LKCD Installation and Configuration

IBM Global Services
Linux Education

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM

Other company, product, and service names may be trademarks or service marks.

About This Lesson

Purpose

This lesson describes how to install and configure the **L**inux **K**ernel **C**rash **D**ump facility (**LKCD**).

What is not covered

This lesson will not describe how to use LKCD tools to analyze a Linux crash dump. It will tell you how to obtain a crash dump for later analysis.

Prerequisites

You need to know how to build, install and boot a Linux kernel.

Table of contents

This document covers the following topics:

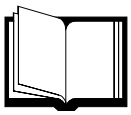
Continued on next page

About This Lesson, Continued

Objectives

At the completion of this lesson you will be able to:

- Describe the purpose of the LKCD
- Install the LKCD on a Linux system
- Configure the LKCD to take a system crash dump
- Verify the validity of a system crash dump file.



REFERENCE

Additional information about the LKCD and the LKCD project can be found at:
http://lkcd.sourceforge.net/

Author

David Wilder of the IBM® Global Service Linux Change Team wrote this lesson. Please send comments and questions to wilder@us.ibm.com.

Disclaimer

This work represents the view of the author, and does not necessarily represent the view of IBM.

Introduction to the LKCD

Introduction

Often when a Linux system fails it is necessary to preserve an image of system memory so that a post-analysis of the failure may be preformed. Once the preserved image (called a crash dump) is saved to disk the system can be returned to production.

What is the LKCD?

The **L**inux **K**ernel **C**rash **D**ump (LKCD) project has created a set of utilities and kernel patches that allow a crash dump to be captured. Tools are included that allow kernel developers to analyze these crash dump.

When should I install the LKCD?

The LKCD must be installed before a failure occurs! Systems administrators should be encouraged to install the LKCD during the initial set-up of their system. Waiting until after a failure to install the LKCD will require the failure to be duplicated thus delaying the time to resolution.

When is a crash dump taken?

Once the LKCD is installed a crash dump will automatically be created when:

- A kernel Oops occurs
- A kernel panic occurs
- The system administrator initiates a crash dump by typing **Alt-SysRq-c** on the console.

What platforms will the LKCD support?

The following architectures are supported in version 4 of the LKCD:

