|  |
| --- |
| ajinathkumbhar@gmail.com  5-27-2017 |

|  |
| --- |
| Diagnosing Native Crashes |
| tcrash-utility guide |
|  |

# Setup tcrash-utility workspace

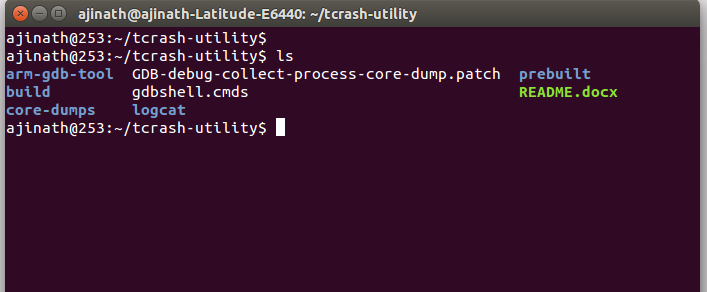
First copy the tcrash-utility tool in to local desktop with below steps.

Using git clone

# Debug native crash with core dump file

Go to tcrash-utility

$ cd tcrash-utility



$ source build/envsetup.sh

We need three things

1. **Process name which one is crashed** (For process name check 'adb logcat' with DEBUG tag)
2. **Symbols** ( Check build directory e.g. out/target/product/xxxx/symbols)
3. **Core dump of crashed process.** (ref. Document section 4 of **SIPL-AFW-01A** to collect core dump [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/xxxx/symbols

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

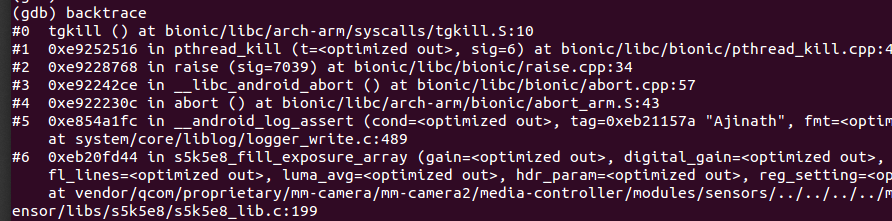
Now gdb terminal started. Run below gdb command to load core dump file

(gdb) core-file /home/ajinath/distout/!system!bin!mm-qcamera-daemon.639.CAM\_MctServ

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

e.g

(gdb) backtrace



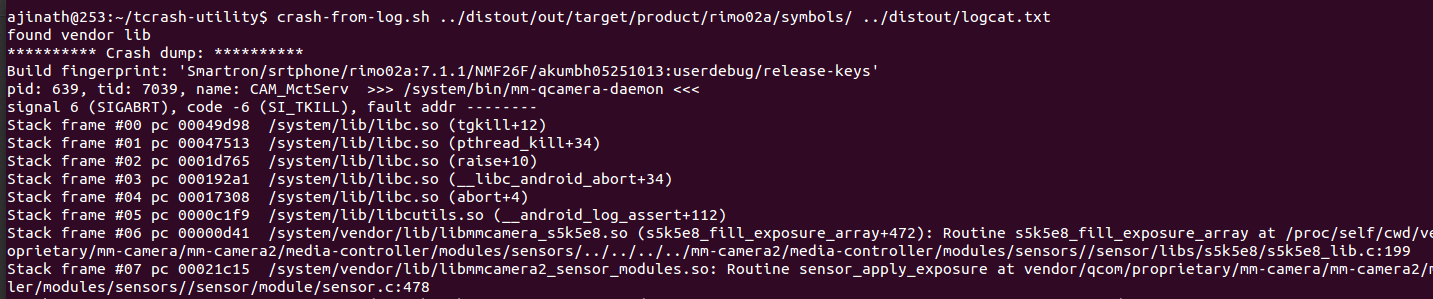
# Debug native crash with logcat file

We need two things

1. **Symbols** ( Check build directory e.g. out/target/product/xxxx/symbols)
2. **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/xxxx/symbols/ ../distout/logcat.txt



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