MachXO4 Root of Trust Security Checklist

Technical Note

FPGA-TN-02435-1.0

September 2026

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

Contents

Tables

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
AES	Advanced Encryption Standard
CRE	Cryptographic Engine
ECDSA	Elliptic Curve Digital Signature Algorithm
EFB	Embedded Function Block
GUI	Graphic User Interface
HSM	Hardware Security Module
JEDEC	Joint Electron Device Engineering Council
NDA	Non-disclosure Agreement
OTP	One-Time Programmable
SPI	Serial Peripheral Interface
UFM	User Flash Memory

1. Introduction

Security is a rising concern with new systems. Lattice MachXO4™ Root of Trust (RoT) devices feature cryptographic functions that help secure the systems. This technical note summarizes all the security-related features in MachXO4 RoT devices, which cover the initial Lattice Radiant™ software setup for security functions, critical security requirements related to MachXO4 RoT devices during the provisioning/programming stage (bitstream security, lock policy for ports, function, and so on) to user mode security functions managed by the Cryptographic Engine (CRE). This technical note does not provide detailed step-by-step instructions but offers a high-level summary of security functions.

This technical note assumes that the reader is familiar with the MachXO4 RoT device features as described in the MachXO4 Family Root-of-Trust Devices Data Sheet (FPGA-DS-02126).

The critical security areas covered in this technical note include:

- Enabling MachXO4 RoT device security settings in Lattice Radiant software
- Enabling bitstream security settings in the Radiant Bitstream Security Settings tool
- Security key programming for bitstream authentication and encryption in Lattice Radiant Programmer
- Bitstream authentication/signing with Hardware Security Module (HSM)-generated signature
- MachXO4 RoT device lock policy for ports and assets
- User mode security functions covered by CRE
- Tamper Detection feature in MachXO4 RoT device.

2. Bitstream Security Settings

You can enable flash protection passwords for read/write lock access, authentication, and encryption in the configuration bitstream. You can access all the MachXO4 RoT device bitstream security options using the Bitstream Security Settings Graphic User Interface (GUI) in the Lattice Radiant software.

The Flash Protection Password feature works as follows:

- MachXO4 RoT devices offer 128-bit password protection for device configuration access, which you can optionally enable.
- The password security feature uses a Flash Protect Key to control access to the configuration and programming modes of the device.
- Write, Verify, and Erase operations are allowed only when presented with a Flash Protect Key that matches the key stored in the device.

Table 2.1. Authentication and Encryption on the Bitstream

Security Protocol	Encryption (AES-256)	Authentication (ECDSA)	Bitstream Format
AES Encryption	Yes, AES Key	No	Encrypted Bitstream
Authentication (ECDSA)	No	Yes, ECDSA private and public key	Plain bitstream + Signature generated using ECDSA private key
Authentication (ECDSA) and AES Encryption	Yes, AES Key	Yes, ECDSA private and public key	AES Encrypted data (Plain bitstream + Signature generated using ECDSA private key)

- Refer to the MachXO4 RoT Cryptographic Engine (FPGA-TN-02396) - Bitstream Security section for more details on how to enable security settings on bitstream.
- Refer to the Device Bring Up Examples section of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) for use case examples.

3. Bitstream Authentication

MachXO4 RoT devices support Elliptic Curve Digital Signature Algorithm (ECDSA)-based authentication. You can program the public key into the device through Lattice Radiant Programmer to enable authentication for prototyping purposes. For details on how to program the public key, refer to the Device Bring Up Examples section in the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).

For a production environment, use an HSM as a secure application for authentication. You can use OpenSSL to generate an ECDSA key pair and signature. Lattice Semiconductor provides two Python scripts to extract the digest and replace the signature in the bitstream, with the flow demonstrated below:

1. Generate a dummy signed FPGA bitstream using the Bitstream Security Settings tool in Lattice Radiant software.
   - Use the Lattice Radiant software to generate a signed FPGA bitstream (JEDEC file). The signed bitstream has a signature embedded in it. The signature in this setup is generated using an auto key pair using the Bitstream Security Settings tool as a placeholder for the signature in the JEDEC file.
2. Get the keys ready.
   - For secure applications, generate a key pair from an HSM.
3. Source Lattice-provided reference Python scripts.
   - There are two Python scripts:
     a. One is used to extract the configuration data from the bitstream, which is used for generating a signature.
     b. The other one is used to replace the existing signature with the actual signature generated from an HSM.
4. Program the Public key and signed bitstream (JEDEC file) into the MachXO4 RoT FPGA.

For details on how to enable the HSM signing flow, refer to Signing JEDEC with HSM-Generated Signature (FPGA-TN-02260).

In addition, the public key sector (PUBKEY) can be secured using One-Time Programmable (OTP) to avoid any further updates to this public key. The PUBKEY sector has three lock policies: Erase Lock, Write/Program Lock, and Read Lock.

