Diagnosing Native Crashes

tcrash-utility guide

SIPL-AFW-02A

ajinath.kumbhar@smartron.com

5-27-2017

1. Setup tcrash-utility workspace

First copy the tcrash-utility tool in to local desktop with below steps.

Using git clone

```
$ git clone –b dev <LDAP username>@10.11.10.46:/home/akumbhar/tcrash-utility.git
```

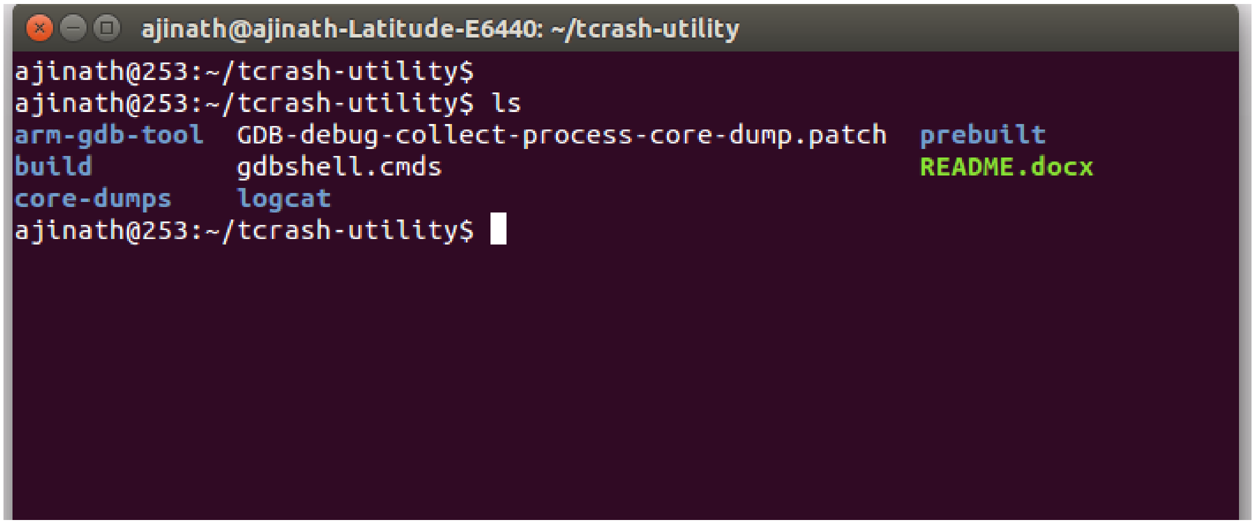
Download it from server

https://drive.google.com/a/smartron.com/file/d/0B31OIf7pWdJKWUZiTVI4WXBPVHc/view?usp=sharing

2. Debug native crash with core dump file

Go to tcrash-utility

```
$ cd tcrash-utility
```



```
ajinath@253:~/tcrash-utility$
ajinath@253:~/tcrash-utility$ ls
arm-gdb-tool  GDB-debug-collect-process-core-dump.patch  prebuilt
build         gdbshell.cmds                              README.docx
core-dumps    logcat
ajinath@253:~/tcrash-utility$ 
```

```
$ source build/envsetup.sh
```

