3 Evaluation of Dump Tools

3.1 Dump collection Tools

This chapter introduces the dump generating tools diskdump, netdump, LKCD, kdump, and mkdump, then compares them (3.1.6), and considers the dump generating tools in several aspects (3.1.7).

3.1.1 diskdump

■ Overview

The diskdump tool was developed by Fujitsu. When a failure occurs, this tool generates a crash dump on a disk. To use this tool, the SCSI driver in the kernel needs to be modified. In the Red Hat Linux distribution, diskdump is included as a standard so that it is not necessary to rebuild the kernel.

The dump file generated by this tool can be analyzed with the crash, lcrash, and Alicia utilities. When using the crash, it is necessary to build the kernel with a debug option.

http://sourceforge.net/projects/lkdump/

■ Installation Environment

Kernel: kernel 2.6.9-11.EL

Machine Architecture: i386 (DELL PowerEdge 2600)

Operating System: Red Hat Enterprise Linux AS 4 update1

Memory: 2 GB

■ Supported Dump Analysis Tools

crash, lcrash, and Alicia

■ Installation Procedure

(1) Checking diskdump Package

```
# rpm -qa |grep diskdump
diskdumputils-1.0.1-5
```

(2) Preparing Disk

The diskdump tool can generate a dump on disk devices that are accessed with the

following drivers:

aic7xxx, aic79xx, mpt fusion, megaraid, ata_piix, and sata_promise

(3) Creating a Dump Partition

Create a partition on the disk that will store the dump file. The size of the partition should be 1.5 times the actual memory capacity. In this chapter, the partition for diskdump is created on /dev/sdb1.

(4) Setting a Dump Device

Modify /etc/sysconfig/diskdump to describe a dump device name.

```
DEVICE=/dev/sdb1
```

(5) Initializing a Dump Device

Use the following command to format the partition registered in /etc/sysconfig/diskdump with the dump disk format:

```
# service diskdump initialformat
```

(6) Starting the diskdump Service

Execute the following command: The diskdump service will be launched upon system startup.

```
# chkconfig diskdump on
```

Use the following command to start the diskdump service:

```
# service diskdump start
```

The /proc/diskdump is created. If the service started normally, this file should contain information about the dump device.

```
# cat /proc/diskdump
/dev/sdb1 63 39086082
```

■ Checking Operation

(1) Enabling the SysRq Magic Key

```
# echo 1 > /proc/sys/kernel/sysrq
```

(2) Generating the Dump

```
# echo c > /proc/sysrq-trigger
```

After the tool generates a dump, it halts the system.

(3) Checking the Dump File

The tool generates a folder with the name "client_IP_address-date_and_time_of_generation" in /var/crash (e.g., /var/crash/192.168.0.2-2005-09-09-21:59) on the server in which the dump file is saved as a vmcore file.

3.1.2 netdump

■ Overview

The netdump tool was developed by Red Hat. It can generate a crash dump through a network. This tool is included in Red Hat Linux and MIRACLE LINUX (V3.0 SP1 or later) as a standard. It is implemented in a client-server configuration that consists of a client that generates a dump and a server that receives and saves the dump through a network.

The dump file generated by this tool can be analyzed with the crash, lcrash, and Alicia utility. When using the crash, it is necessary to build the kernel with the debug option.

■ Installation Environment

<Client>

Kernel: kernel 2.6.9-11.ELsmp

Machine Architecture: i386　DELL PowerEdge 2600

Operating System: Red Hat Enterprise Linux AS 4 update1

Memory: 2 GB

<Server>

Kernel: kernel 2.6.9-11.EL

Machine Architecture: i386 (DELL OptiPlex GX270)

Operating System: Red Hat Enterprise Linux AS 4 update1

Memory: 512 MB

■ Supported Dump Analysis Tools

crash, lcrash, and Alicia

■ Installation Procedure (Server)

(1) Checking Installed Packages

The communication between the netdump server and client uses SSH authentication. Since netdump-server has dependency on ssh-server, the netdump package must be installed after installing ssh.

```
# rpm -qa |grep ssh-server
openssh-server-3.9p1-8.RHEL4.4
# rpm -qa |grep netdump-server
netdump-server-0.7.7-3
```

If the required packages are not present, install them from the CD, etc..

(2) Setting a Password

The default netdump user is automatically created when netdump-server is installed. It is necessary to change the password of the netdump user. This password is required for authenticating the client.

```
# passwd netdump
Changing password for user netdump.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

(3) Checking Service Invocation

Start the ssh and netdump-server services manually and check the system log to confirm that they are in operation successfully.

