instance, in the above case, your **PROCESSPATTERN** can also be:

*Terminal:*svchost -k DcomLaunch,Remote:*svchost.exe -k netsvcs*

Similarly, to distinctly monitor two processes having the same name, but operating from different locations, your specification can be:

JavaC:c:\javaapp\java.exe,JavaD:d:\app\java.exe

- Before including process paths and/or arguments in your **PROCESSPATTERN** configuration, make sure that the **WIDE** parameter is set to **true**. If not, the test will not work.
- If your **PROCESSPATTERN** configuration includes a process path that refers to the *Program Files* directory, then make sure that you **do not a include a ~** (tilde) while specifying this directory name. For instance, your **PROCESSPATTERN** specification should not be say, *Adobe:C:\Progra~1\Adobe\AcroRd32.exe*.

- **KEONIZEDSERVERCMD** - On Solaris hosts, this test takes an additional **KEONIZEDSERVERCMD** parameter. Keon is a security mechanism that can be used with a multitude of operating systems to provide a centralized base for user account and password management, user access and inactivity control, system integrity checking, and auditing. If the Keon security model is in use on the Solaris host being monitored, then this test may require special user privileges for executing the operating system commands. In such a case, specify the exact command that the test is permitted to execute, in the **KEONIZEDSERVERCMD** text box. For example, if the keon command to be executed by the test is *sudo*, specify *sudo* in the **KEONIZEDSERVERCMD** text box. Alternatively, you can even specify the full path to the *sudo* command in the KEONIZEDSERVERCMD text box. On the other hand, if a Keon security model is not in place, then set the KEONIZEDSERVERCMD parameter to *none*.
- **USEGLANCE** - **This flag applies only to HP-UX systems**. HP GlancePlus/UX is Hewlett-Packards's online performance monitoring and diagnostic utility for HP-UX based computers. There are two user interfaces of GlancePlus/UX -- *Glance* is character-based, and *gpm* is motif-based. Each contains graphical and tabular displays that depict how primary system resources are being utilized. In environments where *Glance* is run, the eG agent can be configured to integrate with *Glance* to pull out the process status and resource usage metrics from the HP-UX systems that are being monitored. By default, this integration is disabled. This is why the **USEGLANCE** flag is set to **No** by default. You can enable the integration by setting the flag to **Yes**. If this is done, then the

test polls the *Glance* interface of HP GlancePlus/UX utility to pull out the desired metrics.

- **USEPS** - **This flag is applicable only for AIX LPARs.** By default, on AIX LPARs, this test uses the **tprof** command to compute CPU usage of the processes on the LPARs. Accordingly, the **USEPS** flag is set to **No** by default. On some AIX LPARs however, the **tprof** command may not function properly (this is an AIX issue). While monitoring such AIX LPARs therefore, you can configure the test to use the **ps** command instead for metrics collection. To do so, set the **USEPS** flag to **Yes**.

Alternatively, you can set the **AIXusePS** flag in the **[AGENT_SETTINGS]** section of the **eg_tests.ini** file (in the **<EG_INSTALL_DIR>\manager\config** directory) to **yes** (default: **no**) to enable the eG agent to use the **ps** command for CPU usage computations on AIX LPARs. If this global flag and the **USEPS** flag for a specific component are both set to **no**, then the test will use the default **tprof** command to compute CPU usage of processes executing on AIX LPARs. If either of these flags is set to **yes**, then the **ps** command will perform the CPU usage computations for such processes.

In some high-security environments, the **tprof** command may require some special privileges to execute on an AIX LPAR (eg., *sudo* may need to be used to run **tprof**). In such cases, you can prefix the **tprof** command with another command (like *sudo*) or the full path to a script that grants the required privileges to **tprof**. To achieve this, edit the **eg_tests.ini** file (in the **<EG_INSTALL_DIR>\manager\config** directory), and provide the prefix of your choice against the **AixTprofPrefix** parameter in the **[AGENT_SETTINGS]** section. Finally, save the file. For instance, if you set the **AixTprofPrefix** parameter to *sudo*, then the eG agent will call the **tprof** command as *sudo tprof*.

- **ISPASSIVE** – If the value chosen is **YES**, then the server under consideration is a passive server in a cluster. No alerts will be generated if the server is not running. Measures will be reported as "Not applicable' by the agent if the server is not up.

30. Upon clicking the **Update** button, the following screen will appear:

Figure 1.15: Viewing a summary of the Processes test parameters

31. The HTTP test emulates a user accessing a web server. Since this test can be executed from a location external to the web server, it presents an unbiased external perspective of the state of the web server. Though this test, by default, does not require any configuration, it offers some specialized parameters, which if configured, generate certain interesting performance statistics pertaining to the web server.

32. To configure the parameters of this test, first open the **AGENTS – TESTS SPECIFIC CONFIGURATION** page by following the menu sequence: Agents -> Tests -> Configure ->Specific.

    First choose the type of component (from the **Component type** list box) and the specific component (from the **Component name** list box). Then choose type of a test (from the **Test type** list box). Doing so will provide the agent summary details and as well the configuration status of all the tests pertaining to the chosen component respectively.

    In this page, you will view the configuration status of all the tests to the chosen component in a broad spectrum of classification such as **UNCONFIGURED TESTS, CONFIGURED TESTS** and **EXCLUDED TESTS**.

    Configured tests are displayed in two types of categories namely,

    - **Tests with specific configuration**

      (Tests that explicitely require user intervention for its execution).

    - **Tests with default configuration**

    (Test that are ready to be executed by its default settings).

    Using this interface, you can do the following:

- Configure an unconfigured test
- Reconfigure a test from the **Tests with specific configuration** category.
- Reconfigure a test from the **Tests with default configuration** category.
- Exclude/Include a test from its execution

To reconfigure a configured test, select the test from the **CONFIGURED TESTS** list box and click on the **Reconfigure** button.

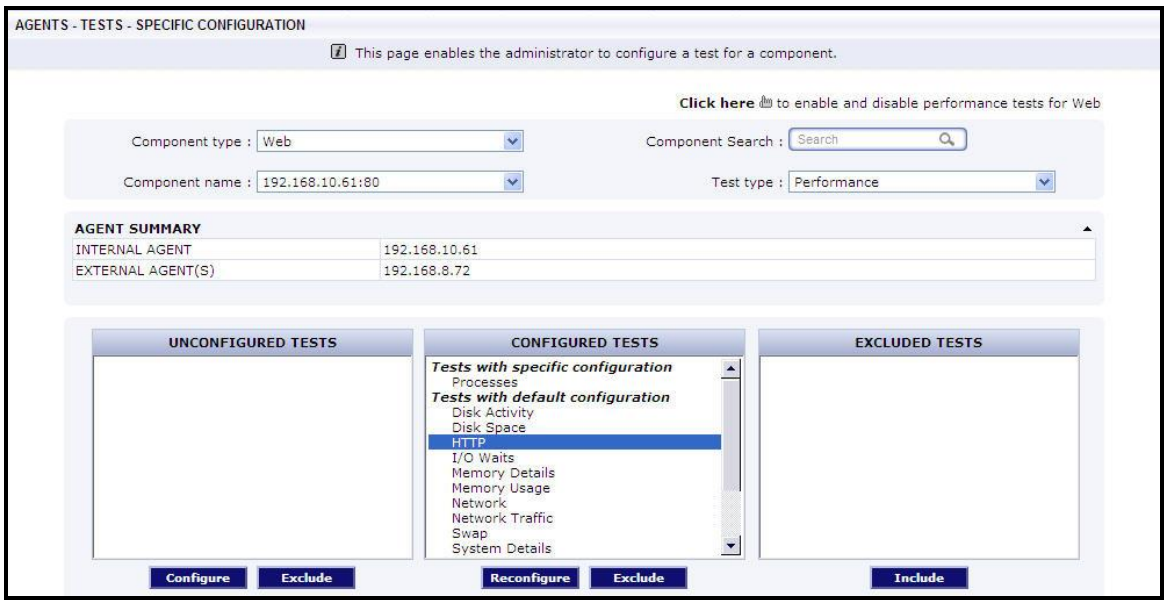

Figure 1.16: Reconfiguring Http test

This will invoke the parameters to be configured for the chosen test. Finally click on the **Update** button to implement the changes as depicted by the Figure 1.16.

33. The test configuration page reveals the following parameters (see Figure 1.17):

- **TEST PERIOD** – how often should the test be executed
- **URL** – The web page being accessed. While multiple URLs (separated by commas) can be provided, each URL should be of the format **URL name:URL value**. **URL name** is a unique name assigned to the URL, and the **URL value** is the value of the URL. For example, a URL can be specified as **HomePage:http://192.168.10.12:7077/**, where **HomePage** is the **URL name** and http://192.168.10.12:7077/ is the **URL value**.
- **HOST** - The host for which the test is to be configured.
- **PORT** - The port number on which the specified **HOST** listens
- **COOKIEFILE** – Whether any cookies being returned by the web server need to be saved locally and returned with subsequent requests
- **PROXYHOST** – The host on which a web proxy server is running (in case a proxy server is to be used)
- **PROXYPORT** – The port number on which the web proxy server is listening
- **PROXYUSERNAME** – The user name of the proxy server
- **PROXYPASSWORD** – The password of the proxy server
- **CONFIRM PASSWORD** – Confirm the **PROXYPASSWORD** by retyping it here.
- **CONTENT –** is a set of instruction:value pairs that are used to validate the content being returned by the test. If the **CONTENT** value is *none:none*, no validation is performed. The number of pairs specified in this text box, must be equal to the number of URLs being monitored. The instruction should be one of *Inc* or *Exc*. *Inc* tells the test that for the content returned by the web server to be valid, the content must include the specified value (a simple string search is done in this case). An instruction of *Exc* instructs the test that the server's output is valid if it does not contain the specified value. In both cases, the content specification can include wild card patterns. For

example, an Inc instruction can be *Inc:*Home page**. An Inc and an Exc instruction can be provided in quick succession in the following format: *Inc:*Home Page*,Exc:*home*

- **CREDENTIALS** –The HttpTest supports HTTP authentication. The **CREDENTIALS** parameter is to be set if a specific user name / password has to be specified to login to a page. Against this parameter, the **URLname** of every configured **URL** will be displayed; corresponding to each listed **URLname**, a **Username** text box and a **Password** text box will be made available. If the web server on which HttpTest executes supports 'Anonymous user access', then this parameter will take either of the following values:
    - a valid **Username** and **Password** for every configured **URLname**
    - *none* in both the **Username** and **Password** text boxes of all configured **URLnames** (the default setting), if no user authorization is required

  Some IIS web servers however, support NTLM (Integrated Windows) authentication, where valid **CREDENTIALS** are mandatory. In other words, a *none:none* specification will not be supported by such IIS web servers. Therefore, in this case, against each configured **URLname**, you will have to provide a valid **Username** in the format: *domainname|username,* followed by a valid **Password**.

  Please be sure to check if your web site requires HTTP authentication while configuring this parameter. HTTP authentication typically involves a separate pop-up window when you try to access the page. Many sites use HTTP POST for obtaining the user name and password and validating the user login. In such cases, the username and password have to be provided as part of the POST information and NOT as part of the **CREDENTIALS** specification for the HTTP test.

  Moreover, SSL-enabled web sites are typically secured by a private key, public key, or a public-private key pair. If the web page configured for this test is SSL-enabled – i.e., if an HTTPS URL is specified against **URL** – and the contents of this web page can only be accessed using a **private key**, then the full path to the private key file will have to be provided against **Private key file path** and the password of the private key file should be specified against **Password**. If no such private key protects the contents of the configured **URL**, then set the **Private key file path** and its **Password** to *none*.

- **TIMEOUT** - Here, specify the maximum duration (in seconds) for which the test will wait for a response from the server. The default **TIMEOUT** period is 30 seconds.
- **ENCODING** - Sometimes the eG agent has to parse the **URL** content with specific encoding other than the default (ISO-8859-1) encoding. In such a case, specify the type of encoding using which the eG agent can parse the **URL** content in the **ENCODING** text box. By default, this value is *none*.

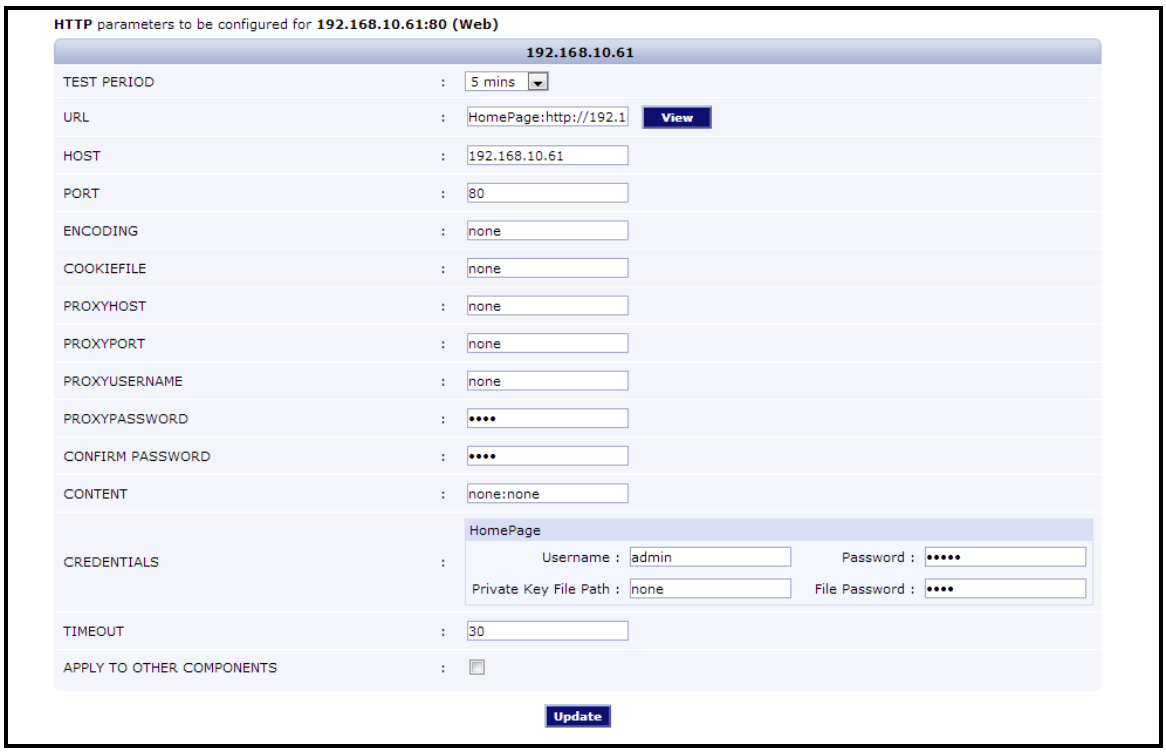

Figure 1.17: Configuring the HTTP test

34. Now, all your web server related tests are configured and the target server is ready to report measures to the eG server.

1.4 Monitoring the Apache/IBM HTTP Web Server

The next step is to log in to eG Enterprise's monitor interface to check the measures reported about the Apache/IBM HTTP web server. For this, login to the eG monitor interface as a supermonitor/monitor user, click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu. By default, a managed web server will show up as an independent server in the eG Enterprise's monitor interface.

To view the measurements pertaining to your Apache/IBM HTTP web server, just click on it. On clicking, a screen displaying the various layers that constitute the Apache/IBM HTTP web server will appear .

Similarly, from the **SERVICES HEALTH** page that appears on clicking the **Services** menu option, web sites, transactions and other services can be monitored.

1.5 Troubleshooting

If the Apache server tests are in an **UNKNOWN** state, then proceed to check whether the web adapter has been configured properly. While configuring an Apache server, setup will request for the full path to the root directory of the server. Ensure that this path is the same as the value displayed against the **ServerRoot** parameter in the **httpd.conf** file in the **<APACHE_SERVER_HOME >\conf** directory (see Figure 1.18).

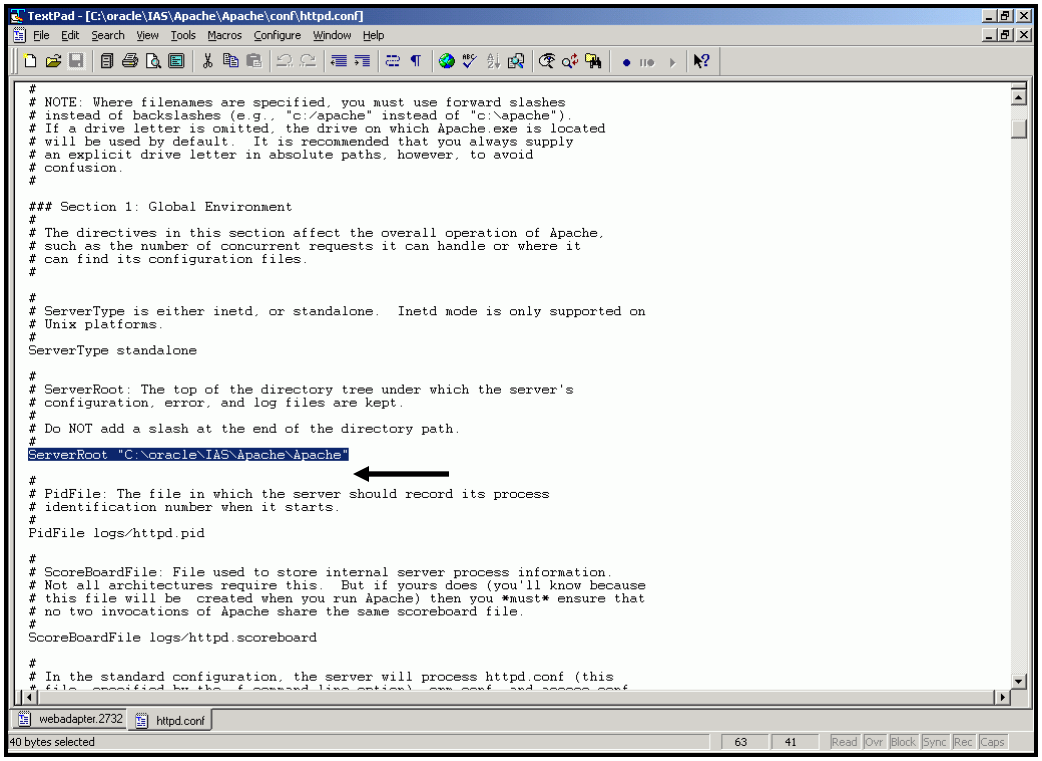

Figure 1.18: The ServerRoot parameter in the httpd.conf file

Next, check whether a file named **webadapter.<PID>** is created in the **<EG_HOME_DIR>\agent\config** directory. This is a clear indicator of the successful deployment of the web adapter. Now, verify whether the **PID** in **webadapter.<PID>** matches with the **PID** of any one of the **Apache.exe** processes in the **Windows Task Manager** (see Figure 1.19). If it does not match, then the web adapter may not work. Under such circumstances, delete the **webadapter.<PID>** file and restart the Oracle Http Server. Sometimes, an additional **webadapter** file will be created with a PID that does not match any of the **Apache.exe** PIDs listed in the **Windows Task Manager**. In such a case, delete the additional **webadapter.<PID>** file and restart the eG agent.

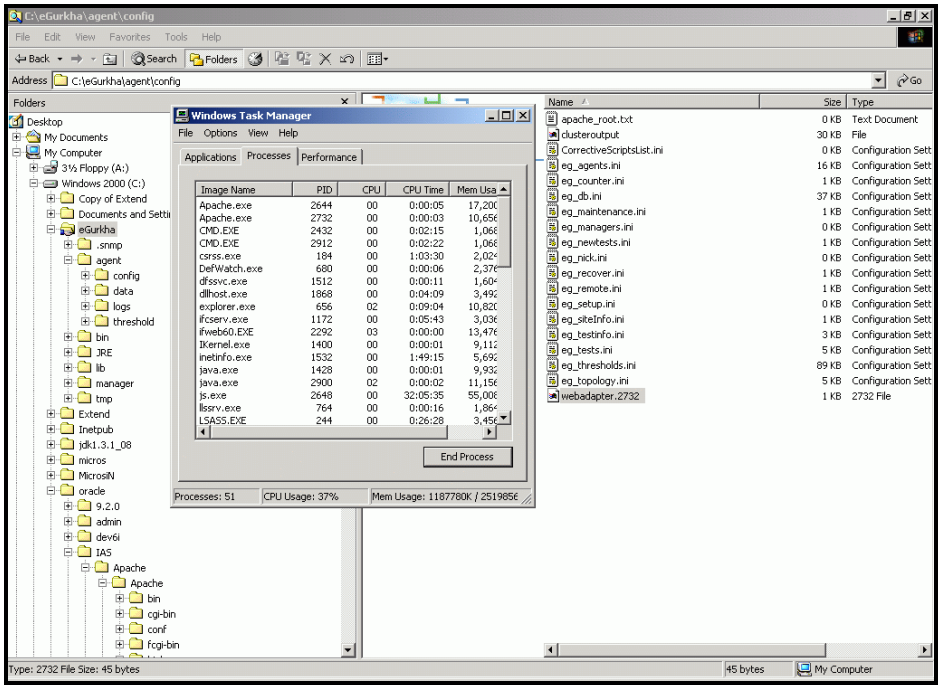

Figure 1.19: PID in the file name matching with the PID of one of the Apache.exe processes

Also, ensure that the **Listen** ports configured in the **webadpater.<PID>** file (see Figure 1.20) are the same as those which are listed in the **httpd.conf** file in the **<APACHE_SERVER_HOME>\conf** directory (see Figure 1.20).

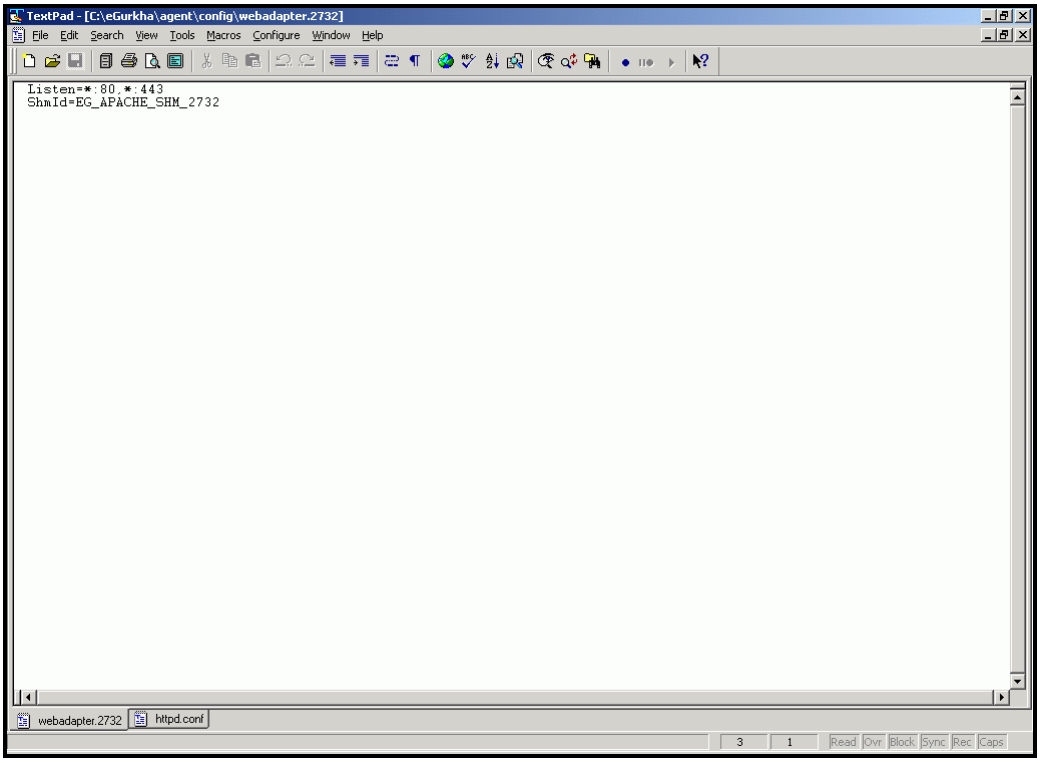

Figure 1.20: Listen ports displayed in the webadpater.<PID> file

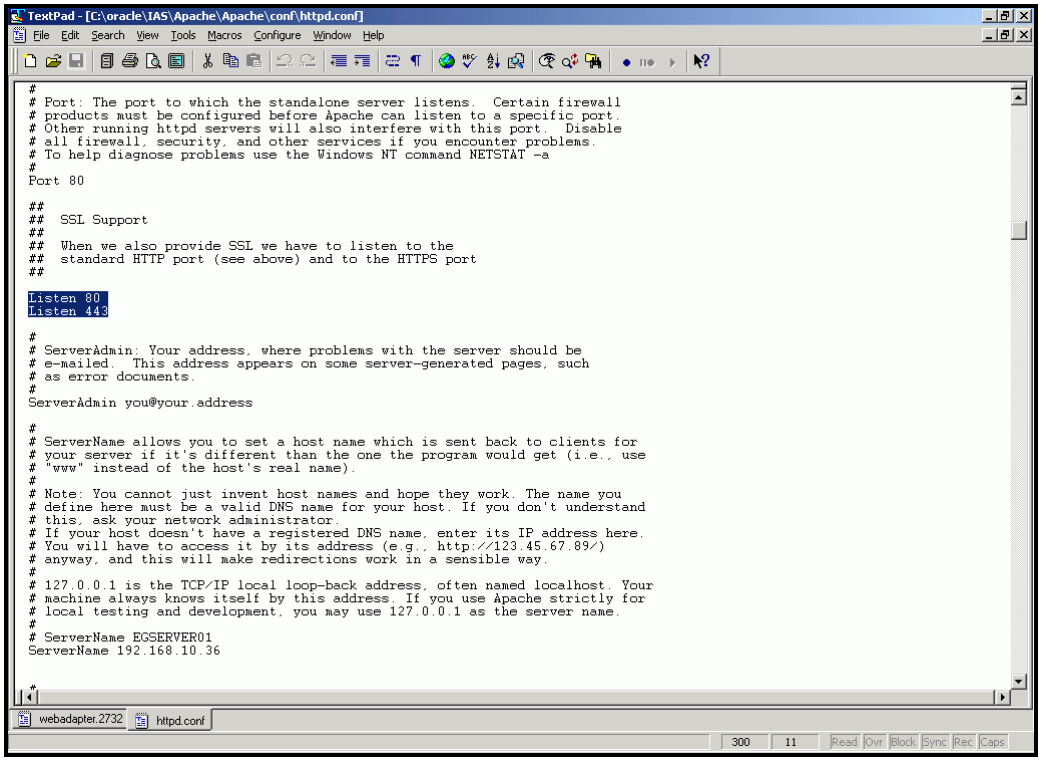

Figure 1.21: Listen ports displayed in the httpd.conf file

Note that the **Listen** ports displayed in the **webadapter.<PID>** file are prefixed by a '*', which typically represents an IP address. However, if a specific IP address substitutes the '*' in the **webadapter.<PID>** file, then, in the eG administrative interface, the Oracle Http server must be managed using that IP address only.

