ITUPublications

International Telecommunication Union
Standardization Sector

ITU-T Technical Report

(04/2025)

TR.dw-lasf

Digital wallet landscape analysis and security features

Technical Report ITU-T TR.dw-lasf

Digital wallet landscape analysis and security features

Summary

The diverse array of services called "digital wallets" has led stakeholders to have different understandings of them, creating obstacles in their implementation, adoption, and secure usage. This Technical Report aims to clarify the concepts and essential requirement of features for secure digital wallets by analysing various use cases in the market, such as blockchain wallets, identity wallets, payment wallets, central bank digital currency (CBDC) wallets, digital keys, etc. It also intends to provide the components of a platform that securely supports diverse digital wallets.

Keywords

Blockchain wallets, central bank digital currency (CBDC) wallets, digital wallet, identity wallets, payment wallets.

Note

This is an informative ITU-T publication. Mandatory provisions, such as those found in ITU-T Recommendations, are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Table of Contents

Technical Report ITU-T TR.dw-lasf

Digital wallet landscape analysis and security features

1 Scope

This Technical Report analyses blockchain wallets, identity wallets, payment wallets, central bank digital currency (CBDC) wallets, and other types of digital wallets designed for specific services, to clarify the concepts, functionalities, and security features of a secure platform required for diverse digital wallets.

2 References

[ITU-T X.1400] Recommendation ITU-T X.1400 (2020), *Terms and definitions for distributed ledger technology.*

[ISO/IEC 18013-5] ISO/IEC 18013-5:2021, *Personal Identification – ISO-Compliant Driver Licenses – Part 5: Mobile Driver Licenses (mDL) Application Standard*.

[ISO/IEC 19790] ISO/IEC 19790:2025, *Information security, cybersecurity and privacy protection – Security requirements for cryptographic modules*.

[ISO TR 23576] ISO 23576:2020, *Blockchain and distributed ledger technologies – Security management of digital asset custodians*.

[W3C CH API 1.0] W3C, Credential Handler API 1.0 (2021), *Draft Community Group Report 23 June 2021*.

[W3C DIDs 1.0] W3C, Decentralized Identifiers (DIDs) v1.0 (2022), *W3C Recommendation 19 July 2022*.

[W3C VC DM 2.0] W3C, Verifiable Credentials Data Model v2.0 (2025), *W3C Proposed Recommendation 20 March 2025*.

[W3C VC API 3.0] W3C, Verifiable Credentials API v0.3 (2025), *Draft Community Group Report 06 April 2025*.

3 Definitions

3.1 Terms defined elsewhere

This Technical Report uses the following terms defined elsewhere:

3.1.1 cold wallet [ISO TR 23576]: Offline application or mechanism used to generate, manage, store, or use private and public keys.

3.1.2 cryptocurrency [b-ISO 22739]: Crypto asset designed to work as a medium of value exchange.

3.1.3 deterministic wallet [ISO TR 23576]: Wallet in which multiple key pairs are derived from a single starting point known as a seed.

3.1.4 hardware wallet [ISO TR 23576]: Wallet which leverages a hardware device (e.g., HSM) to generate, manage, store, or use private and public keys.

3.1.5 hierarchical deterministic wallet [ISO TR 23576]: Deterministic wallet in which child key pairs are derived from the master key pair.

3.1.6 hot wallet [ISO TR 23576]: Online application or mechanism used to generate, manage, store, or use private and public keys.

3.1.7 verifiable credential [b-W3C Universal wallet 2020]: A data model for conveying claims made by an issuer about a subject.

NOTE – A verifiable credential is a tamper-evident credential that has authorship that can be cryptographically verified. Verifiable credentials can be used to build verifiable presentations, which can also be cryptographically verified.

3.1.8 verifiable presentation [b-W3C VCDM]: A tamper-evident presentation encoded in such a way that authorship of the data can be trusted after a process of cryptographic verification.

NOTE – Certain types of verifiable presentations might contain data that is synthesized from, but do not contain, the original verifiable credentials (for example, zero-knowledge proofs)

3.2 Terms defined in this Technical Report

This Technical Report defines the following terms:

3.2.1 binder wallet: A digital wallet that can install and manage several application-specific digital wallet modules and provide common features for them.

3.2.2 digital wallet: A set of features to provide various functions of physical wallets in digital world.

NOTE – A digital wallet that support only single type of application is referred to as application-specific wallet.

4 Abbreviations and acronyms

This Technical Report uses the following abbreviations and acronyms:

API	Application Programming Interface
ARM	Advanced RISC Machine
BLE	Bluetooth Low Energy
CBDC	Central Bank Digital Currency
CVV	Card Verification Value
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
DTLS	Datagram Transport Layer Security
eIDAS	electronic Identification, Authentication and trust Services
EU	European Union
FOB	Free On Board
gRPC	Google Remote Procedure Call
HD wallet	Hierarchical Deterministic wallet
HSM	Hardware Security Module
IC	Integrated Circuit
ID	Identifier
MST	Magnetic Secure Transmission
NFC	Near Field Communication
NFT	Non-Fungible Token
OS	Operating System

PMF	Protected Management Frame
POS	Point of Sale
PWA	Progressive Web Application
(Q)EAA	Qualified Electronic Attestation of Attributes
(Q)TSP	Qualified Trust Service Provider
REE	Rich Execution Environment
SE	Secure Environment
SSL	Secure Sockets Layer
ST	Security Threat
TEE	Trusted Execution Environment
TLS	Transport Layer Security
UWB	Ultra-Wide Band
VC	Verifiable Credential
VP	Verifiable Presentation
VPN	Virtual Private Network

5 Digital wallet landscape

5.1 Introduction

Digital wallets can be classified based on several criteria. First, they can be categorized by application field into blockchain wallets, identity wallets, payment card wallets, central bank digital currency (CBDC) wallets and other services. Other services may encompass functionalities like payments and access control, among others. Depending on the implementation method, wallets can be divided into software wallets and hardware wallets, as well as individual instance wallets and cloud wallets. Furthermore, based on ownership and control, they can be classified as self-custody and third-party custodial methods.

Table 1 categorizes different types of digital wallets based on their primary purpose, the critical data they handle, their main functions, and the trust anchors that support them.

A blockchain wallet is primarily used for cryptocurrency transactions. It manages cryptographic keys and facilitates transaction signing and blockchain inquiries. The key transaction data in this type of wallet comprise transaction details recorded on the blockchain, which serves as the trust anchor.

An ID wallet is designed for digital identification and verification. It stores essential identity-related data such as cryptographic keys, verifiable credentials (VCs), and verifiable presentations (VPs). Its main function is to request and verify VCs, as well as generate and submit VPs. The integrity of this system relies on a VC issuer as the trust anchor.

A payment wallet is used for processing payments and securely storing card information along with cryptographic keys. It generates payment tokens and facilitates transaction requests. The trust anchors for payment wallets include card terminals and payment tokens that ensure secure transactions.

A CBDC (central bank digital currency) wallet is specifically designed for retail transactions. It maintains cryptographic keys and balance information to manage payments and balance tracking. This wallet supports digital currency transfers and transaction processing, with the central bank server acting as the trust anchor to ensure the reliability and authenticity of transactions.

Other services category includes wallets used for access control and other miscellaneous functions. These wallets manage cryptographic keys and are responsible for providing proof of rights and temporal proofs. The critical transaction data consist of tokens that verify ownership or access, and the system relies on a management server and a service terminal as trust anchors.

Table 1 – Types and characteristics of digital wallets

Type	Primary purpose	Critical data	Main function	Critical transaction data	Trust anchor
Blockchain wallet	Cryptocurrency transaction	Key	Transaction signing and request, blockchain inquiry	Transaction information	Transactions Record on blockchain
ID wallet	Identification	Key, VC, VP	VC request/verification, VP Generation/submission	VC, VP	VC issuer
Payment wallet	Payment	Key, card information	Payment token generation	Transaction request information, payment token	Card terminal, payment token
CBDC wallet	Retail transactions	Key, balance information	Payment signature, balance management	Currency transfer, transaction information	Central bank server
Other services	Access control, etc.	Key	Evidence of rights, temporal proof	Token of rights	Management server and service terminal

5.2 Types of digital wallets by application fields

5.2.1 Blockchain wallet

Blockchain-based cryptocurrencies utilize hashed addresses derived from users' public keys as account identifiers. Transactions are generated by signing transaction records using the corresponding private key. Users are tasked with generating their key pairs, securely storing their private keys, sharing their addresses with transaction counterparts, signing transactions, and requesting nodes to record these on the blockchain.

At its essence, the blockchain wallet centres around managing asymmetric keys and signature functionalities. Cryptographic key management, a well-developed area in security, has prompted recent blockchain wallets to integrate diverse key recovery methods and other key management features to ensure key safety, addressing concerns such as loss or compromise.

5.2.2 ID wallet

In the concept of decentralized identity management, unlike traditional identity systems, a user's identity is not managed by a centralized identity service provider. Instead, users create their own decentralized identifiers (DIDs) and store them in trusted repositories like blockchains. They request certification of attributes related to these DIDs from identity verification issuers to obtain verifiable credentials (VCs). Users then utilize these VCs to present a verifiable presentation (VP) when requested by verifiers.

Hence, the core of an identity wallet requires additional functionalities beyond traditional key management. It involves requesting, storing VCs, generating VPs using these VCs, and validating them. In contrast to blockchain wallets, issued VCs are not stored on public servers but shared only among the issuer, holder, and verifier. Consequently, for identity wallets, a crucial function is the storage repository for the acquired VCs.

5.2.3 Other services

Traditional banking and card payment services were once referred to as "payments," but there's a growing trend of integration with the term "digital wallets." These services are gradually expanding to include offerings like reward cards, digital tickets, digital keys, and more. They are utilized for payment, physical access, or authorization purposes.