```
# service sshd start
   Starting sshd:                                          [ OK ]
# service netdump-server start
   Starting netdump-server                                 [ OK ]
# tail /var/log/messages
(Omitted)
Sep   9 21:10:38 rhas4u1ipa sshd:   succeeded
Sep   9 21:10:45 rhas4u1ipa netdump-server: netdump-server startup succeeded
```

(4) Setting Automatic Service Invocation

After confirming that the services can be invoked manually, set up automatic

invocation that will be available after rebooting the system.

```
# chkconfig sshd on
# chkconfig netdump on
# chkconfig –list sshd
   sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
# chkconfig –list netdump-server
   netdump-server      0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

■ Installation Procedure (Client)

(1) Checking Installed Packages

```
# rpm -qa |grep netdump
netdump-0.7.7-3
```

If the required package is not present, install it from the CD, etc..

(2) Editing the Setting File

Edit /etc/sysconfig/netdump to set the network interface card (NIC) used for sending a dump and the IP address of the target server.

```
DEV=eth0
NETDUMPADDR=192.168.0.4
```

NETDUMPADDR accepts an IP address in dotted quad form. Since name resolution with a name server is not supported, it is impossible to assign a host name to this variable.

(3) Setting the DSA Public Key

Register a DSA public key in netdump-server. Connect the client and server through a network and, on the server, set up netdump-server and start the service. Then execute the command in the following listing:

```
# service netdump propagate
  netdump@nn.nn.nn.nn's password:XXXXXXXXXXXXX
```

nn.nn.nn.nn indicates the IP address of the server.

Enter the netdump user password in place of XXXXXXXXXX.

(4) Starting the Services

Start the netdump service manually and check the system log to confirm its operation.

```
# service netdump start
   initializing netdump                                    [ OK ]
   initializing netconsole local IP nn.nn.nn.nn       [ OK ]
# tail /var/log/messages
(Omitted)
Sep   9 21:28:57 rhas4u1ipa netdump: initializing netdump succeeded
Sep   9 21:28:57 rhas4u1ipa netdump: initializing netconsole succeeded
Sep   9 21:28:57 rhas4u1ipa kernel: netconsole: network logging started
```

(5) Setting Automatic Service Invocation

After confirming that the services can be invoked manually, set up automatic invocation that will be available after rebooting the system.

```
# chkconfig netdump on
# chkconfig –list netdump
   netdump              0:off    1:off    2:on     3:on     4:on     5:on     6:off
```

■ Checking Operation

(1) Enabling the SysRq Magic Key

```
# echo 1 > /proc/sys/kernel/sysrq
```

(2) Generating the Dump

```
# echo c > /proc/sysrq-trigger
```

The dump is transferred to the server through the network. After generating the dump, the client is automatically rebooted.

(3) Checking the Dump File

The tool creates a folder with the name "client_IP_address-date_and_time_of_generation" in /var/crash (e.g., /var/crash/192.168.0.2-2005-09-09-21:59) on the server in which the dump file is saved as a vmcore file.

3.1.3 LKCD

■ Overview

Linux Kernel Crash Dump (LKCD) was developed by Silicon Graphics, Inc. (SGI) and released under the GPL license to provide a memory dump generating function and a memory dump analysis tool (lcrash utility). Since then, this tool has been enhanced in

a joint effort by IBM, NEC, Hitachi, and Fujitsu.
MIRACLE LINUX V3.0 installs LKCD from the minimum configuration.

http://lkcd.sourceforge.net/

■ Installation Environment
Kernel: kernel 2.4.21-20.19AXsmp
Machine Architecture: i386　DELL PowerEdge 2600
Operating System: MIRACLE LINUX V3.0
Memory: 2 GB

■ Supported Dump Analysis Tools
crash, lcrash, and Alicia

■ Installation Procedure
(1) Checking the LKCD Installation

```
# rpm -qa | grep lkcd
        lkcdutils-4.2-10AX
        lkcdutils-devel-4.2-10AX
```

(2) Creating a Dump Partition
Create a partition on the disk that will store a dump file. The size of the partition should be 1.5 times the actual memory capacity.
After allocating the dump partition, make a symbolic link to /dev/vmdump. In this chapter, the dump partition is created on /dev/sdb1. Using the default swap partition for the dump output allows fast system booting.

```
# ln -s /dev/sdb1 /dev/vmdump
```

(3) Setting LKCD
Edit /etc/sysconfig/dump to set up LKCD.

```
DUMP_ACTIVE=1
DUMPDEV=/dev/vmdump
DUMPDIR=/var/log/dump
DUMP_SAVE=1
DUMP_LEVEL=8
DUMP_FLAGS=0x80000000 # disruptive disk dump is default for 2.4
```

```
DUMP_COMPRESS=2
PANIC_TIMEOUT=5
```

• DUMP_ACTIVE

⇒ This parameter specifies whether to enable LKCD. The default is "1."

