Balancing Privacy and Public Policy Objectives: A study of eCash applied to the Digital Euro

Bachelor Thesis

Joel Auerbach

from

Zürich, Switzerland

Faculty of Science, University of Bern

January 22, 2025

Prof. Christian Cachin
François-Xavier Wicht
Cryptology and Data Security Group
Institute of Computer Science
University of Bern, Switzerland

Abstract

Digital money has been in circulation for quite some time. The use of physical cash is declining while digital payments are rising rapidly, and most people in developed countries have e-banking accounts. Most digital payments currently involve American Express, Visa, or Mastercard as the payment service owners. However, central banks are exploring options to introduce digital payment systems, called central bank digital currencies (CBDCs), which do not rely on foreign payment owners. Central banks already provide digital money in the form of wholesale CBDC, but only to large banks in the form of overnight lending. More recently, central banks have started to explore the potential of CBDC for retail customers and publish reports about their findings. The goal is to create a digital payment system for the general population of a country. These ideas and proposals vary from central bank to central bank, usually depending on their perceived importance on the three factors: efficiency, scalability, and privacy. The Swiss National Bank and the Bank for International Settlements have identified eCash, a privacy-focused digital payment system introduced by David Chaum, as a potential technology for CBDCs in their exploration reports. In this thesis, we investigate eCash's privacy properties with respect to transaction privacy and payer anonymity and the feasibility of using it for the Digital Euro. We accomplish this by comparing the goals of the Digital Euro's investigation phase to an eCash-based CBDC to identify synergies and incompatibilities between them.

Acknowledgement

I sincerely thank my advisor, François-Xavier Wicht, for his extensive guidance during this thesis. François-Xavier Wicht supported me with almost weekly meetings, during which he took extensive time to help and make suggestions for my thesis. Additionally, I would like to thank Professor Christian Cachin, who allowed me to write this thesis in his research group.

Contents

Chapter 1

Introduction

Central banks are in charge of controlling monetary policy, one of the most powerful tools any country can have. However, technology can challenge this power. The introduction of Bitcoin in 2008 demonstrated that creating digital currencies outside of government control is successfully possible [20]. We believe that the threat of Bitcoin to the monetary system was perceived as minimal by central banks. However, the Libra proposal by Meta (previously called Facebook) in 2019 presented a more significant concern [2]. Libra aimed to create a digital currency pegged to a basket of existing currencies like the US dollar. It later rebranded to Diem, focusing on providing digital currencies pegged to a single currency. Diem ceased its operations in 2022 and sold all its assets. This initiative by Meta has led to heavy discussions about the future of money and the potential of digital currencies to disrupt the traditional financial system. Presumably, due to this new threat, central banks started working on their central bank digital currencies.

A survey conducted by the Bank for International Settlements (BIS) in 2023 revealed the reasons that central banks have regarding a potential introduction of a retail central bank digital currency (CBDC) [18]. The most prevalent reason is to ensure the singleness of money, which means maintaining convertibility at par across different forms of money because central banks are concerned that new forms of privately issued money could threaten this principle. For example, numerous digital stablecoins have been introduced privately with a peg to the US dollar. However, these pegs have proved unsustainable, leading to a decline in value relative to the US dollar. Another motivation for introducing a retail CBDC is to improve domestic payment efficiency and financial inclusion, particularly in emerging markets. Moreover, a central bank is arguably one of the most valuable assets of any country or union, as it is able to conduct monetary policy by using tools for controlling the money supply and interest rates. Therefore, it makes sense for a central bank to provide optimal forms of payment to its citizens to maintain relevance and ensure sovereignty over its currency. A retail CBDC is a digital version of the country's existing currency provided by the central bank that is available to the general public[1].

In this report, central banks representing 81% of the world's population and 94% of the economic output, 94% of them said that they engage in some form of work related to CBDC by the end of 2023. It is noteworthy that 54% of the central banks are engaging in experiments and proof-of-concept activities while 31% have even initiated the development of pilot projects. The European Central Bank also proposed a digital version of its currency, the Digital Euro. The idea is to create another payment form that complements current payment solutions, such as cash and digital payments.

While the report demonstrates the interest of central banks regarding CBDCs, the general population should keep track of the progress because of the potential drawbacks [1]. One significant concern is the potential loss of privacy compared to physical cash. Retail CBDCs could expose certain types of

[1] There are two distinct types of CBDC, retail and wholesale. Wholesale CBDC transactions are exclusively used between financial institutions, while retail CBDC is accessible to the general public. Whenever we mention CBDC in this thesis, we refer to retail CBDC.

sensitive information to their operators, especially if they use computationally expensive solutions to avoid performance losses. Researchers have identified two main privacy and transparency concerns regarding CBDCs. Specifically *user privacy*, which ensures that it is not possible to link the activity of a sender or recipient to the transaction, and *transaction privacy*, which hides information about the transaction itself. Another problem is the technological vulnerability or design mistakes that may occur during the creation of a CBDC. Furthermore, the introduction of a CBDC could have a significant impact on the banking system. If not implemented carefully, it could lead to bank runs if everyone tried to convert their holdings into CBDC.

Privacy and security drawbacks are the most prevalent concerns of the European Union's population about the Digital Euro. The European Central Bank conducted a public consultation to gather the opinions of professionals and citizens regarding the Digital Euro [15]. The findings demonstrate that privacy is the most essential feature for both professionals and citizens, with 43% of the respondents identifying it as their primary concern. Security is the second most important factor, selected by 18% of the respondents. These findings suggest that privacy should be one of the core attributes of a CBDC, as it may hinder adoption otherwise.

This thesis aims to evaluate the privacy features of the proposals by the Swiss National Bank [10] and the Bank for International Settlements [3], which use eCash, and the European Central Bank's Digital Euro [17] in order to address the privacy concern. Furthermore, we want to investigate the viability of eCash as a system for the ECB to utilize for the Digital Euro and explain our reasoning for why we think it is possible to do so. We begin with the necessary technical background in Chapter 2. Chapter 3 explores the potential implementation of a CBDC based on eCash. Chapter 4 examines the investigation phase of the Digital Euro. The objective of Chapter 5 is to explore whether the findings of the Digital Euro's investigation phase are compatible with a CBDC solution based on eCash.

Chapter 2

Background

In this section, we review and explain the key technologies behind eCash. First, we present the four-party model as the baseline for interactions between participants of the protocol. Second, we present the necessary cryptographic background.

2.1 Four-party model

The four-party model shows the operation of a debit card payment system [24]. The model uses four actors: the customer, the merchant, the acquirer, and the issuer. The payment owner does not count as an actor.

