Passport Prime Security Audit

Sep 2, 2025

Prepared by:

Keylabs

Nedos Consulting EMEA FZ-LLC
FDRK3801
Compass Building,
Al Shohada Road,
AL Hamra Industrial Zone-FZ,
Ras Al Khaimah
United Arab Emirates
contact@keylabs.io

Table of Contents

Executive Summary

This document presents a comprehensive security audit of the Foundation Passport Prime hardware wallet, conducted to evaluate its resistance to cryptocurrency-specific threats and validate its security architecture. The analysis encompasses four key areas: a detailed threat model identifying adversary profiles and attack vectors specific to hardware wallets; an architecture assessment examining the device's defenses against malware, physical attacks, supply chain compromises, and social engineering; a hardware security audit evaluating the SAMA5D2 microcontroller and ATECC608C secure element implementation, including tamper detection, secure boot, and cryptographic operations; and a firmware security audit identifying potential vulnerabilities in the software implementation.

The audit methodology included physical device examination, architecture review, threat modeling based on industry-standard attack scenarios, and security testing of both hardware and firmware components. The Passport Prime demonstrated excellent security through its seed protection scheme, sophisticated use of hardware security features including SECURAM with automatic clearing, robust tamper detection mechanisms, and effective isolation of wireless communication. While several low-impact findings were identified across firmware and hardware implementations, all critical issues have been addressed through the audit process. The device's architecture and security features, as well the use of KeyOS makes it a particularly robust design against many types of attack vectors. Fault injection did not yield actionable results against the SAMA5D2 microcontroller and the way the seed is split across the SAMA5D2 and ATECC608C means that both devices would need to be exploited to extract the key in most scenarios. All known attack vectors for the ATECC608 family and the SAMA5D2 were taken into consideration in the design of the Passport Prime and effectively mitigated.

Passport Prime Threat Model

Summary

The hardware wallet threat model outlined in this section was specifically adapted to account for the features of the Passport Prime. Three main adversary types are considered: remote attackers who can compromise host devices, local attackers with physical access and hardware expertise, and insider threats who may abuse privileged access during manufacturing or distribution. The model details various threats including physical access attacks, supply chain compromises, remote malware attacks, communication interception via NFC/Bluetooth, side-channel attacks, firmware vulnerabilities, cloning attacks, and key extraction attempts. These threats can manifest through various attack vectors such as malicious software targeting the eMMC storage interface, evil maid scenarios, supply chain tampering, social engineering exploiting the high-resolution display, authentication bypasses, firmware exploitation, insecure communications between the SAMA5D2 MCU and wireless ICs, and both physical and memory-based attacks. To counter these threats, the model recommends implementing strong physical security measures leveraging the ATECC608C secure element, device verification mechanisms, secure boot and firmware processes utilizing the SAMA5D2's security macrocell, multi-factor authentication requirements, robust encryption with hardware-backed keys stored in secure elements, and regular security audits. This model was used and referenced as a basis for conducting the remaining aspects of this comprehensive security audit.

Adversary Profiles

Remote Attackers: Actors capable of remotely compromising the host or mobile phone that is used to connect to the wallet and leveraging access to it.

Local Attackers: Actors with permanent or temporary access to the wallet. These actors generally have expertise in hardware hacking, knowledge of hardware vulnerabilities of the wallet and/or vulnerabilities in the current firmware version of the wallet.

Insider threats: Actors with authorized physical and/or remote access to the wallets who may misuse their privileges, for example during device manufacturing or provisioning or by shipping malicious software or firmware.

Threats

Physical Access/Evil Maid Attacks: Temporary or permanent physical access to a user's wallet. Sufficient for a malicious actor to interact with the device.

Supply Chain Attacks: Software, firmware or hardware tampered with during manufacturing, development, distribution, or delivery. Generally in conjunction with Internal threats.

Remote Attacks, Malware and Software Attacks: Malicious code on the host or mobile phone capable of interacting with the hardware wallet.

