Whitepaper

Ledger Recovery Key White paper

1 - Introduction

Welcome to this technical white paper, which is presenting an overview and deep-dive into the Ledger Recovery Key product, its architectural design, and its security.

We will present the way the product is built, its underlying technology and its functionalities, as well as the cryptographic protocols used when it comes to communicating with it. Our goal is to provide you with all the necessary information allowing anyone interested to see the measures we have taken to allow Ledger users to continue doing with Ledger Recovery Key what they used to do with our Hardware Wallets: own and manage their secret data in a secure and self-custodial way.

The technology described in this white paper is the subject of one or more pending patent applications. The publication of this white paper does not grant, either expressly or impliedly, any license, right, or permission to make, use, sell, or otherwise distribute the described technology.

2 - Keywords tables

2.1 - General

Keyword	Meaning
HSM	Hardware Security Module
NFC	Near Field Communication
PIN	Personal Identification Number
SE	Secure Element

2.2 - Keys

Keyword	Meaning
cc	Card Challenge
ccr	Card Cryptogram
cpk	Card Public Key
csc	Card Static Certificate
csc_enc	Encrypted Card Static Certificate

cec	Card Ephemeral Certificate
cec_enc	Encrypted Card Ephemeral Certificate
hc	Host Challenge
hcr	Host Cryptogram
hc_enc	Encrypted Host Challenge
hpk	Host Public Key
hsc	Host Static Certificate
hsc_enc	Encrypted Host Static Certificate
ipk	Issuer Public Key
key-SMAC	MAC Session Key
key-SENC	ENC Session Key

2.3 - Algorithms

Keyword	Meaning
AES-CBC	AES symmetric encryption algorithm used in CBC mode
AES-CMAC	AES-CMAC algorithm
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
HMAC	Keyed-Hash Message Authentication Code
SCP03	Secure Channel Protocol ‘03’
SECP256K1	Recommended Elliptic Curves Domain Parameters
SHA256 and SHA512	Secure Hash Standard

3 - Ledger Recovery Key - From a user perspective

Ledger Recovery Key is a smart card that can be used to store a copy of your master secret (from which stems your Secret Recovery Phrase) from your Ledger Hardware Wallet, provided that this Hardware Wallet supports NFC.
The Ledger Recovery Key product can only use the NFC protocol to communicate with a Ledger Hardware Wallet.

> ✒️Note
>
> Ledger Recovery Key is not a Hardware Wallet, since it is not possible to use it to manage assets nor sign transactions.

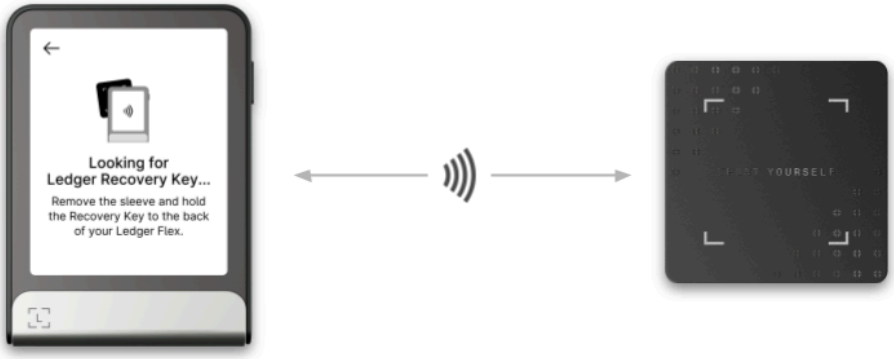
This product's security is guaranteed by several pieces of design, within which we are going to deep dive in the next sections:

- Security at rest
    - The product embeds a Secure Element chip, the same type of hardware security Ledger is also using within its Ledger Hardware Wallets, powered by a certified operating system,
- Security at use
    - The Ledger Recovery Key and Ledger Hardware Wallet perform a mutual authentication allowing the Ledger Recovery Key to ensure the Ledger Hardware Wallet is genuine, and allowing the Ledger Hardware Wallet to ensure the Ledger Recovery Key product is also genuine,
    - The product is PIN-protected, and three wrong PIN verification attempts will trigger the product to wipe its memory,
    - The communication protocol is secured and among others ensures the confidential and authenticated transfer of secret data,
- Secure display
    - All the interactions with the Ledger Recovery Key are triggered and confirmed on the secure screen of the Ledger Hardware Wallet,
    - The Ledger Recovery Key Identifier is checked by the Ledger OS to ensure consistency between the presented product in the use cases requiring two taps, and always displayed on the Ledger Hardware Wallets' screen so that the user can also make sure that the intended Ledger Recovery Key product is the correct one,
    - The Ledger Hardware Wallet will display an information screen indicating whether the presented Ledger Recovery Key matches the Secret Recovery

Phrase currently stored on the device. This helps users verify consistency, especially if they manage multiple Ledger Recovery Keys.

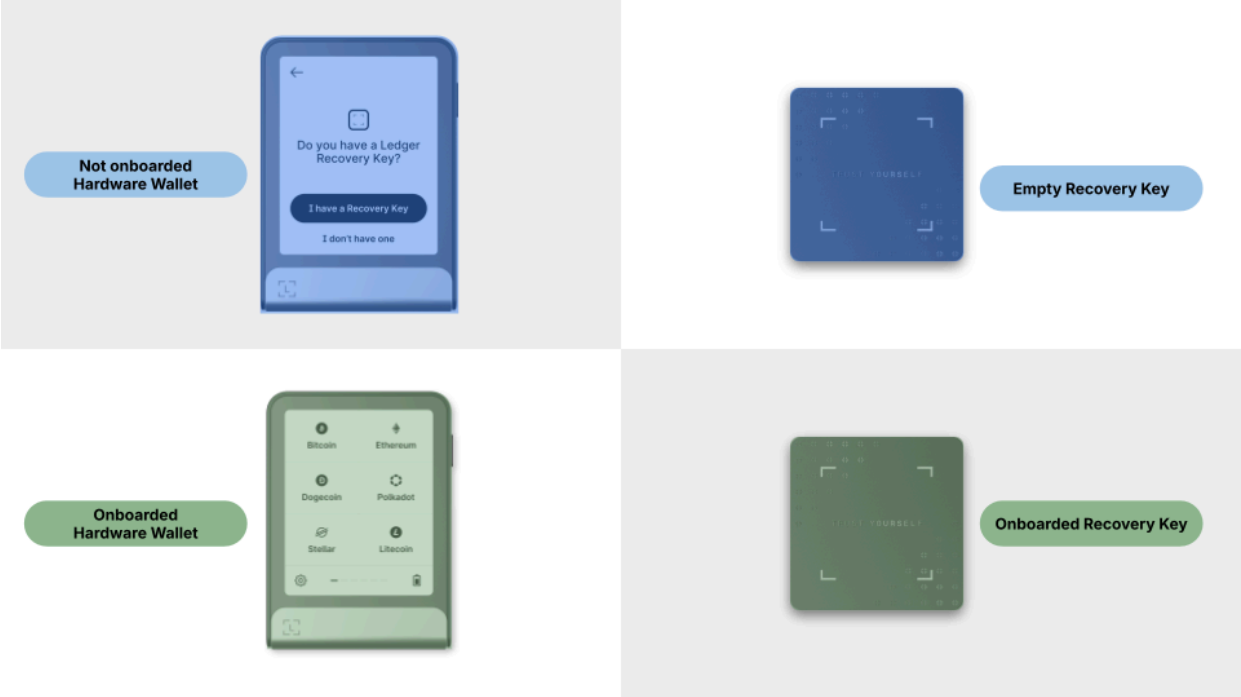
This product can exclusively be used with a Ledger Hardware Wallet that supports the NFC communication protocol. In this case, the trusted display of Ledger Hardware Wallets acts as the trusted display used to manage the Ledger Recovery Key product as well, and its use can be broken down into three main parts:

- Backing up the secret from an onboarded Ledger Hardware Wallet to an empty Ledger Recovery Key,
- Restoring the secret from an onboarded Ledger Recovery Key to a not-yet-onboarded Ledger Hardware Wallet,
- Manage the Ledger Recovery Key's contents (PIN, name for instance) from the Ledger Hardware Wallet.



The two products communicate over NFC

For the rest of the paragraph, the color legend is represented on the next picture. When no color is explicitly shown, the onboarded state of either the Ledger Hardware Wallet or the Ledger Recovery Key products does not change throughout the described operations.

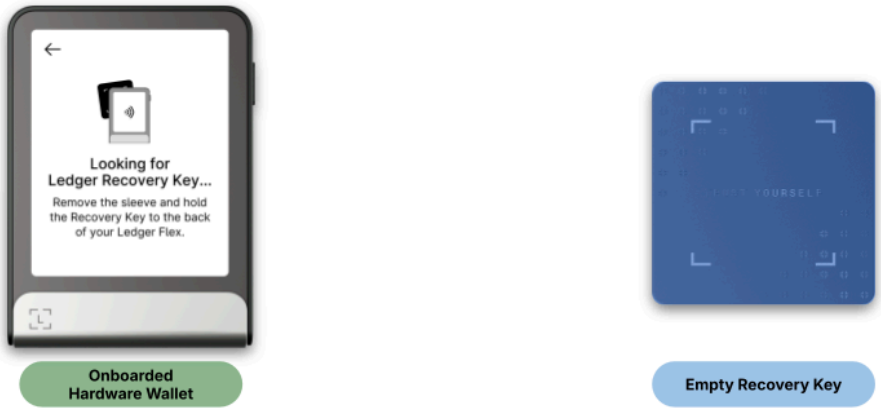


The onboarding states of the two Ledger products

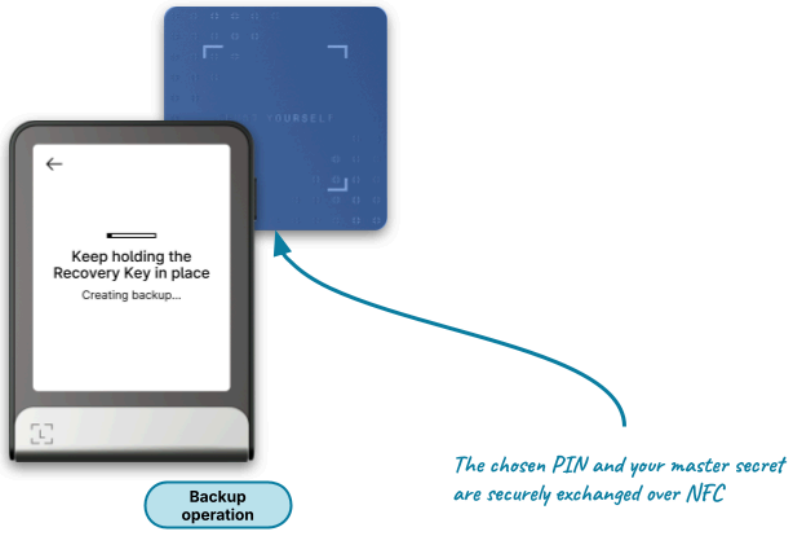
The next sections of this paragraph will describe the main features of the Ledger Recovery Key product from a user experience perspective, while the following paragraphs will focus on the technology perspectives.

3.1 - Feature - Back up seed to a Recovery Key

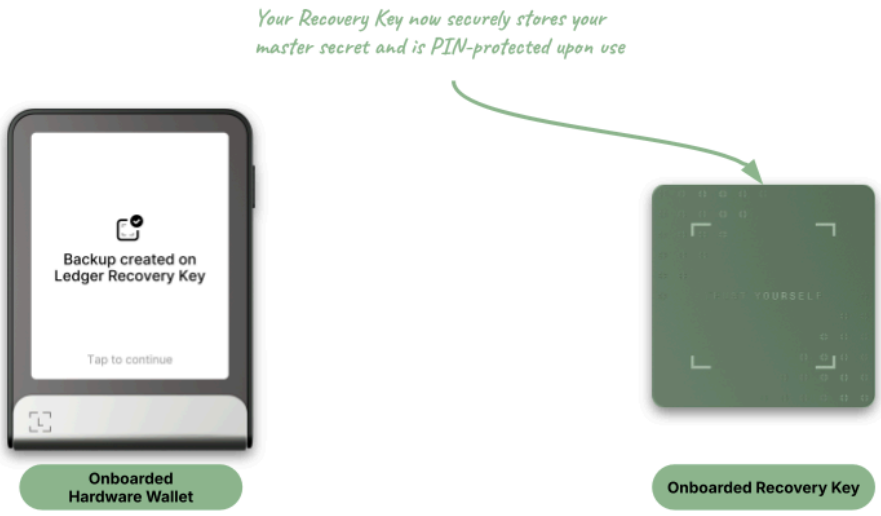
The first essential feature consists in giving the possibility to the user to transfer their master secret from their Ledger Hardware Wallet to their Ledger Recovery Key. To onboard a Ledger Recovery Key, the user shall follow the instructions prompted on the Ledger Hardware Wallet interface (either at the end of its onboarding process or later within the settings), create a PIN for the Ledger Recovery Key, and then tap the Ledger Recovery Key on the Ledger Hardware Wallet so that all the information can be securely transferred to the Ledger Recovery Key. In this experience, the Ledger Hardware Wallet's PIN is required to proceed up to interacting with the Ledger Recovery Key product.



Step 1/3 - Initial setup for a Recovery Key backup



Step 2/3 - After having followed the instructions, NFC communication between the two products

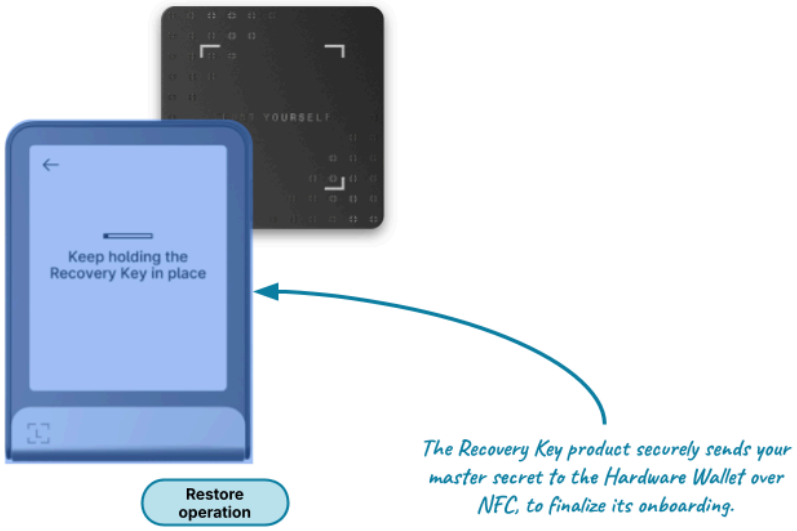


Step 3/3 - Result at the end of the backup operation

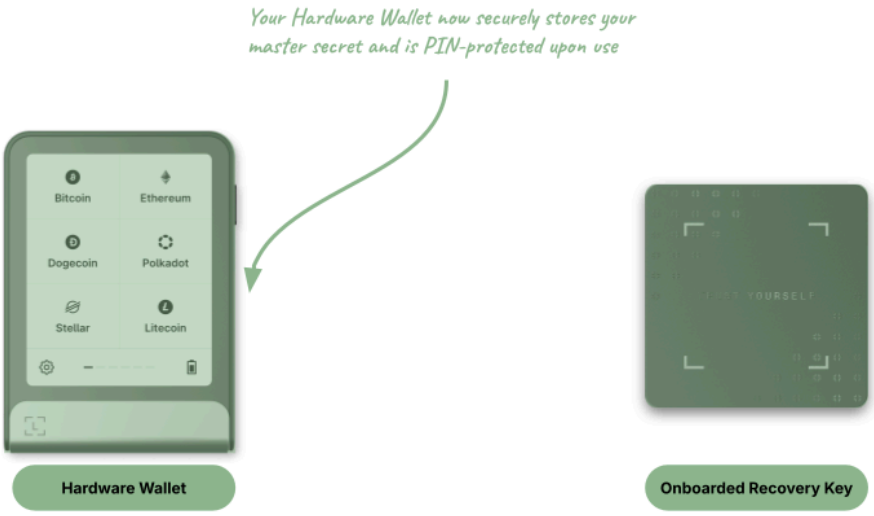
3.2 - Feature - Restore seed from a Recovery Key

The second essential feature consists in giving the possibility to the user to transfer their master secret from their Ledger Recovery Key to their Ledger Hardware Wallet. To use this feature, the Ledger Hardware Wallet must not yet be onboarded and thus does not yet contain a master secret.