- The payment owner is the operator of the payment system. It is in charge of defining the fee structure, the cards, and the processing rules. It also operates a backend that clears, settles, and authorizes card transactions.
- The customer who makes a purchase using a card supplied by the issuer.
- The issuer is a payment service provider who supplies the customer with the cards, manages the customer account, authorizes card transactions, and provides a guarantee of payment settlement to the acquirer.
- The merchant who provides the good or service to the customer.
- The acquirer is the processor for the merchant. It manages the flow of card transactions between the merchant and the issuer.

It is often the case that the acquirer and the issuer are the same company. However, for this thesis, we treat them separately. Figure 2.1 shows the model visually with all the involved actors.

Privacy analysis When a customer makes a purchase with a card, the card issuer does not receive exact information about the specific good or service bought by the customer, but it does learn the name of the merchant's company. Intuitively, this information does not seem valuable, but the opposite is true. Data points about how often customers go to specific shops, their locations, and the amount spent can be valuable to competitors, for example, for marketing or strategic purposes. Additionally, the company's name is usually enough to infer the type of goods that the customer bought. If the customer makes a purchase at Coop, the issuer knows that it is most likely a necessary good. If the shop is called Cartier, it is a luxury good. The issuer and the acquirer do not exchange any customer data, nor do the merchant and the acquirer. This is because there is no direct contact between these parties and the customer. However, the acquirer might know the type of good or service that a merchant has just sold because they usually have access to a merchant's category code. The payment owner most likely knows the most. They

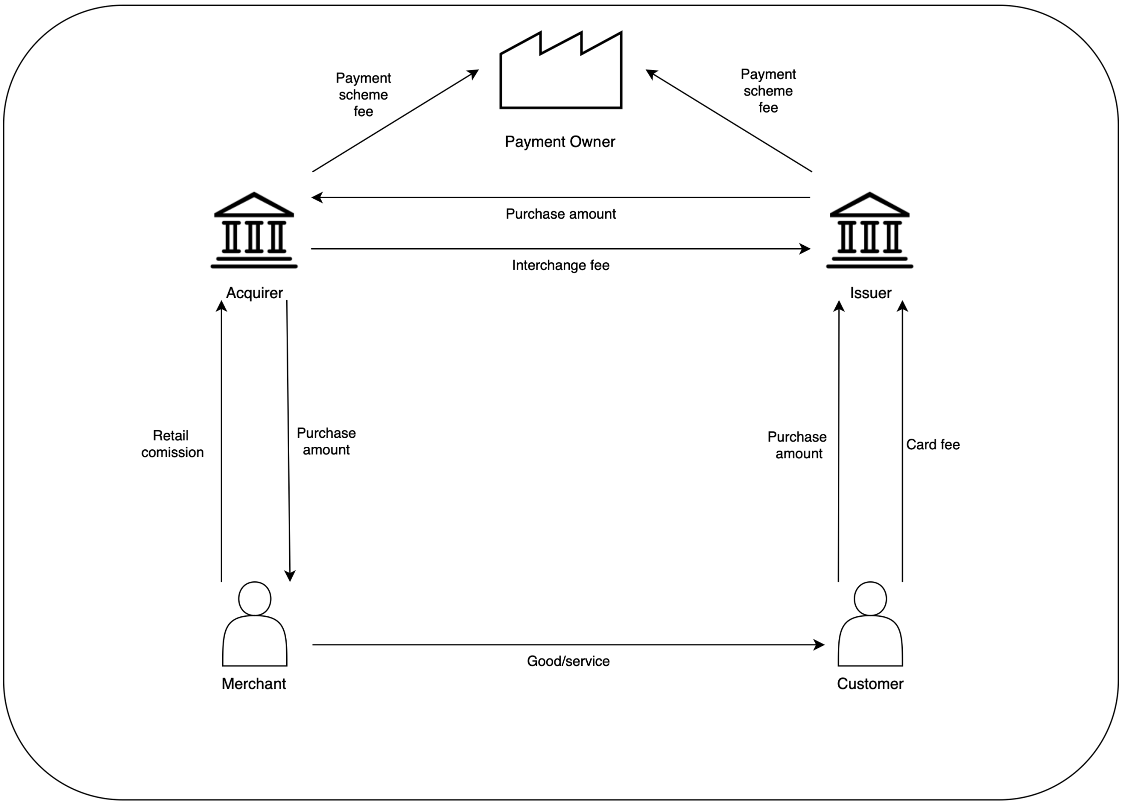


Figure 2.1. Four-party model

process the transaction data, which includes the amount, location, and merchant name. While payment owners can use this data for data analytics, it is important to note that it is also used for fraud prevention.

In the following sections of the background, we examine the cryptographic background needed to understand eCash.

2.2 Symmetric encryption

Symmetric encryption is a process where two or more parties who share the same secret key use this key to encrypt and decrypt information [19]. Since all parties use the same key, this is called symmetric cryptography. Let m be the message that Alice wishes to send to Bob. Firstly, Alice or Bob generates a private key $sk \leftarrow \mathsf{Gen}(1^\lambda)$ and shares it with each other. Secondly, Alice encrypts the message m by utilizing the private key sk to generate a ciphertext $c \leftarrow \mathsf{Enc}(sk, m)$ and transmits it to Bob. Lastly, Bob needs to decrypt the ciphertext c to be able to read the content of the message $m \leftarrow \mathsf{Dec}(sk, c)$ by using the private key sk.

1. The key generation algorithm $sk \leftarrow \mathsf{Gen}(1^\lambda)$ uses a security parameter λ as input and outputs the secret key sk. The security parameter influences the secret key's length, such that $|sk| \geq \lambda$, and the computational difficulty. While an increase in the security parameter leads to an improved security of the system, it also increases the runtime of the scheme and the size of the key.

2. The encryption algorithm $c \leftarrow \mathsf{Enc}(sk, m)$ takes the message m and the secret key sk as input and outputs the ciphertext c.

3. The decryption algorithm $m \leftarrow \mathsf{Dec}(sk, c)$ takes the private key sk and the ciphertext c as input and outputs the original message m.

Symmetric encryption is relatively faster and more efficient than asymmetric encryption, which we discuss in the next paragraph. Therefore, one potential application is the encryption of large amounts of data. However, symmetric encryption has its downsides. If two or more parties are involved, the players must derive a method for safely exchanging the secret key. If Alice lives in Switzerland and Bob in Australia, this could be challenging as meeting in person is not viable. A private channel, like a trusted third-party, could establish a connection between Alice and Bob, but this option would also be cumbersome. Transmitting the secret key over an insecure channel would not be a feasible solution, as it would be possible for a third-party (Eve) to eavesdrop on the communication.

