SYMCUBE: Timing Trace Smoothing and Jitter Randomization Design

1. Introduction

This document details the design and implementation of timing trace countermeasures within the SYMCUBE unlock validation process. The goal is to mitigate the risk of timing side-channel attacks, where an adversary attempts to deduce information about the unlock sequence by precisely measuring the time taken for various operations. This design incorporates interrupt-based timing equalization, randomized delay insertion, and power-jitter synchronization to obscure the timing signature of a valid unlock attempt.

2. System Overview

The SYMCUBE unlock validation process involves a sequence of rotations and symbolic transformations. Each stage of this process takes a certain amount of time to complete. An attacker might try to measure these timings to gain information about the correct sequence. The countermeasures described here aim to make these measurements unreliable.

3. Timing Equalization Strategies

To minimize variations in execution time between different stages of the unlock process, we employ interrupt-based timing equalization:

1. **Interrupt Handling:**Critical sections of the unlock validation code are segmented into smaller, interruptible units. A high-priority timer interrupt is configured to trigger at pseudo-random intervals during the validation process.
2. **Context Switching:**When the interrupt occurs, the current validation process is temporarily suspended, and the interrupt handler is executed. The interrupt handler performs a minimal amount of work unrelated to the validation.
3. **Equalized Execution Time:**By strategically inserting these interrupts, we break up the execution flow and introduce a degree of timing variability. This makes it more difficult for an attacker to isolate the timing of specific operations. The interrupt frequency is varied to prevent easy subtraction of the interrupt overhead.

4. Randomized Delay Insertion (RDI) Mechanisms

Further timing obfuscation is achieved through the insertion of randomized delays:

4. **Entropy-Driven Delay:**After each stage of the unlock validation (e.g., after a rotation or symbolic transformation), a small, cryptographically secure delay is introduced. The duration of this delay is determined by a value derived from the SYMCUBE's environmental entropy source.
5. **Dynamic Delay Range:**The range of possible delay values is dynamically adjusted based on the overall timing profile of the unlock attempt. This prevents an attacker from simply averaging out the delays.
6. **Non-Blocking Delays:**The delays are implemented using non-blocking techniques (e.g., hardware timers with interrupts) to minimize performance impact on legitimate unlock attempts.

5. Power-Jitter Synchronization

To further decouple timing variations from the actual unlock process, we introduce power-jitter synchronization:

7. **Controlled Power Fluctuations:**The SYMCUBE's power consumption is intentionally modulated during the unlock validation process. This modulation is synchronized with the randomized delays, making it difficult to distinguish between timing variations caused by the unlock operations and those caused by power fluctuations.
8. **Frequency Modulation:**The frequency and amplitude of the power fluctuations are varied pseudo-randomly, driven by the environmental entropy source.
9. **Decoupling Effect:**This technique makes it significantly harder for an attacker to use power analysis in conjunction with timing analysis, as the power consumption, which often correlates with execution time, is now deliberately obfuscated.

6. Evaluation of Trace Signature Suppression vs. Unlock Accuracy

A critical trade-off exists between the effectiveness of timing trace suppression and the accuracy of legitimate unlock attempts. Excessive randomization can introduce timing variations that make it difficult for the SYMCUBE itself to reliably validate a correct unlock sequence.

10. **Statistical Analysis:**We employ extensive statistical analysis to characterize the timing variations introduced by the countermeasures. This involves measuring the timing of a large number of successful unlock attempts under various conditions (temperature, voltage fluctuations, etc.).
11. **Adaptive Thresholds:**The unlock validation logic uses adaptive thresholds that account for the expected timing variations. These thresholds are dynamically adjusted based on the measured statistical distribution of unlock timings.
12. **False Rejection Rate:**The goal is to minimize the false rejection rate (i.e., the probability of a legitimate unlock attempt being rejected) while maximizing the effectiveness of timing trace suppression. We aim for a false rejection rate of less than 0.001% under normal operating conditions.

7. Metrics for Timing Noise Entropy Contribution

To quantify the effectiveness of the jitter randomization, we define metrics for the entropy contribution of the timing noise:

13. **Jitter Entropy:**We measure the amount of randomness (entropy) introduced into the timing traces by the randomized delays and power-jitter synchronization. This is done using statistical tests such as the NIST Statistical Test Suite.
14. **Trace Entropy Ratio:**We calculate the ratio of entropy in the timing traces introduced by the countermeasures to the entropy that would be present in the traces without the countermeasures. A higher ratio indicates more effective suppression of the original timing signature.