

# AIXCHINA

## AIX 问题检测工具和技巧 学习笔记（三）

[www.aixchina.com](http://www.aixchina.com)

AIX 中国论坛发表的所有文章版权均属相关权利人所有，受《中华人民共和国著作权法》及其它相关法律的保护。

如出于商业目的使用本资料或有牵涉版权的问题请速与论坛管理员联系。管理员电子邮件：[aixchina@21cn.com](mailto:aixchina@21cn.com)

# AIX Problem Determination Tools and Techniques

IBM Certification Study Guide Test 185

第五章 系统 dumps .....	4
1. 配置 DUMP 设备 .....	4
2. 发动一次系统 dump .....	5
(1) 用命令行 .....	5
(2) 用 SMIT .....	5
(3) 用 reset 键 .....	5
(4) 用特别的键盘序列 .....	6
3. 系统 dump 状态检查 .....	6
(1) 状态码 .....	7
4. 增加 dump 设备大小 .....	7
5. 拷贝系统 dump .....	7
6. 读 dump .....	9
7. Core dump .....	9
(1) 检查 core dump .....	10
(2) 找到 core 文件 .....	11
(3) 在 core 中找到导致 core dump 的程序 .....	11

## 第五章 系统 dumps

当系统非正常停机，dump 对系统做了一个快照，dump 数据首先写到主 dump 设备，如果主设备不可用，再写到次设备。Dump 也可以由用户发动，并写到指定设备。

### 1 . 配置 DUMP 设备

AIX4.1 前，默认的 dump 设备是/dev/hd7，AIX4.1 以后，默认的设备是/dev/hd6，同时也是默认的交换区设备。次设备是/dev/sysdumpnull，系统重启时，/dev/hd6 中的内容从/dev/hd6 拷贝到 /var/adm/ras。

当前的 dump 配置可以用 sysdumpdev 列出。

```
# sysdumpdev
primary          /dev/hd6
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression OFF
```

主 dump 设备必须总在 rootvg 中，次 dump 设备可以在 rootvg 以外。

注意：不要用镜像的或有拷贝的 lv 做 dump 设备，否则 dump 将失败，而且没有错误信息。不要用软盘设备作为 dump 设备。

AIX4.2.1 以后支持用 rootvg 中的交换区设备作为次 dump 设备。

Sysdumpdev 可以指定远程 dump 设备，下面的条件必须满足：

本地和远程都支持 TCPIP

本地安装 NFS

远程支持 NFS

远程可访问

可以用 sysdumpdev 永久修改主 dump 设备：

```
# sysdumpdev -P -p /dev/hd3
primary          /dev/hd3
secondary        /dev/sysdumpnull
copy directory   /var/adm/ras
forced copy flag TRUE
always allow dump FALSE
dump compression OFF
```

用 sysdumpdev 永久修改次 dump 设备：

```
# sysdumpdev -P -s /dev/rmt0
primary          /dev/hd3
secondary        /dev/rmt0
copy directory   /var/adm/ras
forced copy flag TRUE
```

always allow dump FALSE

dump compression OFF

用 sysdumpdev 临时修改主 dump 设备，下次重启时恢复原设置：

```
# sysdumpdev -p /dev/rmt0
```

primary /dev/rmt0

secondary /dev/sysdumpnull

copy directory /var/adm/ras

forced copy flag TRUE

always allow dump FALSE

dump compression OFF

## 2. 发动一次系统 dump

用户可以发动一次系统 dump，但是，如果面板显示 888，表示已经有一次 dump 发生了，如果没有将 dump 数据从设备中转移出来，就发动 dump，将覆盖上一次的数据。

