Passport Prime Security Audit Response

September 29, 2025

Executive Summary

This document presents Foundation Devices' response to the Passport Prime Security Audit which was performed by Keylabs.

The purpose of this document is two-fold:

1. Respond to each of the minor findings in the audit
2. Provide additional details on the threat scenarios outlined in the audit document, and describe how Passport Prime's security features work together to eliminate or mitigate these threats.

Findings Response

The Keylabs audit lists five specific findings, all low impact and not directly exploitable. The following sections describe the improvements we have made to address each of these issues.

1. Debug Interface and Test Pad Accessibility (Impact: Low)

Keylabs Finding

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.

Response

The JTAG debug interface is exposed on the circuit board so that Foundation developers can connect a debug board during development. While the connector for this debug board is not mounted for production, we completely disable the JTAG interface during device provisioning. In addition, as noted in the report, physical access to these test pads is only possible if an attacker can manage to remove the screen or drill through it. The tamper system would trigger if you removed the screen, and drilling through the screen would leave obvious damage, but still require the attacker to somehow engineer a successful glitch on both the MCU and the Secure Element.

2. PIN Check Timing Not Randomized (Impact: Low)

Keylabs Finding

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.

Response

Random delays have been added into the PIN check code as suggested by Keylabs.

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

Keylabs Finding

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.

Response

Earlier builds had the code clear and reboot intentionally removed to ease development. This code has been added in now to properly clear the SECURAM

and reboot Passport Prime when the final PIN entry attempt has been exhausted.

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

Keylabs Finding

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.

Response

All RAM in the device is scrambled with a random value which is different on each boot. The scrambling nonce is lost when power is lost or on reset. This is more than sufficient to ensure that an attacker cannot read any secrets from memory from the previous boot.

However, it was an easy addition to reset all memory to zero on boot, so it has been implemented.

5. Shamir Shares Not Zeroized After Use (Impact: Low)

Keylabs Finding

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

Response

This structure has now been marked as `ZeroizeOnDrop` so the memory is automatically cleared once the Shamir shares have been written to the Keycards.

Attack Vectors & Mitigations

Core Security Features

Firmware Security

- **2-of-4 Multisignature Firmware Protection**: All firmware must be signed by at least 2 of Foundation's 4 signing keys, preventing unauthorized firmware installation, single-point-of-failure compromises, and custom firmware bypasses for replay attacks
- **AES-CMAC Secure Boot**: SAMA5D28 processor uses encrypted First Stage Bootloader (FSBL) to verify KeyOS authenticity through hardware-guaranteed chain of trust
- **Anti-Downgrade Protection**: Latest Firmware Timestamp in Slot 9 prevents rollback to vulnerable versions; timestamp survives factory reset and is only updated by officially-signed firmware
- **Bootloader Error Display**: Clear error shown when non-Foundation firmware is detected; bootloader will not boot an unsigned OS image

Tamper Detection & Physical Security

- **Hardware Tamper Detection**: Two active tamper pins with normally-open switches detect device opening from either side and trigger immediate SECURAM clearing
- **Debug Interface Protection**: JTAG/SWD interfaces disabled during provisioning and physically located under display assembly, requiring tamper event for access
- **SECURAM Hardware Protection**: Hardware-guaranteed memory scrambling on physical intrusion/tamper or power loss; critical secrets permanently lost on tamper

Three-Factor Seed Protection

- **Split Architecture Security**: Seed protection requires three components: encrypted seed in ATECC608C, One-Time Pad in SECURAM, and PIN hash - no single component compromise exposes the seed
- **Seed Reconstruction Formula**: `Slot 10 = Seed Bytes ⊕ OTP ⊕ SHA256(PIN + "Encrypt")`
- **Hardware Secure Element**: ATECC608C provides tamper-resistant storage with locked configuration and keys never readable outside the element