- i386
- ia64
- alpha

Continued on next page

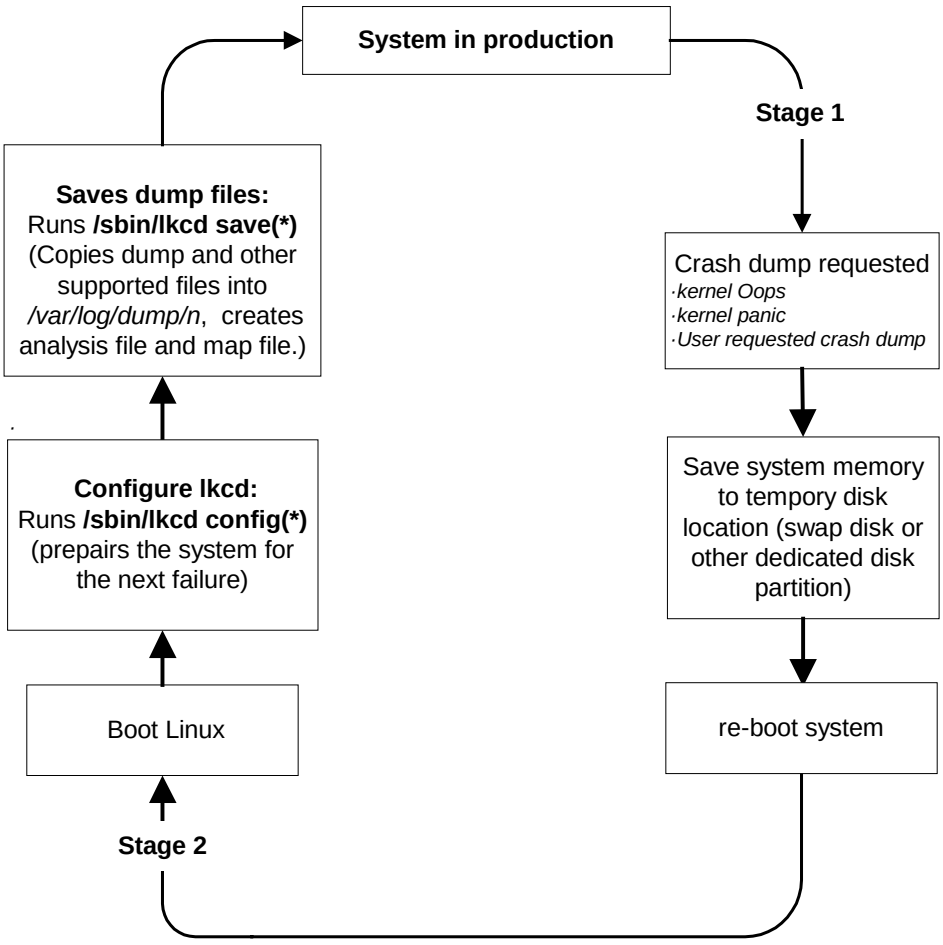
The Crash Dump Process

Introduction

The process used by the LKCD to create a crash dump is discussed in this section.

Process

The process to obtain a crash dump involves two stages. In *stage one*, the contents of the systems memory is copied into a temporary disk location called a dump device. In *stage two*, Linux is booted and the memory image previously saved in the dump device is moved to a permanent location in the directory */var/log/dump*. Both the locations of the dump device and the dump directory are configurable.



* **lkcd** is run by the */etc/rc.sysinit* script in Redhat and Turbo distributions. In SuSE distributions **lkcd** is run by */sbin/init.d/boot*.

Continued on next page

The Crash Dump Process, Continued

What is saved?

After crash dump processing has completed the files in this table will have been created. These file are saved in a sub-directory of */var/log/dump.* The sub-directories are named 0, 1, 2, etc. A new numbered directory is created for each new dump.

File	Description
dump.*n*	The crash dump file. *n* will be incremented for each new crash dump file. The file */var/log/dump/bounds* holds the value of n for the "next" crash dump.
kerntypes.*n*	This file contains information about the data structures that are used by the linux kernel. Kerntypes is initially created when the kernel is built. Its copied into **/boot** during kernel installation. Following a system crash a copy is made in */var/log/dump.*
map.*n*	This file contains symbol table information from the Linux kernel. It is a copy of /boot/System.map.
analysis.*n*	This file is automatically generated following a system crash. It contains a textual analysis of the crash that may be helpful in the crash analysis.
lcrash.*n*	A copy of the **lcrash** utility found in **/sbin** at the time of the crash.
bounds	Contains the name of the dump sub-directory to be created on the next dump.

Non-disruptive dumps

LKCD provided the option to perform non-disruptive dumps. This means that the system will continue to run (if possible) after processing a dump. When configured for non-disruptive, the dump process stops prior to the system re-boot in stage 1. A non-disruptive dump is only possible for a kernel Ooops or when the Alt-SysRq-c key is presses (version 4.01 or later). A panic is always a disruptive operation therefor the system will re-boot after a panic even if configured for non-disruptive dumps.

Notes:

1. The **lkcd save** command must be run after a non-disruptive dump has occurred to create the dump files.

2. A dedicated dump device must be used when configuring for non-disruptive dumps (see section on Selecting a Dump Device on page 9).

Commands, Scripts, and Files

Introduction

This section introduces the commands, scripts, utilities and other important files that are included or used by the LKCD package.

Commands and scripts

The commands and scripts used to configure and run the LKCD package are:

Command	Function
/sbin/lkcd	This script is used to configure and save a crash dump. It accepts two arguments: **config** or **save. Config** is used to set system dump parameters into the running kernel. **Save** is used to save a crash dump on disk (stage two processing).
/etc/sysconfig/dump	Contains configuration variables for the LKCD. These variables are used by the **lkcd config** command. Editing this file sets dump configuration.
/sbin/lkcd_config	This binary is called by **/sbin/lkcd** to set system dump parameters.
/sbin/lcrash	Linux crash dump analyzer.