如果安装了 software service aids package，可以用命令行方式或 SMIT 调用 sysdumpstart 发动 dump。

如果没有安装 software service aids package，可以用 reset 键或特别的键盘序列发动 dump。

### (1) 用命令行

```
#sysdumpstart -p
```

dump 数据将写入主 dump 设备，可以用-s 参数让数据写入次设备。

Dump 成功，在重启时如果 forced copy flag 被设为 TRUE，主控台上会出现选择，数据是拷贝到 /dev/rmtx 还是 /dev/fd0。/dev/hd6 中 Dump 的大小也会显示。

用 sysdumpdev -K 设置 force copy flag 为 TRUE。

### (2) 用 SMIT

```
#smit dump
```

### (3) 用 reset 键

钥匙要在 service 档，如果没有钥匙，要用

```
#sysdumpdev -K
```

将 Always Allow System Dump 设为 true。

```
# sysdumpdev
```

primary /dev/hd6

secondary /dev/sysdumpnull

copy directory /var/adm/ras

forced copy flag TRUE

always allow dump TRUE

dump compression      OFF

按 reset 键即可，完成后没有钥匙的机器要用

#sysdumpdev -k

将 Always Allow System Dump 设为 false。

## （4）用特别的键盘序列

过程和按 reset 键一样，用 Ctrl-Alt 1 表示写入主 dump 设备，用 Ctrl-Alt 2 表示写入次 dump 设备。

AIX4.3.2 以后可以通过串口远程启动系统，SMIT 菜单中对串口的配置如下：

Add a TTY	
Type or select values in entry fields. Press Enter AFTER making all desired changes.	
[MORE...14]	[Entry Fields]
STTY attributes for RUN time	[hupcl,cread,brkint,icr] +
STTY attributes for LOGIN	[hupcl,cread,echoe,cs8]
LOGGER name	[ ]
STATUS of device at BOOT time	[available] +
REMOTE reboot ENABLE	no +
<b>REMOTE reboot STRING</b>	<b>[#@reb@#]</b>
TRANSMIT buffer count	[16] +#
RECEIVE trigger level	[3] +#
STREAMS modules to be pushed at OPEN time	[ldterm] +
INPUT map file	[none] +
OUTPUT map file	[none] +
CODESET map file	[sbcs] +
[MORE...17]	
F1=Help	F2=Refresh
F5=Reset	F6=Command
F9=Shell	F10=Exit
F3=Cancel	F7=Edit
F4=List	F8=Image
Enter=Do	

REMOTE reboot Enable settings 设置如下：

No：禁止通过这个串口发动远程 reboot。

Reboot：当收到 reboot string 时，系统重启。

Dump：当收到 reboot string 时，系统 dump。

Reboot string 可以由用户自定。

## 3 . 系统 dump 状态检查

如果 dump 是用户发动的，完成后显示 0cx；如果 dump 是因为系统失败，完成后是闪烁的 888。

注意：如果 dump 失败，重启后错误日志中有 DSI\_PROC 或 ISI\_PROC，详细数据显示 EXVAL 为 000 0005，这可能是交换区 IO 错误。如果交换区就是 dump 设备或在同一个硬盘上，可能是硬盘的问题。

## (1) 状态码

- 000 进入核心调试模式，在连接在串口的字符终端上，会有调试提示符>，输入 q dump 等待 888 出现。
- 0c0 dump 成功。
- 0c1 发生一个 IO 错误。
- 0c2 用户发动的 dump 未完成，如果持续一分钟，说明有未知错误发生。
- 0c4 dump 设备的空间不足。
- 0c5 发生一个内部错误，等待一分钟，观察是否有新代码出现。
- 0c7 一个网络 dump 正在执行，在等待服务器端响应，代码在 0c7、0c2、0c9 间切换，如果代码不变化，说明有错误发生。
- 0c8 没有 dump 设备。
- 0c9 系统引起的 dump 没有完成。如果持续一分钟，说明有未知错误发生。
- 0cc AIX4.2.1 以后，dump 到主 dump 设备出错，dump 切换到次 dump 设备，等待一分钟，观察是否有新代码出现，如果持续一分钟，说明有未知错误发生。
- c20 退出核心调试模式，没有发动 dump，读新的代码。