One potential solution to this problem is using asymmetric encryption.

2.3 Asymmetric encryption

Asymmetric encryption algorithms, such as the RSA algorithm [21] or the Diffie-Hellman key exchange protocol [14] allow users to exchange a key over an insecure channel. They rely on the computational difficulty of deriving a secret key from its corresponding public key using current technology [5]. Alice first generates a key pair consisting of a public and private key $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$. Alice publishes the public key, and anyone wishing to communicate with Alice can retrieve it, encrypt the message using the public key $c \leftarrow \mathsf{Enc}(pk, m)$, and transfer it to Alice. Alice is then able to decrypt the message with her private key $m \leftarrow \mathsf{Dec}(sk, c)$. The encryption relies on the assumption that factoring large numbers is difficult [19].

1. The key pair generation algorithm $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$ uses a security parameter λ as input and outputs the public and private key (pk, sk).

2. The encryption algorithm $c \leftarrow \mathsf{Enc}(pk, m)$ takes the message m and the public key pk as input and outputs the ciphertext c.

3. The decryption algorithm $m \leftarrow \mathsf{Dec}(sk, c)$ takes the private key sk and the ciphertext c as input and outputs the original message m.

There are also disadvantages to asymmetric encryption. It is generally slower and more computationally intensive than symmetric key encryption. Key management can also be problematic since the private key needs to be kept secret. The public keys must be published to allow for universal access within the system. This can be a safety concern because it allows for man-in-the-middle attacks. In such an attack, an adversary could replace the registered public key with their own. This allows the adversary to read the message exchange between Alice and Bob. Additionally, algorithms based on the underlying assumption of mathematical difficulties like large number factoring are not quantum resistant.

Asymmetric encryption can encrypt or decrypt messages. In a message exchange, however, an important feature is missing: non-repudiation. This feature is important because, in a message exchange, there is usually no direct link between a sender and a recipient. If Alice sends a message to Bob, the message will go through a third-party that provides the service. It is in the interest of that third-party to prove to Bob that the message originated from Alice, as this will prevent Alice from disputing that she sent the message. One mechanism that allows for non-repudiation is digital signatures.

Digital signatures based on public key cryptography Diffie and Hellman define digital signatures as a scheme where a user with a public and private key pair can create a digital signature on a document [14]. Anyone can verify the signature using the public key, but no one can forge it because deriving a private key from a public key is considered computationally very hard. To provide non-repudiation to Bob, Alice generates a key pair (sk, pk) and publishes the public key. Alice writes the email and, once finished, uses her private key to create a digital signature s on the email m and sends the email together with the signature (m, s) to Bob. In order to verify the authenticity of the signature, Bob must first obtain Alice's public key and then use the verification algorithm to verify that the email originated from Alice [5]. The scheme consists of three probabilistic algorithms: $\mathsf{Gen}(1^\lambda), \mathsf{Sign}(sk, m)$ and $\mathsf{Vrfy}(pk, m, s)$ [19].

1. The key generation algorithm $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$ takes a security parameter λ as input and outputs a key pair (pk, sk).

2. The signing algorithm $s \leftarrow \mathsf{Sign}(sk, m)$ takes the private key sk and the message m as input and outputs a signature s.

3. The verification algorithm $b := \mathsf{Vrfy}(pk, m, s)$ takes the public key pk, the message m, and the signature s as input and outputs a bit b where $b = 1$ if the signature is valid and 0 otherwise.

We present in the following a public key cryptosystem that allows the implementation of digital signatures.

2.4 RSA protocol

We first describe a simple encryption scheme based on the RSA problem [19]. The RSA assumption relies on the factoring problem where we have a large number N, which is the product of two distinct prime numbers p and q. It is widely believed to be very hard to find the factors p and q.

The secret key puts the digital signature onto the message. The public key can verify the signature, which cannot be forged and guarantees non-repudiation. We need the following algorithms to create a "textbook" RSA algorithm [21].

Key Generation $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$

1. The first step is to generate two λ-bit prime numbers p and q.

2. This is followed by calculating $N \leftarrow p \times q$ and the Euler phi function $\phi(N) = (p-1)(q-1)$.

3. We select a small exponent e, such that e is coprime to $\phi(N)$ and $1 < e < \phi(N)$, meaning that the two numbers' greatest common divisor $\mathsf{gcd}(e, \phi(N)) = 1$.

4. Then, we compute d, such that $d \equiv e^{-1} \bmod \phi(N)$.

5. The public key $pk \leftarrow (N, e)$ is then computed and publish.
   The private key $sk \leftarrow (N, d)$ is stored safely.

Encryption $c \leftarrow \mathsf{Enc}((N, e), m)$

In order to encrypt a message m, it is necessary to compute the ciphertext $c \leftarrow m^e \bmod N$.

Decryption $m \leftarrow \mathsf{Dec}((N, d), c)$

The decryption of the ciphertext c can be achieved by calculating $\widetilde{m} \leftarrow c^d \bmod N$. The original message m is equivalent to the newly deciphered message $\widetilde{m}$ because $\widetilde{m} \equiv c^d \equiv (m^e)^d \equiv m^{ed} \equiv m \bmod N$.

Plain RSA Signatures We can use RSA signatures to provide non-repudiation to messages [6]. In order to sign a message m, the private key holder needs to use a full domain hash function $H : \{0,1\}^* \to \mathbb{Z}_n$ that maps an input of arbitrary length to a random element in $\mathbb{Z}_n$. We need three algorithms for this: $\mathsf{Gen}(N)$, $\mathsf{Sign}(d, m)$ and $\mathsf{Vrfy}((N, e), m, s)$.

1. The key-generation algorithm $(pk, sk) \leftarrow \mathsf{Gen}(N)$ which is the same to above.
2. The signing algorithm $s \leftarrow \mathsf{Sign}(d, m)$ takes the decryption exponent d and the message m as input and outputs the signature s, such that $s \leftarrow H(m)^d$.
3. The verification algorithm $\mathsf{Vrfy}((N, e), m, s)$ which verifies the signature by calculating if $s^e \equiv H(m) \bmod N$.

This scheme ensures that the signer will know the content of the message once they sign it. However, this is not always in the interest of the message sender, as discussed in Chapter 2.8. One scheme that enables the signer to sign a message without seeing its content is blind digital signatures.