1: Enables LKCD.

0: Disables LKCD.

• DUMPDEV

⇒ This parameter specifies the device used for saving the memory contents. The default is /dev/vmdump/dev/vmdump, which is a symbolic link to the swap device.

• DUMPDIR

⇒ This parameter specifies the directory to which the dump will be saved. The default is /var/log/dump.

• DUMP_SAVE

⇒ This parameter specifies whether to perform both dump generation and crash reporting (dump summary display). The default is "1."

1: Enables both dump generation and crash reporting.

Other than 1: Enables only crash reporting.

• DUMP_LEVEL

⇒ This parameter specifies the area to be dumped. The default is "8."

0 (DUMP_NONE): Does not dump.

1 (DUMP_HEADER): Dumps the dump header that is the first 128 bytes.

2 (DUMP_KERN): Dumps the dump header and kernel pages in use.

4 (DUMP_USER): Dumps the dump header, kernel pages in use, and user pages.

8 (DUMP_ALL): Dump everything (including kernel pages not in use).

• DUMP_FLAGS

⇒ This parameter specifies whether to reboot after dump generation. The default is "0."

0 (DUMP_FLAG_NONE): Reboots the system.

1 (DUMP_FLAGS_NONDISRUPT): Does not reboot the system.

• DUMP_COMPRESS

⇒ This parameter specifies whether to compress dump data and the compression method. The default is "2."

0 (DUMP_COMPRESS_NONE): Does not compress dump data.

1 (DUMP_COMPRESS_RLE): Compresses dump data using the REL algorithm.

2 (DUMP_COMPRESS_GZIP): Compresses dump data using the GZIP algorithm.

• PANIC_TIMEOUT

⇒ This parameter specifies the kernel behavior after dump generation. The default is "5."

0: The kernel will keep a spin-loop. (Will not reboot itself.)

Other than 0: The kernel will reboot itself in a specified time (in seconds) after dump generation.

(4) Activating the LKCD configuration

After setting LKCD, execute the following command so that the activating the LKCD configuration.

```
# lkcd config
```

(5) Checking the Contents of the Dump Directory

The dump directory specified by the DUMPDIR parameter consists of the following directory and file:

• Dump file directory (DUMPDIR/xxx, where "xxx" is a number.)

• bounds file

The bounds file contains the number that will be used for saving the next dump file. For example, if the contents of the bounds file is '1,' when a kernel panic occurs, the dump file will be saved in the /var/log/dump/1 directory. The contents of the bounds file then will be incremented to '2.'

The dump file directory will contain the following items: 'n' is a number that indicates the name of the directory created under /var/log/dump/.

• dump.n

⇒ This is a crash dump file. (In LKCD format)

• kerntypes.n

⇒ This file will contain the data structure of the Linux kernel. (In binary format)

The Kerntypes file is usually created when the Linux kernel is compiled and saved in the /boot directory as Kerntypes-*kernel_version*.

The kerntypes.n file is a copy of the Kerntypes- *kernel_version* file.

Example: /boot/Kerntypes-2.4.21-20.19AXsmp -> Kerntypes.n

• map.n

⇒ This file will contain the symbol table (external reference) of the Linux kernel. (In text format)

The map file is usually created when the Linux kernel is compiled and saved in the /boot directory as System.map-*kernel_version*. The map.n file is a copy of the System.map-*kernel_version* file.

Example: /boot/System.map-2.4.21-20.19AXsmp -> map.n

• analysis.n

⇒ This is a crash report file that will be created automatically. (In text format)

• lcrash.n

⇒ This file is a copy of the lcrash utility that resides in /sbin. (In binary format)

■ Checking Operation

(1) Enabling the SysRq Magic Key

```
# echo 1 > /proc/sys/kernel/sysrq
```

(2) Generating the Dump

```
# echo c > /proc/sysrq-trigger
```

After generating a dump, the kernel will reboot itself if the PANIC_TIMEOUT parameter is other than 0.

