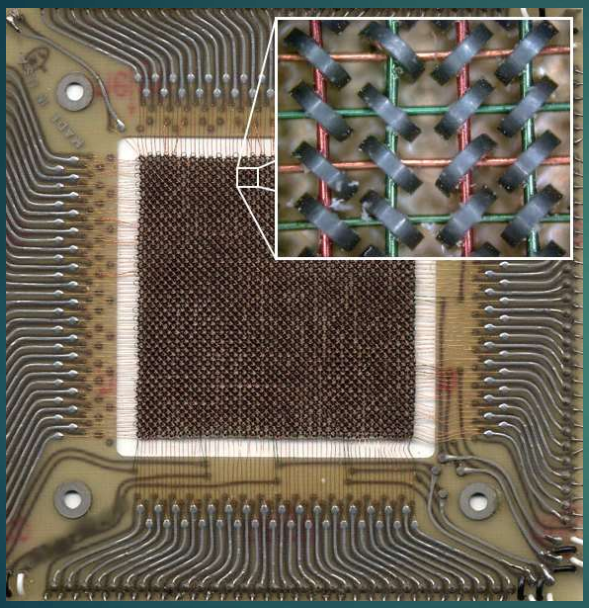


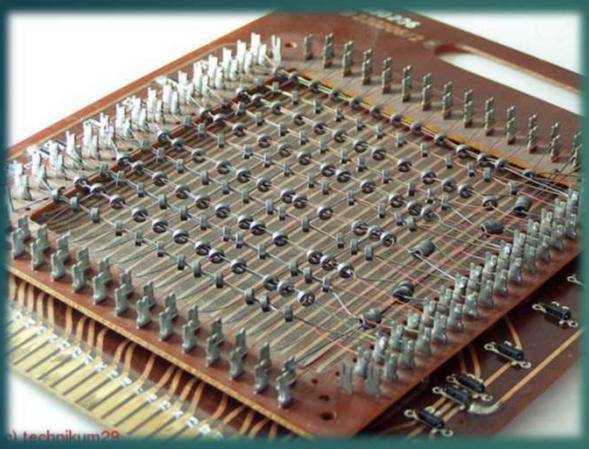
LINUX内核开发与调试 -- Core

张银奎

2017/2/16

1

何谓Core?



Magnetic-core memory was the predominant form of random-access computer memory for 20 years between about 1955 and 1975. Such memory is often just called core memory, or, informally, core

2

Core dump

In computing, a core dump (in Unix parlance), memory dump, or system dump[1] consists of the recorded state of the working memory of a computer program at a specific time, generally when the program has crashed or otherwise terminated abnormally.

3

回到现实

```
ge@gewubox:~/work/hdtrap$ ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
```

- 但却找不到文件，why?

4

ulimit

ulimit [-HSTabcdefilmnpqrstuvx [limit]]

资源是有限的，合理限制，合理使用

- Provides control over the resources available to the shell and to processes started by it, on systems that allow such control.
- The -H and -S options specify that the hard or soft limit is set for the given resource.
- A hard limit cannot be increased by a non-root user once it is set; a soft limit may be increased up to the value of the hard limit. If neither -H nor -S is specified, both the soft and hard limits are set. The value of limit can be a number in the unit specified for the resource or one of the special values hard, soft, or unlimited, which stand for the current hard limit, the current soft limit, and no limit, respectively.
- If limit is omitted, the current value of the soft limit of the resource is printed, unless the -H option is given.
- When more than one resource is specified, the limit name and unit are printed before the value.

5

参数和示例

ulimit [-HSTabcdefilmnpqrstuvx [limit]]

