



Sun Gathering Debug Data for Sun Java System Web Server

Sun Java™ Enterprise System Technical Note



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820-2483
June J2007

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux États-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivés du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux États-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux États-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux États-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des États-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Contents

1 Sun Gathering Debug Data for Sun Java System Web Server	5
1.1 Technical Note Revision History	6
1.2 About This Technical Note	6
1.2.1 Prerequisites for Gathering Web Server Debug Data	7
1.2.2 Variables Used in This Technical Note	7
1.3 Overview of Gathering Debug Data for Web Server	8
1.4 Creating a Service Request with the Sun Support Center	8
1.5 Types of Web Server Debug Data	9
▼ To Gather General Debug Data for Any Web Server Problem	9
▼ To Gather Debug Data on Web Server Installation Problems	11
▼ To Gather Debug Data on Web Server Startup Problem	12
▼ To Gather Debug Data on a Hung or Unresponsive Web Server Process	13
▼ To Gather Debug Data on Web Server Crashed Process	19
1.6 Configuring Solaris to Generate Core Files	22
▼ To Configure Solaris to Generate Core Files	22
1.7 Running the Web Server Debugging Scripts	24
▼ To Run the webinfo Script	24
▼ To Run the wshang Script	25
▼ To Run the pkg_app Script	25
1.8 Reporting Problems	26
1.9 Accessing Sun Resources Online	26
1.10 Third-Party Web Site References	26
1.11 Sun Welcomes Your Comments	27

Sun Gathering Debug Data for Sun Java System Web Server

This technical note describes how to use Sun™ Gathering Debug Data (Sun GDD or GDD) that the Sun Support Center requires to debug problems with Sun Java™ System Web Server. By collecting this data at the time of raising a Service Request, you can substantially reduce the time needed to analyze and resolve the problem. For more information on how this document and associated scripts can help you in better dealing with Web Server problems, see:

<http://www.sun.com/service/gdd/index.xml>

This document is intended for anyone who needs to raise a Service Request about Web Server issues with the Sun Support Center.

This technical note contains the following sections:

- “1.1 Technical Note Revision History” on page 6
- “1.2 About This Technical Note” on page 6
- “1.3 Overview of Gathering Debug Data for Web Server” on page 8
- “1.4 Creating a Service Request with the Sun Support Center” on page 8
- “1.5 Types of Web Server Debug Data” on page 9
- “1.6 Configuring Solaris to Generate Core Files” on page 22
- “1.7 Running the Web Server Debugging Scripts” on page 24
- “1.9 Accessing Sun Resources Online” on page 26

1.1 Technical Note Revision History

Version	Date	Description of Changes
1.2	June 2007	Addressed review comments and added information about the new script <code>wshang.ksh</code> .
1.1	January 2007	Updated “To Configure Solaris to Generate Core Files” on page 22.
1.0	December 2006	Initial release of this technical note.

1.2 About This Technical Note

This document covers the following versions of Sun Java System Web Server on the Solaris™, HP-UX, Linux, and Microsoft Windows platforms:

- Sun Java System Web Server 7.0
- Sun Java System Web Server 6.1 (SunONE Web Server)
- iPlanet™ Web Server 6.0

Note – The versions mentioned above include all update release or service packs for the products.

You can use this document in all types of environments, including test, pre-production, and production. Verbose debugging is not used (to reduce performance impact), except when it is deemed necessary. At the same time, it is possible that the problem could disappear when you configure logging for debug mode. However, this is the minimum requirement to understand the problem. In the majority of cases, the debug data described in this document is sufficient to analyze the problem.

This document does not provide workarounds, techniques or tools to analyze debug data.

Note – If your problem does not fit into any of the specific categories, provide the general information described in [“1.5 Types of Web Server Debug Data”](#) on page 9 and clearly explain your problem.

If the information you initially provide is not sufficient to find the root cause of the problem, Sun Support Center will ask for more details, as needed.

