TECHNICAL REPORT

ISO/TR 23576

First edition
2020-12

Blockchain and distributed ledger technologies — Security management of digital asset custodians

Reference number
ISO/TR 23576:2020(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 307, *Blockchain and distributed ledger technologies*.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

A digital asset custodian holds customers' digital assets for safekeeping in order to minimize the risk of their theft or loss. This document illustrates the security risks, threats, and measures which digital asset custodians consider, design, and implement in order to protect the assets of their customers, based on best practices, existing standards and research. For example, the management of signature keys for digital assets requires special attention, taking into account the specific nature of blockchains and DLT systems and the security challenges they face. A key topic discussed is the appropriate management of signature keys by digital asset custodians in order to prevent misuse and transactions by unauthorized individuals.

Blockchain and distributed ledger technologies — Security management of digital asset custodians

1 Scope

This document discusses the threats, risks, and controls related to:

- systems that provide digital asset custodian services and/or exchange services to their customers (consumers and businesses) and management of security when an incident occurs;
- asset information (including the signature key of the digital asset) that a custodian of digital assets manages.

This document is addressed to digital asset custodians that manage signature keys associated with digital asset accounts. In such a case, certain specific recommendations apply.

The following is out of scope of this document:

- core security controls of blockchain and DLT systems;
- business risks of digital asset custodians;
- segregation of customer's assets;
- governance and management issues.

2 Normative reference

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 22739, *Blockchain and distributed ledger technologies — Vocabulary*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 22739 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at https://www.iso.org/obp
- IEC Electropedia: available at http://www.electropedia.org/

3.1
digital asset custodian system
system that holds customers' digital assets for safekeeping in order to minimize the risk of their theft or loss

Note 1 to entry: In this document, holding assets is considered in a broad sense, as it includes for instance, the case of physically or digitally storing the assets, but also the case of holding the private keys associated with the assets, or even the case of protecting access to the assets, like holding one of the keys protecting the access to the assets.

3.2
cold wallet
cold storage
offline application or mechanism used to generate, manage, store, or use private and public keys

3.3
hot wallet
hot storage
online application or mechanism used to generate, manage, store, or use private and public keys

3.4
hardware wallet
wallet which leverages a hardware device (e.g. HSM) to generate, manage, store, or use private and public keys

3.5
deterministic wallet
wallet in which multiple key pairs are derived from a single starting point known as a seed

3.6
hierarchical deterministic wallet
deterministic wallet (3.5) in which child key pairs are derived from the master key pair

Note 1 to entry: Descendant key pairs can be derived from the child key pairs, in a hierarchical manner, hence the name of the wallet. Child key pairs can be used and shared without having to share the master key pair. It is defined within Reference [7].

4 Abbreviated terms

AML	anti-money laundering
API	application programming interface
APT	advanced persistent threat
CFT	countering financing of terrorism
DLT	distributed ledger technology
DNS	domain name system
FATF	Financial Action Task Force
FQDN	fully qualified domain name
HSM	hardware security module
SMS	short message service
ISMS	information security management system
ISP	internet service provide
KYC	know your customer
OS	operating system
OWASP	Open Web Application Security Project
PII	personally identifiable information

PKI	public key infrastructure
PoW	proof of work
TLS	transport layer security

5 Basic description of a model of online system for digital asset custodianship

5.1 General

In this clause, an example implementation for an online digital asset custodian system is presented. This model will then be used to explain the concepts and provisions in this document. However, it is also worth noting that other types of custodian systems exist. For example, decentralized exchanges (DEXs), have quite a different implementation compared to the one illustrated in Figure 1. Furthermore, protocols like atomic swap and multisignature can be considered a type of custodian as well. Therefore, although most of the content of this document applies to any kind of custodian, some of the risks and potential controls discussed may or may not apply to other types of custodian systems.

5.2 Example of a system for digital asset custodians and its functional components

Figure 1 shows an example model for a digital asset custodian system.