选项 [options]	含义	例子
-H	设置硬资源限制，一旦设置不能增加。	ulimit – Hs 64；限制硬资源，线程栈大小为 64K。
-S	设置软资源限制，设置后可以增加，但是不能超过硬资源设置。	ulimit – Sn 32；限制软资源，32 个文件描述符。
-a	显示当前所有的 limit 信息。	ulimit – a；显示当前所有的 limit 信息。
-c	最大的 core 文件的大小， 以 blocks 为单位。	ulimit – c unlimited； 对生成的 core 文件的大小不进行限制。
-d	进程最大的数据段的大小，以 Kbytes 为单位。	ulimit -d unlimited；对进程的数据段大小不进行限制。
-f	进程可以创建文件的最大值，以 blocks 为单位。	ulimit – f 2048；限制进程可以创建的最大文件大小为 2048 blocks。
-l	最大可加锁内存大小，以 Kbytes 为单位。	ulimit – l 32；限制最大可加锁内存大小为 32 Kbytes。
-m	最大内存大小，以 Kbytes 为单位。	ulimit – m unlimited；对最大内存不进行限制。
-n	可以打开最大文件描述符的数量。	ulimit – n 128；限制最大可以使用 128 个文件描述符。
-p	管道缓冲区的大小，以 Kbytes 为单位。	ulimit – p 512；限制管道缓冲区的大小为 512 Kbytes。
-s	线程栈大小，以 Kbytes 为单位。	ulimit – s 512；限制线程栈的大小为 512 Kbytes。
-t	最大的 CPU 占用时间，以秒为单位。	ulimit – t unlimited；对最大的 CPU 占用时间不进行限制。
-u	用户最大可用的进程数。	ulimit – u 64；限制用户最多可以使用 64 个进程。
-v	进程最大可用的虚拟内存，以 Kbytes 为单位。	ulimit – v 200000；限制最大可用的虚拟内存为 200000 Kbytes

6

$ ulimit -c unlimited

```
ge@gewubox:~/work/hdtrap$ ulimit -c unlimited
ge@gewubox:~/work/hdtrap$ ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
ge@gewubox:~/work/hdtrap$ ll
total 328
drwxrwxr-x 2 ge ge   4096 Feb 17 11:35 ./
drwxrwxr-x 8 ge ge   4096 Jan  7 14:16 ../
-rw------- 1 ge ge 253952 Feb 17 11:35 core
-rwxrwxr-x 1 ge ge     34 Jan  7 14:16 hd.sh*
```

7

core_pattern

- 决定core文件名和位置，也影响是否产生
- $ sysctl kernel.core_pattern
- /etc/sysctl.conf
- $ sysctl -p

8

默认传给apport

- ge@gewubox:~/work/hdtrap$ cat /proc/sys/kernel/core_pattern
- |/usr/share/apport/apport %p %s %c

```
ge@gewubox:~$ cat /proc/sys/kernel/core_pattern
|/usr/share/apport/apport %p %s %c
ge@gewubox:~$ cat /proc/sys/kernel/core_uses_pid
0
```

```
root@gewubox:/home/ge/work/hdtrap# echo "core_%e_%t_%s">/proc/sys/kernel/core_pa
ttern
root@gewubox:/home/ge/work/hdtrap# ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
root@gewubox:/home/ge/work/hdtrap# ls
core                          hd.sh     hdtrap.map  HeadTrap.c  md5.o
core.3165                     hdtrap    hdtrap.o    Makefile
core_hdtrap_1487421402_11.3173  hdtrap.c  hdtrapr     md5.c
root@gewubox:/home/ge/work/hdtrap#
```

9

定制文件名


```
root@gewubox:/home/ge/work/hdtrap# echo "core_%e_%t_%s">/proc/sys/kernel/core_pa
ttern
root@gewubox:/home/ge/work/hdtrap# ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
root@gewubox:/home/ge/work/hdtrap# ls
core                          hd.sh     hdtrap.map  HeadTrap.c  md5.o
core.3165                     hdtrap    hdtrap.o    Makefile
core_hdtrap_1487421402_11.3173  hdtrap.c  hdtrapr     md5.c
root@gewubox:/home/ge/work/hdtrap#
```


10

可变字段

- %% 单个%字符
- %p 所dump进程的进程ID
- %u 所dump进程的实际用户ID
- %g 所dump进程的实际组ID
- %s 导致本次core dump的信号
- %t core dump的时间 (由1970年1月1日计起的秒数)
- %h 主机名
- %e 程序文件名

11

定制路径

```
root@gewubox:/home/ge/work/hdtrap# echo "/var/core_%e_%t_%s">/proc/sys/kernel/co
re_pattern
root@gewubox:/home/ge/work/hdtrap# ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
root@gewubox:/home/ge/work/hdtrap# ll /var/core*
-rw------- 1 root root 253952 Feb 18 20:38 /var/core_hdtrap_1487421527_11.3175
root@gewubox:/home/ge/work/hdtrap#
```

