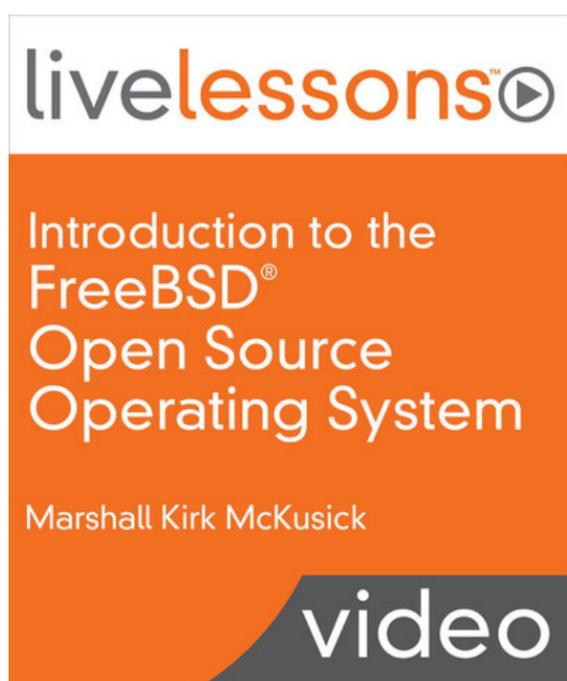


Introduction to the FreeBSD Open-Source Operating System

★★★★★ [9 reviews](#)

By [Marshall Kirk McKusick](#)



Start

TIME TO COMPLETE:
10h 32m

LEVEL:
Intermediate

TOPICS:
[BSD](#)

PUBLISHED BY:
[Pearson](#)

PUBLICATION DATE:
September 2015

This ten-hour video starts with an introduction to the FreeBSD community and explains how it differs from the Linux ecosystem. The video then goes on to provide a firm background in the FreeBSD kernel. The POSIX kernel interfaces are used as examples where they are defined. Where they are not defined, the FreeBSD interfaces are described. The video covers basic kernel services, locking, process structure, scheduling, signal handling, jails, and virtual and physical memory management. The kernel I/O structure is described showing how I/O is multiplexed and the virtual filesystem interface is used to support multiple filesystems. Devices are described showing disk management and their auto-configuration. Next the organization and implementation of the fast filesystem is described, concluding with a discussion of how to maintain consistency in the face of hardware or software failures. The video includes an overview of the ZFS filesystem. The video also covers the socket-based network architecture and layering along with a discussion of routing issues.

The presentations emphasize code organization, data structure navigation, and algorithms.

About the Instructor

Dr. Marshall Kirk McKusick writes books and articles, teaches classes on UNIX- and BSD-related subjects, and provides expert-witness testimony on software patent, trade secret, and copyright issues, particularly those related to operating systems and filesystems. He has been a developer and committer to the FreeBSD Project since its founding in 1994. While at the University of California at Berkeley, he implemented the 4.2BSD fast filesystem and was the Research Computer Scientist at the Berkeley Computer Systems Research Group (CSRG) overseeing the development and release of 4.3BSD and 4.4BSD. He earned his undergraduate degree in electrical engineering from Cornell University and did his graduate work at the University of California at Berkeley, where he received master's degrees in computer science and business administration and a doctoral degree in computer science. He has twice been president of the board of the Usenix Association and is currently a member of the Usenix Foundation Board of Directors, a member of the editing committee of the Usenix Journal, a senior member of the IEEE, and a member of the Usenix Association, ACM, and AAAS.

Skill Level

- Intermediate

What You Will Learn

- The video starts with an introduction to the FreeBSD community and explains how it differs from the Linux ecosystem. The video then goes on to provide a firm background in the FreeBSD kernel. The POSIX kernel interfaces are used as examples where they are defined. Where they are not defined, the FreeBSD interfaces are described.
- The video covers basic kernel services, locking, process structure, scheduling, signal handling, jails, and virtual and physical memory management.
- The kernel I/O structure is described showing how I/O is multiplexed and the virtual filesystem interface is used to support multiple filesystems.
- Devices are described showing disk management and their auto-configuration.
- The organization and implementation of the fast filesystem is described concluding with a discussion of how to maintain consistency in the face of hardware or software failures.
- The video includes an overview of the ZFS filesystem and covers the socket-based network architecture, layering and routing issues.
- The presentations emphasize code organization, data structure navigation, and algorithms.