1.2.1 Prerequisites for Gathering Web Server Debug Data

The prerequisites for Sun GDD for Web Server are as follows:

- Make sure you have superuser privileges.
- For the Solaris platform, obtain the `pkg_app` and the `wshang` scripts and for UNIX and Linux, obtain the `webinfo` scripts from the following location:
<http://www.sun.com/bigadmin/scripts/indexSjs.html>
- For Windows platform, download the free Debugging Tools for Windows to analyze the process hang problems.

Note – The debugger Dr. Watson is not useful for process hang problems because it cannot generate a crash dump on a running process.

Download the free Debugging Tools from the following location:

<http://www.microsoft.com/whdc/devtools/debugging/default.aspx>

Install the latest version of Debugging Tools and the OS Symbols for your version of Windows. Also, you must add the environment variable `NT_SYMBOL_PATH` in the windows environment variable. Refer to Operating System version documentation or follow the installation guide for the OS Symbols package.

Use the command `drwtsn32 -i` to select Dr. Watson as the default debugger. Use the command `drwtsn32`, check all options, and choose the path for crash dumps.

1.2.2 Variables Used in This Technical Note

This section describes the variables used in the procedures in this document. Gather the values of the variables before you try to do the procedures.

- *slapd-identifier*: The Directory Server instance name used during installation. The installation program automatically added the prefix `slapd-` to the name you specified. For example, if you named the identifier `tango`, the installation program created is `slapd-tango` and `slapd-tango` is the *slapd-identifier*.
- *web-pid*: Process ID of a Web Server daemon.
- *web-port*: Port number on which the Web Server is listening.
- *web-identifier*: The Web Server instance name used during installation. The installation program automatically adds the prefix `https-` to the name you specified. For example, if you named the identifier `tango`, the installation program created `https-tango`, it is the *web-identifier*.

- *server-root*: The directory on the Web Server machine that stores the server specific information. This directory holds the server program, configuration, maintenance, and log files.
- *windbg-root*: The directory on the Windows Web Server machine that stores the Win Debugger program, and configuration, maintenance, and information files.

1.3 Overview of Gathering Debug Data for Web Server

Sun GDD for a Web Server problem involves these basic operations:

1. Gather system information.
2. Gather specific problem information (installation problem, process hang, Startup, or process crash).
3. Create a tar .gz file of all the debug information and upload it for the Sun Support Center.
4. Create a Service Request with the Sun Support Center.

1.4 Creating a Service Request with the Sun Support Center

When you create a Service Request with the Sun Support Center, either online or by phone, provide the following information:

- A clear problem description
- Details of the state of the system, both before and after the problem started
- Impact on end users
- All recent software and hardware changes
- Any actions already attempted
- Whether the problem is reproducible; when reproducible, provide the detailed test case
- Whether a pre-production or test environment is available
- Name and location of the archive file containing the debug data

Upload your debug data archive file to one of the following location:

<http://supportfiles.sun.com/upload>

<https://supportfiles.sun.com/upload>

For more information on how to upload files to this site, see:

<http://supportfiles.sun.com/show?target=faq>

Note – When opening a Service Request by phone with the Sun Support Center, provide a summary of the problem in a text file named `Description.txt`. Be sure to include `Description.txt` in the archive along with the rest of your debug data.

1.5 Types of Web Server Debug Data

This section describes the various kinds of debug data that you need to provide to the Sun Support Center. The procedure to obtain debug data based on the kind of problem you are experiencing is described in-detail.

This section contains the following topics:

- “To Gather General Debug Data for Any Web Server Problem” on page 9
- “To Gather Debug Data on Web Server Installation Problems” on page 11
- “To Gather Debug Data on Web Server Startup Problem” on page 12
- “To Gather Debug Data on a Hung or Unresponsive Web Server Process” on page 13
- “To Gather Debug Data on Web Server Crashed Process” on page 19

▼ To Gather General Debug Data for Any Web Server Problem

To report problems described in this technical note, you need to gather some basic information. Basic information includes System details and date and time when the problem occurred. Follow these steps to gather the basic information.