(3) Checking the Dump File

Check that files are created in the /var/log/dump directory.

3.1.4 kdump

■ Overview

kexec is a utility that quickly boots the system and starts another kernel by skipping firmware and boot loader operations that take much time to reactivate the system. The kdump tool uses kexec to start a dump generation kernel and allow accessing of the failed memory space when a failure occurs.

In the following description, the kernel that will be dumped is called the first kernel, and the kernel that will perform the dump generation, the second kernel. After the first kernel starts, the kexec command loads the second kernel into a memory space. When the first kernel crashes, the second kernel is started.

http://lse.sourceforge.net/kdump/

■ Installation Environment

Kernel: kernel 2.6.13

Machine Architecture: i386　DELL PowerEdge 2600

Operating System: Red Hat Enterprise Linux AS 4 update1

Memory: 2 GB

■ Supported Dump Analysis Tools

gdb

■ Installation Procedure

(1) Copying Files

Download the following files and copy them to any directory:

- kexec-tool Source Code　　kexec-tools-1.101.tar.gz
- kdump Patch　　kexec-tools-1.101-kdump.patch
- Kernel Source Code　　linux-2.6.13.tar.bz2

(2) Apply the kdump Patch and Install the kexec-tool

```
# tar zxvf kexec-tools-1.101.tar.gz
# patch -p0 < kexec-tools-1.101-kdump.patch
# cd <directory in which kexec-tool was installed>
# ./configure
# make
# make install
```

(3) Uncompress Kernel 2.6.13

```
# tar jxvf linux-2.6.13.tar.bz2
```

(4) Creating the First Kernel

```
# make menuconfig
    Processor type and features    --->
        [*] kexec system call (EXPERIMENTAL)
        (0x100000) Physical address where the kernel is loaded
    File systems    --->
      Pseudo filesystems    --->
          [*] sysfs file system support
# make
# make modules_install
# make install
```

Edit /boot/grub/menu.lst to assign a value to the crashkernel parameter.

```
title Red Hat Enterprise Linux AS (2.6.13)
root (hd0,0)
kernel /vmlinuz-2.6.13 ro root=LABEL=/ rhgb quiet crashkernel=64M@16M
initrd /initrd-2.6.13.img
```

In the assignment, "64M" indicates the capacity of the memory area retained for the second kernel, and "16M" is the start address of the memory area.

After setting, reboot the system to start the first kernel.

```
# reboot
```

When the system is booted up, check that the kernel version is correct.

```
# uname –r
2.6.13
```

(5) Creating the Second Kernel

```
# make menuconfig
    Processor type and features    --->
        [ ] Symmetric multi-processing support
```

```
        [*] Local APIC support on uniprocessors
        [*]     IO-APIC support on uniprocessors
        [*] kernel crash dumps (EXPERIMENTAL)
        (0x1000000) Physical address where the kernel is loaded
    File systems    --->
      Pseudo filesystems    --->
          [*]     /proc/vmcore support (EXPERIMENTAL)
          [*] /dev file system support (OBSOLETE)
    Device Drivers    --->
        SCSI device support    --->
            <*> SCSI device support
            <*>     SCSI disk support
        Fusion MPT device support    --->
            <*> Fusion MPT (base + ScsiHost) drivers
# make
# make    vmlinux
```

This avoids the "Warning: unable to open an initial console" message.

This setting supports an MPT SCSI device.

Note: Use the vmlinux image for the second kernel for kexec –p instead of bzImage.

(6) Execute kexec

Use the kexec command to load the second kernel that will start and generate a dump when the first kernel panics.

```
# kexec -p ./vmlinux-2.6.13-2nd --crashdump --args-linux --append="root=/dev/sdb1 init
1 irqpool"
```

vmlinux-2.6.13-2nd in the above listing is the vmlinux image of the second kernel created in (5).

(7) Add /dev/console

When the second kernel starts, it sometimes generates the "Warning: unable to open an initial console" message instead of the regular screen. To avoid this problem, create /dev/console statically.

```
# mkdir test
# mount --bind / test
# cd test/dev
# mknod -m 660 console c 5 1
# cd ../..
# umount test
```

```
# rmdir test
```

■ Checking the Operation

(1) Switching to Second Kernel on Kernel Panic

When the first kernel panics, the second kernel starts to generate a dump.

