**Oracle® iPlanet Web Server 7.0.9
Troubleshooting Guide**

ORACLE®

Contents

Preface

This *Troubleshooting Guide* provides troubleshooting tips to debug common administration issues in Oracle iPlanet Web Server (Web Server).

The following topics are addressed here:

- "Who Should Use This Book" on page 7
- "Web Server Documentation Set" on page 7
- "Documentation Conventions" on page 9
- "Default Paths and File Names" on page 11
- "Documentation, Support, and Training" on page 12
- "Searching Oracle Product Documentation" on page 12
- "Third-Party Web Site References" on page 12

Who Should Use This Book

This book is intended for Web Server administrators to troubleshoot the server in production environments. This guide assumes you are familiar with the following topics:

- Installing software
- Using web browsers
- Performing basic system administration tasks
- Issuing commands in a terminal window

Web Server Documentation Set

The Web Server documentation set describes how to install and administer the Web Server. The URL for Web Server documentation is http://docs.sun.com/coll/1653.9.

For an introduction to Web Server, refer to the books in the order in which they are listed in the following table.

TABLE P–1 Web Server Documentation

Document Title	Contents
Documentation Center	Web Server documentation topics organized by tasks and subject
Release Notes	- Late-breaking information about the software and documentation - Supported platforms and patch requirements for installing Web Server
Installation and Migration Guide	Performing installation and migration tasks: - Installing Web Server and its various components - Migrating data from Sun ONE Web Server 6.0 or Sun Java System Web Server 6.1 to Oracle iPlanet Web Server 7.0
Administrator's Guide	Performing the following administration tasks: - Using the Administration and command-line interfaces - Configuring server preferences - Using server instances - Monitoring and logging server activity - Using certificates and public key cryptography to secure the server - Configuring access control to secure the server - Using Java Platform, Enterprise Edition (Java EE) security features - Deploying applications - Managing virtual servers - Defining server workload and sizing the system to meet performance needs - Searching the contents and attributes of server documents, and creating a text search interface - Configuring the server for content compression - Configuring the server for web publishing and content authoring using WebDAV
Developer's Guide	Using programming technologies and APIs to do the following: - Extend and modify Web Server - Dynamically generate content in response to client requests and modify the content of the server
NSAPI Developer's Guide	Creating custom Netscape Server Application Programmer's Interface (NSAPI) plug-ins
Developer's Guide to Java Web Applications	Implementing Java Servlets and JavaServer Pages (JSP) technology in Web Server

TABLE P–1 Web Server Documentation *(Continued)*

Document Title	Contents
Administrator's Configuration File Reference	Editing configuration files
Performance Tuning, Sizing, and Scaling Guide	Tuning Web Server to optimize performance
Troubleshooting Guide	Troubleshooting Web Server
CLI Reference Manual	Administration commands that allow you to administer the Web Server through the CLI

Documentation Conventions

This section describes the following conventions used in Web Server documentation:

- "Typographic Conventions" on page 9
- "Symbol Conventions" on page 10
- "Shell Prompts in Command Examples" on page 10

Typographic Conventions

The following table describes the typographic changes that are used in this book.

TABLE P–2 Typographic Conventions

Typeface	Meaning	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your .login file.
		Use ls -a to list all files.
		machine_name% you have mail.
AaBbCc123	What you type, contrasted with onscreen computer output	machine_name% **su**
		Password:
AaBbCc123	A placeholder to be replaced with a real name or value	The command to remove a file is rm *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized (note that some emphasized items appear bold online)	Read Chapter 6 in the *User's Guide*.
		A *cache* is a copy that is stored locally.
		Do *not* save the file.

Symbol Conventions

The following table explains symbols that might be used in this book.

TABLE P–3 Symbol Conventions

Symbol	Description	Example	Meaning
[ ]	Contains optional arguments and command options.	ls [-l]	The -l option is not required.
{ | }	Contains a set of choices for a required command option.	-d {y|n}	The -d option requires that you use either the y argument or the n argument.
${ }	Indicates a variable reference.	${com.sun.javaRoot}	References the value of the com.sun.javaRoot variable.
-	Joins simultaneous multiple keystrokes.	Control-A	Press the Control key while you press the A key.
+	Joins consecutive multiple keystrokes.	Ctrl+A+N	Press the Control key, release it, and then press the subsequent keys.
→	Indicates menu item selection in a graphical user interface.	File → New → Templates	From the File menu, choose New. From the New submenu, choose Templates.

Shell Prompts in Command Examples

The following table shows default system prompts and superuser prompts.

TABLE P–4 Shell Prompts

Shell	Prompt
C shell on UNIX and Linux systems	machine_name%
C shell superuser on UNIX and Linux systems	machine_name#
Bourne shell and Korn shell on UNIX and Linux systems	$
Bourne shell and Korn shell superuser on UNIX and Linux systems	#

Default Paths and File Names

The following tables describe the default paths and file names used in Web Server documentation.

TABLE P–5 Default Paths and File Names

Placeholder	Description	Default Value
install-dir	Represents the base installation directory for Web Server.	Installation as the `root` user on the Solaris, Linux, AIX, and HP-UX platforms: `/opt/oracle/webserver7` Installation as a non-root user on the Solaris, Linux, AIX, and HP-UX platforms: *user-home-dir*`/oracle/webserver7` All installations on the Windows platform: *SystemDrive*`:\Program Files\Oracle\Webserver7`
instance-dir	Directory that contains the instance-specific subdirectories.	For all installations, the same as *install-dir*.

