A Sample *ACM* SIG Proceedings Paper in LaTeX Format*

Ke Zhang
3030058805
Department of Computer Science
University of Hong Kong
Pokfulam Road, Hong Kong
kzhang2@cs.hku.hk

Tianxiang Shen
3030058776
Department of Computer Science
University of Hong Kong
Pokfulam Road, Hong Kong
txshen2@cs.hku.hk

ABSTRACT

This paper provides a sample of a LaTeX document which conforms to the formatting guidelines for ACM SIG Proceedings. It complements the document *Author's Guide to Preparing ACM SIG Proceedings Using LaTeX2$_\epsilon$ and BibTeX*. This source file has been written with the intention of being compiled under LaTeX2$_\epsilon$ and BibTeX.

The developers have tried to include every imaginable sort of "bells and whistles", such as a subtitle, footnotes on title, subtitle and authors, as well as in the text, and every optional component (e.g. Acknowledgments, Additional Authors, Appendices), not to mention examples of equations, theorems, tables and figures.

To make best use of this sample document, run it through LaTeX and BibTeX, and compare this source code with the printed output produced by the dvi file.

1. INTRODUCTION

This is introduction.

2. BACKGROUND

2.1 Secure Enclave

A secure enclave is a set of software and hardware features that together provide an isolated execution environment to enable a set of strong security guarantees for applications running inside the enclave. Enclave allows user-level as well as Operating System (OS) code to define private regions of memory, whose contents are protected and unable to be either read or saved by any process outside the enclave itself, including processes running at higher privilege levels [].

Primally, secure enclaves can provide confidentiality, integrity, and attestation. Confidentiality guarantees that an adversary outside of the enclave cannot inspect the state of execution inside the enclave even if they compromise the operating system or correctness of the computation running inside the enclave even if the operating system has been compromised or a user attempts to subvert the execution of the program inside the enclave. Finally, hardware-based attestation provides an unforgeable proof that enables a remote party to verify what has run inside the enclave even if they dont have physical access to the machine. A secure enclave thus provides a powerful cornerstone for secure computing and development of secure systems in general.

Intuitively, secure enclave fundamentally ensures the correctness and isolation in executing given process. The confirmation of input data freshness is hard to achieve, especially when the enclave encounters crash or restart. There are several wildly used secure enclave services [], one of the most popular security architectures is Intel Software Guard Extensions (SGX) []. However, a mature secure enclave designation as SGX still shows unsatisfied performance towards rollback attacks. In this report, we focus on the SGX architecture and its existing promotions in proposing protection against rollback attacks.

2.2 Rollback Attack

Rollback attacks, also as known as the replay attacks, remain a potential secure problem in secure enclave. In a rollback attack, attackers replace the latest data with an older version without being identified by the system.

Data integrity violation through rollback attacks can have severe implications. Consider, for example, a financial application implemented as an enclave. The enclave repeatedly processes incoming transactions at high speed and maintains an account balance for each user or a history of all transactions in the system. If the adversary manages to revert the enclave to its previous state, the maintained account balance or the queried transaction history does not match the executed transactions.

In reality, enclaves cannot easily detect this replay, because the processor is unable to maintain persistent state across enclave executions that may include reboots or crash. Another way to carry out rollback attacks in secure enclaves is to create multiple instances of a same process and route update requests to one instance and read requests to the other. Due to the characteristic of secure enclave, the instances are indistinguishable to remote clients or OS.

*(Produces the permission block, copyright information and page numbering). For use with ACM_PROC_ARTICLE-SP.CLS V2.6SP. Supported by ACM.

To avoid rollback attacks, most commonly considered direction is to record the time related information for every state change. In this paper, we mainly discuss three designations in rollback attacks protection built on the SGX architecture.

3. PROBLEM

4. SOLUTIONS

In this section, we mainly introduce three state-of-art solutions in solving the problem we mention in Section 3. We separately describe their motivation, inner designations, and respective improving directions in detail.

4.1 SGX Counter

4.2 ROTE

4.3 Speicher

5. RUNTIME

This is runtime.

6. EVALUATION PLAN

This is evaluation part.

7. CONCLUSION

This is conclusion.

8. ACKNOWLEDGMENTS

This section is optional.

9. REFERENCES