12

持久化

- 放入配置文件
- **/etc/sysctl.conf**
- kernel.core_pattern = %e.core.%p
- 并保存退出，执行**sysctl -p**使其生效

```
root@gewubox:/home/ge/work/hdtrap# gedit /etc/sysctl.conf &
[1] 3177
root@gewubox:/home/ge/work/hdtrap# sysctl -p
kernel.core_pattern = %e.core.%p
[1]+  Done                    gedit /etc/sysctl.conf
root@gewubox:/home/ge/work/hdtrap# ./hdtrap
Stuntman for xsw by Raymond (rev1.0)
running...
Segmentation fault (core dumped)
root@gewubox:/home/ge/work/hdtrap# ll
total 812
drwxrwxr-x 2 ge   ge     4096 Feb 18 20:44 ./
drwxrwxr-x 8 ge   ge     4096 Jan  7 14:16 ../
-rw------- 1 ge   ge   253952 Feb 18 18:45 core
-rw------- 1 root root 253952 Feb 18 20:29 core.3165
-rw------- 1 root root 253952 Feb 18 20:36 core_hdtrap_1487421402_11.3173
-rwxrwxr-x 1 ge   ge       34 Jan  7 14:16 hd.sh*
-rwxrwxr-x 1 ge   ge    10794 Jan  7 14:16 hdtrap*
-rwxrwxr-x 1 ge   ge     2685 Jan  7 14:16 hdtrap.c*
-rw------- 1 root root 253952 Feb 18 20:44 hdtrap.core.3219
```

13

持久化ulimit设置

- #vi /etc/profile 然后，在profile中添加：
- ulimit -c 1073741824
- 或者ulimit -c unlimited

```
if [ -d /etc/profile.d ]; then
  for i in /etc/profile.d/*.sh; do
    if [ -r $i ]; then
      . $i
    fi
  done
  unset i
fi
ulimit -c unlimited
ulimit unlimited
```

14

$ objdump –h core

```
core:     file format elf32-i386

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 note0         0000057c  00000000  00000000  00000334  2**0
                  CONTENTS, READONLY
  1 .reg/3067     00000044  00000000  00000000  00000390  2**2
                  CONTENTS
  2 .reg          00000044  00000000  00000000  00000390  2**2
                  CONTENTS
  3 .auxv         000000a0  00000000  00000000  00000510  2**2
                  CONTENTS
  4 load1         00001000  08048000  00000000  00001000  2**12
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  5 load2         00001000  08049000  00000000  00002000  2**12
                  CONTENTS, ALLOC, LOAD, READONLY
  6 load3         00001000  0804a000  00000000  00003000  2**12
                  CONTENTS, ALLOC, LOAD
  7 load4         00001000  b7544000  00000000  00004000  2**12
                  CONTENTS, ALLOC, LOAD
  8 load5a        00001000  b7545000  00000000  00005000  2**12
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  9 load5b        00000000  b7546000  00001000  00006000  2**12
                  ALLOC, READONLY, CODE
 10 load6         00002000  b76e8000  00000000  00006000  2**12
```

15

$ readelf -h core

```
ge@gewubox:~/work/hdtrap$ readelf -a core
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              CORE (Core file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x0
  Start of program headers:          52 (bytes into file)
  Start of section headers:          0 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         24
  Size of section headers:           0 (bytes)
  Number of section headers:         0
  Section header string table index: 0

There are no sections in this file.

There are no sections to group in this file.
```

16

Program Headers

```
ge@gewubox:~/work/hdtrap$ readelf -l core

Elf file type is CORE (Core file)
Entry point 0x0
There are 24 program headers, starting at offset 52

Program Headers:
  Type           Offset   VirtAddr   PhysAddr   FileSiz MemSiz  Flg Align
  NOTE           0x000334 0x00000000 0x00000000 0x0057c 0x00000     0
  LOAD           0x001000 0x08048000 0x00000000 0x01000 0x01000 R E 0x1000
  LOAD           0x002000 0x08049000 0x00000000 0x01000 0x01000 R   0x1000
  LOAD           0x003000 0x0804a000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x004000 0xb7544000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x005000 0xb7545000 0x00000000 0x01000 0x1a3000 R E 0x1000
  LOAD           0x006000 0xb76e8000 0x00000000 0x02000 0x02000 R   0x1000
  LOAD           0x008000 0xb76ea000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x009000 0xb76eb000 0x00000000 0x03000 0x03000 RW  0x1000
  LOAD           0x00c000 0xb76ee000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x00d000 0xb76ef000 0x00000000 0x01000 0x17000 R E 0x1000
  LOAD           0x00e000 0xb7706000 0x00000000 0x01000 0x01000 R   0x1000
  LOAD           0x00f000 0xb7707000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x010000 0xb7708000 0x00000000 0x02000 0x02000 RW  0x1000
  LOAD           0x012000 0xb770a000 0x00000000 0x01000 0x2a000 R E 0x1000
  LOAD           0x013000 0xb7734000 0x00000000 0x01000 0x01000 R   0x1000
  LOAD           0x014000 0xb7735000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x015000 0xb7745000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x016000 0xb7746000 0x00000000 0x02000 0x02000 RW  0x1000
  LOAD           0x018000 0xb7748000 0x00000000 0x01000 0x01000 R E 0x1000
  LOAD           0x019000 0xb7749000 0x00000000 0x01000 0x20000 R E 0x1000
  LOAD           0x01a000 0xb7769000 0x00000000 0x01000 0x01000 R   0x1000
  LOAD           0x01b000 0xb776a000 0x00000000 0x01000 0x01000 RW  0x1000
  LOAD           0x01c000 0xbf982000 0x00000000 0x22000 0x22000 RW  0x1000
```


