Monitoring Oracle Web Server

eG Innovations Product Documentation

www.eginnovations.com

Total Performance Visibility

Table of Contents

CHAPTER 1: INTRODUCTION 1

CHAPTER 2: ADMINISTERING EG MANAGER TO WORK WITH ORACLE WEB SERVER 2

CHAPTER 3: MONITORING ORACLE WEB SERVERS 4

3.1 The Oracle Web Server Layer 5

3.1.1 Oracle HTTP Server Test 5

3.1.2 Oracle HTTP Response Test 7

3.2 The Oracle Mod PL/SQL Layer 8

3.2.1 Oracle Content Cache Test 9

3.2.2 Oracle Session Cache Test 10

3.2.3 Oracle PL/SQL Response Test 11

CHAPTER 4: TROUBLESHOOTING 13

ABOUT EG INNOVATIONS 22

Table of Figures

Figure 2.1: Adding the details of a new Oracle web server 2
Figure 2.2: List of tests to be configured for the Oracle Web server 3
Figure 3.1: The layer model of an Oracle Web server 4
Figure 3.2: The tests associated with the Oracle Web Server layer 5
Figure 3.3: Tests associated with the Oracle Mod PL/SQL layer 9
Figure 4.1: OracleHTTPServer not started 13
Figure 4.2: Starting the OracleHTTPServer service 14
Figure 4.3: Figure 1.5: The OracleHTTPServer service running in the LocalSystem account 15
Figure 4.4: Figure 1.6: Selecting the Properties option 15
Figure 4.5: Selecting the Local System account 16
Figure 4.6: The ServerRoot parameter in the HTTPd.conf file 17
Figure 4.7: PID in the file name matching with the PID of one of the Apache.exe processes 18
Figure 4.8: Listen ports displayed in the webadpater.<PID> file 19
Figure 4.9: Listen ports displayed in the HTTPd.conf file 20
Figure 4.10: eG-specific directives in the HTTPd.conf file 21

Chapter 1: Introduction

Oracle HTTP Server (OHS) is a web server that provides the necessary infrastructure to support dynamic applications. Based on the Apache infrastructure, Oracle HTTP Server allows developers to program their site in a variety of languages and technologies.

In environments where the OHS is used to support critical end-user services, it is imperative to ensure that the OHS is continuously functioning to peak capacity, and is available 24x7 to serve client requests. Intermittent web server failures and even slight dips in the responsiveness of the server, could significantly delay, and sometimes even halt, service delivery, causing the business to lose millions. To avoid such adversities, the internal health and availability of the OHS needs to be constantly monitored. The eG Enterprise helps administrators in this task.

Chapter 2: Administering eG Manager to work with Oracle Web 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 web server to be monitored using the **COMPONENTS** page (see Figure 2.1). To navigate to the **COMPONENTS** page, follow the menu sequence: Infrastructure -> Components -> Add/Modify.