Memory Protection

- **Runtime Memory Scrambling**: All RAM scrambled with per-boot unique nonce; scrambling key lost on power loss or reset
- **Boot Memory Clearing**: All system memory reset to zero on boot
- **ZeroizeOnDrop Implementation**: Security-sensitive Rust structures automatically clear secrets from RAM when no longer needed
- **Comprehensive Memory Clearing**: Non-sensitive memory areas zeroized when returned to OS

PIN & Authentication Security

- **Hardware Monotonic Counters**: ATECC608C Counter 0 tracks PIN attempts with tamper-resistant counting that cannot be reset by software
- **PIN Stretching**: Multi-component PIN hash using `SHA256(PIN + PIN_Stretch + PIN_Attempt)` with secure element operations
- **Enhanced PIN Options**: Optional alphanumeric PINs significantly expand brute force space
- **Randomized Timing**: Random delays inserted in PIN verification and other critical code paths to prevent timing attacks

Communication Security

- **QuantumLink Protocol**: AES256-encrypted communications using the quantum-resistant ML-KEM algorithm for key exchange for all Bluetooth transmissions with sequence numbers to prevent replay attacks
- **IO Protection Secret**: 32-byte hardware-derived secret (`SHA-256(fuse_entropy + extra_entropy + "IO Protection Secret")`) encrypts communications between SAMA5D2 and ATECC608C
- **Air-Gap Isolation**: Independent execution environment on dedicated microcontroller with no direct electrical connection to host systems (except when USB connected)
- **PCB Communication Encryption**: Board-level communication between main MCU and Bluetooth MCU is encrypted

Device Authentication & Supply Chain Protection

- **Security Check Process**: Device-unique private keys stored in the secure element enable comprehensive device attestation
- **Automatic Optical Inspection**: Every PCB checked against golden sample; discrepancies noted and boards pulled from production
- **Secure Provisioning**: Bootloader is pre-signed by Foundation before factory delivery; uses MCU SecureBoot feature; signing keys are never accessible at factory

User Interface Security

- **Security Words System**: Device and master seed combination provides unique authentication words allowing users to detect device swaps
- **High-Resolution Display**: Independent transaction verification with address/amount display separate from potentially compromised host systems
- **Transaction Validation**: Series of checks ensure amounts make sense (no huge fees) and change addresses belong to wallet
- **Direct PIN Entry**: PIN entry occurs only on device, never transmitted to or processed by host

---

1. Physical Access & Tampering Attacks

Threat Overview: What Could Go Wrong

Physical access to hardware wallets presents one of the most serious threat categories:

Evil Maid Scenarios
- Attackers briefly access an unattended device to swap hardware or reflash firmware
- Install monitoring equipment or hardware keyloggers to capture PINs
- Replace the device with a compromised version that appears identical to capture

PIN for the attacker to use on the original device
- Modify firmware to create backdoors for future access

Component-Level Attacks
- Use sophisticated equipment to probe debug interfaces and extract memory contents
- Perform fault injection attacks (EMFI, voltage glitching) to bypass security measures
- Side-channel analysis to extract cryptographic secrets through power or electromagnetic emissions
- Direct chip-level attacks requiring expensive equipment and expertise

Case Tampering
- Open the device to access internal components and PCB
- Replace or modify security-critical components like the secure element
- Install hardware implants or monitoring devices inside the case
- Bypass software-based security measures through direct hardware access

Supply Chain Insertion
- Modify devices during manufacturing before they reach legitimate distribution
- Compromise devices during shipping or storage in distribution centers
- Replace genuine components with malicious versions during assembly
- Install backdoors or monitoring capabilities at the factory level

Passport Prime Protections

Evil Maid Scenarios
- Hardware tamper detection with automatic SECURAM clearing
- Security words authentication enables device swap detection
- Factory reset function erases critical secrets after maximum login attempts (10)
- 2-of-4 multisignature firmware protection prevents unauthorized installations

Component-Level Attacks
- Debug interface protection under display assembly
- Keylabs fault injection testing (EMFI, voltage glitching) yielded no actionable results
- Split architecture requires simultaneous compromise of both SAMA5D2 and ATECC608C
- PIN hash encryption provides additional brute force protection even with dual

component compromise
- Passphrase support provides further brute force protection
- All known ATECC608 family attack vectors effectively mitigated