Other files and directories

Several other important files and directories are described here:

File	Function
*/proc/sys/dump/**	Files in this directory can be used to view the kernel's current crash dump configuration. One file is created in this directory for each configuration parameter found in */etc/sysconfig/dump*. Use the **cat** command to read the files.
lkcd-x.x.x.diff	Kernel patches supplied by the LKCD. Separate patch files have been created for each version of the Linux kernel, *x.x.x* represent the kernel version.
Kerntypes	A file containing kernel type information that **lcrash** needs in order to properly access kernel data structures in a crash dump. This file is built as part of the kernel build process. A copy must be placed into */boot* when installing the kernel
System.map	Contains symbol table information needed by **lcrash** to reference symbolic names when accessing a crash dump. This file is built as part of the kernel build process. When the kernel is installed a copy is placed into */boot*.

Selecting a Dump Device

Introduction

When configuring LKCD you must select a disk partition to be used as a dump device. The dump device is the temporary storage used by LKCD for stage one processing. You may use a swap partition or dedicate an unused disk partition as a dump device.

Using a swap partition

By default LKCD will use the first swap partition as the dump device. If the swap partition is to be used it must be large enough to hold the entire dump.

Dedicating dump devices

Dedicating disk partitions for the exclusive use of system crash dumps is highly recommended. Dedicated dump device will speed-up the recovery time following a system crash. When a swap partition is used system startup must wait for the dump to be copied from the swap partition before enabling the partition as swap. When a dedicated dump partition is used the copy may be run in the background while system processing continues or later after the system is back up. This may provide considerable advantage on a large memory system.

Warning

The disk partition you select will be erased when a dump is created. Insure the partition you select is not used for any other purpose. The only exception is when a swap partition will be used as a dump device.

Non-disruptive dumps

A dedicated dump device must be used when configuring for non-disruptive dumps.

How much disk space is needed?

The dump device must be at least as large as the amount of physical memory on the system. For example if your system has 128MB of physical memory the dump partition must be at least:

128MB * 2048 blocks/MB = **262144** blocks (512 bytes/block).

To determine the amount of physical memory run the command:
cat /proc/meminfo

Continued on next page

Selecting a Dump Device, Continued

Finding a partition

Available disk partitions and sizes may be listed using the command:

#**cat /proc/partitions**

Using compression

Some versions of the LKCD provide the ability to compress a crash dump when writing to the dump device. Using compression allows a smaller dump device to be used. However, the exact amount of compression may very between dumps making the sizing of the dump partition difficult.

Configuring a dump disk

Once a disk partition has been selected, configure the dump device as shown here. Substitute the partition name you have selected for *hdb1* in the following example. Note: LKCD must be fully installed before configuring a dump device.

Step	Action
1	Logon or su to root.
2	Build a symbolic link to the partition. # **ln -s /dev/hdb1 /dev/vmdump** Note: If you are using first swap partition as a dump device this link will be automatically built by the **lkcd** command in the next step.
3	Tell the kernel to update the dump configuration with the command: # **/sbin/lkcd config**

Configuration Options

Introduction

In addition to selecting a dump device several LKCD configuration options are available to you. This section discusses these options.

Activating the lkcd configuration

The lkcd configuration settings are activated by the command **/sbin/lkcd config**. This command is typically run by */etc/rc.sysinit* at system startup (*/sbin/rc.d/boot* on SuSE distributions).

Options

Edit the file */etc/sysconfig/dump* to select the LKCD options described in this table. On a typical installation you will *not* need to modify the default options.

Option	Function	Default (v4)
DUMP_ACTIVE	Set to 1 to activate the dump process. A value of 0 disables dumping.	1
DUMPDEV	Represents the name of the dump device used in stage one dump processing. Generally set to */dev/vmdump*, a symbolic link to the actual dump device	*/dev/vmdump*
DUMPDIR	The directory where crash dumps will be saved in stage 2.	*/var/log/dump*
DUMP_SAVE	Defines whether to save the memory image to disk or not. If the value is 1, the dump image is stored, and a crash report is created from the saved dump. If it is not set to 1, only a crash report will be created, and the dump will not be saved. Saving only reports is not recommended. This option can be used on systems that do not want their disk space consumed by large crash dump images.	1
DUMP_LEVEL	In version 4 the valid setting for dump level are: 0 - Do nothing, just return if called. 1 - Dump the dump header and first 128K bytes out. 8 - All memory. Future enhancements: 2- Dump the header, the first 128K bytes and only the kernel pages. 4 - Everything except kernel free pages.	8

