



## ADVANCED LINUX FOR ARCHITECTS

**Course Duration:** 5 Days

**Course Overview:** This Five day course introduces architects and technology specialists to the Linux development philosophy, the development tools, and the kernel programming interface. The program is intended to provide an overview of Linux development for individuals who will be delivering Linux driven or based solutions.

**Intended Audience/Prerequisites:** This course is for people who already know C programming and the basics of Unix/Linux as a user, including file manipulation and basic Unix/Linux tools.

### COURSE OUTLINE:

#### Introduction

- Student and Instructor introductions
- Assumptions about your background
- Class logistics, schedule
- Optional companion texts
- What the class covers
- Typographic conventions

#### Linux background

- The UNIX philosophy
- The Cathedral and the Bazaar
- Standards
- Standards and portability
- Free software licenses
- Linux distributions
- Linux versus Unix
- GNU Info
- System Calls and Library Functions
- Error handling
- Lab

#### GNU make

- Introduction
- The Makefile
- Variables
- Implicit Rules
- Adding to make's Implicit Rules
- Special variables
- Special target conventions
- The commands to build the target
- Including other Makefiles
- GNU make versus other makes
- Lab

#### Autotools

- Introduction
- Ideas behind Autotools
- autoconf
- automake
- Libraries under Linux
- Shared Libraries
- Shared Library naming
- Library utilities
- libtool
- Autotools organization summary
- Summary
- Lab

#### Debugging with gdb

- Introduction
- Compiling programs to be debugged
- Starting and exiting gdb
- Working in gdb
- Breakpoints and watchpoints
- Continuing and single stepping
- Viewing data and the stack
- Demonstration
- GUI front ends for gdb
- Demonstration
- Lab

### **Version Control Systems**

- Uses of version control packages
- Method of using a version control system
- Concepts in version control
- Versions
- rcs keywords
- The Utility cvs
- BitKeeper
- Summary
- SourceForge
- Lab

### **Programming Tools**

- UI generators
- Demonstration
- Syntax-directed editors/Integrated development environments
- Demonstration
- Other tools
- Lab

### **Miscellaneous Functions, part 1**

- Time functions and values
- Determining system limits at runtime
- Password and group file functions
- Lab

### **Working with files**

- The stat functions
- chmod and fchmod
- File ownership of new files
- Changing ownership
- File size
- File times
- rename
- Link count
- Linking files
- Unlinking files

### **Other I/O Topics**

- Atomic appending
- File Sharing
- Synchronous I/O
- Memory mapped I/O
- Lab

### **Advanced I/O, part 1**

- I/O Redirection
- fcntl
- File locking
- Lab

### **Working with Processes**

- fork
- Waiting for children
- Zombies
- The exec family of functions
- Lab

### **The Environment of a Process**

- Command line arguments
- Command-line argument processing
- Environment variables
- Process termination
- Dynamic memory allocation
- Set user-ID and group-ID files
- Lab

### **Daemon Processes**

- What is a daemon?
- Becoming a daemon
- Daemon error logging
- Lab

## POSIX Threads

- The POSIX thread model
- Example
- Synchronization
- Thread attributes
- Linux thread model differences from the POSIX model
- Converting from Solaris threads
- Newer thread systems
- Lab

## Interprocess Communication

- Pipes
- Shared memory
- Command-line access to IPC structures
- Lab

## Booting, part 1

- The hardware boot process
- grub overview
- grub commands
- Booting single user
- init
- Run levels
- Interactive boot
- chkconfig
- service
- Boot floppies/CDs
- Summary
- Lab

## Networking, part 1

- Network configuration
- DHCP client configuration
- xinetd
- NFS client
- NFS server
- NIS
- Automounter
- nmap
- ethereal
- Network Troubleshooting
- Summary
- Lab

## Kernel configuration

- Introduction

- Ways to configure the kernel
- Configuration options
- Other make targets
- Kernel modules
- Initial ramdisks
- Summary
- Lab

## Kernel performance issues

- Overview
- General guidelines
- Recompiling
- Benchmarks
- ext2 filesystem tuning
- ext3 filesystem tuning
- hdparm
- System limits
- Summary
- Lab

## General Security Issues

- General OS security
- Patches and keeping current
- Lab

## Security scanners

- Testing for security
- nmap
- nessus
- Lab

## Local security issues

- Disk partitioning
- Setuid files
- The sticky bit on a directory
- File attributes
- Local security scanners
- Lab

## **Intrusion Detection Systems**

- Introduction
- tripwire
- snort
- portsentry
- Commercial virus scanners
- Research directions
- Lab

## **Network access control**

- Introduction
- TCP wrappers
- xinetd
- IP Tables
- Lab

## **Java**

- Java Introduction
- Aspects to consider
- Blackdown
- Blackdown components
- Sun
- IBM
- Kaffe
- GNU GCJ
- GCJ at runtime
- Java Deployment settings
- Deploying Java
- Environment variables
- Other Java-related variables
- Executing Java programs
- Executing Java applets
- Lab

## **Linux Graphical System**

- X Windows System
- Gnome
- KDE

## **Main Open Source Server Components**

- Apache Web Server
- Database Management with MySQL/PostgreSQL
- Scripting with Perl/PHP/Python
- Tomcat for Servlets and JSPs
- Lab

## **High Availability**

- Introduction
- Storage
- CODA
- CODA on Linux
- Testing CODA
- Linux Virtual Server
- Linux Virtual Server Setup
- LVS-NAT
- LVS-DR
- LVS-TUN
- ipvsadm
- Linux HA
- Heartbeat
- Networking redundancy
- Fake
- ARP Spoofing
- Automation for true network redundancy
- Lab

## **Linux Clustering**

- Introduction
- Types of clusters
- Beowulfs
- Beowulf performance
- Running software on a Beowulf
- PVM
- MPI
- RSH
- SSH
- Real life Beowulf experiments
- Real life Beowulf experiments
- Mosix
- Running software on Mosix
- Mosix installation
- Mosix user tools
- Mosix File System (MFS)
- Activating Mosix
- Using Mosix tools
- Grid computing on Linux
- Globus Toolkit
- Sun Grid Engine
- Lab