The following table shows the default paths and file names for Web Server when installed as a component of Sun Java Enterprise System (Java ES).

TABLE P–6 Default Paths and File Names for Java ES Installations

Placeholder	Description	Default Value

install-dir	Represents the base installation directory for Web Server.	On the Solaris platform:
		
		`/opt/SUNWwbsvr7`
		
		On the Linux and HP-UX platforms:
		
		`/opt/sun/webserver7`
		
		On the Windows platform:
		
		SystemDrive`:\Program Files\Sun\JavaES5\WebServer7`

TABLE P–6 Default Paths and File Names for Java ES Installations *(Continued)*

Placeholder	Description	Default Value
instance-dir	Directory that contains the instance-specific subdirectories.	On the Solaris platform:
		`/var/opt/SUNWwbsvr7`
		On the Linux and HP-UX platforms:
		`/var/opt/sun/webserver7`
		On the Windows platform:
		SystemDrive:`\Program Files\Sun\JavaES5\WebServer7`

Documentation, Support, and Training

The Oracle web site provides information about the following additional resources:

- Documentation (`http://docs.sun.com/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://education.oracle.com/`)

Searching Oracle Product Documentation

Besides searching Oracle product documentation from the docs.sun.com web site, you can use a search engine by typing the following syntax in the search field:

search-term `site:docs.sun.com`

For example, to search for "proxy," type the following:

`proxy site:docs.sun.com`

To include other Oracle web sites in your search (for example, java.sun.com, www.sun.com, and developers.sun.com), use sun.com in place of `docs.sun.com` in the search field.

Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Oracle is not responsible for the availability of third-party web sites mentioned in this document. Oracle does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Oracle will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

1

Overview

This chapter provides a description of the tools, methods, and information sources available for troubleshooting the Oracle iPlanet Web Server 7.0 Update 8. Guidelines for evaluating and investigating a problem are included.

Planning Ahead

As applications get deployed, un-deployed, and redeployed, and as you experiment with different server configuration settings, there may be times when your server gets into an unstable state. In such cases, it is useful to have a previously saved working configuration on which to fall back. This is not problem solving, but rather a way to avoid problems in the first place.

Refer to the *Web Server Administrator's Guide* for complete instructions on using the CLI and GUI options. Briefly, however, for the purposes of this *Troubleshooting Guide*, use the following procedure to backup the virtual servers in your configuration:

▼ To back up your virtual server

1 Go to Common Tasks page.

The Common Tasks page is the home page when you access the administration console. Fore information on accessing the administration console, refer to the Administrator's Guide.

2 Select the Configuration.

From the Configuration Tasks page, select the configuration from the drop down box.

3   Copy Virtual Servers.

Select the virtual server from the list and click Copy button. A window will pop up. Provide the new virtual server name and click OK. The web applications also gets copied.

Identifying the Problem

Web Servers are typically deployed in complex and highly sophisticated operating environments. Oracle iPlanet Web Server covers a broad range of technologies, including Java, Java Servlets, XML, JSP, JDBC data sources, and more. Understanding and diagnosing complex issues involving so many disparate components requires thorough knowledge and a careful diagnostic process.

Gathering any or all of the following information will make it easier to classify a problem and search for solutions. Note that operating system utilities, such as `pkginfo` and `showrev` on Solaris and `rpm` on Linux, are helpful in gathering system information.

▼ Verifying Server Platform

1   What are the exact version numbers of the operating system and products installed?

2   Have any patches been applied? If so, specify product and operating system patch numbers.

3   How is the system configured?

  a.   What system resources does the system have (memory, disk, swap space, and so on)?

  b.   How many web servers and directory servers are installed?

  c.   How is the Web Server connected to the directory server?

  d.   Are web servers in a cluster or not?

  e.   Was any upgrade done? If so, what were source and target versions?

  f.   Was a migration done? If so, what were source and target versions?

4   Have any new web applications been deployed?

5   Is SSL enabled or not?

6   What database is being used?

7   What JDBC driver is being used to access the database?

8   What JDK version is being used?

9   What are the JVM heap, stack, and garbage collection-related parameters set to?

10   What are the JVM options?

11   What third-party technologies are being used in the installation?

12   Are the interoperating component versions in compliance with the compatibility matrix specified in the release notes?

After gathering this information:

- Collect web server error and access log data (web server instance-specific).

- Collect any Web Server stack traces. Note that a fresh set of logs associated with the specific problem should be run. This avoids scanning gigabytes of irrelevant log information.

- Determine the sequence of events that occurred when the problem first appeared, including any steps that may already have been taken to resolve the problem.

Note – When you encounter a problem, do not panic. It is better to approach the problem more systematically by collecting the necessary system specific details.

Seeking a Solution

After identifying the problem, you are ready to do some investigation.