Table 3.1 summarizes the PUBKEY memory access policy settings available for you to choose:

Table 3.1. PUBKEY Memory Access Policy Settings

Radiant Programmer Option	Description
Read Lock	When enabled, the read access to PUBKEY is locked (soft-locked). This is an OTP operation, in which PUBKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together with *Lock Wishbone Access (Hard Lock)* option.
Write Lock	When enabled, the write/program access to PUBKEY is locked (soft-locked). This is an OTP operation, in which PUBKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together *with Lock Wishbone Access (Hard Lock)* option.
Erase Lock	When enabled, the erase access to PUBKEY is locked (soft-locked). This is an OTP operation, in which PUBKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together with *Lock Wishbone Access (Hard Lock)* option.
OTP	Turning on this option is similar to enabling Write Lock and Erase Lock.
Lock Wishbone Access (Hard Lock)	Turn on this option if you want to block Wishbone access for any unlock operation. This option needs to be turned on together with one or more options from Read/Write/Erase lock.

- For details on how to enable OTP settings, refer to the Device Feature and Security Programming sections of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).
- For details on how to unlock memory access through Wishbone, refer to the Lock Altering Through WISHBONE Interface section of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).

4. Bitstream Encryption

MachXO4 RoT devices support AES encryption. The AES key can be programmed into the device through Lattice Radiant Programmer to enable encryption. For details on how to program the AES key, refer to the Device Bring Up Examples – Program the AES Encryption Key section of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).

In addition, the AES key sector (AESKEY) can be secured with OTP to avoid any further updates to this AES key. The AESKEY sector has three lock policies: Erase Lock, Write/Program Lock, and Read Lock. Table 4.1 summarizes the AESKEY memory access policy settings available for you to choose:

Table 4.1. AESKEY Memory Access Policy Settings

Radiant Programmer Option	Description
Read Lock	When enabled, the read access to AESKEY is locked (soft-locked). This is an OTP operation, in which AESKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together with *Lock Wishbone Access (Hard Lock)* option.
Write Lock	When enabled, the write/program access to AESKEY is locked (soft-locked). This is an OTP operation, in which AESKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together with *Lock Wishbone Access (Hard Lock)* option.
Erase Lock	When enabled, the erase access to AESKEY is locked (soft-locked). This is an OTP operation, in which AESKEY is always locked unless you unlock it through Wishbone access. If Hard Lock (disable Wishbone access) is required, turn on this option together with *Lock Wishbone Access (Hard Lock)* option.
OTP	Turning on this option is similar to enabling Write Lock and Erase Lock.
Lock Wishbone Access (Hard Lock)	Turn on this option if you want to block Wishbone access for any unlock operation. This option needs to be turned on together with one or more options from Read/Write/Erase lock.

- For details on how to enable OTP settings, refer to the Device Feature and Security Programming sections of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).
- For details on how to unlock memory access through Wishbone, refer to the Lock Altering Through WISHBONE Interface section of the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).

Optionally, you can enable clock randomization and background noise when you program the AES key to mitigate side-channel attacks. In addition, you can choose to enable *Encrypted Bitstream Only* mode if an unencrypted bitstream is prohibited from loading into the device.

5. Protecting Data in Assets

Assets are the different flash sectors in MachXO4 RoT devices. Besides the PUBKEY and AESKEY sectors discussed earlier, there are ten more flash sectors, for a total of twelve flash sectors shown in Table 5.1. The same lock policies (read, write, and erase lock) apply to all the flash sectors.

Each flash sector has its own local lock and central lock. In the local locking scheme, you can enable or disable memory access (program and read operations) to all flash sectors by selecting the appropriate flash sector and enabling the local security bit for that sector. On the other hand, in the central locking scheme, the centralized security settings flash sector controls the following by setting appropriate bits: the memory accesses (erase and read specifically) to the CFG, UFM, Feature, and Security Keys flash sectors.

Table 5.1. Memory Access and Central Bits Location of Flash Sectors

Flash Sector/SRAM	Memory Access			Central Bits Location
	Erase	Read	Program	
CFG/Feature/Security Keys/SRAM				
CFG0	Central	Central/Local	Local	CSEC
CFG1	Central	Central/Local	Local	
FEA	Central	Central/Local	Local	
PUBKEY	Central	Central/Local	Local	
AESKEY	Central	Central/Local	Local	
SRAM	Central	Central/Local	Local	
UFM				
UFM0	Central	Central/Local	Local	USEC
UFM1	Central	Central/Local	Local	
UFM2	Central	Central/Local	Local	
UFM3	Central	Central/Local	Local	
Central/User Security Settings				
CSEC	Local	Local	Local	—
USEC	Local	Local	Local	—

Refer to the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) - Memory Access section for available settings to enable/disable locks.

The locking feature adds a layer of protection on top of the existing authentication and encryption features. OTP is recommended if User Flash Memory (UFM) contains critical data.

5.1. Updating Data in Assets

Use soft locks to allow data updates in the future.

- To modify soft locks, use the internal Wishbone interface.

Note: Critical data, such as all keys (public key, encryption key), should be hard-locked and OTP-protected to prevent further updates.

Refer to the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) - Fixed Lock versus Dynamic Lock and Lock Policy Settings and Lock Bits Description sections for details about the various locking options available in MachXO4 RoT devices.