Step 1/3 - Initial setup for a Recovery Key restore



Step 2/3 - After having followed the instructions, NFC communication between the two products

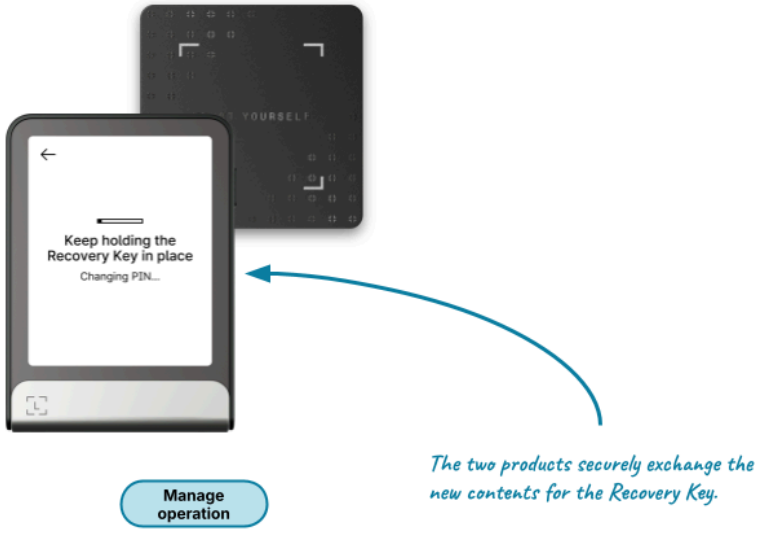


Step 3/3 - Result at the end of the restore operation

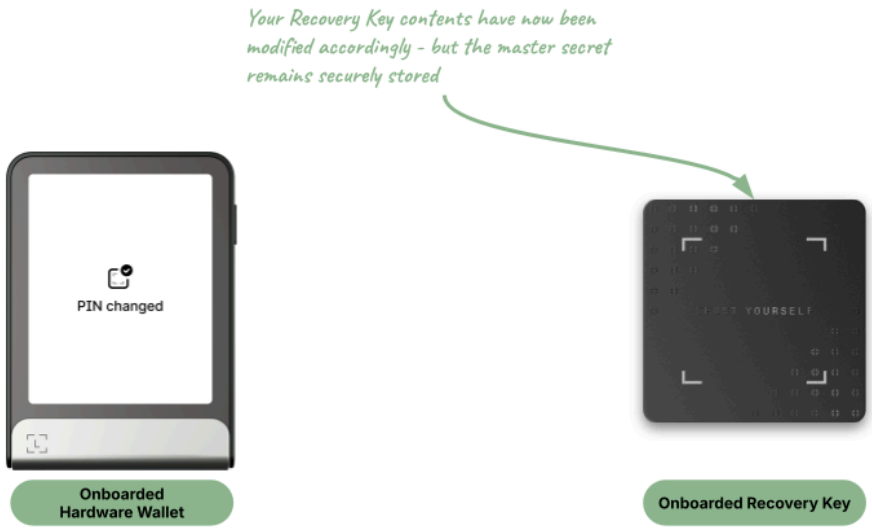
3.3 - Feature - Manage Recovery Key contents

The third essential feature consists in giving the possibility to the user to manage their Ledger Recovery Key contents, such as changing its PIN, creating, changing or deleting its name, or wiping the contents. To use this feature, the Ledger Hardware Wallet and the Ledger Recovery Key products must both be onboarded.

Step 1/3 - Initial setup for a Recovery Key contents management



Step 2/3 - After having followed the instructions, NFC communication between the two products



Step 3/3 - Result at the end of the manage operation

3.4 - Feature - Recovery Key Update

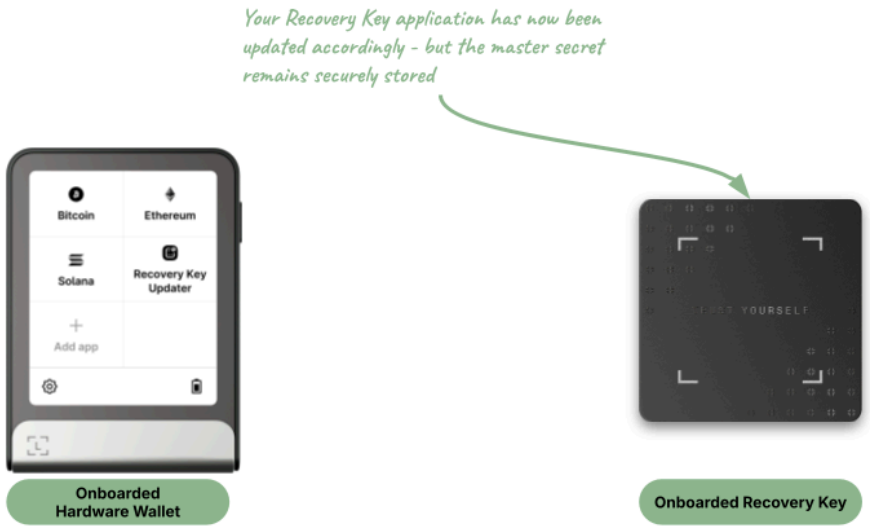
The last essential feature consists in giving the possibility to the user to update a portion of their Ledger Recovery Key software to reach several goals, such as adding features, fixing issues, and improving the security of the card over time. At Ledger we consider that the security of our products cannot be static, and that we need to make sure we implement and deploy the necessary security improvements when needed. To this extent, the Ledger source code of the application running on the Ledger Recovery Key product is made available on github to make sure our users can verify the implementation.

In the same vein that updating an onboarded Ledger Hardware Wallet is not possible without validating the user consent via verifying his PIN, updating an onboarded Ledger Recovery Key product can only be performed after having verified its own PIN, which is one of the reasons why updating the Ledger Recovery Key software can only be performed from a Ledger Hardware Wallet.

Step 1/4 - Initial setup

Step 2/4 - Download of the Recovery Key Updater device application

Step 3/4 - Transferring the Recovery Key updated application to the card



Step 4/4 - Final result after the operation

4 - Secure Elements and Hardware Security Modules

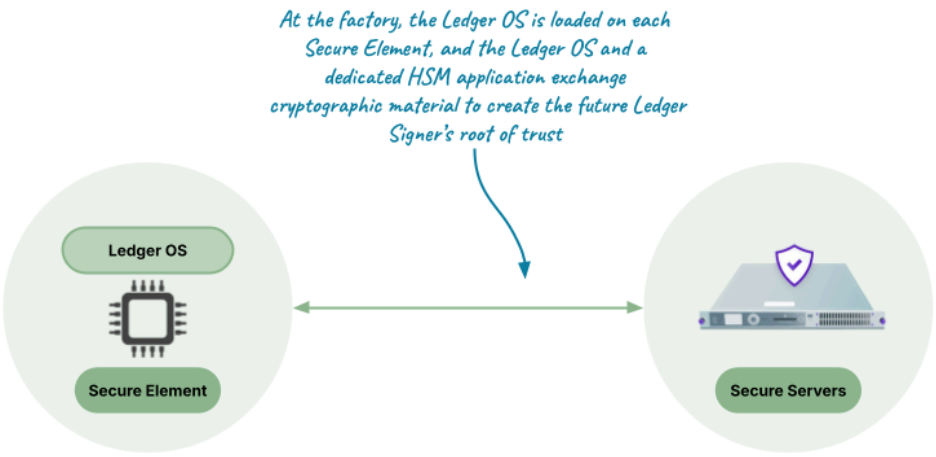
As mentioned in the Ledger Recover white paper, a Secure Element is a tamper-resistant processor chip, providing security countermeasures aiming to make a product embedding such a chip resist a wide range of attacks from fault attacks to side-channel attacks for instance.

Operating Systems powering these Secure Elements usually leverage these security features to protect secret data, to isolate the execution of the different components from each other, and to resist attacks aiming to extract these secret data. The embedded software stack powering Ledger Hardware Wallets is designed to provide several security mechanisms to this extent, as mentioned in our Donjon threat model.

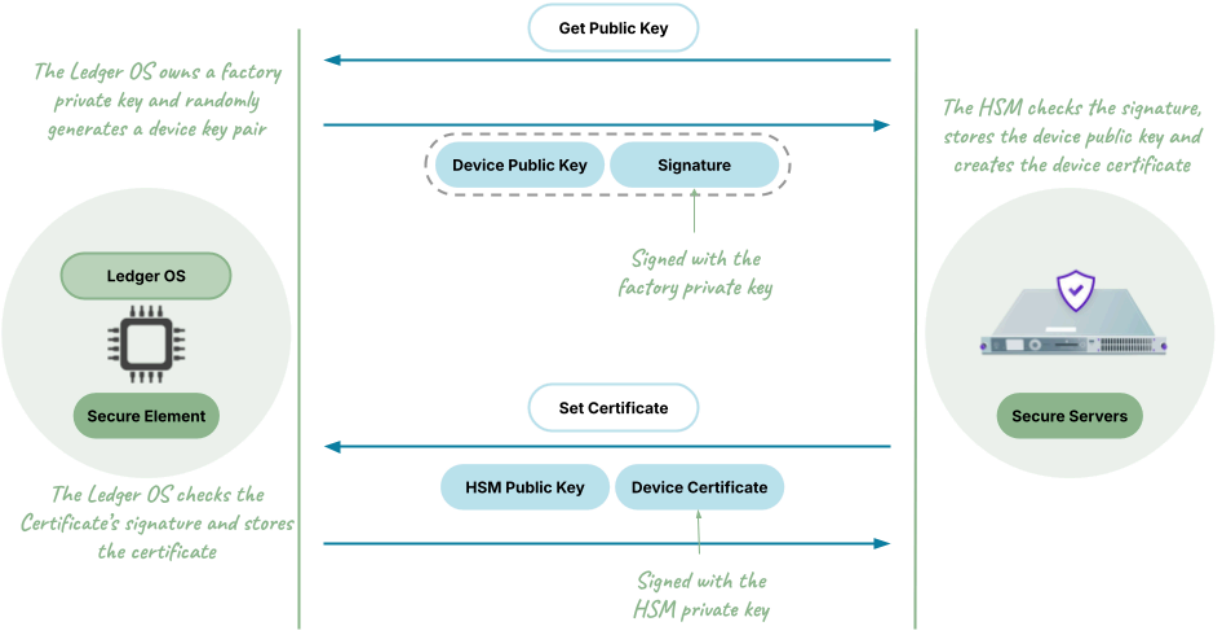
A HSM is a physical device, most commonly found under the form factor of a network interface controller, used to manage and securely store secret data, usually cryptographic keys, and which provides an interface dedicated to perform cryptographic computations with these keys from within the secure environment. Typical use cases range from managing keys for website security to payment transaction processing, banking cards production, and many more.

At Ledger we use HSMs – and we develop the software powering them – to various extents. One of our miscellaneous use cases consists in securely hosting the device applications and OS updates users can install on their Ledger Hardware Wallet, as well as provide a secure way to co-create the cryptographic material, on a per Ledger product basis, dedicated to install a root of trust within the Ledger Hardware Wallets and allowing all Ledger products to successfully undergo software genuineness checks once they are deployed in the field. As another example, the Ledger Recover feature discussed at length in the associated white paper also makes extensive use of this secure combination of hardware and software.

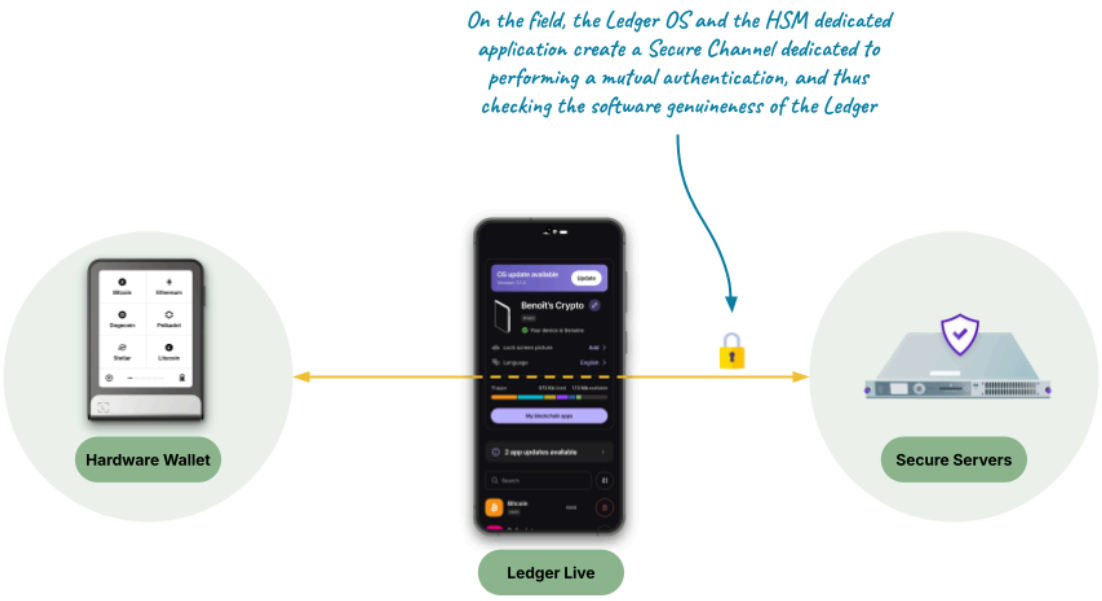
In the sense of the security guarantees these two types of hardware components provide, one could consider a Secure Element as being a portable HSM.