The following topics are addressed in this section:

- "Verify System Configuration" on page 18
- "Evaluate Messages" on page 18
- "Examine Log Files" on page 19
- "See if the Problem has been Solved Before" on page 20
- "Search the Product Documentation" on page 20
- "Search the Knowledge Base" on page 20
- "Search or Participate in the Online Forum" on page 21
- "Contact Support" on page 21

Verify System Configuration

Sometimes the most obvious solutions are overlooked, and so the first step is to verify the system configuration. Refer to the *Oracle iPlanet Web Server 7.0.9 Release Notes* for the most up-to-date system requirements and dependencies.

Evaluate Messages

Messages generally include information about the attempted action, the outcome of the action, and, if applicable, the cause of jeopardy or failure.

Types of Messages

The log files contain the following general types of message entries:

- **Error** – These messages mark critical failures that cause status to be reported as Failed. Error messages generally provide detailed information about the nature and the cause of the problem that occurred.
- **Warning** – These messages mark non-critical failures. Warning messages generally contain information about the cause and the nature of the failure, and also provide possible remedies.
- **Information** – These messages mark normal completion of particular tasks.

Error Messages

A problem is often accompanied by an error message that prevents the application from proceeding.

- In some cases, the message is very clear about what is wrong and what needs to be done, if anything, to fix it. For example the 'Deployment Pending' link that appears on the administration console whenever an instance configuration is modified.

  In this case, the message gives clear guidance and the problem can be disregarded.

- Sometimes an error message gives only general information about the problem or solution, or suggests multiple possibilities.

  In this case, the problem is not obvious, or there might be multiple things wrong. You might have to consider various possibilities and perhaps a number of solutions. If the proposed fix is time consuming or costly, take steps to ensure that the fix is likely to be correct before actually doing anything.

- Some error messages are either not helpful or provide little guidance.

  In this case, there is very little information to go on. It is especially important to identify the exact situation that caused the error, and what the symptoms are before proceeding.

  For descriptions of all the Web Server error messages, refer to the later sections in this guide.

Examine Log Files

A number of the Web Server subsystems create log files and log their events to these files. The primary purpose of these log files is to provide troubleshooting information.

Note – Web Server Error Logs are the first place you should look for information, when you need to troubleshoot a runtime issue. For issues related to installation, see the installation log files.

In addition to the message text, a logged message provides the following information:

- Date and time of the event
- Log level for the event — Web Server-specified log level ID or name
- Process identifier (PID) — PID of the Web Server process
- (optional) Virtual server identifier (VSID) — VSID that generated the message
- Message identifier (MID) — subsystem and a four digit integer
- Message data

   The specific logs associated with each Web Server problem area are discussed in the associated chapters of this manual.

Log Levels

The Web Server has many log levels that can be set in the Admin Console (FINEST, FINER, FINE, CONFIG, INFO, WARNING, SEVERE, ALERT, and FATAL). All messages are logged when the log level is set to FINEST and only serious error messages appear if the log level is set to FATAL.

Note that the more detailed log levels (FINEST, FINER, FINE) can generate high volumes of log information for certain events, which may make it appear at first glance that there is an error condition when in fact there is not.

All messages with a log level less than the default level of INFO (FINEST, FINER, FINE, and CONFIG) provide information related to debugging and must be specifically enabled. Instructions for doing this are contained in the *Oracle iPlanet Web Server Administrator's Guide*.

In addition to the standard JDK log levels, the Web Server has added log levels designed to map more intuitively to the Web Server log file (server.log) and to tightly integrate with Solaris. The log levels ALERT and FATAL are specific to the Web Server and are not implemented in the JDK1.4 logging API.

> **Note** – For information on the event log mechanism used in the Microsoft Windows operating environment, refer to the Windows help system index using the keywords Event Logging. If you choose to send logs to the Windows `server.log` file, only messages with a log level of INFO, WARNING, SEVERE, ALERT, or FATAL are logged to the Windows Event Log.

Log Options

The Admin Console provides the following two logging options:

- **Option 1** — Log stdout (`System.out.print`) content to the event log
- **Option 2** — Log stderr (`System.err.print`) content to the event log

  When these options are set, stdout and stderr messages are written to the `server.log` file. (The event log is a `syslog` daemon on Solaris and Event Log on Microsoft Windows.)

  If the above options are not set:

- Anything written to stdout or stderr (that is, using `System.out` or `System.err`) will not appear in the logs.
- Messages logged with the JDK logger will appear in the logs.
- Messages written to stdout or stderr appear with the INFO level, but do not have a message ID.

See if the Problem has been Solved Before

A good initial step is to scan this Troubleshooting Guide to see if the problem is addressed here. If so, select the appropriate solution. Many of the solutions contain references to other documents in the Web Server document collection for additional details, explanations, or examples.

Search the Product Documentation

Start by reading the Release Notes for the version of the product you are troubleshooting.

The documentation for this Web Server product release is available at *Oracle iPlanet Web Server 7.0.9 Documentation Center*.

Search the Knowledge Base

The Knowledge Base is a collection of articles on product issues that provide information helpful for troubleshooting. To access the Knowledge Base:

▼ To search the Knowledge Base

1 Go to SunSolve (`http://sunsolve.sun.com/pub-cgi/show.pl?target=home`).

2 Under SunSolve Collections, click the Search Collections link.

3 Select the checkbox for the collection(s) to search.

4 Click Next.

5 Enter the search criteria.

6 Click Go.

Search or Participate in the Online Forum

Browse directly in any of the online forums, or log in and register to start posting messages. The Web Server online forum is available at: `http://forums.sun.com/forum.jspa?forumId=759`.