Man-in-the-Middle (MitM) or Replay Attacks: Intercepting or manipulating the communications between the wallet and the host software as it's being sent over the wire (USB) or via wireless protocols, such as NFC or Bluetooth.

Side-Channel Attacks: Extracting information from unintended channels like power consumption or electromagnetic emissions.

Firmware Vulnerabilities: Exploiting vulnerabilities in the firmware, including the upgrade routines and bootloaders to recover or compromise the wallet and/or seed.

Cloning Attacks: Attempts to create unauthorized duplicates of the wallet hardware or software configuration to gain access to funds or compromise multiple users.

Key Extraction: Techniques to recover the seed, private key or seed phrase at rest. This may include having to brute force the PIN space.

Downgrade attacks: Techniques to downgrade the software and/or firmware to a previous vulnerable version. For hardware wallets this is generally enforced by a bootloader.

Attack Vectors

Malware: Compromised hosts and mobile phones used to interact with the wallet stealing the user's pin and/or preventing transactions or directing them to different addresses.

Evil Maid Attacks: Temporary physical access to the device, sufficient to swap the device, reflash the device, but not necessarily to physically modify the device.

Loss, Theft or Destruction: physically losing access to the hardware wallet and the cryptographic seed by physically losing, having the device stolen or destroyed.

Supply Chain: Compromised hardware introduced during manufacturing or distribution and/or compromised firmware and/or software delivered via updates.

Social Engineering: Tricking the user into entering the pin incorrectly or sending funds to the wrong address.

Authentication Bypass: Insufficient or poorly implemented mitigation of brute forcing or bypassing the authentication scheme entirely.

Firmware Exploitation: Identifying and exploiting vulnerabilities in the firmware allowing for the seed to be recovered.

Insecure Communications: Intercepting data during communication between the wallet and the host as well as between different ICs on the wallet.

Fault Injection and Physical Attacks: Several forms of physical attacks against electronic components can cause them to operate abnormally, potentially bypassing authentication and enabling features that compromise device security, such as debugging.

Secrets in Non-Volatile Memory: Physical attacks against the device often result in the full extraction of the Non-Volatile Memory (NVM) contents which may include the seed.

Secrets in Volatile Memory: It is common to be able to recover the volatile memory contents of devices.

Mitigation Strategies

Strong Physical Security: Wallets should implement tamper detection and should have a case that provides sufficient tamper-evidence for the user to see a wallet that has been tampered with.

Device Verification: Implement mechanisms for users to verify the authenticity of their own wallet, such as nicknames and device pairing. Additionally users should be able to rely on firmware signatures to check the authenticity of their device.

Secure Boot and Firmware: Employ secure boot to verify the firmware prior to execution, regularly update firmware to address vulnerabilities. Since wallets may be stored, for example in a drawer, it's important to offer firmware updates whenever they're plugged in.

Multi-Factor Authentication: Require multiple forms of authentication to access the wallet. All interaction with the wallet should require the user PIN or Passphrase or at least a button press to confirm.

Encryption: Communications between the host or mobile phone and the wallet should be encrypted to prevent malware from MitM the communications. The seed should also be encrypted at rest on the device using hardware-backed encryption with keys stored in secure elements.

Regular Security Audits: Conduct regular security assessments of changes to the device firmware to ensure that the firmware mitigates all known attacks.

Documentation Recommendations

Create detailed documentation outlining security practices, threat mitigation strategies, and incident response procedures. Foundation should publish and maintain a comprehensive Software Bill of Materials (SBOM) that lists all libraries, dependencies, and third-party components used in the firmware, enabling users and security researchers to assess potential vulnerabilities in the supply chain. Additionally, maintain a public threat model specifically tailored to the Passport Prime security architecture and unique security features of the device, providing transparency about known risks and implemented countermeasures. These threat models should be specific to the particular context and product, regularly reviewed and updated as the threat landscape evolves, and made accessible to the security community to foster collaborative security improvements.