Continued on next page

Configuration Options, Continued

Options ….Continued

Option	Function	Default (v4)
DUMP_FLAGS	Flag parameters to use when configuring system dumps. In version 4 the only valid settings are: 0 - No flags are required. 1 - Non-disruptive, do not reboot after dumping; continue running.	0
DUMP_COMPRESS	Indicates which compression mechanism the kernel should attempt to use for compression (see below for more on dump compression).	0
PANIC_TIMEOUT	Represents the timeout (in seconds) before reboot after a panic occurs. If set to 0 the kernel sits and spins until someone resets the machine. This is not the preferred action if we want to recover the dump after the reboot.	5

Sample file This is a sample of a */etc/sysconfig/dump* file:

```
DUMP_ACTIVE=1
DUMPDEV=/dev/vmdump
DUMPDIR=/var/log/dump
DUMP_SAVE=1
DUMP_LEVEL=8
DUMP_FLAGS=0
DUMP_COMPRESS=0
PANIC_TIMEOUT=5
```

Dump Compression Two types of compression are available in the LKCD, RLE and GZIP. To use compression one of the LKCD compression kernel modules must be built. These modules can be compiled into the kernel or as separately loaded modules. I recommend build all available compression modules into the kernel. Once the kernel is built compression can be selected or deselected by changing the value of the DUMP_COMPRESS options, available values are:

- 0 - Don't compress this dump.
- 1 - Use RLE compression.
- 2 - Use GZIP compression.

Installing LKCD

Introduction

This section describes how to install and configure the LKCD.

Pre-requests

Installing LKCD requires that a new Linux kernel be built. Before installing the LKCD insure that the kernel sources matching your running kernel are installed on the system. It is a good idea to verify that a kernel can be built, installed and booted before installing the LKCD patches.

Obtaining the LKCD files

Version 4 of the LKCD files can be downloaded from the following site:
http://lkcd.sourceforge.net/

Two files must be downloaded

- LKCD kernel patches
- lkcdutils rpm (source or binary)

Check the LKCD site for the current version.

LKCD kernel patches

The LKCD kernel patches must match the version of the Linux you are running. If patches are not available for your version of the kernel you will need to modify the patches for you version. Modifying these patches is beyond the scope of this lesson. If you are not familiar with kernel programming obtain help from an expert.

LKCD utilities

If you are running on an i386 (Intel) platform a binary rpm is available. For all other platforms you will need to obtain the source rpm.

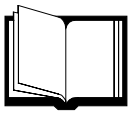
Continued on next page

Installing LKCD, Continued

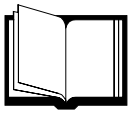
Three part installation

There are three stages to the installation of the LKCD. The three stages are described in this table:

Stage	Description
1	Install the LKCD kernel patches and build a new kernel. This will add the capability to dump system memory to the dump device.
2	Install LKCD utilities. This adds all the commands and scripts required to configure LKCD, create the dump files and analyze crash dumps.
3	Edit the system startup scripts. Depending on the distribution you are running different files will need to be edited. These edits will insure that LKCD will be automatically configured and crash dumps will be saved after a system crash.



REFERENCE
More information on the patch utility can be found in the online manual pages: **patch(1)**



REFERENCE
For more information on building Linux kernels see the file: */usr/src/linux/README.*

Continued on next page

Installing LKCD, Continued

stage 1 - Building an LKCD kernel

Once you have obtained the LKCD utilities and the kernel patches you are ready to start the installation. Follow the steps in this table to patch and build an LKCD kernel.

Step	Action
1	Logon or **su** to the root user.
2	Insure you can configure, build and boot a new kernel on the target system before proceeding. If something goes wrong with your kernel build you will know that the LKCD patches are not the cause of your trouble.
3	Locate the kernel source directory and make a copy of it: # **cp -r /usr/src/linux-***x.x.x* **/usr/src/linux-***x.x.x***.lkcd** *x.x.x* is the kernel version
4	Move to the directory you just created # **cd /usr/src/linux-***x.x.x***.lkcd**
5	In this step you will test if the LKCD patches can be applied to your kernel without any errors. If errors are reported by the following command stop and seek help. # **patch -p1 --dry-run <** <*path*>/**lkcd-***x.x.x***.diff** *path* - pathname to the directory containing the patch file x.x.x - Kernel version
6	If the last step did not report any errors, apply the LKCD kernel patches using the following command: # **patch -p1 <** <*path*>/**lkcd-***x.x.x***.diff** *path* - pathname to the directory containing the patch file x.x.x - Kernel version If the command reports any errors stop and seek help.