## 4 . 增加 dump 设备大小

只有实存中的数据被 dump，交换区中的数据没有 dump，dump 的最小大小可以这样估算：

```
# sysdumpdev -e
```

```
0453-041 Estimated dump size in bytes: 38797312
```

如果dump设备是/dev/hd6，可以用命令lsps -a看其大小：

```
# lsps -a
```

Page Space	Physical Volume	Volume Group	Size	%Used	Active	Auto	Type
hd6	hdisk0	rootvg	512MB	1	yes	yes	lv

用smit chps加大空间；

如果dump设备是文件，可以用命令smit chfs改文件系统的大小。

## 5 . 拷贝系统 dump

snap 命令可以将 dump 拷贝到外部介质，是否有可用的 dump，可以用命令：

```
# sysdumpdev -L
```

```
0453-039
```

```
Device name: /dev/hd6
```

```
Major device number: 10
```

```
Minor device number: 2
```

```
Size: 42568192 bytes
```

```
Date/Time: Wed Jul 12 14:53:55 CDT 2000
```

```
Dump status: 0
```

```
dump completed successfully
```

```
Dump copy filename: /usr/dumpdir/vmcore.0
```

这个例子中dump是成功的可以用snap拷贝到磁带等介质中：

```
# snap -gfkD -o /dev/rmt0
Setting output device to /dev/rmt0... done.
Checking space requirement for general
information..... done.
Checking space requirement for kernel information..... done.
Checking space requirement for dump information..... done.
Checking space requirement for filesys information.....done.
Checking for enough free space in filesystem... done.
*****Checking and initializing directory structure
Directory /tmp/ibmsupt/filesys already exists... skipping
Directory /tmp/ibmsupt/dump already exists... skipping
Directory /tmp/ibmsupt/kernel already exists... skipping
Directory /tmp/ibmsupt/general already exists... skipping
Directory /tmp/ibmsupt/testcase already exists... skipping
Directory /tmp/ibmsupt/other already exists... skipping
*****Finished setting up directory /tmp/ibmsupt
Gathering general system
information..... done.
Gathering kernel system information..... done.
Gathering dump system information.... done.
Gathering filesys system information..... done.
Copying information to /dev/rmt0... Please wait... done.
*****
*****
***** Please Write-Protect the output device now...
*****
*****
*****
*****
***** Please label your tape(s) as follows:
***** snap blocksize=512
***** problem: xxxxx Wed Jul 12 15:41:42 CDT 2000
***** 'your name or company's name here'
*****
*****
*****
```

以后 dump 文件可以用 tar -x 从磁带上恢复：

```
# tar -tvf /dev/rmt0
drwx----- 0 0          0 Jul 12 13:48:44 2000 ./dump/
-rw----- 0 0          2555 Jul 12 15:40:21 2000 ./dump/dump.snap
-rw----- 0 0      1770955 Jul 12 13:48:29 2000 ./dump/unix.Z
-rwx----- 0 0    41761792 Jul 12 11:03:29 2000 ./dump/dump_file
...
drwx----- 0 0          0 Jul 12 11:23:06 2000 ./kernel/
-rw----- 0 0        75122 Jul 12 15:40:21 2000 ./kernel/kernel.snap
```



```
drwx----- 0 0          0 Jul 12 11:22:58 2000 ./testcase/
drwx----- 0 0          0 Jul 12 11:22:58 2000 ./other/
dump.snap、unix.Z、dump_file 包含了 dump 数据。
```

## 6 . 读 dump

检查 dump 是否可读，用 crash 命令（或在 AIX 5L 下用 KDB），crash 需要一个核心文件（unix）匹配 dump 文件，如果不指定，将默认用/unix，必须和原机的操作系统版本一致。

```
# crash dump unix
> stat
  sysname: AIX
  nodename: sp5i
  release: 3
  version: 4
  machine: 000126774C00
  time of crash: Tue May 4 04:56:10 CDT 1999
  age of system: 4 min.
  xmalloc debug: disabled
  abend code: 300
  csa: 0x2ff3b400
  exception struct:
    dar: 0x00000003
    dsisr: 0x00000000:
    srv: 0x04000000
    dar2: 0x3c160040
    dsirr: 0x06001000: "(unknown reason code)"
```