Finally, check whether the directives indicated by Figure 1.22 exist in the **httpd.conf** file in the **<APACHE_SERVER_HOME>\conf** directory.

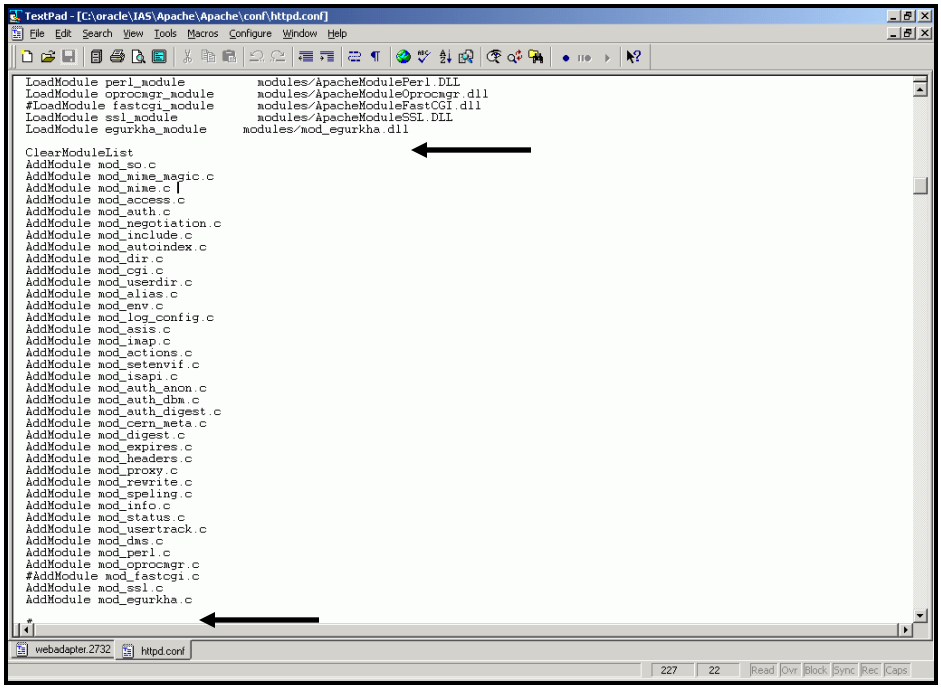

Figure 1.22: eG-specific directives in the httpd.conf file

Chapter 2

Configuring and Monitoring an IIS Web Server

The eG agent is capable of monitoring IIS web servers (ver. 4, 5, 6, and 7) in an agent-based and an agentless manner; however, note that, in the agentless mode, the solution cannot monitor web transactions to web sites configured on the target IIS web server.

To enable eG to monitor an IIS web server, the following activities need to be performed:

- Configuring an IIS web server to work with the eG agent
- Administering the eG manager through its user interface to monitor the IIS web server

2.1 Configuring an IIS Web Server to work with the eG Agent on Windows Environments

2.1.1 Configuring the eG Agent to Monitor Web Transactions to Web Sites on an IIS Web Server Operating on Windows 2000/2003

To enable the eG agent to monitor an IIS web server on Windows 2000/2003, follow the steps below:

- First, make sure 'logging' is enabled on these platforms;
- Next, make sure that the eG agent configuration is modified to support web transaction monitoring.

The sub-sections that follow will discuss each of these steps elaborately.

2.1.1.1 Enabling Logging on the IIS Web Server

Logging triggers the creation of log files that track the URLs accessed on the IIS web server. The eG Enterprise suite requires these log files for monitoring the transactions to the web sites hosted on the IIS web server. In the absence of these log files, an eG agent will not be able to monitor web site transactions for Microsoft IIS web servers. Therefore, in order to enable the eG Enterprise suite to perform effective web transaction monitoring, logging must be enabled for the managed web sites.

2.1.1.1.1 Enabling Logging for Web Sites on Windows 2003

In the case of an IIS web server on Windows 2003, logging can be enabled using the procedure discussed below:

1. Open the **Internet Information Services (IIS) Manager** on the IIS web server host using the menu sequence Start -> Programs -> Administrative Tools -> Internet Information Services (IIS) Manager. Figure 2.1 will then appear.

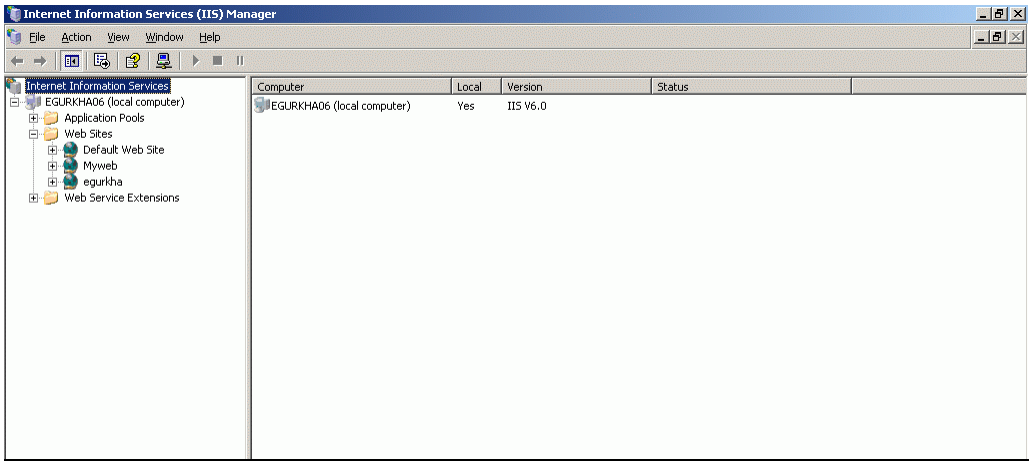

Figure 2.1: The IIS console

2. If all the web sites on the IIS web server are being monitored by eG, then you will have to enable logging for all. To achieve this, right-click on the **Web Sites** node in the tree structure on the left pane of Figure 2.1, and select **Properties** (see Figure 2.2) from the shortcut menu that appears.

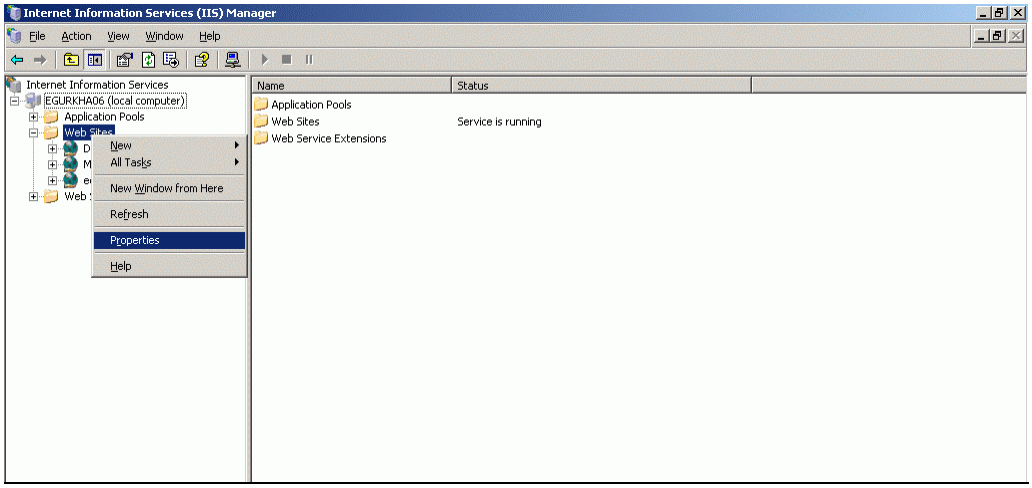

Figure 2.2: Selecting the Properties option from the shortcut menu of the Web Sites node (Windows 2003)

3. Next, click the **Web Site** tab of the **Properties** dialog box (see Figure 2.3) that appears, and ensure that the **Enable logging** check box is selected.

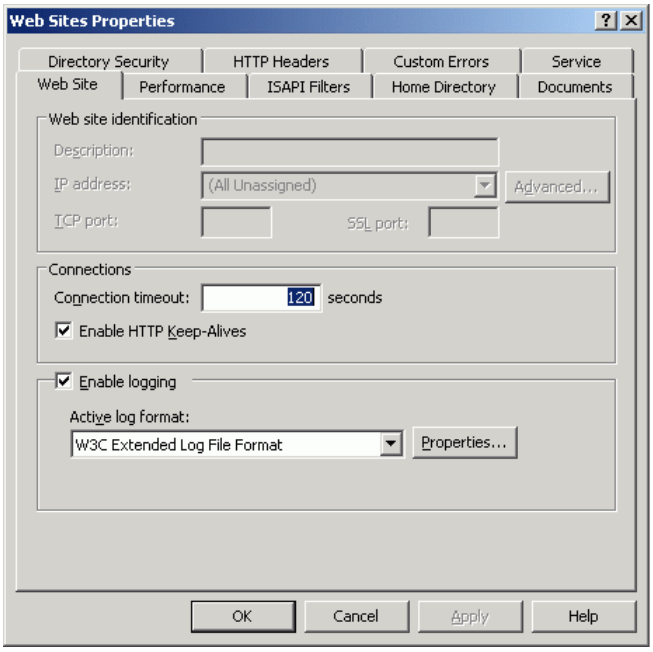

Figure 2.3: Enabling logging for all the web sites

4. Finally, click on the **Apply** and **OK** buttons to register the changes.

5. If only a few selected web sites on the IIS web server are being monitored by the eG Enterprise suite, then logging needs to be enabled for those specific sites only. To achieve this, right-click on the web site being monitored from the tree-structure in the left pane the IIS Manager, and select **Properties** from the shortcut menu (see Figure 2.4).

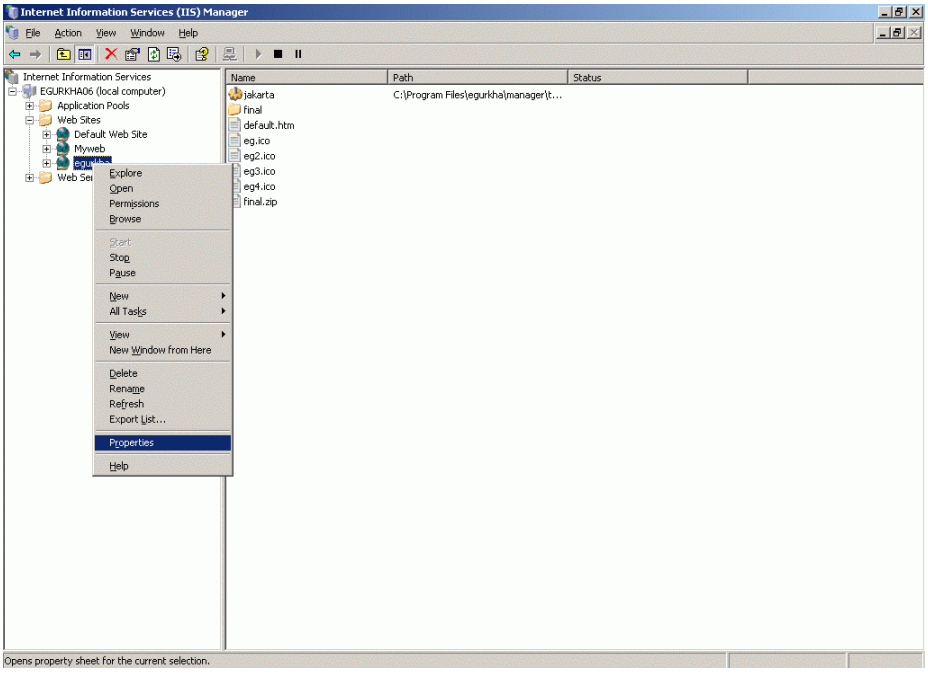

Figure 2.4: Selecting the Properties option for the egurkha web site

6. Next, select the **Web Site** tab from the **Properties** dialog box, and select the **Enable Logging** check box

as depicted by Figure 2.5. Finally, click on the **Apply** button and then the **OK** button.

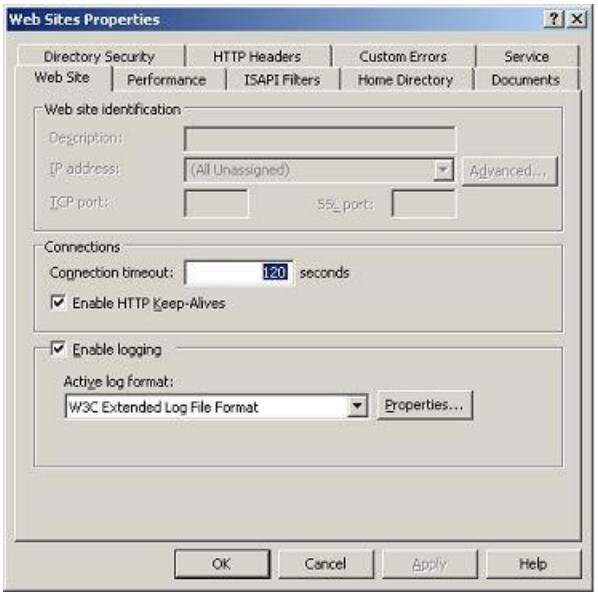

Figure 2.5: Enabling access logging for the egurkha web site

2.1.1.1.2 Enabling Logging for Web Sites on Windows 2000

In the case of an IIS web server on a Windows 2000 host, follow the steps below to enable logging for the web sites.

1. Open the **Internet Information Services** console on the IIS host using the menu sequence Start -> Programs -> Administrative Tools -> Internet Services Manager.
2. If all the web sites on the IIS web server are being monitored by eG, you will have to enable logging for all. To achieve this, right-click on the node representing the IIS host in the tree structure in the left pane of the IIS console (see Figure 2.6), and select **Properties** from the shortcut menu that appears.

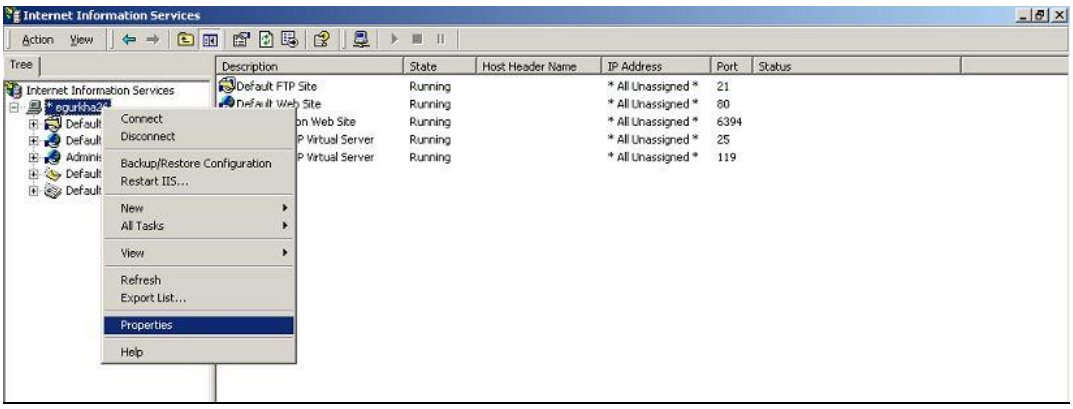

Figure 2.6: Selecting the Properties option from the shortcut menu of the IIS host node

3. Click on the **Edit** button in Figure 2.7 to modify the **Properties** of the IIS web server.

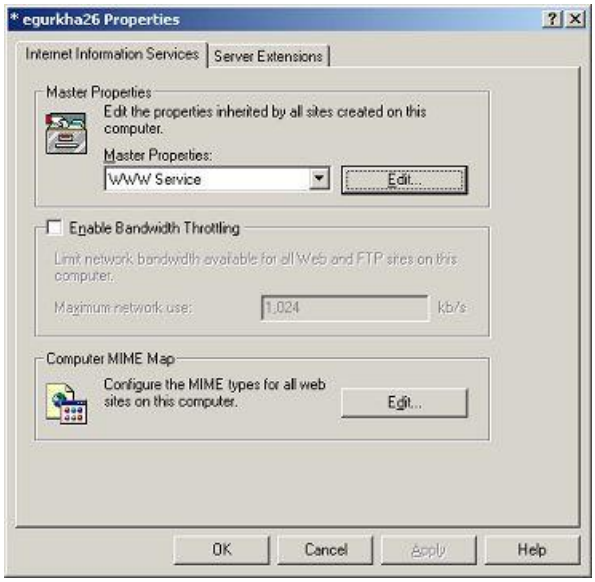

Figure 2.7: Clicking on the Edit button

4. Next, click the **Web Site** tab of the **Properties** dialog box (see Figure 2.8) that appears, and ensure that the **Enable Logging** check box is selected.

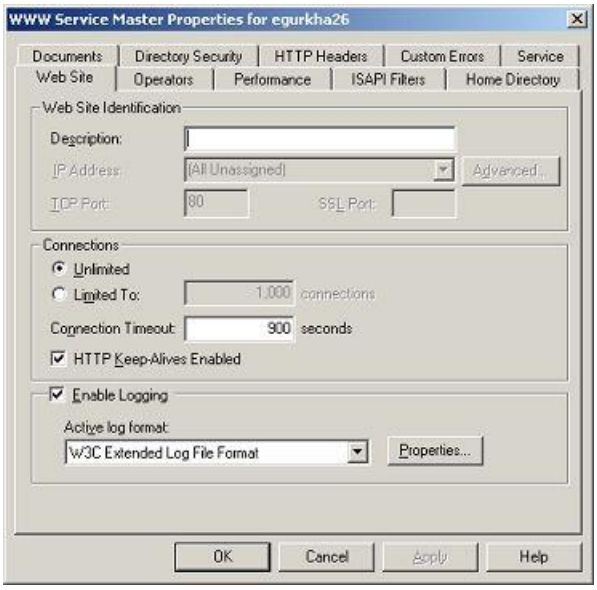

Figure 2.8: Selecting the 'Enable Logging' checkbox

5. Finally, click on the **Apply** and **OK** buttons to register the changes.

6. If only a few selected web sites on the IIS web server are being monitored by the eG Enterprise suite, then logging needs to be enabled for those specific sites only. To achieve this, right-click on the web site being monitored from the tree-structure in the left pane of the IIS console, and select **Properties** from the shortcut menu (see Figure 2.9).

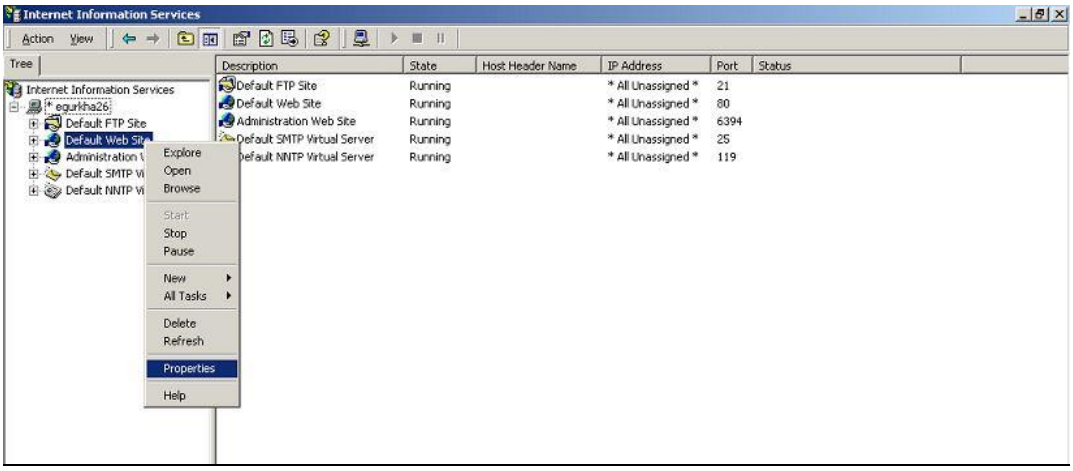

Figure 2.9: Selecting the Properties option for the Default web site

7. Next, select the **Web Site** tab from the **Properties** dialog box, and select the **Enable Logging** check box as depicted by Figure 2.10. Finally, click on the **Apply** button and then the **OK** button.

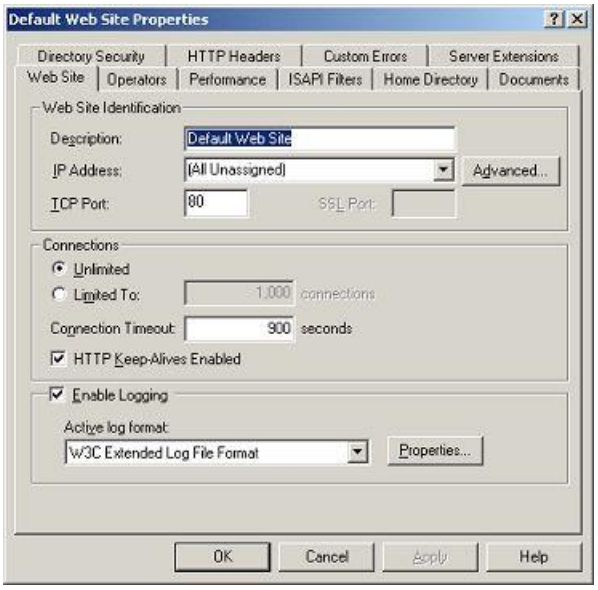

Figure 2.10: Enabling access logging for the Default web site

2.1.1.2 Modifying the eG Agent Configuration to Enable Web Transaction Monitoring

In order to monitor the web transactions to the web sites on an IIS web server, a specific filter needs to be installed on the IIS web server to track all requests to and from the web server. To achieve this, the eG agent on the IIS web server has to be modified. To do so, perform the steps given below:

1. Select **Uninstall Agent** from the options available under the eG Monitoring Suite -> eG Agent menu. The screen depicted by Figure 2.11 will appear. Here, select the **Modify** option and click the **Next >** button.