Continued on next page

stage 1 - Building an LKCD kernel

….Continued

<table>
<tr><td>7</td><td>Configure the kernel adding LKCD support (compiled into kernel not as a module) and enabling <code>Magic SysRq Keys*</code>. You may use any kernel configuration program you wish. The following example uses menuconfig.
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>A</td><td>Type the command:
make menuconfig</td></tr>
<tr><td>B</td><td>When the menu appears navigate to:
<code>Kernel Hacking</code> and type <enter></td></tr>
<tr><td>C</td><td>Select <code>kernel debugging</code> and type: <space></td></tr>
<tr><td>D</td><td>Navigate to <code>Magic SysRq key</code> and type <space> an asterisk should appear next to the line <code>Magic SysRq key.</code></td></tr>
<tr><td>E</td><td>Navigate to Linux Kernel Crash Dump (LKCD) and type <space> until an asterisk appears. If compression options are presented select all available.</td></tr>
<tr><td>F</td><td>Press <tab> <enter> twice until you are prompted to save your configuration. Type <enter> to save and exit menuconfig.</td></tr>
</table></td></tr>
<tr><td>8</td><td>Run the following command to build the new kernel:
make dep; make bzImage</td></tr>
<tr><td>9</td><td>Once the kernel has been built you will need to install the kernel image. Each distributions and system platform may have different procedures to install a kernel. Refer to the documentation that came with for your distribution. On many systems you can simply type:
make install</td></tr>
</table>

* It is not a requirement of the LKCD to enable Magic SysRq Keys; however, enabling this feature will allow a crash to be created when a system has hung. You will also use this feature later in the lesson to test your LKCD configuration.

Continued on next page

stage 1 - Building an LKCD kernel

Continued…..

10	The kernel build process will have built the file *Kerntypes* in the kernel source directory. Verify that this file was copied to */boot*. If needed copy this files yourself: # **cp Kerntypes /boot**
11	The kernel build process builds the file *System.map* in the kernel build directory. The kernel install process copies this file into */boot*. Verify that */boot/System.map* matches the copy in the kernel source directory: # **diff System.map /boot/System.map** If the two files do not match make a fresh copy in */boot*: # **cp System.map /boot** Note: some system may have symbolic links in */boot*. Take care not to break these links when copying this file.
12	Boot the new kernel. # **/sbin/init 6**
13	Once the system has booted verify that the directory */proc/sys/dump* exists by typing the command: # **ls -d /proc/sys/dump** This directory is automatically created when an LKCD kernel is booted. If the directory is missing the kernel has not been patched, or configured properly for LKCD. Or you have booted the wrong kernel. Check your work before proceeding.

Continued on next page

Stage two - installing the lkcdutils

The lkcdutils can be installed using ether the binary rpm or the source rpm. Note: the binary rpm is only available for the i386 (Intel) platform.

Step	Action
1	To install lkcdutils from the binary rpm use the following command: # **rpm -i lkcdutils-4_0-1_i386.rpm**
2	To install from the source rpm install the source rpm and run the following commands. # **cd** *<lkcdutils source directory>* # **configure** # **make** # **make install**

Continued on next page

Stage three - edit system startup scripts

In this step you will edit the system startup scripts to configure LKCD and save crash dumps.

Step	Action

1	Locate the appropriate system startup script from your distribution. For Redhat and Turbo the script is */etc/rc.sysinit*. In SuSE the file is */sbin/init.d/boot*. If you are unsure of what file to use examine the file */etc/inittab*. Look for the line starting with si: the script named on this line is the file you will be modifying. For example:
	.
	`# System initialization.`
	`si::sysinit:/etc/rc.d/rc.sysinit`
	.
	.

2	Examine the script identified in the last step. Locate the line that mounts all file systems except *root* (and possibly */proc*). For example in the Redhat *rc.sysinit* script you will find the lines:
	`# Mount all other filesystems (except for NFS and /proc, .)`
	`action $"Mounting local filesystems: "mount -a -tnonfs,smbfs,ncpfs`

3	Following these lines add the following text: `/sbin/lkcd config`

4	If you are using a swap partition as the dump device the dump must be saved before swap is activated. Locate the line with the **swapon** command in the script and comment it out. Place a **lkcd save** command following **lckd config** command then place the **swapon** command following. You file should look something like this:
	
	`/sbin/lkcd config`
	`/sbin/lkcd save`
	`# Start up swapping.`
	`action $"Activating swap partitions: " swapon -a -e`

Continued on next page

Installing LKCD, Continued

Setting the dump device

The last step is to configure your dump device.

