Exploiting the Linux Kernel (3 days)

Training description

Andrey Konovalov, xairy.io

Title

Exploiting the Linux Kernel

Overview

A 3-day Linux kernel exploitation frenzy!

This training guides researchers through the field of Linux kernel exploitation. In a series of practical labs, the training explores the process of exploiting kernel bugs in a modern Linux distribution on the x86-64 architecture.

The training is structured as a series of lectures, each followed by one or more hands-on labs. The goal of each lab is to write a Linux kernel exploit following the techniques described during the lecture.

The training starts with beginner topics but proceeds into advanced areas as well. The beginner chapters include learning how to escalate privileges and bypass foundational mitigations in x86-64 kernels. The advanced chapters are dedicated to the modern slab (heap) exploitation techniques and include an in-depth analysis of the kernel allocators' internals.

The core requirement for this training is the ability to read and write C code. Basic knowledge of the x86-64 architecture and assembly, GDB, and the common binary exploitation techniques would also come in handy. There is no need to know any Linux kernel internals: all required parts are covered during the training.

Key learning objectives

- Security-relevant Linux kernel internals and attack surface.
- Kernel privilege escalation techniques.
- Exploiting stack, global, and slab (heap) vulnerabilities.
- Exploiting use-after-free and out-of-bounds vulnerabilities.
- KASLR, SMEP, SMAP, and KPTI bypasses.
- In-kernel Return-Oriented Programming (ROP).
- Cross-cache and cross-allocator exploitation.
- Data-only kernel exploitation.

Student requirements

- Working C knowledge.
- Familiarity with x86-64 architecture and x86-64 assembly.
- Familiarity with GDB (GNU Debugger).
- Familiarity with common types of vulnerabilities and exploitation techniques for userspace applications.

No knowledge about Linux kernel internals is required.

Hardware requirements

- At least 100 GB of free disk space.
- At least 16 GB of RAM.
- Ability to plug in an untrusted USB drive (relevant for corporate laptops).

Software requirements

- Host OS: Linux (recommended) or Windows.
- VMWare Workstation Player or Pro.
- 7-Zip.

Provided to students

A USB drive with:

- Presentation slides.
- Detailed lab guides with step-by-step instructions.
- Virtual machine images with tools, exercise binaries, and source code.

Course agenda

Day 1 — Internals and exploitation basics:

- Internals and debugging: x86-64 architecture refresher; security-relevant Linux kernel internals and attack surface; types of kernel vulnerabilities; setting up a debugging environment with VMWare; using GDB to debug kernel and its modules.
- Escalating privileges: ret2usr, overwriting cred structure, overwriting modprobe_path; arbitrary address execution and arbitrary address read/write primitives.

Day 2 — Mitigation bypasses and slab exploitation basics:

- Bypassing mitigations: KASLR, SMEP, SMAP, and KPTI bypass techniques; in-kernel Return-Oriented Programming (ROP).

- Exploiting slab corruptions: slab out-of-bounds and use-after-free vulnerabilities; in-depth SLUB internals; slab spraying; slab-specific mitigations.

Day 3 — Modern slab exploitation:

- Cache merging and accounting; userfaultfd and FUSE; data-only exploitation; elastic objects; msg_msg-based exploitation techniques; cross-cache and cross-allocator attacks.
- Beyond: learning more advanced exploitation techniques; useful references.

Trainer's bio

Andrey Konovalov is a security researcher focusing on the Linux kernel.

Andrey found multiple zero-day bugs in the Linux kernel and published proof-of-concept exploits for these bugs to demonstrate the impact. Andrey contributed to several security-related Linux kernel subsystems and tools: KASAN — a fast dynamic bug detector; syzkaller — a production-grade kernel fuzzer; and Arm Memory Tagging Extension (MTE) — an exploit mitigation.

Andrey spoke at security conferences such as OffensiveCon, Android Security Symposium, Linux Security Summit, LinuxCon, and PHDays. Andrey also maintains a collection of Linux kernel security–related materials and a channel on Linux kernel security.

See xairy.io for all of Andrey's articles, talks, and projects.