Publication date : 08/26/2024

Course : Collecting and analyzing information on kernel malfunctions

Practical course - 2d - 14h - Ref. CZL
Price : 1480 € E.T.

★★★★★ 5 / 5

The Linux kernel generates various messages to identify a bug related to a malfunction. To extract and exploit these messages, you need to know their sources and locations.

Teaching objectives

At the end of the training, the participant will be able to:

- Know the sources of information relating to the operation of the Linux kernel
- Collect exhaustive information on core malfunctions
- Analyze the information gathered

Intended audience

Linux/Unix developers.

Prerequisites

Good knowledge of Linux/Unix and C programming.

Practical details

Teaching methods

This highly interactive course is supported by numerous progressive exercises and case studies.

Course schedule

PARTICIPANTS

Linux/Unix developers.

PREREQUISITES

Good knowledge of Linux/Unix and C programming.

TRAINER QUALIFICATIONS

The experts leading the training are specialists in the covered subjects. They have been approved by our instructional teams for both their professional knowledge and their teaching ability, for each course they teach. They have at least five to ten years of experience in their field and hold (or have held) decision-making positions in companies.

ASSESSMENT TERMS

The trainer evaluates each participant's academic progress throughout the training using multiple choice, scenarios, hands-on work and more.
Participants also complete a placement test before and after the course to measure the skills they've developed.

TEACHING AIDS AND TECHNICAL RESOURCES

• The main teaching aids and instructional methods used in the training are audiovisual aids, documentation and course material, hands-on application exercises and corrected exercises for practical training courses, case studies and coverage of real cases for training seminars.
• At the end of each course or seminar, ORSYS provides participants with a course evaluation questionnaire that is analysed by our instructional teams.
• A check-in sheet for each half-day of attendance is provided at the end of the training, along with a course completion certificate if the trainee attended the entire session.

1 File systems and debugging

- procfs virtual file system.
- sysfs virtual file system.
- Collect debug information with debugfs.
- Store information persistently with pstore.

Hands-on work
Recovering a core dump. Use gdb.

2 Kernel errors and dialog with the kernel

- Head corruption detection with heap / alloc.
- cktrace.
- Warn, Kernel tainted, list of flags.
- Oops, panic, bug.

3 Configuring your kernel to improve debugging

- Debug info.
- Kdump / kexec.

Hands-on work
Configuration of spin lock, mutex, use of printk locks.

4 Kernel debugging tools

- System.map.
- Set up a serial console.
- Specifics of using a serial console under Xen.
- kgbd (serial port).
- Crash / kdump.
- Racing / ftrace.
- Useful kernel parameters: panic=oops, vga=, earlyprintk=, ignore_loglevel, initcall_debug, log_buf_len.

Hands-on work
Using the tools. Setting up a netconsole. Use qemu for debugging.

5 Analyze the information gathered

- Identify memory addresses with addr2line.
- Gdb, the Swiss army knife of debugging.
- Kernel analysis tool: crash.
- Analysis tool: printk.
- Define a message format with pr_*.
- Extract device and driver with dev_*printk versus dev_*?

Hands-on work
Analysis of bug reports.

TERMS AND DEADLINES
Registration must be completed 24 hours before the start of the training.

ACCESSIBILITY FOR PEOPLE WITH DISABILITIES
Do you need special accessibility accommodations? Contact Mrs. Fosse, Disability Manager, at psh-accueil@orsys.fr to review your request and its feasibility.