2.5 Blind digital signatures

Chaum first introduced blind digital signatures [8]. He proposed a new cryptographic payment protocol with three main principles in mind:

1. Payments must be private. No third-party should be able to determine a transaction's amount, destination, and time.
2. The possibility of providing proof of payment and unveiling the payee involved in a transaction whenever necessary.
3. A governing body with the ability to halt the usage of payments because of stolen assets.

To illustrate the system, we consider a donation scheme in which the donors want their payment amount to remain confidential with respect to a third-party, while ensuring the payment receipt. Additionally, only trustee-verified donations are allowed. Therefore, each donor deposits their coins in a special carbon paper[1] lined envelope. On the exterior of the envelope, the donor writes his signature. The donor then places the special envelope in a second, regular envelope with the trustee's address and sends it to them. The trustee removes the outer envelope, verifies the donor's signature, and signs the outside of the carbon-lined paper envelope. This results in a signature on the inside of the envelope. The trustee-signed envelope is then placed in a new outer envelope and returned to the donor.

As the trustee has not opened the envelope, they are unaware of the payment amount. However, the donor now possesses an authorized donation sheet with a valid signature from the trustee. The donor can remove both envelopes and send the payment with the signature to the trustee without a return address. The trustee can display all the payments and signatures, enabling each donor to verify their payment by looking for unique identifiers, such as fiber patterns on their sheet.

Blind digital signatures can be helpful when the message needs to be signed, but the content of the message should remain hidden. We want to define abstract algorithms that we can use in Chapter 3 and 5 that represent the blind digital signature protocol without specifying a particular underlying technology. Therefore, we define the following algorithms:

1. The key generation algorithm $(sk, pk) \leftarrow \mathsf{Gen}(1^\lambda)$ that uses a security parameter N as input and outputs the secret key sk and a public key pk.

[1] This is a special type of paper where it creates a copy simultaneously once written on them.

2. The hashing function $f \leftarrow H : \{0,1\}^*$ that maps an input of arbitrary length to an element of fixed length. It inputs the message m and outputs the hashed value f of the message.

3. The blinding algorithm $f' \leftarrow \mathsf{Blind}(m, b, pk)$ takes a message m, the blinding factor b and the public key pk as input and outputs a blinded value f'.

4. The signing algorithm $s' \leftarrow \mathsf{Sign}(sk, m)$ takes a message m and a secret key sk of the signer as an input and outputs the signature s'.

5. The signature unblinding algorithm $s \leftarrow \mathsf{Unblind}(s', b)$ that takes the blinded signature s' and the blinding factor b as input and outputs the unblinded signature s.

6. The verification algorithm $v \leftarrow \mathsf{Vrfy}(s, m, pk)$ takes the signature s, the original message m and the public key of the signer pk as an input and outputs the boolean value $v \equiv 1$, if the signature is valid and 0 otherwise.

The content of the message will remain hidden from the signer. However, if Alice sends a message to Bob for a blind signature, Bob will not know the content of the message but will know that it originated from Alice. One scheme that can hide the origin of the message is through the use of a mix network.

2.6 Mix Networks

Mix networks [7] improve the privacy of digital communication by using cryptographic techniques to ensure that the transmission of messages and, in our case, payments are anonymous [3]. For the network to function, it requires at least one honest party and a sufficiently large enough player count. Under these conditions, tracking the message's origin and destination becomes nearly impossible. Mix networks are similar to parlor games. Each player at the table takes a deck of cards from the right person and shuffles it without revealing the shuffling to anyone.[2] This leads to an unknown re-ordering. Messages are encrypted in a mix network, making it impossible for various nodes to learn the content of the messages and to link the sender to the recipient.

These networks are important when we discuss eCash 2.0 in Chapter 3, where the central bank keeps a list of unspent coins instead of a spent list. In order to demonstrate mix networks better, let us consider a scenario in which a customer, Alice, and the central bank are engaged in a CBDC scheme. In this scenario, Alice creates a randomly generated serial number sn, which represents the coin, and sends it to the central bank, thereby enabling the central bank to append the serial number to the unspent CBDC list to make the coin spendable in the future. The randomly generated serial number is a unique identifier. If the central bank receives this serial number from a different person in the future to verify its authenticity, the central bank knows that Alice bought an item from them. To avoid this, we can use a mix network that enables Alice sender-anonymity. This allows Alice to transmit the serial number to the central bank without it being associated with Alice.

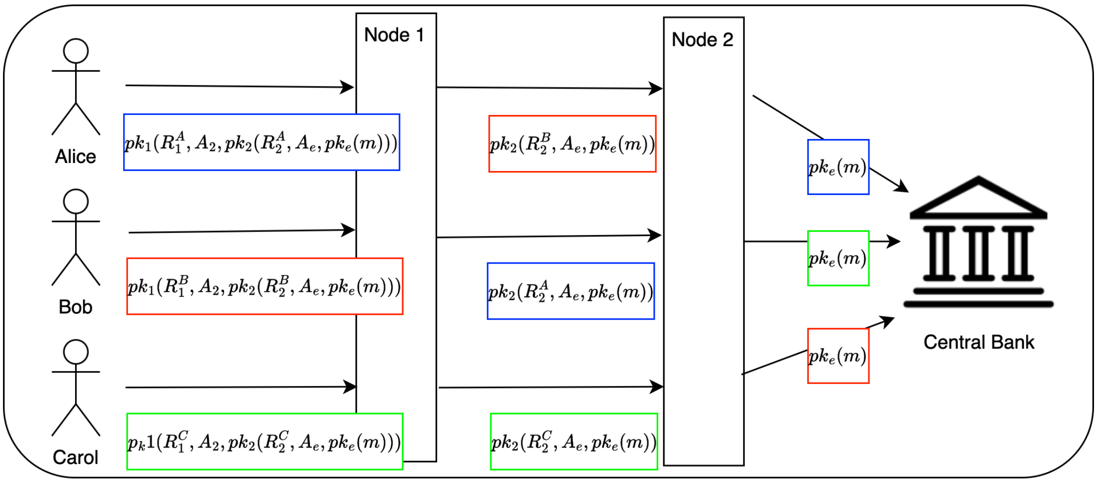
Sender-anonymity Alice is a participant in such a mix network. Alice, for example, could choose to implement a sequence of mixes, whereby the coin is directed from Alice $\rightarrow \mathsf{Node}_1 \rightarrow \mathsf{Node}_2 \rightarrow$ central bank. In order to achieve this, Alice requires two random numbers, R_1 and R_2, which are necessary to achieve non-deterministic encryption. Additionally, she needs the public keys pk_1 and pk_2, as well as the addresses A_1 and A_2, to associate the nodes. Finally, Alice needs the public key pk_e and the address A_e from the central bank.