- 1 **Note the day(s) and time(s) the problem occurred.**
- 2 **Provide a graphical representation of your deployment. Include all hosts and IP addresses, host names, operating system versions, role they perform, and other important systems such as load balancers, firewalls, and so forth.**

- 3 **Note the Version of the operating system.**

Solaris `uname -a`

HP-UX `uname -r`

Linux `more /etc/redhat-release`

Windows `C:\Program Files\Common files\Microsoft Shared\MSInfo\msinfo32.exe`
`/report C:\report.txt`

- 4 **Note the patch level.**

Solaris `showrev -p`

HP-UX `swlist`

Linux `rpm -qa`

Windows Already provided in the `C:\report.txt` file above.

5 Note the version of Web Server.

If a configured JDK is used instead of the default JRE then provide the output of the command `java -version`.

Web Server version is indicated in the error log of the instance during the start.

- Start Instance Script

UNIX (Solaris and HP-UX) and Linux `cd server-root/web-identifier/start`

- Error logs

UNIX (Solaris and HP-UX) and Linux `cd server-root/web-identifier/logs/errors`

Windows `cd server-root\web-identifier\logs\errors`

- Access logs

UNIX (Solaris and HP-UX) and Linux `cd server-root/web-identifier/logs/access`

Windows `cd server-root\web-identifier\logs\access`

6 Create a tar file of the Web Server configuration directory.

- Sun Java System Web Server :

UNIX (Solaris and HP-UX) and Linux `cd server-root/web-identifier/config`
Create a tar file of the `server-root/config` directory.

Windows `cd server-root\web-identifier\config`
Create a compressed file of the `server-root\config` directory.

Note – If possible, provide an explorer (`SUNWexplo`) of the machine where the problem occurs. For UNIX and Linux systems, the customer can use the script `webinfo`. For more information on how to run the `webinfo` script, see [“To Run the webinfo Script” on page 24](#).

▼ To Gather Debug Data on Web Server Installation Problems

Follow these steps if you are unable to complete the installation or if you get a “failed” installation status for Web Server.

1 See the following troubleshooting information:

- Sun Java Enterprise System 5:
<http://docs.sun.com/app/docs/doc/820-0464>
- Sun Java Enterprise System 2005Q4:
Chapter 9, “Troubleshooting,” in *Sun Java Enterprise System 2005Q4 Installation Guide for UNIX*
- Sun Java Enterprise System 2005Q1:
<http://docs.sun.com/source/819-0056/troubleshooting.html>
- Sun Java Enterprise System 2004Q2:
<http://docs.sun.com/source/817-5760/troubleshooting.html>
- Sun Java Enterprise System 2003Q4:
<http://docs.sun.com/source/816-6874/std-troubleshooting.html>

If the problem persists after using this troubleshooting information, then continue with this procedure to gather the necessary data for the Sun Support Center.

2 Gather the general system information as explained in “[To Gather General Debug Data for Any Web Server Problem](#)” on page 9.

3 Specify whether this is a first-time installation or a Hot Fix installation on a pre-existing Web Server.

4 Get the installation logs.

- Sun Java System Web Server (Web Server 7.0):
Web Server 7.0 log files mostly reside in the *server-root*/log directory. However, the initial configuration log files reside in the *server-root*/install directory, which also contains information on the initial configuration.

Solaris /var/sadm/install/logs
The log file names start with
Java_Enterprise_System*_install.Bdatetime, where *date* and *time* correspond to the failing installing (for example, B12161532).

HP-UX and Linux /var/opt/sun/install/logs
The log file names start with
Java_Enterprise_System*_install.Bdatetime, where *date* and
time correspond to the failing installing (for example, B12161532).

Windows C:\DocumentsandSettings\current-user\LocalSettings\Temp
The log file names start with MSI*.log (usually a text file). The
asterisk (*) represents a random number in the Temp directory for
each MSI based setup.

- Sun Java System Web Server (Web Server 6.1):

Web Server 6.1 log files mostly reside in the *server-root*/log directory. However, the initial configuration log files reside in the *server-root*/install directory, which also contains information on the initial configuration.

Solaris /var/sadm/install/logs
The log file names start with
Java_Enterprise_System*_install.Bdatetime, where *date* and
time correspond to the failing installing (for example, B12161532).

HP-UX and Linux /var/opt/sun/install/logs
The log file names start with
Java_Enterprise_System*_install.Bdatetime, where *date* and
time correspond to the failing installing (for example, B12161532).