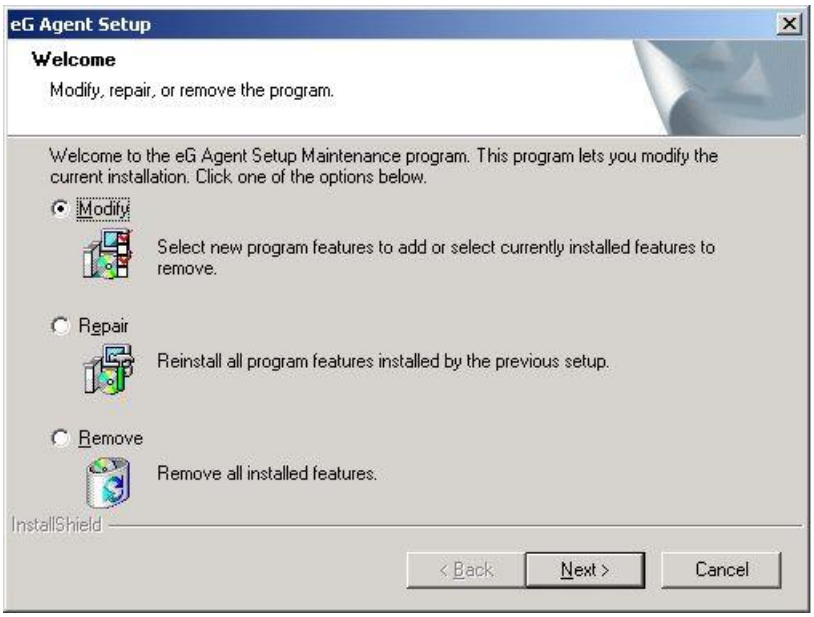

Figure 2.11: Modifying the agent configuration

2. If the eG agent setup program identifies an IIS server in the user environment, it expects the user to state if he/she wants to monitor this IIS server as depicted by Figure 2.12. If the user chooses **Yes**, the Setup installs a specific filter that will be used to track all requests to and from a web server. The default option is **No.**

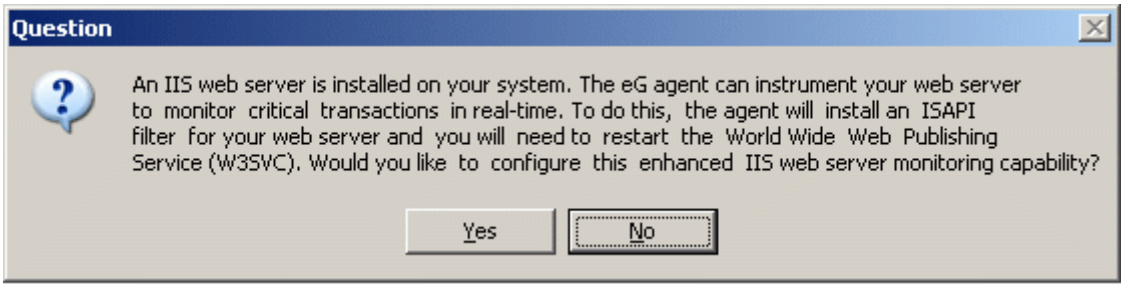

Figure 2.12: IIS web server monitoring

3. Then, restart the **World Wide Web Publishing Service**.

2.1.2 Configuring the eG Agent to Monitor an IIS Web Server Operating on Windows 2008

The eG agent can monitor an IIS web server operating on Windows 2008 only if the **Web Server** role is configured on the target Windows 2008 server.

Typically, for an IIS web server to function on a Windows 2008 server, a **Web Server Role** should be configured on the server. The **Web Server** role in Windows Server® 2008 lets you share information with users on the Internet, an intranet, or an extranet. If such a role does not exist on a Windows 2008 server, then, you cannot monitor the transactions to the IIS web server on that host; this is because, the ISAPI filter required for transaction monitoring cannot be installed on a Windows 2008 server without the **Web Server** role.

To configure this **Web Server** role on a Windows 2008 server, follow the steps detailed below:

1. Login to the Windows 2008 server as a local/domain administrator.
2. Open the **Server Manager** console by following the menu sequence, Start -> Programs -> Administrative Tools -> Server Manager (see Figure 2.13).

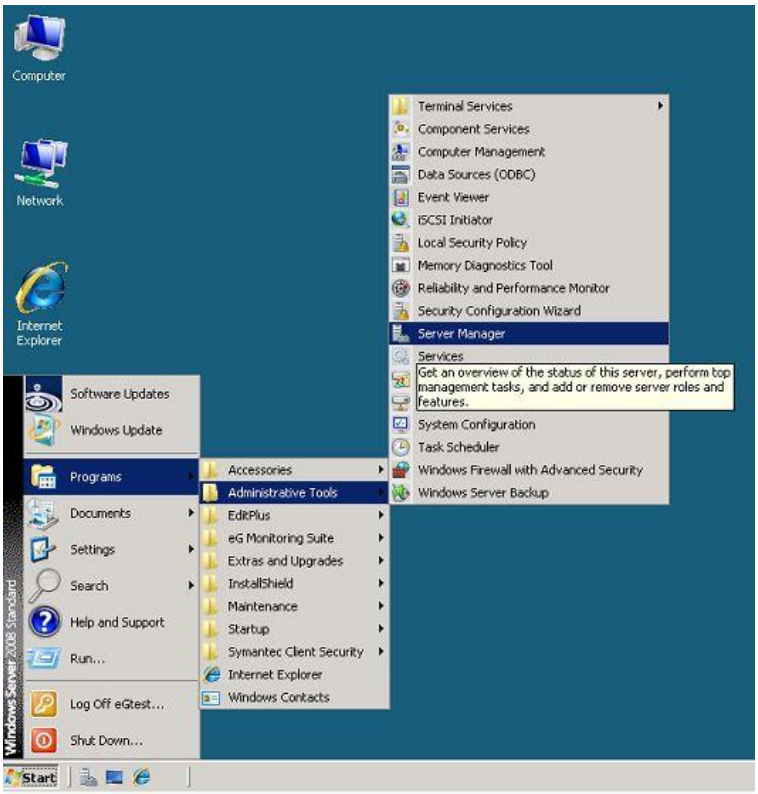

Figure 2.13: Opening the Server Manager

3. The **Server Manager** console then appears (see Figure 2.14).

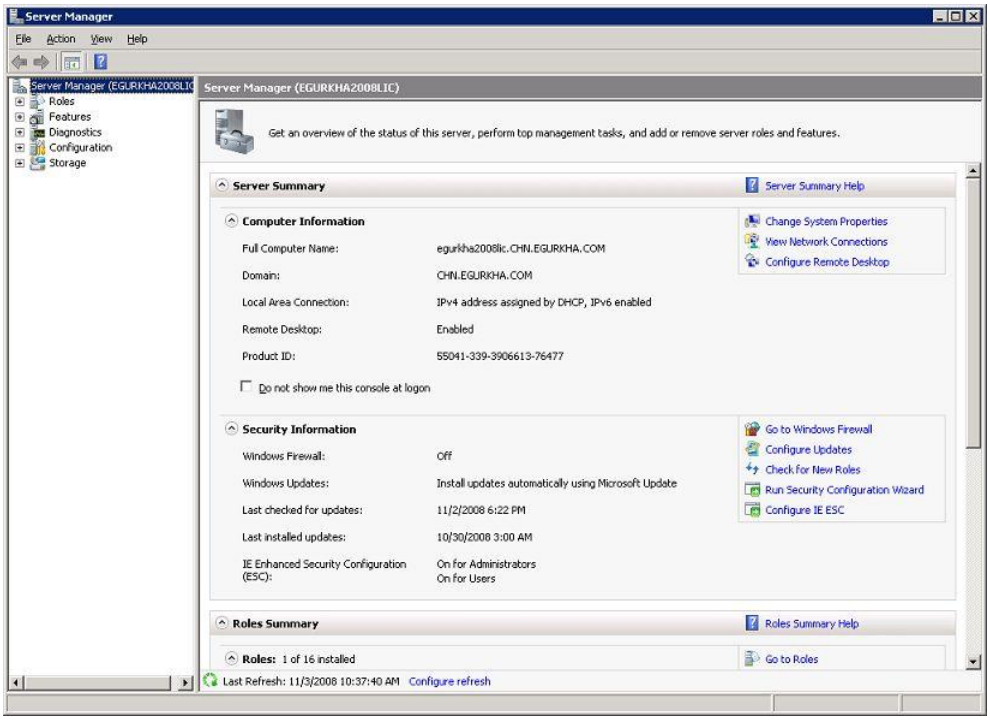

Figure 2.14: The Server Manager console

4. In the **Server Manager** console, click on the **Roles** node in the tree-structure in the left panel of the console. The information in the right-panel will change to display a **Roles Summary** and related details. To add a new role, click on the **Add Roles** option in the right panel of Figure 2.15.

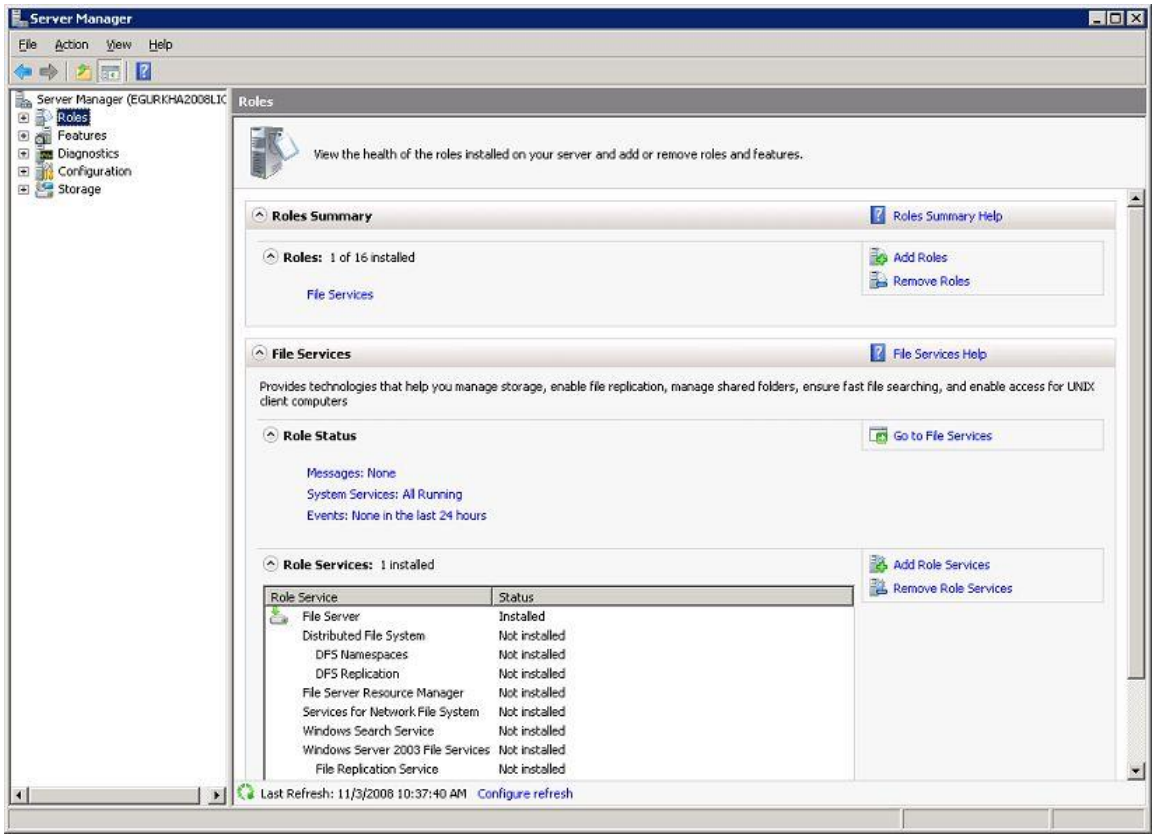

Figure 2.15: Clicking on the Roles node in the tree-structure

5. This will invoke the **Add Roles Wizard**. Click on the **Next** button in the welcome screen of Figure 2.16 to proceed with the role creation.

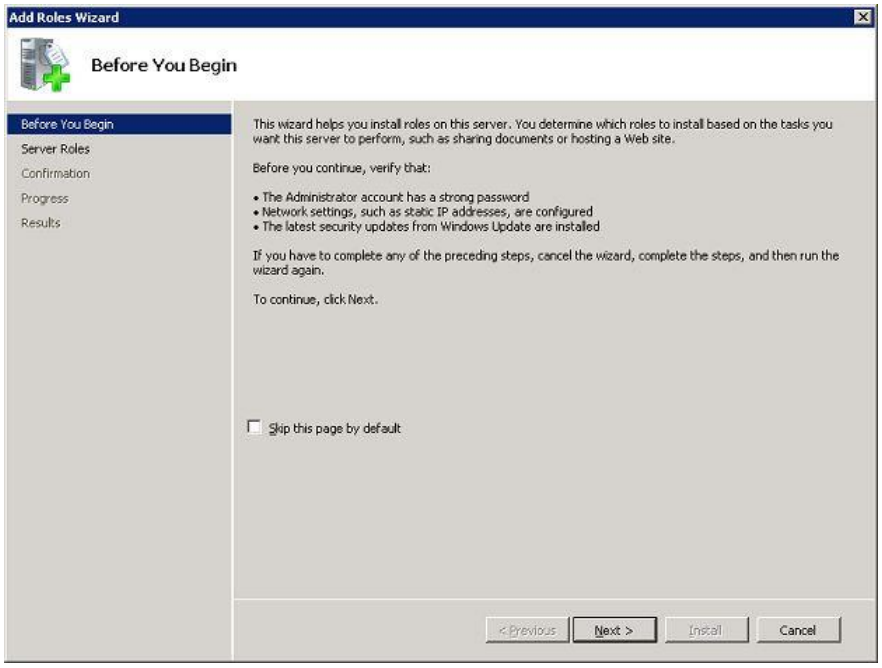

Figure 2.16: Clicking on the Next button in the welcome screen of the Add Roles Wizard

6. The next step of the wizard prompts you to pick one/more roles to install on the Windows 2008 server. Select the **Web Server (IIS)** role depicted by Figure 2.17 to install it. Then, click the **Next** button to proceed.

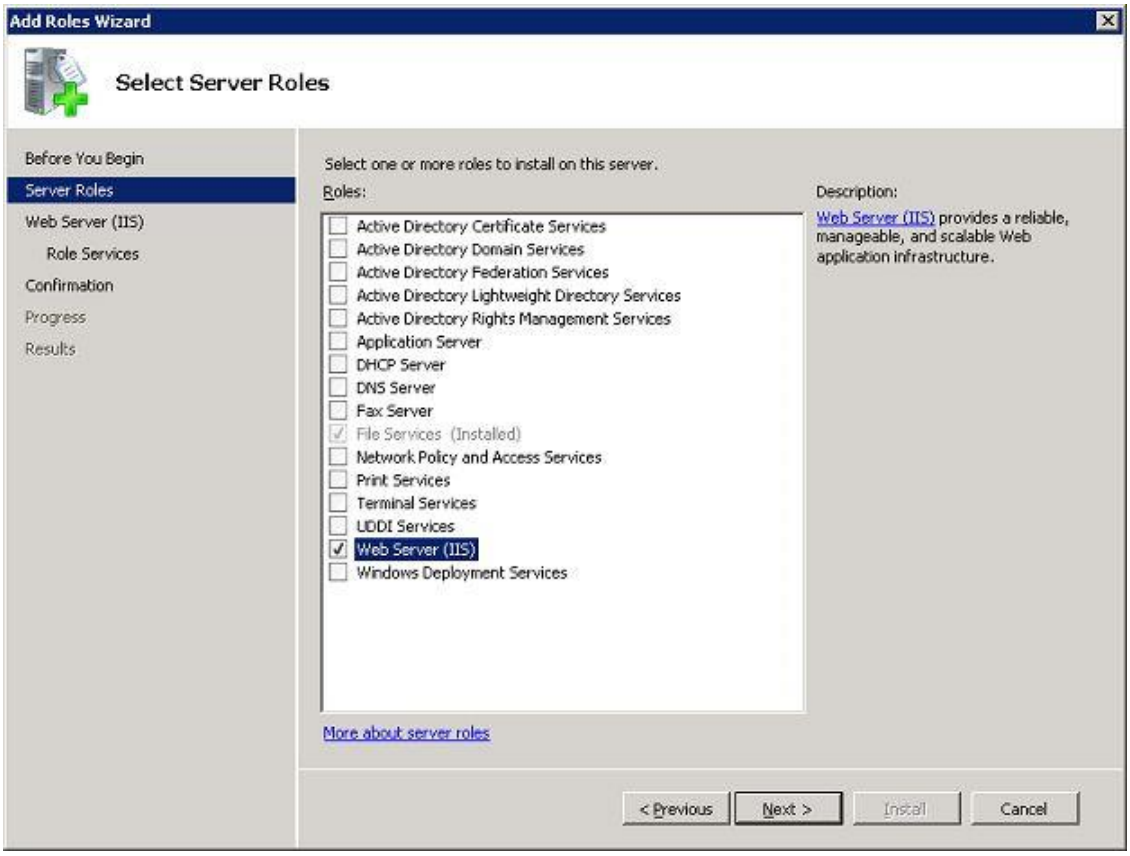

Figure 2.17: Selecting the Web Server (IIS) role

7. Then, when Figure 2.18 appears, click on the **Next** button to switch to the next step of the role installation.

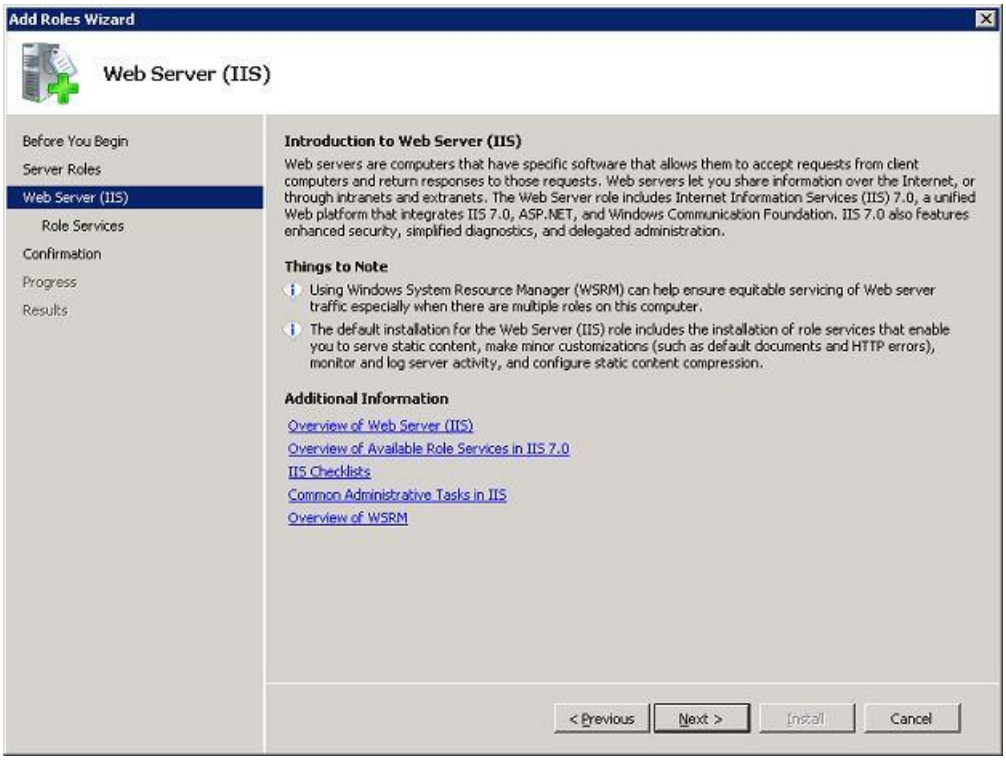

Figure 2.18: An introduction to the web server role

8. The next step will prompt you to choose the role services. Select all the listed services and click the **Next** button to proceed. **Make sure that the IIS Management Scripts and Tools feature in particular is installed and enabled for the 'Web Server' role.**

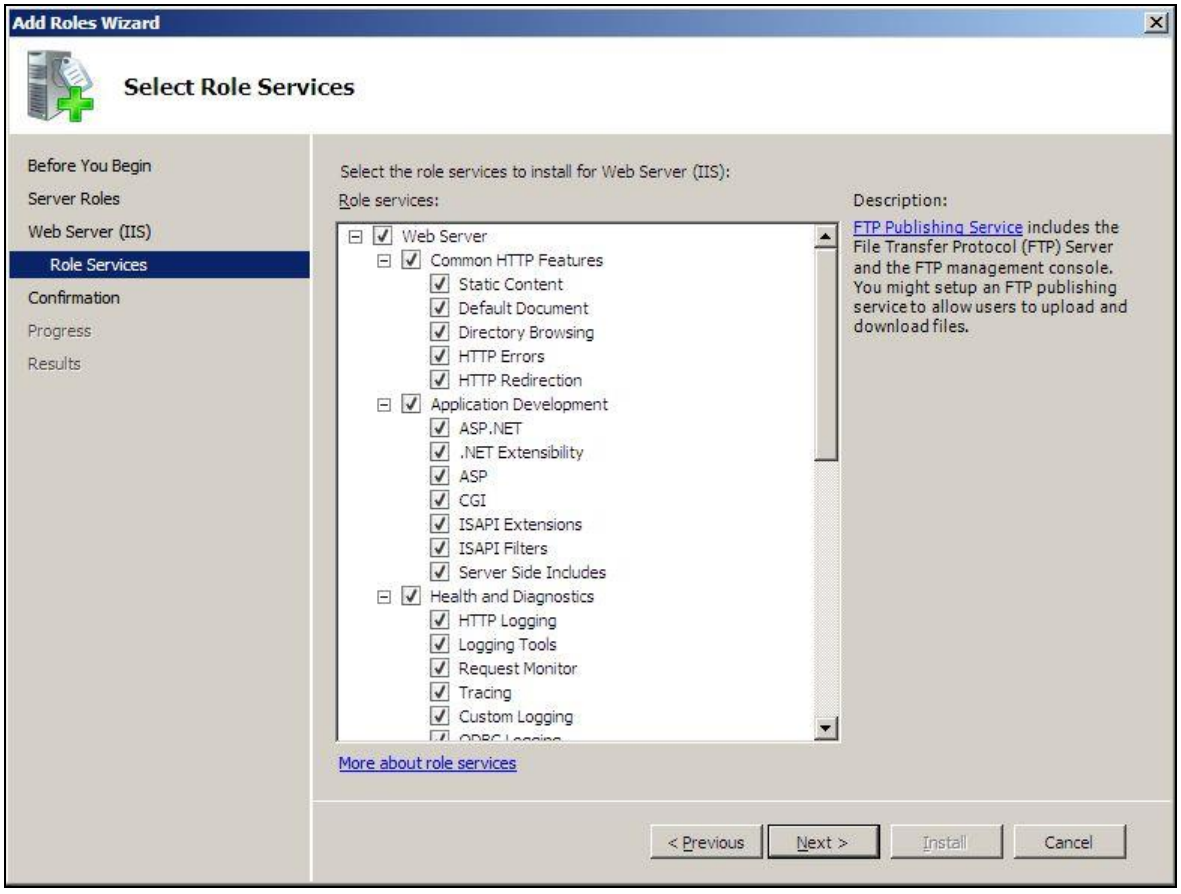

Figure 2.19: Selecting the required role services

9. The screen that appears subsequently provides a summary of your specifications. After reviewing your selections, you can confirm installation of the chosen web server role by clicking on the **Install** button in Figure 2.20.