Case Tampering
- SECURAM hardware protection with memory scrambling on intrusion
- Three-factor seed protection ensures no single component compromise gives fund access
- One-Time Pad permanently lost on tamper, making seed retrieval impossible

Supply Chain Insertion
- Device authentication using device-unique private keys
- Security Check process for supply chain validation
- Physical PCB design with minimal component labeling and strategic placement
- Keylabs confirmation of tamper detection and firmware authentication systems

---

2. Supply Chain Security

Threat Overview: What Could Go Wrong

Supply chain attacks target the device before it reaches the user, potentially compromising security at the source:

Pre-loaded Malicious Seeds
- Attackers configure devices with known seed phrases during manufacturing
- Pre-programmed wallets appear ready-to-use but give attackers control over user funds
- Difficult for users to detect since devices function normally
- Can affect large numbers of users if inserted early in supply chain

Hardware Substitution
- Replace genuine components with counterfeit or modified versions during manufacturing
- Substitute secure elements with compromised versions that leak keys
- Use lower-grade components that are more vulnerable to attacks
- Install hardware backdoors in microcontrollers or other critical components

Firmware Tampering
- Install malicious firmware that appears legitimate but contains backdoors
- Modify bootloaders to bypass security checks
- Insert code that transmits keys or PINs to remote servers
- Create firmware that appears to function normally but compromises security over time

Software Dependencies
- Compromise upstream dependencies in the build process (like the xz backdoor incident)
- Insert malicious code into development tools or compilers
- Target continuous integration and deployment systems
- Compromise code signing infrastructure

Manufacturing Infiltration
- Gain unauthorized access to production facilities to modify devices
- Steal signing keys or other critical manufacturing secrets
- Insert additional steps in production that compromise device security
- Bribe or coerce manufacturing personnel to install backdoors

Passport Prime Protections

Pre-loaded Malicious Seeds
- 2-of-4 multisignature firmware protection prevents unauthorized installations
- Security Check process using device-unique private keys reports compromised devices
- Device authentication keys recorded in Foundation database with no customer tracking
- Automatic Security Check during onboarding verifies authenticity before first use

Hardware Substitution
- AES-CMAC secure boot ensures only Foundation-signed bootloaders are executed
- MCU without Foundation’s secret AES key cannot boot signed KeyOS bootloader
- Device traceability with monitoring at each production stage
- Detection of unauthorized device creation through Security Check verification

- Secure provisioning with locked secure element configuration
- Automatic optical inspection with golden sample comparison

Firmware Tampering
- 2-of-4 multisignature firmware protection prevents rogue releases
- Multisignature verification provides resilience against individual key compromise
- Bootloader error display for non-Foundation firmware with boot prevention

Software Dependencies
- KeyOS uses cargo-audit for Rust dependency monitoring
- Software Bill of Materials (SBOM) to be published for transparency
- Keylabs validation of supply chain security measures
- AI code reviews in addition to human reviews

Manufacturing Infiltration
- Comprehensive device authentication from factory to user
- Secure provisioning with locked configurations
- Keylabs confirmation of robust device attestation system
- Signing keys never accessible at factory

---

3. Remote Attacks & Malware Protection

Threat Overview: What Could Go Wrong

Remote attacks target the host devices (computers, phones) that interact with hardware wallets:

Host Compromise
- Malware on connected devices attempts to steal PINs during entry
- Malicious software redirects transactions to attacker-controlled addresses
- Keyloggers capture PIN entry if performed on host device
- Screen capture malware records transaction details and authentication

Transaction Manipulation
- Malicious software modifies transaction details displayed to users
- Address substitution attacks replace legitimate addresses with attacker addresses

- Amount manipulation to redirect portions of transactions
- Smart contract interaction tampering to execute unintended operations