One notably important new service is the digital wallet designed for central bank digital currency (CBDC). These services are configured in various forms depending on the service or provider. A key function of the digital wallet is to safeguard information related to these services from unauthorized access. Additionally, it must provide functionalities to authenticate external devices that offer services based on this information and securely communicate with them. Encryption techniques are employed for this purpose, and key management alongside encryption modules constitutes the core functions of the digital wallet.

5.3 Types of digital wallets according to implementation methods

5.3.1 Software wallet and hardware wallet

A software wallet is a program that operates on a computer or smartphone, implementing the functions of a wallet. While easily accessible through program installation, software wallets are susceptible not only to software vulnerabilities but also to vulnerabilities in computers or smartphones. For instance, during encryption, keys may be stored and used in memory, making them vulnerable to memory hacks. Additionally, computer-based wallets are constrained by mobility. A software certificate module running on a PC is also considered a software wallet. Typically, password-protected software certificate modules are vulnerable to certificate file copying and brute-force attacks.

A hardware wallet implements wallet functions using a separate physical device. By performing encryption key processes within dedicated hardware, it can protect keys and encryption procedures from issues such as memory hacking. Generally, hardware wallets provide a higher level of security than software wallets, but there's a risk of theft or loss of the physical device.

Integrated circuit (IC) chip cards following international standards, utilize asymmetric key encryption within the chip to encrypt embedded card data. Furthermore, they generate unique codes for each transaction, safeguarding transaction data and preventing reuse. IC card terminals authenticate users via PIN input and activate private keys. Hence, while not commonly referred to as such, IC chip cards can fall under the category of hardware digital wallets.

Devices known as hardware digital wallets come in smartcard form but are commonly operated by connecting via USB to a PC or smartphone. Some devices employ alternative connections like secure Bluetooth. They may incorporate PIN pads or fingerprint recognition for user authentication. The [ISO/IEC 19790] standardizes requirements across 11 areas into four security assurance levels, even for hardware implementations. Therefore, the security level can vary based on the implemented security functionalities, even if it is implemented as hardware.

5.3.2 Hot wallet and cold wallet

When blockchain-based wallets are connected online, the risk of hacking incidents increases. Therefore, they are classified as 'hot wallets' when constantly connected online and 'cold wallets' when used as long-term storage without being connected to the network.

While software wallets are often considered hot wallets and hardware wallets as cold wallets due to the higher risk posed by network connectivity, this is not always the case. Recent software wallets provide a locking feature. If the programme is not running and the data is stored in an encrypted state, it cannot be classified as a hot wallet. In terms of security, considering hot wallets and software wallets as lower in security than cold wallets and hardware wallets is a matter of degree, not an absolute concept. Rather, there is a need to mitigate the risks of hot wallets by using hardware wallets to lower the risks. For instance, servers conducting continuous signature verification online could use hardware security modules (HSMs) to enhance encryption performance and key security.

A cold wallet is used as a repository for relatively long-term preservation of highly sensitive information, such as signature keys for controlling large amounts of cryptocurrency or root CA signature keys. While a ruggedized notebook-shaped hardware wallet is a prominent example of a cold wallet, physical means such as recording private keys on paper or metal plates are also considered part of the cold wallet classification.

5.3.3 Mobile wallet and cloud wallet

In terms of convenience and mobility, smartphones, which have already been popularized, emerge as the most suitable means to store and operate digital wallets. Particularly, mobile wallets can utilize various interfaces embedded in smartphones, such as near field communication (NFC), Bluetooth, and the camera. Hardware manufacturers can provide additional physical security features, enabling the provision of hybrid-wallet functionality utilizing both software and firmware.

Software wallets, hardware wallets, and mobile wallets exist as individual instances that users possess and control. In contrast, cloud wallets are software services hosted on the internet, accessible via HTTP or Google remote procedure call (gRPC). Cloud wallets have security advantages through centralization in terms of logs and dynamic scaling. Moreover, application programming interface (API)-based messages can be easily integrated into existing systems. cloud wallets can be accessed through a browser on PCs and via progressive web apps (PWA) on mobile devices.

Recently, there have been companies offering subscription-based cloud-based cryptographic services utilizing HSMs.

5.3.4 Classification of digital wallets based on control rights

Until now, the wallets discussed have been of a form where users hold ownership. Even in the case of cloud-based wallets, where key pairs and related data are stored in the cloud, users retain ownership and control of that wallet. However, many exchanges directly manage users' keys, grant users an account, and enable transactions through account authentication. In this scenario, users do not have direct authority over the keys. These wallets are referred to as custody wallets. There's a risk of mass key exposure due to the security management of exchanges in such cases. Most blockchain-related hacking incidents stem from vulnerabilities in the key management of exchanges. To mitigate this, a control method is added by storing a bulk of coins in cold wallets protected by a dual-key system and only keeping a portion in hot wallets for transactions.

[ISO TR 23576] covers extensively the management and security of delegated keys, as seen in exchanges.

6 Functional components of digital wallets

6.1 Blockchain wallet

Blockchain wallets can be implemented with software blockchain wallets and hardware wallets.

6.1.1 Software blockchain wallet

A typical blockchain digital wallet, such as [b-Meta] consists of modules for key and address generation, key storage, key recovery, transaction signing, state management, and external interfaces, as shown in Figure 1.

The key and address generation module creates a pair of private and public keys and generate addresses using the public key. The key storage module manages the storage of these generated keys. The key recovery module restores necessary keys. The transaction signing module constructs and signs transaction information. State management oversees information within the wallet itself, connected blockchains, and additionally stores coin balances or ownership related non-fungible token (NFT) information.

External interfaces communicate with blockchains, hosts and other applications. They transmit transactions to the blockchain or retrieve information from it. Furthermore, they provide information to users via graphical interfaces operated by the host running the digital wallet, allowing users to input necessary data. These interfaces also facilitate the exchange of information with other diverse applications.

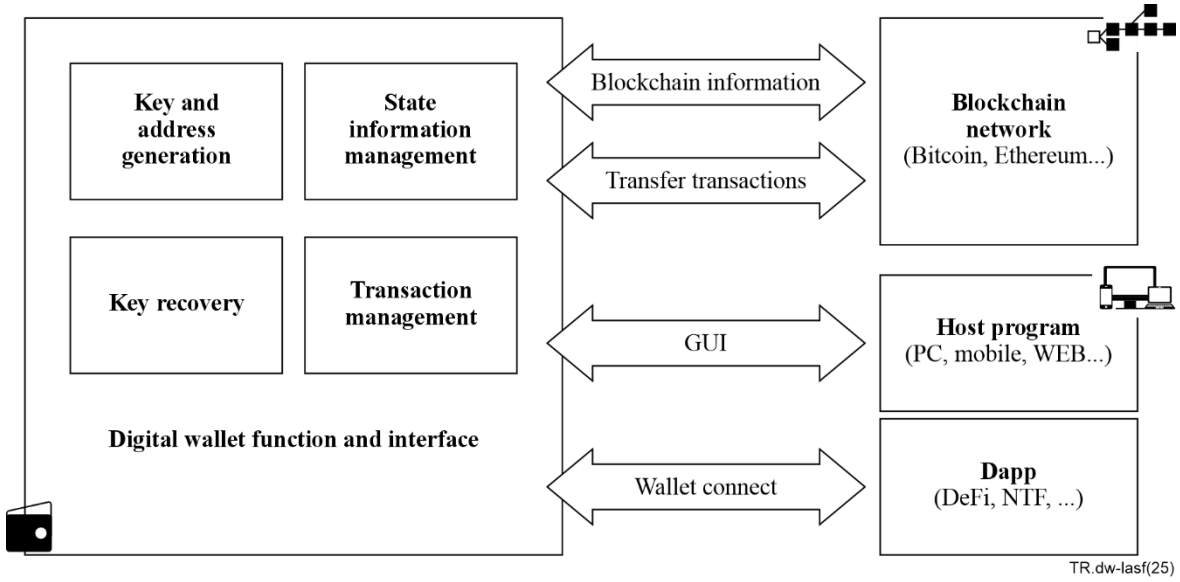

Figure 1 – The structure of a software blockchain wallet

6.1.2 Hardware blockchain wallet

A hardware blockchain wallet, as shown in Figure 2, is a physically secured implementation of a blockchain digital wallet on separate hardware, designed to enhance security. Typically, hardware wallets operate independently without direct Internet connectivity, instead communicating with PCs, smartphones, tablets and other devices via USB or Bluetooth through dedicated applications.

In essence, modules for key and address generation, key storage, key recovery, transaction signing, state management, and interfaces with hosts are built internally within the hardware. These interfaces connect to clients on the host via USB, Bluetooth or physical card interfaces. The client interacts with the hardware wallet, providing a graphical user interface for users and facilitating interfaces with blockchains.