Passport Prime Hardware Security Audit

Architecture Summary

The Foundation Passport Prime demonstrates several well-designed architectural elements that provide robust security through thoughtful hardware integration and cryptographic design. The firmware has been analyzed, and together with the hardware architecture and documented security protocols, reveals sophisticated engineering decisions.

Strengths in Security Architecture

This section highlights several security features that significantly contribute to the overall security architecture that ensure a high level of security of the overall device.

Multi-Layer Seed Protection

The device implements an elegant three-factor seed protection scheme using:
Slot 10 = Seed Bytes ⊕ OTP ⊕ SHA256(PIN + "Encrypt")`
This design cleverly distributes critical components across different security domains - the encrypted seed in the tamper-resistant ATECC608C, the One-Time Pad in hardware-cleared SECURAM, and the PIN as user knowledge. This ensures that compromise of any single component does not expose the seed.

SECURAM Utilization

The architecture makes sophisticated use of the SAMA5D28's secure SRAM capabilities. By storing the OTP and other ephemeral secrets in SECURAM with hardware automatic clearing, the design ensures that the most critical component for seed decryption is guaranteed to be destroyed on tamper detection, regardless of software execution or power availability.

Tamper Response

The split tamper architecture effectively works around the inherent limitation that Secure Elements cannot be hardware-cleared. By ensuring the hardware-clearable component (OTP) is essential for accessing the software-clearable component (encrypted seed), the design maintains security even if tamper response is interrupted by power removal.

Secure Boot Implementation

The AES-CMAC secure boot implementation with signatures provides a robust chain of trust while still making device recovery and factory resets possible. The multi-signature requirement adds resilience against individual key compromise while maintaining recoverability.

ATECC608C Integration

The secure element configuration serves the recovery-focused design goals while maintaining cryptographic protection through the multi-layer encryption scheme. The integration of PIN attempts utilizing Counter 0 and the use of various slots for different security functions demonstrates thoughtful hardware security element utilization.

Cryptographic Operations

The hardware acceleration of AES, SHA, and TRNG operations provides both performance benefits and inherent side-channel resistance compared to software implementations. The on-the-fly encryption capabilities for external memory add additional layers of protection.

Hardware Security Summary

The Foundation Passport Prime hardware implementation demonstrates attention to security through both component selection and physical design. The hardware security features provide multiple layers of protection that work cohesively to protect against both logical and physical attacks.

Hardware Security Strengths

The hardware implementation provides a robust foundation for the device's security model, with properly functioning tamper detection, effective use of secure memory, and thoughtful physical design that balances manufacturing requirements with security protection.

SECURAM

The device makes excellent use of the SAMA5D28's Security Module (SECUMOD) capabilities. The 5KB secure SRAM is properly utilized for storing critical ephemeral data including the One-Time Pad, IO Protection Secret, and other security-sensitive information. The hardware automatic clearing functionality has been implemented and tested, ensuring that tamper events reliably trigger immediate memory erasure without software dependency.

Functioning Tamper Response System

Physical testing confirmed that the tamper detection system operates as designed. The normally-open tamper switches properly detect case opening attempts and successfully trigger the security response chain. The hardware clearing of SECURAM occurs automatically upon tamper detection, providing guaranteed protection of critical secrets even in power-loss scenarios.

Secure Element Integration

The ATECC608C implementation demonstrates sophisticated hardware security practices. The secure element is properly integrated with the main processor through encrypted SWI communications using the IO Protection Secret. The hardware random number generator, secure key storage, and cryptographic acceleration are effectively utilized. The monotonic counter is effectively employed for PIN attempt limiting, showing thoughtful adaptation of hardware features to security requirements. The use of additional keyslots to secure additional secrets using the secure element is also commendable.