Contact Support

When necessary, gather together the information you have acquired and contact technical support at `http://www.sun.com/service/contacting`.

2

Installation

This chapter describes how to troubleshoot the most common installation, uninstallation, and migration issues.

This chapter contains the following sections:

- "Installation Problems" on page 23
- "Uninstallation Problems" on page 26
- "Migration Problems" on page 26

Installation Problems

As a critical first step, examine the installation log files.

The install log file is the `Oracle_iPlanet_Web_Server_install.log` file located in *install-dir*/setup/ directory. Install log file entries contain information about the attempted action, the outcome of the action, and the cause of failure if applicable. Entries fall into three categories: INFO, WARNING, and ERROR.

For more information on the installation log, see "Viewing Installation Log Files" in *Oracle iPlanet Web Server 7.0.9 Installation and Migration Guide*.

Note – Check if the DISPLAY environment variable is set properly on the machine used for installation. The DISPLAY variable needs to be set to the IP address of that machine, followed by a screen address, which can always be assumed to be :0.0. On Linux/Unix, execute `echo $DISPLAY` to find the value.

Failed Installation

If your installation fails, investigate the installation log.

Existing Installation Detected

The installer maintains an entry in the product registry. If installation files are removed incorrectly (for example, if the files are manually deleted instead of being removed by running the uninstaller), the files may be gone but the Web Server entries remain in the product registry. If you then try to re-install to that location, a message displays indicating that an existing installation has been detected, even though it appears that the installation does not exist. To remedy the situation, go to the product registry and delete the entries manually.

The location of the product registry differs for each platform.

Note – Again, the information in this section applies only if files were removed in some way other than by using the Web Server uninstaller.

Product Registry on Solaris

On Solaris, the product registry is in `/var/sadm/install/productregistry`.

In the `productregistry` file, remove the content between `<components>` and `</components>`, and you should then be able to install to the location. The following example shows a product registry file on Solaris.

EXAMPLE 2–1   Example Product Registry File

```
<productregistry>
    <version>7.0
    </version>
    <components>
        <compid>webcore
            <compversion>1.0
                <uniquename>webcore
                </uniquename>
                <compinstance>1
                    <comptype>COMPONENT
                    </comptype>
                    <location>/sun/gj/ws7
                    </location>
                </compinstance>
            </compversion>
        </compid>
        <compid>java
            <compversion>1.0
                <uniquename>java
                </uniquename>
                <compinstance>1
                    <comptype>COMPONENT
                    </comptype>
                    <location>/sun/gj/ws7
                    </location>
                </compinstance>
            </compversion>
```

EXAMPLE 2–1 Example Product Registry File   *(Continued)*

```
            </compid>
            <compid>snmp
                <compversion>1.0
                    <uniquename>snmp
                    </uniquename>
                    <compinstance>1
                        <comptype>COMPONENT
                        </comptype>
                        <location>/sun/gj/ws7
                        </location>
                    </compinstance>
                </compversion>
            </compid>
        </components>
</productregistry>
```

Product Registry on Linux

On Linux, the product registry is in `/var/opt/sun/install/productregistry`.

Product Registry on Windows

On Windows, the product registry is in `C:\WINNT\system32`.

The installation will fail if the Windows registry contains the `HKEY_LOCAL_MACHINE\SOFTWARE\Sun Microsystems\WebServer\7.0` entry. Remove this entry using Registry Editor, and you should then be able to install to the desired location.

Note – If you get an error related to the `https-admserv70` service, remove the following entry from the registry and try installing again:
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\https-admserv70`.

Defaults to Command-Line Mode

On UNIX systems, the installer and uninstaller default to command-line mode if the `DISPLAY` environment variable has not been exported to your local machine and if X Windows is not supported.

Uninstallation Problems

As a critical first step, examine the uninstallation log file.

Migration Problems

As a critical first step, examine the migration log.

◆ ◆ ◆   **C H A P T E R   3**

3

Startup and Access

My Server Fails to Start

When your server fails to start, consider the following cases:

1. If you have configured Web Server to run on port 80, then you will need to start the server as 'root' user on Unix/Linux. However in Solaris 10, you don't need to run the Server as root to bind to port 80 (or < 1024). Execute the following commands:

   ```
   # su
   # /usr/sbin/usermod -K defaultpriv=basic,net_privaddr webservd
   ```

2. When you encounter server startup issue, Server's error log or console output (on UNIX/Linux platforms) should most likely contain the reason for the startup failure.

3. If the Web Server is configured to run in 64-bit and any of the plug-ins mentioned in the magnus.conf is of 32-bit, then Web Server 7.0 would fail to startup , throwing out error message like wrong ELF class: ELFCLASS32. Similarly, if you see an error message like wrong ELF class: ELFCLASS64 it means that 32-bit Web Server is trying to load a 64 bit NSAPI plugin or vice-versa.

4. While starting up the Web Server, if you do not see the message info: CORE3274: successful server startup on UNIX/Linux platforms, then it is most likely that the server has startup issues.