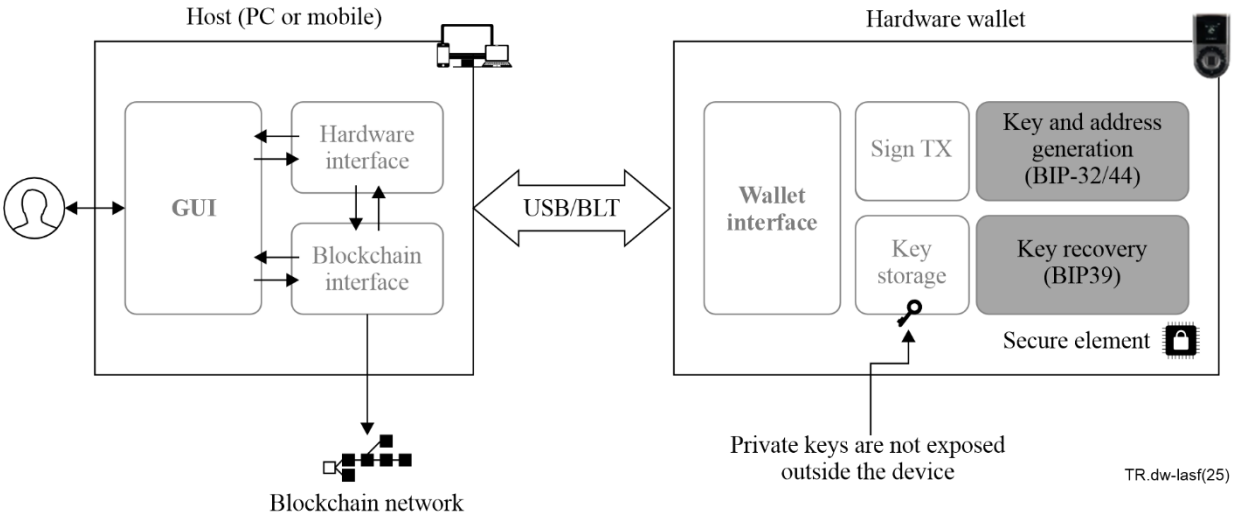

Figure 2 – A structure of a hardware blockchain wallet

6.2 ID wallet

An ID wallet comprises key management and cryptographic modules, a management module, a data repository for verifiable credentials (VC) and verifiable presentations (VP), and external interfaces.

The key management and cryptographic module generates, stores, and manages key pairs for decentralized identifiers (DID). It also provides cryptographic functions for requesting and verifying VCs and generating and submitting VPs. The data repository stores and manages relevant data such as user verification data, user identifiers, user attributes, issued and generated VCs and VPs, trusted issuers, verifier lists, usage history, and other related information.

The VP generation module creates VPs requested by trusted parties from stored VCs, as per user choice. External interfaces are necessary for interactions with verifiers and verifiable credential issuers. These interfaces can be facilitated through browsers or physical interfaces supporting both online or offline interactions.

6.3 Payment wallet

Some existing digital financial transaction methods fall within the conceptual scope of digital wallets, but typically, when using the term 'digital wallet,' it refers to an app operating on a smartphone such as [b-AW], [b-GW] and [b-SW].

A smartphone digital payment app communicates with traditional point-of-sale (POS) terminals using near field communication (NFC) or magnetic secure transmission (MST) interfaces provided by the smartphone. These apps may include functionalities for key management and encryption modules in software or leverage security features offered by smartphone manufacturers through collaboration.

For enhanced security, if smartphone manufacturers provide trusted execution environment (TEE) features, services can be developed in the form of regular apps registered with a trusted app. Additionally, independent trusted apps can be developed using trusted execution environment APIs. In other words, one can register and use reinforced security-enabled digital wallet apps provided by phone manufacturers or develop digital wallets utilizing the phone manufacturer's security environment. Without a secure environment, direct access to the phone's interfaces can perform payment functions, but this may lead to security vulnerabilities due to threats like operating system (OS) rooting.

6.4 CBDC wallet

Recently, countries worldwide are preparing for the issuance of central bank digital currencies (CBDCs) by their respective central banks. CBDCs are categorized into wholesale CBDCs, retail

CBDCs, and cross-border CBDCs, each implemented differently. Here, we focus on the most common, retail CBDC.

CBDC serves as a centrally managed digital currency and does not necessarily require the use of blockchain, but many countries consider blockchain-based CBDCs to enhance transaction record integrity. Unlike physical currency, CBDC records are centrally managed and typically do not ensure anonymity. While anonymity is not guaranteed, the advantage lies in minimizing financial losses through identity verification and reissuance in cases of loss or damage.

Users of CBDC create digital wallets and store their bank balances in the form of digital currency or transfer them to their accounts. Digital currency stored in the digital wallet can be used for both online and offline payments. Offline payments can be conducted similarly to digital payments or through agreed-upon amounts between digital wallets, necessitating payment request and reception functionalities.

CBDCs, operated under the central bank's management in accordance with national laws, need to comply with regulations such as financial transparency, prevention of illegal property, money laundering, and prohibition of funding for threatening activities. Hence, apart from typical key and encryption modules, storage and management of various additional user information, accounts, balances, and monitoring functionalities to curb illicit transactions are necessary. Operational policies akin to traditional banks, such as limitations on usage and payment amounts, may be imposed. CBDCs aiming to emulate physical currency also anticipate the ability to transact for a certain duration even during network failures, leading to the inclusion of offline payment capabilities.

CBDC wallets, handling diverse functionalities and information, necessitate robust security. Hence, secure digital wallet software operating on a secure operating system (OS) providing tamper resistance based on hardware chips for safe encryption and key storage becomes imperative.

6.5 Vehicle key wallet

A digital key uses NFC, Bluetooth low energy (BLE), or ultra-wide band (UWB) to unlock doors of vehicles or buildings within a proximity.

A digital car key involves installing the vehicle manufacturer's app on a smartphone, registering car information, and physically being present inside the vehicle to register it within the app. During this registration process, an encryption key is generated and connected to the manufacturer's server.

After registration, the smartphone's NFC capability allows for unlocking car doors and starting the vehicle via the smartphone holder or locking the doors. For foreign vehicles, it may also allow the setting of driving modes. The owner of the digital key can share access to the vehicle's digital key with family or others and revoke shared access permissions.

Access through digital keys using NFC can open doors and, upon automatic closure or leaving a designated area, can be set to lock the doors. Similar to vehicle digital keys, the owner can grant access permissions to visitors, set access periods, or revoke access permissions. While NFC chips/tags currently provide encryption using pre-shared secret keys, digital wallet apps offering such functionality protect access using asymmetric-key encryption.

Access management systems utilizing digital keys can be implemented in hotels, companies, schools, or multi-family homes. Through NFC communication, tasks such as verifying entrants, specifying entry zones and periods, and maintaining entry logs can be conducted.

6.6 Door key wallet

Wristbands utilizing free on board (FOB) tags or barcode prints, commonly used in amusement parks, resorts, or cruises, interact with readers to provide access management services by presenting stored information in NFC memory or a barcode. The information provided by the reader is compared with server-provided data to facilitate access. Additionally, if prepayment is involved, payment

calculations are based on the reader-provided details, and are settled upon service completion. These functionalities can be implemented in a safer and user-controlled manner through a smartphone's digital wallet app.

Depending on the design and structure of the digital wallet app, functionalities such as hotel reservations and payments, access via digital keys, and reward accumulation can be performed either through separate apps or unified within a single hotel-digital-wallet app.

Event tickets for movies, shows, or games can also be securely stored, gifted, archived, and integrated with reward systems using digital wallet features. This setup can vary depending on the business collaboration between service providers and digital wallet app providers.

7 Common requirements on digital wallets

7.1 General requirements for digital wallets

For a digital wallet to be widely used, it should provide the functionality of a physical wallet. A physical wallet can hold some cash, credit or debit cards, ID cards, sometimes receipts, loyalty cards, flying or movie tickets, and family pictures, and it could have key rings that hold car or house keys. On the other hand, many digital wallet providers focus on a single-purpose, application-specific wallet of their own. This makes sense, because each wallet is merely a client application for a user that needs to communicate with a specific service network in order to use the service it is intended to provide. However, people would not want to bring several single-purpose digital wallets. They want to put everything in one place and carry it with them everywhere. Of course, it should be easy to use, and as secure as a physical wallet.

In the real market, many companies provide wallets that can install various wallets. These wallets use the hardware and/or OS with enhanced security, to provide a secure wallet application, which can install other single-function wallets. This "wallet of wallets" provides some benefits that can meet user needs. For simplicity, let's refer to this as a binder wallet, to distinguish a multi-purpose wallet that supports several services.

7.1.1 Mobility and portability

A digital wallet should be able to be carried everywhere for convenience. If an application-specific digital wallet is stored on dedicated hardware, this is not always true. But if a digital wallet is an app on a mobile phone, it can even be taken to the toilet and used for shopping there. It also should have strong portability. Everything can be moved from an old physical wallet to the brand-new one. Some digital wallets allow users to install a similar wallet that can continue working on a new device, while others require users to back up their data to recover their assets. The W3C credential community group has developed an abstract data model and interface for a portable, extensible universal wallet to support digital currencies and credentials.

7.1.2 Versatility

A binder wallet should support various types of application-specific wallets, including those already mentioned, and be extensible to support new types of wallets. To do this, it should be able to install new wallet apps, manage them, and support various interfaces that the applications need to use in order to communicate with their service-dedicated networks and systems.

Some payment services should communicate with traditional card readers, barcode readers, and NFC. New applications communicate with Bluetooth, Wi-Fi, and various types of mobile telecommunication networks, like CDMA, LTE, 5G, or 6G, especially while in motion. They might use less popular methods like IR, satellite, or any other P2P communications.

They should also communicate with users to interact and respond correctly and, above all, authenticate the authorized user. Therefore, they should support various types of communication and human interfaces.

7.1.3 Security and privacy protection

Each digital wallet handles sensitive and important data. Each digital wallet has its own unique data and protocols that must be protected for each function, and each wallet app must have the security and privacy protection features necessary for securing the service.

The security of digital wallets largely depends on cryptographic algorithms and key management. Cryptography is used to ensure the confidentiality and integrity of data, but it can also be used to establish tamper-proof features within the wallet itself and establish trusted channels.

As complete tamper-proofing is practically impossible, digital wallets are required to be tamper-resistant or at least tamper-evident. Certified HSMs have clear advantages over software cryptographic modules in terms of performance and security, especially in tamper resistance.