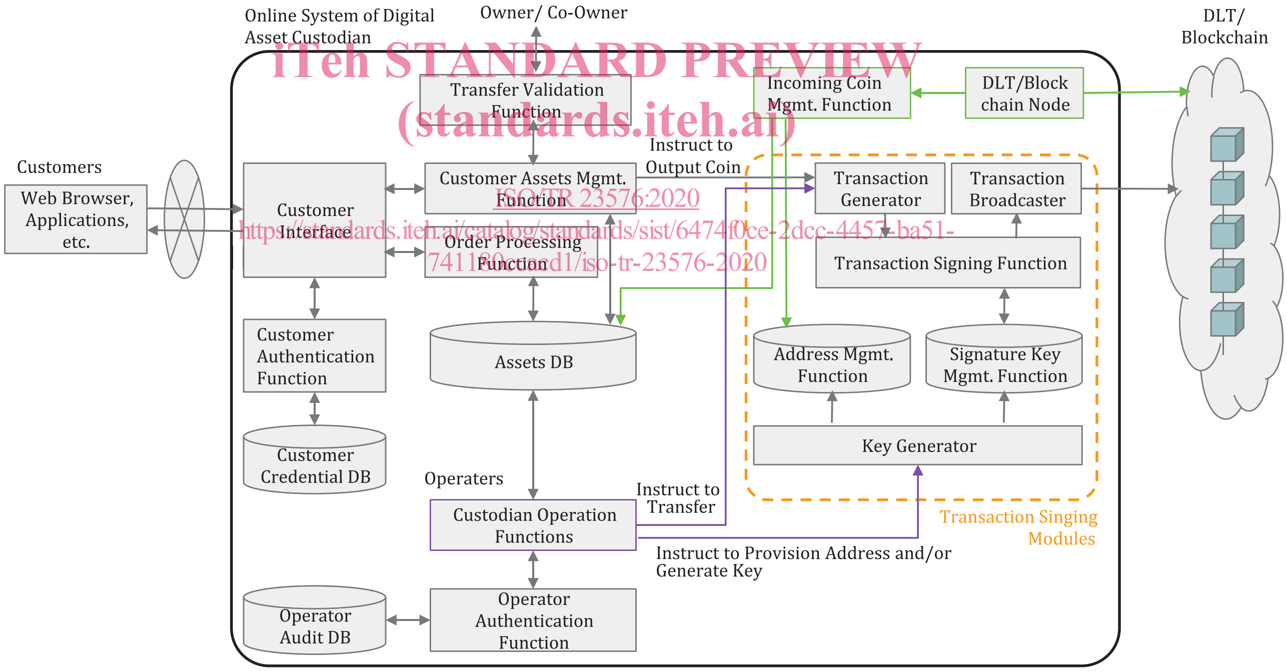
Online System of Digital Asset Custodian
Owner/ Co-Owner
DLT/ Blockchain
Transfer Validation Function
Incoming Coin Mgmt. Function
DLT/Block chain Node
Customers
Web Browser, Applications, etc.
Customer Interface
Customer Assets Mgmt. Function
Instruct to Output Coin
Transaction Generator
Transaction Broadcaster
Order Processing Function
Transaction Signing Function
Customer Authentication Function
Assets DB
Address Mgmt. Function
Signature Key Mgmt. Function
Customer Credential DB
Key Generator
Operaters
Instruct to Transfer
Custodian Operation Functions
Transaction Singing Modules
Instruct to Provision Address and/or Generate Key
Operator Audit DB
Operator Authentication Function

Figure 1 — Basic example model of a digital asset custodian

Table 1 — Functional components of a digital asset custodian

Functional components	Explanation
Interface (web application, APIs)	Provides screen and input functions such as login, account management (deposit/withdrawal) and trading for the customers (users). The most common interfaces are web applications, APIs, and mobile apps.
Customer authentication function	Performs user authentication for login purposes to the system.
Customer credential database	Manages required IDs for login and verification information related to user authentication process (e.g. password verification information).