Who Should Take This Course

- Applications developers, to learn how to effectively and efficiently interface with the system
- Systems programmers without direct experience with the FreeBSD kernel, to learn how to maintain, tune, and interface with such systems
- Those involved in technical and sales support, to learn the capabilities and limitations of the system
- Enterprise and mobile (Android) communities needing to understand the intricacies of FreeBSD and use its advanced features

Course Requirements

This video is directed to users who have had at least a year of experience using a UNIX-like system. They should have an understanding of fundamental algorithms (searching, sorting, and hashing) and data structures (lists, queues, and arrays).

Table of Contents

Introduction to Introduction to the FreeBSD Open Source Operating System LiveLessons

Lesson 1: Introduction to the FreeBSD Community

Lesson 1 describes the way that the FreeBSD community is organized and the way in which it designs, builds, and manages its distributions. The lesson concludes by describing the best use cases for FreeBSD versus Linux.

Lesson 2: Introduction to the Kernel

Lesson 2 provides an introduction to the FreeBSD kernel. It introduces the user process as the central facility that the kernel provides. It shows how a user process communicates with the kernel through the system call interface and uses kernel-based timers to synchronize time-based events. The lesson concludes with a description of the locking mechanisms that the kernel uses to control access to its data structures.

Lesson 3: Processes

Lesson 3 begins by detailing the components that describe a user process. It then outlines the facilities that the kernel provides to support user processes. These facilities include the scheduler that controls when processes get to run, the signal infrastructure for delivering asynchronous events to processes, and the group and session mechanisms for organizing sets of related processes. The lesson concludes by explaining the use of jails as a lightweight virtualization facility.

Lesson 4: Virtual Memory

Lesson 4 explains the management of the virtual memory used by a process. It starts by describing the layout of a process in its virtual memory and shows how the virtual memory is backed by physical memory, swap space, and executable files and libraries in the filesystem. Next it describes the different types of regions that make up an address space and shows how the kernel tracks and manages these regions. It concludes by explaining how the kernel manages the physical memory both for its own use and on behalf of the user processes.

Lesson 5: I/O System Overview

Lesson 5 examines the kernel from the perspective of its I/O hierarchy. It starts by detailing the data structures used by the kernel. [Downloading "Learning objectives.srt"...](#) shows how the kernel exports the capability to multiplex I/O operations to user processes. It describes the data structures used

to manage the socket, file, and device I/O descriptors for a process. It concludes by motivating the addition of a virtual filesystem interface and describing the virtual file (vnode) data structure and associated support facilities provided for using the virtual filesystem interface.

Lesson 6: Devices

Lesson 6 gives you an overview of how the kernel manages devices. It introduces the special device concept including the mechanisms for naming and controlling access to devices. Next it describes the volume management and disk I/O-subsystem frameworks. It concludes by showing how devices are initially identified and made available for user processes to use.

Lesson 7: The Fast Filesystem

Lesson 7 begins with a multi-decade history describing the evolution of the local FreeBSD filesystem. It reviews the kernel data structures that link from the user-process file descriptor to the local filesystem's inode data structure used to control access to and describe the location of a file on disk. Next it describes how the kernel optimizes both the I/O to a file and the layout of the file on the disk. It concludes by describing the requirements for maintaining filesystem consistency in the face of either a hardware or a software crash and the various strategies that are used to implement these requirements.

Lesson 8: The ZFS Filesystem

Lesson 8 presents an overview of the Zettabyte filesystem (ZFS) imported from Open Solaris to FreeBSD. It introduces the modules that make up ZFS and describes their functional and structural organization.

Lesson 9: Networking and Interprocess Communication

Lesson 9 introduces the networking and interprocess communications facilities of FreeBSD. It starts by showing how data flows end-to-end between two user-process sockets through the TCP/IP protocols. It then describes the socket and network protocol structures, the mbuf data buffers used to hold the data packets as they flow through the network, and the interface to the networking hardware. Next it explains the migration from version 4 to version 6 of the IP protocol. It concludes with a discussion of the design of and strategies for routing packets through the network.

Summary of Introduction to the FreeBSD Open Source Operating System LiveLessons

About LiveLessons Video Training

The LiveLessons Video Training series publishes hundreds of hands-on, expert-led video tutorials covering a wide selection of technology topics designed to teach you the skills you need to succeed. This professional and personal technology video series features world-leading author instructors published by your trusted technology brands: Addison-Wesley, Cisco Press, IBM Press, Pearson IT Certification, Prentice Hall, Sams, and Que. Topics include: IT Certification, Programming, Web Development, Mobile Development, Home and Office Technologies, Business and Management, and more. View all LiveLessons on InformIT at:

<http://www.informit.com/livelessons>.

About the Publisher

Pearson Publishing is an established educational publisher, software developer, and provider of training systems. Specialising in mobile learning, it helps raise achievement, improve staff performance and reduce costs.

[More about Pearson](#)