Laboratory Assignment: Digital Rights Management and Trusted Computing

Daniel Bosk*

drm.tex 2169 2015-01-14 22:37:01Z danbos

Contents

1 Introduction

This laboratory assignment treats the area of trusted computing and the problem facing digital rights management (DRM). The problem with DRM systems is that they must protect something in a hostile environment where the adversary controls everything.

A possible solution to this problem is to introduce a hardware module as support. One such module is the Trusted Platform Module (TPM), which was introduced in a cooperation between Microsoft, Intel, IBM, HP, and Compaq [1]. Their purpose was to support DRM. A newer development in this area is the UEFI secure boot, functionality utilized by the latest versions of Microsoft Windows, where the hardware refuses to boot the operating system if it is not cryptographically signed by a given key [2].

However, without this support we will see that it is basically impossible to protect programs from modification or data from copying. The only reason the TPM prevents this is because it is hard for the adversary to modify this hardware. But with an adversary like NSA, how do you know you still run on the

same hardware? NSA's Tailored Access Operations (TAO) unit can accordingly to [3] intercept your hardware on its way to you and install implants before you even see your hardware.

2 Aim

After completion of this assignment you will:

- Have insight into the problems concerning digital right management systems.
- Be able to reflect on the security of software with a trusted computing base with hardware support, e.g. Trusted Platform Module.

3 Theory

For this assignment you should first read chapters 3, 4, 5, 16, 18, 22 in *Security Engineering* [1]. Then you should read chapters 10, 14, 15 in *Computer Security* [4].

After reading the material given above you need to know about programming in assembler, specifically x86-64 assembler and some tools. For this you should read "x86-64 Machine-Level Programming" by Bryant and O'Hallaron [5]. You also need to be acquainted with some tools, study the manual pages for objdump(1), as(1), and gdb(1).

4 Assignment

This section covers the work to be done and the next section covers how it will be examined, and what to be done to pass it.

There is a program with a very simple DRM found in URL

`http://ver.miun.se/courses/security/labs/cpager-drm`.

It is an ELF 64-bit LSB executable (x86-64, dynamically linked, stripped) for a GNU/Linux system. (This same program happens to have its source code published under a BSD-license without any DRM on the same server.) The first of this assignment is to break this DRM. This first part of the assignment will be solved together during a full-class hackathon in the computer lab. There will be a projector with the code for all to see, then we will rotate who will be by the keyboard writing what the rest of the class is saying. This way we will discuss together and write the code together, everyone will thus participate in the process.

The second part of the assignment is to dicsuss the consequences of this, among other things we will discuss the following two questions:

- What is the purpose of DRMs that can easily be circumvented?
- Can we implemnt a DRM which actually works? What do we need for this?

5 Examination

To pass this assignment you must first actively participate in the hackathon lab session. If you cannot participate in the lab session you have to solve the lab yourself, then orally present your solution during one of the lab sessions after the course-end.

You must also actively contribute to the post-coding discussions. For those who cannot attend the hackathon there will be post-coding discussions during the lab sessions after the course-end.

References

[1] Ross J. Anderson. *Security Engineering. A guide to building dependable distributed systems*. 2nd ed. Indianapolis, IN: Wiley, 2008. ISBN: 978-0-470-06852-6 (hbk.) URL: `http://www.cl.cam.ac.uk/~rja14/book.html`.

[2] Microsoft. *Secure Boot Overview*. Feb. 2014. URL: `http://technet.microsoft.com/en-us/library/hh824987.aspx`.

[3] Bruce Schnier. "More about the NSA's Tailored Access Operations Unit". In: *Schnier on Security* (Dec. 2013). URL: `https://www.schneier.com/blog/archives/2013/12/more_about_the.html`.

[4] Dieter Gollmann. *Computer Security*. 3rd ed. Chichester, West Sussex, U.K.: Wiley, 2011. ISBN: 9780470741153 (pbk.)

[5] David R. Bryant Randal E. and O'Hallaron. *x86-64 Machine-Level Programming*. Sept. 2005. URL: `https://www.cs.cmu.edu/~fp/courses/15213-s07/misc/asm64-handout.pdf`.