Technology Transfer in Computing Systems

D3.2: Individual TTP2 abstract

Project no.:	609491
Funding scheme:	Collaborative project
Start date of the project:	1st September 2013
Duration:	36 months
Work programme topic:	FP7-ICT-2013-10
Deliverable type:	Report
Deliverable reference number:	ICT-609491 / D3.2
WP and tasks contributing:	WP 3 / all
Due date:	31/10/2014
Actual submission date:	15/11/2014 (updated on: 30.03.15)
Responsible Organization:	UGENT
Dissemination Level:	Public
Revision:	1.0

TETRACOM D3.2: Software Protection of Native Android Libraries

Bjorn De Sutter, Koen De Bosschere (Ghent University), Parashuram Chawan (Samsung Research UK)

Need for software protection of native Android libraries

As more and more sensitive data are stored on and communicated through almost always connect devices such as smart phones, tablets, and smart TVs, the need for protecting applications against attacks grows.

To protect software against reverse engineering, which is almost always the first step in so-called Man-At-The-End attacks that aim for identifying vulnerabilities or possibilities for tampering and leaking data, software obfuscation plays an important role. Obfuscations are software transformations that increase the apparent complexity of an application, thus forcing attackers to invest more effort in their reverse-engineering and tampering attacks. They can be applied at the source code level, in a compiler, or in post-pass tools such as link-time rewriters.

On modern platforms, such as the Android platform, the most sensitive and performance critical code is often found in native dynamic libraries. It is therefore important to be able to protect such libraries. To avoid increasing the time-to-market, the protection approach should as much as possible with standard compilers, such that developers do not have to change their development processes.

Diablo: a link-time binary code rewriting framework

For over a decade, a link-time code rewriting framework called Diablo has been developed in Ghent University's Computer System Lab. On top of this framework, tools have been developed to improve the performance of applications, to reduce their size, to protect them against hardware fault-injection attacks, and to obfuscate binaries. Diablo supports multiple architectures, incl. x86 and ARM, but as this was a research tool, it lacked some important features to deploy it on real-life platforms such as Android.

Extending Diablo for use on Android

In this technology transfer project, a major development effort was invested in the Diablo codebase to support features needed to protect native Android libraries. New supported features include

- full ARMv7 instruction set support, including NEON and Thumb2,
- support for dynamically linked binaries, besides the existing support for statically linked binaries, and including PIE as well as non-PIE binaries;
- support for dynamically linked libraries;
- support for profiling such binaries and libraries;
- support for rewriting C++ applications, besides the existing support for C and Fortran applications,
- support for recent GCC, LLVM, and binutils versions used in the software development kits for Android: GCC 4.8, GCC4.6, LLVM3.2, 3.3, and 3.4, and binutils 2.23, incl. support for Linux as well as Android API level 18;
- support for protected memory segments in line with modern SELinux requirements.

The developed support was not only tested on SPEC benchmarks, but also on several system tools that invoke very different parts of the standard system libraries.

On top of those features that related to the basic link-time binary code rewriting infrastructure of the Diablo framework, additional software protections were developed as well. More concretely, the existing obfuscations in Diablo, that were previously limited to x86 code, have been improved, and ported to the ARMv7 platform. As such, Diablo now supports obfuscations based on opaque predicates, control flow flattening, branch functions, code inlining and outlining, and code layout randomization for both the x86 and ARMv7 architectures.

The obfuscations as well as some of the code optimizations have also been integrated in a feedback-directed software diversification framework that supports both x86 and ARMv7 binaries and libraries.

A security evaluation indicates that reverse engineering tools use by hackers all over the work, like IDA Pro and BinDiff are effectively hampered when Diablo's obfuscations and transformations are applied to protect applications.