Troubleshooting Application Crash or Core Dump

This document **(7004526)** *is provided subject to the disclaimer at the end of this document.*

Environment

Novell Open Enterprise Server 2 (OES 2) Linux
Novell Open Enterprise Server 1 (OES 1) Linux
SUSE Linux Enterprise Server 11
SUSE Linux Enterprise Desktop 11
SUSE Linux Enterprise Server 10
SUSE Linux Enterprise Desktop 10
SUSE Linux Enterprise Server 9

Situation

What is the procedure to gather and submit information regarding application core files?

Resolution

When an application crashes, the general procedure is:

1. Open a service request with Novell Technical Services (NTS)
2. Configure the application to dump a core image
3. Check the application's health using **chkbin**
4. Make sure the application has in fact dumped a core image to disk
5. Install the **novell-getcore** utility
6. Gather the system and application core information with **novell-getcore**
7. Gather application core image information
8. Submit the information to Novell Technical Services

For eDirectory specific instruction, refer to "Handling ndsd (eDirectory) core files on Linux and Solaris (3078409)"

For GroupWise specific instruction, refer to "How to prepare for and obtain a GroupWise Agent or Gateway core file on Linux (3447847)"

Resolution Details

1. Open a service request with Novell Technical Services (NTS)

Use the Novell's web portal to create a new Service Request

2. Configure the application to dump a core image

Refer to "How to obtain application core dumps (3054866)"

3. Check the application's health using **chkbin**
The **chkbin** utility comes in the supportutils package. It validates the RPM package that owns the application having trouble and each of the application's shared library dependencies. Reading an application core file is the least product troubleshooting step. The **chkbin** analysis files are saved in /var/log/nts_chkbin_* and are gathered by **supportconfig.** Information from **chkbin** dramatically reduces the problem resolution time. The following is **chkbin** output.

```
ipf-2:/var/spool/cron # chkbin /usr/sbin/cron
###############################################################
Binary Check Tool, v1.01-16
Date:   09/24/09, 13:55:33
Kernel: 2.6.5-7.319-default, Hardware: ia64
###############################################################

Checking Binary Ownership                                 ... Done
Checking for Shared Libraries                             ... Done
Checking for dlopen Libraries                             ... N/A
Checking Shared Library Ownership                         ... Done
Validating Unique RPMs                                    ... Warning
Fetching Environment Variables                            ... Done
Fetching Configuration Files                              ... Done
Including Shared Library Dependencies                     ... Done
Including System Library Cache                            ... Done
Including All Open Files                                  ... Done

###############################################################
Binary Checked: /usr/sbin/cron
Log File:       /var/log/nts_chkbin_cron_5860.txt
STATUS:         Warning
###############################################################
```

4. Make sure the application has in fact dumped a core image to disk

A core file will be saved into the application's current working directory, or the location specified by **sysctl kernel.core_pattern**. Supportconfig will look for application core files, and list them in the crash.txt file. An example from **supportconfig'scrash.txt** of core files found on a system follows:

```
#==[ Application Core Files ]=========================#
# Core File List in lsof-cwd, $PATH, ld.so.conf
-rw-------  1 root root 18423808 Aug 14 08:39 /core.10249
-rw-------  1 root root 94171136 Aug 14 08:40 /core.7118
-rw-------  1 root root   401408 Aug 14 08:40 /core.8044

#==[ Command ]====================================#
# /usr/bin/file /core.10249 /core.7118 /core.8044
/core.10249: ELF 32-bit LSB core file Intel 80386, version 1 (SYSV), SVR4-style, SVR4-style,
from 'smdrd'
/core.7118:  ELF 32-bit LSB core file Intel 80386, version 1 (SYSV), SVR4-style, SVR4-style,
from 'httpstkd'
/core.8044:  ELF 32-bit LSB core file Intel 80386, version 1 (SYSV), SVR4-style, SVR4-style,
from 'powersaved'
```

5. Install the **novell-getcore** utility

The **novell-getcore** utility gathers up the core file and copies of the application and it's shared libraries to assist NTS in analyzing the core image.