To test this behavior, artificially induce a kernel panic by entering the following commands.

```
# echo 1 >/proc/sys/kernel/sysrq
# echo c >/proc/sysrq-trigger
```

(2) Writing to the Dump File

On the second kernel, it is possible to access the crashed memory area in either of the following formats:

- elf format /proc/vmcore
- raw format /dev/oldmem

```
# cp   /proc/vmcore   <dump-file>
```

The copied dump file can be analyzed in gdb by using vmlinux generated with the -g option. This dump file has not been supported in the crash utility yet.

linux-2.6.13/Documentation/kdump/ contains a gdb macro "gdbmacros.txt," in which bttnobp, btt, btpid, and trapinfo have been added.

[Example]

```
# gdb ./vmlinux ./vmcore --command=gdbmacros.txt
#0   0xc011543c in crash_get_current_regs (regs=0xf6e81ef4)
     at arch/i386/kernel/crash.c:102
102                  regs->eip = (unsigned long)current_text_addr();
(gdb) bttnobp
pid 1; comm init:
==================
__alloc_pages + 617 in section .text
schedule_timeout + 82 in section .text
process_timeout in section .text
do_select + 880 in section .text
copy_to_user + 98 in section .text
```

```
cp_new_stat64 + 234 in section .text
__pollwait in section .text
sys_select + 744 in section .text
sysenter_past_esp + 84 in section .text
```

3.1.5 mkdump

■ Overview

Mini Kernel Dump (mkdump) was developed by NTT Data and VA Linux. Similar to kdump, it generates a dump by using a kernel dedicated for dump generation (mini kernel) that is separate from the failed kernel. The design of the mini kernel is based on kexec, but it has the following advantages.

- The mini kernel writes the dump image onto the disk.
- The dump image can be converted to a form that is supported by the lcrash analysis tool.
- In kexec, starting the second kernel overwrites the first kernel.

mkexec is loaded in the kernel to be dumped, and when the kernel panics, the mini kernel is started for dump generation.

http://mkdump.sourceforge.net/

■ Installation Environment

Kernel: kernel 2.6.9
Machine Architecture: i386　DELL PowerEdge 2600
Operating System: Red Hat Enterprise Linux AS 4 update1
Memory: 2 GB

■ Supported Dump Analysis Tools

crash, lcrash, and Alicia

■ Installation Procedure

(1) Copying the Files

Download the following file and copy them to any directory:

- Mini Kernel Patch　　minik-2.6.9.-1.0.patch
- Kernel Source Code　　linux-2.6.9.tar.bz2
- mkexec Source Code　　mkexec-2.6.9-1.0.taz

(2) Apply the Mini Kernel Patch

```
# tar jxvf linux-2.6.9.tar.bz2
# mv linux-2.6.9 linux-2.6.9-minik
# cp minik-2.6.9-1.0.patch    linux-2.6.9-minik
# cd linux-2.6.9-minik
# patch -p1 < minik-2.6.9-1.0.patch
```

(3) Creating the Mini Kernel

```
# make menuconfig
    General setup    --->
      [*] Make dump mini kernel
      [ ]     --- Run the mini kernel on PAE mode (default) (i386)
      [ ] Support for paging of anonymous memory (swap)
    Processor type and features    --->
      [ ] Symmetric multi-processing support
    Device Drivers    --->
        Networking support    --->
            [ ] Networking support
        SCSI device support    --->
            <*> SCSI device support
            <*>     SCSI disk support
        Fusion MPT device support    --->
            <*> Fusion MPT (base + ScsiHost) drivers
    File systems    --->
      Remove all items leaving "[ ]".
# make bzImage
```

This setting supports an MPT SCSI device.

Copying the created kernel (with a different name).

```
# cp arch/i386/boot/bzImage /boot/vmlinuz-2.6.9-minik
# cp arch/i386/boot/compressed/vmlinux.bin /boot/vmlinux.bin-2.6.9-minik
```

(4) Creating a Dump Partition

Create the partition on the disk that will store a dump file. The size of the partition

should be 1.5 times the actual memory capacity. In this example, the dump partition is created on /dev/sdb1. After creation, check the major number and minor number of the dump partition.

```
# ls –l /dev/sdb1
brw-rw----    1 root disk 8, 1 Jun 24 12:59 /dev/sdb1
```

Major Number: 8

Minor Number: 1

(5) Edit the /boot/grub/menu.lst

Add the mini kernel entry to the grub to check if the mini kernel will work correctly.

```
title Linux-2.6.9-minik
      kernel (hd0,0)/vmlinuz-2.6.9-minik mem=8M dump_dev=0x800001
```

"8" of dump_dev is the major number and "1" is the minor number.