Notes

readelf --notes core

```
Notes at offset 0x00000334 with length 0x0000057c:
  Owner                 Data size       Description
  CORE                  0x00000090      NT_PRSTATUS (prstatus structure)
  CORE                  0x0000007c      NT_PRPSINFO (prpsinfo structure)
  CORE                  0x00000080      Unknown note type: (0x53494749)
  CORE                  0x000000a0      NT_AUXV (auxiliary vector)
  CORE                  0x000002a8      Unknown note type: (0x46494c45)
  LINUX                 0x00000030      Unknown note type: (0x00000200)
```

http://stackoverflow.com/questions/17972945/core-dump-note-section

安装elfutils

```
ge@gewubox:~/work/hdtrap$ sudo apt-get install elfutils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libasm1 libdw1
The following NEW packages will be installed:
  elfutils libasm1 libdw1
0 upgraded, 3 newly installed, 0 to remove and 366 not upgraded.
Need to get 559 kB of archives.
After this operation, 1,414 kB of additional disk space will be used.
Do you want to continue [Y/n]? Y
Get:1 http://cn.archive.ubuntu.com/ubuntu/ precise-updates/main libasm1 i386 0.152-1ubuntu3.1 [18.4 kB]
Get:2 http://cn.archive.ubuntu.com/ubuntu/ precise-updates/main libdw1 i386 0.152-1ubuntu3.1 [214 kB]
Get:3 http://cn.archive.ubuntu.com/ubuntu/ precise-updates/universe elfutils i386 0.152-1ubuntu3.1 [326 kB]
Fetched 559 kB in 0s (610 kB/s)
Selecting previously unselected package libasm1.
(Reading database ... 146523 files and directories currently installed.)
Unpacking libasm1 (from .../libasm1_0.152-1ubuntu3.1_i386.deb) ...
Selecting previously unselected package libdw1.
Unpacking libdw1 (from .../libdw1_0.152-1ubuntu3.1_i386.deb) ...
Selecting previously unselected package elfutils.
Unpacking elfutils (from .../elfutils_0.152-1ubuntu3.1_i386.deb) ...
Setting up libasm1 (0.152-1ubuntu3.1) ...
Setting up libdw1 (0.152-1ubuntu3.1) ...
Setting up elfutils (0.152-1ubuntu3.1) ...
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
```

19

$eu-readelf –n core

- 根据下面链接，可能产生更详细的信息

```
65  CORE            144  PRSTATUS
    info.si_signo: 11, info.si_code: 0, info.si_errno: 0, cursig: 11
    sigpend: <>
    sighold: <>
    pid: 31614, ppid: 31544, pgrp: 31614, sid: 31544
```

https://www.cs.swarthmore.edu/~kwebb/cs31/s15/bucs/elf.html#coredump_gdb

```
ge@gewubox:~/work/hdtrap$ eu-readelf -n ./core

Note segment of 1404 bytes at offset 0x334:
  Owner          Data size  Type
  CORE                 144  PRSTATUS
  CORE                 124  PRPSINFO
  CORE                 128  <unknown>: 1397311305
  CORE                 160  AUXV
    SYSINFO: 0xb7748414
    SYSINFO_EHDR: 0xb7748000
    HWCAP: 0x78bfbff
    PAGESZ: 4096
    CLKTCK: 100
    PHDR: 0x8048034
    PHENT: 32
    PHNUM: 8
    BASE: 0xb7749000
    FLAGS: 0
    ENTRY: 0x80485a0
    UID: 1000
    EUID: 1000
    GID: 1000
    EGID: 1000
    SECURE: 0
    RANDOM: 0xbf9a27bb
    EXECFN: 0xbf9a3ff3
    PLATFORM: 0xbf9a27cb
    NULL
  CORE                 680  <unknown>: 1179208773
  LINUX                 48  386_TLS
```