- Goto http://download.novell.com and search for Keyword: **novell-getcore**
- Download the latest version of **novell-getcore**
- Follow the installation instructions given
- Run **rpm -q gdb** to confirm the **gdb** package is installed

6. Gather the system information

- *First,* run **chkbin /path/to/application/that/coredumped**
- *Second,* run **supportconfig -ur <SR_number>** to gather system information, the **chkbin** logs, and automatically upload the tar ball to Novell's FTP incoming directory.
- If your server does not have direct Internet access, just run **supportconfig -r <SR_number>** and upload the tar ball as directed below.

7. Gather application core image information with **novell-getcore**

The following demonstrates how to use **novell-getcore** to create a tar ball with the application

core image that needs to submitted to NTS.

```
ipf-2:/var/spool/cron # ls -l
total 272
drwx------   4 root root    152 Sep 22 15:40 ./
drwxr-xr-x  11 root root    296 Dec 27  2007 ../
-rw-------   1 root root 442368 Sep 22 15:40 core.18705
-rw-------   1 root root     11 Jul  9  2007 deny
drwxr-xr-x   2 root root     80 Sep 24 13:30 lastrun/
drwx------   2 root root     48 Jul  9  2007 tabs/

ipf-2:/var/spool/cron # which cron
/usr/sbin/cron

ipf-2:/var/spool/cron # novell-getcore --createbundle /var/spool/cron/core.18705
/usr/sbin/cron
Novell GetCore Utility 1.1.37 [Linux]
Copyright (C) 2009 Novell, Inc. All rights reserved.

[*] User specified binary that generated core: /usr/sbin/cron
[*] Processing '/var/spool/cron/core.18705' with GDB...
[*] PreProcessing GDB output...
[*] Parsing GDB output...
[*] Core file /var/spool/cron/core.18705 is a valid Linux core
[*] Core generated by: /usr/sbin/cron
[*] Obtaining names of shared libraries listed in core...
[*] Counting number of shared libraries listed in core...
[*] Total number of shared libraries listed in core:  2
[*] Corefile bundle: core_20090924_134046_linux_cron_ipf-2
[*] Generating GDBINIT commands to open core remotely...
[*] Generating ./opencore.sh...
[*] Gathering package info...
[*] Creating core_20090924_134046_linux_cron_ipf-2.tar...
[*] GZipping ./core_20090924_134046_linux_cron_ipf-2.tar...
[*] Done. Corefile bundle is ./core_20090924_134046_linux_cron_ipf-2.tar.gz

ipf-2:/var/spool/cron # ls -l
total 1177
drwx------   4 root root    216 Sep 24 13:40 .
drwxr-xr-x  11 root root    296 Dec 27  2007 ..
-rw-------   1 root root 442368 Sep 22 15:40 core.18705
-rw-r--r--   1 root root 922097 Sep 24 13:40 core_20090924_134046_linux_cron_ipf-2.tar.gz
-rw-------   1 root root     11 Jul  9  2007 deny
drwxr-xr-x   2 root root     80 Sep 24 13:30 lastrun
drwx------   2 root root     48 Jul  9  2007 tabs
```

8. Submit the information to Novell Technical Services

- If you ran **supportconfig** with the -u switch, the tar ball was automatically uploaded to

Novell's anonymous FTP server.

- Upload the **novell-getcore** and **supportconfig** tar balls to Novell's anonymous FTP server (ftp.novell.com:/incoming/) or attach it directly to the service request using the web portal (https://secure-www.novell.com/center/eservice/).

Disclaimer

This Support Knowledgebase provides a valuable tool for NetIQ/Novell/SUSE customers and parties interested in our products and solutions to acquire information, ideas and learn from one another. Materials are provided for informational, personal or non-commercial use within your organization and are presented "AS IS" WITHOUT WARRANTY OF ANY KIND.