Trusted channels provide the confidence needed in communications. They provide not only for the confidentiality and integrity of data in transfer, but also the identification and authentication of communicating entities. Trusted channels for user authentication are often referred to as trusted paths and encompass both physical and logical aspects of secure communication.

Meanwhile, a binder wallet, which can include multiple single-purpose digital wallets, needs additional security features to securely store, execute, and manage each of these wallet apps.

Even if application-specific digital wallets have the security features required for the services they support, when they run on the same OS and memory simultaneously, they should be appropriately separated. Also, when they use the interfaces on the same device, trusted channels (secure channels and secure authentication methods) should be provided for each wallet process.

Up to now, each application-specific wallet has created and used its own asymmetric key pairs for authentication and secure execution of its functionality. However, future convergence services can be provided through the cooperation of different applications. Accordingly, security and privacy protection should also be strengthened.

A binder wallet with hardware-based enhanced security can provide this feature efficiently, but application-specific wallets may become reliant on platform functionality that they cannot fully control.

Each digital wallet handles sensitive and important data. Each digital wallet has its own unique data and protocols that must be protected for each function, and each wallet app must have the security and privacy protection features necessary for securing the service. Meanwhile, a binder wallet that can include multiple single-purpose digital wallets needs additional security features to securely store, execute, and manage each of these wallet apps.

Even if application-specific digital wallets have sufficient security features required for the specific services they support, if they are run on the same OS and memory simultaneously, they should be properly separated. Also, when they use the interfaces on the same device, secure channels and secure authentication methods should be provided for each wallet process.

The security of digital wallets largely depends on cryptographic algorithms and key management. It is desirable to use certified HSMs for sufficient performance and security. A binder wallet with hardware-based enhanced security can provide this feature efficiently, but the application-specific wallets may become reliant on functionality that they cannot fully control.

Up to now, each application-specific wallet creates and uses its own asymmetric key pairs for authentication and secure execution of its functionalities. However, new convergence services may be provided by the cooperation of different applications in the future. Accordingly, security and privacy protection should also be strengthened.

7.2 Tentative conclusion: a mobile platform with enhanced security

Based on what has been discussed in clause 7.1 so far, a hardware-based security-enhanced mobile platform can be considered as the most realistic means of providing the required characteristics. It can support various application-specific digital wallets with mobility, convenience of use, and security. The mobile platform can be used for a single digital wallet, or for a binder wallet managing multiple application-specific wallets. Since the common functionality required in the interactions between the platform and the wallets are fully covered in the case of a binder wallet, the discussion below will be focused on the latter.

8 Security threats (STs) to digital wallets

ST1 Vulnerable cryptographic algorithms and key management

ST2 Private key loss, theft, exposure, and destruction

e.g., key exposure due to accumulated reuse, conspiracy of custodians.

ST3 Malicious code

ST4 Sensitive or important information exposure or modification

e.g., user's personal data, such as name and address, card number, card verification value (CVV), consumption patterns, user identity, bank account and balance, digital currency balance, and unauthorized modification of security policies set by authorities such as the Central Bank.

ST5 Transaction data modification

e.g., price, delivery address, discount rate, number of instalment months, purchase items, receiver address, authentication reuse.

ST6 Vulnerable applications and smart contracts

ST7 Application tampering

ST8 Memory hacking

ST9 OS rooting

ST10 Manipulation of VC or VP

ST11 Device loss, theft, destruction and change

ST12 Inactivation or bypass of user authentication on device

ST13 Wrong source of download/ update

ST14 Human interface channel hooking

ST15 Sniffing, tampering, spoofing, jamming, disconnection of telecommunication channels

e.g., fake terminal, fake QR codes or NFC tags, unprotected Wi-Fi, vulnerable Bluetooth protocol, signal relaying, man-in-the-middle.

ST16 Network disconnection or battery drain

e.g., functional inability or off-line double spending.

ST17 Lack of interoperability or standardization

e.g., unreadable VC or VP.

ST18 Inappropriate data sharing

e.g., car, house or office key sharing with wrong person or temporal ones without limit.

ST19 Lack of user awareness

ST20 Social engineering

9 Security features for digital wallets

9.1 Security features for digital wallets by application area

9.1.1 Security features for payment wallets

1) Overview of payment wallet

Among various electronic financial transaction methods, access using asymmetric keys like certificates can be included within the concept of digital wallets. However, when the term "digital wallet" is generally used, it refers to an app operating on a smartphone.

Smartphone electronic payment apps communicate with existing merchant point of sale (POS) terminals using the NFC or MST interface provided by the smartphone. These apps can include key management and encryption modules in software or use the security features provided by the smartphone in collaboration with the manufacturer.

To enhance security, if the smartphone manufacturer provides a trusted execution environment (TEE) feature, services can be developed as additional modules registered and used within the manufacturer's digital wallet app. Alternatively, a proprietary digital wallet app can be developed using the TEE API. In other words, users can either register and use the manufacturer-provided secure digital wallet apps or develop digital wallets that utilize the security environment provided by the smartphone manufacturer.

In the case of a general smartphone without a security environment, electronic payment apps can directly access the smartphone's NFC interface to perform payment functions. However, the security may be vulnerable to threats like OS rooting.

2) Protection of sensitive information

When a user requests registration for a payment service, measures must be implemented to limit the leakage of personal and sensitive information provided during the user and authorization verification process.

If the personal information verification process for user and authorization confirmation is conducted through a third-party digital wallet app during the payment service registration request, it is important to ensure that this information is not provided to the digital wallet app developer. This can be most effectively guaranteed by verifying service availability through a secure connection with the service provider when the service registration request is made.

If the digital wallet app collects the user's personal information to send to the payment service provider, this information should not be transmitted to the digital wallet app provider and must be immediately deleted after verifying service availability.

Card numbers stored and used for payment services should be generated separately from the physical card number and CVV.

3) Security features for payment service execution

User authentication must be required at the start of the payment service. This authentication method should be stored separately for each payment card.

The number of consecutive authentication failures must be limited. If the specified number of failures is exceeded, the service should be suspended and enhanced user authentication should be required for reactivation.

The waiting time for the payment service after successful authentication must be limited. If the specified waiting time is exceeded, the payment service should be terminated.

Payment authorization information should be tokenized as a single-use token to prevent reuse.

The standards set forth in the ISO 12812-x series must be adhered to. The series of standards are as follows:

[b-ISO 12812-1]: General framework

[b-ISO 12812-2]: Security and data protection for mobile financial services

[b-ISO 12812-3]: Financial application lifecycle management

[b-ISO 12812-4]: Mobile payments-to-persons

[b-ISO 12812-5]: Mobile payments to businesses

A security level that complies with requirements of [b-PCI-DSS] must be maintained.

4) Support for International NFC Payment Standards

The international standards set forth in the ISO/IEC 14443-x series for card and security devices for personal identification – contactless proximity objects must be supported. The series of standards are as follows:

[b-ISO/IEC 14443-1]: Physical characteristics

[b-ISO/IEC 14443-2]: Radio frequency power and signal interference

[b-ISO/IEC 14443-3]: Initialization and anticollision

[b-ISO/IEC 14443-4]: Transmission protocol

5) QR code security

When using quick response (QR) codes, dynamic QR codes (not static ones) must be used.

When utilizing QR codes, the [b-CFIPSTMOPAY] standard for mobile payment service – general QR Code, developed by the financial information technology promotion council, must be adhered to.

6) Magnetic payment security

To prevent reuse, tokens should be generated with time limits and device numbers included.

Since support for magnetic cards is being phased out internationally, it is recommended to avoid supporting them.

9.1.2 CBDC wallet security

1) Overview of Central Bank CBDC wallet

Countries around the world are preparing for the issuance of central bank digital currencies (CBDCs). CBDCs are classified into wholesale CBDCs, retail CBDCs, and cross-border CBDCs, each implemented in different ways. Wholesale CBDCs are used for clearing between financial institutions and central banks and do not require a separate digital wallet. Cross-border CBDCs need to include functionality for international payments, whether they are wholesale or retail CBDCs, and are therefore still in very early stages. Generally, the common type of CBDC that uses digital wallets is the retail CBDC, so this discussion will focus on retail CBDCs.

CBDCs, as centrally managed digital currencies, do not necessarily require the use of blockchain technology; however, many countries are considering blockchain-based CBDCs to enhance the integrity of transaction records. Unlike physical cash, CBDC transactions are centrally managed and generally do not guarantee anonymity. Instead of anonymity, CBDCs offer the advantage of minimizing financial loss through identity verification and reissuance in case of loss or damage.

Users of CBDCs open digital wallets and can store their bank balances in the form of digital currency in their wallets or transfer them to their accounts. Digital currency stored in digital wallets can be used for both online and offline payments. Offline payments can be conducted similarly to electronic payments or through a method where an agreed amount is exchanged between digital wallets. Thus, payment request and receipt functionalities are necessary.

CBDCs operated under the central bank's management according to national laws must adhere to regulations concerning financial identity verification, prohibition of illegal assets, money laundering, and funding for extortion, among others. Consequently, in addition to standard keys and encryption modules for digital currency transactions, there is a need for the storage and management of additional information such as user identification, account details, and balances. Monitoring functions to detect illegal transactions are also required. Restrictions on usage frequency and payment amounts may be applied based on general banking operational policies. Furthermore, CBDCs aiming to provide characteristics similar to physical cash are generally expected to support transactions for a certain period even during network outages, which necessitates the addition of offline payment functionality.

CBDC digital wallets must offer a high level of security capable of resisting hacking attempts by unauthorized users. Therefore, secure digital wallet software, operating on a secure OS and based on HSMs for secure encryption and key storage, is necessary.

2) Independent CBDC wallet provision

CBDC wallets should be developed and operated as standalone digital wallet apps, without being added to apps for other purposes or incorporating additional services, to enhance security.