Table 1 *(continued)*

Functional components		Explanation
Transfer validation function		Verifies the granted permission to proceed to the transfer of digital data or assets by the owner or co-owners when the transfer implies the validation of multiple parties. For example, the use of multisignatures schemes to validate an authorized outgoing transaction.
Customer assets management function		A group of functions which provide customer account management. For example, these functions perform deposits or withdrawals (output coins) and, more generally, other asset manipulation processes according to user instructions. The functions provided may refer to or update asset data.
DLT / Blockchain node		A node in a DLT / blockchain system, which communicates with its peers (i.e. other nodes).
Incoming transaction management function		Verifies transactions stored in DLT / blockchain to confirm whether incoming assets refer to the specified addresses. Updates the asset database according to the transaction retrieved from the DLT / blockchain.
Order processing function		A group of functions for the management of sales instructions from customers. The order processing function performs actions related to trading of digital assets. This function refers to and updates asset data.
Assets database		Manages the record of assets both for fiat currencies and digital assets. The asset database does not include the signature keys for signing transactions. These are managed separately from the assets for each customer.
Transaction signing modules	Transaction generator	Generates transactions to be sent to the DLT / blockchain based on instructions from the customer asset management function or the custodian's operation function.
	Transaction broadcaster	Sends the signed transaction to the DLT / blockchain. Transactions are broadcasted to blockchain nodes through network protocols.
	Transaction signing function	Generates digital signatures based on the instructed transaction contents using the relevant signature key (i.e. IDs and addresses).
	Address management function	Manages verification keys related to the signature keys, or to addresses (i.e. such as values calculated from the verification keys).
	Signature key management function	Manages the signature keys of the digital assets (i.e. the keys used for the signature of the transactions). Signature keys may be stored in a cold wallet as a security measure.
	Signature key generator	Generates signature keys. The generated keys are registered in the signature key management function, and the verification keys and addresses are registered in the address management function.
Custodian operation functions		A group of functions dedicated to the custodian's operators and/or administrators. Administrator and operators can instruct the module to perform function such as generating new signature keys or transfer digital assets.
Operator authentication function		Authenticates the operators and administrators of the system.
Operator audit database		Manages auditing data related to the authentication processes of operators and administrators for the system.

The functional components described in Table 1 are intended to logically distinguish the various functions within the system, and do not represent an actual architecture of such a system. As an example, in a real-world implementation, the address management function would probably be implemented using a database. Also, there are implementations in which multiple functional components are packaged together. All the functional components of the transaction signature system could be integrated within the customer asset management system, or they could be operating as a separate system. Many implementations of Bitcoin wallets provide all functions for the transaction signature system as a single atomic system. It is also possible to imagine some of these functions being provided by an external "subcontractor" system, such as a remote server.

5.3 Examples of transactions

— Fiat currency deposit

a) The customer sends fiat currency to the custodian's bank account.

b) The custodian confirms the reception of the fiat currency transfer and updates its assets database to reflect the customer’s asset status in relation to the transfer just received.

— Digital asset deposit

a) The customer transfers digital assets to an address specified by the custodian. The transfer is performed by using the customer’s digital assets wallet (i.e. other custodian or web/app wallet).

b) The custodian confirms that the digital assets have been transferred to the correct address and updates its assets database to reflect the customer’s asset status in relation to the transfer just received.

— Trading transaction

a) The customer accesses the interface made available by the custodian and instructs the system to perform some actions (e.g. trading).

b) The instructions to perform an action are received by the custodian and are processed by the custodian operations functions module. The result of the trade operations is processed by the custodian operations functions module which updates the asset database accordingly.