Windows C:\DocumentsandSettings\current-user\LocalSettings\Temp
The log file names start with MSI*.log (usually a text file). The
asterisk (*) represents a random number in the Temp directory for
each MSI based setup.

- iPlanet Web Server (Web Server 6.0):

Rerun the installation with the following command and save the output file.

Solaris truss -ealf -rall -wall -vall -o /tmp/install-web.truss ./setup

HP-UX tusc -v -fealT -rall -wall -o /tmp/install-web.tusc ./setup

Linux strace -fv -o /tmp/install-web.strace ./setup

Windows Use DebugView tool. You can download this tool from
<http://www.sysinternals.com/Utilities/DebugView.html>

▼ To Gather Debug Data on Web Server Startup Problem

Follow these steps if you are unable to start a Web Server instance.

- 1 **Gather the general system information as explained in “To Gather General Debug Data for Any Web Server Problem” on page 9.**

- 2 **Run the netstat command and save the output.**

UNIX (Solaris and HP-UX) and Linux `netstat -an | grep web-port`

Windows `netstat -an`

- 3 **Run the following command on the Web Server start script and provide the resultant file.**

Solaris `truss -eafL -wall -vall -rall -o /tmp/web-start.truss ./start`

HP-UX `tusc -v -fealT -rall -wall -o /tmp/web-start.tusc ./start`

Linux `strace -fv -o /tmp/web-start.strace ./start`

Windows Use DebugView tool. You can download this tool from <http://www.sysinternals.com/Utilities/DebugView.html>

- 4 **If logs file does not contain any error message about the problem, do the following:**

Edit and add the following line to the configuration file to get more debug information during the start.

UNIX (Solaris and HP-UX) and Linux Edit the file
`server-root/web-identifier/config/magnus.conf`
and add the following line at the end:
`LogLevel=finest` in `server.xml`.

Windows Edit the file
`server-root\web-identifier\config\magnus.conf`
and add the following line at the end:
`LogLevel=finest` in `server.xml`.

▼ To Gather Debug Data on a Hung or Unresponsive Web Server Process

A process hang is defined as one of the Web Server processes not responding to requests while the `httpd` process is still running.

Before You Begin Make sure that you collect all the data over the same time frame in which the problem occurs. See “1.6 Configuring Solaris to Generate Core Files” on page 22 if a core file is not generated.

Gather the following information for process hang problems. Run the commands in the order when the problem occurs. Be sure to specify the time when the process hanged and list the affected processes, if possible.

- 1 **Gather the general system information as explained in “To Gather General Debug Data for Any Web Server Problem” on page 9.**
- 2 **For Solaris, use the `ptree` command on the `uxwdog` process to find about the process.**

Note – If you are using Web Server 6.1 or Web Server 7.0, instead of the `uxwdog` process, use the `websrvd-wdog` process.

Output

```
ptree 11449
11449 ./uxwdog -d /prods/crypto/60SP6/https-sun/config
  11450 ns-httpd -d /prods/crypto/60SP6/https-sun/config
    11451 ns-httpd -d /prods/crypto/60SP6/https-sun/config
```

Note – Gather the data on the highest PID process, which in this example is 11451. The Web Process is either `ns-httpd` or `websrvd`, depending on the Web Server version.

- 3 **Run the `netstat` command and save the output.**

UNIX (Solaris and HP-UX) and Linux `netstat -an | grep web-port`

Windows `netstat -an`

- 4 **(For Solaris), `wshang` script captures the debug data.**

The `wshang` script is available at: <http://www.sun.com/bigadmin/scripts/indexSjs.html>

Run the script `pkg_app` on one of the core file generated by the `wshang` script. For more information on how to run the `wshang` script, see “To Run the `wshang` Script” on page 25.