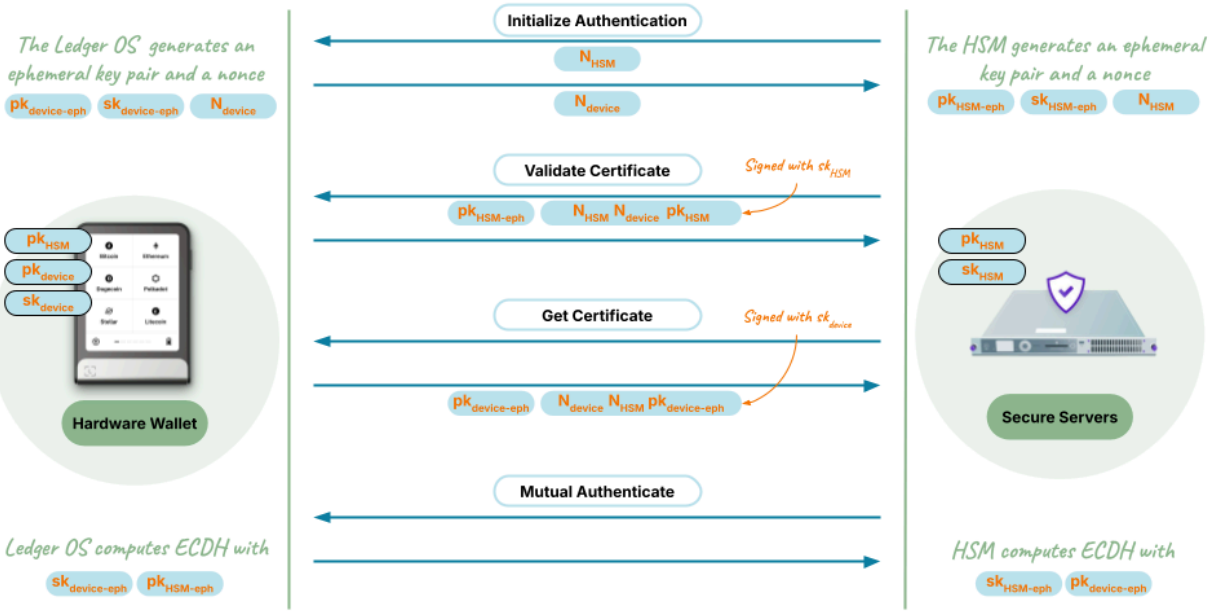
Secure Element and HSM when preparing a Secure Element at the factory



Secure Element and HSM when preparing a Secure Element at the factory - Detailed



Hardware Wallet and HSM in use - Creating a Secure Channel



Hardware Wallet and HSM in use - Creating a Secure Channel - Details

In the context of the Recovery Key product, Ledger once again heavily relies on the security brought with this existing environment. Next paragraphs will deep dive into the

design of the product and how we capitalized on our existing processes when including the Recovery Key product within the Ledger environment.

✒️Note

The first paragraphs of the Ledger blogpost discussing the Ledger Recover's shares distribution also discusses in detail the way Ledger creates Secure Channels between two secure endpoints.

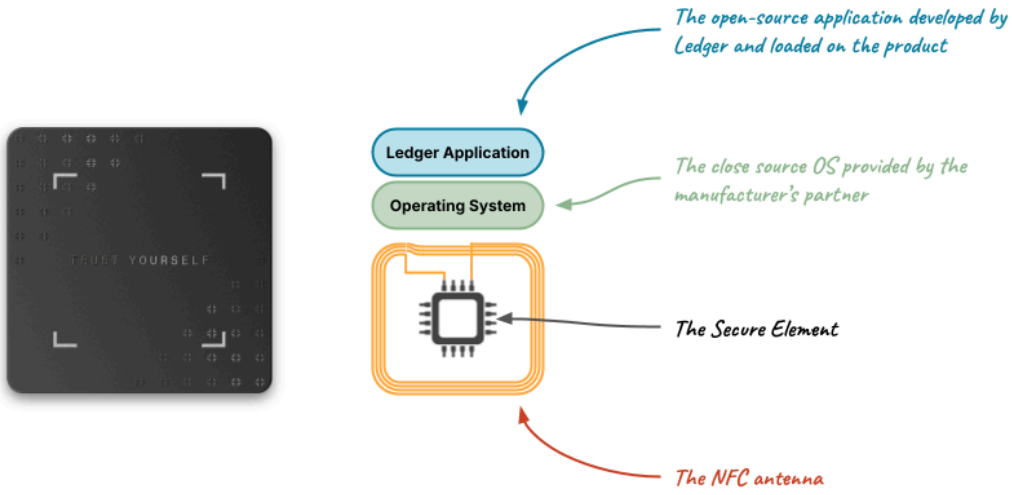
5 - Recovery Key - From a design perspective

As previously mentioned, the Ledger Recovery Key product embeds a Secure Element which provides the product with its main security layer. This secure chip's capabilities are leveraged by an operating system, allowing Ledger to develop a fully open source application which has the responsibility to execute the business logic and combine secure data transfers with secure storage and cryptographic computations. The Github repository allowing everyone to check the Ledger implementation on the Ledger Recovery Key product can be found here: https://github.com/LedgerHQ/applet-recovery-key.

Within the Ledger Recovery Key product, the Secure Element is a NXP P71D600 provided with a JCOP4.5 operating system. The combination of these two items has passed a Common Criteria EAL6+ security certification:

- Common Criteria Light Security Target
- Common Criteria Certification Report

The secure storage in the Ledger Recovery Key product thus relies on both the Secure Element and associated operating system embedded in the product, but also on the implementation of the Ledger application which manages the PIN, the cryptographic keys dedicated to perform the necessary procedures to securely transfer data between the Ledger Recovery Key product and the Ledger Hardware Wallet.

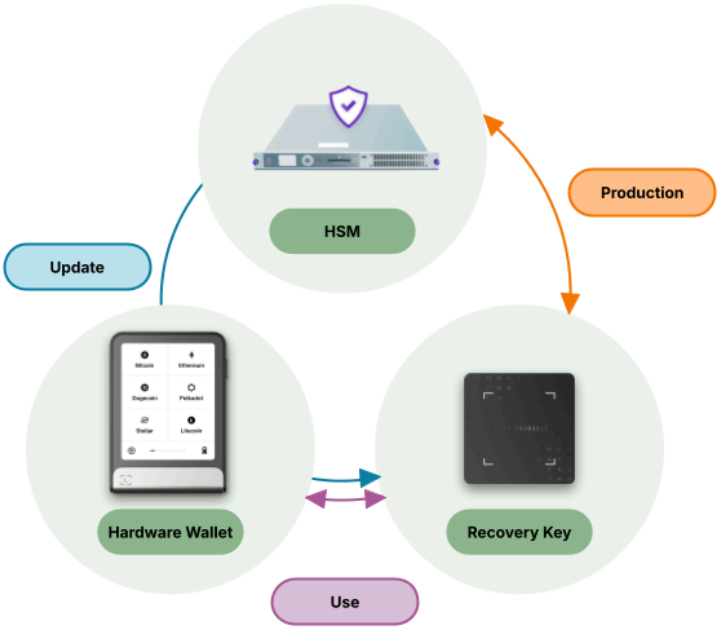


Recovery Key - From the inside

6 - Recovery Key - From a components perspective

This paragraph presents the interactions of the Ledger Recovery Key product with the other main technical Ledger components, from a high-level perspective. These main components, as mentioned in the previous paragraphs and as represented in the next picture, are:

1. The HSM which will both be involved in producing the Recovery Key at the factory and updating it once in the field,
2. The Ledger Hardware Wallet with NFC capabilities,
3. The Ledger Recovery Key product itself.

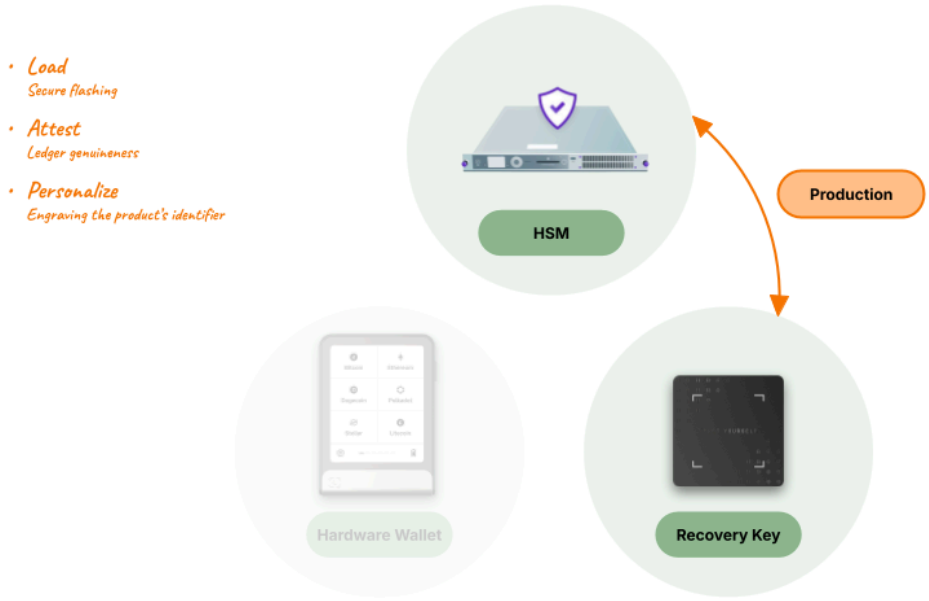


Recovery Key - Overall high-level environment

6.1 - Factory

The Ledger Recovery Key factory production environment is controlled in Ledger facilities, at Vierzon. This geographical positioning, along with initial sets of cryptography keys, allows Ledger to make sure the produced Ledger Recovery Key products are originating from Ledger. The environment is twofold:

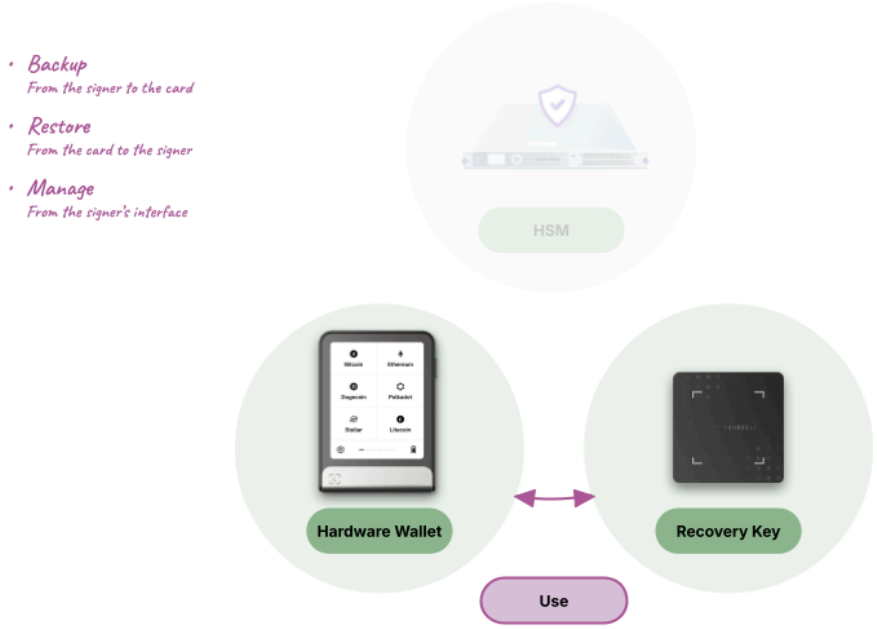
- The way it is manufactured – the Operating System is provided with specific cryptographic keys ensuring that applications can only be loaded if signed by Ledger HSMs,
- The way it is prepared by Ledger from a functional point of view – the application is loaded on top of the OS, and cryptographic operations are conducted between Ledger Recovery Key and our HSMs to create and securely store the unique attestation data aiming at making the Ledger Recovery Key product successfully pass the genuine check once in the field.



Producing the Recovery Key - from a high-level perspective

6.2 - Usage in the field

As presented from a user experience perspective in previous paragraphs, using the Ledger Recovery Key product is performed locally between the Ledger Hardware Wallet and the product, via NFC.

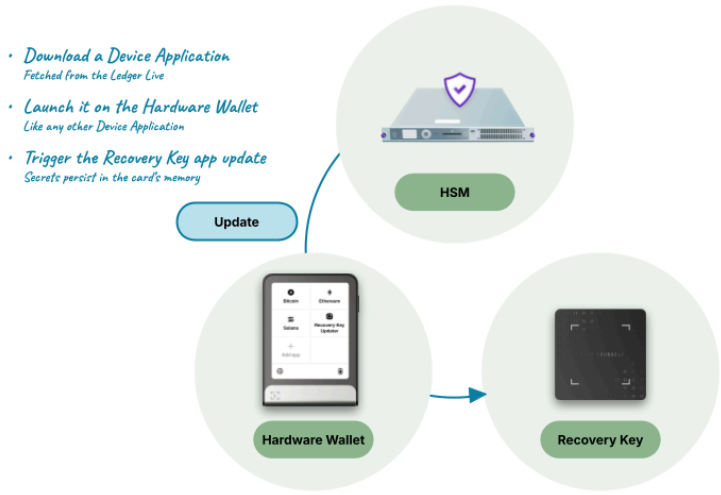


Using the Recovery Key - from a high-level perspective

6.3 - Updating in the field

As presented from a user experience perspective in previous paragraphs, updating the Ledger Recovery Key product to benefit from an improved application on the product itself is performed in several steps:

- Downloading a specific application on the Ledger Hardware Wallet, containing the Ledger Recovery Key update,
- Launch it on the Ledger Hardware Wallet,
- Follow the associated steps and wait for the update to be fully transferred onto the Ledger Recovery Key.



Updating the Recovery Key - from a high-level perspective

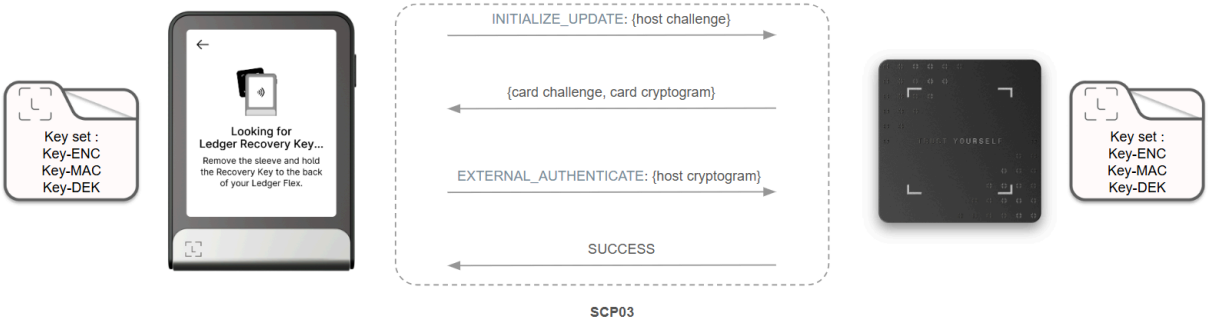
7 - Communication protocols and cryptography algorithms

The cryptography algorithms used within the overall communication protocols are the following.