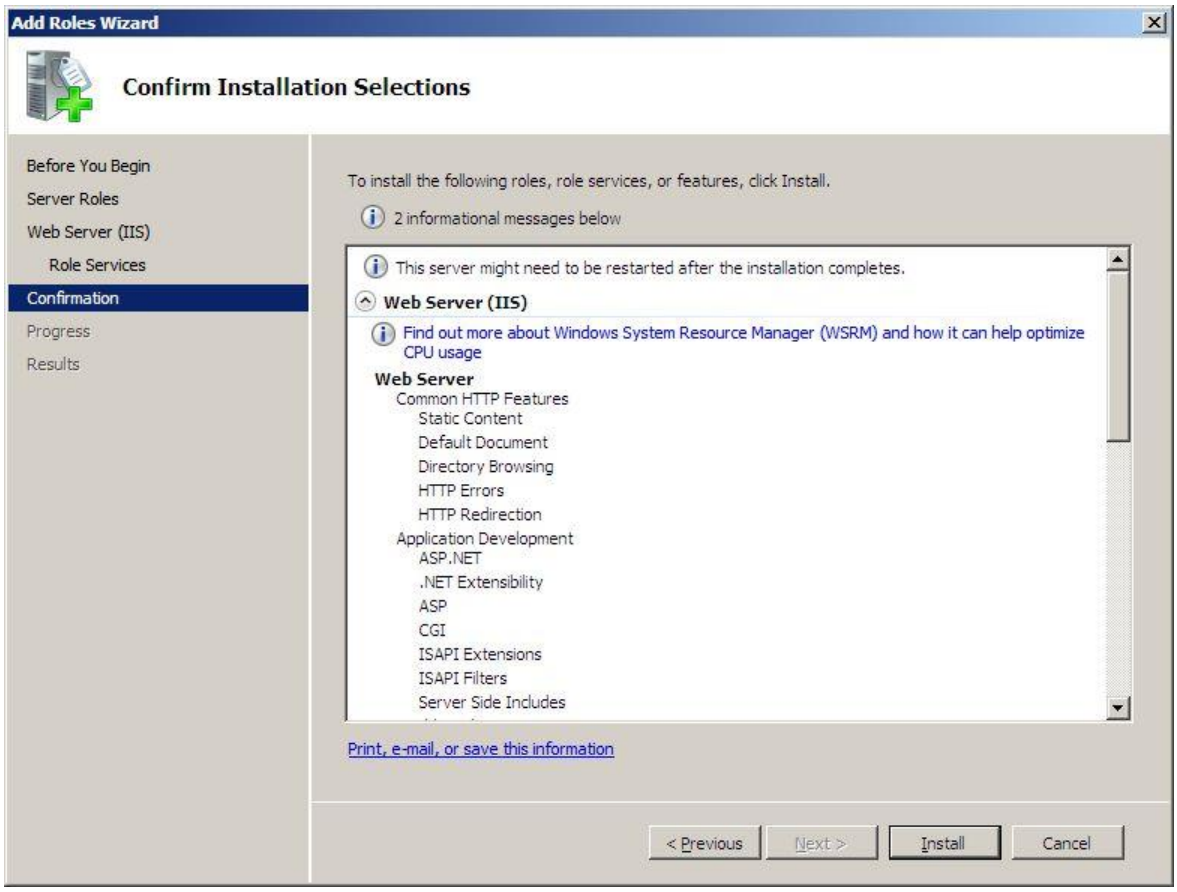

Figure 2.20: Installing the web server role

10. Once installation completes successfully, Figure 2.21 will appear confirming the success of the installation.

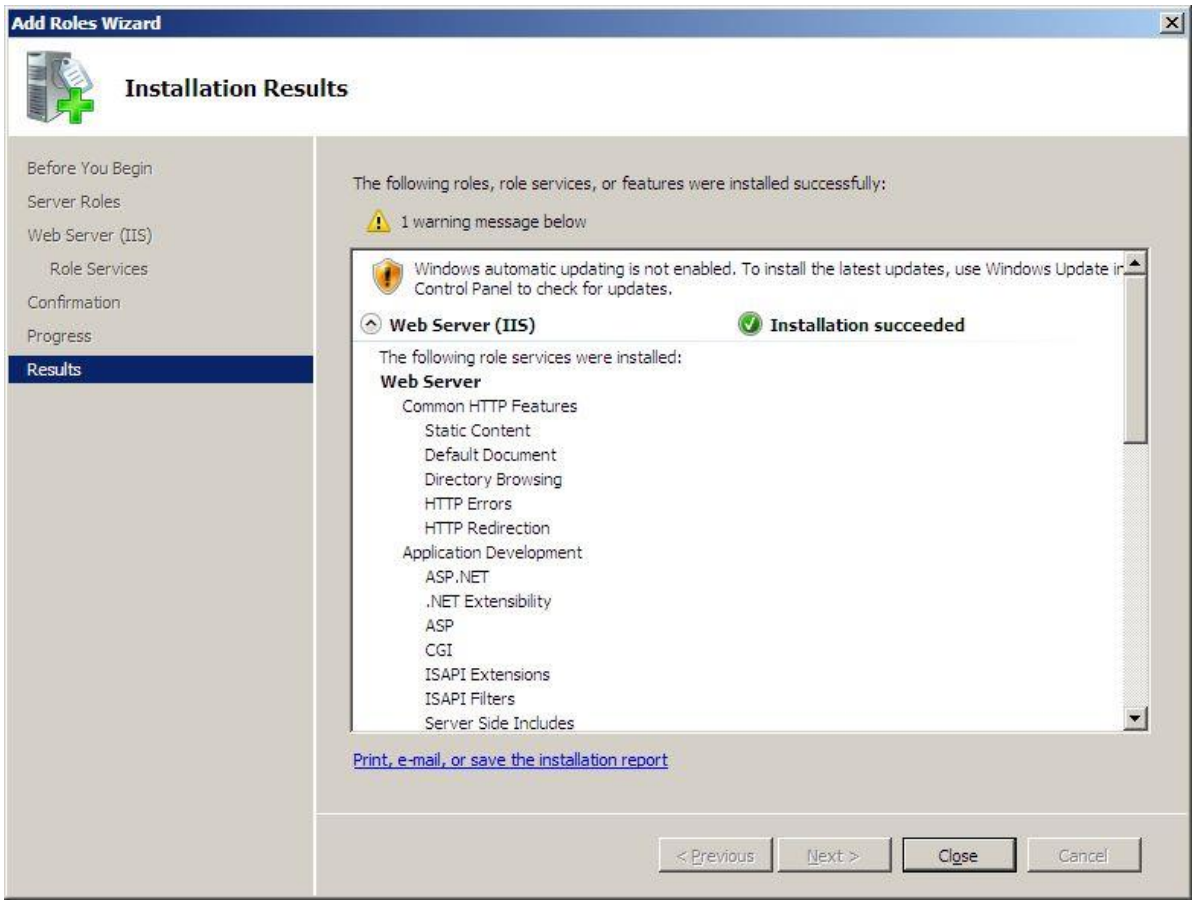

Figure 2.21: A message indicating that installation was successful

11. Click on the **Close** button in Figure 2.21 to close the wizard. Figure 2.22 will then appear displaying the newly installed role.

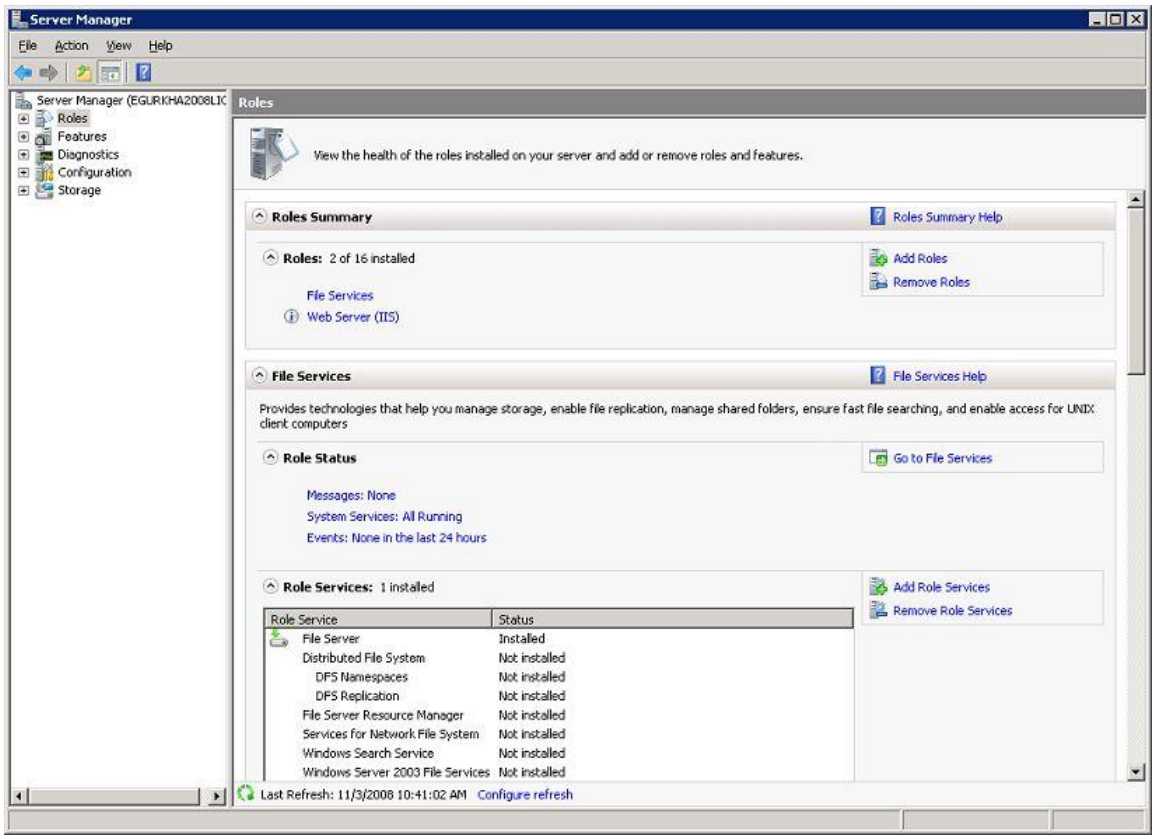

Figure 2.22: The Roles page in the right panel displaying the Web Server (IIS) role that was just installed

2.1.3 Configuring the eG Agent to Monitor the Web Transactions to Web Sites on an IIS Web Server Operating on Windows 2008

To perform web site transaction monitoring on an IIS web server executing on Windows 2008, you need to install and configure **Advanced Logging** on the target IIS web server, soon after you create the **Web Server** role on the Windows 2008 server.

IIS Advanced Logging is an extension for Internet Information Services (IIS) 7 that provides enhanced data collection and real-time server and client-side logging capabilities. It can be managed by using IIS Manager and other tools that can work with the IIS 7 configuration system.

The Advanced Logging feature supports complex Web and media delivery scenarios that demand flexibility and control. These scenarios may require custom logging fields, real-time access to data, greater control over what gets logged and when, extensibility for new sources of data, the ability to consolidate log data posted by clients and correlate it to server data, the option of sharing data from various sources and storing it in multiple logs, capturing system-state information, inclusion of canceled requests in logs, and even logging multiple times per request.

In order to monitor the web transactions to IIS 7 (that is bundled with the Windows 2008 server), the eG agent requires that the **Advanced Logging** be installed and configured on IIS 7. The steps in this regard have been discussed below:

1. Login to the IIS host.

2. Download the executable that installs the **Advanced Logging** feature from any of the following URLs, depending upon whether the IIS 7 installation is a 32-bit one or a 64-bit one:

32-bt/64-bit	URL
32-bit	http://www.microsoft.com/downloads/en/details.aspx?FamilyID=4d110e78-95cb-4764-959c-b8afc33df496&displaylang=en
64-bit	http://www.microsoft.com/downloads/en/details.aspx?FamilyID=793051A8-36A0-4342-BDFE-47A6B0E3488F

3. Once the download is complete, go to the directory to which the executable was downloaded and double-click on it.
4. Figure 2.23 will then appear. Accept the license by selecting the **I accept the terms in the License Agreement** check box, and click on the **Install** button to proceed with the installation.

Figure 2.23: Accepting the license agreement

5. Once the installation ends, Figure 2.24 will appear indicating the successful installation of the **Advanced Logging** feature. Click the **Finish** button to exit the wizard.

Figure 2.24: Finishing the installation

6. Next, proceed to configure the Advanced Logs. For that, first, open the **Internet Information Services (IIS) Manager** console using the menu sequence: Start -> Programs -> Administrative Tools -> Internet Information Services (IIS) Manager. Figure 2.25 will then appear.

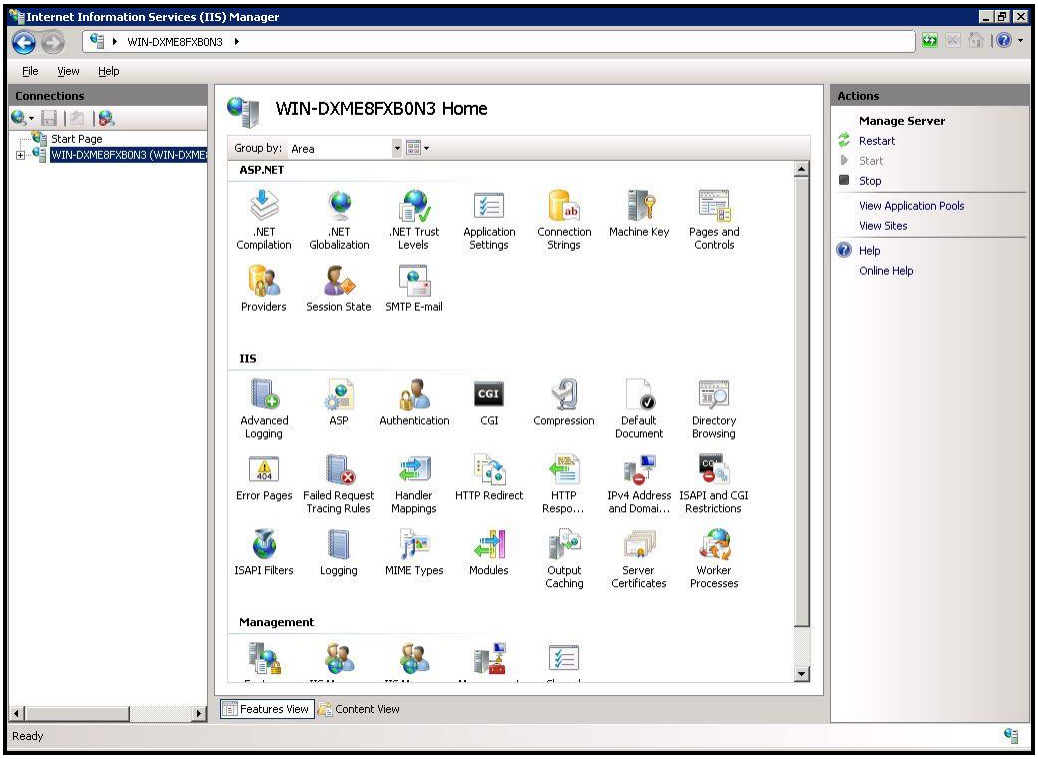

Figure 2.25: The Internet Information Services (IIS) Manager console

7. Click on the node representing the IIS web server host in the tree-structure in the left panel of the console. The right panel will change to display a variety of options. In the **IIS** section of the right panel, click on the **Advanced Logging** option. Figure 2.26 will then appear. In the **Actions** list in the right panel, click on the **Add Log Definition** option (as indicated by Figure 2.26) to add a new log definition.

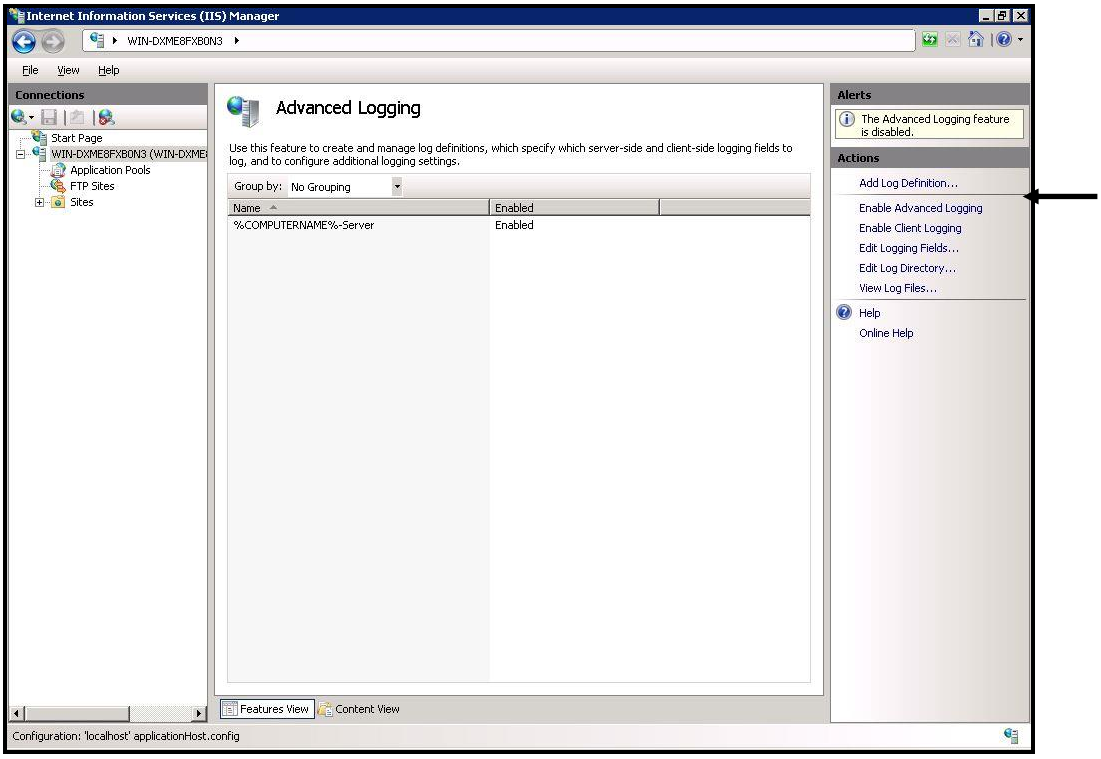

Figure 2.26: Viewing the list of log definitons that pre-exist

8. In the **Log Definition** page that appears, specify **WebAdapterFile** as the **Base file name**. Check the **Enabled** option, the **Publish real-time events** option, and the **Write to disk** sub-option.

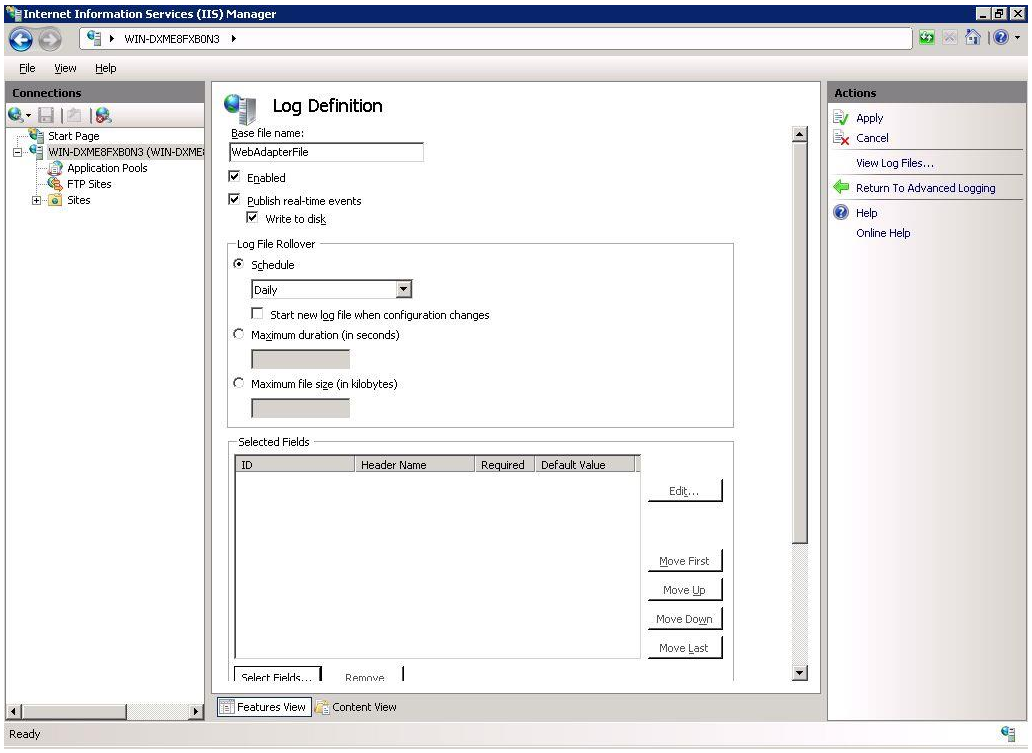

Figure 2.27: Adding a new log file definition

9. Then, click on the **Select Fields** button at the bottom of the **Log Definition** page to select he server-side and client-side logging fields to be logged in the specified log file. Doing so will invoke Figure 2.28, from which you wil have to select the following fields:

   - UserName
   - URI-Stem
   - URI-QueryString
   - Time-Local
   - Time Taken
   - Status
   - Server-IP
   - Server Port
   - Server Name
   - Site Name
   - CPU-utilization
   - Bytes Sent
   - Bytes Received
   - Host
   - Client Ip
   - Date-local

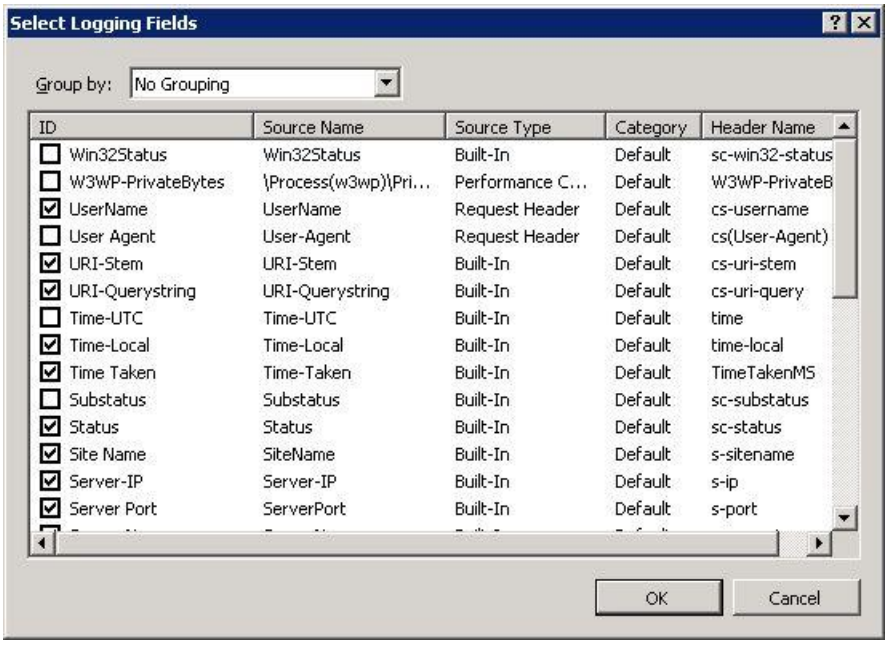

Figure 2.28: Selecting the logging fields to be logged

10. Click on the **OK** button in Figure 2.28 to confirm the selection. When this is done, the **Selected Fields** section of the **Log Definition** page will get updated with your selection (see Figure 2.29). Use the **Move First**, **Move Up**, **Move Down**, and **Move Last** buttons adjacent to your selection to re-arrange the sequence of the logging fields. The desired sequence is as follows:

    - Time-Local
    - Host
    - Server-IP
    - Server Port
    - Status
    - URI-stem
    - URI-QueryString
    - CPU-utilization
    - Bytes Sent
    - Bytes Received
    - Time Taken
    - Server Name
    - Site Name
    - User Name
    - Client Ip
    - Date-local

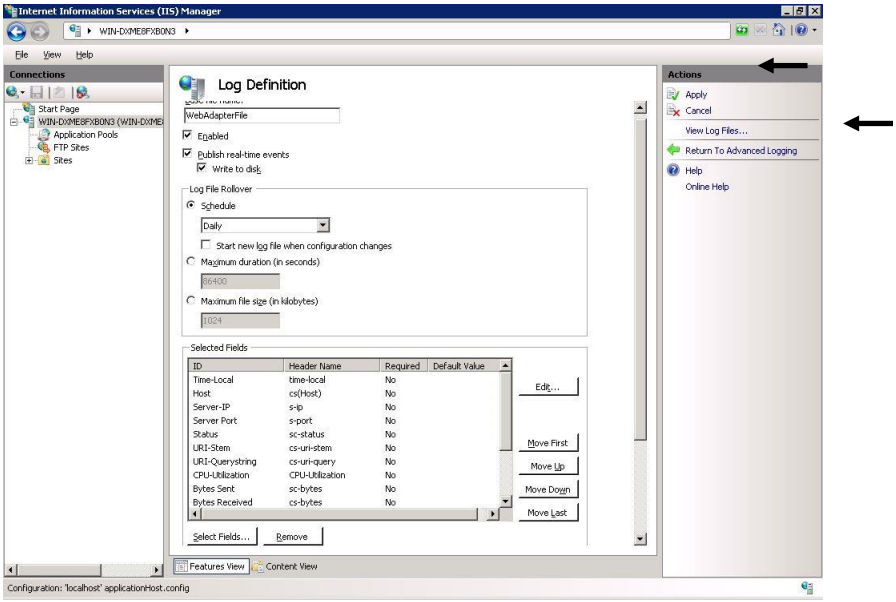

Figure 2.29: Re-arranging the sequence of the logging fields

11. Then, apply the changes by clicking on the **Apply** button indicated by Figure 2.29 above. Once the changes are saved, click on the **Return to Advanced Logging** option indicated by Figure 2.29 above. Figure 2.30 will then appear. In the right panel of Figure 2.30, you will find that the newly added **WebAdapterFile** is appended to the list of log file definitions that pre-exist.