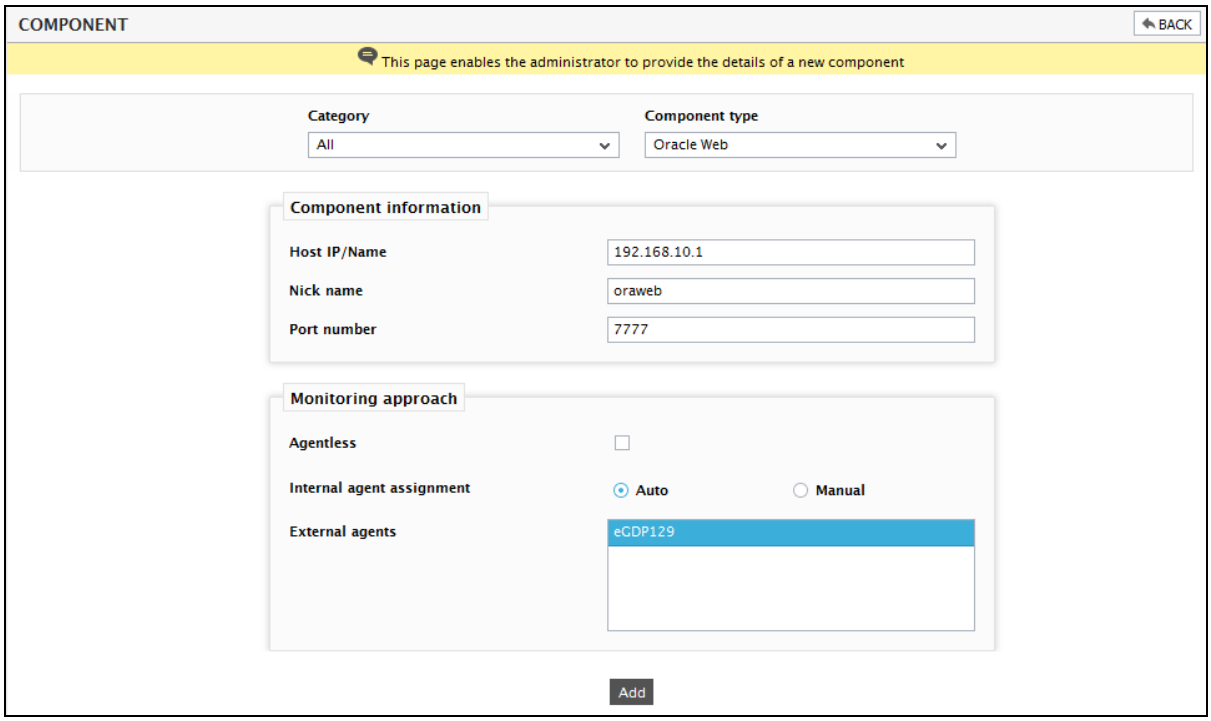

Figure 2.1: Adding the details of a new Oracle web server

4. Specify the **Host IP/Name** and **Nick name** of the Oracle Web server as shown in Figure 2.1. Then click the **Add** button to register the changes. The Oracle web server so added will be managed automatically by eG.
5. When you try to sign out of the eG administrative interface, a list of unconfigured tests appears (see)

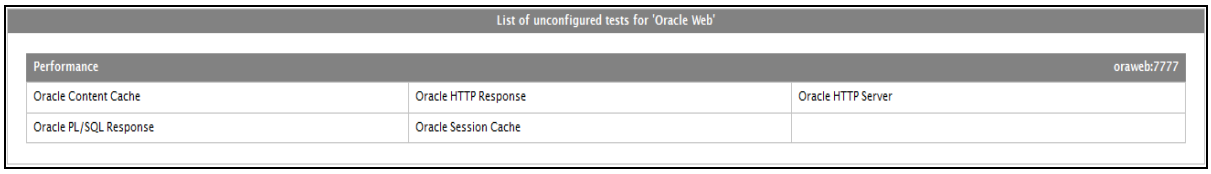

Figure 2.2: List of tests to be configured for the Oracle Web server

6. Now, click on the **Oracle Content Cache** test to configure it. To now how to configure the test, refer to the Section **3.2.1** . The others tests of the Oracle Web server will be configured automatically.
7. Finally, signout of the eG administrative interface.

Chapter 3: Monitoring Oracle Web Servers

eG Enterprise provides a specialized monitoring model for the Oracle Web server (see Figure 3.1), which runs periodic health checks on the server, determines its current state, detects performance issues (if any), and proactively alerts administrators to these issues, so that they can hasten the remedial measures.

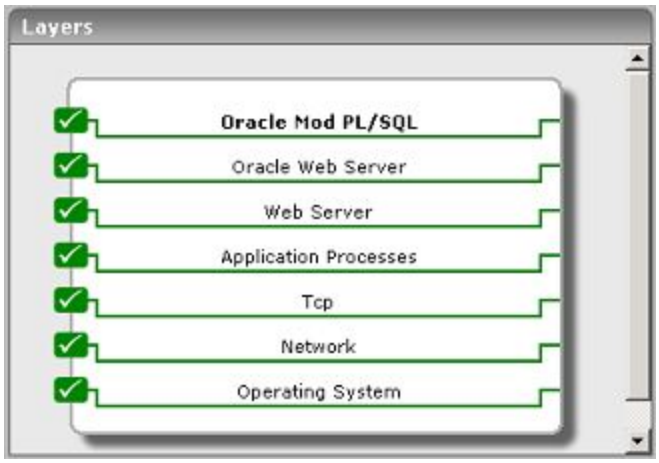

Figure 3.1: The layer model of an Oracle Web server

This model constitutes a set of hierarchical layers, which are mapped to a wide variety of tests. When the eG agent on the Oracle HTTP server executes these tests, it pulls out a wealth of performance information from the server. In the light of the statistics so retrieved, administrators can find answers to the following performance queries:

- Is the web server responding quickly to user requests?
- What is the current load on the server in terms of data traffic?
- Do errors occur on the web server? If so, how frequently do they occur, and what are the corresponding error codes?
- Are the PL/SQL Gateway's session and content caches utilized effectively, or are there too many cache misses?
- How well does the PL/SQL gateway respond to user requests? Are there too many errored responses?

Note:

Any component on an Oracle 9i Application server (be it OC4J or the Oracle Web server) can be monitored by eG only if the Oracle 9i Application Server Release 2.0 is installed, with *dmctl* and *dmstool*.

The sections to come deal only with the top 2 layers of Figure 3.1, as all other layers have been dealt with extensively in the *Monitoring IIS Web servers* document .

3.1 The Oracle Web Server Layer

Besides measuring the responsiveness of the web server, this layer also reveals the internal health of the server by reporting on its request processing ability, the data traffic to the server, etc.

Figure 3.2: The tests associated with the Oracle Web Server layer

3.1.1 Oracle HTTP Server Test

This test reports the performance statistics pertaining to an Oracle Web server.