— Customer digital asset withdrawal

a) The customer accesses the interface made available by the custodian and instructs the system to transfer their digital assets to another address (i.e. output coins).

b) The instruction to output coins is processed by the customer assets management functions module. The transaction generator creates a transaction message based on the received instructions such as receiving address and the amount of digital assets to transfer.

c) The transaction message will be digitally signed by the transaction signing functions module.

d) The signed transaction message is delivered to all nodes on the DLT / blockchain by the transaction broadcaster module.

— Internal transfer by operator or administrator

a) The administrator or operator instructs the system to transfer digital assets to another address through the custodian operations functions module. For example, the digital assets may be sent between addresses managed within the custodian.

b) The instructions to output coins are then processed by the custodian operations functions module. The transaction generator creates a transaction message based on the received instructions such as receiving address and the amount of digital assets to transfer.

c) The transaction message will be digitally signed by the transaction signing functions module.

d) The signed transaction message is delivered to all nodes on the DLT / blockchain by the transaction broadcaster module.

5.4 Description of keys used for signature and encryption

5.4.1 Type of keys

Table 2 describes the different types of keys which can be used for signature and encryption within a digital asset custodian system.

Table 2 — Types of keys

Types	Description
Signature key	A signature key for signing transactions (for digital signature schemes standardized in ISO/IEC 9796 (all parts) and ISO/IEC 14888 (all parts))
Verification key	A public key for verification of transactions (for digital signature schemes standardized in ISO/IEC 9796 (all parts) and ISO/IEC 14888 (all parts)) It is common practice in public blockchains to calculate addresses as unique values derived from the verification key. In private DLT systems / blockchains this may not be necessary
Encryption/decryption key for signature key	Secret key (symmetric key cryptography) used to keep signature key confidential / protected
Master seed	A seed to generate a signature key in a deterministic wallet

5.4.2 Flow for key generation and key usage

Figure 2 shows a typical lifecycle for the different type of keys described in Table 2.

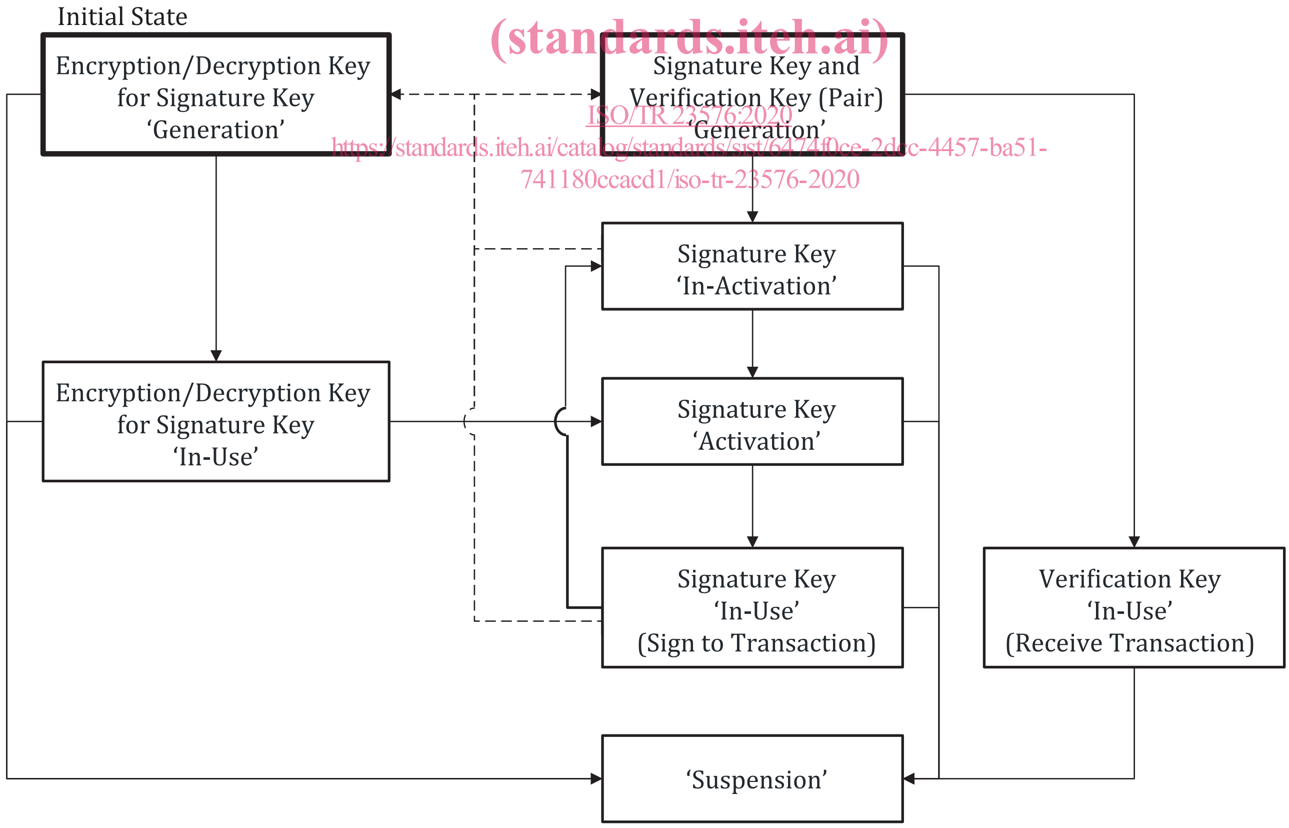
Initial State
Encryption/Decryption Key for Signature Key 'Generation'
Signature Key and Verification Key (Pair) 'Generation'
Signature Key 'In-Activation'
Encryption/Decryption Key for Signature Key 'In-Use'
Signature Key 'Activation'
Signature Key 'In-Use' (Sign to Transaction)
Verification Key 'In-Use' (Receive Transaction)
'Suspension'