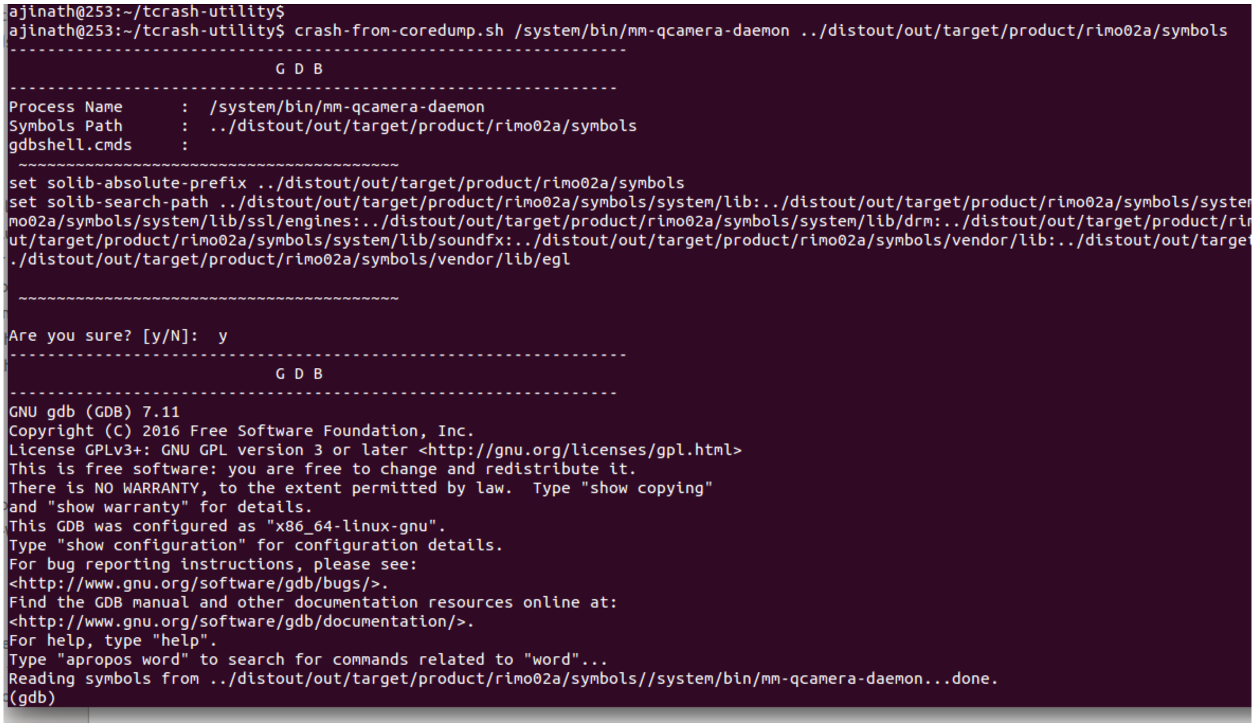
We need three things

a. **Process name which one is crashed** (For process name check 'adb logcat' with DEBUG tag)
b. **Symbols** (Check build directory e.g. out/target/product/rimo02a/symbols)
c. **Core dump of crashed process.** (ref. Document section 4 of **SIPL-AFW-01A** to collect core dump [http://10.11.10.66:81/smartdev66/Smartron/Documents/GDB/Debug-process-crash-with-gdb.pdf]

Now run below command with arguments as process name and symbol path

```
$ crash-from-coredump.sh /system/bin/mm-qcamera-daemon
./distout/out/target/product/rimo02a/symbols
```

