LINUX Internals

Introductory Course

Course Title

- Introduction to LINUX Internals

Objectives

- To understand the concept of a time sharing / multi tasking OS
- To understand the kernel architecture
- Introduction to kernel sub systems – Process, File, Memory & I/O Management

Introductory Course…

Training Road Map

- Multiprogramming, multi tasking & multi user OS
- Over view of kernel architecture
- Process Management
  - Process states, Process state transitions, context switching, process table
  - Scheduling – Round robin, priority based
  - Inter Process Communication
- Memory management
  - Virtual memory
  - Paging

Introductory Course…

Training Road Map…

- File management
  - Concept of file descriptors
  - Files Types
  - Directory Structure, inode
  - File Systems
- I/O management
  - Device Controllers
  - Device drivers
- Overview of system calls

Introductory Course…

- Duration
  - 8 hours (6 hours lecture + 2 hours demo)
- Pre-requisites
  - Familiarity with LINUX – directory structure & commands
- Trainer requirement
  - Good understanding of the LINUX System
  - Advanced LINUX commands
  - Knowledge of processor architecture
  - Excellent knowledge on Operating Systems concepts
  - Excellent knowledge on system calls

Introductory Course…

- System requirements
  - Hardware
    - LINUX server & at least one terminal for 2 participants or LINUX desktop
  - Software
    - Any LINUX distribution with gcc

Intermediate Course

Course Title

- LINUX Internals

Objectives

- To understand the concept of a time sharing / multi tasking OS
- To understand the kernel architecture
- Introduction to kernel sub systems – Process, File, Memory & I/O Management
- To familiarize with system calls for kernel sub systems
- To understand signal management
- To familiarize with user level multi-threading

Intermediate Course

Training Road Map

- Multiprogramming, multi tasking & multi user OS
- Over view of kernel architecture
- Process Management
  - Process states, Process state transitions, context switching, process table
  - Scheduling – Round robin, priority based
  - Inter Process Communication
- Memory management
  - Virtual memory
  - Paging
  - Demand paging

Intermediate Course

Training Road Map…

- File management
  - Concept of file descriptors
  - Files Types
  - Directory Structure, inode
  - File Systems
- I/O management
  - Device Controllers
  - Device drivers
- System calls for kernel subsystems

Intermediate Course

Training Road Map…

    - Kernel level & user level threads
    - Reentrancy
- Inter process Communication
    - Pipes
    - Synchronization mechanisms
        - File / record locking
        - Mutex, conditional variables
- Debugging

Intermediate Course

- Duration
  - 3 days (Each day 4 hours lecture + 2 hours demo + 2 hours hands on)
- Pre-requisites
  - Familiarity with LINUX – directory structure & commands
  - Proficiency in C programming
- Trainer requirement
  - Good understanding of the LINUX System
  - Advanced LINUX commands
  - Knowledge of processor architecture
  - Excellent knowledge on Operating Systems concepts
  - Excellent knowledge on C programming
  - Excellent knowledge on system calls

Intermediate Course

- System requirements
    - Hardware
        - LINUX server & at least one terminal for 2 participants or LINUX desktop
    - Software
        - Any LINUX distribution with gcc

Advanced Course

Course Title

- LINUX Internals & Kernel programming

Objectives

- To understand the concept of a time sharing / multi tasking OS
- To understand the kernel architecture
- Introduction to kernel sub systems – Process, File, Memory & I/O Management
- To familiarize with system calls for kernel sub systems
- To understand signal management
- To understand kernel-level & user level multi-threading
- To familiarize with Inter Process Communication mechanisms
- To familiarize with kernel modules

Advanced Course

Training Road Map

- Multiprogramming, multi tasking & multi user OS
- Over view of kernel architecture
- Process Management
  - Process states, Process state transitions, context switching, process table
  - Scheduling – Round robin, priority based
  - Inter Process Communication
- Memory management
  - Virtual memory
  - Paging
  - Demand paging

Advanced Course

Training Road Map…

- File management
    - Concept of file descriptors
    - Files Types
    - Directory Structure, inode
    - File Systems
- I/O management
    - Device Controllers
    - Device drivers
- System calls for kernel subsystems

Advanced Course

Training Road Map…

  - Kernel level & user level threads
  - Thread signals
  - Reentrancy & name space pollution

- Inter process Communication
  - Pipes
  - FIFOS
  - Message Queues
  - Shared memory
  - Synchronization mechanisms
    - File / record locking
    - Mutex, conditional variables
    - Semaphores
  - Socket programming

Advanced Course

Training Road Map…

- Kernel programming
  - Application programming Vs Kernel modules
  - Writing simple kernel modules
  - Loading & unloading modules
  - Kernel Synchronization mechanisms
- Debugging
  - User level – gdb, strace, proc
  - Kernel level – kdb, kgdb, core-dump analysis
  - LINUX trace tools kit

Advanced Course

- Duration
  - 10 days (Each day 4 hours lecture + 2 hours demo + 2 hours hands on)
- Pre-requisites
  - Familiarity with LINUX – directory structure & commands, make utility
  - Proficiency in C programming
  - Network basics
- Trainer requirement
  - Excellent understanding of the LINUX System
  - Advanced LINUX commands
  - Knowledge of processor architecture
  - Excellent knowledge on Operating Systems concepts
  - Excellent knowledge on C programming
  - Excellent knowledge on system calls
  - Knowledge on Network basics
  - Expertise in kernel hacking

Advanced Course

- System requirements
  - Hardware
    - LINUX desktop
  - Software
    - Any LINUX distribution with gcc
    - Kernel source code