Target of the test : An Oracle Web server

Agent deploying the test : An internal Agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port to which the specified **HOST** listens.
4. **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Response time:	Indicates the time taken by the web server to respond to requests. Only requests for which successful responses are received are considered while computing the response time.	Secs	A very high value for this measure indicates a bottleneck on the web server or one of the applications on the server.
Requests:	Indicates the rate of requests to the web server.	Reqs/Sec	This value is indicative of the server workload.
Active connections:	Indicates the number of connections currently open to the server.	Number	A high value can indicate a slow-down of the web server or applications hosted by it.
Connection rate:	Indicates the rate of connections to the web server during the last measurement period.	Conns/Sec	An increase or decrease in connection rate can represent a change in the user workload.
Data transmit rate:	Indicates the rate at which the data was transmitted by the server to clients during the last measurement period.	KB/Sec	A large increase in the data transmission rate can be indicative of an increase in the popularity of one or more web sites hosted on the server.
Errors:	Indicates the rate at which errors occurred on the server during the last measurement period.	Errors/Sec	The details of the type of errors occurring are reported by the test OracleHttpResponse test.
POST requests:	Indicates the rate of post requests to the	Reqs/Sec	

Measurement	Description	Measurement Unit	Interpretation
	web server.		
GET requests:	Indicates the rate of get requests to the web server.	Reqs/Sec	

3.1.2 Oracle HTTP Response Test

This test measures the request-handling capability of the Oracle9iAS HTTP server.

Target of the test : An Oracle Web server

Agent deploying the test : An internal Agent

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

Configurable parameters for the test

1. **TEST PERIOD -** How often should the test be executed
2. **HOST -** The host for which the test is to be configured.
3. **PORT -** The port to which the specified **HOST** listens.
4. **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
200 responses:	Indicates the percentage of requests that were successfully serviced by the web server during the last measurement period.	Percent	
Errors:	Indicates the percentage of responses with errors	Percent	Responses with a 400 or 500 status code are counted as errors.

Measurement	Description	Measurement Unit	Interpretation
	during the last measurement period.		
300 responses:	Indicates the percentage of responses in the last measurement period with a status code in the range of 300-399.	Percent	300 responses could indicate page caching on the client browsers or redirection of requests. A sudden change in this value is indicative of a problem condition.
400 errors:	Indicates the percentage of responses in the last measurement period with a status code in the range of 400-499.	Percent	A high value indicates a number of missing/error pages.
500 errors:	Indicates the percentage of responses in the last measurement period with a status code in the range of 500-600.	Percent	Since responses with a status code of 500-600 indicate server side processing errors, a high value reflects an error condition.

3.2 The Oracle Mod PL/SQL Layer

The PL/SQL Gateway provides support for building and deploying PL/SQL-based applications on the web. PL/SQL stored procedures can retrieve data from database tables and generate HTTP responses containing data and code to display in a Web browser.

This layer assesses how effectively the gateway uses the content and session caches while processing requests, and keeps track of the responses the gateway sends out to a web browser.

Figure 3.3: Tests associated with the Oracle Mod PL/SQL layer

3.2.1 Oracle Content Cache Test

To improve performance of PL/SQL web applications, you can leverage the caching feature provided by the PL/SQL Gateway. This feature allows caching of PL/SQL procedure Web content in the middle-tier. Subsequent requests for the content may be retrieved from the cache, with or without validation from the database, thereby decreasing the database workload.

This test reports the performance metrics pertaining to the mod_plsql gateway's content cache.

Target of the test : An Oracle Web server

Agent deploying the test : An internal Agent

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

Configurable parameters for the test

1. **TEST PERIOD** - How often should the test be executed
2. **HOST** - The host for which the test is to be configured.
3. **PORT** - The port to which the specified **HOST** listens.
4. **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Request rate:	Indicates the rate of requests to the content cache during the last measurement period.	Reqs/Sec	A very high value for this measure indicates that the content cache is handling many requests.
Cache hits:	Indicates the rate of content cache hits.	Conns/Sec	Reading from the cache is less expensive than reading directly from the database. Therefore, the higher this value, the better. Remember that caching increases the scalability of a web application.
Cache misses:	Indicates the rate of content cache misses	Conns/Sec	

3.2.2 Oracle Session Cache Test

This test reports the performance metrics pertaining to the mod_plsql gateway's session cache.

Target of the test : An Oracle Web server

Agent deploying the test : An internal Agent

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

Configurable parameters for the test

1. **TEST PERIOD -** How often should the test be executed
2. **HOST -** The host for which the test is to be configured.
3. **PORT -** The port to which the specified **HOST** listens.
4. **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Request rate:	Indicates the rate of requests to the session cache during the last measurement period.	Reqs/Sec	A very high value for this measure indicates that the session cache is handling many requests.
Cache hits:	Indicates the rate of content cache hits.	Conns/Sec	Reading from the cache is less expensive than reading from the database. Therefore, the higher this value, the better. Remember that caching increases the scalability of a web application.
Cache misses:	Indicates the rate of content cache misses.	Conns/Sec	

3.2.3 Oracle PL/SQL Response Test

This test reports how well the mod_plsql gateway of the Oracle Web server responds to user requests.

Target of the test : An Oracle Web server

Agent deploying the test : An internal Agent

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

Configurable parameters for the test

1. **TEST PERIOD -** How often should the test be executed
2. **HOST -** The host for which the test is to be configured.
3. **PORT -** The port to which the specified **HOST** listens.
4. **HOMEDIR** – The path to the directory in which the Oracle 9i application server has been installed

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
200 responses:	Indicates the percentage of requests that were successfully serviced by the web server during the last measurement period.	Percent	
Errors:	Indicates the percentage of responses with errors during the last measurement period.	Percent	Responses with a 400 or 500 status code are counted as errors.
300 responses:	Indicates the percentage of responses in the last measurement period with a status code in the range of 300-399.	Percent	300 responses could indicate page caching on the client browsers or redirection of requests. A sudden change in this value is indicative of a problem condition.
400 errors:	Indicates the percentage of responses in the last measurement period with a status code in the range of 400-499.	Percent	A high value indicates a number of missing/error pages.
500 errors:	Indicates the percentage of responses in the last measurement period with a status code in the range of 500-600.	Percent	Since responses with a status code of 500-600 indicate server side processing errors, a high value reflects an error condition.