3) Operation on separate hardware chips

CBDC wallets must use a separate HSM from other digital wallets. The CBDC wallet should be initially stored within this HSM and updated using secure communication methods such as mobile networks, from the central bank's (e.g., Bank of Korea [b-BoK2023]) certified CBDC management system.

Before execution, check for and apply any updates.

4) App validation

CBDC wallets must be securely developed, tested, and certified according to the central bank's policies. The CBDC wallet app should be developed as a trusted application running within a trusted execution environment (TEE).

The app must be signed with a certificate from the central bank to verify its source and integrity. Apps failing verification should not be installed.

Even after successful verification and installation, the app's integrity must be verified before execution. Apps failing integrity verification should not be executed.

5) Security policy monitoring

The CBDC app must set security policies according to the criteria defined by the central bank. The user's ability to change policies and their scope should be minimized.

The CBDC app must monitor compliance with security policies in real-time and report any violations immediately to the CBDC management system.

6) User registration

For the use of the CBDC app, user authentication and real-name verification must be conducted, and users must be registered. During user registration, two or more different types of authentication information must be provided using a secure input/output path.

7) CBDC app key security

The private keys of the CBDC app must not be exposed outside of the HSM.

8) User authentication at execution

User authentication must be performed when the CBDC app is executed.

9) Use during network failures and battery depletion

The CBDC payment service must remain usable for a certain period even during network outages with the central server or when the smartphone's battery is depleted.

As soon as network issues are resolved, communication with the CBDC management system must occur to perform settlement procedures for transactions that occurred during the outage.

10) Incident reporting and response

In case of suspected illegal or accidental transactions, more than three integrity verification failures, attempts to modify the CBDC app, abnormal operation of the CBDC app, or loss of the smartphone, the incident must be reported immediately through designated methods and usage should be suspended.

Such reporting can be performed directly by the user or automatically by a monitoring module within the wallet. Additionally, the CBDC management system can detect incidents and halt the wallet's use.

11) CBDC wallet app deletion

Deletion of the CBDC wallet app must be performed either upon the user's request or according to the central bank's policies, through the CBDC management system.

9.1.3 Blockchain wallet security

1) Overview of blockchain wallet

A typical blockchain digital wallet consists of modules for key and address generation, key storage, key recovery, transaction signing, state management and external interfaces.

The key and address generation module creates a pair of private and public keys and generates an address using the public key. The key storage module stores and manages these generated keys. The key recovery module restores necessary keys. The transaction signing module composes and signs transaction information. State management handles information related to the digital wallet itself and the connected blockchain and additionally stores information about coin balances or owned NFTs.

The external interface communicates with the blockchain, host, and other apps. It transmits transactions to the blockchain or queries information from it. It also provides information to users or receives necessary data via a graphical interface from the host running the digital wallet. Additionally, it provides an interface for exchanging information with other applications.

2) Secure key management using HSM

Security incidents in blockchain environments are primarily caused by vulnerabilities in key management and smart contract code. Therefore, blockchain wallets must use the hardware security module (HSM) functionality as a requirement.

Blockchain wallets vary according to the blockchain instance in terms of encryption protocols, transaction formats, and coin types. As a result, a blockchain wallet is not a single wallet but should be able to expand either into separate wallets for each blockchain or by incorporating apps within a single wallet for different blockchain services.

3) Separation and integration of digital wallet and general functions

Requesting transactions and receiving change events from each blockchain are core functions of a blockchain digital wallet. Functions that provide information on the status or coin price fluctuations of each blockchain within the wallet or those of interest should be implemented as general apps provided by the operating system.

If a user wants to perform transactions through the digital wallet based on the analysis results, the digital wallet app can designate the relevant app as an integrated app and open a service port for connection.

4) Considerations when using centralized exchanges

When using exchange services, users' private keys are often managed by the exchange. It is important to be aware that the exchange-provided digital wallet app may not store private keys for blockchain accounts.

If the exchange uses multi-signature or multi-party computation techniques, users' control may be partially guaranteed, but it should be recognized that the security of the exchange cannot be controlled by the user.

5) Blocking the use of known vulnerable smart contracts

The safety of smart contracts extends beyond the scope of digital wallet security. However, it is recommended to include functionality that detects and avoids transactions involving known vulnerable smart contracts.

6) Prohibition on recording personal and sensitive information on blockchains

Personal and sensitive information must not be recorded on a public blockchain. Such information should not be included in additional fields of transactions. If necessary, this information should be stored in an off-chain storage, and a pointer to this data should be included in the transaction.

Sensitive information should be exchanged securely between authorized parties who require the data.

9.1.4 ID wallet security

1) Overview of ID wallet

The identity digital wallet consists of the following components: key management and cryptographic modules, a management module, a data repository for verifiable credentials (VC) and verifiable presentations (VP), and external interfaces.

The key management and encryption module generates, stores, and manages key pairs for decentralized identifiers (DIDs). It also provides cryptographic functions for VC applications and verification, as well as for the creation and submission of VPs. The data repository stores and manages relevant data, including basic data for user verification, user identifiers, user attributes, issued and created VCs and VPs, trusted issuers, lists of verifiers, and usage history. The VP generation module creates VPs requested by trusted parties from stored VCs based on user choices.

The external interfaces require connections with verifiers and verifiable credential issuers. These interfaces may operate through online or offline browsers or physical interfaces.

Figure 3 illustrates the configuration of the European digital identity (EUDI) digital wallet. The European digital identity wallet uses electronic identifiers (eID) issued by the government in accordance with the electronic identification, authentication and trust services (eIDAS) regulation for citizens and businesses within European Union (EU) member states, rather than decentralized IDs.

Additionally, the relying party interface is the EUDI wallet interface to qualified trust service provider ((Q)TSP), qualified electronic attestation of attributes ((Q)EAA) providers, member states infrastructures, national e-ID, relying parties and other sources of EEAs communication channels (online/offline) between the EUDI wallet and other parties.

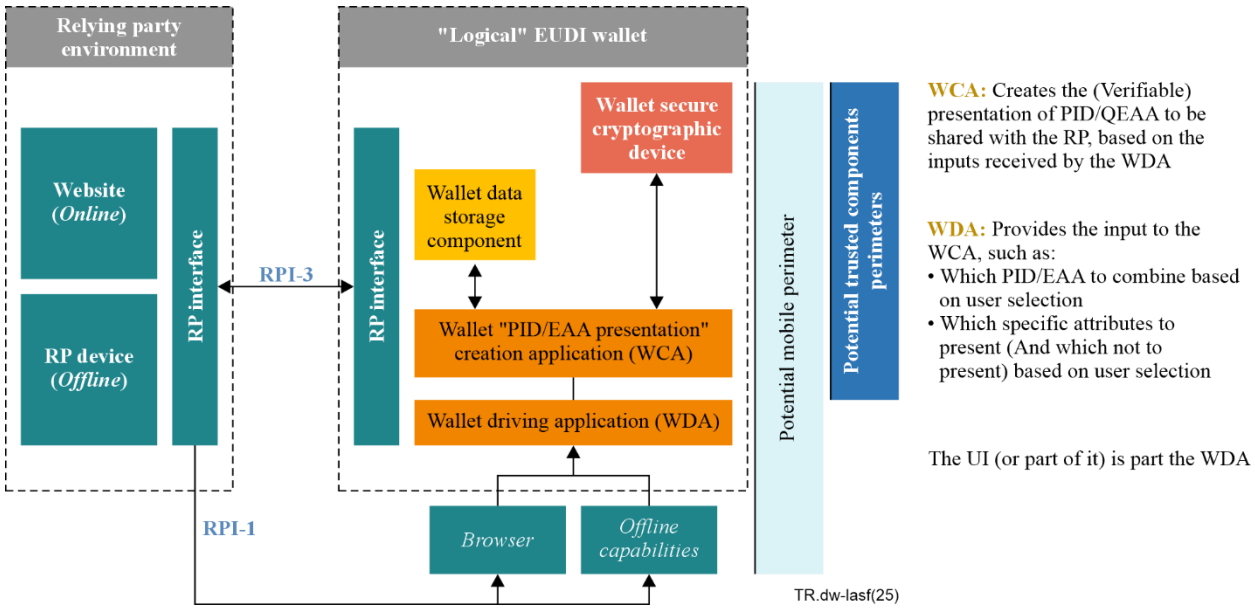

Figure 3 – EUDI wallet configurations conceptual model [b-EDIWARF]

2) Compliance with domestic and international standards

It is recommended that ID wallets comply with international standards, including [ISO/IEC 18013-5], [W3C DIDs 1.0] and [W3C VC DM 2.0]. Additionally, support for [W3C VC API 3.0] and [W3C CH API 1.0] is recommended to ensure interoperability and secure credential management.

3) Privacy protection

Personal information must not be included in DID and DID documents. When using blockchain-based identity verification, documents containing personal information (VCs and VPs) must not be recorded on a public blockchain. Only the hash of the document should be recorded on the blockchain if needed for issuance evidence.

4) Minimizing DID connections

When using the same DID across multiple VCs/VPs, there is a risk of deriving identifiable information from the combination of information included in each VC/VP. Therefore, a new DID should be created for each necessary VC.

Even when using different DIDs, if the same key is used in the DID documents, it could link different DIDs to the same entity. Hence, unique key pairs should be generated for each DID.

5) DID Key security

For effective management of multiple DIDs and keys, it is common to derive keys from a master key. Since a breach of this master key could lead to identity theft, it must be securely stored within an HSM.

6) VC security

When requesting a VC that verifies physical identity, the issuer must confirm the identity and issue a VC that connects to an anonymized DID, including the information requested. The request and transmission of the VC must be done through a secure end-to-end encrypted communication channel.

As personal information included in VC requests may be exposed by the issuer, VC requests should only be made to trusted issuers.

