Debugging Embedded Linux Systems: Kernel Logging System Overview

Debugging Embedded Linux Training Series [Part 2]

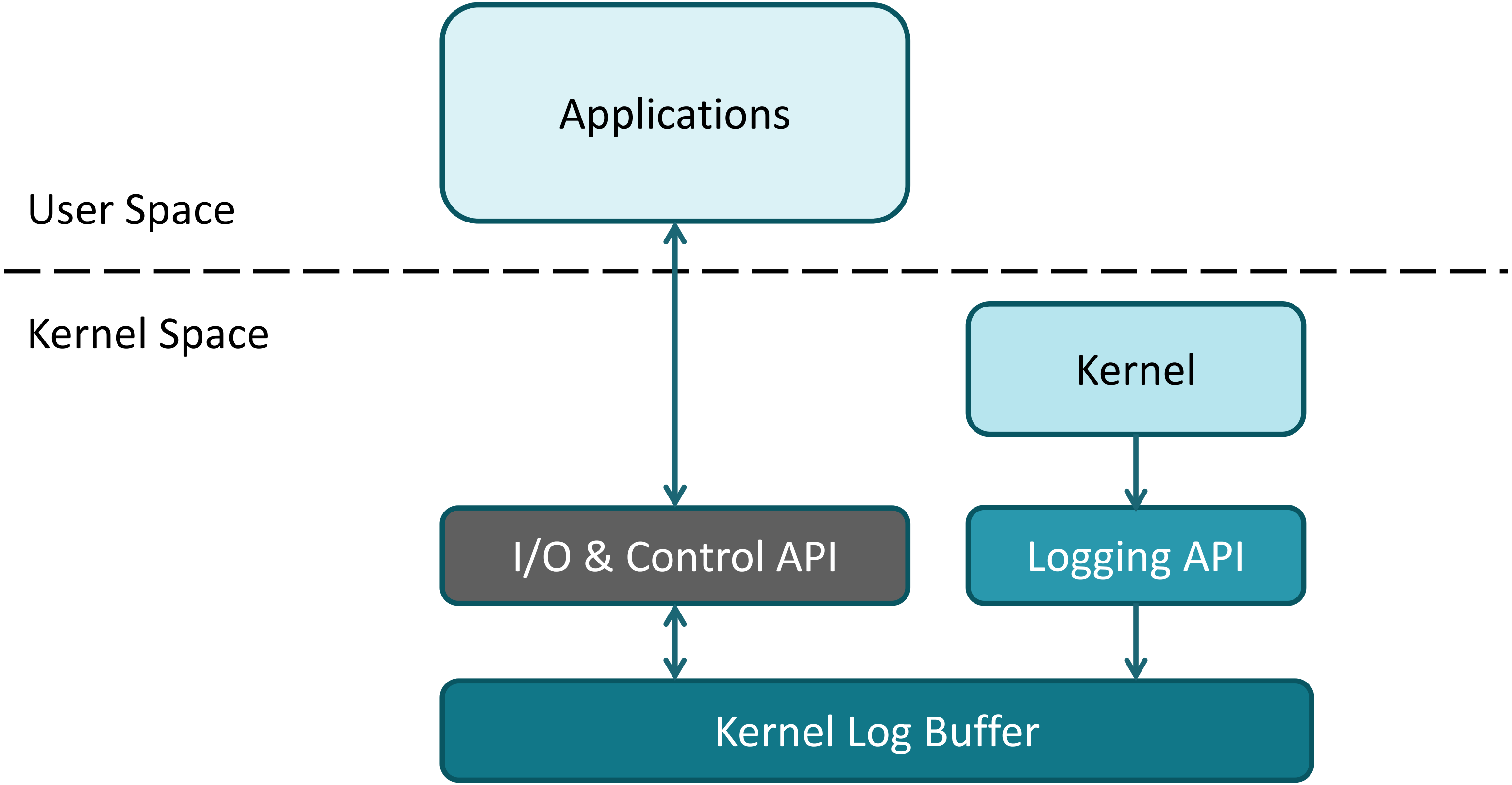
Debugging Embedded Linux Training Series

- Part 1: Linux/Kernel Overview
- **Part 2: Kernel Logging System Overview**
- Part 3: printk and Variations
- Part 4: Dynamic Debug
- Part 5: Locate Device Driver Source Code
- Part 6: Understand Kernel Oops Logs

Agenda

- Kernel logging architecture
- Kernel log example
- Retrieve kernel logs
- Kernel log buffer size
- Adding log messages from application

Kernel logging system architecture



Kernel log example

```
[    0.002777]  memory used by lock dependency info: 5167 kB
[    0.002796]  per task-struct memory footprint: 1536 bytes
[    0.002851] Calibrating delay loop... 795.44 BogoMIPS (lpj=3977216)
[    0.157906] pid_max: default: 32768 minimum: 301
[    0.158592] Security Framework initialized
[    0.158793] Mount-cache hash table entries: 2048 (order: 1, 8192 bytes)
[    0.158825] Mountpoint-cache hash table entries: 2048 (order: 1, 8192 bytes)
[    0.165069] CPU: Testing write buffer coherency: ok
[    0.165368] ftrace: allocating 22031 entries in 65 pages
[    0.248302] CPU0: thread -1, cpu 0, socket -1, mpidr 0
[    0.250093] Setting up static identity map for 0x80100000 - 0x80100070
[    0.255812] smp: Bringing up secondary CPUs ...
[    0.255856] smp: Brought up 1 node, 1 CPU
[    0.255881] SMP: Total of 1 processors activated (795.44 BogoMIPS).
[    0.255902] CPU: All CPU(s) started in SVC mode.
[    0.262088] devtmpfs: initialized
```

Retrieve kernel logs

- **dmesg** command
  - prints/controls the log buffer
- Common **dmesg** usage:
  - **dmesg** *# print the log buffer*
  - **dmesg -C** *# clear the log buffer*
  - **dmesg -c** *# print then clear the log buffer*

Kernel log buffer size

- Default size is 64KB
- Adjust the size
  - Method #1: Kernel Config Option - CONFIG_LOG_BUF_SHIFT=*n*
    - menuconfig: “General Setup”
  - Method #2: uboot bootargs: **log_buf_len=*n***
  - Buffer Size = 2^n
    - n=16: 64KB
    - n=17: 128KB, …

Adding log messages from user space

- Interface:

```
/dev/kmsg
```

- Usage:

```
echo “some comments” > /dev/kmsg
```

- Example:

```
echo “### TESTNOTE: unplugged thumb drive” > /dev/kmsg
echo “### TESTNOTE: waited for a couple seconds” > /dev/kmsg
echo “### TESTNOTE: re-plugged thumb drive” > /dev/kmsg
```

Summary

- Kernel modules use the “Logging API” to generate logs.
- Kernel uses an internal buffer to store logs.
- `dmesg` command can be used to retrieve the logs.
- The log buffer size can be adjusted.

For more information

- Processor SDK Training Series: http://training.ti.com/processor-sdk-training-series
- Debugging Embedded Linux Training Series: http://training.ti.com/debug-embedded-linux-training-series
- Processor SDK Linux Getting Started Guide: http://processors.wiki.ti.com/index.php/Processor_SDK_Linux_Getting_Started_Guide
- Download Processor SDK Linux for Embedded Processors: http://www.ti.com/processorsdk
- For questions about this training, refer to the E2E Embedded Linux Community Forum: http://e2e.ti.com/support/embedded/linux/f/354

Texas Instruments

This material is provided strictly “as-is,” for informational purposes only, and without any warranty.
Use of this material is subject to TI’s **Terms of Use**, viewable at TI.com