- 5 **Run the following commands and save the output.**

Solaris

```
ps -aux
server-root
vmstat 5 5
iostat [ -t ] [ interval [ count ] ]
top
uptime
```

HP-UX

```
ps -aux
server-root
vmstat 5 5
```

```
iostat [ -t ] [ interval [ count ] ]
top
sar
```

Linux

```
ps -aux
server-root
vmstat 5 5
top
uptime
sar
```

Windows Obtain the WEB process PID:

```
C:\windbg-root>tlist.exe
```

Obtain the process details of the WEB running process PID:

```
C:\windbg-root>tlist.exe web-pid
```

6 Get the swap information.

Solaris `swap -l`

HP-UX `swapinfo`

Linux `free`

Windows Already provided in `C:\report.txt` as described in [“To Gather General Debug Data for Any Web Server Problem” on page 9.](#)

7 If the Web Server uses a Directory Server, provide the access, errors and audit logs of the Directory Server used by the Web Server.

■ Access log

UNIX (Solaris and HP-UX) and Linux `server-root/slapd-identifier/logs/access`

Windows `server-root\slapd-identifier\logs\access`

■ Errors log

UNIX (Solaris and HP-UX) and Linux `server-root/slapd-identifier/logs/errors`

Windows `server-root\slapd-identifier\logs\errors`

■ Audit log

UNIX (Solaris and HP-UX) and Linux `server-root/slapd-identifier/logs/audit`

Windows `server-root\slapd-identifier\logs\audit`

Note – The paths of these logs files are specified by the following parameters in the `dse.ldif` file. `nsslapd-accesslog`, `nsslapd-errorlog`, and `nsslapd-auditlog`

The `dse.ldif` file is located in the `config` directory.

UNIX (Solaris and HP-UX) and Linux `server-root/slapd-identifier/config/dse.ldif`

Windows `server-root\slapd-identifier\config\dse.ldif`

8 (For Solaris) If you are able to isolate the hanging process, get the following debug data for that process. Otherwise, get the following data for each of the Web Server processes.

For Solaris Using the PID obtained in Step 3, get a series of five of the following commands (one every 10 seconds):

`pstack web-pid`

`pmap -x web-pid`

Additionally, get the outputs of the following commands:

`prstat -L -p web-pid`

`pfiles web-pid`

`pmap web-pid`

9 Search for any core file that could have been dumped by one of the Web Server processes. If you find one, see “To Gather Debug Data on Web Server Crashed Process” on page 19.

10 Get the output of the following command.

Solaris `truss -ealf -rall -wall -vall -o /tmp/WEBProc-PID -p web-pid`

HP-UX `tusc -v -fealT -rall -wall -o /tmp/WEBProc-PID -p web-pid`

Linux `strace -fv -o /tmp/WEBProc-PID.strace -p web-pid`

Windows Use DebugView tool. You can download this tool from
<http://www.sysinternals.com/Utilities/DebugView.html>

Note – Wait for a minute after launching the appropriate command (`truss`, `strace`, `tusc`, or `DebugView`) then stop it by pressing **Control+C** in the terminal where you launched the command.

11 Get core files and the output of the following commands.

If a process hangs, it is helpful to compare several core files to review the state of the threads over time. Make a copy of the core file to a new name, wait for approximately one minute then rerun the following commands, so that the core files are not overwritten. Do this three times to obtain three core files.

Note – For HP-UX, you need PHKL_31876 and PHCO_32173 patches to use the `gcore` command. If you cannot install these patches, use the HP-UX `/opt/langtools/bin/gdb` command from version 3.2 and later, or the `dumpcore` command.

Solaris `cd server-root/bin/https/bin;`
 `gcore -o /tmp/web-core web-pid;`
 `pstack /tmp/web-core`

HP-UX

```
# cd server-root/bin/https/bin
gcore -p web-pid
(gdb) attach web-pid
Attaching to process web-pid
No executable file name was specified
(gdb) dumpcore
Dumping core to the core file core.web-pid
(gdb) quit
The program is running. Quit anyway (and detach it)? (y or n) y
Detaching from program: , process web-pid
```

Note – The `core.web-pid` should be generated in the `web-identifier/config` directory.

Linux

```
# cd server-root/bin/https/bin
gdb
(gdb) attach web-pid
Attaching to process web-pid
No executable file name was specified
(gdb) gcore
Saved corefile core.web-pid

(gdb)backtrace
(gdb)quit
```

Windows Get the WEB process PID:

```
C:\windbg-root>tlist.exe
```

Generate a crash dump on the WEB running process PID:

```
C:\windbg-root>adplus.vbs -hang -p web-pid -o C:\crashdump_dir
```

Note – For Windows, provide the complete generated folder under C:\crashdump_dir.

