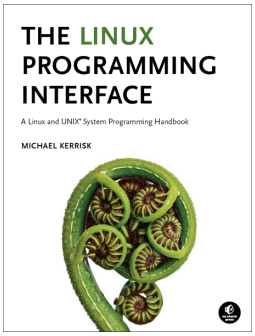
man7.org
Training and Consulting

Linux Capabilities and Namespaces

Course code: M7D-CAPNS01

This course provides an in-depth exploration of Linux namespaces, which are used in a wide array of virtualization and sandboxing technologies such as Docker, LXC, Flatpak, Firejail, Systemd, and various web browsers. In addition, the course covers the Linux capabilities model, since an understanding of that model is essential to understanding the operation of user namespaces, which are a cornerstone of many of the aforementioned applications. Detailed presentations coupled with carefully designed practical exercises provide participants with the knowledge needed to understand, design, develop, and administer such applications.



Audience and prerequisites

The primary audience comprises designers and programmers building privileged applications, container applications, and sandboxing applications. Systems administrators who are managing such applications are also likely to find the course of benefit.

Participants should have a good reading knowledge of the C programming language and solid programming experience in a language suitable for completing the course exercises (e.g., C, C++, Go).

Course materials

- A course book (written by the trainer) that includes all course slides and exercises
- A source code tarball containing all of the (many) example programs written by the trainer to accompany the presentation

Course duration and format

Two days, with around 40% of the course time devoted to practical sessions.

Course inquiries and bookings

For inquiries about courses and consulting, you can contact us in the following ways:

- Email: training@man7.org
- Phone: +49 (89) 2155 2990 (German landline)

Prices, dates, and further details

For course prices, upcoming course dates, and further information about the course, please visit the course web page, *http://man7.org/training/sec_isol_apis/*.

About the trainer

Michael Kerrisk has a unique set of qualifications and experience that ensure that course participants receive training of a very high standard:

- He has been programming on UNIX systems since 1987 and began teaching UNIX system programming courses in 1989.
- He is the author of *The Linux Programming Interface*, a 1550-page book widely acclaimed as the definitive work on Linux system programming.
- He is actively involved in Linux development, working with kernel developers on testing, review, and design of new Linux kernel–user-space APIs.
- Since 2004, he has been the maintainer of the Linux *man-pages* project, which provides the manual pages documenting the Linux kernel–user-space and GNU C library APIs.

Linux Capabilities and Namespaces: course contents in detail

Topics marked with an asterisk (*) will be covered as time permits.

1. **Course Introduction**
2. **Privileged Programs**
   - Process credentials
   - Set-user-ID and set-group-ID programs
   - Changing process credentials
   - A few guidelines for writing privileged programs
3. **Capabilities**
   - Process and file capabilities
   - Setting and viewing file capabilities
   - Text form capabilities
   - Capabilities and *execve()*; further capability sets
   - Ambient capabilities
4. **Capabilities: Further Topics (*)**
   - Root, UID transitions, and capabilities
   - Making a capabilities-only environment: securebits
   - Programming with capabilities
5. **Namespaces**
   - Namespace types
   - Mount namespaces
   - UTS, IPC, cgroup, and network namespaces
   - PID namespaces
6. **Namespaces APIs**
   - API Overview
   - Creating a child process in a new namespace: *clone()*
   - `/proc/PID/ns`
   - Entering a namespace: *setns()*
   - Creating a namespace: *unshare()*
   - PID namespaces idiosyncrasies
   - *ioctl()* operations
   - Namespace lifetime
7. **User Namespaces**
   - Overview of user namespaces
   - Creating and joining a user NS
   - User namespaces: UID and GID mappings
   - User namespaces, *execve()*, and user ID 0
   - Security issues
   - Use cases
   - Combining user namespaces with other namespaces
8. **User Namespaces and Capabilities**
   - User namespaces and capabilities
   - User namespaces and capabilities revisited
   - Namespaced file capabilities (*)
9. **Mount Namespaces and Shared Subtrees (*)**
   - Mount namespaces
   - Shared subtrees
   - Bind mounts
   - Peer groups
   - Private mounts
   - Slave mounts
   - Unbindable mounts
10. **Network Namespaces (*)**
    - Introduction
    - Creating and deleting network namespaces
    - Executing commands inside a network namespace
    - Virtual networking devices
    - Connecting namespaces with a veth pair
    - Physical networking devices
    - Using a bridge or switch to connect namespaces
    - Connecting a network namespace to the Internet
    - Use cases for network namespaces