Physical PCB Security Design

The printed circuit board demonstrates security-conscious manufacturing practices. The PCB omits component labels and excessive test points that could aid reverse engineering efforts. While JTAG/SWD and secure element test pads are present for manufacturing purposes, they are strategically placed under the display assembly, making them extremely difficult to access without triggering tamper detection. This design provides necessary manufacturing capabilities while maintaining strong physical security.

Component Selection and Layout

The choice of the SAMA5D28 variant provides access to advanced security features including environmental monitoring capabilities and TrustZone support. The physical layout places security-critical components in protected areas of the PCB, with the secure element and main processor positioned to benefit from the tamper detection coverage.

Manufacturing Security Considerations

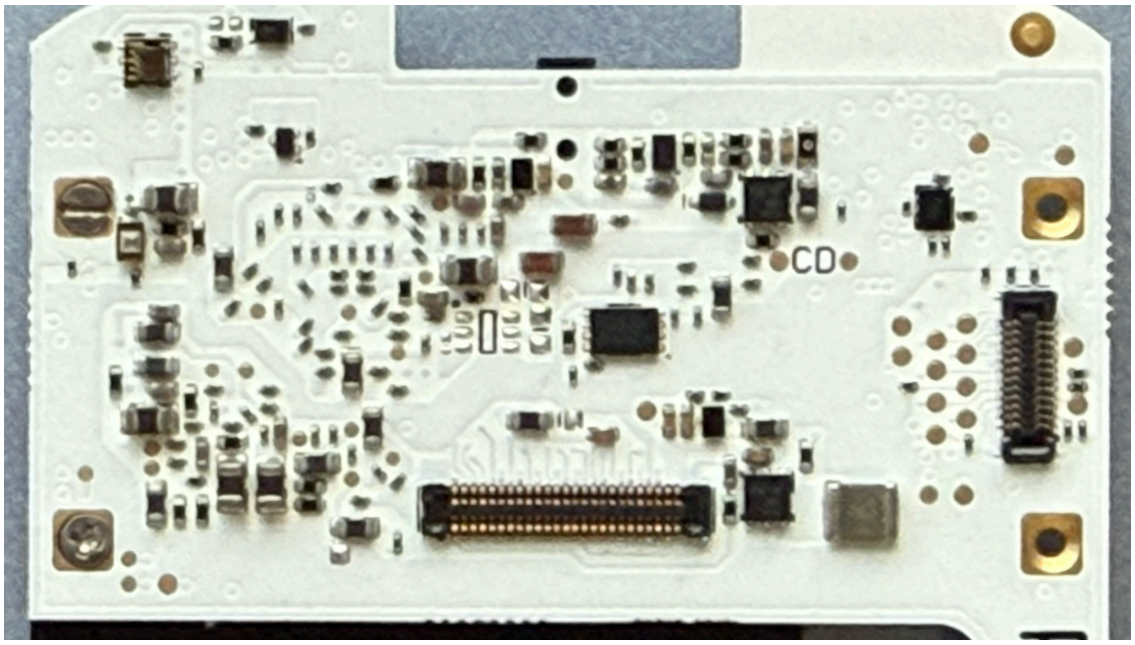
The hardware design shows awareness of supply chain security concerns. The minimal labeling and clean PCB design reduce the information available to potential attackers during brief physical access scenarios. The integration of tamper protection into the fundamental device structure (rather than as an add-on) demonstrates security-first design principles.

Hardware Security Findings

Debug Interface and Test Pad Accessibility (Impact: Low)

Description

Although debugging is disabled in production devices during factory provisioning debug and test interfaces are physically accessible on the PCB. These include JTAG/SWD debug signals for the main processor and test pads connected to the communication of the ATECC608C secure element. These interfaces provide potential attack vectors for sophisticated hardware analysis, allowing monitoring or manipulation of processor debug functions and secure element communications. All of these interfaces are covered by the device's display, which itself is protected by tamper detection pins that would trigger security responses upon disassembly attempts.



Debug header and test points.