Step	Action
1	Locate the disk partition you will use as the dump device. In this example I used */dev/hdb1*.
2	Build a symbolic link to this partition. # **ln -s /dev/hdb1 /dev/vmdump**
3	Tell the kernel to update the dump configuration with the command: # **/sbin/lkcd config**

Testing Your LKCD Installation

Introduction

Once you have installed and configured LKCD it is important to verify that a crash dump can be taken. This section describes a procedure to test the LKCD installation.

Warning

Forcing a crash dump will cause the system to crash. Insure that all production has been shutdown before testing the LKCD. It is a good idea to **umount** any unneeded filesystems.

Test procedure

Follow the steps in this table to test crash dump configuration. The following procedure assumes that *Magic SysRq key* has been enabled in your kernel.

Step	Action
1	Logon or **su** to root.
2	To verify that LKCD has been enabled in the currently running kernel check for the existence of the directory */proc/sys/dump*. # **ls -d /proc/sys/dump** If this directory is missing LKCD has not been properly built into the kernel. Do not proceed with this procedure.
3	Enable the Magic SysRq key with the following command: # **echo 1 > /proc/sys/kernel/sysrq**
4	If the console is running the X environment type ctrl-Alt-F1.
5	On the system console type: **Alt-SysRq-u** (hold all three key down at the same time) This will cause all filesystems to be re-mounted as read-only. This saves the system from running **fsck** on all the file systems when the system reboots.
6	On the system console type: **Alt-SysRq-c** (hold all three key down at the same time) This will force the system to panic and a crash dump to be taken. You should see the following message on the console indicating stage one of dump processing is occurring. `Dump: dumping to device xxxxxx`
7	The system should now re-boot
8	If your system startup scripts don't contain the **lkcd save** command run this command as the root user to create the dump files. # **/sbin/lkcd save**
9	Verify that the appropriate file have been created in */var/log/dump/n*.

Verifying a Crash Dump

Introduction

It is often a good idea to verify the validity of a crash dump before engaging help to analyze the crash. This section describes how to verify that a crash dump is valid.

lcrash

lcrash is a utility that generates detailed kernel information about crash dumps. **lcrash** also provides the ability to generate reports about system crash dumps.

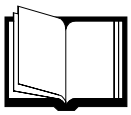
Running lcrash

lcrash requires three file:

- The crash dump itself
- A copy of /boot/System.map
- A copy of /boot/Kerntypes

To start **lcrash** use the command:
$ **/sbin/lcrash** *map dump kerntypes*

The three required files are usually saved in the dump directory with the names: dump.*x*, map.*x and* kerntypes.x. In this case **lcrash** can be started with the command:
$ **/sbin/lcrash -n** ***x***



REFERENCE
More information on the **lcrash** utility can be found in the online manual pages: **lcrash(1)**

Continued on next page

Verifying a Crash Dump, Continued

Testing a dump Follow the procedure in this table to verify that a crash dump is valid

Step	Action
1	Locate the directory containing your system crash dump. Each time a crash dump is created a new directory is created. For example the first crash dump is saved in */var/log/dump/0* the second */var/log/dump/1*. The file */var/log/dump/bounds* contains an index number used to name the directory and to name the dump files. Each time a dump is created the index number is incremented.
2	Verify that all required files were created for the new crash dump: $ **ls /var/log/dump/0** `analysis.0  dump.0  kerntypes.0  lcrash.0  map.0`
3	Move to the directory containing the crash dump: $ **cd /var/log/dump/***x*
4	Start **lcrash** on the crash dump: $ **/sbin/lcrash -n** *x* *x* is the crash directory number.
5	After a few seconds you will see the **lcrash** prompt: >>
6	Issue the command **ps** to get a listing of the processes that were running at the time of the crash.
7	Issue the command **trace** to display a stack trace from the time of the crash.
8	Type **q** to exit **lcrash**.