Since personal information included in the VC may be exposed to the service provider (trusted party), the information included in the VC should be minimized as much as possible.

The issued VC must be verified to ensure it was issued accurately according to the request. Verified VCs should be stored in a secure area, such as an HSM.

7) VP security

When a service provider requires user authentication, a VP containing the information requested by the service provider must be signed and generated using one or more issued VCs.

To prevent the inclusion of information not needed by the service provider, privacy protection technologies such as selective disclosure should be used to prevent the exposure of critical personal information. If such technologies are applied, the system must provide functionality to select whether critical information is exposed during VP creation.

The generated VP must be transmitted to the service provider through a secure end-to-end encrypted communication channel.

8) Digital wallet VC backup

Since VCs are stored only in the digital wallet, there is a risk of damage or theft of the VC and personal keys used for identity verification due to smartphone or digital wallet data loss. Therefore, backup, recovery, and restoration functions for the identity wallet must be provided.

9) Blocking known vulnerable smart contracts

The safety of smart contracts extends beyond the scope of digital wallet security. However, it is recommended to include functionality that collects and avoids using transactions involving smart contracts known to be vulnerable.

9.1.5 Physical access service wallet security

1) Overview of physical access service wallet

Physical access services provide functionality to unlock cars or building doors from a short distance using NFC, Bluetooth low energy (BLE), or ultra-wide band (UWB) technologies.

For digital car keys, a vehicle manufacturer's app must be installed on a smartphone. The car information is registered, and the physical key must be used to enter the car before it can be registered with the app. During this registration process, an encryption key is generated, and the vehicle manufacturer's server is contacted.

After registration, the car door can be unlocked using the smartphone's NFC function, and the engine can be started, or the door can be locked through the smartphone dock. For overseas vehicles, driving mode settings may also be available. The owner of the digital key can share the digital key with others, such as family members, or revoke access as needed.

For access control with digital keys, doors can be unlocked via NFC and configured to automatically lock when the door closes or when the area is exited. Like vehicle digital keys, the owner can grant access permissions to visitors, set access durations, or revoke access.

Digital key-based access control systems can be used in hotels, companies, schools, and multi-family housing. They allow for tasks such as verifying access, specifying access areas and durations, and recording access logs through NFC communication.

Other integrated services

FOB tags or wristbands with barcode printing used in theme parks, resorts, cruises, etc., present information stored in NFC memory or barcodes to a reader. These systems not only control access but also offer various services based on the information provided by the server and reader. If pre-paid costs are involved, the reader calculates the payment amount based on the provided information and handles payment processing at the end of the service.

While NFC chips/tags currently provide encryption using pre-shared secret keys, digital wallet apps that offer these features use asymmetric key encryption to provide enhanced and more secure user control.

Event tickets for movies, performances, and sports events can also be securely stored, gifted, or linked to reward programs using digital wallet functions. This integration can vary depending on the business cooperation between the service providers and digital wallet app developers.

2) Hotel room key security

For the following functions, appropriate information must be provided, and user approval must be obtained:

– When the hotel's server provides the function to add a hotel room key to the user's wallet, before activating the key.

– When automatic check-in and check-out are supported through the registered room key, before executing this function.

– Key updates for changes in stay from the hotel server.

– When a single key provides access to multiple rooms, for both the key and the users of those room keys.

– Before processing expired keys, a decision must be made whether to automatically delete or archive the key.

3) Access pass security

For the following functions, appropriate information must be provided, and user approval must be obtained:

– When adding an access pass to another smart device, such as a smartwatch.

– When setting up automatic addition to another smart device.

– When an access pass that is valid on only one device is moved to another device.

When sharing an access pass with another user, the default expiration period should be automatically set, and the user should have the ability to change it. Re-sharing of shared access passes should be prohibited unless explicitly permitted by the user.

4) Vehicle key security

Before pairing with the vehicle, the vehicle manufacturer's method must be used to verify ownership of the vehicle. The pairing process should be initiated through the vehicle manufacturer or the vehicle's built-in features, not the digital wallet app.

Communication used for pairing must be securely encrypted.

For the following functions, appropriate information must be provided, and user approval must be obtained:

– When adding a vehicle key to another smart device, such as a smartwatch.

– When setting up automatic addition to another smart device.

When sharing a vehicle key with another user, the default expiration period should be automatically set and the user should have the ability to change it. Re-sharing of shared vehicle keys should be prohibited.

5) Communication security during digital key sharing

To protect against man-in-the-middle or relay attacks during communication for sharing digital keys, a secure, end-to-end encrypted communication channel must be established using one-time session keys. Additionally, whenever possible, interference or eavesdropping detection features should be used.

9.2 Common security features for digital wallets

9.2.1 Security features of mobile digital wallet

Figure 4 shows the security mobile platform for digital wallet.

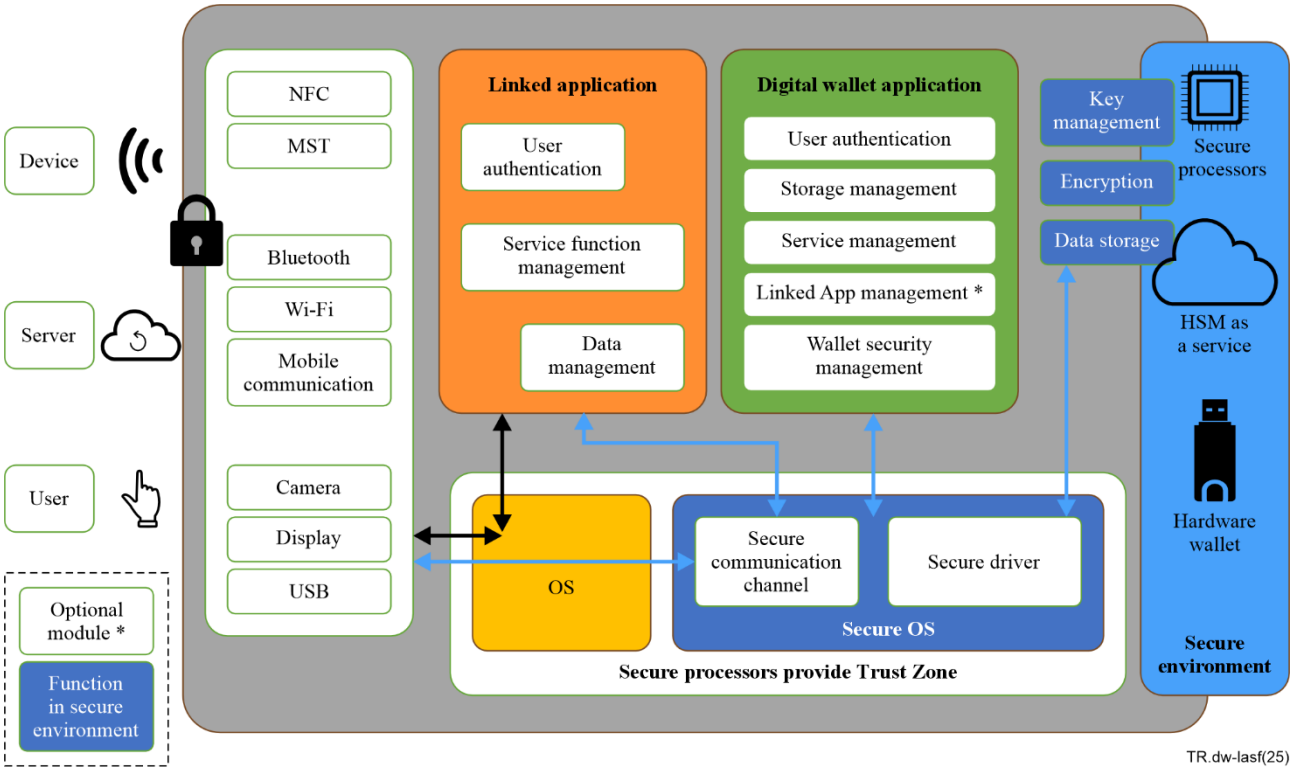

Figure 4 – A secure mobile platform for digital wallets

9.2.1.1 Secure processors -trust zone

Trust zone is a hardware security technology that separates the trusted execution environment (TEE), used for executing secure applications and sensitive data within a processor, from the rich execution environment (REE), where general applications and data are executed, as shown in Figure 5. It divides the CPU, address space, and memory at a hardware level to isolate the secure and non-secure areas.

The TEE comprises hardware features supporting strict isolation between environments and monitor software responsible for managing entry into the security mode. For enhanced security, it can communicate with a separate security chip or hardware security module (HSM). The execution environment within this chip is referred to as the secure environment (SE).

The security OS of the TEE boots before the general OS providing the REE, using separate APIs to establish connections with input/output devices, thus preventing data leaks through OS jailbreaking or malicious software.