Confirm that the system is rebooted after a dump generation, starting the mini kernel.

(6) Apply the mkexec Patch

```
# tar zxvf mkexec-2.6.9-1.0.taz
# tar jxvf linux-2.6.9.tar.bz2
# mv linux-2.6.9    linux-2.6.9-mkexec
# cd linux-2.6.9-mkexec
# patch -p1 < ../mkexec/PATCH/mkexec-v1.patch
```

(7) Creating mkexec

```
# make menuconfig
      Processor type and features    --->
          [ ] Symmetric multi-processing support
     Device Drivers    --->
        SCSI device support    --->
             SCSI low-level drivers    --->
                 Remove all Qlogic drivers
# make ; make modules_install ; make install
```

This avoids an error due to the smp description during make.

This avoids an error due to the Qlogic description during make.

After setting, reboot the system to start mkexec.

```
# reboot
```

When the system is booted up, check that the kernel version is correct.

```
# uname –r
2.6.9-mkexec
```

(8) Creating mkexec.ko

Edit mkexec/kernel/Makefile.base according to the specific environment.

```
KVER=        2.6.9-mkexec
KDIR=        <directory in which mkdump was installed>/linux-2.6.9-mkexec
KMODDIR=    /lib/modules/2.6.9-mkexec
```

Preparing make.

```
# cd <directory in which mkdump was installed>/mkexec
# ln -s asm-i386 include/asm
```

Execute make.

```
# make ARCH=i386
```

mkexec/kernel/mkexec.ko is created.

■ Checking Operation

(1) Execute a Mini Kernel Dump

Load mkexec.ko.

```
# insmod <directory in which mkdump was installed>/mkexec/kernel/mkexec.ko
```

(2) Preparing for the Dump

Specify parameters for proc/mkexec.

```
# echo 8 > /proc/mkexec/mem
# echo “0x800001” > /proc/mkexec/dumpdev
# echo “/boot/vmlinux.bin-2.6.9-minik” > /proc/mkexec/path
# echo 1 > /proc/mkexec/stats
```

(3) Generating a Dump

When a kernel panic occurs with mkexec loaded, the mini kernel is started.

Artificially start the mini kernel by entering the following command:

```
# echo 2 > /proc/mkexec/stats
```

Confirm that, after generating the dump, the system returns to the startup screen and the kernel in which mkexec was installed is started.

(4) Convert Dump File Format

Convert the file in the dump partition into the LKCD format.

Create the conversion tool.

```
# cd /usr/src
# tar zxvf mkd_conv.taz
# cd mkd_conv
# make
```

elf2lkcd_i386, mkd2lkcd_i386, mkd2lkcd_x86_64 are generated.

- elf2lkcd_i386 Converts the i386 vmcore format to LKCD.
- mkd2lkcd_i386 Converts the i386 mkdump format to LKCD.
- mkd2lkcd_x86_64 Converts the x86_64 mkdump format to LKCD.

Convert the dump format.

```
# ./mkd2lkcd_i386 /dev/sdb1 <dump-file>
```

The dump data saved in /dev/sdb1 is converted to a file.

3.1.6 Evaluation of Dump Generation Tools

We generated dumps using the above tools (diskdump, netdump, LKCD, kdump, and mkdump) to evaluate their features and ease of operation. We did not investigate the functions of each tool at the source code level.

Table 3.1-1 shows the comparison between dump generation tools as of September 2005.

Table 3.1-1 Comparison between Dump Generation Tools (Part 1) (as of September 2005)

Item		diskdump	netdump	LKCD	kdump	mkdump
Distribution	RHEL3.0	○(update3 or later)	○(For IA64, update5 or later)	×	×	×
	RHEL4.0	○	○	×	×	×
	SLES8	×	○	○	×	×
	SLES9	×	○	○	×	×

Item		diskdump	netdump	LKCD	kdump	mkdump
	MIRACLE LINUX V3.0	○(SP1 or later)	○(SP1 or later)	○	×	×
	Turbolinux10S	×	×	○	×	×
Latest Version		1.0.1-5	0.7.7-3	6.1.0	1.101	2.0
Dump File Format		netdump	netdump	LKCD	ELF	LKCD (Requires manual conversion.)
Supported Analysis Tool	crash	○	○	○	×	○
	lcrash	○	○	○	×	○
	Alicia	○	○	○	×	○
	gdb	-	-	-	○	-
Dump Trigger	oops	○	○	○	○	○
	NMI watchdog	○	○	○	○	○
	SysRq magic key (manual)	○	○	○	○	○
Dumped Information	Execution context	○	○	○	○	○
	Memory	○	○	○	○	○
	Cache memory	×	×	×	×	×
	Swap space	×	×	×	×	×
Rebooting after Dump		Manual	Automatic	Automatic or manual (selectable)	Manual	Automatic