PIN Harvesting
- Keyloggers capture PIN entry on compromised host systems
- Screen recording malware captures PIN entry through visual interfaces
- Clipboard monitoring to intercept copied seed phrases or addresses
- Social engineering attacks to trick users into entering PINs on compromised systems

Communication Interception
- Man-in-the-middle attacks on data transmitted between wallet and host
- Network-based interception of wireless communications
- USB communication tampering through compromised cables or hubs
- Protocol-level attacks on communication channels

Filesystem Attacks
- Malicious files uploaded to wallet storage exploit parsing vulnerabilities
- Buffer overflow attacks through crafted file uploads
- Directory traversal attacks to access system files
- Malware disguised as legitimate wallet data files

Passport Prime Protections

Host Compromise
- Air-gap isolation on dedicated microcontroller
- Airlock mode limits host access to user-chosen files when USB is connected
- Keylabs validation of air-gap isolation architecture
- Direct PIN entry on device only, never on host

Transaction Manipulation
- High-resolution display for independent transaction verification
- Transaction validation checks for reasonable amounts and change address ownership
- User confirmation required before transaction signing

PIN Harvesting
- Direct PIN entry on device (never sent outside the device)
- Security words provide PIN verification feedback without host exposure
- Hardware-based PIN processing within secure element

Communication Interception
- QuantumLink protocol encryption prevents data interception by malware and over-the-air (OTA) sniffing
- Encryption keys are stored with AES256 encryption
- Bluetooth chip isolation with pre-encrypted communications preventing decode/modification on all sent and received messages

Filesystem Attacks
- Careful filesystem implementation with input sanitization
- File length and encoding validation preventing buffer overflow
- App isolation preventing private filesystem access between apps
- Encrypted app private data never exposed over USB
- Keylabs assessment found no adverse findings in eMMC storage

---

4. Authentication & PIN Security

Threat Overview: What Could Go Wrong

Hardware wallets face unique authentication challenges due to limited computational resources:

Brute Force Attacks
- Attackers with powerful hardware attempt to try PIN combinations faster than the device allows
- Offline attacks against extracted authentication data
- Dictionary attacks using common PIN patterns
- Parallel processing to accelerate PIN guessing

PIN Attempt Bypass
- Vulnerabilities that allow unlimited PIN attempts without proper rate limiting
- Hardware glitching to reset attempt counters
- Firmware exploits that bypass attempt tracking
- Physical attacks to disable attempt limiting mechanisms

Side-Channel Analysis
- Power analysis attacks to extract PIN information during verification
- Electromagnetic analysis to observe cryptographic operations

- Timing attacks based on PIN verification duration
- Acoustic analysis of device operation during authentication

Authentication State Manipulation
- Attacks that bypass PIN requirements entirely
- Session hijacking to maintain authenticated state
- Privilege escalation after partial authentication
- State confusion attacks in multi-factor authentication

Recovery Vulnerabilities
- Weaknesses in PIN reset or device recovery processes
- Social engineering attacks against recovery mechanisms
- Backup system compromises that bypass authentication
- Recovery seed phrase attacks

Passport Prime Protections

Brute Force Attacks
- Hardware monotonic counters provide tamper-resistant login attempt tracking
- Match Count comparison ensures maximum attempts cannot be exceeded
- Hardware-guaranteed lockout when all 10 attempts are exhausted (SECURAM is scrambled)

PIN Attempt Bypass
- Hardware monotonic counter implementation validated through testing
- PIN processing security confirmed within Secure Element using HMAC operations
- PIN stretching with additional entropy increases brute force difficulty

Side-Channel Analysis
- Hardware AES, SHA, and TRNG operations provide inherent side-channel resistance
- Secure element design hardened against power analysis attacks
- Randomized timing delays in PIN verification and critical code paths
- Side-channel resistance verified through hardware testing

Authentication State Manipulation
- Enhanced PIN options with optional alphanumeric PINs significantly expanding brute force space
- Security Words System for PIN verification feedback

- Enhanced display after 4+ characters with user-controlled verification granularity
- Backspace prevention after minimum length slows enumeration attacks