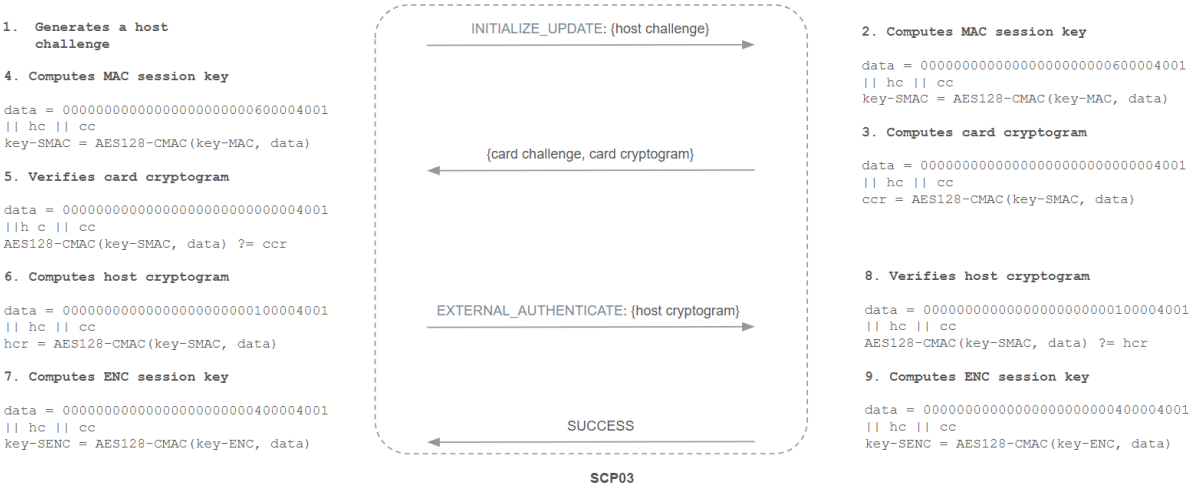
	Encryption	MAC	Hash	Signature verification	Key exchange	KDF
SCP03	AES-CBC	CMAC-AES				CMAC-AES
Genuine check			SHA256	ECDSA SECP256K1	ECDH SECP256K1	SHA256
PIN	AES-CBC	HMAC SHA512	SHA256			
Seed transfer	AES-CBC	HMAC SHA512				

7.1 - SCP03 Secure Channel

The first layer of security protocol Ledger is setting up between the Ledger Hardware Wallet and the Ledger Recovery Key product is a standardized Secure Channel - SCP03.



SCP03 creation - High level

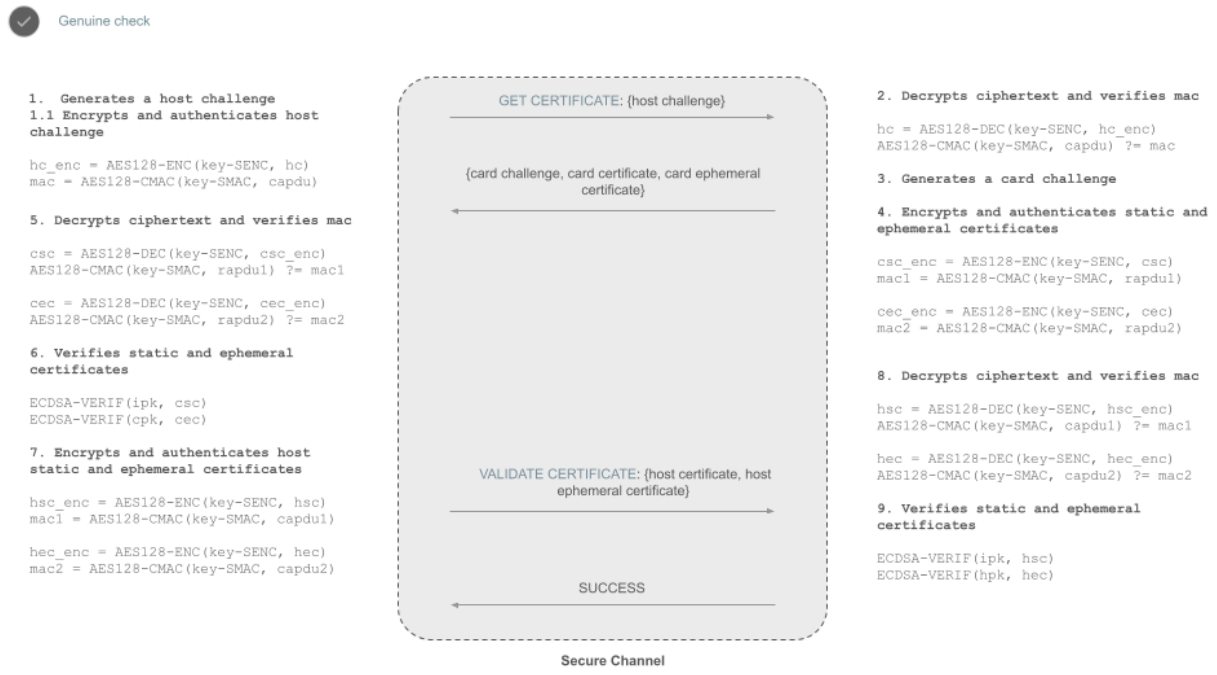


SCP03 creation - Details

7.2 - Ledger Secure Channel

Once the SCP03 secure channel has been created between the two products, they exchange information to create another Secure Channel, which Ledger is already using between its Hardware Wallets and its HSMs. This Ledger-based protocol is also used by the two products to mutually check their respective software genuineness by cross-checking their cryptographic attestation.

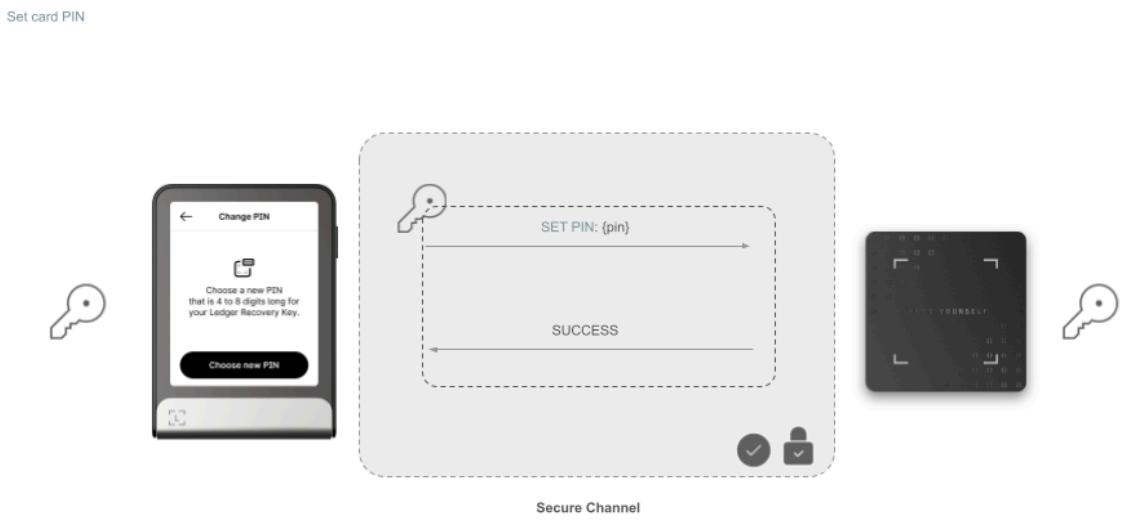
SCP Ledger creation - High level



SCP Ledger creation - Details

7.3 - Creating the Ledger Recovery Key PIN

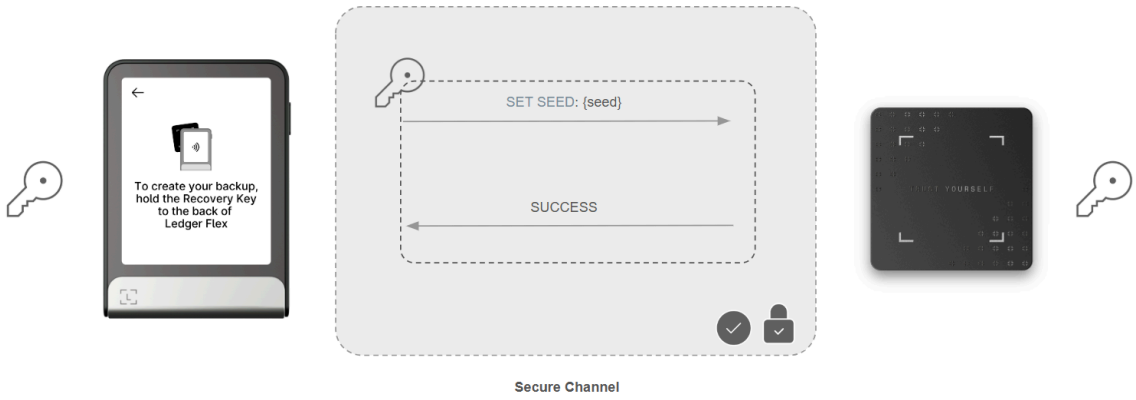
Once the two Secure Channels have been created one within the other, secret data can be securely communicated between the Ledger Hardware Wallet and the Ledger Recovery Key products, the picture below depicts the PIN exchange.



Securely creating the PIN, protected with the two Secure Channels

7.4 - Storing the secret data in Ledger Recovery Key

Once the two Secure Channels have been created one within the other, secret data can be securely communicated between the Ledger Hardware Wallet and the Ledger Recovery Key products, the picture below depicts the seed exchange (your master secret stored within the Ledger Hardware Wallet under the form of the Recovery Phrase).

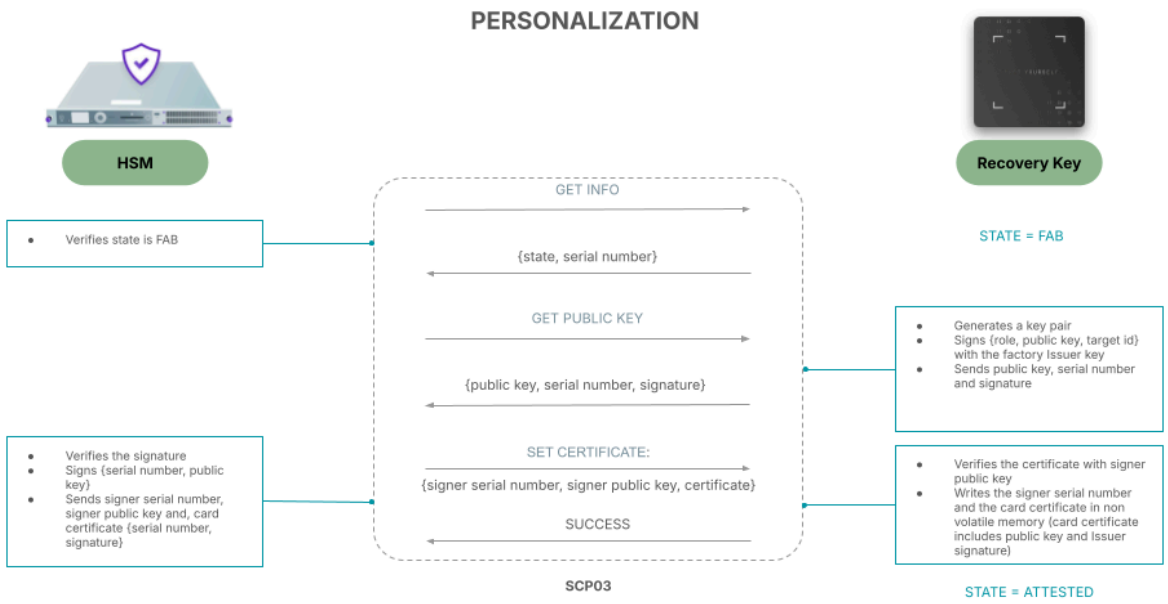


Securely storing the secret, protected with the two Secure Channels

8 - Recovery Key - From a technical perspective

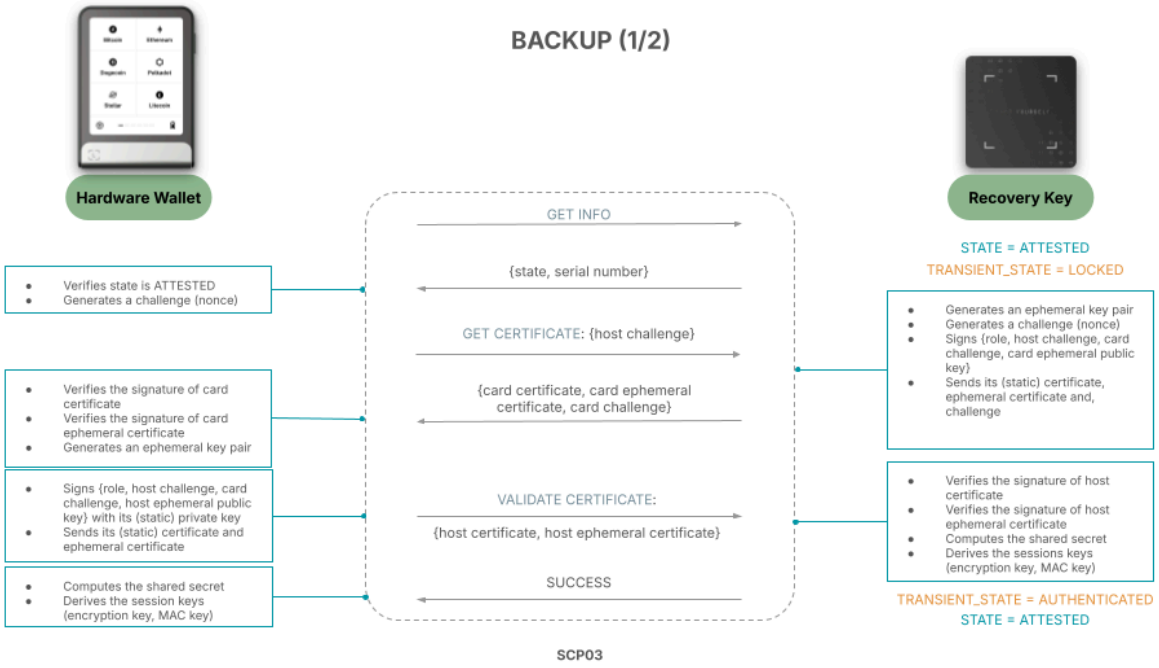
The next paragraphs describe all the communication interactions between the Ledger Hardware Wallet and the Ledger Recovery Key, protected with the combination of the two Secure Channels described in the previous section.