$ Are you sure? [y/N]: y



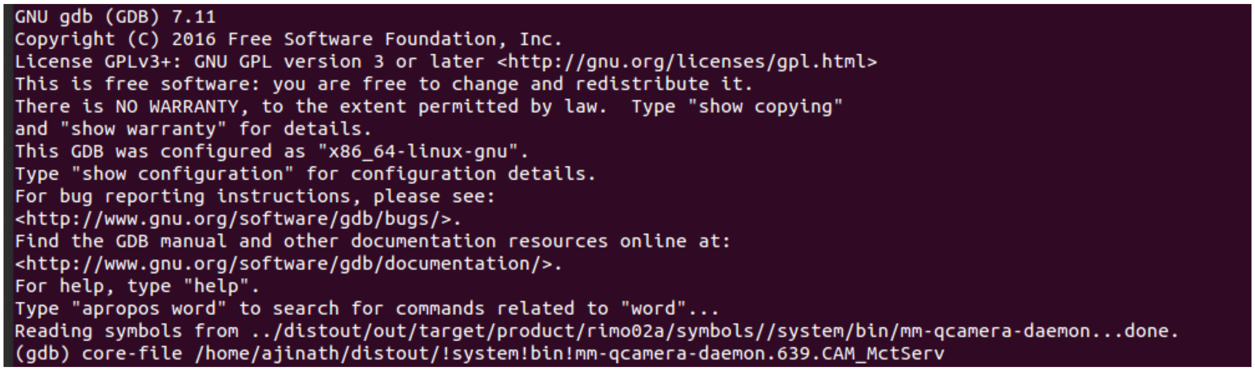
```
ajinath@253:~/tcrash-utility$
ajinath@253:~/tcrash-utility$ crash-from-coredump.sh /system/bin/mm-qcamera-daemon ../distout/out/target/product/rimo02a/symbols
----------------------------------------------------------------
                            G D B
----------------------------------------------------------------
Process Name      :  /system/bin/mm-qcamera-daemon
Symbols Path      :  ../distout/out/target/product/rimo02a/symbols
gdbshell.cmds     :
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
set solib-absolute-prefix ../distout/out/target/product/rimo02a/symbols
set solib-search-path ../distout/out/target/product/rimo02a/symbols/system/lib:../distout/out/target/product/rimo02a/symbols/systen
mo02a/symbols/system/lib/ssl/engines:../distout/out/target/product/rimo02a/symbols/system/lib/drm:../distout/out/target/product/rin
ut/target/product/rimo02a/symbols/system/lib/soundfx:../distout/out/target/product/rimo02a/symbols/vendor/lib:../distout/out/target
./distout/out/target/product/rimo02a/symbols/vendor/lib/egl

 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Are you sure? [y/N]:  y
----------------------------------------------------------------
                            G D B
----------------------------------------------------------------
GNU gdb (GDB) 7.11
Copyright (C) 2016 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 "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ../distout/out/target/product/rimo02a/symbols//system/bin/mm-qcamera-daemon...done.
(gdb)
```

Now gdb terminal started. Run below gdb command to load core dump file

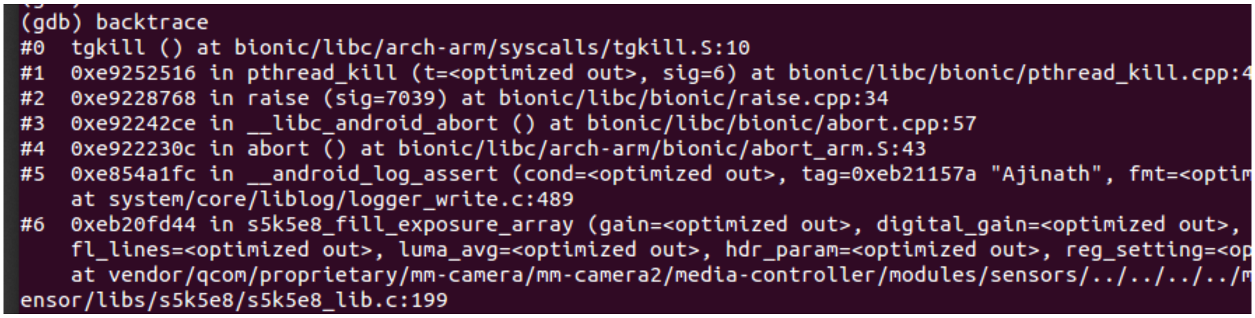
```
(gdb) core-file /home/ajinath/distout/!system!bin!mm-qcamera-daemon.639.CAM_MctServ
```



```
GNU gdb (GDB) 7.11
Copyright (C) 2016 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 "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ../distout/out/target/product/rimo02a/symbols//system/bin/mm-qcamera-daemon...done.
(gdb) core-file /home/ajinath/distout/!system!bin!mm-qcamera-daemon.639.CAM_MctServ
```

gdb will load all required symbols. Now we can use gdb commands to debug crash
e.g

```
(gdb) backtrace
```



```
(gdb) backtrace
#0  tgkill () at bionic/libc/arch-arm/syscalls/tgkill.S:10
#1  0xe9252516 in pthread_kill (t=<optimized out>, sig=6) at bionic/libc/bionic/pthread_kill.cpp:4
#2  0xe9228768 in raise (sig=7039) at bionic/libc/bionic/raise.cpp:34
#3  0xe92242ce in __libc_android_abort () at bionic/libc/bionic/abort.cpp:57
#4  0xe922230c in abort () at bionic/libc/arch-arm/bionic/abort_arm.S:43
#5  0xe854a1fc in __android_log_assert (cond=<optimized out>, tag=0xeb21157a "Ajinath", fmt=<optim
    at system/core/liblog/logger_write.c:489
#6  0xeb20fd44 in s5k5e8_fill_exposure_array (gain=<optimized out>, digital_gain=<optimized out>,
    fl_lines=<optimized out>, luma_avg=<optimized out>, hdr_param=<optimized out>, reg_setting=<op
    at vendor/qcom/proprietary/mm-camera/mm-camera2/media-controller/modules/sensors/../../../../m
ensor/libs/s5k5e8/s5k5e8_lib.c:199
```