Recovery Vulnerabilities
- Keycard backup system provides “master unlock” capability
- Seed phrase recovery maintains security while preventing device bricking
- Factory reset counter prevents confusion about device state
- Clear user feedback about device authentication status
- Recovery mechanisms tested and confirmed functional

---

5. Communication Security

Threat Overview: What Could Go Wrong

Hardware wallets must securely communicate with host devices (mobile phones and personal computers) and between internal components:

Wireless Interception
- NFC and Bluetooth communications monitored by nearby attackers
- Passive eavesdropping on wireless transmissions
- Signal analysis to extract transmitted data
- Replay attacks using captured wireless communications

Man-in-the-Middle Attacks
- Attackers position themselves between wallet and host to intercept data
- Rogue access points or Bluetooth devices that impersonate legitimate hosts
- Protocol downgrade attacks to force less secure communication modes
- Certificate or key substitution attacks

Replay Attacks
- Capturing and replaying previous communication sessions
- Transaction replay to duplicate previous operations
- Authentication replay to gain unauthorized access
- Session token reuse attacks

Internal Bus Monitoring
- Physical access to communications between ICs within the device

- Probing of SPI, I2C, or other internal communication buses
- Logic analyzer attacks on internal data paths
- Side-channel analysis of internal communications

Protocol Weaknesses
- Vulnerabilities in communication protocols allowing data extraction
- Weak encryption or authentication in wireless protocols
- Implementation flaws in standard communication protocols
- Missing integrity checks that allow data manipulation

Passport Prime Protections

Wireless Interception
- QuantumLink protocol encryption prevents eavesdropping on wireless transmissions
- Secure pairing and authentication protocols for wireless connections using out-of-band communication (e.g., QR codes)

Man-in-the-Middle Attacks
- IO Protection Secret encrypts MCU-to-secure-element communications
- Single Wire Interface (SWI) communications with cryptographic protection
- IO Protection Secret implementation confirmed through code analysis
- Anti-downgrade protection prevents installation of older firmware versions

Replay Attacks
- QuantumLink random request IDs plus a short sliding window time-based message validation provide resistance against replay of old messages

Internal Bus Monitoring
- Careful attention to Over-the-Air (OTA) and board-level data paths
- Hardware tamper response detects and responds to physical communication interception
- PCB communication encryption between main MCU and Bluetooth MCU

Protocol Weaknesses
- Avoidance of unencrypted serial communication in Bluetooth implementations
- Strong AES256 encryption for all wireless data transmission
- Bluetooth Low Energy implementation with comprehensive security considerations

6. Firmware & Boot Security

Threat Overview: What Could Go Wrong

Firmware represents a critical attack surface for hardware wallets:

Malicious Firmware Installation
- Loading unauthorized firmware that appears legitimate but contains backdoors
- Social engineering users to install compromised firmware updates
- Supply chain attacks that distribute malicious firmware
- Exploiting firmware update mechanisms to install unauthorized code

Bootloader Vulnerabilities
- Exploiting the boot process to gain persistent access to the device
- Bootloader bypass techniques that skip security checks
- ROM-level vulnerabilities that cannot be patched through field updates
- Secure boot bypass through hardware or software means

Downgrade Attacks
- Rolling back to older firmware versions with known vulnerabilities
- Exploiting version checking mechanisms to allow downgrades
- Using legitimate but vulnerable firmware versions
- Bypassing anti-rollback protections

Boot Process Manipulation
- Interfering with secure boot to load malicious code
- Fault injection during boot to bypass security checks
- Boot-time race conditions that allow code injection
- Manipulating boot configuration to disable security features

Firmware Update Attacks
- Compromising the update mechanism to distribute malicious firmware
- Man-in-the-middle attacks on firmware update channels
- Signature verification bypass in update mechanisms
- Update channel hijacking to distribute unauthorized firmware

Memory Persistence
- Sensitive data remaining in memory between boots or after power loss

- Cold boot attacks to extract keys from RAM
- Memory remanence attacks on various memory types
- Insufficient memory clearing allowing data recovery