20

有请GDB

```
ge@gewubox:~/work/hdtrap$ gdb ./hdtrap core
GNU gdb (Ubuntu/Linaro 7.4-2012.04-0ubuntu2.1) 7.4-2012.04
Copyright (C) 2012 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-linux-gnu".
For bug reporting instructions, please see:
<http://bugs.launchpad.net/gdb-linaro/>...
Reading symbols from /home/ge/work/hdtrap/hdtrap...done.
[New LWP 3067]

warning: Can't read pathname for load map: Input/output error.
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/i386-linux-gnu/libthread_db.so.1".
Core was generated by `./hdtrap'.
Program terminated with signal 11, Segmentation fault.
#0  0x08048930 in calc_md5 (data=0x8048a00 "testing data-xxxxxxx", nLen=20, md5=0x0) at md5.c:7
7           md5[0] = A;
(gdb)
```

- gdb program core debug coredump core produced by program

21

另一种方法

```
ge@gewubox:~/work/hdtrap$ gdb -c core
GNU gdb (Ubuntu/Linaro 7.4-2012.04-0ubuntu2.1) 7.4-2012.04
Copyright (C) 2012 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-linux-gnu".
For bug reporting instructions, please see:
<http://bugs.launchpad.net/gdb-linaro/>.
[New LWP 3067]
Core was generated by `./hdtrap'.
Program terminated with signal 11, Segmentation fault.
#0  0x08048930 in ?? ()
```

(gdb) file ./hdtrap
Reading symbols from /home/ge/work/hdtrap/hdtrap...done.
(gdb) bt
#0 0x08048930 in calc_md5 (data=0x8048a00 "testing data-xxxxxxx", nLen=20, md5=0x0) at md5.c:7
#1 0x08048698 in get_file_id (filename=0x8048abe "filea", fileid=0xbf9a25c0) at hdtrap.c:27
#2 0x0804891d in main (argc=1, argv=0xbf9a2674) at hdtrap.c:115

22

```
3.4 /proc/<pid>/coredump_filter - Core dump filtering settings
--------------------------------------------------------------
When a process is dumped, all anonymous memory is written to a core file as
long as the size of the core file isn't limited. But sometimes we don't want
to dump some memory segments, for example, huge shared memory. Conversely,
sometimes we want to save file-backed memory segments into a core file, not
only the individual files.

/proc/<pid>/coredump_filter allows you to customize which memory segments
will be dumped when the <pid> process is dumped. coredump_filter is a bitmask
of memory types. If a bit of the bitmask is set, memory segments of the
corresponding memory type are dumped, otherwise they are not dumped.

The following 7 memory types are supported:
  - (bit 0) anonymous private memory
  - (bit 1) anonymous shared memory
  - (bit 2) file-backed private memory
  - (bit 3) file-backed shared memory
  - (bit 4) ELF header pages in file-backed private memory areas (it is
            effective only if the bit 2 is cleared)
  - (bit 5) hugetlb private memory
  - (bit 6) hugetlb shared memory

  Note that MMIO pages such as frame buffer are never dumped and vDSO pages
  are always dumped regardless of the bitmask status.

  Note bit 0-4 doesn't effect any hugetlb memory. hugetlb memory are only
  effected by bit 5-6.
```

23

Eglibc2.15/signal/sigandset.c

```
/* Combine sets LEFT and RIGHT by logical AND and place result in DEST.  */
int
sigandset (dest, left, right)
     sigset_t *dest;
     const sigset_t *left;
     const sigset_t *right;
{
  if (dest == NULL || left == NULL || right == NULL)
    {
      __set_errno (EINVAL);
      return -1;
    }

  return __sigandset (dest, left, right);
}
```