12 For Solaris, Archive the result of the script pkg_app (at least one core file is required).

```
./pkg_app.ksh pid-of-application corefile
```

The Sun Support Center requires the output from the pkg_app script to properly analyze the core file(s). For more information on how to run the pkg_app script, see [“To Run the pkg_app Script” on page 25](#)

Note – Make sure that the appropriate limitations are set by using the ulimit command, and that the user is not *nobody*. Also check the coreadm command for additional control. See [“1.6 Configuring Solaris to Generate Core Files” on page 22](#) if a core file is not generated.

Note – If you are using Web Server 6.1 or Web Server 7.0, do not proceed further with the next step.

13 For UNIX and Linux, If JVM is used for the Web applications, provide the JVM Stack traces during a hang situation.

A series of three to five Stack traces will be required.

To enable thread dumps for version 6.0, perform the following steps:

a. Edit the configuration file

```
server-root/https-host/obj.conf
```

b. Modify the following line

```
Init fn="NSServletLateInit" LateInit=yes
```

```
to
```

```
Init LateInit="yes" fn="NSServletInit" CatchSignals="yes" Signals=SIGQUIT
```

c. Add or modify the following line in /server-root/https-host/jvm12.conf

```
jvm.printerrors=1
```

d. Restart Web Server.

Note – When a problem occurs during a restart, issuing a kill —3 against the process dumps the stack traces into the Web Server errors log.

▼ To Gather Debug Data on Web Server Crashed Process

Use this task to gather data when a Web Server process has stopped (crashed) unexpectedly. Run all the commands on the actual machine where the core file(s) were generated.

- 1 **Gather the general system information as explained in “To Gather General Debug Data for Any Web Server Problem” on page 9.**
- 2 **Try to restart Web Server.**
- 3 **If the Web Server is using a Directory Server, provide the access, errors and audit logs of the Directory Server used by the Web Server**

- Access log

UNIX (Solaris and HP-UX) and Linux	<i>server-root/slapd-identifier/logs/access</i>
------------------------------------	---

Windows	<i>server-root\slapd-identifier\logs\access</i>
---------	---

- Errors log

UNIX (Solaris and HP-UX) and Linux	<i>server-root/slapd-identifier/logs/errors</i>
------------------------------------	---

Windows	<i>server-root\slapd-identifier\logs\errors</i>
---------	---

- Audit log

UNIX (Solaris and HP-UX) and Linux	<i>server-root/slapd-identifier/logs/audit</i>
------------------------------------	--

Windows	<i>server-root\slapd-identifier\logs\audit</i>
---------	--

Note – The paths of these logs files are specified by the following parameters in the `dse.ldif` file. `nsslapd-accesslog`, `nsslapd-errorlog`, and `nsslapd-auditlog`

The `dse.ldif` file is located in the `config` directory.

UNIX (Solaris and HP-UX) and Linux	<i>server-root/slapd-identifier/config/dse.ldif</i>
------------------------------------	---

Windows	<i>server-root\slapd-identifier\config\dse.ldif</i>
---------	---

- 4 **Get the output of the following commands.**

Solaris `ps -aux | server-root`
 `vmstat 5 5`
 `iostat -x`
 `top`
 `uptime`

HP-UX `ps -aux | server-root`
 `vmstat 5 5`
 `iostat -x`
 `top`
 `sar`

Linux `ps -aux | server-root`
 `vmstat 5 5`
 `top`
 `uptime`
 `sar`

Windows Obtain the WEB process PID:

`C:\windbg-root>tlist.exe`

Obtain process details of the WEB running process PID:

`C:\windbg-root>tlist.exe web-pid`

5 Get the swap information.

Solaris `swap -l`

HP-UX `swapinfo`

Linux `free`

Windows Already provided in `C:\report.txt` as described in [“To Gather General Debug Data for Any Web Server Problem”](#) on page 9.