8.1 - Personalization

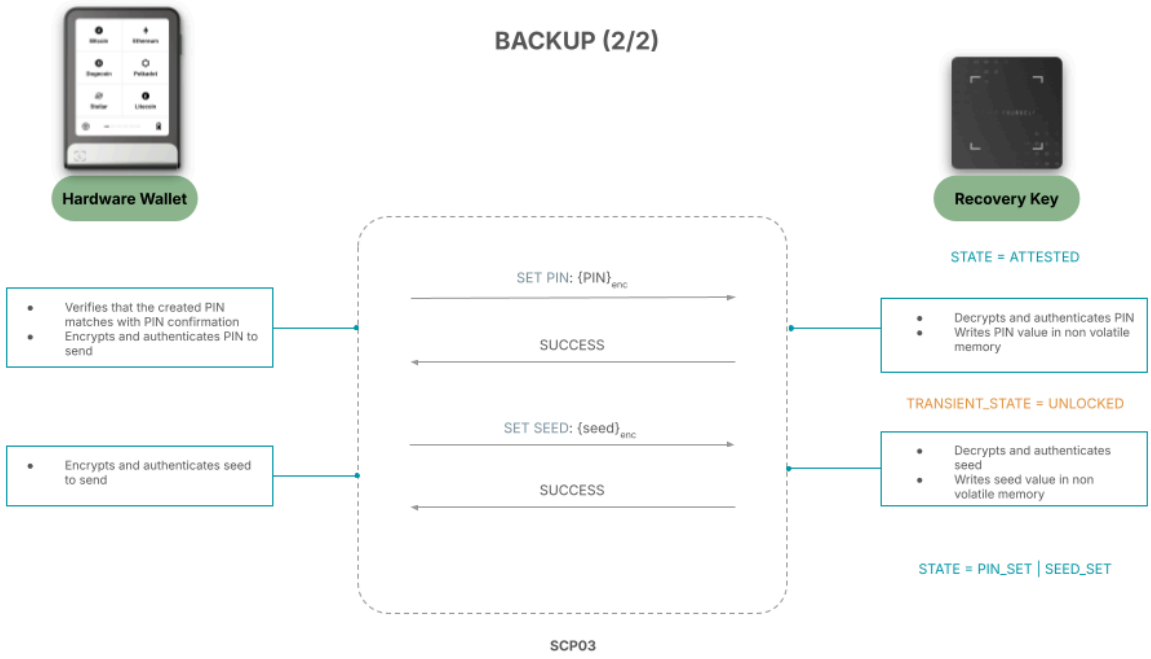


Recovery Key - Personalization

8.2 - Feature - Backup seed to a Recovery Key

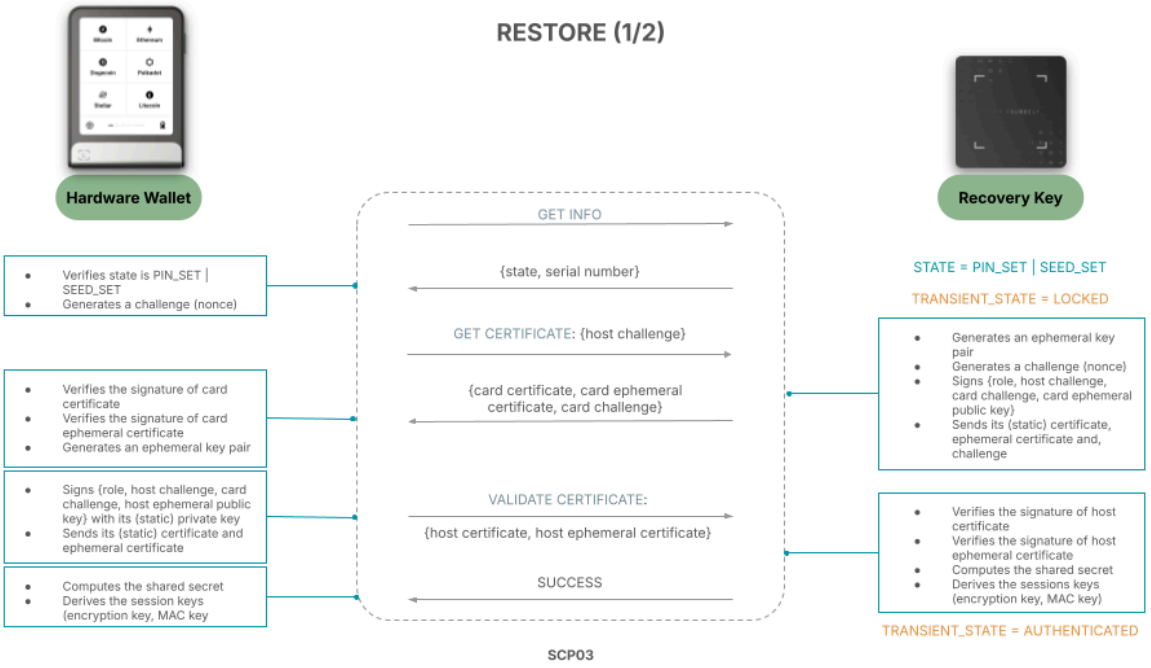


Recovery Key - Backup feature, part 1

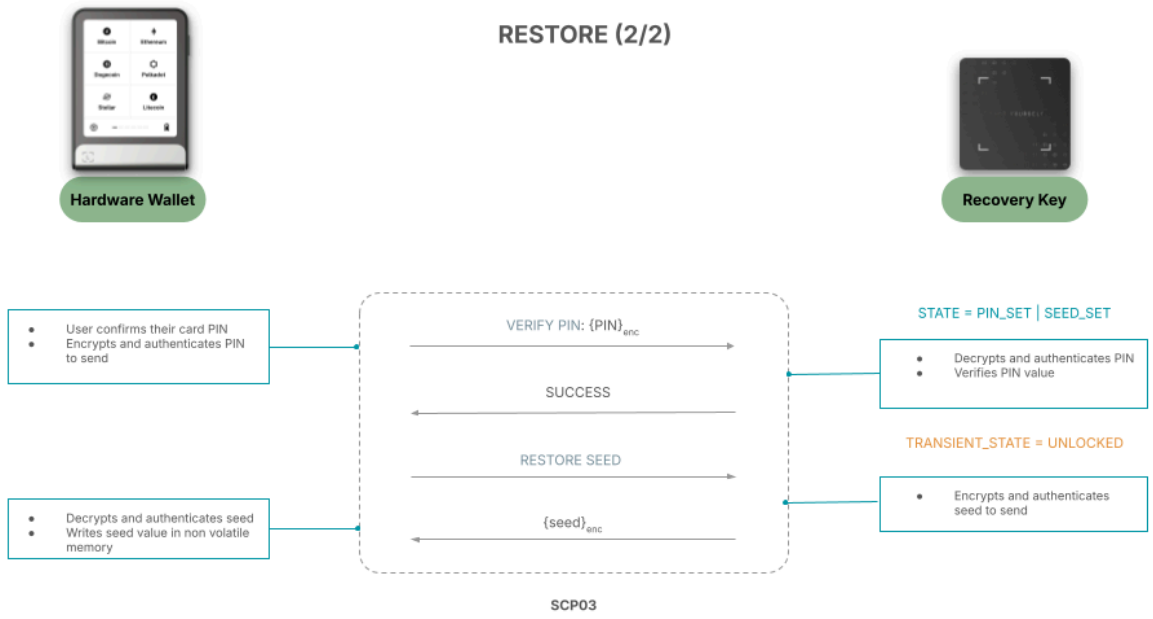


Recovery Key - Backup feature, part 2

8.3 - Feature - Restore seed from a Recovery Key

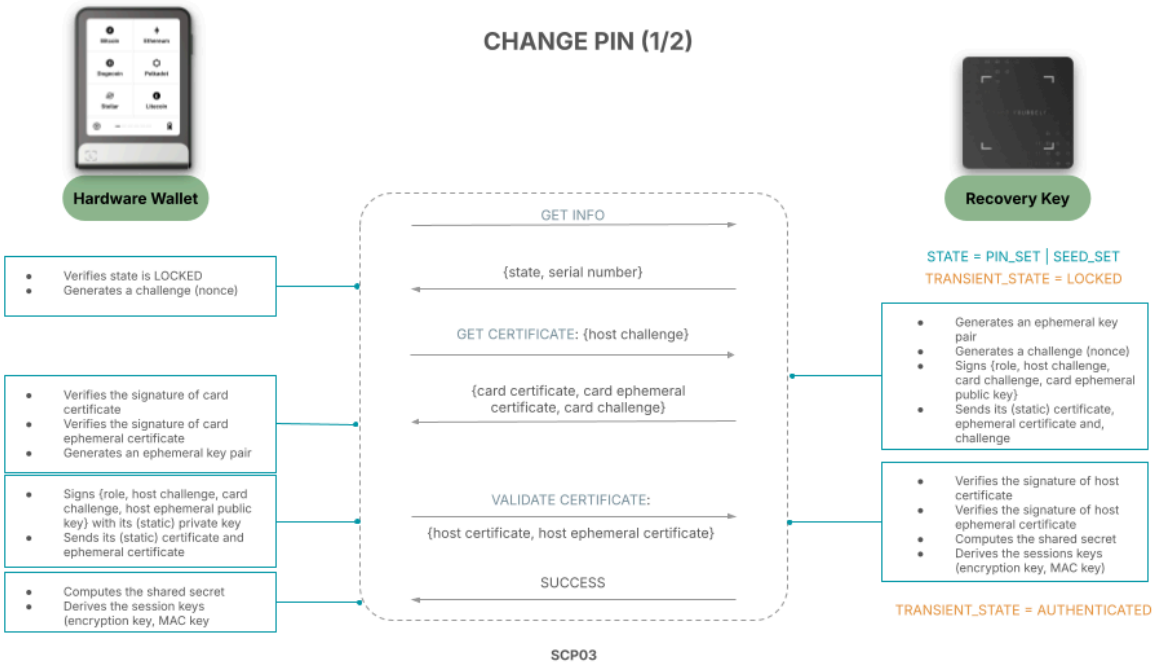


Recovery Key - Restore feature, part 1

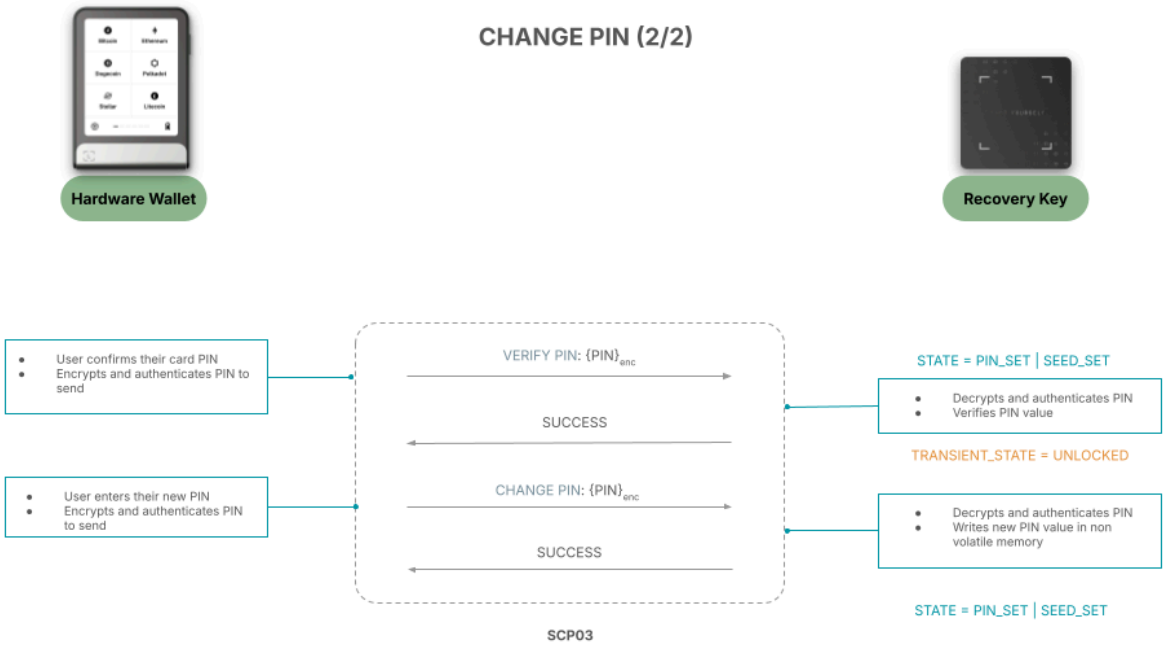


Recovery Key - Restore feature, part 2

8.4 - Feature - Manage Recovery Key contents



Recovery Key - Change PIN feature, part 1



Recovery Key - Change PIN feature, part 2

9 - Conclusion

In conclusion, and as demonstrated within this document, Ledger makes use of state of the art secure storage capabilities coupled with secure cryptographic protocols to ensure the Ledger Recovery Key product gets provided with end-to-end security, in particular when it comes to:

- Producing and attesting it with our HSMs,
- Using it in the field, on both the security at rest and security at use aspects,
- Updating it in the field if needed.

10 - Afterwords

The Ledger Recovery Key product is eligible for the Ledger Bug Bounty program.
For any inquiries or feedback related to the present white paper, feel free to contact us.

Blogpost

Ledger Recovery Key
High-level blogpost

1 - Introduction

This blogpost presents an overview of Ledger Recovery Key, its architectural design, security, and used cryptographic protocols.

The technology described in this blogpost is the subject of one or more pending patent applications. The publication of this blogpost does not grant, either expressly or impliedly, any license, right, or permission to make, use, sell, or otherwise distribute the described technology.

Before you dive in:

- Ledger Recovery Key is a PIN-protected backup card securely storing a copy of your 24-word Secret Recovery Phrase.
- It features an embedded Secure Element chip powered by an operating system, the combination of which is certified from a security perspective.
- Ledger Recovery Key communicates with NFC supported Ledger devices.
- Ledger Recovery Key is a complement to, not a replacement of, the traditional Recovery Sheet that remains the recommended form of backup.

2 - Keywords

Keyword	Meaning
HSM	Hardware Security Module
NFC	Near Field Communication
PIN	Personal Identification Number
SE	Secure Element

3 - Ledger Recovery Key - From a user perspective

Ledger Recovery Key is a physical backup card that can be used to store a copy of your master secret (from which stems your Secret Recovery Phrase) from your Ledger Hardware Wallet, provided that this Hardware Wallet supports NFC.

The Ledger Recovery Key product can only use the NFC protocol to communicate with a Ledger Hardware Wallet.

> ✒️Note
>
> Ledger Recovery Key is not a Hardware Wallet, since it is not possible to use it to manage assets nor sign transactions.

Let's take a look at the security design:

Security	Comments
Security at rest	• The product embeds a Secure Element chip, the same type of hardware security Ledger is also using within its Ledger Hardware Wallets, powered by a certified operating system.
Security at use	• The Ledger Recovery Key and Ledger Hardware Wallet perform a mutual authentication allowing them to attest the genuineness of their counterpart, • The product is PIN-protected, and three wrong PIN verification attempts will trigger the product to wipe its memory, • The communication protocol is secured and ensures the confidential and authenticated transfer of data and secret data.
Secure display	• All the interactions with the Ledger Recovery Key are triggered and confirmed on the secure screen of the Ledger Hardware Wallet, • The Ledger Recovery Key Identifier is checked by the Ledger OS to ensure consistency across the different taps of Ledger Recovery Key on the Ledger Hardware Wallet, and always displayed on its screen so that the user can also verify it,

	• The Ledger Hardware Wallet will display an information screen indicating whether the presented Ledger Recovery Key matches the Secret Recovery Phrase currently stored on the device. This helps users verify consistency, especially if they manage multiple Ledger Recovery Keys.