如果出现下面的提示，说明 dump 有问题：

```
# crash dump unix
Cannot locate offset 0x02052b8 in segment 0x0000000.
endcomm 0x00000000/0x011c5e70
WARNING: dumpfile does not appear to match namelist
Cannot locate offset 0x00ccf10 in segment 0x0000000.
0452-179: Cannot read v structure from address 0x ccf10.
Symbol proc has null value.
Symbol thread has null value.
Cannot locate offset 0x00ccf10 in segment 0x0000000.
0452-179: Cannot read v structure from address 0x ccf10.
Cannot locate offset 0x00034c4 in segment 0x0000000.
0452-1002: Cannot read extension segment value from address 0x 34c4
```

## 7 . Core dump

当系统发生 core dump 时，一个 core 文件会在当前目录创建。导致 core dump 的原因有内存地

址错误，指令错误，总线错误，用户发出的信号等。Core 文件包含中断进程的内存印象。

## (1) 检查 core dump

core dump 发生后，错误日志将记录：

```
# errpt
IDENTIFIER TIMESTAMP T C RESOURCE_NAME      DESCRIPTION
...
C60BB505    0705101400  P  S SYSPROC SOFTWARE PROGRAM ABNORMALLY TERMINATED
...
# errpt -a -j C60BB505
-----
LABEL: CORE_DUMP
IDENTIFIER: C60BB505
Date/Time: Wed Jul 5 10:14:59
Sequence Number: 8
Machine Id: 000BC6DD4C00
Node Id: client1
Class: S
Type: PERM
Resource Name: SYSPROC
Description
SOFTWARE PROGRAM ABNORMALLY TERMINATED
Probable Causes
SOFTWARE PROGRAM
User Causes
USER GENERATED SIGNAL
    Recommended Actions
    CORRECT THEN RETRY
Failure Causes
SOFTWARE PROGRAM
    Recommended Actions
    RERUN THE APPLICATION PROGRAM
    IF PROBLEM PERSISTS THEN DO THE FOLLOWING
    CONTACT APPROPRIATE SERVICE REPRESENTATIVE
Detail Data
SIGNAL NUMBER
    4
USER'S PROCESS ID:
    15394
FILE SYSTEM SERIAL NUMBER
    5
INODE NUMBER
    2
```

PROGRAM NAME

netscape\_aix4

ADDITIONAL INFORMATION

Unable to generate symptom string.

Too many stack elements.

上例中说明 netscape\_aix4 是导致 core dump 的程序。

## (2) 找到 core 文件

```
# find / -name core -ls
```

```
737 10188 -rw-r--r-- 1 root system 10430807 Jul 5 10:14 /core
```

上例说明 core 文件在根目录下。

## (3) 在 core 中找到导致 core dump 的程序

有两种方法：

用 string 命令：

```
# strings core | grep _=
```

```
_=/usr/netscape/communicator/us/netscape_aix4
```

用 lquerypv 命令：

```
# lquerypv -h core 6b0 64
```

```
000006B0 7FFFFFFF FFFFFFFF 7FFFFFFF FFFFFFFF |.....|
000006C0 00000000 000007D0 7FFFFFFF FFFFFFFF |.....|
000006D0 00120000 137084E0 00000000 00000016 |....p....|
000006E0 6E657473 63617065 5F616978 34000000 |netscape_ai x4...|
000006F0 00000000 00000000 00000000 00000000 |.....|
00000700 00000000 00000000 00000000 0000085E |.....^|
00000710 00000000 00000F5A 00000000 00000776 |.....Z.....v|
```