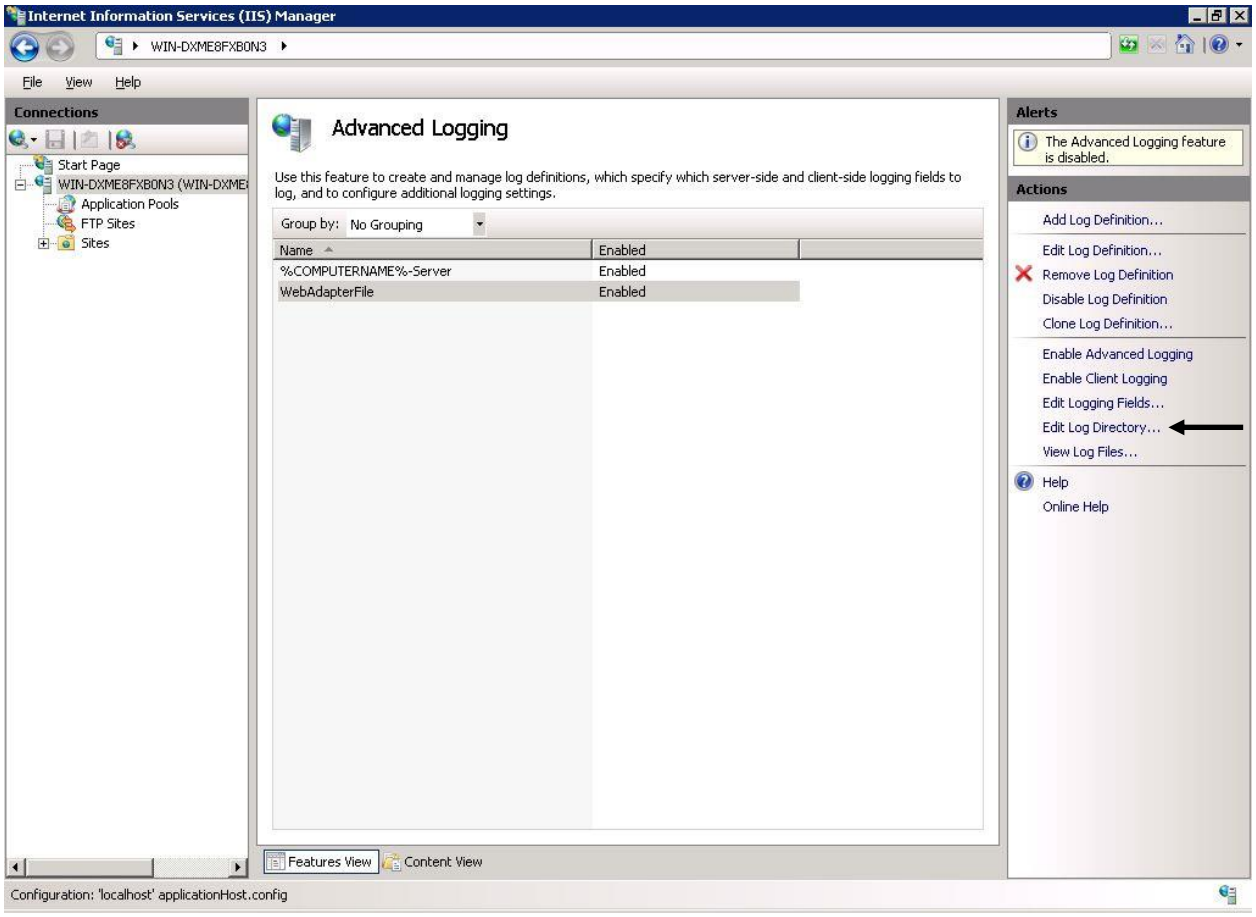

Figure 2.30: The newly added log definition displayed in the list of log files that pre-exist

12. Now, select the **WebAdapterFile** entry in Figure 2.30 and click on the **Edit Log Directory** option in the **Actions** list, as indicated by Figure 2.30. When Figure 2.31 appears, change the default values of the **Server log directory** and **Default site log directory** text boxes to **<EG_INSTALL_DIR>\agent\logs\IISAdvlogs** directory. Then, click the **OK** button therein.

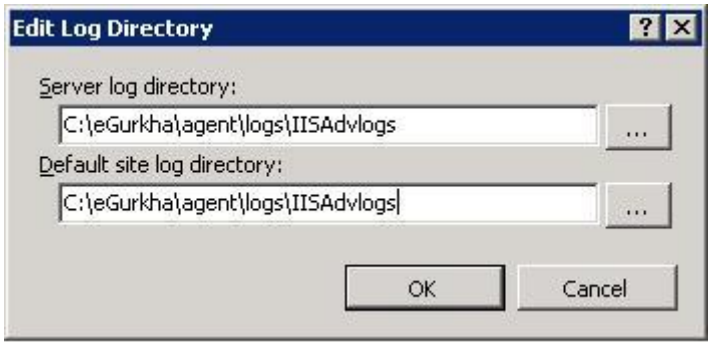

Figure 2.31: Changing the server log and default site log directories

13. You will then return to Figure 2.30. Select the **WebAdapterFile** entry yet again, and this time, click on the **View Log Files** option in the **Actions** list. This will invoke Figure 2.32, where all the log files saved to the **<EG_INSTALL_DIR>\agent\logs\IISAdvlogs** directory will be displayed.

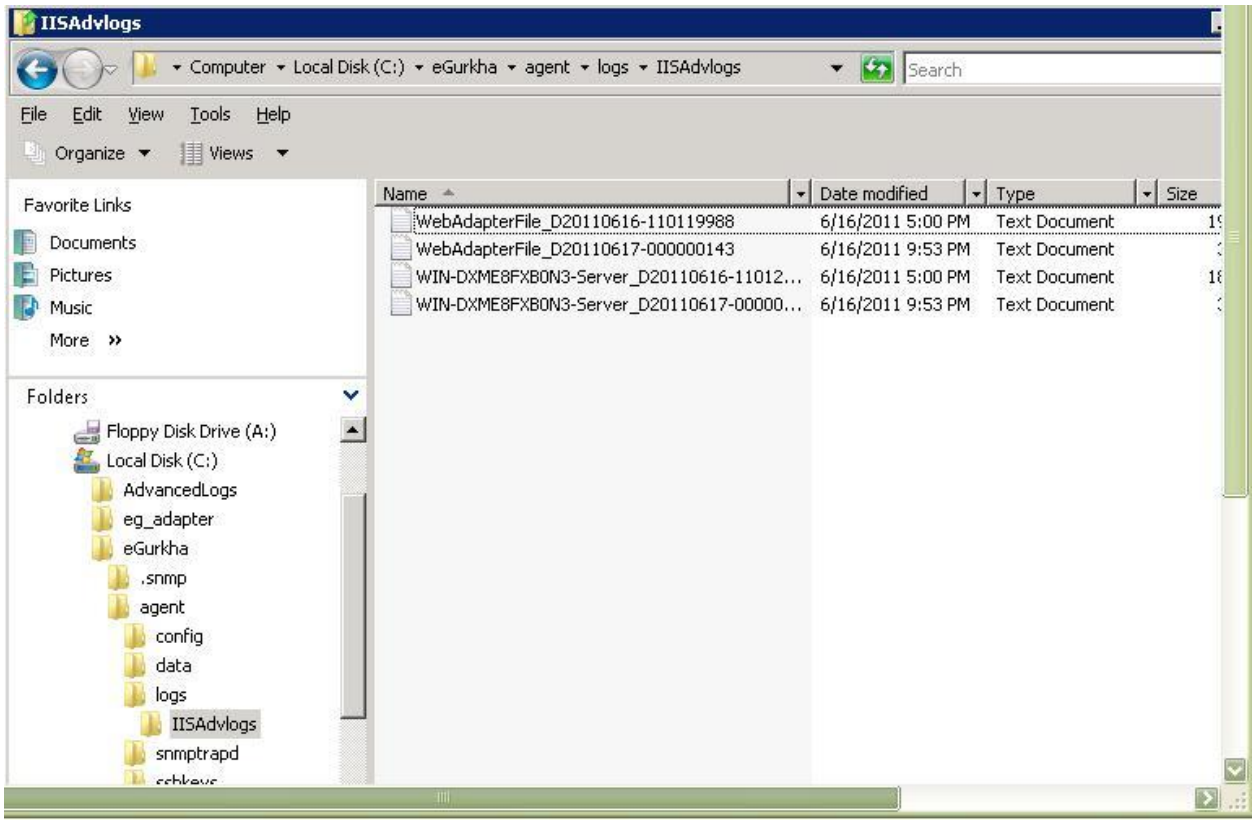

Figure 2.32: List of log files saved to the AdvancedLogs directory

14. To view a log file, click on any of the log files in the list of Figure 2.32. The chosen log file will then open in Notepad as depicted by Figure 2.33.

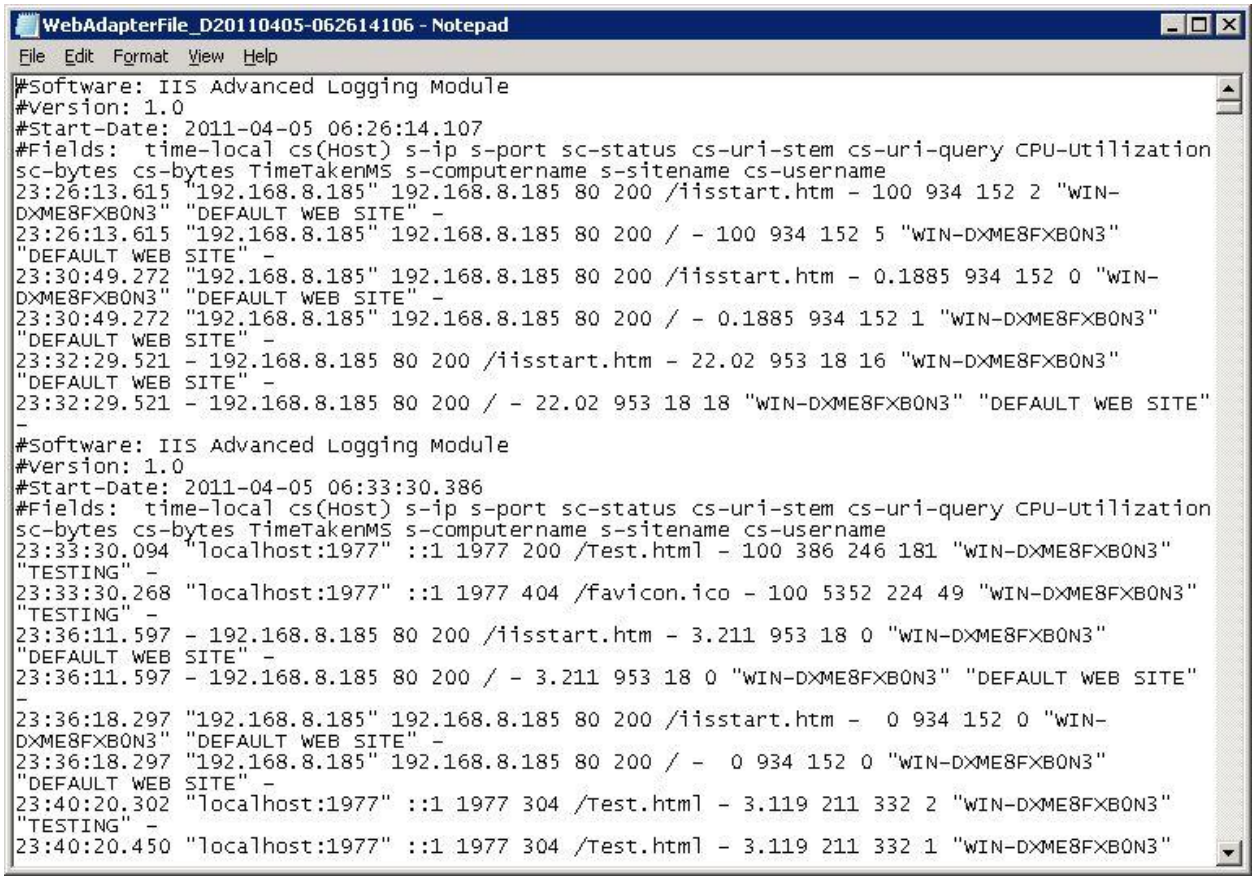

Figure 2.33: Viewing the log file

2.2 Administering the eG Manager to monitor the IIS Web Server

After installation of eG agent, please follow the following steps to configure eG to monitor an IIS web server.

1. Login to eG user interface as an administrator.
2. If the IIS Web Server is already discovered, navigate to the **COMPONENTS- MANAGE/UNMANAGE** page following the menu Infrastructure -> Components -> Manage/Unmanage, to manage it.
3. On the other hand, if the IIS Web Server is yet to be discovered, then run the discovery procedure to get IIS Web servers discovered, or manually add the IIS Web server. To run the discovery, open the **START DISCOVERY** page using the Infrastructure -> Components -> Discover menu sequence, and click the **Start Discovery** button.
4. To manually add the IIS Web Server, go to the **ADD/MODIFY COMPONENTS** page through the Infrastructure -> Components -> Add/Modify menu sequence and then add the server as indicated by Figure 2.34.

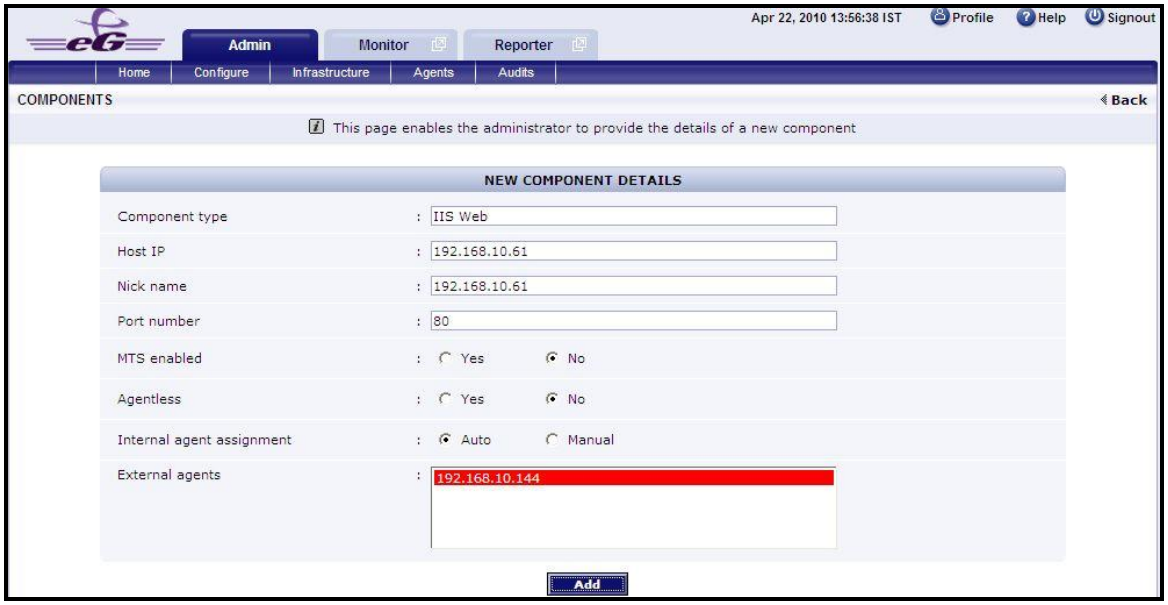

Figure 2.34: Adding an IIS web server

5. If a Microsoft Transaction server (MTS) is available on the target IIS web server, then, you can manage the MTS server along with your IIS web server by setting the **MTS enabled** flag to **Yes**. This will automatically add a **Microsoft Transaction** server component, with the same IP-nickname as the IIS web server (see Figure 2.35)

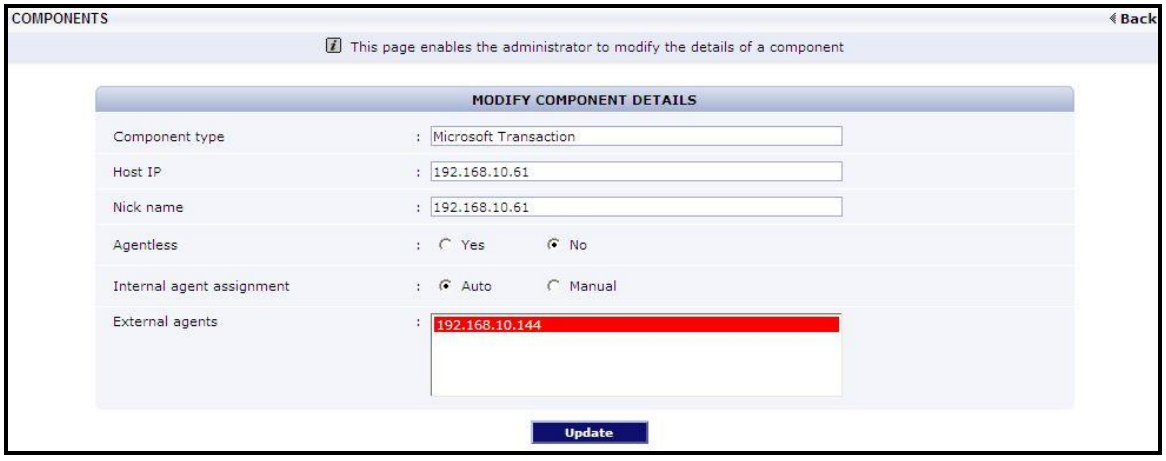

Figure 2.35: An MTS server being automatically added

6. Components added using the **ADD/MODIFY COMPONENTS** page will automatically appear in the **MANAGED COMPONENTS** list of the **COMPONENTS - MANAGE/UNMANAGE** page (see Figure 2.36). Discovered servers, on the other hand, need to be managed manually using the **COMPONENTS - MANAGE/UNMANAGE** page (see Figure 2.36). For accessing this page, use the menu sequence Infrastructure -> Components -> Manage/Unmanage. The screen below shows all the IIS Web Servers discovered in a given range but not managed. Select the component-type that requires monitoring from the **Component type** list. To manage a particular component of the selected type, select the component from the **UNMANAGED COMPONENTS** list and click on the **<< Manage** button and finally, the **Update** button.

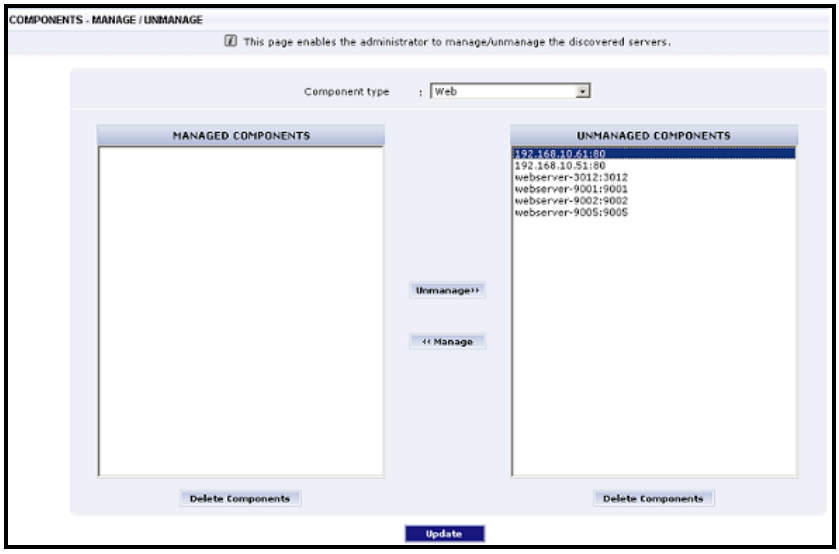

Figure 2.36: Viewing the list of unmanaged IIS web servers

7. After managing the web server, the screen would appear as shown below:

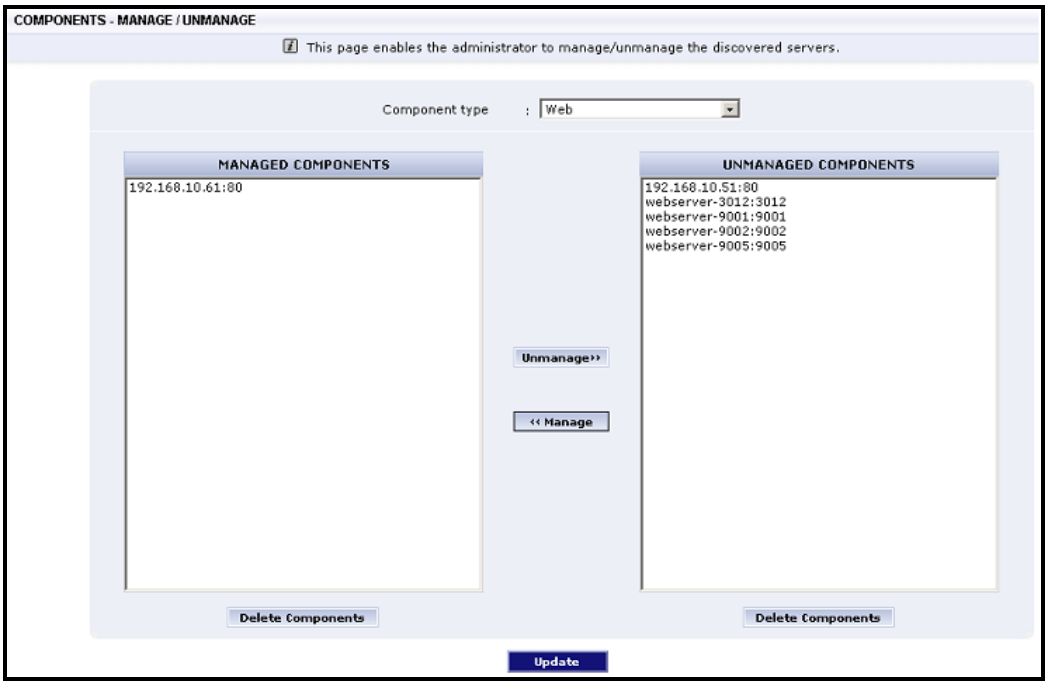

Figure 2.37: Managing an IIS web server

8. Then, proceed to configure web sites and related transactions for the IIS web server in the same manner as done for the Apache web server.

T Please refer to Section 1.3 above for a more elaborate discussion on how to configure web sites and transactions.

9. Once this is done, sign out of the administrative interface.

2.3 Monitoring the IIS Web Server

To view the measurements reported by the eG agent, log in as the monitor / supermonitor user. Click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu. By default, the IIS web server will appear as an independent server. To view the measurements pertaining to your IIS web server, just click on the symbol representing the same in the **COMPONENT LIST** page. On clicking, a screen displaying the layer model, tests, and measurements of the web server will appear .

If you wish to have eG plot a graph of the measurements, click on the **Graph** button against the corresponding measurement. To view the history of a measurement, click on the **History** button. Clicking on the **Feedback** button will enable you to provide details of action taken to fix errors (if any).

Once you are through with viewing measurements, exit the monitor user interface by selecting the **SIGNOUT** option.

2.4 Troubleshooting

If the eG agent does not report any measures pertaining to the transactions that have been configured for an IIS web server, then restart the World Wide Web (WWW) Publishing service. To achieve this, do the following:

1. Select the **Services** option from the Start -> Programs -> Administrative Tools menu (see Figure 2.38).

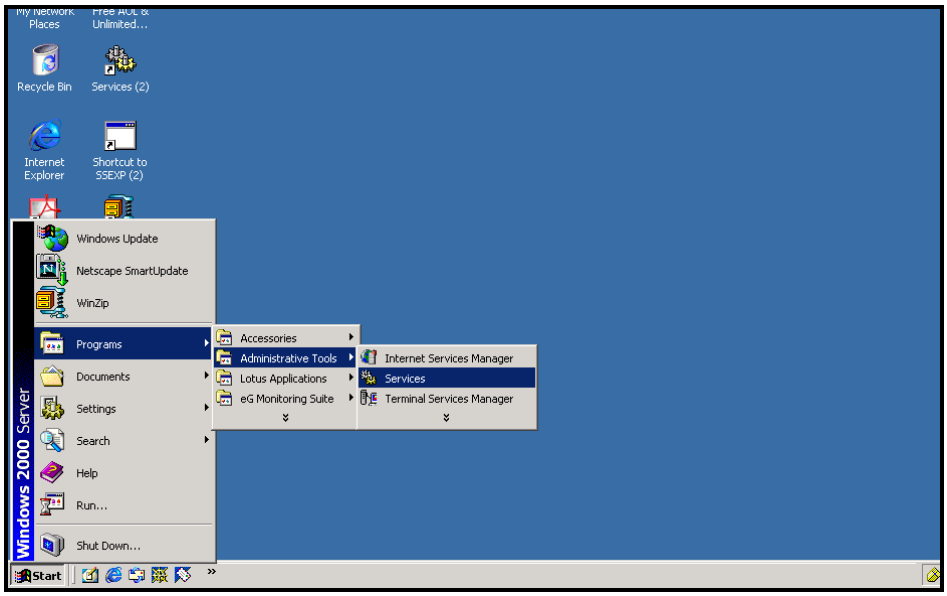

Figure 2.38: Selecting the Services option from the Administrative Tools menu

2. From the right pane of the window that appears, select **World Wide Web Publishing Service**, right-click on it, and then, select **Stop** from the shortcut menu that appears to stop the service (see Figure 2.39).