Chapter 4: Troubleshooting

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

1. The eG agent is not running

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

2. The Oracle web 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 4.1). Therefore, start the service by right-clicking on the **OracleHTTPServer** service and selecting the **Start** option from the shortcut menu (see Figure 4.2).

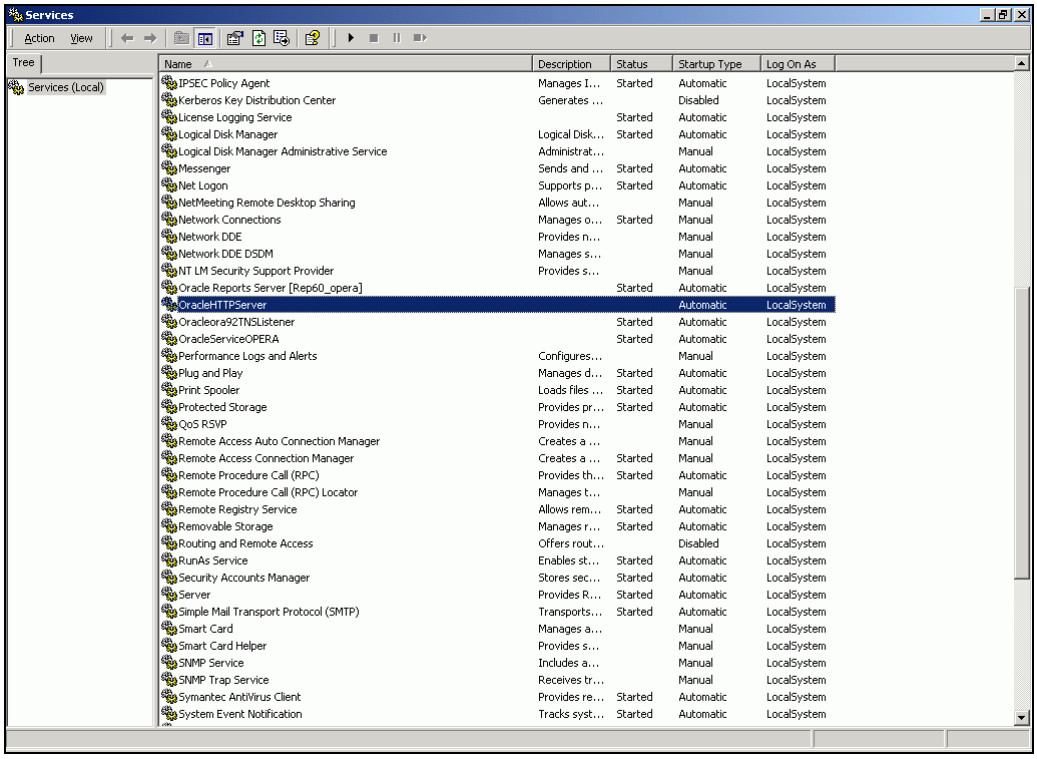

Figure 4.1: OracleHTTPServer not started

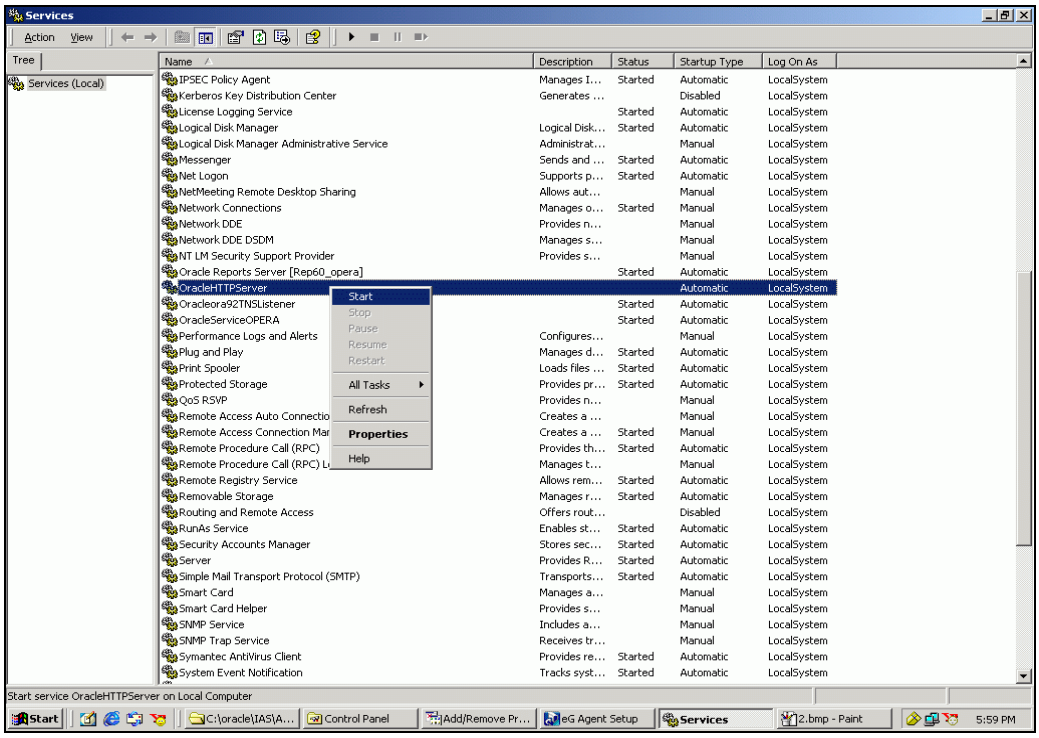