In the flow diagrams throughout this article, the different colors indicate:
- An onboarded device in green,
- A not onboarded device in blue.

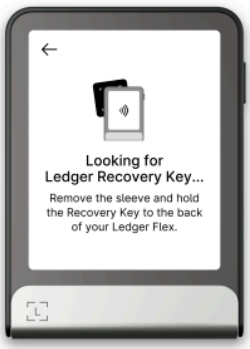
We can now look at the four main user features of Ledger Recovery Key:
- Backup,
- Restoration,
- Management,
- Updates.

3.1 - Back up to a Ledger Recovery Key

The first feature allows the user to make a backup of their master secret from their Ledger Hardware Wallet to their Ledger Recovery Key, and this will occur under one of two circumstances:
- At the end of the onboarding process, for a new Ledger Hardware Wallet,
- At any moment, for an already onboarded device.

The user shall:
- Follow the instructions prompted on the Ledger Hardware Wallet interface,
- Create a PIN for the Ledger Recovery Key,
- Tap it on the Ledger Hardware Wallet so that all the information can be securely transferred to the Ledger Recovery Key via NFC.

The Ledger Hardware Wallet's PIN is required to proceed up to interacting with the Ledger Recovery Key product.

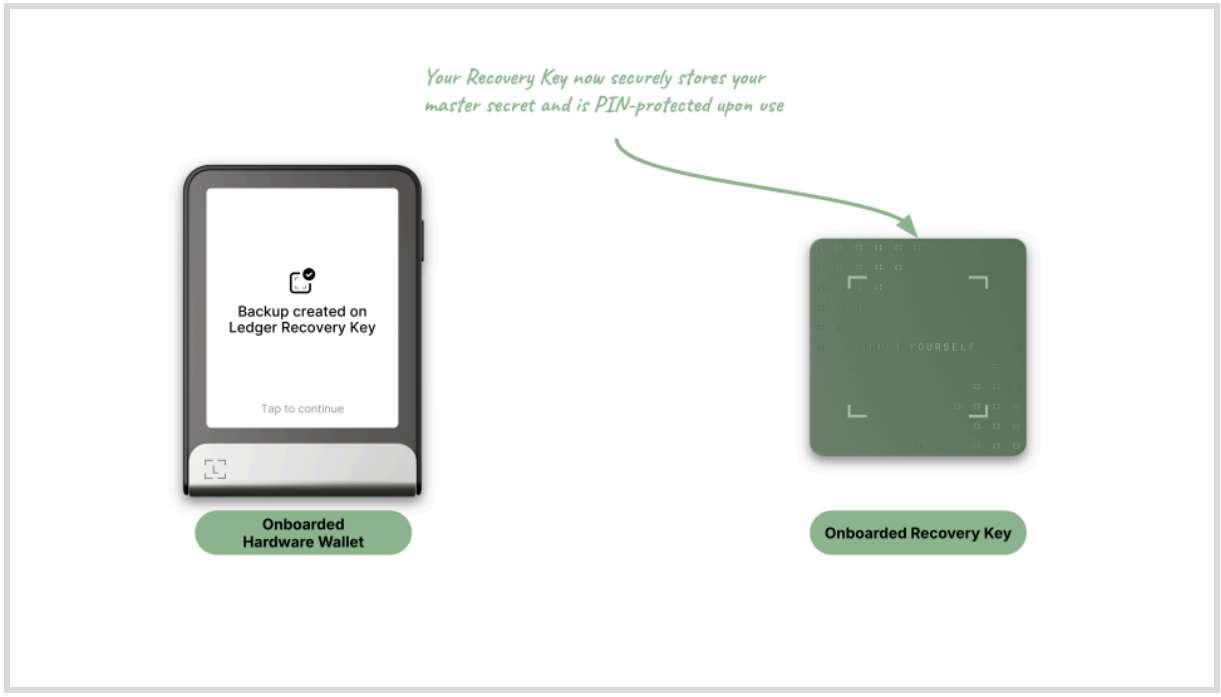


Onboarded Hardware Wallet

Empty Recovery Key

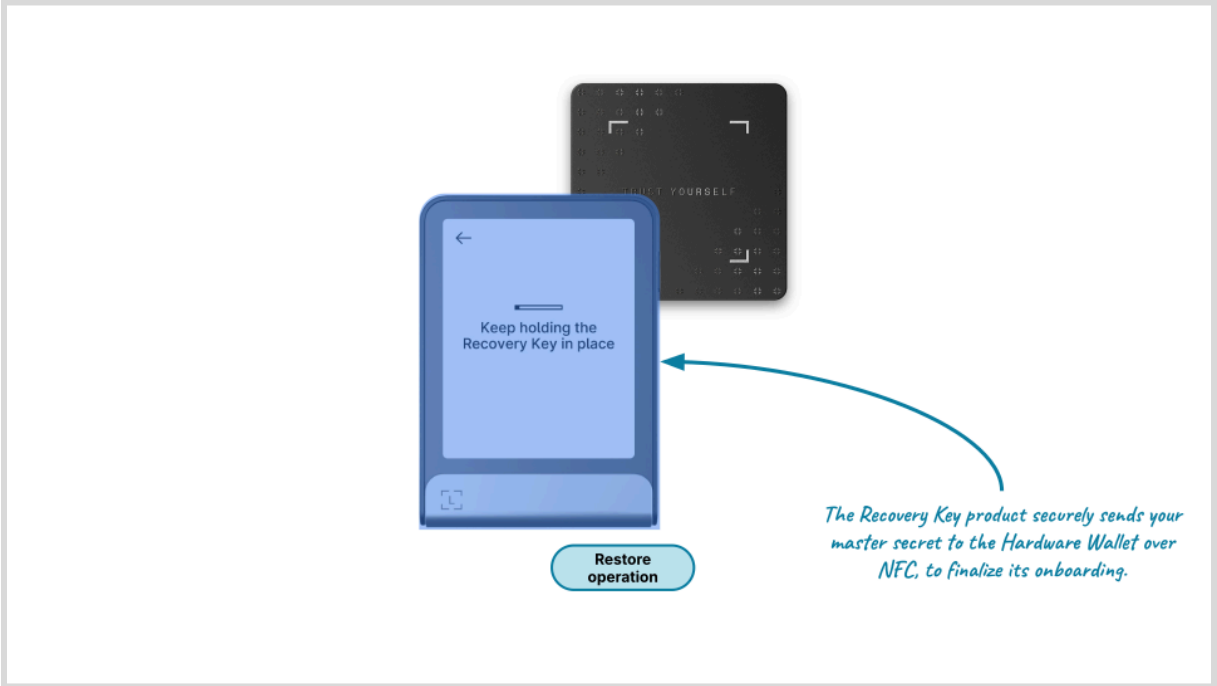
Backup operation

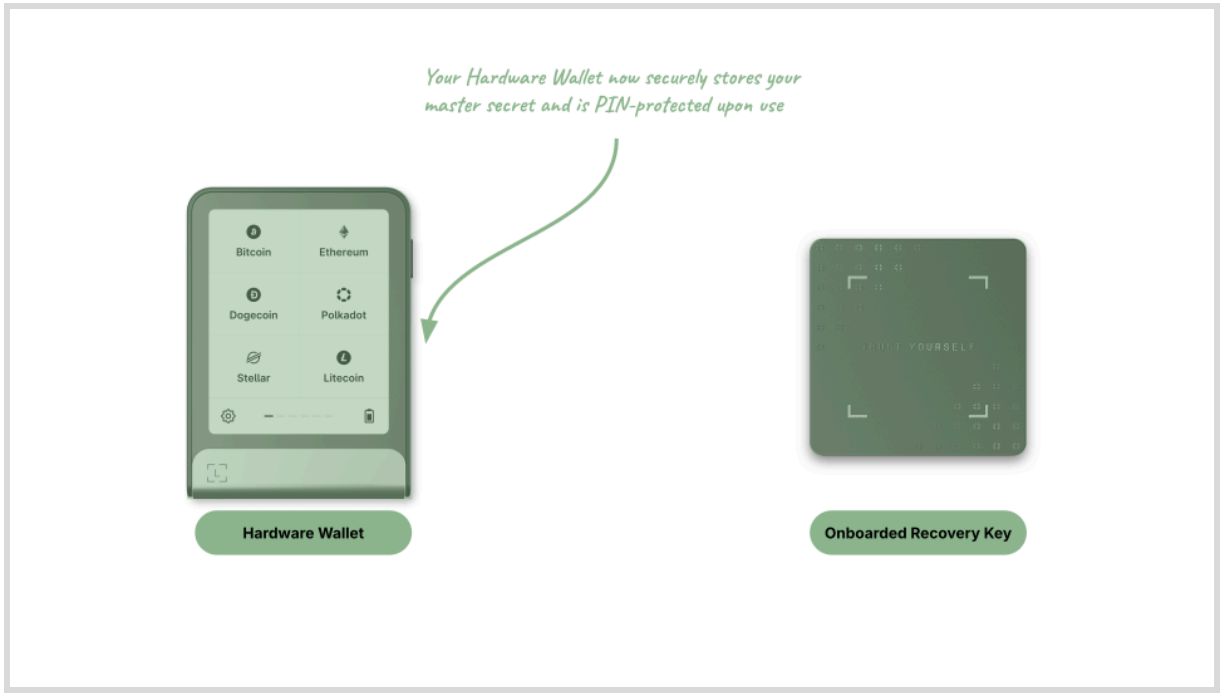
The chosen PIN and your master secret are securely exchanged over NFC


Your Recovery Key now securely stores your master secret and is PIN-protected upon use
Backup created on Ledger Recovery Key
Tap to continue
Onboarded Hardware Wallet
Onboarded Recovery Key

3.2 - Restore from a Ledger Recovery Key

The second feature allows the user to copy their master secret from their Ledger Recovery Key to their Ledger Hardware Wallet. To use this feature, the Ledger Hardware Wallet must not yet be onboarded and thus does not yet contain a master secret.



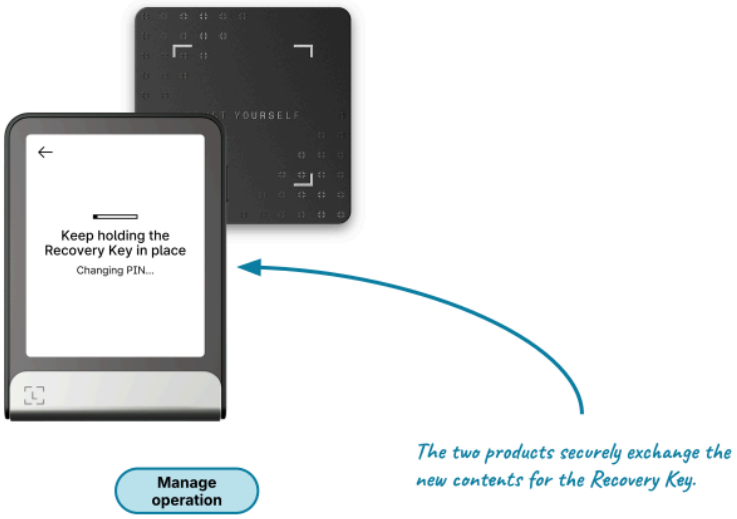

Your Hardware Wallet now securely stores your master secret and is PIN-protected upon use
Bitcoin
Ethereum
Dogecoin
Polkadot
Stellar
Litecoin
Hardware Wallet
Onboarded Recovery Key

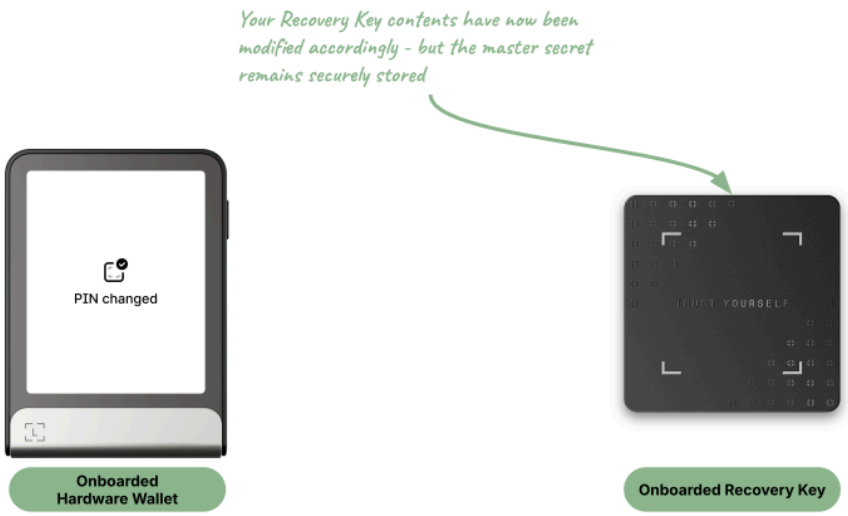
3.3 - Manage Ledger Recovery Key contents

The third feature allows the user to manage their Ledger Recovery Key contents, such as:

- Changing its PIN,
- Creating, changing or deleting its name,
- Wiping the user's data (master secret, PIN and name) to restore the product to its factory settings.

The Ledger Hardware Wallet and the Ledger Recovery Key products must both be onboarded.


Keep holding the
Recovery Key in place
Changing PIN...
The two products securely exchange the new contents for the Recovery Key.
Manage operation


Your Recovery Key contents have now been modified accordingly - but the master secret remains securely stored
PIN changed
Onboarded Hardware Wallet
Onboarded Recovery Key

3.4 - Ledger Recovery Key Update

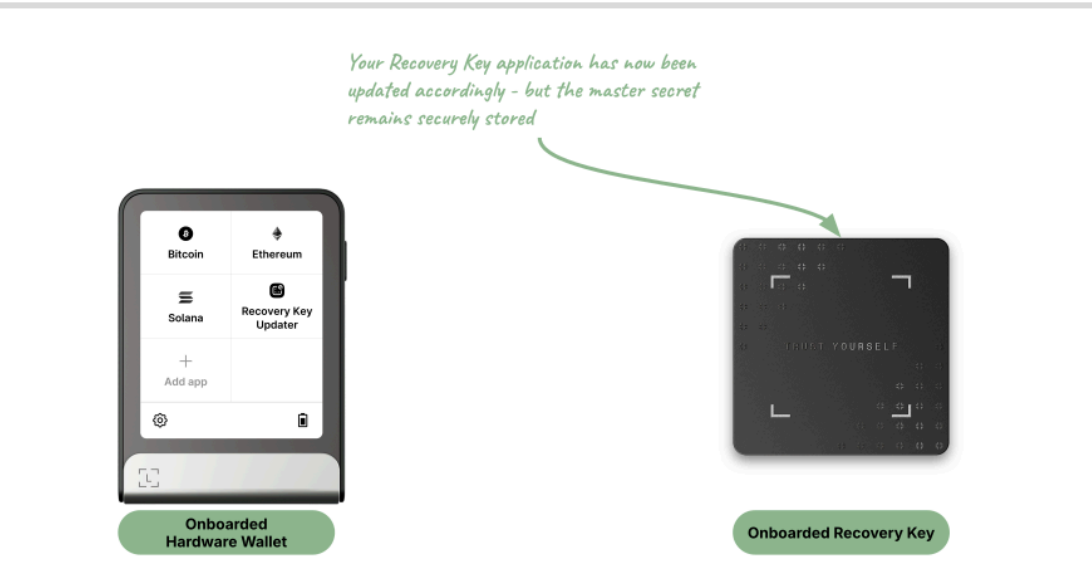
The fourth and last feature allows the user to update a portion of their Ledger Recovery Key software. This can happen for several reasons:

- Adding features,
- Fixing issues,
- Improving the security of the card over time.