6. Port Locking

The port access controls access to the external configuration ports of MachXO4 RoT devices. These configuration ports can be either partially or totally locked. To provide finer control of instructions/commands that are used to access the MachXO4 RoT device, port locking can be further expanded into four lock modes. Refer to the Lock Policy and Access Types section in the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397).

Table 6.1 summarizes the locking mechanism for all five types of configuration ports in MachXO4 RoT devices.

Table 6.1. Locking Mechanism of Configuration Ports

Port Access	Hard Lock	Locking Mechanism	
		Partial Lock	Total Lock
JTAG	Supported	Supported	Supported
Target SPI	Supported	Supported	Supported
Target I2C	Supported	Supported	Supported
JTAG to SPI Bridge	Supported	—	Supported
I2C Bridge	Supported	—	Supported

- Refer to the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) - Port Access section for details about the port locking mechanism in MachXO4 RoT devices.
- Refer to the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) - CFG Port Lock Option Programming section for details about the port locking OTP programming guide.

7. User Mode Security Function

To establish Root of Trust or better security control on the device during user mode, the hardened control functions in the Embedded Function Block (EFB) can be used in the user design to perform read/write on internal flash sectors. The Tamper Detection feature can be enabled by configuring the EFB module to enable Tamper Detection/Response. Once enabled, this feature continuously monitors the activity on external configuration ports as well as memory access (flash sector and SRAM).

Apart from EFB, as per the soft-lock mechanism, ensure that an unlock/lock algorithm is implemented in the core fabric design if soft-lock is needed for port locking or memory access lock.

The MachXO4 RoT device offers a suite of enhanced security algorithms and features, including 128-bit/256-bit Advanced Encryption Standard (AES-128/AES-256), 256-bit Secure Hash Algorithm (SHA256), and Elliptic Curve Digital Signature Algorithm (ECDSA), which are essential for security-sensitive applications. These advanced security functions are implemented using the CRE integrated into the MachXO4 RoT device. The CRE incorporates multiple security blocks used for authentication and configuration functions. To use these advanced security functions, you can incorporate the CRE IP into your design.

- Refer to the following for details about EFB:
  - MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404)
  - MachXO4 Hardened Control Functions User Guide (FPGA-TN-02403)
- Refer to the MachXO4 Root of Trust Tamper Detection/Response (FPGA-TN-02434) for details about Tamper Detection on MachXO4 RoT devices.
- Refer to the MachXO4 RoT Cryptographic Engine (FPGA-TN-02396) for details about all the security features supported by MachXO4 RoT CRE.

8. Device Bring-up Use Case Examples

There are five different MachXO4 RoT device use cases describing how to enable dual boot, authentication, encryption, and different locking options. Refer to the Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397) - General Device Setup Cases section for details. These are common use cases and there can be many different combinations and permutations based on specific implementation.

9. Licenses

Enabling bitstream security and user mode security features in MachXO4 RoT devices requires the following license feature lines:

- To enable bitstream security (authentication and encryption) - LSC_SW_XO4D_SECURITY_ENCRYPT
- To enable Cryptographic Engine (CRE) user primitive - LSC_SW_XO4D_HARDIP_CRE
- To enable MachXO4 RoT device locking features - LSC_SW_XO4D_SECURITY_LOCK

10. Checklist

The following table lists the MachXO4 RoT security checklist items.

Table 10.1. MachXO4 RoT Security Checklist Items

Item	Security Checklist Items	Status	N/A
1	Prerequisite		
1.1	Lattice Radiant software		
1.2	Encryption Pack		
1.3	License		
2	Configuration Security		
2.1	Bitstream Security Settings		
2.2	Keys Provisioning/Programming – Flash Protection Key, Public key, AES key		
2.3	Memory Access Lock Policy		
2.4	Port Locking		
3	User Mode Security		
3.1	EFB and Tamper Detection		
3.2	Lock/Unlock Algorithm for Soft-Lock policy		
3.3	Security Function by CRE		

References

- MachXO4 Family Root-of-Trust Devices Data Sheet (FPGA-DS-02126)
- MachXO4 Programming and Configuration User Guide (FPGA-TN-02393)
- MachXO4 Hardened Control Functions Reference Guide (FPGA-TN-02404)
- MachXO4 Hardened Control Functions User Guide (FPGA-TN-02403)
- MachXO4 Root of Trust Tamper Detection/Response (FPGA-TN-02434)
- Lock Policy Settings for MachXO4 RoT Devices (FPGA-TN-02397)*
- MachXO4 RoT Cryptographic Engine (FPGA-TN-02396)*
- Signing JEDEC with HSM-Generated Signature (FPGA-TN-02260)*
- MachXO4 web page
- Lattice Radiant Software web page
- Lattice Solutions IP Cores web page
- Lattice Solutions Reference Designs web page
- Lattice Insights for Lattice Semiconductor training courses and learning plans

***Note:** Access to this technical note requires a Non-disclosure Agreement (NDA). Contact Lattice Semiconductor.

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/en/Support/AnswerDatabase.

Revision History

Revision 1.0, September 2026

Section	Change Summary
All	Initial release.

www.latticesemi.com