We can envision the operation of a mix network as a sequence of steps, whereby the input into the i-th node is initially encrypted with the public key pk_i of the i-th node. The node will decrypt it with its

[2] If there were no honest players and everyone recorded their shuffles, the final order could be deduced. The more honest and independent the players are, the more difficult it is to track the messages.

private key, thereby gaining access to the random number R_i, which the node discards, and the address A_{i+j} of the next node. The current node transfers the remaining portion of the sequence to the next node, which can be identified via the address A_{i+j}. The mix network repeats these steps until it finally lands at the last destination where the message m is decrypted.

If we go back to our example involving Alice and the central bank, a simplified sequence involving two nodes has been selected by Alice that looks like this: $pk_1(R_1^A, A_2, pk_2(R_2^A, A_e, pk_e(m)))$. Upon receipt of the encrypted message $pk_e(m)$ by the central bank, it can decrypt the message with its private key sk and thereby gain access to the serial number without knowing the point of origin [7]. Figure 2.2 shows this principle with three involved characters and two nodes. The coloring of the frame indicates the point of transaction origination.



Node 1
Node 2
Alice
Bob
Carol
$pk_1(R_1^A, A_2, pk_2(R_2^A, A_e, pk_e(m)))$
$pk_1(R_1^B, A_2, pk_2(R_2^B, A_e, pk_e(m)))$
$p_k1(R_1^C, A_2, pk_2(R_2^C, A_e, pk_e(m)))$
$pk_2(R_2^B, A_e, pk_e(m))$
$pk_2(R_2^A, A_e, pk_e(m))$
$pk_2(R_2^C, A_e, pk_e(m))$
$pk_e(m)$
$pk_e(m)$
$pk_e(m)$
Central Bank

Figure 2.2. Mix network involving three participants and two nodes

2.7 Quantum Computers and Cryptography

Certain quantum computer algorithms can solve certain problems faster than a conventional computer. They employ special properties, such as superposition, which allows a quantum bit to exist simultaneously in a zero, one, or a combination of both states. This threatens many cryptographic protocols currently in use. For instance, people could use these computers to potentially break asymmetric cryptographic protocols, such as the Diffie-Hellman and RSA protocols. Factoring large numbers or solving discrete logarithms would make the encryption of these protocols ineffective. This is problematic because traffic over public channels can be stored and decrypted using such a computer once it is available. The term “post-quantum cryptography” describes algorithms that are resistant to attacks by quantum computers but can be computed by classical computers.

In the following chapter, we will examine the reasons for using blind digital signatures in a payment scheme and why digital signatures are not sufficient for the payer to remain anonymous.

2.8 A digital payment system

We can construct a payment system using RSA with signatures, as the RSA allows for encrypted messaging, and the signature ensures non-repudiation. The following section details such a system and explains the potential shortcomings of relying solely on digital signatures.

A privacy-less digital payment system We consider a hypothetical payment system that imitates a simplified CBDC solution. In this system, only the central bank is involved in a transaction between a customer, Alice, and a merchant, Bob. The central bank creates spendable coins by signing them with its private key and holds a list of all the spent coins.

We consider a hypothetical payment system that imitates a simplified CBDC solution. In this system, only the central bank is involved in a transaction between a customer, Alice, and a merchant, Bob. The central bank creates spendable coins by signing them with its signing function and holds a list of all the spent coins.

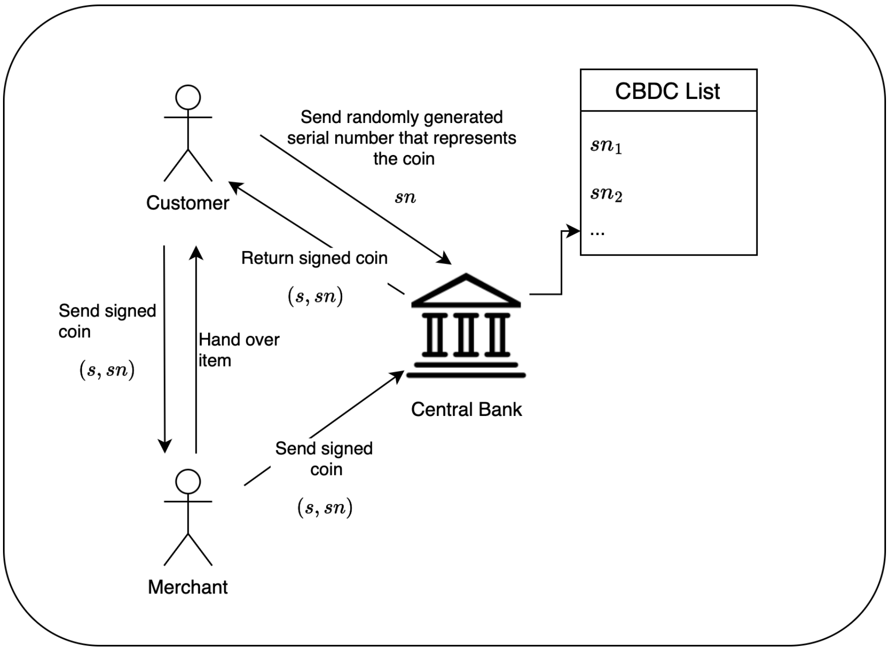


Figure 2.3. CBDC transaction in a privacy-less payment system

Alice must first obtain CBDC to pay Bob for an item. To do so, Alice first creates a randomly generated serial number sn, which represents the coin and transmits it to the central bank. The central bank reduces Alice's account balance by the coin's value and finally signs the coin with its secret key sk to create the digital signature $s \leftarrow \mathsf{Sign}(sn, sk)$. The central bank returns the signed coin (s, sn) to Alice. Alice now possesses enough CBDC for the wanted item and sends the signed coin (s, sn) to Bob. In order to verify the payment made by Alice and to redeem the funds, Bob transmits the signed coin (s, sn) to the central bank. The central bank first verifies the correctness of the signature by verifying whether the public key pk corresponds to the secret key sk and, therefore, checks the output of the binary value $v \leftarrow \mathsf{Vrfy}(pk, s, sn)$ for a true value. The central bank then credits Bob's account and adds the coin sn to the spent list. Once Bob receives the payment from the central bank, he hands the item to Alice. Figure 2.3 illustrates a transaction involving Alice and Bob.