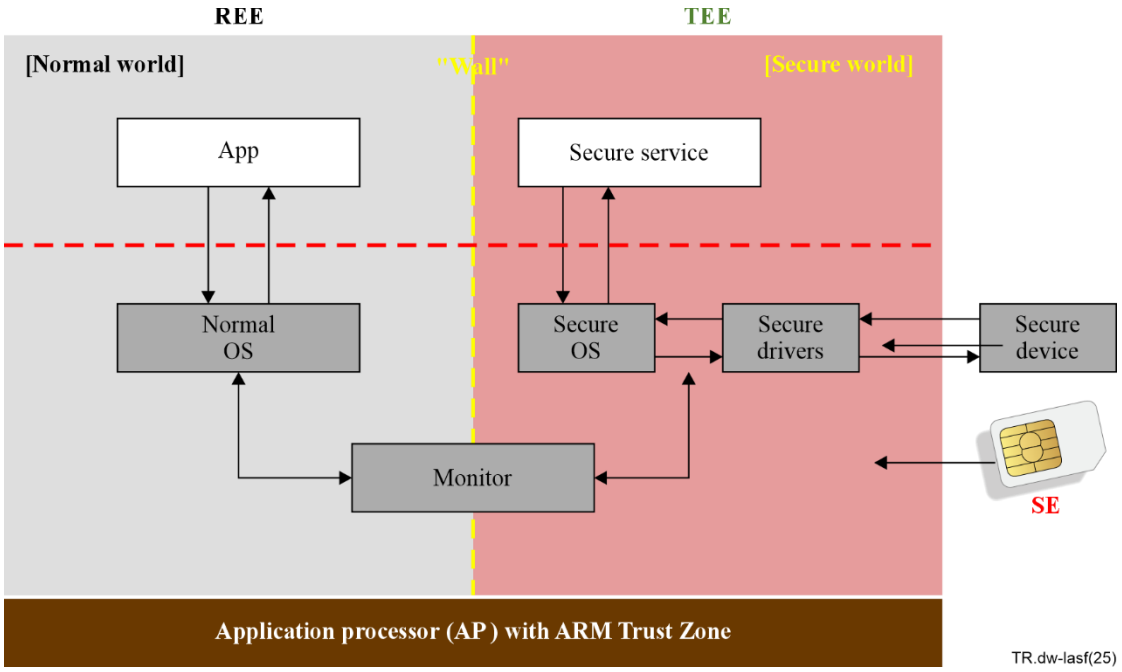

Figure 5 – REE and TEE provided by advanced RISC machine (ARM) trust zone

9.2.1.2 Secure hardware digital wallet – mobile

Many wallets in today's market are often independently developed and used for specific service purposes. As a result, users frequently find themselves installing multiple wallets and occasionally needing to operate each one independently. Smartphone manufacturers are competing to leverage the trusted execution environment (TEE) functionality based on their control over hardware, aiming to provide enhanced security while consolidating all data into a single wallet. There has been a significant increase in partnerships between existing payment apps like credit cards and digital wallets, integrating these functionalities into the digital wallet. However, wallets requiring a high level of encryption and key management, such as blockchain or identity wallets, are still predominantly offered independently by service providers.

Hardware wallets are generally considered the most secure, yet most lack standalone communication capabilities, focusing primarily on securely executing encryption processes, which necessitates integration with mobile devices or PCs. Hence, the safest approach combines smartphone TEE functionality with hardware chips or wallets for enhanced security.

A secure mobile digital wallet operates as a trusted application within the TEE. This wallet directly accesses various communication and authentication functions within the smartphone through the secure connection provided by the TEE, interfacing with both users and service providers. Simultaneously, it interfaces with embedded chips, hardware wallets, or the cloud, providing robust security through secure physical or logical connections.

9.2.1.3 Essential modules for a digital wallet

To securely provide necessary services, a digital wallet must operate within a trusted execution environment (TEE). Specifically, key management and encryption processes should occur within a hardware-secured environment to minimize the risk of key exposure. Critical data, such as digital currency balances, should also be securely stored within this secure environment through encryption or similar means. Within the TEE, the digital wallet operates using secure APIs to authenticate users. It communicates with service-specific terminals or servers of service providers, validating input data using cryptographic functions, processing it, and generating necessary output data using similar encryption methods for transmission, thus delivering services.

To achieve this, as shown in Table 2, essential foundational elements such as key management, encryption, and data storage modules within the secure environment are crucial. Additionally, modules for user authentication, service management, storage management, and wallet security management are essential for performing these tasks.

When registering multiple cards or digital keys within a single digital wallet app, it's vital to generate distinct asymmetric key pairs for each service to separate and protect service-related information. Within the secure environment, service-specific keys and relevant information are stored, while service management involves registering and executing service-specific processing functions based on user preferences.

Trusted apps can register interworking apps to offer various additional functionalities and information. These interworking apps operate within the general operating system (OS) but can utilize opened ports, provided by the digital wallet, to deliver services. If the digital wallet provides multiple service functions, interworking apps must also be registered for each specific service.

Table 2 – Essential modules

Execution Environment	Function Module	Function
Secure environment	Key management	Public and private key generation, storage, use, recovery, disposal, generation of session-specific secret key, use, management, disposal
	Encryption	Perform the encryption of cryptographic libraries and input data, providing output data to the digital wallet app.
	Data storage	Maintain a data repository requiring encryption for protection.
Trusted execution environment (TEE)	User authentication.	Authenticate the legitimate user when the digital wallet app is activated.
	Storage management	Manage information related to the data repository.
	Service management	Manage the services to be executed by the wallet and the necessary information for these tasks, including input validation, output generation, and more.
	Management of linked apps	When providing additional information services related to the e-wallet app, registering linked apps and creating communication ports accessible only to these linked apps ensures the secure delivery of services.
	Digital wallet security management	Perform internal security checks such as integrity verification of the wallet app itself, and incorporate transaction monitoring features when necessary, as these are essential for effective security management.

To securely provide necessary services, a digital wallet must operate within a trusted execution environment (TEE). Specifically, key management and encryption processes should occur within a hardware-secured environment.

9.2.2 Common security aspects for digital wallets

9.2.2.1 User security

1) Overview of user security

Typically, users install software-wallet applications on their devices. Hardware wallets perform critical key management and encryption functions quickly and securely. However, it is common to use a software application that integrates with the hardware wallet for user interface (UI) or network connectivity. For cloud wallets, users can install and use apps through a browser or browser-based functions.

The security of digital wallets can be compromised due to user mistakes or a lack of awareness during the installation and usage process. For more detailed security guidelines for PCs and mobile devices, refer to [b-NSA-Telework-Mobile] and [b-NSA-HomeNetwork].

2) User security awareness

Since the security of user devices and the safe use of digital wallets are the user's responsibility, user awareness and education need to be enhanced. Appropriate security guidance and alerts during wallet usage based on the characteristics of the digital wallet should be provided. The following points should be communicated to users.

3) User responsibility

For mobile devices, actions that could compromise device security, such as rooting, should be avoided. Additionally, users should not attempt to analyse or modify the digital wallet.

4) User device security

To protect user devices securely, the latest OS should be used, and security updates should be kept up to date. Basic security features such as using secure passwords should be applied. If possible, additional security features like biometric authentication should be utilized.

Remote device lock and data wipe features should be set up to respond to theft or loss of mobile devices.

5) Malware prevention

To block the installation and operation of malware that captures memory or keyboard inputs, users should be aware of phishing risks and be cautious when opening attachments or clicking URLs. All programs, including antivirus software, should be downloaded, installed, updated, and patched from trusted sources.

6) Digital wallet backup

Users should be aware of and perform appropriate backup methods, including frequency and means, to prepare for potential wallet damage.

7) Social engineering response

When transferring assets for transactions, investments, or other incidents, it is essential to verify the identity of the counterparty and the trustworthiness of the service.

8) Incident response

If unusual signs or security warnings occur, users should stop the activity and report them to the appropriate response agencies. Response agencies include the digital wallet developer, service provider, and supervisory authority.

9.2.2.2 Interface security

1) Virtual private network (VPN) usage

Whenever possible, always use secure connections with secure sockets layer (SSL), TLS, or datagram transport layer security (DTLS) to provide end-to-end encryption. Verify the certificates of communication partners during secure connections.

If you are using unprotected communication channels or known vulnerable encryption protocols, restrict their use or alert users to the fact that they are connecting through a vulnerable channel, and provide warnings to allow them to decide whether to proceed with communication.

2) Wi-Fi security

When using Wi-Fi, ensure that secure connections such as Wi-Fi protected access 2 (WPA2) or Wi-Fi protected access 3 (WPA3) are provided. For untrusted Wi-Fi networks, provide features such as randomized MAC addresses and random Wi-Fi frame sequence numbers to prevent device tracking.

For unicast and multicast communications, it is recommended to provide protection at the WPA2 and WPA3 levels through protected management frames (PMF) services.

3) Bluetooth security

When using Bluetooth connections, employ protocols that offer secure mutual authentication and encryption features. For privacy protection in Bluetooth low energy (LE) use, address randomization and transmission key derivation methods should be used.

Particular attention is needed when connecting devices via Bluetooth. For simple security pairing, it is recommended to use elliptic-curve Diffie-Hellman (ECDH) exchange to prevent passive eavesdropping, and methods that require user responses such as passkey input or two-number comparison to prevent man-in-the-middle attacks. If possible, use Bluetooth version 4.1 or higher.

For detailed Bluetooth security information, refer to [b-NIST SP 800-121].

4) NFC Security

NFC transmissions are susceptible to eavesdropping, so information transmitted via NFC should use encryption methods supported by the receiving device or server to generate one-time tokens, thus preventing data leakage and reuse. NFC is also vulnerable to interference, data alteration, or URI spoofing based on its transmission characteristics. Data insertion attacks may occur when there are delays in response transmission between devices. RF field checks can detect surrounding interference but are not a fundamental solution. Encryption and secured channels are the most common protective measures.

When using NFC for door locks, unauthorized persons may open doors through relay amplification. For high-value items like vehicles, it is recommended to use ultra-wideband (UWB) technology, which provides precise location detection.

5) MST security

Magnetic secure transmission (MST) signals are also susceptible to eavesdropping, and there have been cases where payment token data were intercepted and reused initially. Time-limited features have since been added to address this vulnerability. However, as magnetic cards are becoming obsolete worldwide, it is recommended that MST not be used.

6) Other interface security

Depending on the type of connected device, restrict unnecessary universal serial bus (USB) device connections. Provide detailed access control for secure digital (SD) cards. Offer additional features such as secure keyboards, capture prevention, and display clearing during screen transitions to block channel hooking.