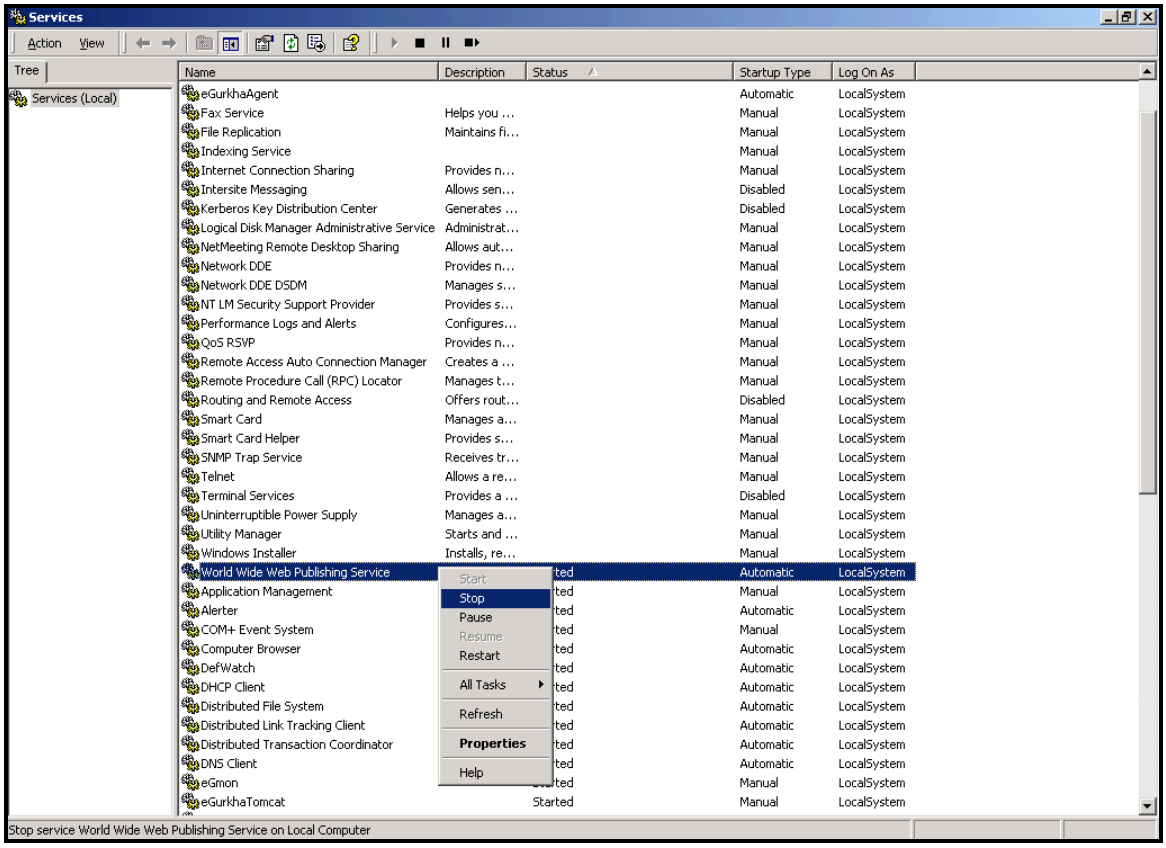

Figure 2.39: Stopping the World Wide Web Publishing Service

3. Now, to start the service, select it and right-click on it again. Then, from the shortcut menu, select **Start** (see Figure 2.40).

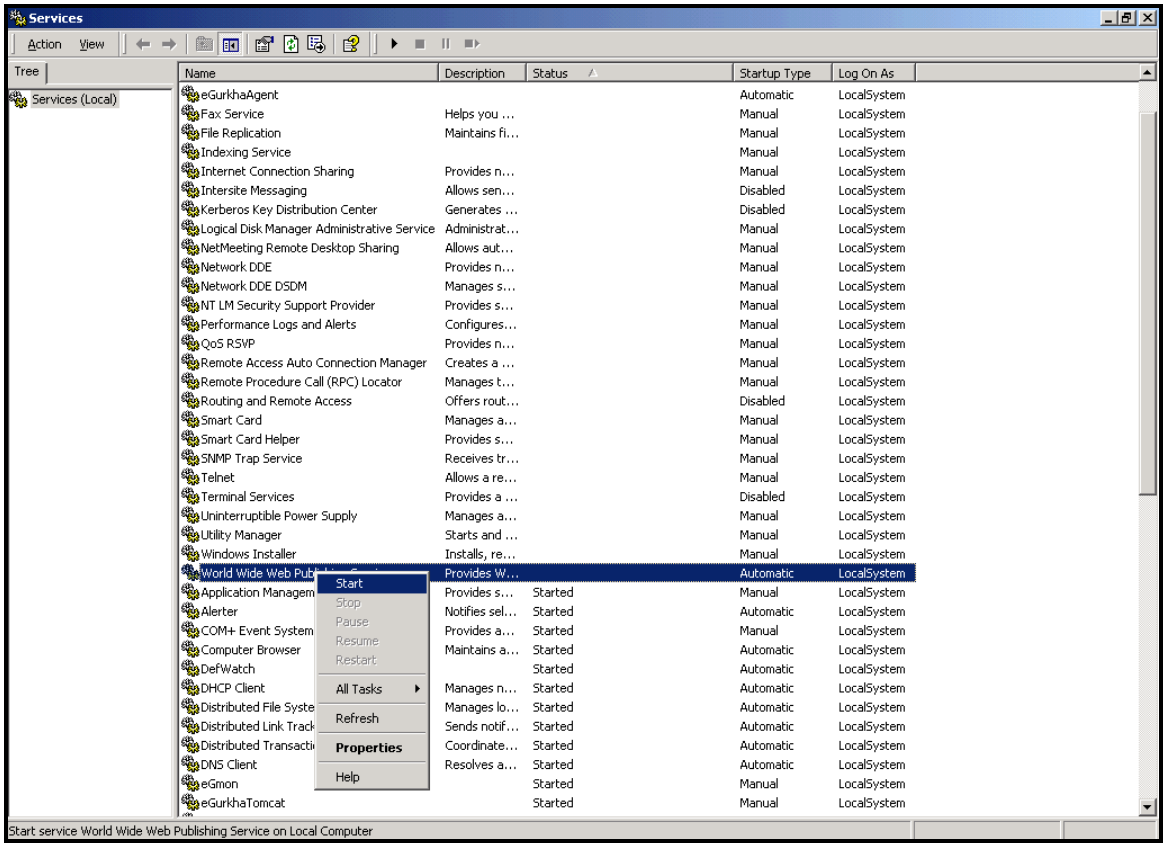

Figure 2.40: Starting the service

Now, log into the monitor interface to check whether the transactions are being monitored. If measures are still not been reported, then, do the following:

1. Select the **Internet Services Manager** option from the Start -> Programs -> Administrative Tools menu (see Figure 2.41).

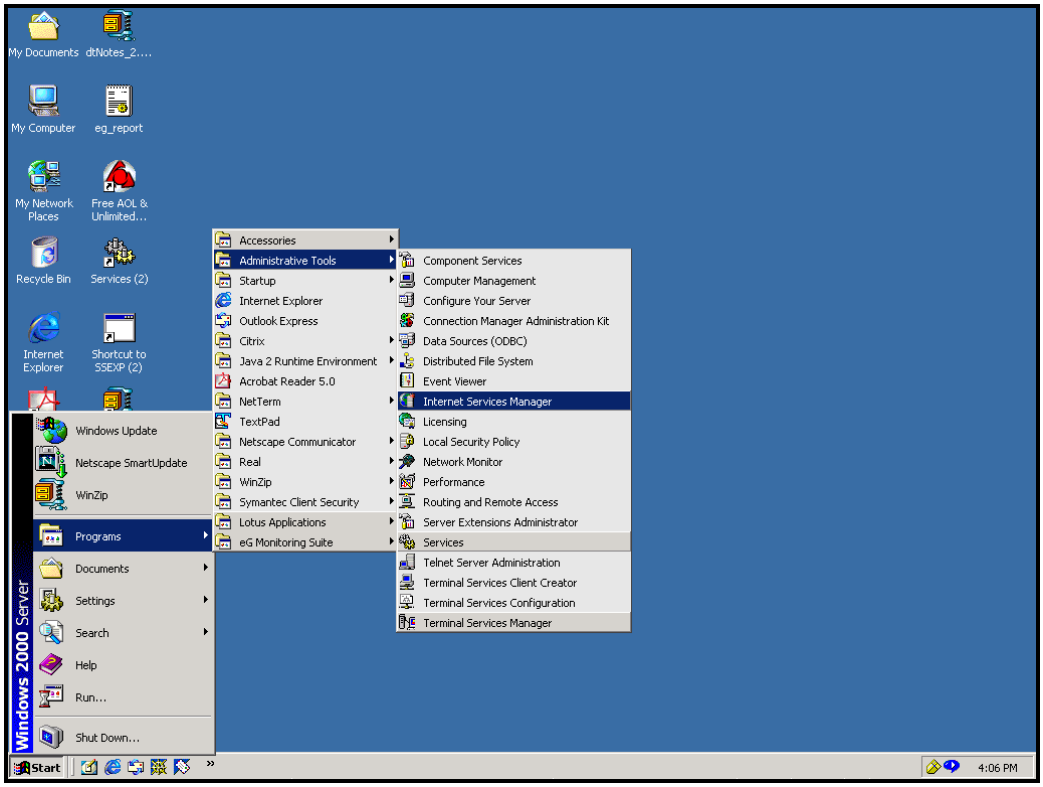

Figure 2.41: Selecting the Internet Services Manager option on Windows 2000

2. If the IIS web server executes on a Windows 2000 host, then, from the left pane of the **Internet Information Services** window that appears, select the IIS web server's host, right-click on it and choose the **Properties** option (see Figure 2.42). In case of the Windows 2003 host on the other hand, expand the node representing the IIS web server's host in the left pane, right-click on the **Web Sites** sub-node within, and pick the **Properties** option (see Figure 2.43).

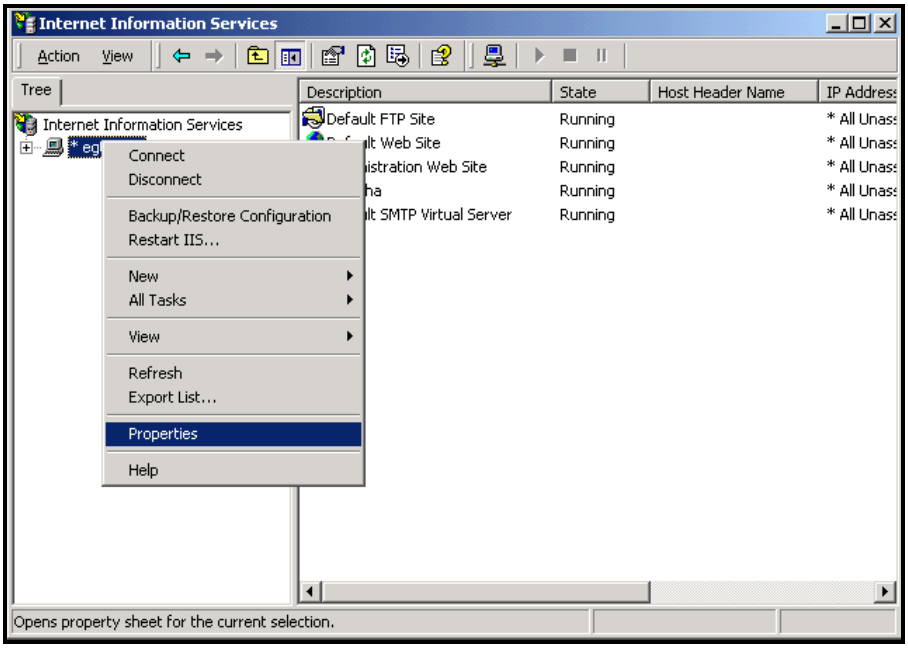

Figure 2.42: Editing the properties of the IIS web server's host (in IIS console on Windows 2000)

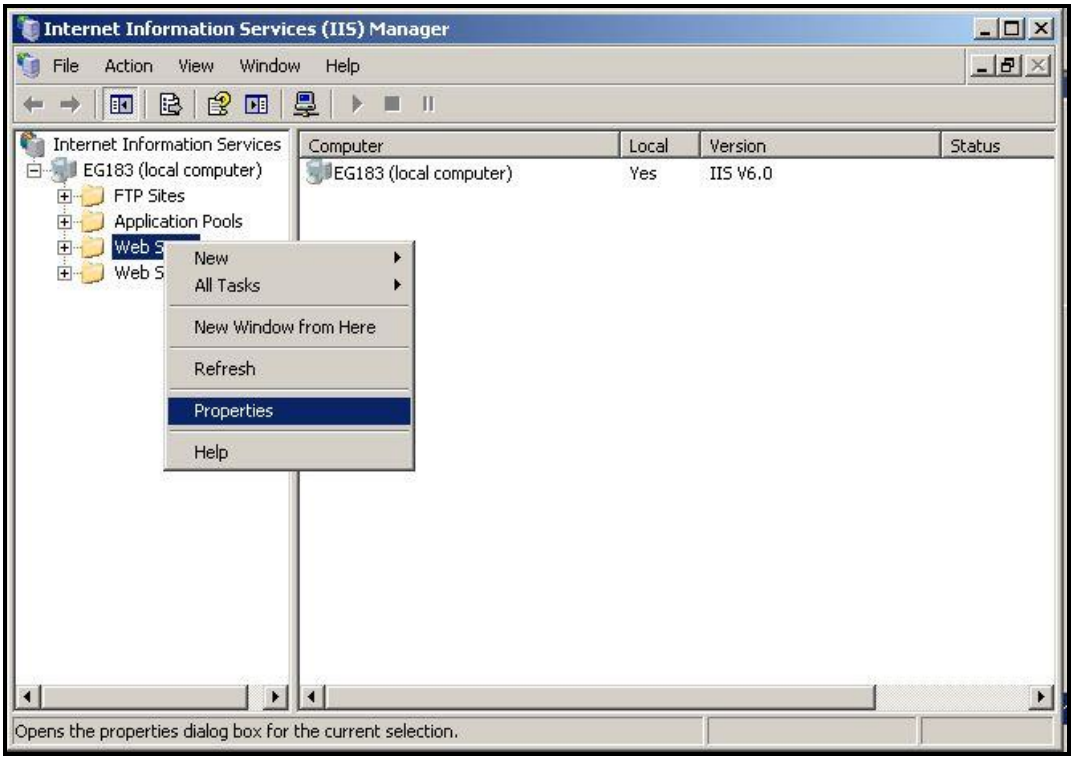

Figure 2.43: Picking the Properties option from the Web Sites tab (in the IIS console on Windows 2003)

3. On a Windows 2000 host, selecting the web server host's **Properties** will lead you to Figure 2.44.

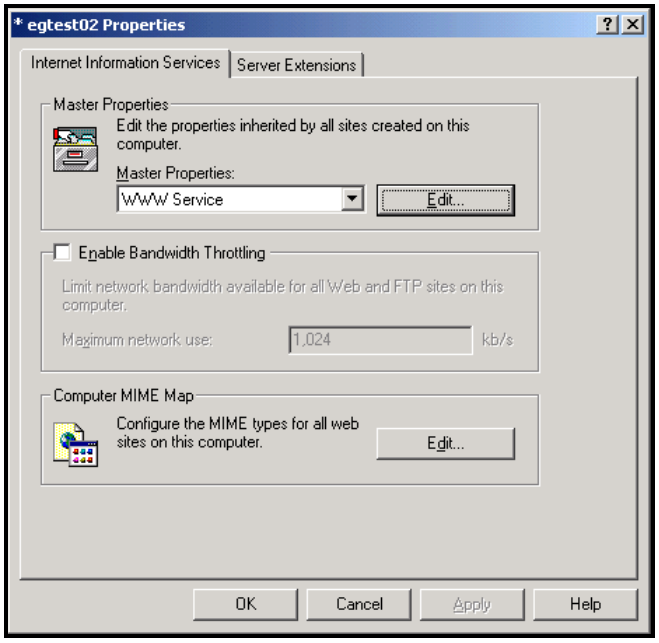

Figure 2.44: The Properties dialog box

As indicated by Figure 2.44, select **WWW Service** from the **Master Properties** list and click the **Edit** button to edit the properties of the selected service. Doing so will result in the display of a dialog box containing many tab pages. Click on the **ISAPI Filters** tab page (see Figure 2.45).

eG Enterprise's WebTransFilter

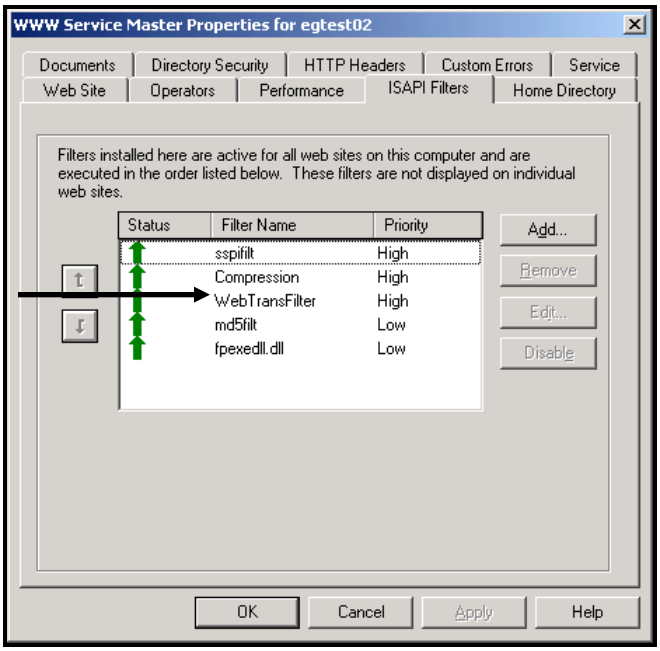

Figure 2.45: Viewing the status of the ISAPI filters

4. On a Windows 2003 host, selecting the **Properties** option of the **Web Sites** node will lead you to a **Web Sites Properties** dialog box. Click on the **ISAPI Filters** tab page in that dialog box, and look for the **WebTransFilter** therein.

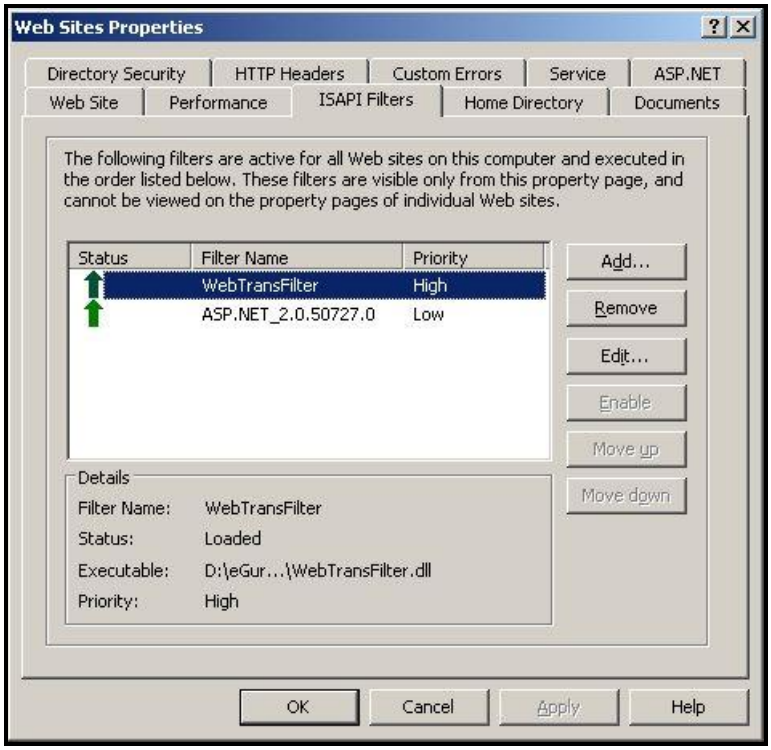

Figure 2.46: The Web Site Properties dialog box

5. Check the status of the WebTransFilter listed in the tab. Transaction monitoring in web servers is governed by this filter. The status of this filter has to be **GOOD** (indicated by an up arrow in green color) (see Figure 2.46), for the eG agent to perform transaction monitoring effectively. If the status of the filter is **BAD** (represented by a down arrow in red color) or **UNKNOWN** (indicated by a down arrow in blue color), then, you might have to reload the filter. For that, first, select the filter in Figure 2.46 and click the **Remove** button alongside it to remove it. Then, click the **Add** button. Doing so will result in the display of a screen wherein the **Filter Name** and the path to the filter **Executable** has to be specified (see Figure 2.47).

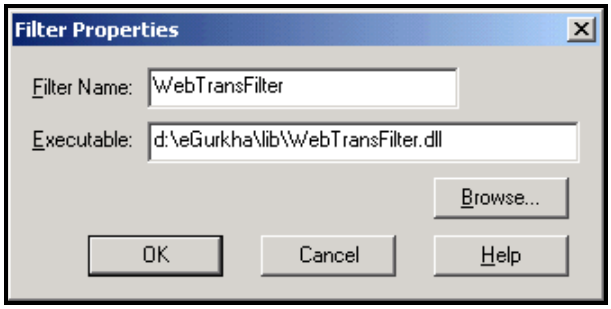

Figure 2.47: Adding the filter

6. The WebTrans filter will be available in the **<EG_HOME_DIR>/lib** directory. Specify the same against the **Executable** text box and then, click the **OK** button to register the changes.
7. Clicking on the **OK** button will take you back to the dialog box depicted by Figure 2.45. Click on the **OK** button in the dialog box and then, on the **OK** button in Figure 2.44.

8. Once the filter is loaded, restart the WWW service once again by following the procedure discussed previously.

Chapter 3

Configuring and Monitoring an iPlanet/SunONE Web Server

The eG Enterprise suite supports internal monitoring of an iPlanet/SunONE web server only on Solaris environment. The process of configuring and monitoring the Netscape / iPlanet Enterprise Server using eG, involves three simple steps:

- Configuring a iPlanet/SunONE Enterprise Server to work with the eG agent
- Administering the eG manager to monitor the iPlanet/SunONE Enterprise Server
- Monitoring the iPlanet/SunONE Enterprise Server

3.1 Configuring a iPlanet/SunONE Web Server to work with the eG Agent

The eG web adapter on an iPlanet/SunONE Enterprise Server can be configured manually, or using the **setup_webadapter** script provided by eG. When setting up the agent (using the **setup_agent** script), the **setup_webadapter** script is called if you choose to configure monitoring for a web server.

Before configuring the web adapter, ensure the following:

- eG expects 2 directories namely "**logs**" and "**config**" under the iPlanet root directory (iPlanet Home dir). If these directories do not exist, the web adapter configuration will fail.
- The **setup_webadapter** script assumes that the webserver start and stop scripts are called **start** and **stop** respectively. These scripts should be available in the server root directory itself. After the web adapter configuration is completed, the original versions are backed up as **start.pre_egurkha** and **stop.pre_egurkha**, respectively.
- If the iPlanet/SunONE directories are not in the above structure, you may need to set up a directory structure in this format (using symbolic links) for the **setup_webadapter** script to work.
- During setup, the web adapter configuration prompts you to confirm whether the server is configured to support SSL or not. Check the `magnus.conf` file in the `config` directory of the iPlanet/SunONE web server. Look for the specification "**Security on**". This directive indicates that SSL is enabled. Otherwise, SSL is not enabled.

3.1.1 Configuring the eG Web Adapter for an iPlanet/SunONE Web Server (before Version 6.0)

eG Enterprise's unique web adapter technology enables individual transactions performed by users of a web site to be tracked in real-time without the need for explicit, expensive logging.

The web adapter is part of the eG agent package for Solaris. When the agent on Solaris is configured, it is enabled to communicate with the web adapter to report statistics in real-time to the eG manager.

For configuring the eG web adapter for an iPlanet/SunONE web server (before version 6.0), you can use the **setup_webadapter** script in the **/opt/egurkha/bin** directory. The steps involved in configuring the iPlanet/SunONE web server to use the web adapter technology are:

A user can install the eG web adapter capability only for the web servers that he/she has the permission to administer.

1. First, run the command on the iPlanet / SunONE web server host: **/opt/egurkha/bin/setup_webadapter**
2. The **setup_agent** script on Solaris executes this command automatically. Hence, if you get here from step 9 of Section 3.1.1 above, you do not have to explicitly run this command.
3. Upon executing the above command, the following message will appear. Type **n** to continue with the setup.

```
Only a SunONE/iPlanet web server can be configured for monitoring using
this script.
For monitoring an Apache or IBM web server, please refer to the eG
Installation Manual.
Do you want to continue  y/n? [y]: y
```

4. Upon confirming, you will view the following message. Type **y** here to continue.

```
Note: Only a web server administrator has permissions to configure the eG
web adapter capability for a web server.
Do you want to continue  y/n? [y]: y
```

5. Next, setup will want to know whether the current user is the web server's administrator or not.

```
Are you the administrator of this web server? y/n [y]:
```

6. Entering **y** or **n** here will bring up the following query:

```
Please enter the user name of the web server's administrator:
Next, you will need to enter the web server administrator's
password...

Password:
```

7. Next, enter the root directory of the web server to be configured. Also, indicate whether the web server is SSL-enabled.

```
Enter the root directory of the SunONE / iPlanet web server:
/usr/local/web
Is this web server enabled for SSL support? y/n [n]:
```

8. Based on the server type and the root directory, the **setup_webadapter** script proceeds to modify the web server startup scripts to use the eG web adapter when the web server starts up.

9. Then the user has to specify whether this web server is enabled for SSL support. If the user chooses **y**, the setup process configures the web adapter for SSL support. The following error message appears if the web server is not configured with Dynamic Shared Object (DSO) support.

```
****************************************************************
This web server has not been configured with DSO support ...
The eG SSL enabled web adapter cannot be installed.
Please reconfigure this web server with DSO support and
run the command /opt/egurkha/bin/setup_webadapter to
configure the web server with eG agent's web adapter capability
****************************************************************
```

10. Next, the configuration process prompts the user to determine if the user is the administrator of the web server that is to be configured for monitoring by an eG agent. If the user is not the web server administrator, the configuration process prompts the user for the web server administrator's login name and password.