5. If you see the error message catastrophe (908): Server crash detected (signal SIGSEGV) in the server's error log file, this means that Web Server's daemon has detected a crash. Web Server crash during startup can happen because of various reasons including:

- Any of the configured 3rd party NSAPI plug-ins is either not following NSAPI specification.
- Improper Server Configuration.

6. Web Server requires at least 512 MB of memory to operate optimally. If your system is running low on swap space then you might get error shown below:

```
warning: CORE3283: stderr: Error occurred during initialization of VM
warning: CORE3283: stderr: Could not reserve enough space for object heap
catastrophe: CORE4005: Internal error: unable to create JVM
failure: server initialization failed
```

You will have to increase the swap space on your system. If you are running Web Server 7 under Solaris 10 zones, then you will need to increase the swap space within the global zone.

Note – Refer to your operating system document on how to add/increase swap space.

On Unix Systems, Web Server Consumes Excessive Memory on Startup

When starting up on Unix systems, Web Server employs an auto-tuning algorithm to provide values for certain elements when they are not specified in server.xml. Because this algorithm is based on the system's file descriptor limit, auto-tuning can cause Web Server to consume an excessive amount of memory when the file descriptor limit is very high and the elements are not explicitly specified in server.xml.

On startup, Web Server interrogates the system for the file descriptor limit, and then it allocates or reserves file descriptors for these items:

1. Web applications (80% of the descriptors)

2. Daemon session threads' connections (an average of 4 descriptors per thread)

3. JDBC pools

4. Access log counts for all virtual servers

5. Listener counts

6. File cache (the max-open-files subelement of the file-cache element)

7. Keep-alive connections (the max-connections subelement of the keep-alive element

8. Thread pool queue (the queue-size subelement of the thread-pool element)

The last three of these items are optional in server.xml, and are assigned auto-tuned values if they are not explicitly specified in server.xml.

The auto-tuning algorithm takes the number of file descriptors remaining after items one through five above are subtracted, and partitions them out according to the following ratios:

Subelement	max-open-files	:	max-connections	:	queue-size
File descriptor limit >= 1024	1	:	16	:	16
File descriptor limit < 1024	1	:	16	:	8

The fact that these ratios favor `max-connections` (and, to a lesser degree, `queue-size`), coupled with the fact that connections and queue members consume a considerable amount of memory, can give rise to excessive memory use on systems with a high file descriptor limit.

To avoid the excessive memory use, make sure to specify values for `max-open-files`, `max-connections`, and `queue-size` on systems that have a high file descriptor limit.

File Descriptor Issues

Under heavy load condition, Web Server may run out of file descriptors. In such cases you will get an error like the following:

```
[18/Dec/2005:20:01:03] failure ( 3014): HTTP3069:
Error accepting connection (PR_PROC_DESC_TABLE_FULL_ERROR: file descriptor table full)
```

Increase the file descriptor limit either per process or per system and restart the system.

Linux limits the number of file descriptors that any one process may open; the default limits are 1024 per process. These limits can prevent optimum performance of Web Server. The open file limit is one of the limits that can be tuned with the `ulimit` command. The command `ulimit -aS` displays the current limit, and `ulimit -aH` displays the hard limit (above which the limit cannot be increased without tuning kernel parameters). For setting the limit to hard limit, execute the following command:

```
ulimit -n unlimited
```

`java.lang.StackOverflowError` when accessing JSP and/or Servlet

The default value for thread stack size is 128K for 32-bit server and 256K for 64-bit server. Depending on the web application, the server may require additional thread stack size to service a request. If you are seeing `java.lang.StackOverflowError` when servicing Java content, you should increase the thread stack size using the Administration Console or the `wadm` CLI utility. It is recommended that you increase the thread stack size in increments of 8K until the problem ceases to exist. For example, to set the thread stack size to the default +8K on a 64-bit server, execute the following command:

```
$wadm set-thread-pool-prop --user=admin --port=8989
--password-file=passwordfile --config=config-name stack-size=270336
```

Performance Issues

See, *Oracle iPlanet Web Server 7.0.9 Performance Tuning, Sizing, and Scaling Guide*.

Getting Information From Running Processes

Web server 7.0 provides several ways to get more information on the running process, if your web server seems to hang or be little non responsive.

Find the running web server process:

1. Linux — Run the /bin/ps -o "pid ppid comm args" | grep webservd | grep -v admin-server command. The output will look similar to the following text:

   ```
   424     1 webservd-wdog    webservd-wdog -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r
   425   424 webservd         webservd -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r /opt/
   426   425 webservd         webservd -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r /opt/
   ```

   pstack -p 426 (the last child PID) will return the web server stack information of the running process.

2. Solaris — Run the /usr/bin/ps -o "pid ppid cmd args" | grep webservd | grep -v admin-server command. The output will look similar to the following text:

   ```
   424     1 webservd-wdog    webservd-wdog -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r
   425   424 webservd         webservd -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r /opt/
   426   425 webservd         webservd -d
         /var/opt/SUNWwbsvr7/https-chilidev1.red.iplanet.com/config -r /opt/
   ```

   pstack -p 426 (the last child PID) will return the web server stack information of the running process.

If you are running dynamic applications that uses JSPs/Servlets and you find that Web Server is becoming unresponsive during their execution, you can get Web Server to print the stack information within the log (logs/errors) by running the command kill -s QUIT 426 (last child PID).