Privacy concerns In this simplified CBDC example, both Alice and Bob lack privacy because of the randomly generated serial number sn, which is a unique identifier that connects Alice to the coin. As the serial number is transmitted from the central bank to Alice, from Alice to Bob, and finally from Bob to the central bank, the central bank knows that Alice has purchased an item from Bob including transaction amount. The serial number serves as the unique identifier for the transaction, and given that the serial number is transferred to the central bank after every transaction to redeem the funds, the central bank has complete knowledge of all transactions. One potential solution to privatize the transaction is through the use of blind digital signatures.

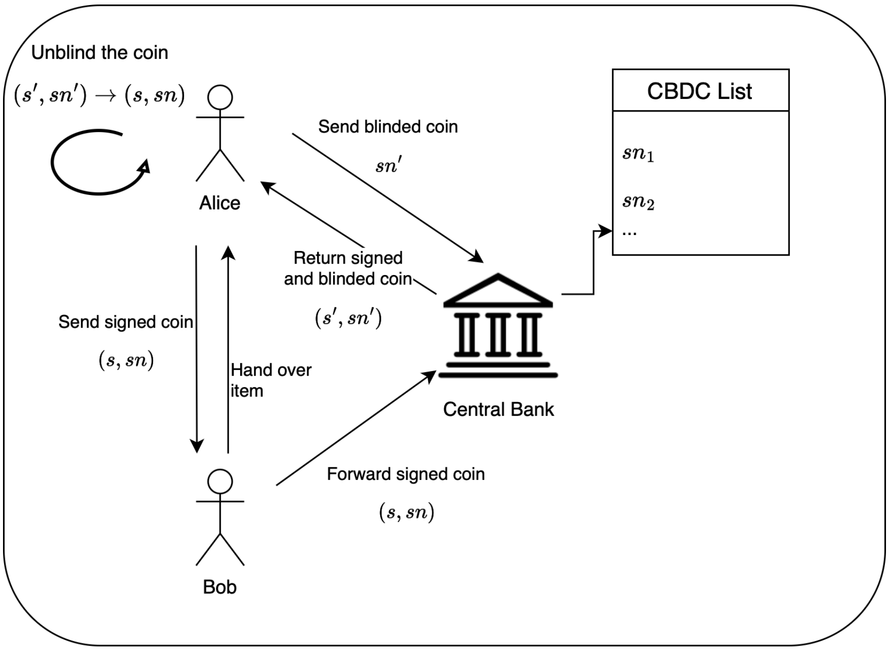


Figure 2.4. CBDC transaction in a private payment system

Blind Digital Signatures based on RSA This section explores the implementation of blind digital signatures in a payment system. We can use these two technologies to create a simplified CBDC solution with fewer privacy concerns than the previously discussed privacy-less digital payment system. Alice creates a randomly generated serial number sn of the coin and blinds it with a blinding factor b, such that the uniquely identifying serial number is blinded $sn' \leftarrow \mathsf{Blind}(sn, b, pk)$. Alice then sends the coin to the central bank. The blinding allows the central bank to sign the serial number $s' \leftarrow \mathsf{Sign}(sn', sk)$ without gaining information about the unique identifier of the coin. A more formal definition follows:

1. Alice generates a random blinding factor b and calculates the blinded coin $sn' \equiv snb^e \bmod N$ using the respective provided public key (e, N) for the value x of the coin issued by the central bank.

2. Alice sends the blinded coin sn' to the central bank.

3. The central bank signs the blinded coin with its private key d by computing the blind signature $s' \equiv (sn')^d \bmod N$.

4. The central bank returns the signed and blinded coin (s', sn') back to Alice.

5. Alice unblinds the coin by computing $s \equiv s'b^{-1} \equiv (sn')^d b^{-1} \equiv sn^d b^{ed} b^{-1} \equiv sn^d b b^{-1} \equiv sn^d \bmod N$ and stores the coin.

The central bank issues different key pairs depending on the value x of the coins. The corresponding public keys are published and used by the customers. The keys can have an expiration date, which means that spending the coins is only possible for a certain amount of time. This has two main benefits. Firstly, it increases the system's efficiency by eliminating expired entries from the stored list. Secondly, it has security advantages since the central bank does not have to watch out for vulnerabilities in the security system of the expired private keys.

Figure 2.4 shows a comparable payment interaction to the privacy-less digital payment system. We achieve transaction privacy in this example because Alice blinds the serial number during CBDC creation and then unblinds it. Bob only receives the signed serial number coin (s, sn). Therefore, the central bank cannot draw any connection between Alice and the incoming coin (s, sn) from Bob because the serial number sn and the blinded serial number sn' are not connectable.

Chapter 3

Central Bank Digital Currencies using eCash

This chapter introduces central bank digital currencies based on different eCash versions.[1] The first two sections establish the privacy standards central banks aim for and provide a brief overview of account-based versus token-based CBDCs. In the later sections, we take a closer look at both online and offline eCash.

3.1 Privacy

The Bank for International Settlements published a report about the foundational principles and core features of CBDCs in collaboration with central banks of Canada, EU, Japan, Sweden, Switzerland, England, and USA [4]. From this report, we can take the privacy objectives that central banks try to balance: "Full anonymity is not plausible. While anti-money laundering and combating the financing of terrorism (AML/CFT) requirements are not a core central bank objective and will not be the primary motivation to issue a CBDC, central banks are expected to design CBDCs that conform to these requirements (along with any other regulatory expectations or disclosure laws). For a CBDC and its system, payments data will exist, and a key national policy question will be deciding who can access which parts of it and under what circumstances. Striking this balance between public privacy (especially as data protection legislation continues to evolve) and reducing illegal activity will require strong coordination with relevant domestic government agencies (eg tax authorities)". The challenge is to achieve a balance between user privacy and the detection of illicit activities. While we believe that this definition is vague, it is a definition that many central banks agree upon. Therefore, we can analyze whether eCash successfully implements these objectives.

In the next section, we dive into the key differences between account-based and token-based CBDCs. This distinction will be important in Chapter 5 when we examine the potential synergies and conflicts between the Digital Euro and eCash.

3.2 Account-based versus token-based CBDCs

A central bank typically chooses between two distinct design types for a retail CBDC. These are either account-based or token-based CBDCs. Both versions correspond to the two already existing types of central bank money, the central bank as a reserve is the account-based version, while banknotes represent the token-based system [10]. In the account-based version, the transfer only affects the balance of a payer and a payee by reducing or increasing the respective bank account. In contrast, in a token-based CBDC, a coin is transferred from the payer to the payee. The key distinction between these two is the information

[1] Chaum introduced several versions of eCash. A website exists at https://chaum.com/ecash/ detailing the history of eCash.

conveyed by the system. In the account-based version, a bank account holds the transaction history with the user's spending and deposit history. In a token-based system, the token contains the information regarding value and the token's issuer.

