universitätbonn

Address Space Layout Randomization in Linux 4.19.19

Felix Bilstein

Seminar Paper

Institute for Computer Science IV
Work group Communication Systems
University Bonn

Statement of Authorship

I hereby confirm that the work presented in this seminar paper has been performed and interpreted solely by myself except where explicitly identified to the contrary. I declare that I have used no other sources and aids other than those indicated. This work has not been submitted elsewhere in any other form for the fulfilment of any other degree or qualification.

Bonn, April 16, 2019

Felix Bilstein

Abstract

While modern operating systems offer a high level of security, there are still working exploits being actively developed. Known and simple vulnerabilities like buffer overflows, integer overflows, format string attacks and more complex vulnerabilities like race conditions, usage of uninitialized memory and use-after-free are used together to exploit vulnerable software and therefore mitigate defense technologies like non-executable memory sections, stack canaries, address space randomization as well as modern compiler tool-chain hardening approaches.

In this work, we will provide an overview about the current solutions defending different attack scenarios with a focus on Address Space Layout Randomization (ASLR) and its impact on the security of running software.

Contents

1 Introduction

Today, information systems are faced by different types of attacks on a daily base[?]. There exist various overflow and injection attacks used by different threat actors when exploiting modern operating systems and their software.

In this work, we will take a deeper look into the current implementation of the Address Space Layout Randomization (ASLR) of the Linux operating system and will briefly discuss various hardening techniques focussing on how they are affected by ASLR implemented by the Linux kernel developers as well as how these solutions work in real world requirements (e.g. with glibc implementations, 32 bit vs. 64 bit architectures). We focus on the Linux Kernel version 4.19.19 which was released on 31st January 2019[?].

2 ASLR Design

2.1 Limitations

3 Related work

In this section several security tools and related work are presented. A wide range of projects which are capable to protect binaries from being exploited or help to detect the exploitation attempts when executing exploit code.

ASLR was introduced to Linux by the Pax Team in July, 2001 [?].

4 Summary

5 Bibliography