Impact

These debug interfaces could offer access to important signals including JTAG/SWD of the wireless interface, the SAMA5D2 and the ATECC608C secure element. An attacker with access to these pins would be able to leverage them in more advanced hardware

attacks against the device, such as fault injection. Because this is not directly exploitable and would require additional attacks, the severity of this finding is low.

Recommendation

The tamper protection offered by the wallet is sufficient to protect against access to these signals. Nevertheless, it is recommended to remove all debug interfaces and test pads in production hardware to eliminate this attack vector entirely. While the current physical protection significantly reduces the risk, removing these interfaces eliminates a potential avenue for sophisticated attackers able to work around tamper protection.

Passport Prime
Firmware Security Audit

Audit Performed On:
Commit cbfe4ff50ecdfc20e3a1e01ba2fb6f92175359d8 (tag: dev-v0.9.0)

Summary

The Passport Prime's security architecture achieves defense-in-depth through hardware-enforced protections at every layer. The ATECC608C secure element provides tamper-resistant key storage with hardware monotonic counters for PIN attempt tracking, while SECURAM ensures all sensitive data is erased upon physical tamper detection. The device's innovative approach stores critical key material split between SECURAM (hardware-protected volatile memory) and the secure element, with the final encryption keys derived by XORing components from both locations with PIN-derived keys. This ensures that even if an attacker compromises both the secure element and main processor, they cannot reconstruct the seed or disk encryption keys without the user's PIN. Combined with secure boot verification, firmware rollback protection, hardware AES acceleration, and triple-source entropy generation, these layered defenses create a formidable security barrier.

Our comprehensive audit identified only 5 low-severity findings, all requiring physical device access and sophisticated attack capabilities. The absence of any critical or high-severity vulnerabilities validates the robustness of the security implementation.

Firmware Security Findings

PIN Check Timing Not Randomized (Impact: Low)

Description

PIN verification lacks random timing delays making it possible to target more easily as part of side-channel analysis (SCA) or a fault injection attack.

```
pub fn pin_login_attempt(
    crypto: &Crypto,
    se: &cryptoauthlib::Device,
    pin: &Pin,
    secrets: &InputSecrets,
) -> Result<LoginAttempt, Error> {
    let auth_hash = AuthPinHash::new(crypto, se, pin, secrets)?;
    // No random delays in the PIN verification path
    // Direct HMAC comparison without timing randomization

    match_count = se_read_counter(se, 0, &secrets)?;
    counter = se_read_counter(se, 1, &secrets)?;
    attempts_left = match_count - counter;
    // ...
}
```

KeyOS/os/security/src/platform/atsama5d2/se_port.rs:993-1042

Impact

A malicious attacker with physical access seeking to attack the PIN verification and counter increment code may have more predictable timing to perform such attacks. Because this is not directly exploitable, the impact of this finding is considered to be low.

Recommendation

Add random delays similar to bootloader's random_delay() implementation to obfuscate timing patterns during PIN verification.

Key Material Intact When No Attempts Left (Impact: Low)

Description

When PIN attempt counter reaches 0, the device continues operating normally with no automatic SECURAM erasure. Only returns an error but doesn't trigger security lockout.

```
// When PIN attempts are exhausted
attempts_left = match_count - counter;
if attempts_left == 0 {
    // No automatic SECURAM erasure or lockout
    // Device continues to operate normally
    // Only returns AttemptsTooManyAttempts error
    return Ok(LoginAttempt::TooManyAttempts);
}
```

KeyOS/os/security/src/platform/atsama5d2/se_port.rs:1026-1042

Impact

A malicious attacker with physical access to the device may be able to bypass this check with a simple single fault-injection attack. Because the feasibility of this attack depends on how the code is executed in the firmware binary and other external factors, the impact of this finding is considered to be low.

Recommendation

Clear the SECURAM contents and/or the cryptographic contents on the ATECC608 when the pin attempts are exhausted.