Figure 4.2: Starting the OracleHTTPServer service

3. 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 4.3). 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 4.4).

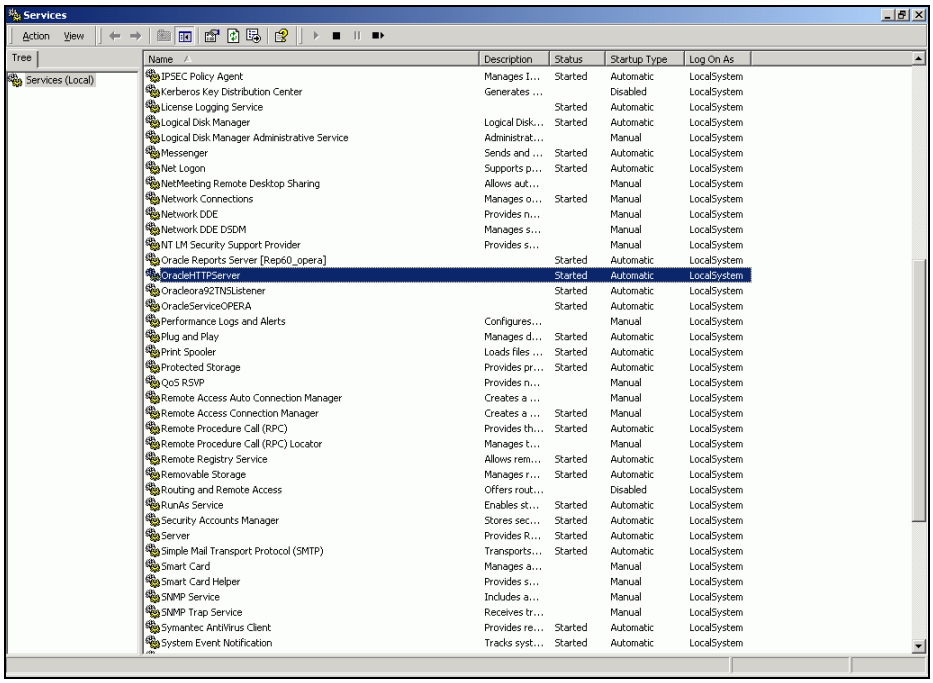

Figure 4.3: Figure 1.5: The OracleHTTPServer service running in the LocalSystem account

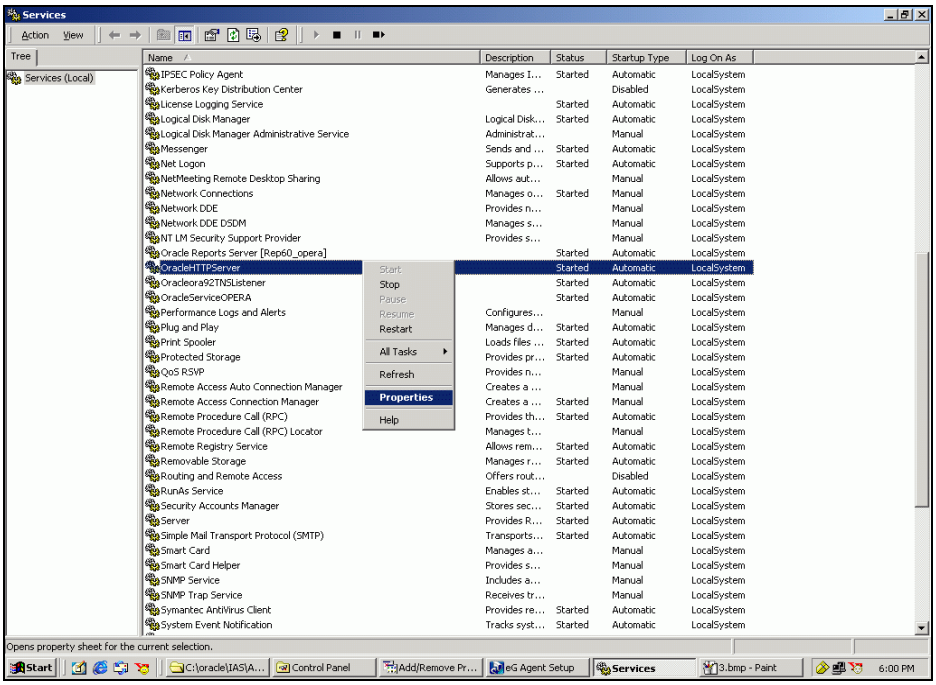

Figure 4.4: Figure 1.6: Selecting the Properties option

4. Then, select the **Log On** tab from the **Properties** dialog box that appears, and choose the **Local**

System account option from it (see Figure 4.5).