6 Get the system logs.

Solaris and Linux `/var/adm/messages`
 `/var/log/syslog`

HP-UX `/var/adm/syslog/syslog.log`

Windows Event log files:
 Start-> Settings-> Control Panel —> Event Viewer-> Select Log
 Then click Action-> Save log file as and type the name for the resulting
 file.

7 Get core files (called “Crash Dumps” in Windows).

Solaris See “1.6 Configuring Solaris to Generate Core Files” on page 22 if a core file was not generated.

Linux Core dumps are turned off by default in the `/etc/profile` file. You can make user-specific changes by editing your `~/ .bash_profile` file. Look for the following line:

```
ulimit -S -c 0 > /dev/null 2>&1
```

You can either comment out the entire line to set no limit on the size of the core files or set your own maximum size.

Windows Generate a crash dump during a crash of Web Server by using the following commands:

Get the WEB process PID :

```
C:\windbg-root>tlist.exe
```

Generate a crash dump when the WEB process crashes, by executing the following commands:

```
C:\windbg-root>adplus.vbs -crash -FullOnFirst -p web-pid -o
C:\crashdump_dir
```

The `adplus.vbs` command monitors `web-pid` until it crashes and generates the dmp file. Provide the complete generated folder under `C:\crashdump_dir`.

Note – If you have not installed the Debugging Tools for Windows, you can use the `drwtsn32 -i` command to select Dr. Watson as the default debugger. Use the `drwtsn32` command, check all options, and choose the path for crash dumps. Then provide the dump and the `drwtsn32.log` files.

8 (Solaris) For each core file, provide the output of the following commands.

```
cd server-root/bin/https/bin
file corefile
pstack corefile
pmap corefile
pflags corefile
```

9 (Solaris) Archive the result of the script `pkg_app` (one core file is sufficient).

```
./pkg_app.ksh Pid-of-application corefile
```

Note – The Sun Support Center must have the output from the `pkg_app` script to properly analyze the core file(s). For more information on how to run the `pkg_app` script, see [“To Run the `pkg_app` Script” on page 25](#).

All these commands must be executed on the actual machine where the core file(s) were generated.

1.6 Configuring Solaris to Generate Core Files

Core files are generated when a process or an application terminates abnormally. You can manage the core files with the `coreadm` command. This section describes how to use the `coreadm` command to configure a system so that all process core files are placed in a single system directory. This enables you to track problems by examining the core files in a specific directory whenever a Solaris process or daemon terminates abnormally.

Before configuring your system for the core files, make sure that the `/var` file system has sufficient space. Once you configure Solaris to generate the core files, a core file is written to the `/var/cores` directory every time a process crashes.

▼ To Configure Solaris to Generate Core Files

1 Run the following commands as root.

```
mkdir -p /var/cores
coreadm -g /var/cores/%f.%n.%p.%t.core -e global -e global-setid -e
log -d process -d proc-setid
```

In this command:

- g Specifies the global core file name pattern. Unless a per-process pattern or setting overrides it, core files are stored in the specified directory with a name such as *program.node.pid.time.core*, for example: *mytest.myhost.1234.1102010309.core*.
- e Specifies options to enable. The preceding command enables:
 - Use of the global (that is, system-wide) core file name pattern (and thereby location)
 - Capability of `setuid` programs to also dump core as per the same pattern
 - Generation of a syslog message by any attempt to dump core (successful or not)
- d Specifies options to disable. The preceding command disables:
 - Core dumps per the per-process core file pattern
 - Per-process core dumps of `setuid` programs

The preceding command stores all core dumps in a central location with names identifying what process dumped core and when. These changes only impact processes started after you run the `coreadm` command. Use the `coreadm -u` command after the preceding command to apply the settings to all existing processes.

2 Display the core configuration.

```
# coreadm global core file pattern: /var/cores/%f.%n.%p.%t.core
    init core file pattern: core
        global core dumps: enabled
    per-process core dumps: disabled
        global setid core dumps: enabled
    per-process setid core dumps: disabled
        global core dump logging: enabled
```

See the `coreadm` man page for further information.

3 Set the size of the core dumps to unlimited.

```
# ulimit -c unlimited
# ulimit -a

    coredump(blocks) unlimited
```

See the `ulimit` man page for further information.