3.3 SNB Work in progress

This section looks at a collaborative report between the Swiss National Bank[2] and Chaum, Grothoff, and Moser that explores a potential CBDC solution [11]. The proposed design is a token-based, software-only CBDC without a distributed ledger. The report argues that distributing the central bank's ledger provides no tangible benefits and could increase transaction costs. Besides eCash, which uses blind digital signatures, the CBDC builds upon GNU Taler, a free software program released under the GNU Affero General Public License by the GNU project, which uses EdDSA signatures. The paper highlights the necessity of establishing the CBDC under the Free/Libre and Open Source Software (FLOSS), as reliance on vendors would likely hinder the adoption. Using FLOSS can achieve integration and interoperability between different providers much quicker and easier because all parties can view the details and customize the software according to their standards. Moreover, a FLOSS approach enables the central bank to provide transparency, accountability and allows for peer-reviewing of the code, which can help to find potential security vulnerabilities. The FLOSS approach also enables users to verify the privacy standards of the solution. The aim is to provide transaction privacy, which has three important features.

1. Firstly, it protects users from government scrutiny and the misuse of surveillance.
2. Secondly, Payment Service Providers (PSPs), such as commercial banks, usually collect and possess detailed transaction data for each user, including the amount and recipient of the payment.
3. Thirdly, the user is protected from a merchant's failure or neglect of customer data protection. For example, a merchant should not be able to reveal identity disclosing information about the customer.

We analyze the privacy standards regarding these three transaction privacy features. The following two paragraphs explore how customers can acquire and spend CBDC. The Figures 3.1 and 3.2 illustrate the logical steps involved, while the accompanying text provides additional mathematical notation using RSA.

Exchange traditional money into CBDC

To obtain x amount of CBDC, the customer logs into their commercial bank account and looks up the correct public key (e, N) provided by the central bank according to the value x. Each coin denomination has a separate public and private key. The customer computes a key pair $(sk_c, pk_c) \leftarrow \mathsf{Gen}(1^\lambda)$ of the coin consisting of the private key sk_c and the corresponding public key pk_c and chooses a blinding factor b. The customer hashes $f \leftarrow \mathsf{H}(pk_c)$ the coin's public key and blinds $f' \leftarrow fb^e \bmod N$ the hash and sends it together with an authorization from the customer to withdraw the coin to the commercial bank over an encrypted channel. The commercial bank debits the customer's account and conducts internal processes, such as digitally authorizing the request with the bank branch's internal digital signature. Finally, the commercial bank forwards the request and the blinded coin to the central bank. The central bank reduces the commercial bank account according to the amount x withdrawn by the customer. The central bank then signs the blinded coin $s' \leftarrow (f')^d \bmod N$ with the correct private key d according to the public key (e, N) of value x and finally sends the signed and blinded coin (s', f') back to the commercial bank. The commercial bank forwards the signed and blinded coin (s', f') to the customer. The customer uses the previously chosen blinding factor b to unblind the signature $s \leftarrow s'b^{-1} \bmod N$,

[2] The paper disclaims that the working paper describes research in progress and the views of Chaum, Grothoff, and Moser do not necessarily represent the views of the SNB.

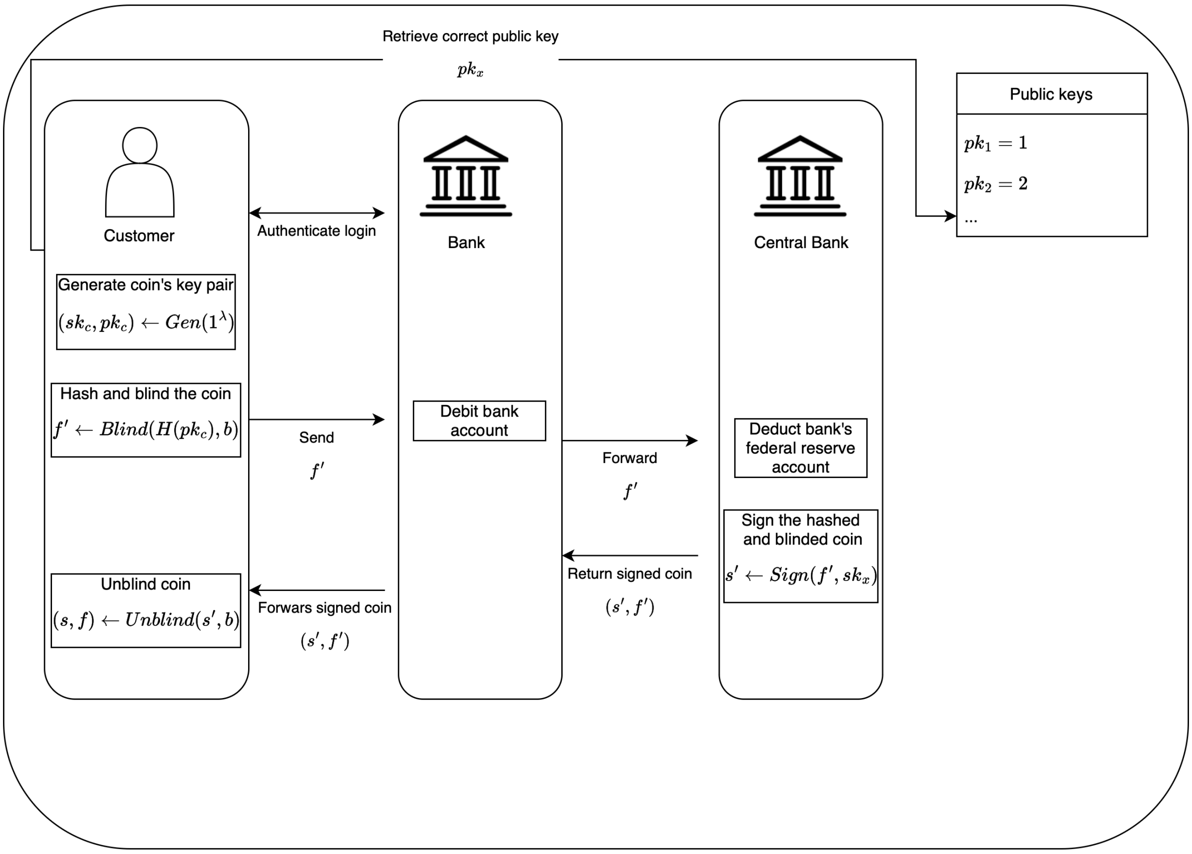


Figure 3.1. Exchange traditional money into CBDC

creating a newly minted coin (s, f), which the customer stores on their phone in a wallet. Figure 3.1 represents these steps involving a customer, the commercial bank, and the central bank.