9.2.2.3 Digital wallet module security

1) Download and update

When downloading and updating the digital wallet app, ensure that it is done through a secure channel from a trusted source. For added security, it is preferable to store the app in the TEE from the time of manufacture and update it through a secure channel.

Check for and apply any new updates before executing the app.

2) App verification

The digital wallet app should be developed as a trusted application running in the TEE. It must be signed with a certificate issued by a trusted publisher to verify the app's source and integrity. The source of the app must be known and approved. Apps that fail verification should not be installed.

Even after installation, verify the integrity of the app before executing it. Apps that fail integrity checks should not be executed.

3) User registration

When first running the digital wallet app, users must be registered. During registration, use secure input/output paths to register two or more different types of authentication information. Alternatively, pre-registered authentication information in a secure area may be used for user authentication in the digital wallet with user approval. Additional information required for user registration may vary depending on the services provided by the digital wallet app.

During user registration, generate and securely store a secure asymmetric key pair for the digital wallet using a separate HSM.

4) App key security

The private key of the digital wallet app should not be exposed in plaintext outside of a separate secure area. The keys of the digital wallet should not be shared with any stakeholders or stored on servers managed by others.

When backing up the digital wallet app keys, they must be transmitted in an encrypted form through a secure channel after user authentication.

5) Two-factor authentication

Use two-factor authentication for user verification when performing important functions. Examples of such functions include:

– Initial activation or download/update of the digital wallet app
– Registration of individual services within the digital wallet app
– Backup of keys and the digital wallet
– Authentication settings for functions requiring continuous operation based on user requests

6) HSM function usage

All key and important data storage and encryption functions must be executed within an HSM module of security level 3 or higher.

ISO/IEC JTC 1/SC 27 provides many cryptography and key management standards.

7) Import the wallets to other devices

When synchronizing a digital wallet for use on another device, the wallet data must be exported in a format readable by the digital wallet on the other device. Important data should be included in an encrypted form.

The digital wallet on another device should read the exported data and reset it appropriately for that device, as wallet keys may vary based on the device's provided seed.

8) Immediate app termination after use

After using necessary service functions, the app should be immediately closed. The app should provide a warning upon completion of the function so that users are aware that the service is finished.

If the user does not manually close the app, it should be automatically terminated after a certain period (e.g., 30 seconds) or upon meeting specific conditions related to the function used. Upon termination, memory should be cleared.

When switching to another app or function while using the digital wallet app, the app should be stopped. Upon returning, user authentication should be required again.

This requirement may not apply to functions requiring continuous operation.

9) Warning during continuous operation

For continuous operation features, apply at least minimal user authentication such as biometric or motion authentication to prevent unintended services from being executed.

If a continuous operation feature is active, provide a warning to users through sound or vibration to ensure they are aware.

Examples of continuous operation features include:

– Opening doors, opening vehicle doors, etc.

– Micropayments for public transportation

10) Usage during battery drain

Essential services should remain available for a certain period even after the smartphone's battery is drained. For example, important functions like access keys, vehicle keys, and payment functions should guarantee a minimum operation time. This can generally be achieved by cutting off power to other smartphone functions once the minimum power level is reached.

– The ability to specify essential services that must guarantee minimum operation time.

– Capability to provide critical NFC-based services even with the phone's power off.

This feature may not apply if the user turns off the power manually.

11) Deleting the digital wallet

When deleting the digital wallet, ensure complete removal by deleting critical information before uninstalling the app. For information not stored in the HSM, use methods like overwriting with random numbers. The deletion procedure should include:

– Removing all card information within the digital wallet

– Removing all keys within the digital wallet

– Removing all digital wallet information

Appendix I

Processes for digital wallet services

I.1 Registration of specific service modules on a digital wallet

In a mobile digital wallet with trusted execution environment (TEE) capabilities, the process of registering other service modules such as payment cards, event tickets, office IDs, home keys, car keys, etc. proceeds as follows:

1) The user launches the digital wallet app.
2) The digital wallet establishes a secure input channel.
3) The user requests to register a service like a card or digital key.
4) The digital wallet verifies if the service provider is eligible for registration, then establishes a secure connection with the service provider's server based on the provided information.
5) The digital wallet provides user and device information to the service provider server.
6) The service provider's server verifies the user's authorization based on the provided information.
7) The service provider's server sends the appropriate service-specific app to the digital wallet based on device information. During this process, the digital wallet can generate a service-specific asymmetric key pair for future security purposes and share the public key with the service provider server.
8) The digital wallet registers the service-specific app information.

I.2 Execution of a specific service modules on a digital wallet

The process for using the registered services after registration is as follows:

1) The user launches the general-purpose digital wallet app and requests a service.
2) The general-purpose digital wallet app establishes a secure connection with the necessary input/output devices (NFC, Bluetooth, fingerprint scanner, MST, camera, screen input, etc.).
3) The digital wallet connects with the secure element (SE) within the TEE to execute the service app.
4) The service app communicates with the input/output devices through a secure connection to facilitate service usage.
5) The digital wallet is closed either upon user request, after a predefined duration, or once specific conditions have been met.

I.3 Deletion of specific service modules on a digital wallet

The process of deleting specific service modules from a digital wallet is as follows:

1) The user launches the digital wallet app and selects the service to be deleted.
2) The digital wallet establishes a secure input channel and requests user authentication.
3) Once the user completes authentication, the digital wallet verifies the status of the service module to ensure it can be safely deleted.
4) The digital wallet securely deletes all data associated with the service module using one of the following methods: secure key deletion via hardware security module (HSM), compliant data deletion methods, or random overwriting and encrypted deletion techniques.
5) The digital wallet performs an integrity check to verify that the service module has been securely deleted.

6) Once the deletion process is complete, the digital wallet displays a deletion confirmation message to the user and provides a backup option if necessary.

7) If required, the digital wallet sends a deletion request to the service provider's server to ensure that related information is also removed from the service provider's database.

Bibliography

[b-ISO 12812-1] ISO 12812-1:2017, *Core banking – Mobile financial services, Part 1: General framework*.

[b-ISO 12812-2] ISO 12812-2:2017, *Core banking – Mobile financial services, Part 2: Security and data protection for mobile financial services*.

[b-ISO 12812-3] ISO 12812-3:2017, *Core banking – Mobile financial services, Part 3: Financial application lifecycle management*.

[b-ISO 12812-4] ISO 12812-4:2017, *Core banking – Mobile financial services, Part 4: Mobile payments-to-persons*.

[b-ISO 12812-5] ISO 12812-5:2017, *Core banking – Mobile financial services, Part 5: Mobile payments to businesses*.

[b-ISO 22739] ISO 22739:2024, *Blockchain and distributed ledger technologies – Vocabulary*.

[b-ISO/IEC 14443-1] ISO/IEC 14443-1:2018, *Cards and security devices for personal identification – Contactless proximity objects, Part 1: Physical characteristics*.

[b-ISO/IEC 14443-2] ISO/IEC 14443-2:2020, *Cards and security devices for personal identification – Contactless proximity objects, Part 2: Radio frequency power and signal interference*.

[b-ISO/IEC 14443-3] ISO/IEC 14443-3:2018, *Cards and security devices for personal identification – Contactless proximity objects, Part 3: Initialization and anticollision*.

[b-ISO/IEC 14443-4] ISO/IEC 14443-4:2018, *Cards and security devices for personal identification – Contactless proximity objects, Part 4: Transmission protocol*.

[b-AW] Apple, Apple Wallet, https://www.apple.com/wallet/

[b-BoK2023] Bank of Korea, *Guidelines on digital wallet security and management*, https://www.bok.or.kr/portal/bbs/P0000274/view.do?menuNo=200730&nttId=10082044

[b-CFIPSTMOPAY] Bank of Korea, *Standard for mobile payment service*, https://www.bok.or.kr/portal/bbs/B0000239/view.do?nttId=10050750&searchCnd=1&searchKwd=&depth2=201157&depth3=201224&depth4=201226&depth5=200729&date=&sdate=&edate=&sort=1&pageUnit=10&depth=200729&pageIndex=1&programType=newsData&menuNo=200729&oldMenuNo=201224

[b-EDIWARF] Comisión Europea, *The common union toolbox for a coordinated approach towards a European digital identity framework*, https://ec.europa.eu/newsroom/dae/redirection/document/93678

[b-GW] Google, *Google Wallet*, https://www.apple.com/wallet/

[b-Meta] Metamask, *A crypto wallet & gateway to blockchain apps*, https://metamask.io

[b-NIST SP 800-121] NIST, Guide to Bluetooth Security (2017). Available at: https://doi.org/10.6028/NIST.SP.800-121r2-upd1

[b-NSA-HomeNetwork] NSA (2023), *Cybersecurity Information (CSI): Best Practices for Securing Your Home Network, February 2023*. Available at: https://media.defense.gov/2023/Feb/22/2003165170/-1/-1/0/CSI_BEST_PRACTICES_FOR_SECURING_YOUR_HOME_NETWORK.PDF

[b-NSA-Telework-Mobile] NSA (2020), *Telework and Mobile Security Guidance (updated 14 August 2020)*. Available at: https://www.nsa.gov/Press-Room/Telework-and-Mobile-Security-Guidance/

[b-PCI-DSS] PCI Security *Standard Council, PCI Data Security Standard*, https://www.pcisecuritystandards.org/document_library/

[b-SW] Samsung, *Samsung Wallet*, https://www.samsung.com/sec/apps/samsung-wallet/

[b-W3C VCDM] W3C Recommendation, *Verifiable Credentials Data Model v2.0*, 2025 https://www.w3.org/TR/vc-data-model-2.0/#dfn-verifiable-presentation

[b-W3C universal wallet 2020] W3C Credentials Community Group, Universal Wallet 2020, https://w3c-ccg.github.io/universal-wallet-interop-spec/#terms