Table 3.1-1 Comparison between Dump Generation Tools (Part 2) (as of September 2005)

Item	diskdump	netdump	LKCD	kdump	mkdump

Timing of Dump File Save	When a crash occurs, a dump image is automatically generated in the dump partition. After rebooting, a shell is automatically started, in which the dump file is converted to the format supported by the analysis tool and saved.	When a crash occurs, a dump image is automatically transferred to the server. After transfer, the dump file in /var/crash is converted to the format supported by the analysis tool and saved.	When a crash occurs, a dump image is automatically generated in the dump partition. After rebooting, a shell is automatically started, in which the dump file is converted to the format supported by the analysis tool and saved.	The dump is saved by a second kernel that is manually started.	When the second kernel starts, it automatically generates a dump image in the dump partition. After rebooting, the dump image is converted with a conversion tool that is manually started to the format that is supported to the analysis tool.
Supported Interface	The devices using the following drivers are supported. aic7xxx aic79xx mptfusion megaraid ata_pix sata_promise	**The NIC using the following drivers are supported.** **e1000** **tg3** **bcm5700**	Basically, all disk drivers in the kernel are supported.	Basically, all disk drivers in the kernel are supported.	Basically, all disk drivers in the kernel are supported.
Features	Red Hat and MIRACLE LINUX include the tool package. The setup is easier than kdump and mkdump, which require kernel rebuild.	Red Hat and MIRACLE LINUX include the tool package. The setup is easier than kdump and mkdump, which require kernel rebuild.	Distributions other than Red Hat include the tool package. The setup is easier than kdump and mkdump, which require kernel rebuild. It is possible to specify whether to reboot after dump and the dump file compression algorithm.	The operation is reliable because a dump image is generated by a different kernel than the failed kernel.	The operation is reliable because a dump image is generated by a different kernel than the failed kernel.

Things to be Aware of	• The disk device or partition must be larger than the physical memory capacity. • The tool can dump with only the disks that use drivers with supported interfaces.	• A separate server for saving the dump is required. - The NIC driver must support netdump. • A teaming NIC cannot be used for sending the dump image. • As the dump image is transferred through the network via the UDP protocol, there is a possibility of data loss.	• The disk device or partition must be larger than the physical memory capacity. It is also possible to dump swap information.	• Since there is no Linux distribution that includes this package as standard, kernel patching and kernel rebuild are always required. • Manual intervention is necessary after the second kernel starts.	• Since there is no Linux distribution that includes this package as standard, kernel patching and kernel rebuild are always required. • It has been found that the tool cannot generate a dump on a system having a SCSI card with an LSI 53C1030 chip. The cause is currently unknown.

■ Comparison between Supported Linux Distributions

There is currently no Linux distribution that supports kdump or mkdump as standard. This means that these packages require kernel patching and kernel rebuild. LKCD is supported by SLES (SUSE Linux Enterprise Server), MIRACLE LINUX and Turbolinux. diskdump and netdump are supported by RHEL and ML3.0 SP1.

■ Comparison between Supported Analysis Tools

The dump image generated by kdump is analyzed in gdb. crash and lcrash are not supported. Dump images taken by other dump generation tools can be analyzed with crash and lcrash. (Alicia internally wraps both crash and lcrash. It can be used for all tools but kdump.)

■ Comparison between Generation Triggers

All tools support dump generation triggered by oops, NMI, or SysRq magic key.

■ Dumped Information

All tools can dump execution contexts and memory contents, etc..

■ Rebooting after Dump

In mkdump and netdump, the kernel is automatically rebooted after dump generation.
In kdump, after the second kernel starts up, it waits for instructions from the console. Therefore the system has to be rebooted after manual dump generation.
diskdump halts after writing the dump image to disk and requires manual rebooting.

LKCD allows selecting automatic or manual rebooting after dump generation.