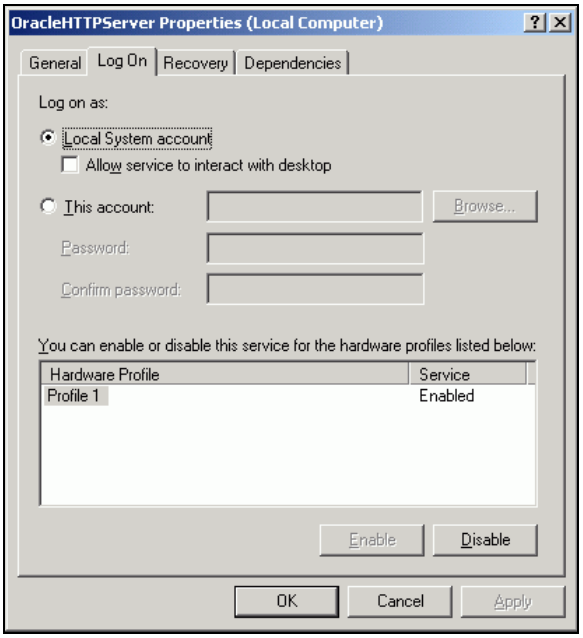

Figure 4.5: Selecting the Local System account

5. Finally, click on the **Apply** button and then the **OK** button in Figure 4.5 to register the changes.
6. If Oracle 9ias Release 1 is being used, then eG will be able to monitor the Oracle web 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 4.6).

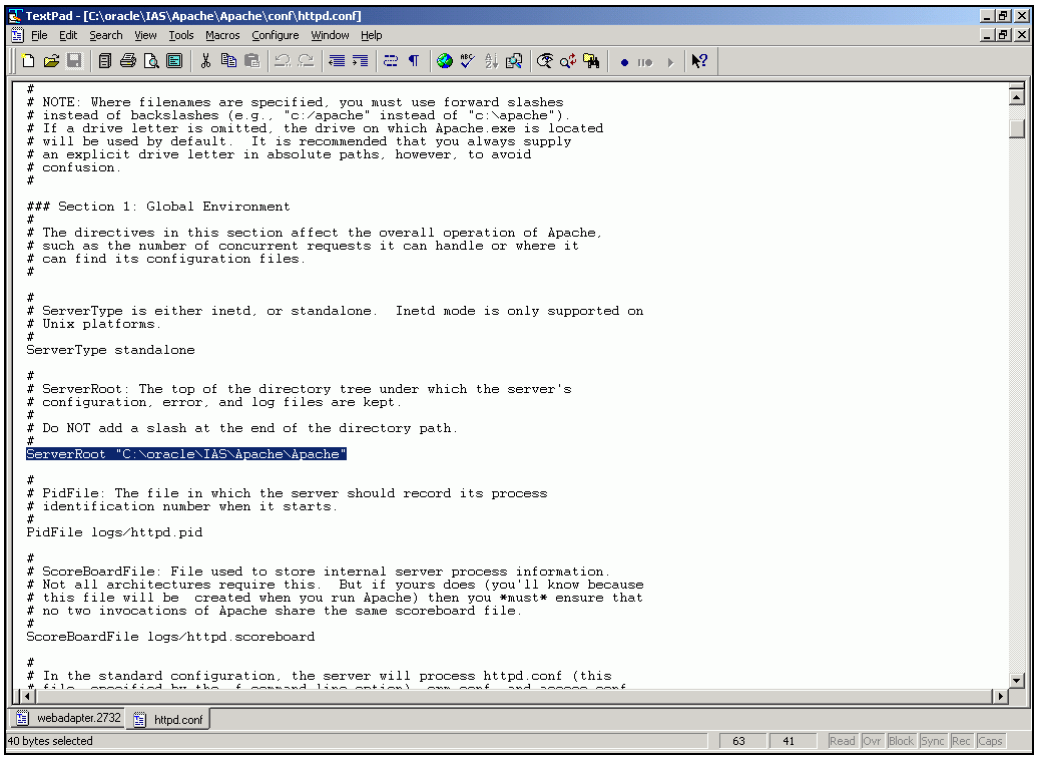

Figure 4.6: The ServerRoot parameter in the HTTPd.conf file

7. 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 4.7). If it does not match, then the web adapter may not work. Under such circumstances, delete the **webadapter.<PID>** file and restart the Oracle web 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.