11. The configuration process attempts to update the web server's startup file(s) to include eG-specific start-up information. The configuration process terminates with the following message:

```
****************************************************************
If there are any errors in the above process, you may not have
permission to update the web server's configuration.
Please have the web server's administrator run the command
   /opt/egurkha/bin/setup_webadapter
If there were no errors, the web adapter has been successfully
configured. For the web adapter to be effective, please restart the
configured web server.
****************************************************************
```

12. In the case of an iPlanet web server (prior to version 6.0), the **start** and **stop** files are modified after

retaining a copy of them called **start.pre_egurkha** and **stop.pre_egurkha**. To uninstall the web adapter capability, replace the existing **start** and **stop** files with **start.pre_egurkha** and **stop.pre_egurkha** respectively.

3.1.2 Configuring the eG Web Adapter for a SunONE Web Server Version 6.x on Solaris

To configure the eG web adapter for a SunONE web server version 6.x on Solaris, follow the steps given below:

1. Login as a SunONE install user.
2. Open **magnus.conf** file in the **<SERVER_ROOT>/<SERVERNAME>/config/magnus.conf** file, and append the following lines.

   Init fn="load-modules" shlib="/opt/egurkha/lib/sun1webadapter_6.so" funcs="onServerInit,onChildInit,onLog"
   Init fn="onServerInit" WEB_SERVER_ROOT=<ServerRoot>\<ServerName>
3. Save the file.
4. Next, open the **obj.conf** file from the same location, and insert the following lines.

   NameTransfn=document-root root="$docroot"

   PathCheck fn="onChildInit"

   AddLog fn=flex-log name="access"

   AddLog fn="onLog"
5. Save the file.
6. Finally, restart the webserver.

3.1.3 Manually Configuring the Web Adapter

While manually configuring the web adapter on an iPlanet/SunONE Enterprise server, you need to explicitly modify the following script files:

- start
- stop

These files will be available in the **<SERVER_ROOT>** directory.

Make the following changes to the **start** script.

```
#!/bin/sh
```

```
CLASSPATH=/opt/egurkha/lib/eg_agent.jar:/opt/egurkha/lib/classes111.zip:$CLASSPATH
export CLASSPATH
/opt/egurkha/jre/bin/java IplanetConfig -serverRoot /data3/iplanet4.1
NUM_SEMAPHORES=2
export NUM_SEMAPHORES
WEB_SERVER=netscape
WEB_SERVER_ROOT=/data3/iplanet4.1
LD_LIBRARY_PATH=/opt/egurkha/lib:$LD_LIBRARY_PATH
export WEB_SERVER
export WEB_SERVER_ROOT
nohup $WEB_SERVER_ROOT/eg_nes_mon $WEB_SERVER_ROOT > /dev/null 2>&1 &
LD_PRELOAD=/opt/egurkha/lib/libnes_eg_wa.so:$LD_PRELOAD
export LD_PRELOAD LD_LIBRARY_PATH


- - - - -

The Remaining lines of the start script.

- - - - -

- - - - -
```

The lines in **Bold** are the ones that need to be manually specified in the **start** script file. If the **setup_webadapter** script were used, then these are the lines which will be automatically written by eG into the **start** script.

The line in white has to be additionally specified for iPlanet server version 6.0 alone. The server root directory reference in this line will have to be changed accordingly.

Next, make the following changes to the **stop** script.

```
!/bin/sh


WEB_SERVER_ROOT=/data3/iplanet4.1
export WEB_SERVER_ROOT
/usr/bin/ps -aef -o pid,args | grep "$WEB_SERVER_ROOT" | grep
"eg_nes_mon" | /usr/bin/awk '{ print $1 }' | xargs kill -9 > /dev/null
2>&1


- - - - -
The Remaining lines of the start script.
- - - - -
```

Note:

If the iPlanet/SunONE Enterprise server (version less than 6.0) is SSL-enabled, then the **obj.conf**, available in the **<SERVER_ROOT>/config** directory, will have to be modified. Make the following changes to this file:

```
Init            fn=flex-init
access="/data3/iplanet4.1/https-sun07.chn.egurkha.com/logs/access"
format.access="%Ses->client.ip% -
%Req->vars.auth-user% [%SYSDATE%]    \"%Req->reqpb.clf-request%\"
%Req->srvhdrs.clf-status%                    %Req->srvhdrs.content-
length%"
Init fn=load-types mime-types=mime.types
Init      fn=load-modules
shlib="/opt/egurkha/lib/lib_eg_ssl.so" funcs=getRequest,getResponse
Init                                                fn="load-
modules"
shlib="/data3/iplanet4.1/bin/https/lib/libNSServletPlugin.so"
funcs="NSServletEarlyInit,NSServletLateInit,NSServletNameTrans,NSServ
letService"
shlib_flags="(global|now)"
Init fn="NSServletEarlyInit" EarlyInit=yes
Init fn="NSServletLateInit" LateInit=yes
<Object name=default>
NameTrans fn="NSServletNameTrans" name="servlet"
NameTrans fn="pfx2dir" from="/servlet"
dir="/data3/iplanet4.1/docs/servlet" name="ServletByExt"
NameTrans fn=pfx2dir from=/ns-icons dir="/data3/iplanet4.1/ns-icons"
name="es-internal"
```

```
PathCheck fn=getRequest
PathCheck                                                fn=unix-uri-
clean
PathCheck                                        fn="check-acl"
acl="default"
PathCheck                                                 fn=find-
pathinfo
PathCheck      fn=find-index          index-
names="index.html,home.html"
ObjectType                                               fn=type-by-
extension
ObjectType          fn=force-type
type=text/plain
Service  method=(GET|HEAD)     type=magnus-internal/imagemap
```

3.2 Administering the eG Manager to monitor the iPlanet/SunONE Web Server

To administer eG so that it monitors the iPlanet/SunONE web server, do the following:

1. Login as an administrator to the eG administrative interface
2. If the Netscape / iPlanet Enterprise Server is not discovered automatically, then either run discovery to get them discovered (Infrastructure ->Components -> Discover) or add them using the **ADD/MODIFY COMPONENTS** page (Infrastructure -> Components -> Add/Modify) (see Figure 3.1). Components manually added will be automatically managed by the eG Enterprise system (see Figure 3.1).

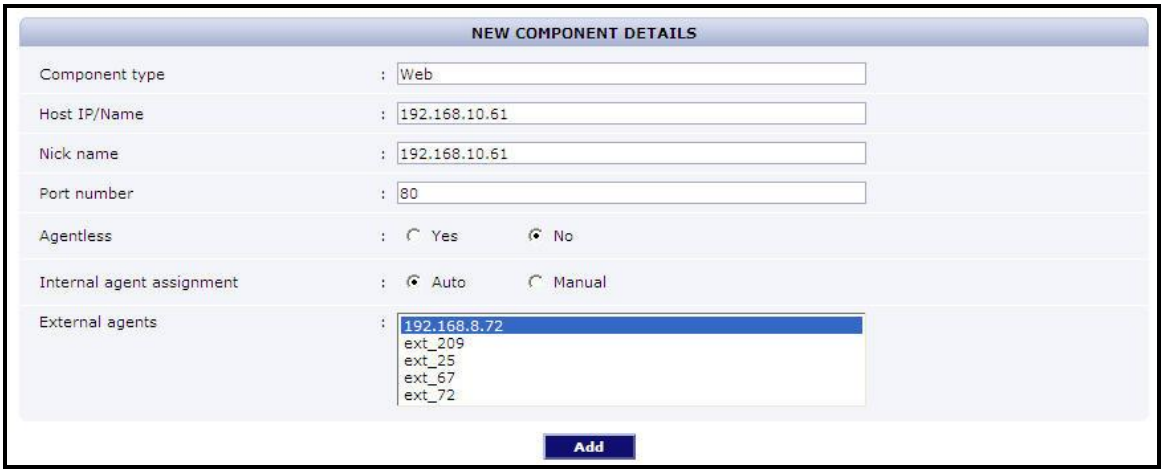

Figure 3.1: Adding an iPlanet/SunONE web server

3. On the other hand, if the iPlanet/SunONE web server is discovered automatically, proceed to manage them manually using the **COMPONENTS - MANAGE/UNMANAGE** page (Infrastructure -> Components -> Manage/Unmanage).
4. Then, proceed to configure the web sites and transactions.

Refer

Refer to Section 1.3 of Chapter 1 above for the procedure for configuring web sites and transactions

5. Now, when you try to sign out, the following screen will appear, prompting you to configure the Processes test.

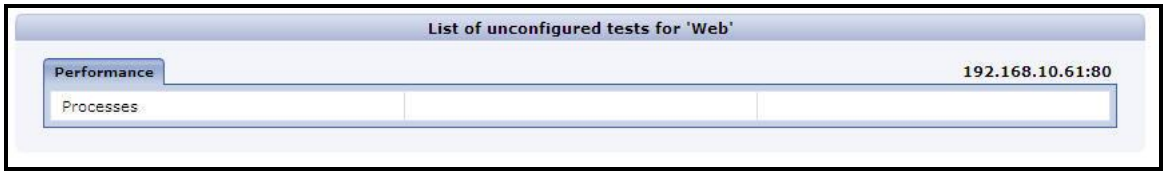

Figure 3.2: A page listing the unconfigured tests for the iPlanet/SunONE web server

6. Click on the test name in this page to configure it. While configuring, remember to provide the **PROCESSPATTERN**.

Refer

Refer to Section 1.3 of Chapter 1 above for details on configuring the Processes test and specifying the **PROCESSPATTERN**.

7. After configuring, sign out of the administrative interface.
8. Start the agent running on the system hosting the iPlanet/SunONE web server.

3.3 Monitoring the iPlanet/SunONE Server

1. Login as a monitor / supermonitor to the eG monitor user interface
2. Click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu.
3. Click on the iPlanet/SunONE Server to view its measurements.

Chapter 4

Configuring and Monitoring Oracle HTTP Servers

This chapter will discuss how to administer and monitor Oracle HTTP servers.

4.1 Administering the eG Manager to work with an Oracle HTTP Server

1. Ensure that the web adapter is configured.
2. Next, login to the administrative interface of eG as an administrator (admin).
3. Manually add the Oracle HTTP server to be monitored using the **ADD/MODIFY COMPONENTS** page (see Figure 4.1). To navigate to the **ADD/MODIFY COMPONENTS** page, follow the menu sequence: Infrastructure -> Components -> Add/Modify.

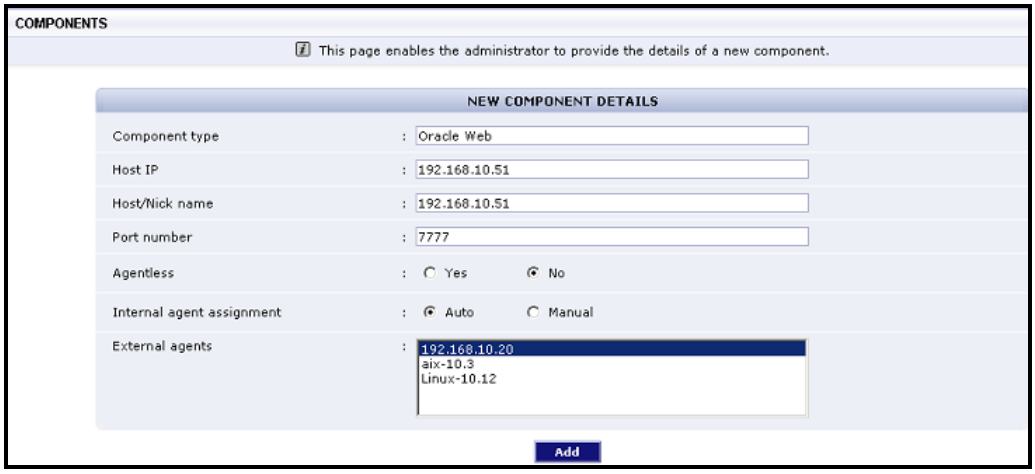

Figure 4.1: Adding the details of a new Oracle Http server

4. The Oracle HTTP Server so added will be managed automatically by eG. Now, try to sign out of the user interface. Doing so, will bring up a page, which prompts you to configure the tests for the Oracle Http server. Click on the OraHttpServer test in the page to configure it. Upon clicking, the test parameters will be displayed (see Figure 4.2).

Figure 4.2: Configuring the Oracle HTTP Server test

5. Specify the following in Figure 4.2:
   - **TEST PERIOD** - How often should the test be executed
   - **HOST -** The host for which the test is to be configured
   - **PORT -** The port to which the specified **HOST** listens
   - **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed
6. Click the **Update** button in Figure 4.2, and then, log out of the administrative interface.

4.2 Monitoring the Oracle HTTP Server

1. Login as a monitor / supermonitor user.
2. Click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu.
3. Click on the Oracle Http server being monitored to view its measurements .

4.3 Troubleshooting Oracle HTTP Server Monitoring

If all the tests associated with the Oracle HTTP server are in an **UNKNOWN** state, it could indicate any/all of the following:

- The eG agent is not running

  In such a case, start the eG agent by following the procedure described in the *eG Installation Manual*.

- The Oracle HTTP server is not running

  To verify this, open the **Services** window (Programs -> Administrative Tools -> Services), and view the **Status** of the **OracleHttpServer** service. If the **Status** column corresponding to the **OracleHttpServer** service is blank, it indicates that the service has not been started (see Figure 4.3). Therefore, start the service by right-clicking on the **OracleHttpServer** service and selecting the **Start** option from the shortcut menu (see Figure 4.4).

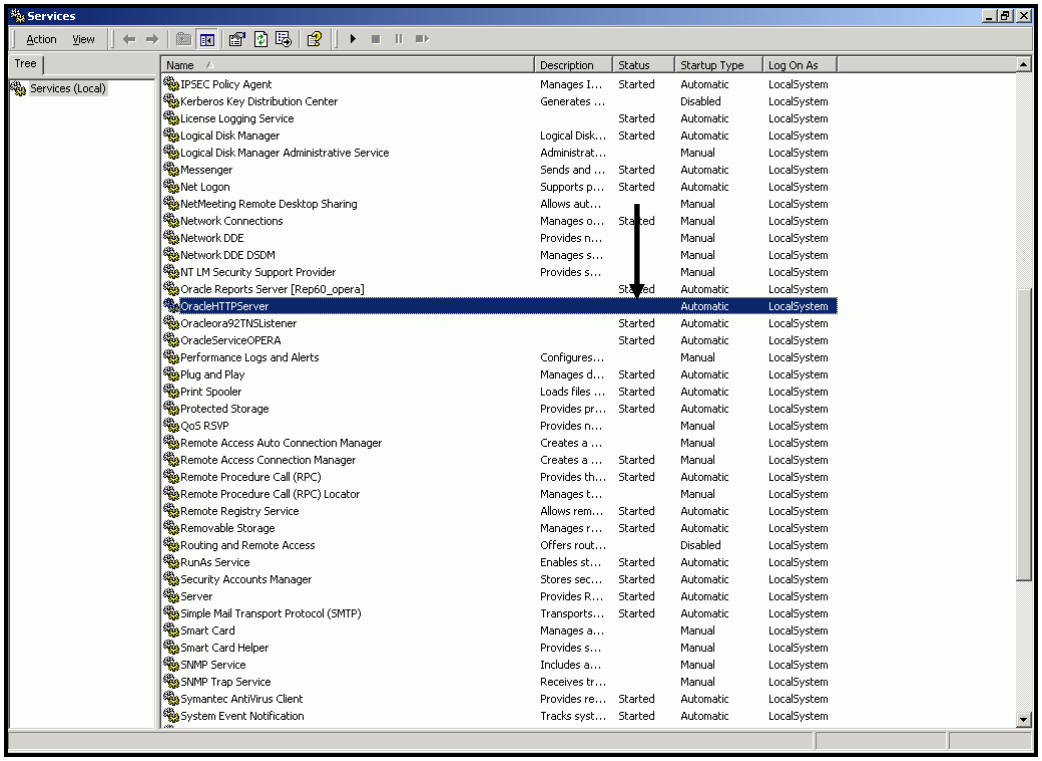

Figure 4.3: OracleHttpServer not started

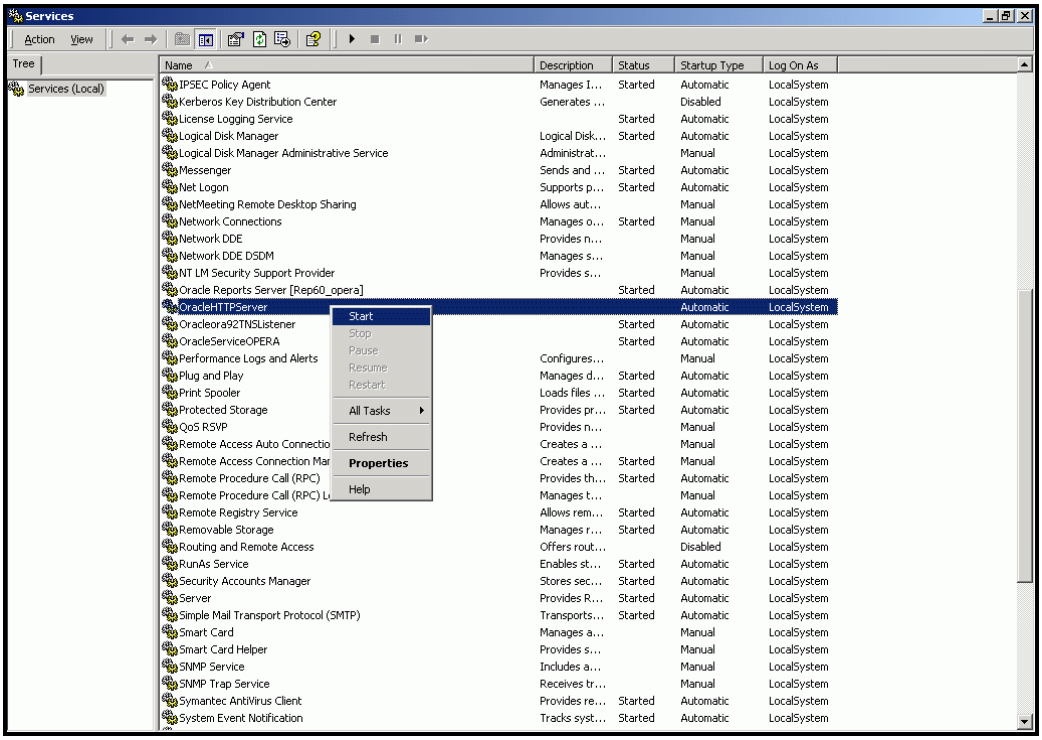

Figure 4.4: Starting the OracleHttpServer service

- If the service has already been started, then check whether the service is running in the local system account. If the entry in the **Log On As** column corresponding to the OracleHttpServer service is **LocalSystem**, it indicates that the service is running in the local system account (see Figure 4.5). If not, then modify the **Log On As** entry by first selecting the service, right-clicking on it, and selecting the **Properties** option from its shortcut menu (see Figure 4.6).

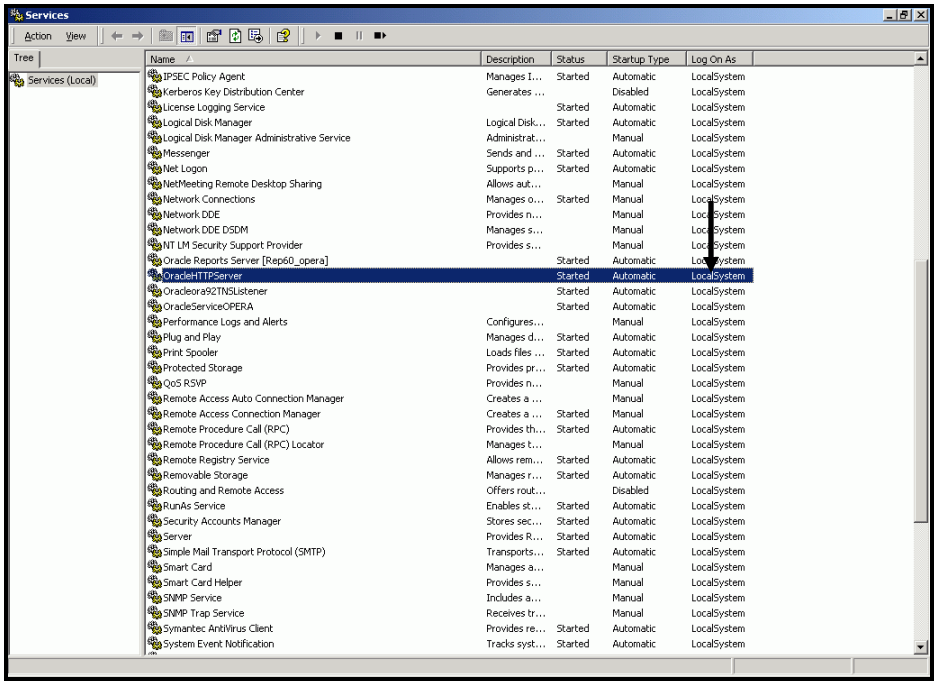

Figure 4.5: The OracleHttpServer service running in the LocalSystem account

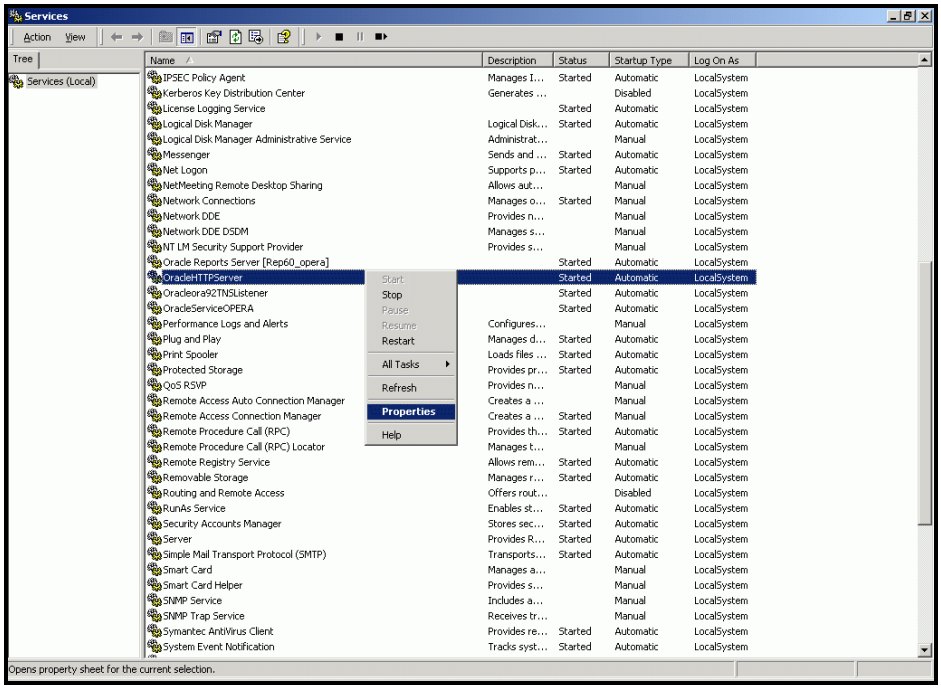

Figure 4.6: Selecting the Properties option

- Then, select the **Log On** tab from the **Properties** dialog box that appears, and choose the **Local System account** option from it (see Figure 4.7).

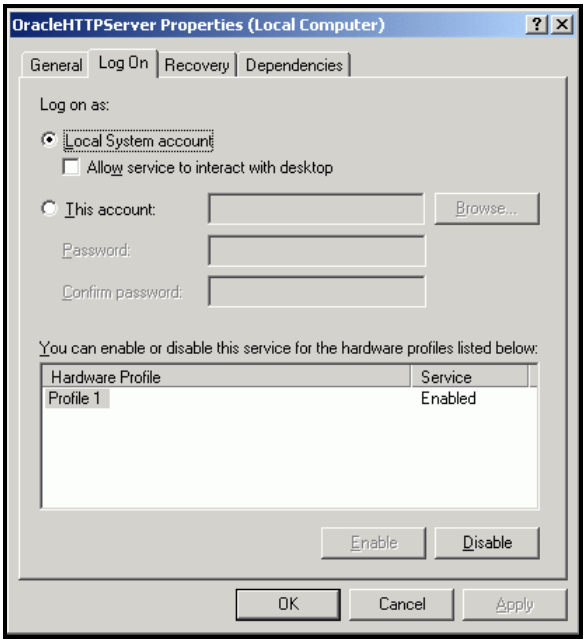

Figure 4.7: Selecting the Local System account

- Finally, click on the **Apply** button and then the **OK** button in Figure 4.7 to register the changes.
- If Oracle 9ias Release 1 is being used, then eG will be able to monitor the Oracle HTTP server running on it, only if it is managed as a **Web_server**. In such a case, if the OracleHttpServer service is running in

the Local system account only, proceed to check whether the web adapter has been configured properly. While configuring an Apache server, setup will request for the full path to the root directory of the server. Ensure that this path is the same as the value displayed against the **ServerRoot** parameter in the **httpd.conf** file in the **<ORA_HTTP_SERVER_HOME>\conf** directory (see Figure 4.8).

