Monitoring and Debugging in the Linux Environment

Course 9147 - 24 Hours

Overview

Description: This course is intended for programmers who wish to monitor their C/C++ apps on a Linux system, tune their Linux system to their needs, modify the boot process and more

What this course does not cover:

- writing scripts
- interacting with the shell
- login/logout
- using debugger (local or remote)

Who Should Attend

Description: This course is intended for programmers who wish to monitor their C/C++ apps on a Linux system, tune their Linux system to their needs, modify the boot process and more

Course Contents

Monitoring memory

- top, ps and more
- How the OS allocates RAM
- Being not lazy in RAM allocation
- Swap (adding, removing, monitoring)
- To swap or not to swap

Monitoring applications

- ps and it's many options
- how does /proc work
- the /proc folder per app
- What are zombies? How do you identify them? How do you handle them?
- Controlling affinity of apps
- Limiting apps resources
- strace (standalone and attaching)
- top and it's many options
- What is a software watchdog?
- Writing a simple Linux watchdog
- Using inittab for simple watchdogging
- How does the Linux scheduler work? nice(1), chrt(1)

Monitoring disk

- The mount paradigm
- using du and df correctly
- Difference between disk utilization and logical size of files
- Seeing all open files of applications
- Controlling disk scheduling priority and class (ioprio).
- iotop(1) and iostat(1)
- sar(1) and other monitoring tools
- The different file systems Linux supports
- Implications of file system selection on application performance.
- Controlling file system features.

Monitoring network

- netstat and it's many options
- Monitoring and controlling the OS network stack (/proc).
- ifconfig and it's many options
- Monitoring the network interface (ethtool).
- ip(1) and it's many options
- Load balancing and failing over ethernet interfaces
- Bridging network interfaces.
- vlans and tunnels.
- lo and what it's for?
- Name resolution and it's problems.
- tap devices.

Monitoring the kernel

- Seeing kernel threads in ps
- Examining interrupts
- Controlling affinity of interrupt handles
- Interrupt load balancing
- Controlling affinity of kernel threads
- Modules: loading, unloading, monitoring

The boot process

- The BIOS
- The bootloader
- Configuring grub
- Using other bootloaders (lilo, syslinux, uboot)
- Advantages and disadvantages of certain boot loaders
- The init process
- SysV init
- Upstart
- inittab and how it works