24

近乎完美的栈回溯

```
Core was generated by `./geheap64'.
Program terminated with signal SIGABRT, Aborted.
#0  0x00007f50ca489428 in __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:54
54   ../sysdeps/unix/sysv/linux/raise.c: No such file or directory.
(gdb) bt
#0  0x00007f50ca489428 in __GI_raise (sig=sig@entry=6) at ../sysdeps/unix/sysv/linux/raise.c:54
#1  0x00007f50ca48b02a in __GI_abort () at abort.c:89
#2  0x00007f50ca4cb7ea in __libc_message (do_abort=do_abort@entry=2,
fmt=fmt@entry=0x7f50ca5e4ed8 "*** Error in `%s': %s: 0x%s ***\n")
at ../sysdeps/posix/libc_fatal.c:175
#3  0x00007f50ca4d437a in malloc_printerr (ar_ptr=<optimized out>, ptr=<optimized out>,
str=0x7f50ca5e4fa0 "double free or corruption (fasttop)", action=3) at malloc.c:5006
#4  _int_free (av=<optimized out>, p=<optimized out>, have_lock=0) at malloc.c:3867
#5  0x00007f50ca4d853c in __GI___libc_free (mem=<optimized out>) at malloc.c:2968
#6  0x0000000004007d3 in main (argc=1, argv=0x7ffd4fbe3238) at geheap.c:42
```

25

x64 CALL Convention

Microsoft x64 calling convention[14]	Windows (Microsoft Visual C++, GCC, Intel C++ Compiler, Delphi), UEFI	RCX/XMM0, RDX/XMM1, R8/XMM2, R9/XMM3	RTL (C) [21]	Caller	Stack aligned on 16 bytes. 32 bytes shadow space on stack. The specified 8 registers can only be used for parameters 1 through 4.For C++ classes, the hidden "this" parameter is the first parameter, and is passed in RCX [22].
vectorcall	Windows (Microsoft Visual C++)	RCX/XMM0, RDX/XMM1, R8/XMM2, R9/XMM3 + XMM0-XMM5/YMM0-YMM5	RTL (C)	Caller	[23]
System V AMD64 ABI[19]	Solaris, Linux, BSD, OS X (GCC, Intel C++ Compiler)	RDI, RSI, RDX, RCX, R8, R9, XMM0–7	RTL (C)	Caller	Stack aligned on 16 bytes boundary. 128 bytes red zone below stack. The kernel interface uses RDI, RSI, RDX, R10, R8 and R9.

https://en.wikipedia.org/wiki/X86_calling_conventions

26

SIGABRT的来源

27

两种终止

NAME
abort - cause abnormal process termination

SYNOPSIS
#include <stdlib.h>

void abort(void);

NAME
exit - cause normal process termination

SYNOPSIS
#include <stdlib.h>

void exit(int status);

28

自杀式调用

```
Program received signal SIGABRT, Aborted.
0xf7fd8dc9 in __kernel_vsyscall ()
(gdb) bt
#0  0xf7fd8dc9 in __kernel_vsyscall ()
#1  0xf7e32d09 in raise () from /lib32/libc.so.6
#2  0xf7e34207 in abort () from /lib32/libc.so.6
#3  0xf7e6d7ac in ?? () from /lib32/libc.so.6
#4  0xf7e73787 in ?? () from /lib32/libc.so.6
#5  0xf7e73fb1 in ?? () from /lib32/libc.so.6
#6  0x080485ad in main (argc=1, argv=0xffffd014) at geheap.c:41
```

29

Libc的输出信息

```
*** Error in `/home/gedu/labs/geheap/geheap': free(): invalid next size
(fast): 0x0804b410 ***
======= Backtrace: =========
/lib32/libc.so.6(+0x667a7)[0xf7e6d7a7]
/lib32/libc.so.6(+0x6c787)[0xf7e73787]
/lib32/libc.so.6(+0x6cfb1)[0xf7e73fb1]
/home/gedu/labs/geheap/geheap[0x80485ad]
/lib32/libc.so.6(__libc_start_main+0xf7)[0xf7e1f637]
/home/gedu/labs/geheap/geheap[0x80483e1]
```