Bootloader Does Not Clear RAM on Boot (Impact: Low)

Description

Only BSS section is cleared during boot, not general RAM. Previous session data may persist in memory between reboots. All RAM should have a memzero or defined pattern applied at boot.

Impact

Cold boot attacks where an attacker with physical access could potentially recover sensitive data from RAM. On a battery powered device this is particularly critical. Because Passport encrypts all RAM, this finding is not directly exploitable, the impact of the finding is considered to be low.

Recommendations

Clear all RAM on boot, not just SECURAM and BSS sections. With a microkernel such as Xous, it may be difficult to predict all memory areas where sensitive data may be written. The safer approach is thus to clear all RAM.

Shamir Shares Not Zeroized After Use (Impact: Low)

Finding 4: Shamir Shares Not Zeroized After Use

Description

The Shard struct containing Shamir secret shares lacks the ZeroizeOnDrop trait. Shares remain in heap memory after keycard operations are completed.

```
pub struct Shard {
    #[cbor(n(0), with = "minicbor::bytes")]
    pub device_id: [u8; 32],
    #[cbor(n(1), with = "minicbor::bytes")]
    pub seed_fingerprint: [u8; 32],
    #[n(2)]
    pub seed_shamir_share: Vec<u8>,  // Missing ZeroizeOnDrop
    #[n(3)]
    pub seed_shamir_share_index: usize,
    #[n(4)]
    pub part_of_magic_backup: bool,
    #[cbor(n(5), with = "minicbor::bytes")]
    pub hmac: [u8; 32],
}
// No ZeroizeOnDrop implementation found
```

KeyOS/os/keycard/src/api.rs:69-83

Impact

Secret shares could be recovered from memory dumps if device is compromised while powered. Because this is not directly exploitable, the impact of this finding is considered to be low.

Recommendations

Implement ZeroizeOnDrop trait for the Shard struct to ensure automatic memory clearing when shares go out of scope.

Conclusion

The audit confirms that KeyOS implements security correctly at nearly every level. All 5 findings are low-severity issues that mostly require an attacker to have physical possession of the device, sophisticated equipment, and deep hardware expertise to exploit. The innovative use of SECURAM for key material storage, combined with the ATECC608C secure element and comprehensive tamper detection, creates multiple independent security barriers.

Passport Prime Security Audit Conclusion

Based on this comprehensive security audit of the Passport Prime, the overall architecture demonstrates exceptional security design principles and sophisticated implementation. The wallet effectively addresses key threats including malware protection, evil maid attacks, and supply chain vulnerabilities through its thoughtful hardware choices and advanced security features. The use of the SAMA5D2 microcontroller with its extensive security features alongside the ATECC608C secure element with monotonic counter functionality provides a robust foundation for secure operations. The architectural decisions, including the elegant seed protection scheme, NFC/Bluetooth wireless isolation for air-gapped transactions, and the innovative use of SECURAM for hardware-guaranteed secret erasure, demonstrate exceptional consideration of real-world attack scenarios.

The implementation showcases particularly strong security engineering in its multi-layer seed protection architecture, which cleverly distributes critical components across different security domains, ensuring that compromise of any single component cannot expose user funds. The sophisticated tamper response system, which combines hardware automatic clearing of SECURAM with secure element protections, provides defense-in-depth against physical attacks. While areas for enhancement were identified, such as leveraging even more of the SAMA5D2's security features and potential implementation of ARM TrustZone technology, these represent opportunities for future security hardening rather than critical vulnerabilities.

Most notably, all identified security findings from the audit have been addressed. The low-impact hardware finding regarding debug interface accessibility is adequately mitigated by the existing tamper protection system, with recommendations provided for complete elimination in future hardware revisions. The proactive approach to security, including the use of cargo-audit for dependency monitoring and the commitment to publishing an SBOM and public threat model, demonstrates Foundation's dedication to transparency and continuous security improvement. This results in a highly secure hardware wallet architecture that exceeds industry standards for protecting users' digital assets.