Figure 2 — Lifecycle of signature key, verification key and encryption/decryption key for signature key

After a pair of keys (signature and verification, hereafter "key pair") is generated, an address, which will be used to receive transactions, is derived from the verification key. A sender will only need this address to be able to transfer one or more assets to it.

A signature key is considered inactive, when it is stored in a manner in which it cannot directly be used to sign (i.e. if it is encrypted). As an example, within the key management function module in Figure 1, a signature key could be encrypted using a pass phrase, rendering it inactive. Decrypting the signature key will return the key in an active state.

In the example model presented in Figure 1, the activation of a key is assumed to be executed within the transaction signing function module. Activation and deactivation of keys are standard functions provided by most wallets. The signature key is only needed when a transaction needs to be signed. Therefore, these can be stored offline for increased security, until needed. On the other hand, verification keys and addresses are stored online as they are needed more often for verification purposes.

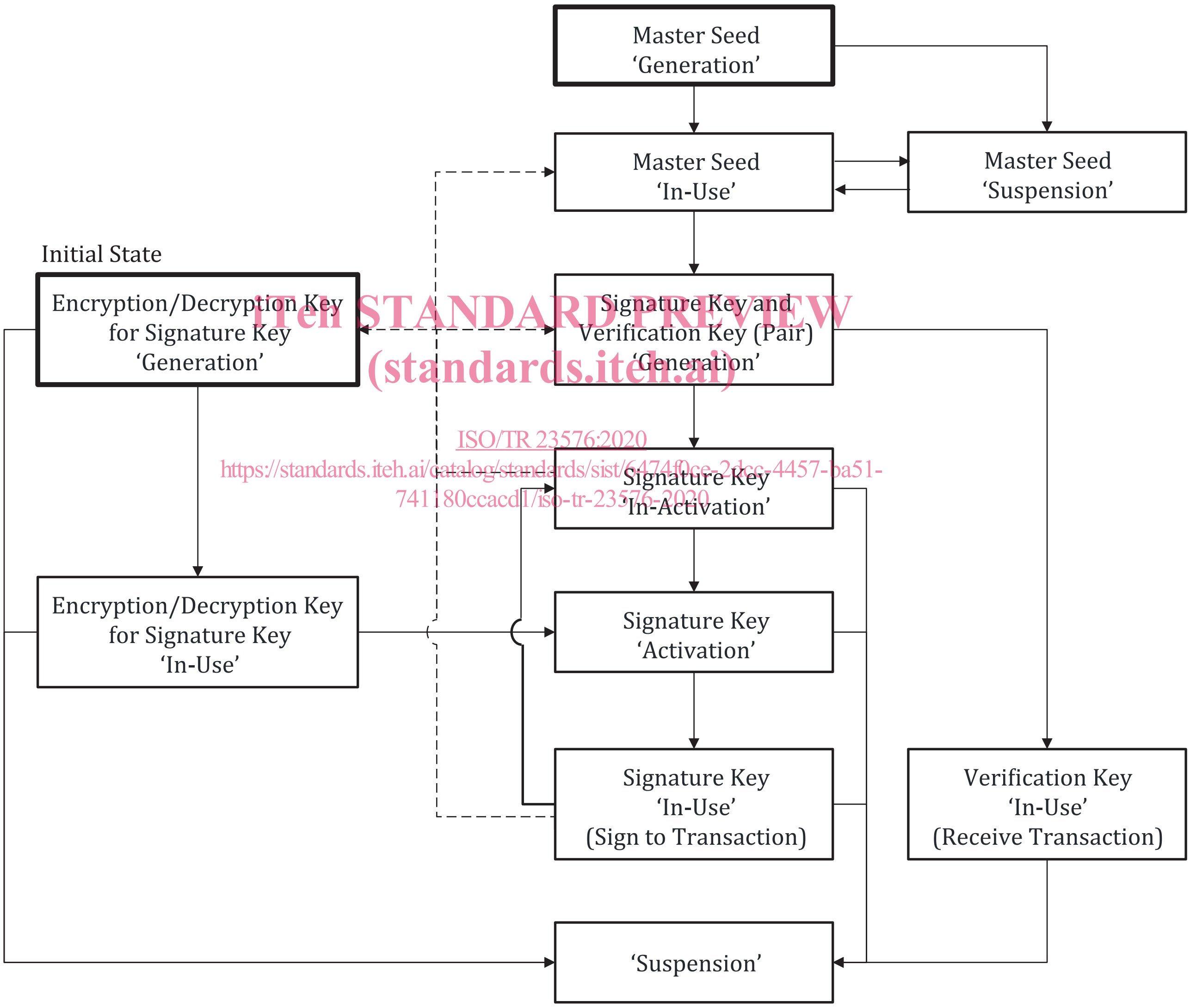
Master Seed 'Generation'
Master Seed 'In-Use'
Master Seed 'Suspension'
Initial State
Encryption/Decryption Key for Signature Key 'Generation'
Signature Key and Verification Key (Pair) 'Generation'
Signature Key 'In-Activation'
Encryption/Decryption Key for Signature Key 'In-Use'
Signature Key 'Activation'
Signature Key 'In-Use' (Sign to Transaction)
Verification Key 'In-Use' (Receive Transaction)
'Suspension'

Figure 3 — Lifecycle of the signature key, verification key and encryption/decryption key for signature key in a deterministic wallet

A deterministic wallet uses a mechanism by which after generating one master seed, multiple signature key pairs are derived from it. Figure 3 shows the lifecycle of the different types of keys within a deterministic wallet. On the one hand, by backing up and restoring the master seed, all derived signature key pairs can be recalculated. On the other hand, if the master seed is compromised (i.e. stolen), all crypto assets which are managed by any of the derived keys (and related addresses) may