4

Administration Server And Nodes

Running Administration Server As Non-Root

Problem: Unable to bind at port <port_number>.

The administration server and the administration nodes should be run by the same user id in Unix. This is because of the following reason:

1. If the instances are running as part of the root process, the Administration Server will not have sufficient permissions to modify the instance's configuration files, if it is not running as part of the root process.

So for the administration server to perform its job of maintaining the server instances, it either needs to be part of the root process or at least the same user as that of the server instance. However in Solaris 10, you don't need to run the Administration Server as root to bind to port 80 (or < 1024). Execute the following commands:

```
# su
# /usr/sbin/usermod -K defaultpriv=basic,net_privaddr webservd
```

Pulling Configuration Changes to the Administration Server

Whenever you make manual changes to the configuration, you should replicate the changes back into the Administration server repository as follows:

1. Manually edit server instance's configuration files as you would do with the earlier version of Web Server (Not Recommended).

2. Start the Web Server 7.0 Administration Server. See the `Administrator's Guide`.

3. For pulling the changes back to the Administration Server repository, execute the following command.

   ```
   wadm> pull-config --user=admin --config=CONFIG_NAME
   ```

Note – The operation may take some time depending on the configuration.

Note – Always use the Administration Console or the `wadm` CLI to edit the settings.

Making Changes to `server.policy` File

If you are deploying your web site containing Java web applications and would like to enable security manager within the web container, then you will need to edit the server instance's `server.policy` file. In this scenario, Administration Server does not provide a direct command line interface or GUI to edit this `server.policy` file. But you can still perform the steps as mentioned in the previous section to make any manual edit to the `server.policy` file.

Deployment Pending Message in Administration Console

Question: When I make changes to a configuration from the GUI, I see a deployment pending message. What does it mean?

Answer: The deployment pending message indicates that there are changes to the configuration that are saved into the administration server's configuration store. You need to click the Deploy button to save these changes to the instances.

Save and Deploy — Two Different Actions

Question: What is the difference between the Save and Deploy buttons that shows up on every page in the Administration Console within a configuration?

Answer: If you click on Save, then the changes that you have made through the Administration Console pages are saved only in the configuration store. When you click the Deploy button, the changes will be saved to the instances. Always use the Deploy button to persist your changes to the running instances.

Manually Editing Configuration Files

Question: Can I make manual edits to my configuration files? Will these be reflected in the CLI and GUI?

Answer: If you need to make manual changes to the configuration files, make the changes within the configuration store. These changes will be reflected in the CLI and GUI and can be deployed using the deploy-config command or by clicking on Deploy button in the GUI. But if you make changes directly in the instance, the Administration Console will show up a warning stating that there are manual modifications to the configuration files and you need to either overwrite these changes with the configuration in the configuration store or else get these changes back into the configuration store overwriting the existing configuration changes.

Note – Always use the Administration Console or the wadm CLI to edit the settings.

Where Are My Changes?

Question: I have made changes to my configuration files in the configuration store. I could not see the configuration effected by my changes.

Answer: Have you executed deploy-config command? ("Manually Editing Configuration Files" on page 33). Log in to the Administration Console and go that particular configuration. Do you notice a 'Deployment Pending' Message? ("Deployment Pending Message in Administration Console" on page 32).

deploy-config **Command Fails**

Question: Sometimes the deploy-config fails saying that a server restart is needed. Why?

Answer: When deploy-config is invoked, the configuration changes in the configuration store are propagated to the instances and the running servers are re-configured so that the configuration changes are picked up. But certain changes cannot be applied dynamically and need a server restart. You will hence need to run a restart-instance command in order to restart the server instances.

Where Can I Find wadm CLI?

If you are using wadm CLI on the same machine where the Administration Server is installed, then you do not need to know the Administration Server's port number. You are required to know the port number only if you have configured Administration Node and need to use wadm CLI to connect to a remote Administration Server.

Question: Where can I find the CLI for Web Server administration?

Answer: The administration CLI is located at install-root/bin/wadm. To use the CLI, you need to know:

- Administration server hostname (default is localhost).
- SSL port for the administration server (default is 8989).
- Administration server user name (default is admin).
- Administration server password.

Note – Administration server needs to be running in order to use the CLI. You can start the server by running install-root/admin-server/bin/startserv.

install-root/samples/admin/scripts directory contains scripts that you can run using the wadm command line utility. wadm is built on a TCLengine and hence supports TCL scripting. These scripts can be used to perform common administrative tasks. They also demonstrate how new utilities can be built on top of existing commands.