■ Timing of Dump File Save

netdump saves the dump file when the kernel crashes, in a format that is supported by the dump analysis tool.

diskdump and LKCD write the dump image to disk when the kernel crashes. Then, after rebooting, the image is converted to a format that is supported by the dump analysis tools.

mkdump generates a dump image on disk when the second kernel starts after the first kernel crashes. After rebooting, it is necessary to use a file converter to manually convert the image to a format supported by the dump analysis tool.

kdump does not automatically generate a dump. When the second kernel starts up after kernel crash, it is necessary to save the dump file manually.

■ Comparison between Supported Interfaces

kdump, mkdump, and LKCD basically support the disk drivers included in the kernel.

diskdump can only dump with disks that use drivers having diskdump support code.

netdump can only dump through NICs that use drivers having netdump support code.

3.1.7 Consideration

There are several criteria for selecting dump generation tools. This section considers the reliability of dump generation, ease of program installation, ease of dump taking, and whether booting is required after taking a dump, which are based on the comparison in 3.1.6. In the following section, the item names in Table 3.1-1 in 3.1.6 are indicated in parentheses following the section title.

■ Reliability of Dump Generation (Timing of Dump File Save, and Features)

This item examines whether a memory dump is certainly generated when a failure occurs and the dump contents are reliable from the viewpoint of the kernel used for dump generation.

In kdump and mkdump, when a crash occurs, a dedicated kernel for dump generation (second kernel) is started in the memory space that is separate from that for the failed kernel (first kernel). For these tools, dump image is generated under control of the second kernel. On the other hand, diskdump, netdump, and LKCD generate a dump under control of the failed kernel itself.

Using the second kernel avoids the structural problem of diskdump, netdump, and

LKCD, in which the kernel that may be the cause of a dump is also used for generating the dump. Comparing the methods for generating a Linux kernel dump, kdump and mkdump, which use the second kernel, have big advantages in the ability of the dump generation and the reliability of the dump contents.

■ Ease of Tool installation (Distribution, and Things to be Aware of)

The second kernel method is not supported as standard in the currently available major distributions. Therefore, when installing kdump or mkdump, the user or the vendor who supports the user has to apply the required patches and rebuild the kernel. Moreover, there is the possibility that the distribution vendor will refuse to provide support when the patched kernel had a failure.

Considering these, it can be said that the second kernel method has great advantages as a dump generation function but it also has some difficulties in applying the second kernel method for the users who run the Linux for their real system. In contrast, diskdump, netdump, and LKCD are supported by each distribution as standard. They are also easy to install and set up as compared to kdump and mkdump.

■ Ease of Dump Generation (Timing of Dump File Save, and Things to be Aware of)

Considering a situation in which the kernel crashed while the system is operating in the user environment, ease of dump taking is as important as sureness of dump generation. In other words, it is necessary to pay attention to how simple the dump taking procedure is. mkdump, LKCD, diskdump, and netdump automatically generate a dump without requiring manual invention. In kdump, when the console accepts input after the second kernel is invoked, it is necessary to start dump generation manually. We hope that the usability of kdump will be improved in the future.

■ Rebooting after Dump Operation (Rebooting after Dump)

When thinking of the operation of a dump generation tool, whether the kernel is automatically rebooted after dump generation is a major point in a system administration.

In systems where fast recovery and short down time are the top priority, the kernel needs to be rebooted immediately after dump generation to resume services as soon as possible. However, in cases of hardware failures, rebooting may result in another crash, which could destroy the dump taken at the initial failure.

Whether to reboot or not after dump generation depends on the system purpose, existence of system redundancy, and failure cause (hardware or software). Therefore,

the best way could be that the user can select between manual and automatic rebooting after dump generation as in LKCD. The implementation of this feature in LKCD indicates that the product was developed with a strong awareness of real operation. We sincerely hope that other dump generation tools will also introduce this feature.

Table 3.1-2 Comparison and Consideration of Dump Generation Tools

Item	diskdump	netdump	LKCD	kdump	mkdump
Sureness of Dump Generation	×	×	×	○	○
Ease of Tool Installation	○	○	○	×	×
Ease of Dump Generation	○	△ (Server is needed)	○	×	○
Rebooting after Dump	△ (Manual reboot)	○	○	×	○

■ Future Prospects

Linux will continue to spread in mission critical fields and dump taking at the time of a failure will become even more important. In such an environment, dump generation using a second kernel is a must. It is necessary for each distribution to support a second kernel so as to facilitate the tool installation procedure. If we focus attention on the mission critical, we should also consider an approach other than the current software method. To be more specific, introducing a hardware memory dump mechanism, such as that available in mainframe computers, into enterprise servers is one possibility.