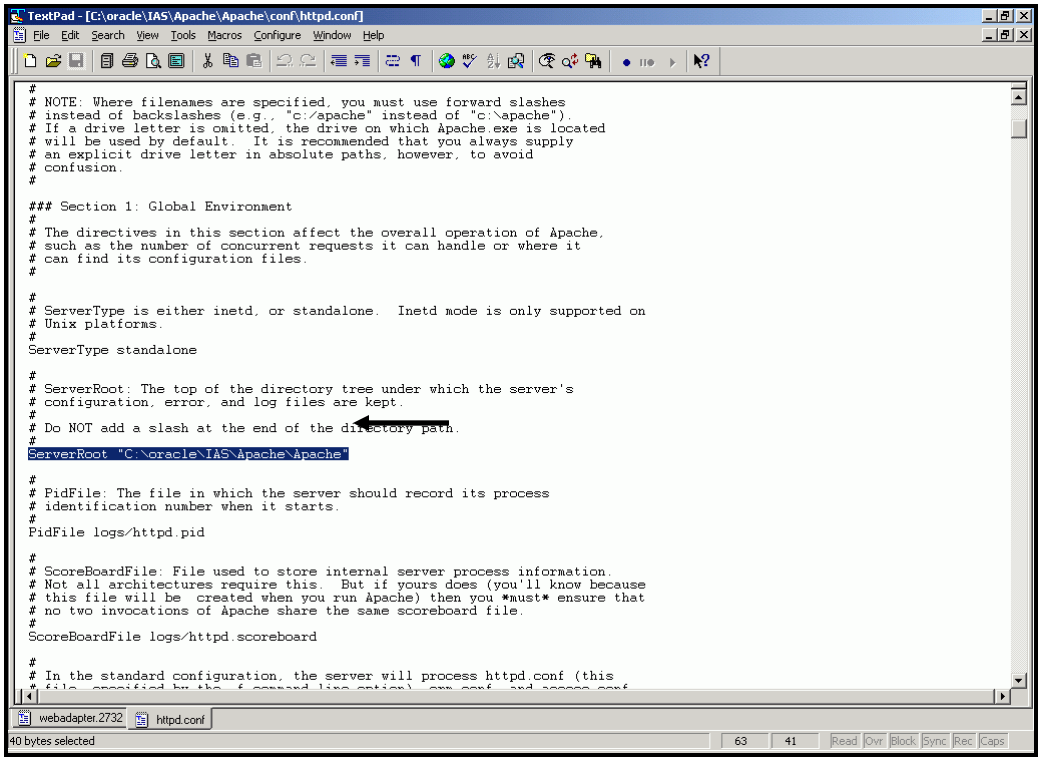

Figure 4.8: The ServerRoot parameter in the httpd.conf file

- Next, check whether a file named **webadapter.<PID>** is created in the **<eg_home_dir>\agent\config** directory. This is a clear indicator of the successful deployment of the web adapter. Now, verify whether the **pid** in **webadapter.<PID>** matches with the **pid** of any one of the **Apache.exe** processes in the **Windows Task Manager** (see Figure 4.9). If it does not match, then the web adapter may not work. Under such circumstances, delete the **webadapter.<PID>** file and restart the Oracle Http Server. Sometimes, an additional **webadapter** file will be created with a PID that does not match any of the **Apache.exe** PIDs listed in the **Windows Task Manager**. In such a case, delete the additional **webadapter.<PID>** file and restart the eG agent.

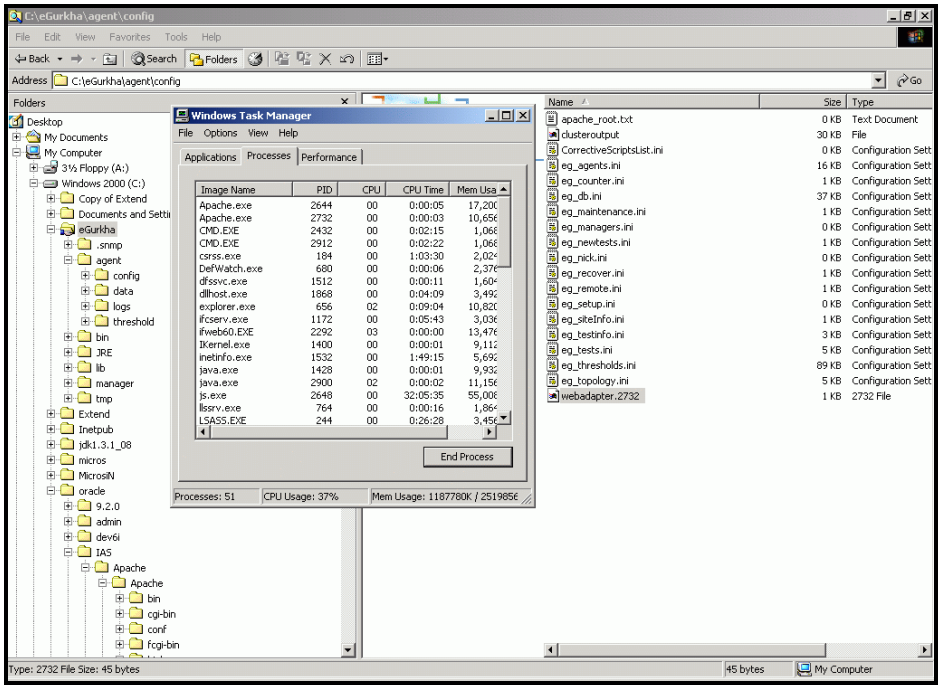

Figure 4.9: PID in the file name matching with the PID of one of the Apache.exe processes

- Also, ensure that the **Listen** ports configured in the **webadpater.<PID>** file (see Figure 4.10) are the same as those which are listed in the **httpd.conf** file in the **<ora_http_server_home>\conf** directory (see Figure 4.11).

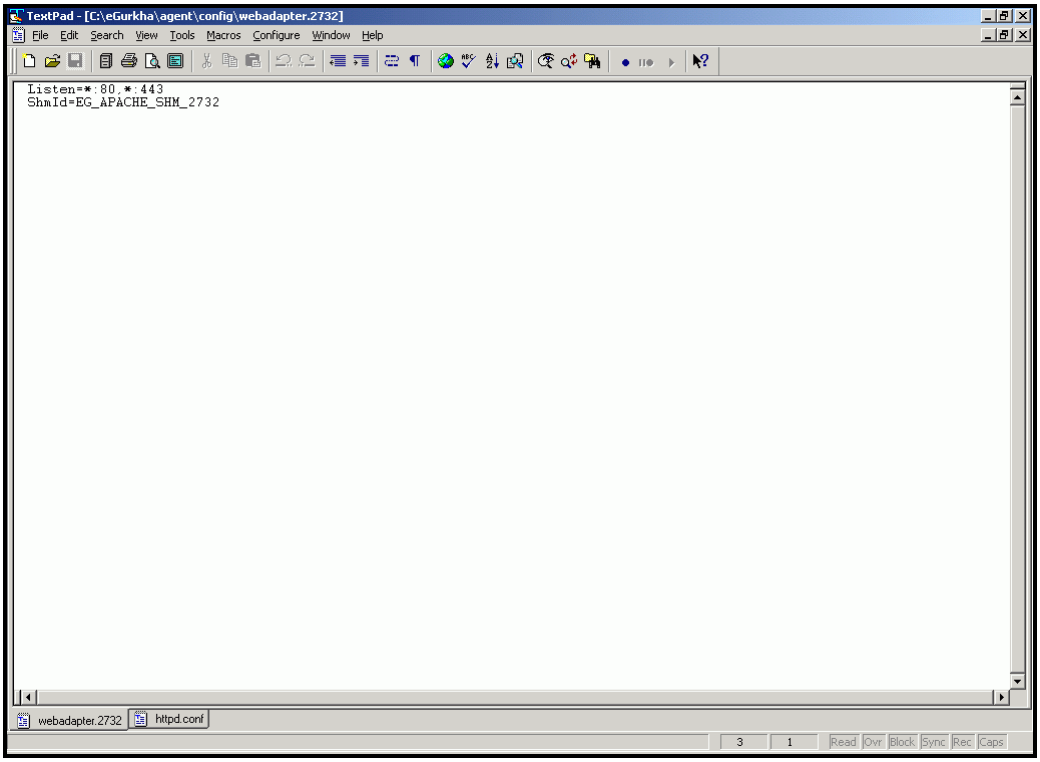

Figure 4.10: Listen ports displayed in the webadpater.<PID> file

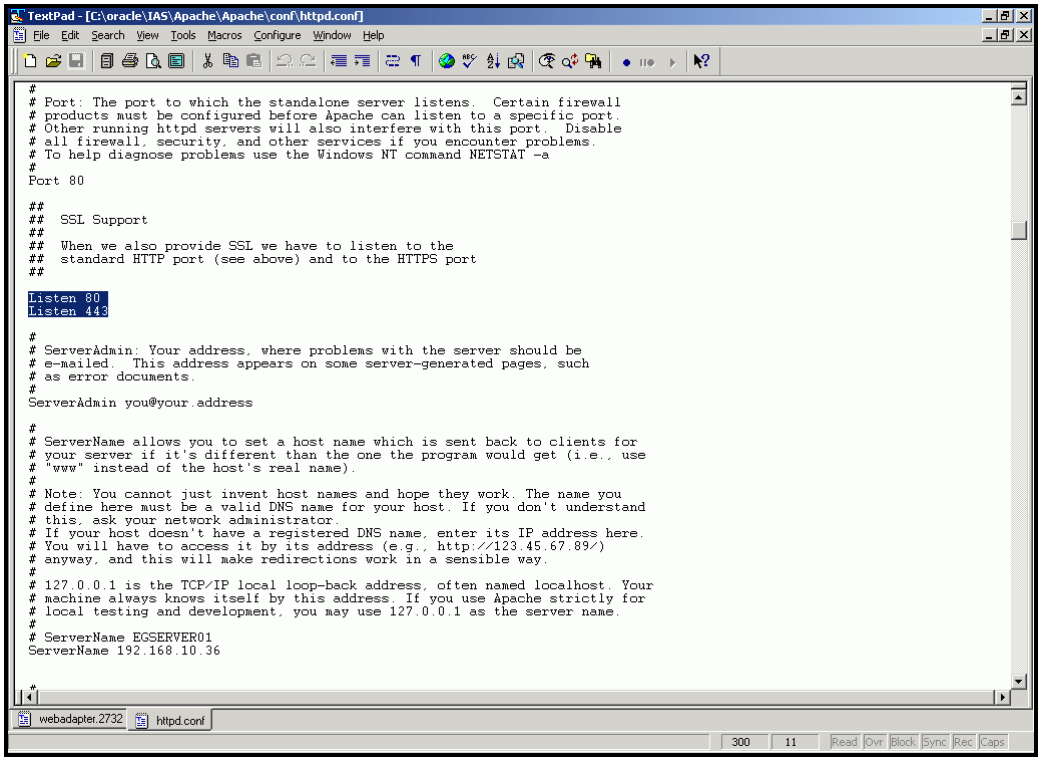

Figure 4.11: Listen ports displayed in the httpd.conf file

- Note that the **Listen** ports displayed in the **webadapter.<PID>** file are prefixed by a '*', which typically represents an IP address. However, if a specific IP address substitutes the '*' in the webadapter.<PID> file, then, in the eG administrative interface, the Oracle Http server must be managed using that IP address only.
- Finally, check whether the directives indicated by Figure 4.12 exist in the **httpd.conf** file in the **<ORA_HTTP_SERVER_HOME>\conf** directory.

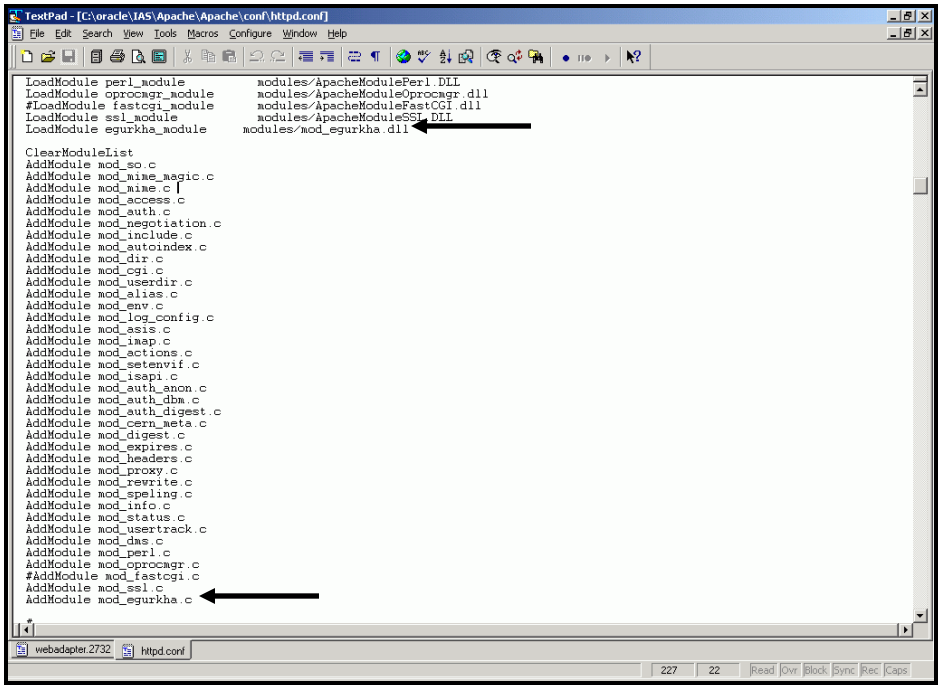

Figure 4.12: eG-specific directives in the httpd.conf file

Chapter 5

Configuring and Monitoring Apache Web Servers

The eG Enterprise suite provides a generic Web server component-type that allows administrators to effectively monitor any web server - be it an Apache web server, IBM HTTP server, or an iPlanet web server. The eG web adapter can be deployed on a Web server, so as to snoop on the real web transactions to the server and retrieve a wide range of real-time statistics pertaining to the critical transactions that are configured using the eG administrative interface.

On the other hand, some administrators might require in-depth insight into the performance of an Apache web server in particular, without enabling the web adapter capability. To cater to the requirements of such administrators, eG provides a specialized Apache_web_server model.

This section delineates the procedures involved in configuring and monitoring an Apache web server.

5.1 Configuring an Apache Web Server for Monitoring by the eG Enterprise suite

To pull out metrics related to the health of the Apache web server, the eG agent accesses a specific URL on the Apache server that contains the required metrics. To allow the eG agent to access this URL, you need to ensure that the following entries in the **<APACHE_HOME>\confi\httpd.conf** file are uncommented (or enabled).

LoadModule status_module modules/mod_status.so

<IfModule mod_status.c>

ExtendedStatus On

</IfModule>

<Location /server-status>

SetHandler server-status

Order deny,allow

Deny from all

Allow from ***<domain name to give access>***

</Location>

In case of the Apache web server v2.2 however, you will have to append the following entries to the **httpd.conf** file, soon after uncommenting the *LoadModule status_module modules/mod_status.so* entry:

<IfModule mod_status.c>

<Location /server-status>

SetHandler server-status

Order deny,allow

Deny from all

Allow from ***<domain name to give access>***

</Location>

</IfModule>

While uncommenting or inserting (depending upon the version of Apache being monitored) the aforesaid block, make sure that the **<domain name to give access>** is configured with the fully qualified domain name that should be permitted to access the URL on the Apache web server. Alternatively, you can even specify the IP address of a particular host that should be granted access, or a space-separated list of 'allowed' IP addresses. Since it is the eG agent that should be allowed access to the URL, specify the fully qualified name of the domain to which the eG agent belongs and/or the IP address of the eG agent in **<domain name to give access>**. For example, your entry can read as follows:

Allow from mas.eginnovations.com

(OR)

Allow from mas.eginnovations.com 192.168.8.56

This will be the local host's IP/host name in case of an internal agent, or the IP/host name of a remote Windows host in case of a remote agent.

5.2 Administering the eG Enterprise suite to Monitor an Apache Web Server

To achieve this, do the following:

1. Login to the administrative interface of eG as an administrator (admin).
2. If the Apache web server is automatically discovered, then use the **COMPONENTS -MANAGE/UNMANAGE** page to manage the server. On the other hand, if the Apache web server is not discovered automatically, then either run discovery to get them discovered (Infrastructure -> Components -> Discover) or add them using the **ADD/MODIFY COMPONENTS** page (Infrastructure -> Components -> Add/Modify) (see Figure 5.1). Components manually added will be automatically managed by the eG Enterprise system (see Figure 5.1).

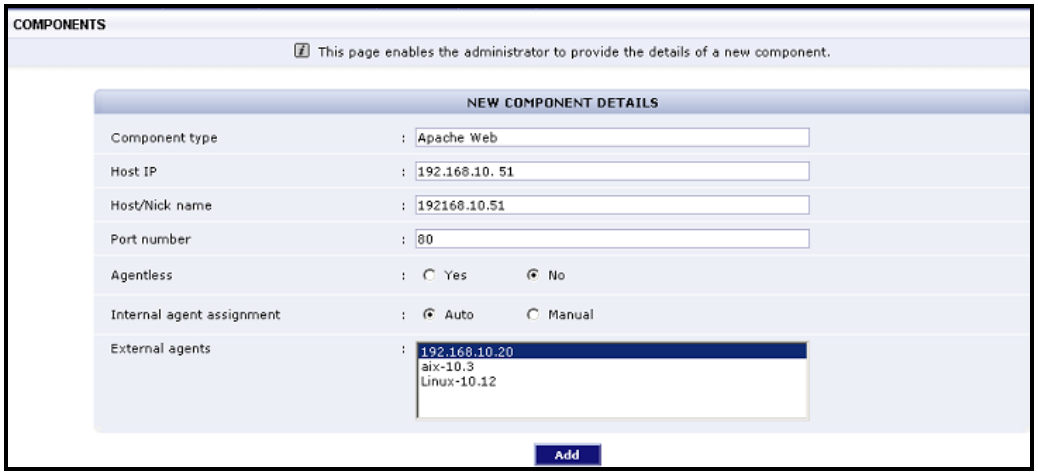

Figure 5.1: Adding the details of a new Apache web server

3. Now, try to sign out of the user interface. Doing so, will bring up a page, which prompts you to configure the tests for the Apache web server.

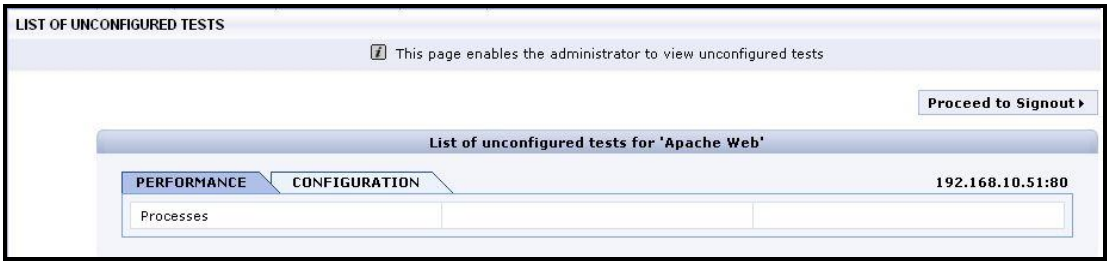

Figure 5.2: The list of unconfigured tests for the Apache web server

4. Click on the Processes test in Figure 5.2 to configure it. Upon clicking, the test parameters will be displayed (see Figure 5.3).

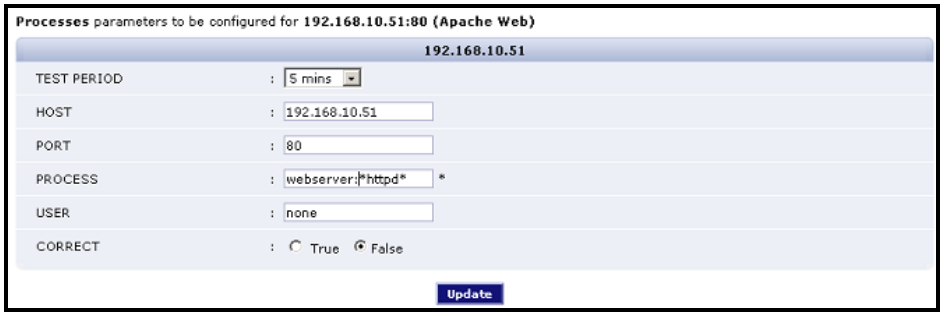

Figure 5.3: Configuring the Processes test

Refer to Section 1.3 of Chapter 1 above for details on configuring the Processes test and specifying the **PROCESSPATTERN**.

5. After configuring the Processes test, sign out of the eG admin interface.

5.3 Monitoring the Apache Web Server

To view the measurements of the *Apache Web* server, do the following:

1. Login as a monitor / supermonitor user.
2. Click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu.
3. Click on the *Apache Web* server being monitored to view its measurements .

Chapter 6

Configuring and Monitoring External Web Servers

There is no doubt that it is imperative to monitor the request processing ability of the web server, the amount of data load that a web server can handle, or the key transactions to a web site hosted by the server. In fact, eG Enterprise addresses these critical internal monitoring needs using dedicated web server monitoring models (*IIS Web*, *Apache Web*, *Web*, etc.), and its unique web adapter technology, all of which have been discussed previously. However, sometimes, administrators might only be interested in knowing whether the web server is available or not, and if so, how responsive it is to user requests. To cater to such monitoring needs, eG Enterprise offers the exclusive, *External Web* model. This model requires only an external agent, which employs native application-level protocols, to ascertain the availability and responsiveness of the web server. This section discusses the procedure involved in configuring and monitoring external web servers.

6.1 Administering the eG Manager to Work with an External Web Server

To administer eG so that it monitors the External web server, do the following:

1. Login as an administrator to the eG administrative interface
2. Next, add the external web server manually using the **ADD/MODIFY COMPONENTS** page (Infrastructure -> Components -> Add/Modify) (see Figure 6.1). Components manually added will be automatically managed by the eG Enterprise system (see Figure 6.1).

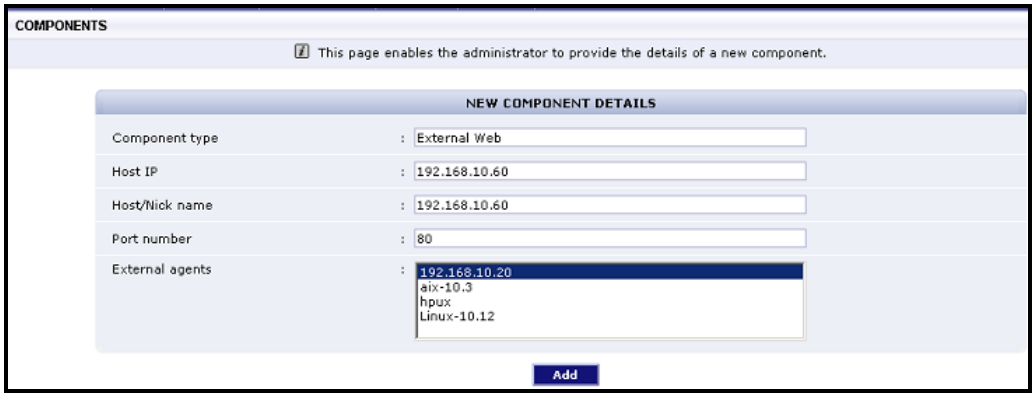

Figure 6.1: Adding an External web server

3. Now, sign out of the administrative interface.

6.2 Monitoring the External Web Server

1. Login as a monitor / supermonitor to the eG monitor user interface.
2. Click on the **Components** option in the menu bar, and select the **Servers** option from the **Components** menu.
3. Click on the *External Web* server to view its measurements .

Chapter 7

Troubleshooting Web Servers

- If the Http test and Network test are reporting current measures, but none of the other tests are reporting measures, make sure that the internal agent for the web server is running.
- If only the Http test and Network test are not reporting any measures, make sure that the external agent for the web server is running. By default, an external agent executes on the same host as the eG manager.
- If the WebServer test, WebSite test and WebSiteTransaction test are not running, check the following:
    - Did you remember to configure the target web server with the eG web adapter?
    - Did you restart the web server after configuring?
    - Is the web adapter running? On Unix systems, check for **config/webadapter.*** files. If they do not exist, it indicates that the webadapter is not running.
    - Verify whether sufficient shared memory segments exist
    - Check the log files
- If the Processes test is not reporting measures, check whether you have configured the web server's process via the admin interface.
- If the Http test is showing 0, while actually, the web server is running, then it means that the parameters of the Http test have not been adequately configured. In such a case, follow the steps provided in Section 1.3 to configure the parameters of this test.

We recognize that the success of any product depends on its ability to address real customer needs, and are eager to hear from you regarding requests for enhancements to the products, suggestions for modifications to the product, and feedback regarding what works and what does not. Please provide all your inputs as well as any bug reports via email to support@eginnovations.com.

Chapter 8

Conclusion

This document has described in detail the steps for configuring and monitoring the **Web Servers**. For details of how to administer and use the eG Enterprise suite of products, refer to the user manuals.

We will be adding new measurement capabilities into the future versions of the eG Enterprise suite. If you can identify new capabilities that you would like us to incorporate in the eG Enterprise suite of products, please contact support@eginnovations.com. We look forward to your support and cooperation. Any feedback regarding this manual or any other aspects of the eG Enterprise suite can be forwarded to feedback@eginnovations.com.