Ref . Document section 6 of SIPL-AFW-01A for more command

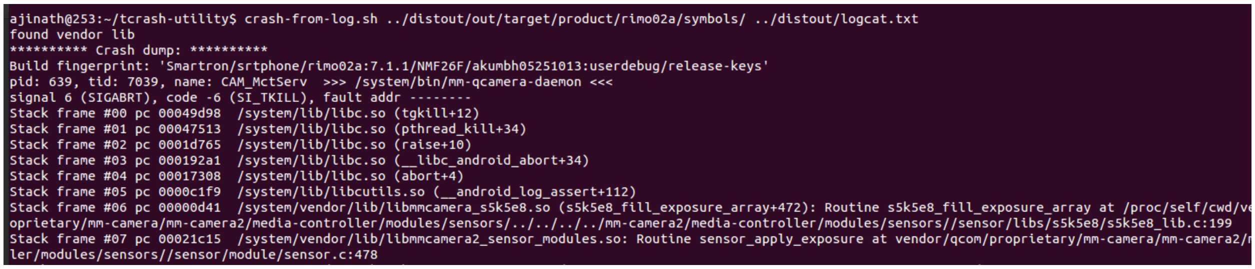
3. Debug native crash with logcat file

We need two things

a. **Symbols** (Check build directory e.g. out/target/product/rimo02a/symbols)
b. **Logcat of process crash** (adb logcat > logcat.txt).

Now run below command with arguments as process name and symbol path

```
$ crash-from-log.sh ../distout/out/target/product/rimo02a/symbols/ ../distout/logcat.txt
```



```
ajinath@253:~/tcrash-utility$ crash-from-log.sh ../distout/out/target/product/rimo02a/symbols/ ../distout/logcat.txt
found vendor lib
********** Crash dump: **********
Build fingerprint: 'Smartron/srtphone/rimo02a:7.1.1/NMF26F/akumbh05251013:userdebug/release-keys'
pid: 639, tid: 7039, name: CAM_MctServ  >>> /system/bin/mm-qcamera-daemon <<<
signal 6 (SIGABRT), code -6 (SI_TKILL), fault addr --------
Stack frame #00 pc 00049d98  /system/lib/libc.so (tgkill+12)
Stack frame #01 pc 00047513  /system/lib/libc.so (pthread_kill+34)
Stack frame #02 pc 0001d765  /system/lib/libc.so (raise+10)
Stack frame #03 pc 000192a1  /system/lib/libc.so (__libc_android_abort+34)
Stack frame #04 pc 00017308  /system/lib/libc.so (abort+4)
Stack frame #05 pc 0000c1f9  /system/lib/libcutils.so (__android_log_assert+112)
Stack frame #06 pc 00000d41  /system/vendor/lib/libmmcamera_s5k5e8.so (s5k5e8_fill_exposure_array+472): Routine s5k5e8_fill_exposure_array at /proc/self/cwd/ve
oprietary/mm-camera/mm-camera2/media-controller/modules/sensors/../../../../mm-camera2/media-controller/modules/sensors//sensor/libs/s5k5e8/s5k5e8_lib.c:199
Stack frame #07 pc 00021c15  /system/vendor/lib/libmmcamera2_sensor_modules.so: Routine sensor_apply_exposure at vendor/qcom/proprietary/mm-camera/mm-camera2/m
ler/modules/sensors//sensor/module/sensor.c:478
```

In output we will get exact routine name and line number where crash triggered.