The following table describes the scripts:

TABLE 4–1   Sample wadm Scripts

Script	Description	Usage
enable-ssl.tcl	Enables SSL on a given virtual server and port.	wadm -f enable-ssl.tcl <config> <vs> <server> <port>
filter-mime.tcl	Fetches the matching MIME types from the given configuration and virtual server.	wadm -f filter-mime.tcl "<regex>" <config> <vs>
remove-mime.tcl	Removes the matching MIME types from the given configuration and virtual server.	wadm -f remove-mime.tcl "<regex>" <config> <vs>
add-mime-ext.tcl	Adds the specified extension to the matching MIME types in the given configuration and virtual server.	wadm -f add-mime-ext.tcl "<regex>" "ext" <config> <vs>

TABLE 4–1 Sample wadm Scripts *(Continued)*

Script	Description	Usage
summary.tcl	Provides a summary of the installation. It contains list of listeners, ports, and SSL status.	wadm -f summary.tcl
list-webapps.tcl	Provides a summary of all the deployed web applications.	wadm -f list-webapps.tcl
collate-logs.tcl	Provides a collated logs across multiple nodes.	wadm -f collate-logs.tcl <config> <node1> <node2> ..
renew-selfsigned-cert.tcl	Allows renewal of self signed certificates with a given nickname	wadm -f renew-selfsigned-cert.tcl <config> <cert-nickname> [<validity>]

Enabling SSL Port on Windows Firewall

On windows, if the firewall is enabled on the node it does not allow any inbound request. The administration server does not respond for a while. This will happen for every node related operation. To troubleshoot this problem, you should perform the following steps:

▼ To Enable Administration Node SSL Port on Windows

1   Start > Settings > Control Panel

2   Click the Windows Firewall icon
A window appears.

3   Click the Exceptions tab

4   Click the Add Port button
A window appears.

5   Enter the name in Name field

6   Enter the Administration Server's port number in Port field

7   Select the TCP option

8   Click the OK button.

Setting up LDAP Over SSL Through the Admin Console

When an administrator tries to setup an authentication database to a LDAPS server for their web server access control. If the LDAP server is setup to use SSL and has a self signed certificate, or is a certificate from a non-CA authority then the error will occur in the admin gui. This error is misleading because in reality the server instance is able to connect to the LDAP server given the CA certificate from the LDAPS server has been installed as trusted in the certificate database for the server instance. What administrators often miss is that the CA certificate has not been trusted for the administrator server's trust database and so the admin reports the following error.

```
"ADMIN3120: Could not connect to LDAP server"
```

For troubleshooting, the administrator needs to check the certificate authorities for both the administrator server and the server instance to make sure that the CA certificate that signed the LDAP server's SSL certificate is installed and trusted for the administration server via Nodes->Administration server->Certificate Authorities-> Install and also for the web instance.

5

WebDAV

■ "HTTP 403 (Forbidden) Error Message" on page 37

HTTP 403 (Forbidden) Error Message

After enabling WebDAV in Web Server, when you send a WebDAV request that needs write permissions, you get 403 Forbidden error.

This error message means you need to set appropriate ACLs in the ACL file for that virtual server. See the *Administrator's Guide*.

Note – If after enabling WebDAV in Web Server, make sure that the directory in which WebDAV is enabled, has Read and Write permissions for the user that owns webservd process (like "webservd" or "root").

CHAPTER 6

Security

- "Issuing Test Request to an SSL Server" on page 39
- "Analyzing SSL Requests" on page 40

Issuing Test Request to an SSL Server

While diagnosing problems between Web Server 7.0 and web browsers, it is useful to analyze the requests and responses. When SSL/TLS is not used, for capturing requests and responses between web browser and the server you can use any network capture tool such as ethereal.

But when SSL/TLS is used for communication, you can use the OpensSSL's s_client application for tapping the communication.

Execute the following command (after successful SSL connection) and enter the test HTTP request as desired.

```
$openssl s_client -host localhost -port 8080 -quiet
```

By using the same command without the -quiet flag, you can see information about the connection, such as the server DN, Certificate name and negotiated cipher suite.

For testing particular cipher suites, check the -cipher option. For example:

```
$openssl s_client -host localhost -port 8080  -cipher DES-CBC-SHA
```

For more information, see the s_client man page at
http://www.openssl.org/docs/apps/s_client.html.

Analyzing SSL Requests

Earlier method of issuing test request works well as long as you can recreate the request content manually. But sometimes you need to diagnose a connection that is being used by a web browser.

There are a number of tools available to observe such request and response data. One such tool is `ssltap`. `ssltap` takes the proxy approach-it serves as a simple proxy between the client and the Web Server and displays information about the connections it forwards (you can also use `ssltap` for observing plain HTTP requests or even requests based on other protocols).

Assume that Web Server is running with an SSL-enabled listener on port 8088 on a machine. Now issue the following command:

```
$ssltap -l -s localhost:8088
Looking up "localhost"...
Proxy socket ready and listening
```

Note – Use the -l option so `ssltap` doesn't exit after a single request.