Libc检测到严重错误状况时，向标准错误（stderr）设备输出信息

30

归去

```
(gdb) disassemble
Dump of assembler code for function __kernel_vsyscall:
   0xf7fd8dc0 <+0>:   push   %ecx
   0xf7fd8dc1 <+1>:   push   %edx
   0xf7fd8dc2 <+2>:   push   %ebp
   0xf7fd8dc3 <+3>:   mov    %esp,%ebp
   0xf7fd8dc5 <+5>:   sysenter
   0xf7fd8dc7 <+7>:   int    $0x80
=> 0xf7fd8dc9 <+9>:   pop    %ebp
   0xf7fd8dca <+10>:  pop    %edx
   0xf7fd8dcb <+11>:  pop    %ecx
   0xf7fd8dcc <+12>:  ret
```

```
(gdb) info registers
eax            0x0    0
ecx            0xa192585
edx            0x6    6
ebx            0xa192585
esp            0xffffcbd80xffffcbd8
ebp            0xffffce98     0xffffce98
esi            0xf7fb7000 -134516736
edi            0xffffcc94 -13164
eip            0xf7fd8dc9     0xf7fd8dc9
<__kernel_vsyscall+9>
eflags         0x286      [ PF SF IF ]
cs             0x23   35
ss             0x2b   43
ds             0x2b   43
es             0x2b   43
fs             0x00
gs             0x63   99
```


原因

```
int main(int argc, char* argv[])
{
    char * p;
    int i = 0;

    display_mallinfo();

    p = malloc(10);
    //p = 12;

    display_mallinfo();

    for(i = 0; i< 20; i++)
    {
        *(p+i) = i;
    }

    free(p);
    free(p);

    return 0;
}
```

/home/ge/labs/geheap

大坑之solib张冠李戴

gedu@DESKTOP-4NBEECU: /mnt/c/temp/core

```
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from geheap64...done.
[New LWP 2974]

warning: .dynamic section for "/lib/x86_64-linux-gnu/libgcc_s.so.1" is not at the expected address (wrong library or v
ersion mismatch?)
Core was generated by `./geheap64'.
Program terminated with signal SIGABRT, Aborted.
#0  0x00007f50ca489428 in get_sysdep_segment_value (name=0x0) at loadmsgcat.c:657
657     loadmsgcat.c: No such file or directory.
(gdb) bt
#0  0x00007f50ca489428 in get_sysdep_segment_value (name=0x0) at loadmsgcat.c:657
#1  _nl_load_domain (domain_file=0x57, domainbinding=0x7f50ca48d786 <__GI_sigaddset+6>) at loadmsgcat.c:974
#2  0x00007f50ca4d437a in _int_new_arena (size=140725941317760) at arena.c:771
#3  arena_get2 (size=140725941317760, avoid_arena=0x7ffd4fbe3080, a_tsd=<optimized out>) at arena.c:890
#4  0x0000000000000000 in ?? ()
(gdb)
```

33

绕过方法

- Gdb
- set solib-absolute-prefix /mnt/c/invaliddir [阻止在本机找到错误模块]
- Core core_geheap64_xx
- File ./geheap64
- Set solib-search-path ./solib

gedu@DESKTOP-4NBEECU: /mnt/c/temp/core

```
#0  0x00007f50ca489428 in ?? ()
#1  0x00007f50ca48b02a in ?? ()
#2  0x0000000000000020 in ?? ()
#3  0x0000000000000000 in ?? ()
(gdb) set solib-search-path /mnt/c/temp/core/solib/
Reading symbols from /mnt/c/temp/core/solib/libc.so.6...(no debugging symbols found)...done.
Loaded symbols for /mnt/c/temp/core/solib/libc.so.6
Reading symbols from /mnt/c/temp/core/solib/ld-linux-x86-64.so.2...(no debugging symbols found)...done.
Loaded symbols for /mnt/c/temp/core/solib/ld-linux-x86-64.so.2
Reading symbols from /mnt/c/temp/core/solib/libgcc_s.so.1...(no debugging symbols found)...done.
Loaded symbols for /mnt/c/temp/core/solib/libgcc_s.so.1
(gdb) bt
#0  0x00007f50ca489428 in raise () from /mnt/c/temp/core/solib/libc.so.6
#1  0x00007f50ca48b02a in abort () from /mnt/c/temp/core/solib/libc.so.6
#2  0x00007f50ca4cb7ea in ?? () from /mnt/c/temp/core/solib/libc.so.6
#3  0x00007f50ca4d437a in ?? () from /mnt/c/temp/core/solib/libc.so.6
#4  0x00007f50ca4d853c in free () from /mnt/c/temp/core/solib/libc.so.6
#5  0x00000000004007d3 in main (argc=1, argv=0x7ffd4fbe3238) at geheap.c:42
(gdb)
```

34

切问而近思

欢迎关注格友公众号