At Ledger we consider that the security of our products cannot be static. We need to make sure we implement and deploy the necessary security improvements when needed. To this extent, the Ledger source code of the application running on the Ledger Recovery Key product is made available on github to make sure our users can verify the implementation.

In the same vein that updating an onboarded Ledger Hardware Wallet is not possible without validating the user consent via verifying their PIN, updating an onboarded Ledger Recovery Key product can only be performed after having verified its own PIN, which is one of the reasons why updating the Ledger Recovery Key software can only be performed from a Ledger Hardware Wallet.


OS update available
Update
Benoît's Crypto
Your device is Genuine
Lock screen picture
Add
Language
English
My blockchain apps
2 app updates available
Search
Bitcoin
Ledger Live
YOURSELF
Update
Ledger Recovery Key?
Version 1.2.0.
To see what's in the update,
visit: ledger.com/g3
Update
Don't update
Update operation
The Hardware Wallet securely sends the new Recovery Key application to the product


Your Recovery Key application has now been updated accordingly - but the master secret remains securely stored
Bitcoin
Ethereum
Solana
Recovery Key Updater
Add app
Onboarded Hardware Wallet
TRUST YOURSELF
Onboarded Recovery Key

4 - Secure Elements and HSMs

As mentioned in the Ledger Recover white paper, a Secure Element is a tamper-resistant processor chip, providing security countermeasures aiming to resist a wide range of attacks.

Operating Systems powering these Secure Elements usually leverage these security features to protect secret data, to isolate the execution of the different components from each other, and to resist attacks aiming to extract these secret data. The embedded software stack powering Ledger Hardware Wallets is designed to provide several security mechanisms to this extent, as mentioned in our Donjon threat model.

A HSM is a physical device, most commonly found under the form factor of a network interface controller, used to manage and securely store secret data, usually cryptographic keys, and which provides an interface dedicated to perform cryptographic computations with these keys from within the secure environment.

Typical use cases range from managing keys for website security to payment transaction processing, banking cards production, and many more. In the sense of the security guarantees these two types of hardware components provide, one could consider a Secure Element as being a portable HSM.

5 - Ledger Recovery Key - From a design perspective

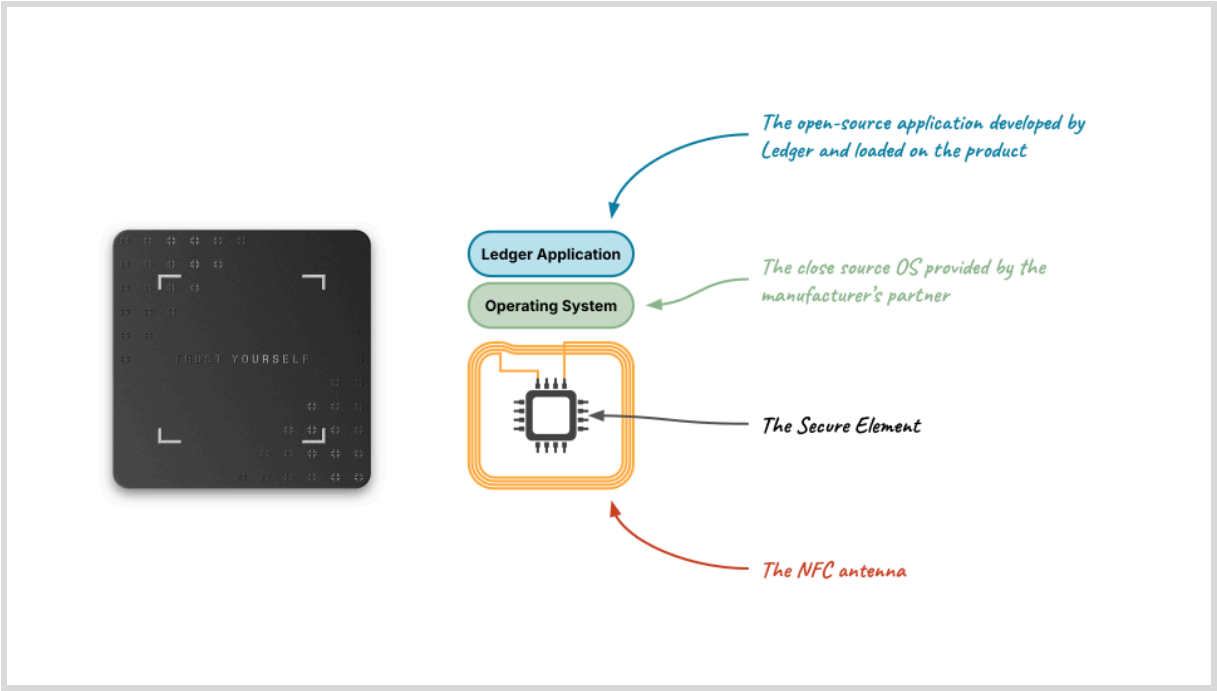
Ledger developed an open source Java Card application responsible for executing the business logic and combining secure data transfers with secure storage and cryptographic computations. The Github repository allowing everyone to check the Ledger implementation on the Ledger Recovery Key product can be found here:

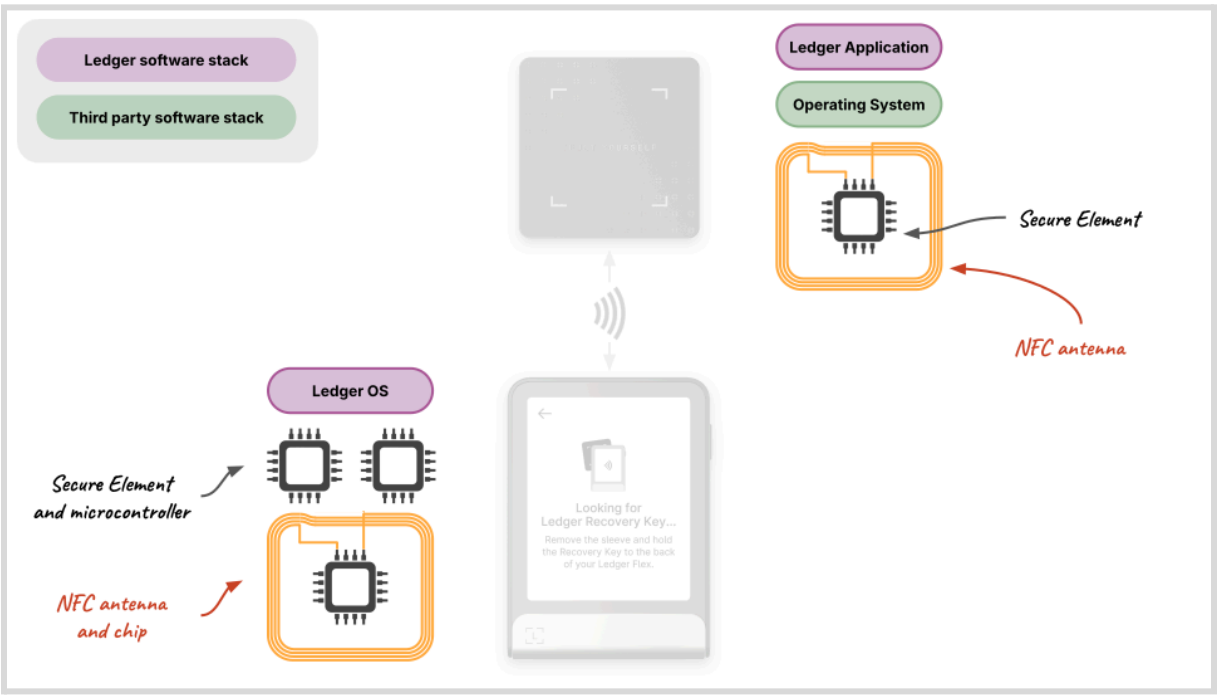
- https://github.com/LedgerHQ/applet-recovery-key.

Within the Ledger Recovery Key product, the Secure Element is a NXP P71D600 provided with a JCOP4.5 operating system. The combination of these two items has passed a Common Criteria EAL6+ security certification:

- Common Criteria Light Security Target
- Common Criteria Certification Report

The secure storage in the Ledger Recovery Key product thus relies on both the Secure Element and associated operating system embedded in the product, but also on the Ledger application managing the PIN and the cryptographic keys.

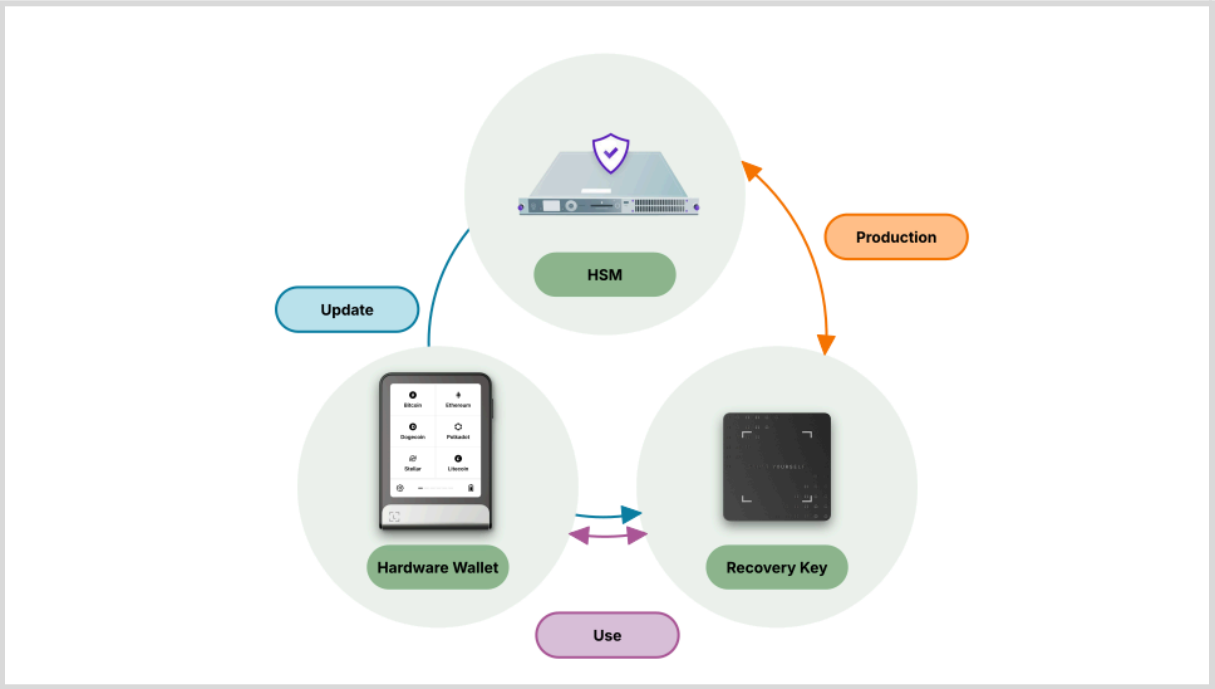

The open-source application developed by Ledger and loaded on the product
Ledger Application
Operating System
The close source OS provided by the manufacturer's partner
The Secure Element
The NFC antenna


Ledger software stack
Third party software stack
Ledger Application
Operating System
Secure Element
NFC antenna
Ledger OS
Secure Element and microcontroller
NFC antenna and chip
Looking for Ledger Recovery Key...
Remove the sleeve and hold the Recovery Key to the back of your Ledger Flex.

6 - Ledger Recovery Key - From a component perspective

We can now look at the interactions of the Ledger Recovery Key product with the other main technical Ledger components:

1. The HSM, involved in the Ledger Recovery Key manufacturing and updates preparation,
2. The Ledger Hardware Wallet with NFC capabilities,
3. The Ledger Recovery Key product itself.

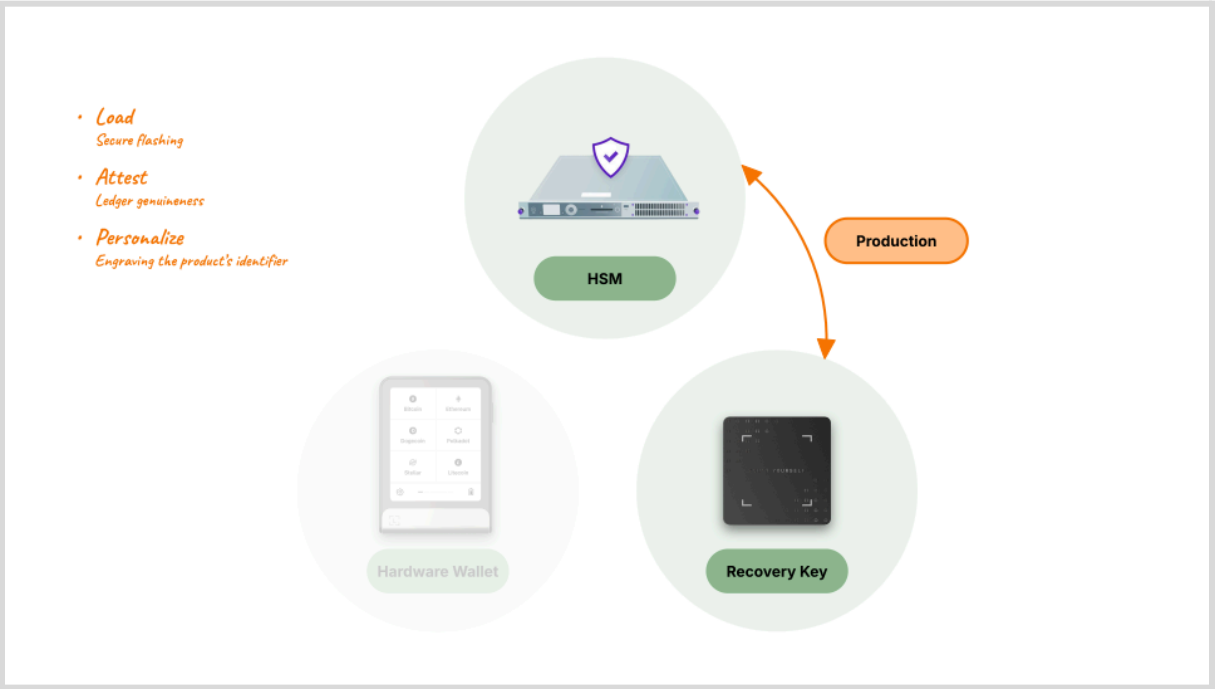


6.1 - Factory

The Ledger Recovery Key factory production environment is controlled in Ledger facilities, at Vierzon. This geographical positioning, along with initial sets of cryptography keys, allows Ledger to make sure the produced Ledger Recovery Key products are originating from Ledger. The environment is twofold:

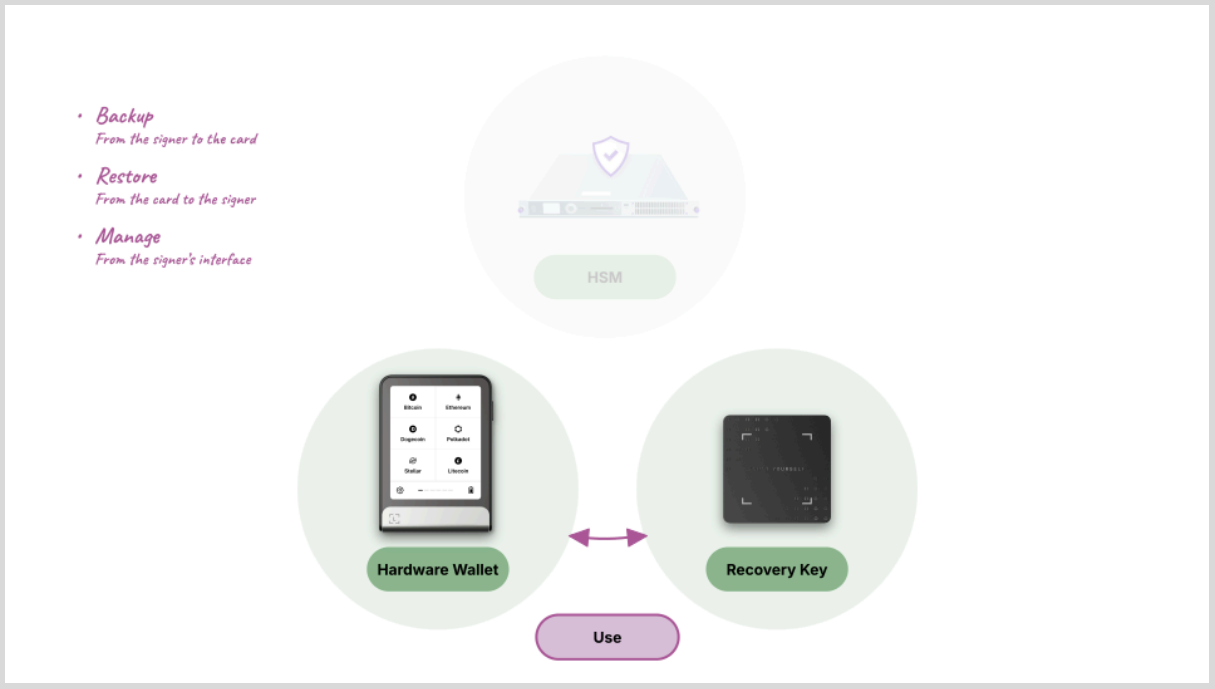
- The manufacturing process – the Operating System is provided with specific cryptographic keys ensuring that applications can only be loaded if signed by Ledger HSMs,
- The functional preparation – the application is loaded on top of the OS, and cryptographic operations are conducted between Ledger Recovery Key and our HSMs to create and securely store the unique attestation data aiming at making the

Ledger Recovery Key product successfully pass the genuine check once in the field.



6.2 - Usage in the field

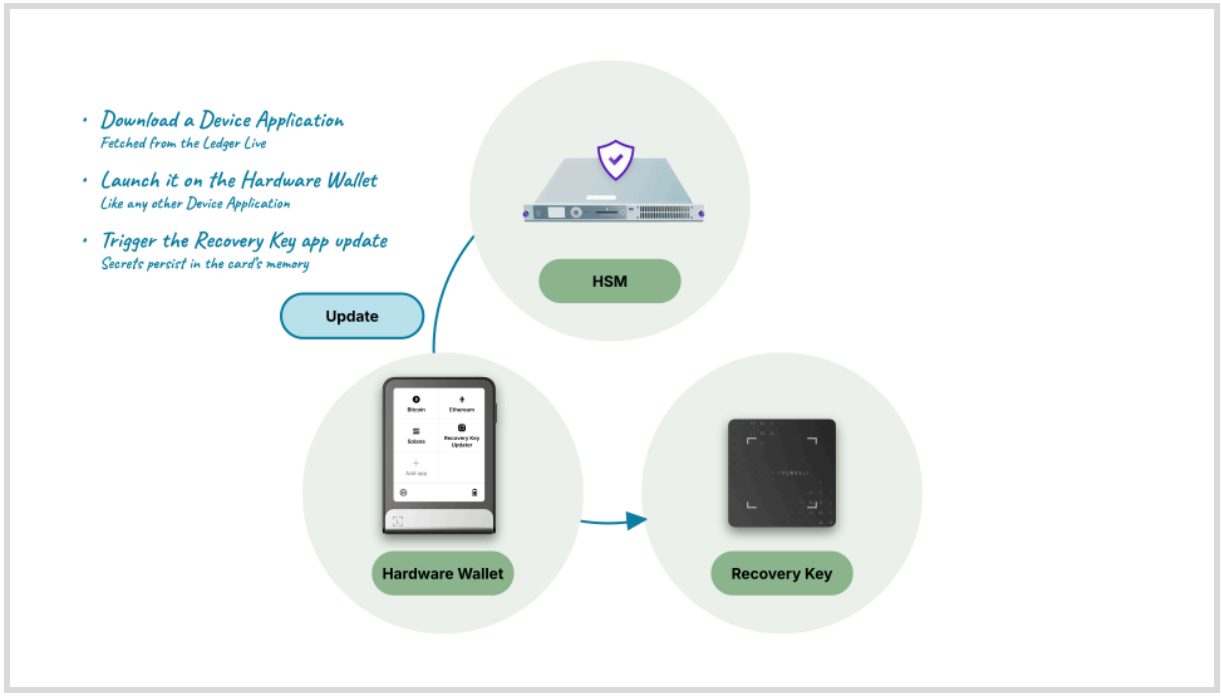
As presented in previous paragraphs, using the Ledger Recovery Key product is performed locally between the Ledger Hardware Wallet and the product, via NFC.



6.3 - Updating in the field

Updating the Ledger Recovery Key to benefit from an improved application on the product itself is performed in several steps:

- Downloading a specific application on the Ledger Hardware Wallet, containing the Ledger Recovery Key update,
- Launching it on the Ledger Hardware Wallet,
- Following the associated steps and waiting for the update to be fully transferred onto the Ledger Recovery Key.


Download a Device Application
Fetched from the Ledger Live
Launch it on the Hardware Wallet
Like any other Device Application
Trigger the Recovery Key app update
Secrets persist in the card's memory
Update
HSM
Hardware Wallet
Recovery Key

7 - Secure Operations

The cryptography algorithms used within the overall communication protocols are the following.

	Encryption	MAC	Hash	Signature verification	Key exchange	KDF
SCP03	AES-CBC	CMAC-AES	-	-	-	CMAC-AES
Genuine check	-	-	SHA256	ECDSA SECP256K1	ECDH SECP256K1	SHA256
PIN	AES-CBC	HMAC SHA512	SHA256	-	-	-
Seed transfer	AES-CBC	HMAC SHA512	-	-	-	-

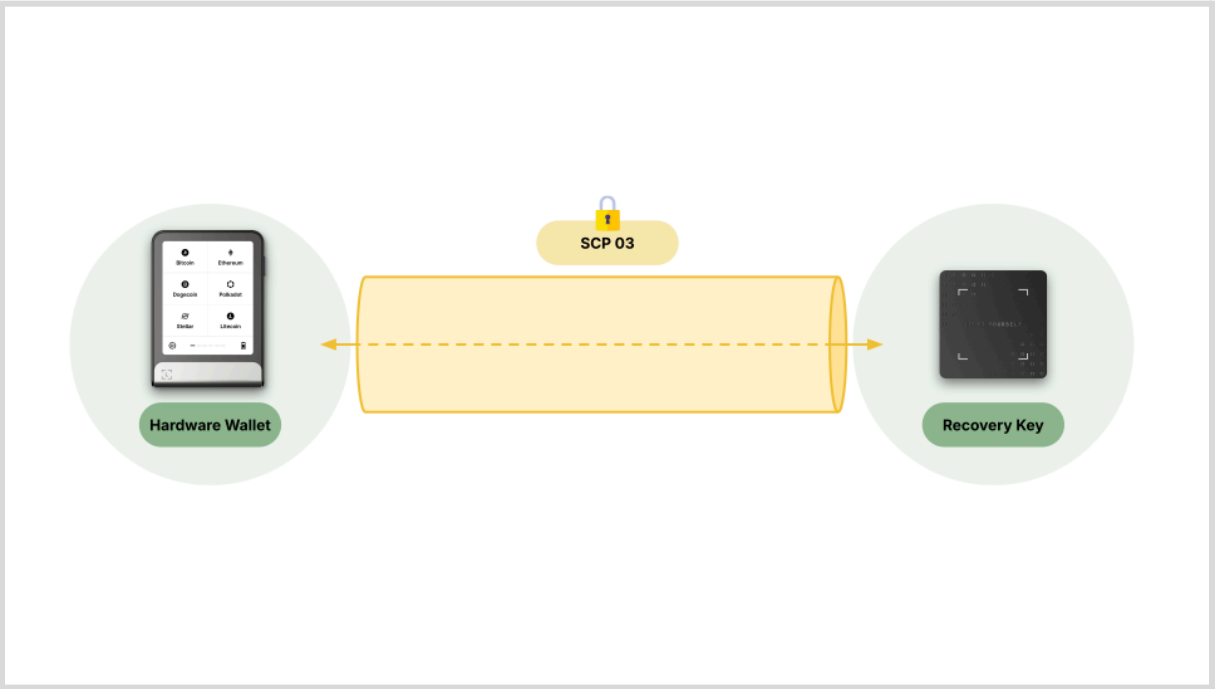
To go further regarding these algorithms, please find related standards and pieces of information in the following table.

Keyword	Meaning
AES-CBC	AES symmetric encryption algorithm used in CBC mode
AES-CMAC	AES-CMAC algorithm
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
HMAC	Keyed-Hash Message Authentication Code
SCP03	Secure Channel Protocol '03'
SECP256K1	Recommended Elliptic Curves Domain Parameters
SHA256 / SHA512	Secure Hash Standard

7.1 - Secure Channel - SCP03

The first layer of security protocol set up between the Ledger Hardware Wallet and the Ledger Recovery Key product is a standardized Secure Channel - SCP03. The two

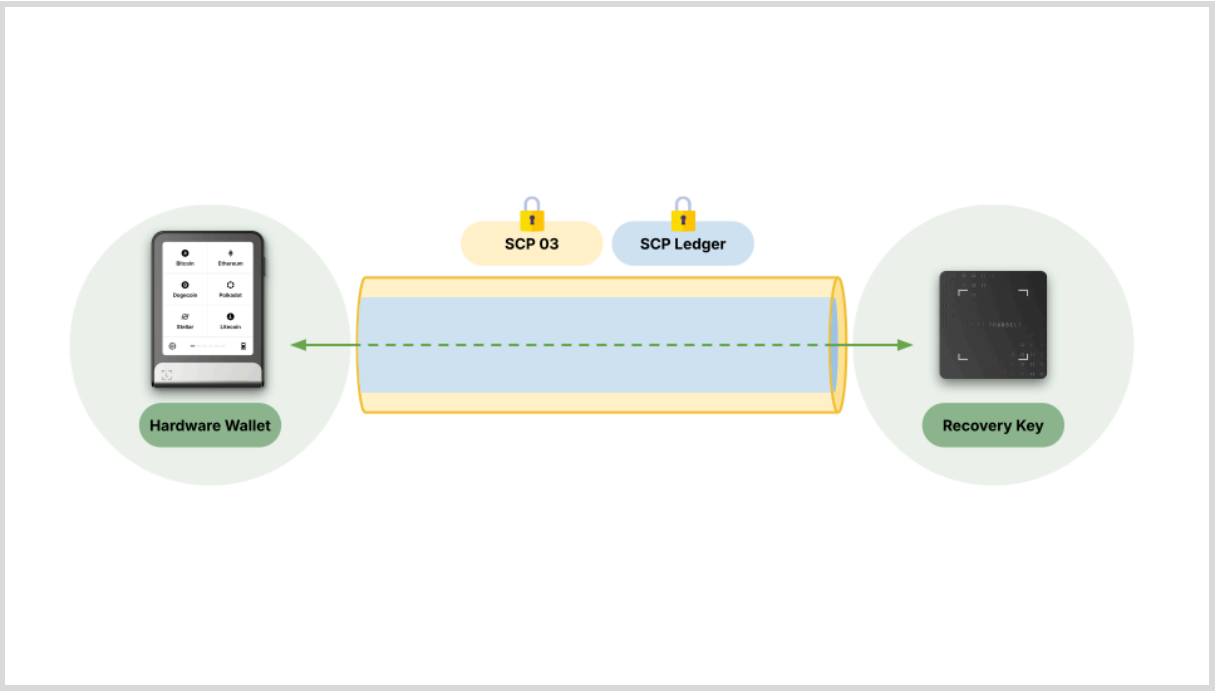
products can create such a Secure Channel thanks to the AES keys they already contain and the manufacturing process.



7.2 - Secure Channel - Ledger

Once the SCP03 secure channel has been created between the two products, they exchange information (securely within this Secure Channel) to create another Secure Channel, which Ledger is already using between its Hardware Wallets and its HSMs.

This Ledger-based protocol is also used by the two products to mutually check their respective software genuineness by cross-checking their cryptographic attestation.



7.3 - Secured operations

All the Ledger Recovery Key features are conducted within the combination of these two Secure Channels, by exchanging data between the two products as represented by the green arrow on the picture above:

- Creating the Ledger Recovery Key PIN,
- Storing the user secret data in the Ledger Recovery Key,
- Verifying the PIN to unlock the Ledger Recovery Key,
- Restoring the master secret from the Ledger Recovery Key to the Ledger Hardware Wallet,
- Changing the Ledger Recovery Key's PIN,
- Creating, changing or deleting the Ledger Recovery Key's name,
- Wiping the Ledger Recovery Key.

8 - Conclusion

In conclusion, and as shown in this document from a high-level perspective, Ledger makes use of state of the art and certified secure storage coupled with secure cryptographic protocols to ensure the Ledger Recovery Key product gets provided with end-to-end security, in particular when it comes to:

- Manufacturing and attesting it with our HSMs,
- Using it in the field, on both the security at rest and security at use aspects,

- Updating it in the field if needed.