4 If the Web Server instance is running in SSL mode, it does not generate a core file. To enable the instance to generate the core file, add the following line to the server start script `start`.

```
SSL_DUMP=1; export SSL_DUMP
```

Note – This step applies to Web Server 6.0 and 6.1. If you are using Web Server 7.0, this step must be skipped.

5 Verify the core file creation.

```
# cd /var/cores
# sleep 100000 &
[1] PID
# kill -8 PID
# ls
```

1.7 Running the Web Server Debugging Scripts

This section describes how to run the `pkg_app`, `webinfo`, and `wshang` scripts.

Note – The `webinfo` script is currently applicable to only Web Server 6.0.

▼ To Run the `webinfo` Script

There may be commands within the `webinfoscript` that you can run manually.

Note – You can download a new version of the script at the following URL:

<http://www.sun.com/bigadmin/scripts/indexSjs.html>

To run the `webinfo` script, perform the following steps:

- 1 Move `webinfo_version.sh` to a temporary directory.**
- 2 Change the following environment variable at the beginning of the file, `info.sh`**
`SERVER_ROOT=`
`INSTANCE=`
`TMPDIR= /tmp`
- 3 Make sure that you have a space between the parameter `TMPDIR=` and the value that you entered.**
- 4 Run the script.**

This creates the `$TMPDIR/webinfo.date` directory and a tar file of that directory
`$TMPDIR/webinfo.date.tar`

Note – Run this script as the Web Server user rather than root.

- 5 Send the tar file to The Sun Support Center.**

Clean up `$TMPDIR`, if necessary.

For error information which is related to your problem, see the Administrator or Web Server errors and access files.

▼ To Run the wshang Script

- 1 The wshang script collects three snapshots of the following information at 15 seconds interval against the hung instance:

```
pstack
pfiles
prstat -L -a
pflags
pmap -x
pldd
```

Note – You can modify the time interval by editing the script and changing the variable *DURATION*

- 2 Run the wshang script. It shows a list of Web Server Process.
- 3 Choose the process that has the problem.

▼ To Run the pkg_app Script

This script packages an executable and all of its shared libraries into one tar file. Optionally, you can provide the PID of the application and the name of the core file. The files are stripped of their directory paths and are stored under a relative directory named `app/` allowing them to be unpacked in one directory.

On Solaris 9 or higher version, the list of files is derived from the core file rather than the process image if the core file is specified. You must provide the PID of the running application to assist in path resolution.

Two scripts are created to facilitate opening the core file when the tar file is unpacked:

- `opencore`. Execute this script after you unpack the tar file. The script sets the name of the core file and the linker path to use the `app/` subdirectory and then invokes `dbx` with the `dbxrc` file as the argument.
- `dbxrc`. This script contains the `dbx` initialization commands to open the core file.

- 1 Copy the script to a temporary directory on the system where Web Server is installed.
- 2 Login as superuser.
- 3 Execute the `pkg_app` script in one of the following three ways:
 - `./pkg_app pid-of-running-application corefile`
 - `./pkg_app pid-of-the-running-application`
(The `pkg_app` scripts prompts for the `corefile` name.)
 - `./pkg_app core file`

1.8 Reporting Problems

Use the following email aliases to report problems on this document and its associated scripts:

- To provide feedback: gdd-feedback@sun.com
- To report problems: gdd-issue-tracker@sun.com

1.9 Accessing Sun Resources Online

The docs.sun.comSM web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. Books are available as online files in PDF and HTML formats. Both formats are made readable with the help of assistive technologies for users with disabilities.

To access the following Sun resources, go to <http://www.sun.com>:

- Downloads of Sun products
- Services and solutions
- Support (including patches and updates)
- Training
- Research
- Communities (for example, Sun Developer Network)

1.10 Third-Party Web Site References

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

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun 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. Sun 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.

1.11 Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to <http://docs.sun.com> and click Send Comments. In the online form, provide the full document title and part number. The part number is a 7-digit or 9-digit number that can be found on the book's title page or in the document's URL. For example, the part number of this book is 820-0429-10.