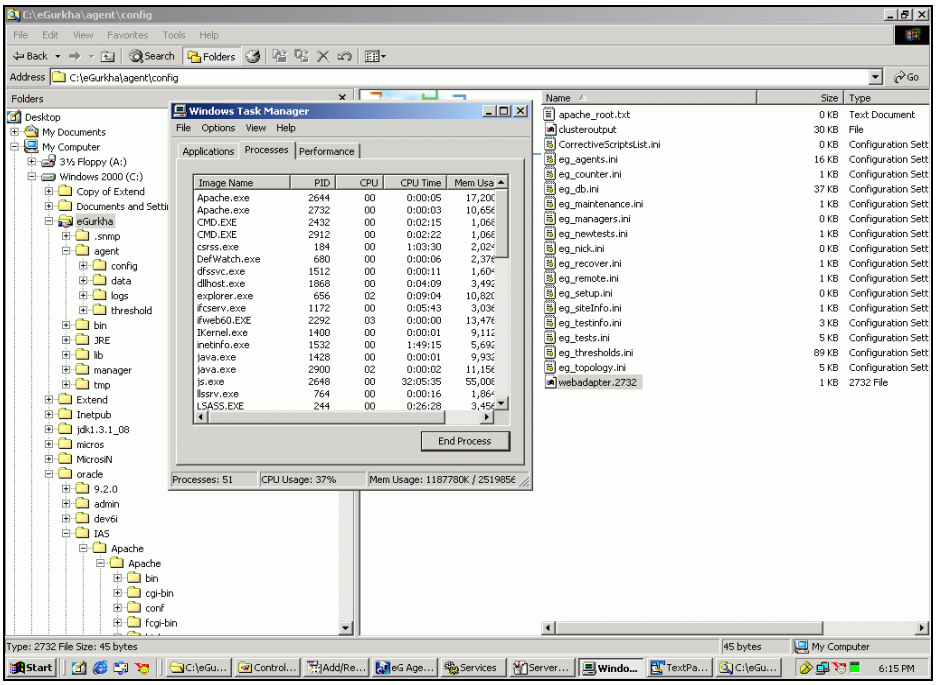

Figure 4.7: PID in the file name matching with the PID of one of the Apache.exe processes

8. Also, ensure that the **Listen** ports configured in the **webadpater.<PID>** file (see Figure 4.8) are the same as those which are listed in the **HTTPd.conf** file in the **<ora_HTTP_server_home>\conf** directory (see Figure 4.9).

