Forking Attacks: Two Ways, No Worries

Matija Roncevic
matija.roncevic@fau.de
Friedricht-Alexander-Universität Erlangen-Nürnberg

1 INTRODUCTION

Almost every noteworthy application has embraced cloud computing. The range of applications that benefit from cloud services spans from small smartphone apps to sophisticated large language models to commonly used GitHub repositories. The clear reason for its popularity is the ability to store, manage, and process data more efficiently than ever before. The shift to cloud services offers numerous advantages, including scalability, flexibility, and cost-effectiveness. However, it also introduces significant security challenges. With the inclusion of numerous critical applications and sensitive data, the importance of secure environments has increased. Cloud security involves a diverse range of practices, technologies, and policies aimed at protecting data, applications, and services distributed across cloud environments. The baseline of protection is realized by implementing firewalls and encryption; however, the sophisticated and evolving nature of cyber threats often outpaces the safety measures these solutions provide. This calls for advanced security solutions that can safeguard data even in potentially compromised environments.
Trusted Execution Environments (TEE) like Intel Software Guard Extensions (SGX) is such an advanced security solution. Intel SGX is a set of hardware-based security features that create isolated execution environments, known as enclaves [1]. (Enclaves are secure areas of memory where sensitive data and code can be executed and stored in isolation from the rest of the system.) These enclaves are designed to protect sensitive data and code from being accessed or modified by unauthorized parties, even if the operating system is compromised. SGX provides a robust mechanism for ensuring the confidentiality and integrity of data processed in the cloud.Despite its strengths, Intel SGX still has its own vulnerabilities. The enclaves itself are in fact isolated from adversaries, however the relaiability on the inputs can not be ensured without additional savety meassures. Those attacks are labeled as rollback attacks [2]. (More explanation about rollbacks, maybe in background?) Forking attacks on the other hand, aim on creating multiple instances of an enclave, leading to unauthorized data access and manipulation.This is accomplished by running those simulainously and exploiting the fact, that all those enclaves will return correct but often stale states. I.e. counters for passwort attempts can be reset to gain unlimited tries, despite the limit for tries is set to an finite number. Understanding these attacks and developing effective mitigation strategies is essential for ensuring the security and trustworthiness of cloud-based applications relying on SGX.

2 BACKGROUND

The concepts which are discussed in this paper, will be explained further to provied an comprenhensive knowledge beforhand.

2.1 Trusted Execution Environment, TEE

TEEs aim to increae the saveness of data and exacerbate the tempering with crucial code sections, by allowing the processor to have protection through different implementations like Intels SGX enclave system.

2.2 Denial of Service (DoS) Attacks

2.3 Liveness

2.4 Safety

3 BODY

This section will introduce two ways to mitigate forking attacks. Each method has its own implementations for tackling the challenges, thereby bringing advantages, disadvantages, and preferable fields of application.

3.1 The Blockchain approach - Narrator-Pro

This system relies on an external blockchain and can be broken down to three main concepts which combined yield in its prevention of attacks. In more detail the goals can be described as:

- Security - The safety and liveness properties of the TEE programms will be protected.
- Performance - While providing the Security Goals there will be no decisive detriment of performance. In detail low latency for state updates and read operations, high throughput for provessing enclae program requests and unlimited state updates, provided by the blockchain. //Check thruthness

Before getting into what these concepts look like //different wording// it is essential to give an overview on the system so the coherences will be clear.
Several SGX enabled maschines are running in a cloud. These maschines can run a number (limited by specifications of the system) of enclaves which are devided into two groups of Application Enclaves (AEs) and State Enclaves (SEs). AEs have applications running, handle client requests and return outputs corresponding to its inputs. SEs on the other hand contain the Narrator-Pro software and are responsible to provide state continuity to AEs. This is accomplished by a connection from the AE to a lokaly running SE, where the AE can use SEs Narrator-Pro libraries to seal data. This data is then used to retrieve the latest sealed state. This priciple is expained in more detail in section .... -Figure 1- provides an overview of the mentioned components.
The main concepts which constitute in the reliance of Narrator-Pro are (1) system initialization §..., (2) state update and read protocols for AEs §... and (3) restart protocols in regard to AEs and SEs §....
Before explaining these protocols we want to state a few ...(premises) which Narrator-Pro does.

- Denial of Service Attacks - It is not the goal to prevent systems from these kinds of attacks. Since TEEs themselves do not have preventing measures includet.
- Hardware - The implementation of Narrator-Pro should neader require any hardware changes nor will there be a need of specific hardware if the cloud TEE is already running.

4 CONCLUSION

REFERENCES

[1] Samira Briongos, Ghassan Karame, Claudio Soriente, and Annika Wilde. No forking way: Detecting cloning attacks on intel sgx applications. In *Proceedings of the 39th Annual Computer Security Applications Conference*, ACSAC '23, New York, NY, USA, 2023. Association for Computing Machinery.

[2] Wei Peng, Xiang Li, Jianyu Niu, Xiaokuan Zhang, and Yinqian Zhang. Ensuring state continuity for confidential computing: A blockchain-based approach. *IEEE Transactions on Dependable and Secure Computing*, pages 1–14, 2024.