By default `ssltap` listens on port 1924. Now connect to `https://localhost:1924` with your browser. You will get an output like the following snippet:

```
Connection #1 [Mon Dec 10 15:49:49 2006]
Connected to localhost:8088
--> [
alloclen = 87 bytes
(87 bytes of 87)
 [Mon Apr 10 15:49:49 2006] [ssl2]  ClientHelloV2 {
         version = {0x03, 0x01}
         cipher-specs-length = 60 (0x3c)
         sid-length = 0 (0x00)
         challenge-length = 16 (0x10)
         cipher-suites = {
             (0x000039) TLS/DHE-RSA/AES256-CBC/SHA
             (0x000038) TLS/DHE-DSS/AES256-CBC/SHA
             (0x000035) TLS/RSA/AES256-CBC/SHA
             (0x000033) TLS/DHE-RSA/AES128-CBC/SHA
             (0x000032) TLS/DHE-DSS/AES128-CBC/SHA
             (0x000004) SSL3/RSA/RC4-128/MD5
             (0x000005) SSL3/RSA/RC4-128/SHA
             (0x00002f) TLS/RSA/AES128-CBC/SHA
             (0x000016) SSL3/DHE-RSA/3DES192EDE-CBC/SHA
             (0x000013) SSL3/DHE-DSS/DES192EDE3CBC/SHA
             (0x00feff) SSL3/RSA-FIPS/3DESEDE-CBC/SHA
             (0x00000a) SSL3/RSA/3DES192EDE-CBC/SHA
             (0x000015) SSL3/DHE-RSA/DES56-CBC/SHA
             (0x000012) SSL3/DHE-DSS/DES56-CBC/SHA
             (0x00fefe) SSL3/RSA-FIPS/DES-CBC/SHA
             (0x000009) SSL3/RSA/DES56-CBC/SHA
             (0x000064) TLS/RSA-EXPORT1024/RC4-56/SHA
```

```
                            (0x000062) TLS/RSA-EXPORT1024/DES56-CBC/SHA
                            (0x000003) SSL3/RSA/RC4-40/MD5
                            (0x000006) SSL3/RSA/RC2CBC40/MD5
                            }
                  session-id = { }
                  challenge = { 0xdfb5 0x1d22 0x6562 0x34f6 0x95b9 0x668a 0x234e 0x38ea }
}
]
```

This is the SSL client hello being sent from the browser to the server. Note the list of cipher suites the browser has sent. This is the set of cipher suites the browser is configured to handle (note that they are sorted in order of preference). The server will pick one of those for the handshake (if the server is not set up to handle any of these, the connection will then immediately fail). In the above snippet, the session-id is empty, which tells you the browser does not have any cached SSL session with this particular server.

For the above request, the Web Server's response is shown below:

```
<-- [
(1015 bytes of 1010)
SSLRecord { [Mon Apr 10 15:49:49 2006]
    type    = 22 (handshake)
    version = { 3,1 }
    length  = 1010 (0x3f2)
    handshake {
        type = 2 (server_hello)
        length = 70 (0x000046)
            ServerHello {
                server_version = {3, 1}
                random = {...}
                session ID = {
                    length = 32
                    contents = {..}
                }
                cipher_suite = (0x0035) TLS/RSA/AES256-CBC/SHA
            }
        type = 11 (certificate)
        length = 928 (0x0003a0)
            CertificateChain {
                chainlength = 925 (0x039d)
                Certificate {
                    size = 485 (0x01e5)
                    data = { saved in file 'cert.001' }
                }
                Certificate {
                    size = 434 (0x01b2)
                    data = { saved in file 'cert.002' }
                }
            }
        type = 14 (server_hello_done)
        length = 0 (0x000000)
    }
}
]
```

The server picked `TLS/RSA/AES256-CBC/SHA` as the cipher suite to use. A session ID was sent, which this client will include in subsequent requests. The server also sent its certificate chain for

the browser to verify. `ssltap` saved these certificates in the files noted `cert.001` and `cert.002`. You can examine these certificates with any tool that can parse X.509 certificates. For example, execute the following command:

```
$openssl x509 -in cert.001 -text -inform DER
```

Note – `ssltap` is a single threaded proxy server. So if you issue multiple requests through it, the requests will get serialized. If you need to analyze a specific problem with your application that only occurs on concurrent requests through SSL, try running multiple `ssltap` instances.

7

FastCGI

- "Getting More Information" on page 43
- "FastCGI Requests Are Not Getting Served" on page 43
- "Dependency Issues" on page 44

Getting More Information

Fastcgistub is a process manager that manages the life cycle of the FastCGI application processes. Fastcgistub logs its messages into Fastcgistub.log file under the Web Server's temporary directory. In case of any error, analyzing this file can help in debugging the problem.

FastCGI Requests Are Not Getting Served

FastCGI request not getting served is one of the common problems.

1. While starting the Web Server, did you see the following message?

   ```
   FCGI1000: Oracle iPlanet Web Server 7.0 FastCGI NSAPI Plugin <build-information>
   ```

   Yes — FastCGI Plug-in has been loaded successfully.

   No — Check the path to the plug-in library within magnus.conf file.

2. Check if the request mapping is correctly specified within obj.conf file. For more information, see *Administrator's Configuration File Reference Guide*.

3. Check the Server Log for any error message.

4. Check the permissions for the stub binary and FastCGI applications. Without sufficient permissions, the plug-in will fail to start the stub/application.

5. Check the *Fastcgistub.log* file for any possible error on the stub side.

6. Try executing the FastCGI application in a standalone mode and check if the application runs normally.

Dependency Issues

If library dependency errors are thrown, then whichever path is required to be added to `LD_LIBRARY_PATH` for resolving the library dependency errors in the standalone mode, the same library path should be specified within `obj.conf` file as `app-env` parameter with `LD_LIBRARY_PATH` value.

8

Gathering Debug Data

- "Gathering Debug Data for Web Server" on page 45

Gathering Debug Data for Web Server

You can use Sun Gathering Debug Data (Sun GDD) tool that the Support Center uses to debug problems with Web Server.

See *Sun Gathering Debug Data for Sun Java System Web Server* at `http://docs.sun.com/doc/820-4983.`

Index