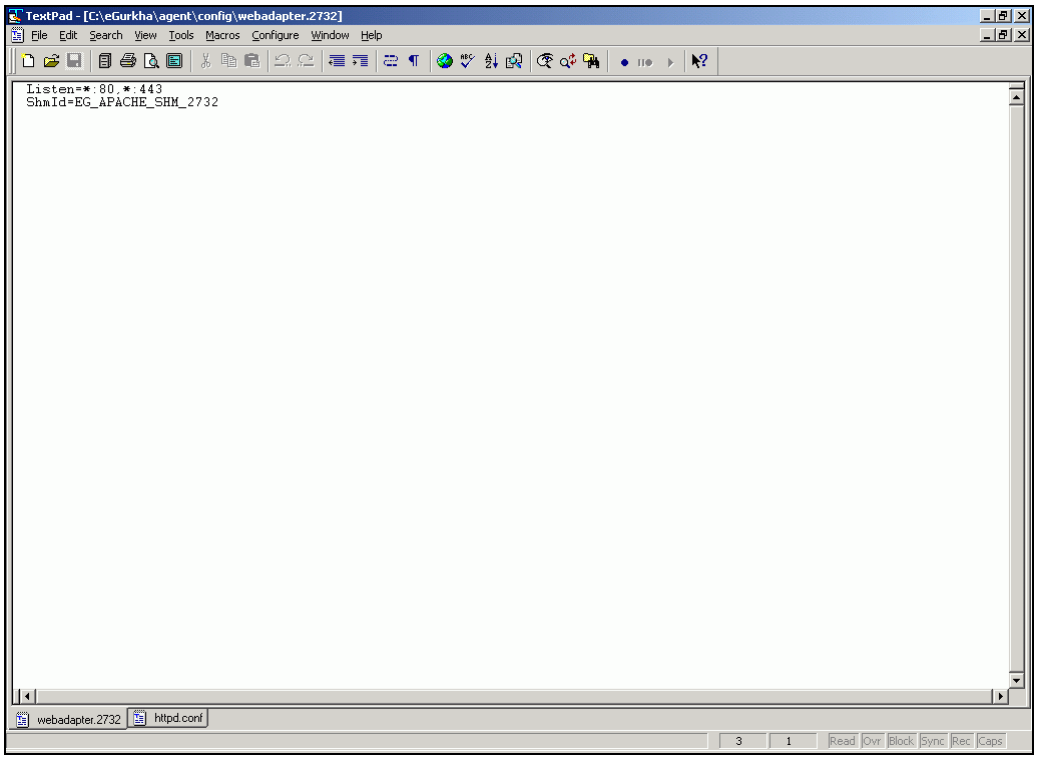

Figure 4.8: Listen ports displayed in the webadpater.<PID> file

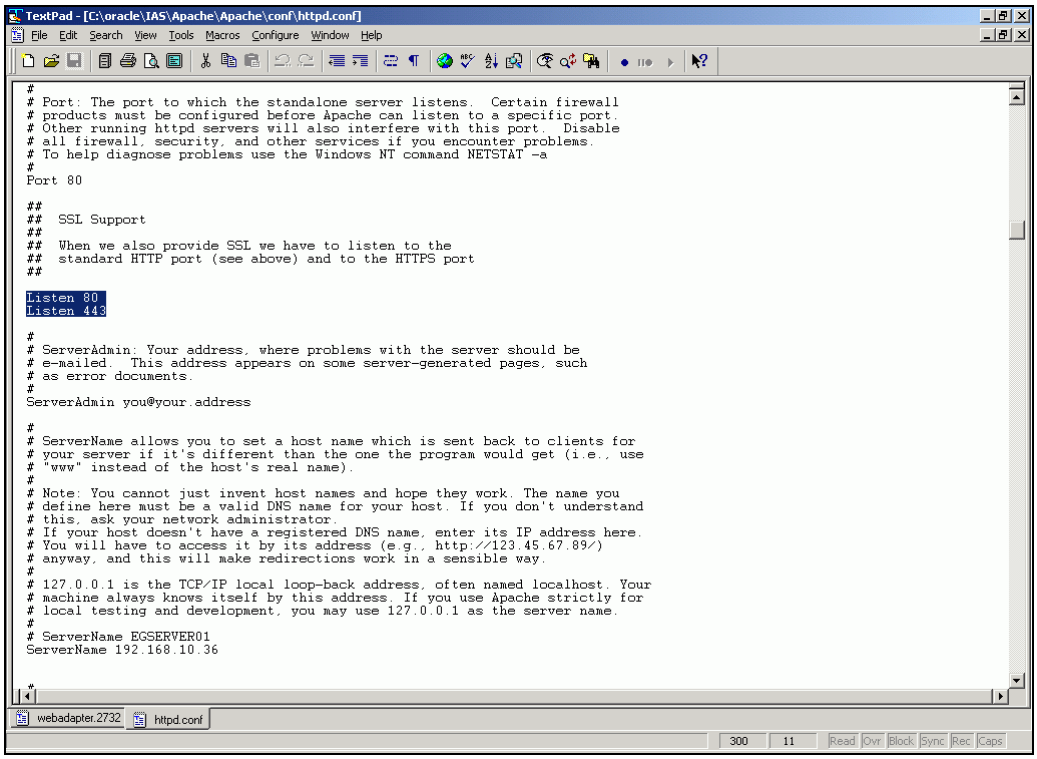

Figure 4.9: Listen ports displayed in the HTTPd.conf file

9. 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 web server must be managed using that IP address only.

10. Finally, check whether the directives indicated by Figure 4.10 exist in the **HTTPd.conf** file in the **<ORA_HTTP_SERVER_HOME>\conf** directory.

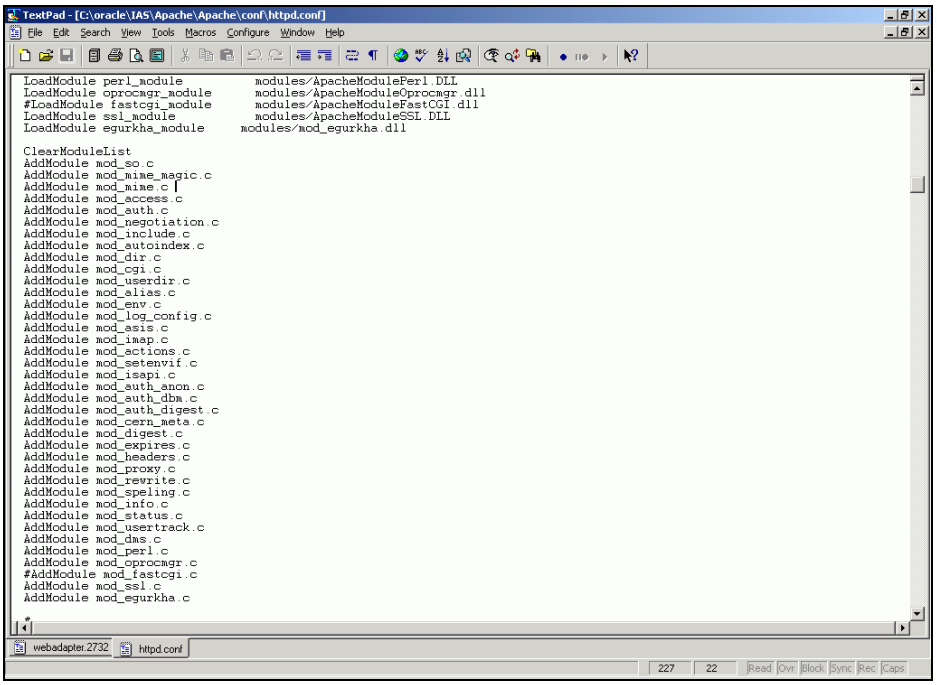

Figure 4.10: eG-specific directives in the HTTPd.conf file

About eG Innovations

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

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

To learn more visit www.eginnovations.com.

Contact Us

For support queries, email support@eginnovations.com.

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

Copyright © 2018 eG Innovations Inc. All rights reserved.

This document may not be reproduced by any means nor modified, decompiled, disassembled, published or distributed, in whole or in part, or translated to any electronic medium or other means without the prior written consent of eG Innovations. eG Innovations 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. The information contained in this document is subject to change without notice.

All right, title, and interest in and to the software and documentation are and shall remain the exclusive property of eG Innovations. All trademarks, marked and not marked, are the property of their respective owners. Specifications subject to change without notice.