Passport Prime Protections

Malicious Firmware Installation

- AES-CMAC secure boot verification and 2-of-4 multisignature firmware protection establish chain of trust from hardware ROM to main firmware
- Users cannot install improperly signed code
- KeyOS open source with reproducible builds allowing independent verification

Bootloader Vulnerabilities

- Secure boot implementation validated through firmware analysis
- Multisignature verification system confirmed operational
- Clear bootloader error display for non-Foundation firmware with boot prevention

Downgrade Attacks

- Anti-downgrade protection with Latest Firmware Timestamp surviving factory reset
- Only officially-signed firmware updates modify timestamp value
- User builds and betas do not update timestamp, maintaining protection

Boot Process Manipulation

- Anti-downgrade protection mechanisms tested and verified
- Hardware acceleration for cryptographic verification operations

Firmware Update Attacks

- AES-CMAC secure boot verification and 2-of-4 multisignature firmware protection establish chain of trust from hardware ROM to main firmware
- Users cannot install improperly signed code

Memory Persistence

- Boot memory clearing, runtime memory scrambling, and ZeroizeOnDrop implementation
- Hardware-guaranteed SECURAM scrambling on tamper events and RAM scrambling on reset

7. Key Management & Storage Security

Threat Overview: What Could Go Wrong

One main purpose of hardware wallets is secure key storage, making this the highest-value target:

Seed Extraction
- Direct recovery of the master seed that controls all cryptocurrency funds
- Memory dump attacks to extract seed material from RAM or storage
- Side-channel attacks to extract seed during cryptographic operations
- Physical attacks to extract seed from secure storage

Key Material in Memory
- Sensitive cryptographic material remaining accessible in RAM
- Key material exposure during processing or computation
- Memory dumps revealing temporary key storage
- Insufficient memory clearing leaving key remnants

Weak Encryption
- Insufficient protection of stored seed material
- Weak encryption algorithms or implementations
- Poor key derivation that allows seed recovery
- Inadequate randomness in encryption key generation

Single Point of Failure
- All security depending on one component that could be compromised
- Monolithic security architecture without defense in depth
- Single secure element failure exposing all secrets
- Lack of redundancy in critical security components

Memory Dumps
- Extracting key material from device memory through various attack methods
- Cold boot attacks on volatile memory
- DMA attacks through external interfaces
- Memory remanence attacks on non-volatile storage

Shamir Share Exposure
- Backup key shares remaining in memory after use
- Insufficient clearing of temporary share data

- Side-channel leakage during share processing
- Share reconstruction attacks when multiple shares are processed

Passport Prime Protections

Seed Extraction

- Three-factor seed protection with split architecture requiring simultaneous compromise of encrypted seed (ATECC608C), One-Time Pad (SECURAM), and PIN hash
- The seed does not exist anywhere in device; components must be recombined at runtime
- Architecture validated through testing

Key Material in Memory

- SECURAM hardware protection with automatic clearing on tamper detection
- Memory protection systems including boot clearing, runtime scrambling, and ZeroizeOnDrop
- One-Time Pad permanently lost on tamper, making encrypted seed indecipherable

Strong Encryption

- Hardware secure element with tamper-resistant key storage and locked configuration
- NIST-compliant randomness from three independent entropy sources (SAMA5D28, ATECC608C, custom Avalanche noise)
- Private keys never readable outside secure element
- Value in Secure Element requires combination with OTP and PIN hash for seed recovery

No Single Point of Failure

- Split architecture prevents single component compromise
- Hardware secure element integration confirmed operational
- Three independent entropy sources provide resilience against individual source compromise

Memory Dumps Attack

- Runtime memory scrambling with per-boot unique encryption key makes memory dumps ineffective
- Keylabs fault injection attempts yielded no actionable results

Shamir Share Exposure

- `ZeroizeOnDrop` implementation for automatic share clearing after use
- Comprehensive memory protection prevents cold boot attacks
- All major attack vectors assessed and found effectively mitigated

---