Security Advisory

Title	Security Advisory Concerning Timing Attacks on ECDSA Peripheral in ESP32-H2
Issue Date	2024/11/26
Advisory Number	AR2024-007
Serial Number	NA
Version	V1.0

Issue Summary

There is a hardware vulnerability in ESP32-H2 SoCs before Chip revision v1.2 where the ECDSA peripheral does not operate in constant time, making it susceptible to timing attacks. However, enabling secure boot significantly reduces the likelihood of such attacks, as they require a large sample set with controlled data patterns.

What is a Lattice Attack?

The lattice attack proves that knowing just a few bits of *k* for multiple signatures can be sufficient to compromise the full private key, where *k* is the nonce in the ECDSA algorithm.

The attacks on the ESP32-H2 described in this statement obtain signatures where the corresponding *k* has many zeros as the most significant bits with high probability. This is achieved by generating numerous signatures, filtering out, and retaining those where the generation time was below a certain threshold, and then applying the BKZ lattice reduction algorithm to obtain the ECDSA private keys.

Issue Details

This attack exploits the differences in calculation time for various multipliers when the ECDSA peripheral performs ECC point multiplication and uses the lattice attack algorithm to obtain the ECDSA private key. Additionally, current eFuse

configurations have no way to distinguish between the ECDSA P192 and ECDSA P256 key purposes, thus reducing the effort required to perform the timing analysis.

Impact Analysis

1. To carry out the attack, an attacker would need to measure the time taken by the hardware to generate the ECDSA signature. This can be done in a few ways, each with significant limitations:

    - An attacker could remotely exploit the ECDSA signature timing, but this requires very stable communication delays and elimination of other time uncertainties.

    - An attacker could use power analysis to measure ECDSA signature time, but this requires a measurement device and physical access to the chip.

    - An attacker could directly operate the ECDSA peripheral to measure signature time, but this requires bypassing several security mechanisms like secure boot and flash encryption, making it very costly.

2. The complexity of the attack can be further reduced by first guessing the 192 bits using ECDSA P-192 curve operations using timing analysis and then brute forcing the remaining 64 bits of the ECDSA P-256 private key.

3. Implementing the attack requires a large number of samples and considerable effort and time. The accuracy of measuring ECDSA signature time affects the success rate. Signatures generated faster than a threshold are targeted, but even a slight decrease in accuracy significantly increases attack time.

Affected Product Series:

The lattice attack based on time analysis is theoretically applicable to all SoCs that include an ECC accelerator and support hardware-accelerated ECDSA signatures, which is the ESP32-H2 (<v1.2). We have implemented software fixes and have introduced hardware countermeasures in ESP32-H2 v1.2. Please note that the hardware countermeasure is already available in ESP32-C5 and ESP32-C61 and shall be incorporated in all future SoCs to prevent this security vulnerability.

Affected SoC:

SoC	Affected Chip Revision	Fixed Chip Revision
ESP32-H2	< v1.2	v1.2 onwards

Note: Please refer to Chip Revision Identification for identifying the Chip revision of the ESP32-H2 SoC.

Mitigation

Software Countermeasure

At present, there is a software countermeasure in the ECDSA driver (to randomize the power profile and to make it appear constant time) for the affected chip revisions of ESP32-H2 (< v1.2). This countermeasure requires Secure Boot to be enabled for full effectiveness. For the affected revisions of ESP32-H2, the software countermeasure is enabled by default in the following versions of the SDK

ESP-IDF Patched Versions:

ESP-IDF Branch	Fixed Commit IDs	Fixed ESP-IDF Versions
master	5bfa1fb	NA
release/v5.3	4f29e3f	Expected in 5.3.2
release/v5.2	2b2869a	Expected in 5.2.5
release/v5.1	8b2abcc	5.1.5

Hardware Countermeasure:

We have implemented a secure mode in the ECDSA and ECC peripheral hardware, in which ECC operations are performed in constant time and constant power. In this way, we can eliminate the leakage of the bit value of k in terms of runtime or power consumption.
In addition, we have added a new eFuse configuration bit to limit the ECDSA hardware to use only a fixed ECC curve for signing/verification. Please note that these fixes are already available in ESP32-H2 v1.2 chip revision.

Other Espressif Product

This issue impacts only those SoCs that support the ECDSA peripheral. Among these, the ESP32-C5 and ESP32-C61 already have hardware countermeasures in place. Other SoCs do not support ECDSA.

Credits

We would like to thank Emil Lenngren, an exceptional hardware and software engineer, for reporting this vulnerability and following up on responsible disclosure.