Spending and Depositing CBDC

To make a payment, a customer sends the coins to the merchant using the merchant's bank account number. The proposal also mentions that the customer and merchant negotiate a business contract, but for this thesis, we disregard the contract. The merchant verifies the coin's signature over the contract and the central bank's signature s over the hash f corresponding to the coin's public key pk_c with the respective public key pk_x of the central bank for the correct value x. The merchant then forwards the verified coin (s, f) and their account details to their commercial bank. The commercial bank checks if the merchant is one of their customers and transmits the coin to the central bank. The central bank validates the signature by checking $v \leftarrow \mathsf{Vrfy}(s, f, pk_x)$ for a positive value, and checks the spent CBDC registry for previous usage of the coin's hash f. If the transaction is valid, the central bank adds the coin to the spent registry, credits the commercial bank's account, and sends a confirmation to it. Upon receipt of the confirmation, the commercial bank credits the merchant's bank account and notifies them of the successful transaction[3]. Figure 3.2 represents these steps involving a customer and merchant, the commercial bank, and the central bank.

[3] The central bank needs to keep its private keys private. If they ever get compromised and coins are lost, the customers can request a refund of unspent coins without compromising privacy. In such a case, the central bank announces a key revocation via its API, which all wallets detect. The user can then disclose the coin's public key pk_c, the central bank's signature s, and the blinding factor b to the central bank. This allows the central bank to verify the withdrawals and refund unspent values.

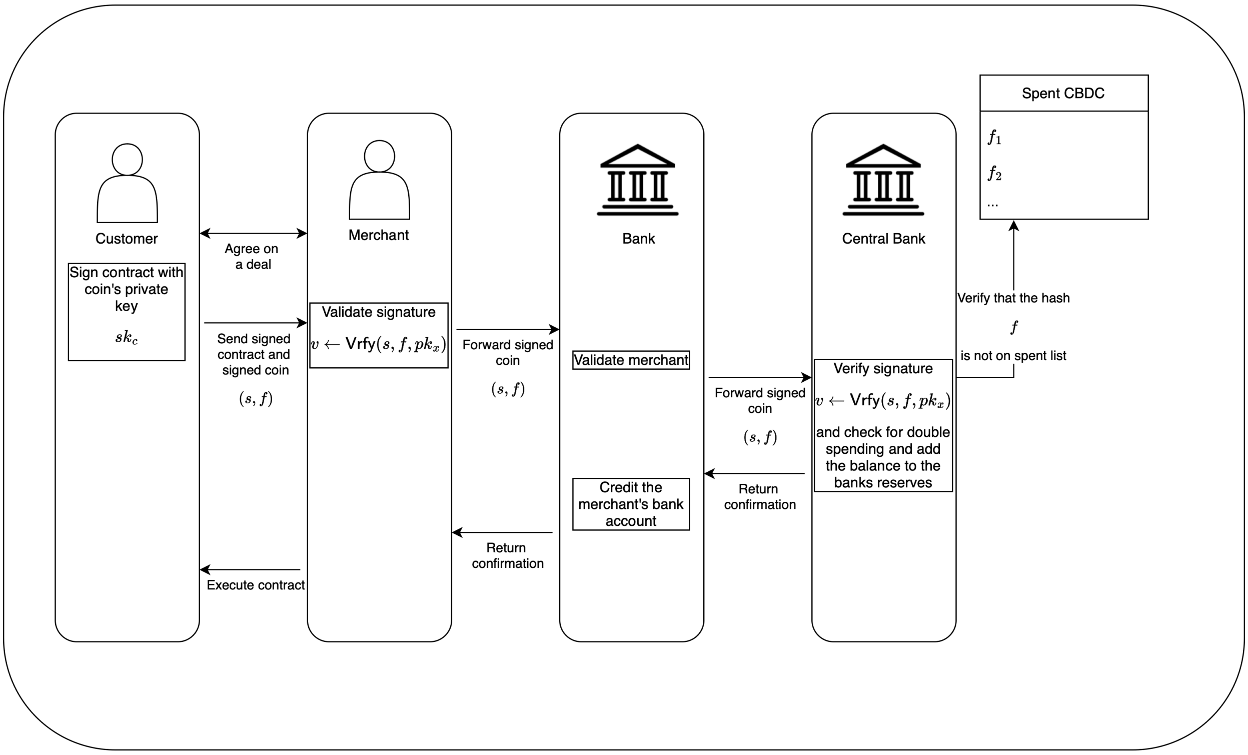


Figure 3.2. Spending and Depositing CBDC

Analysis The goal is to preserve transaction privacy, which has three important features. Therefore, we analyze whether an eCash-based system safeguards these principles. To do this, we look at the three actors involved: the central bank, the commercial bank, and the merchant. If these actors preserve the customer's privacy, the CBDC achieves transaction privacy.

1. **Central Bank** (Government) scrutiny and excessive surveillance are impossible with such a system. When a customer exchanges traditional money for CBDC, the central bank only receives a hashed and blinded coin from the commercial bank. During the redemption of CBDC into traditional money, the central bank only receives a signed and hashed coin from a commercial bank. In neither case can the central bank infer the identities of the customer or merchant solely from this information. While a hashed coin's public key is a unique identifier, the central bank has never associated this hash with a customer because, upon redemption, it only interacted with the hashed and blinded version of it.

2. **Banks** (PSPs) act as an intermediary between the customer, or the merchant, and the central bank. However, for the same reason as above, the PSP cannot link a transaction from a merchant to a specific customer. This is because the PSP only receives the hashed and blinded coin during CBDC generation from the customer and the signed and hashed coin during the redemption by the merchant. The unlinkability of these two data points prevents the PSP from exploiting customer data.

3. **Merchants** only receive the signed contract and the signed and hashed coin. Even if the merchant leaks this data, linking the hashed coin to the hashed and blinded coin is impossible. Consequently, the privacy of the customer remains protected.

Transaction privacy refers to the customer's ability to make transactions without revealing their identity. The blinding and unblinding allow the customer to remain anonymous, and therefore, eCash satisfies transaction privacy. If we take the privacy standards of Chapter 3.1, we believe that the system successfully balances public policy objectives, such as AML/KYC/CFT conducted by the commercial bank, and robust privacy standards due to the successful implementation of transaction privacy. In the next chapter, we examine eCash2.0, where we also analyze potential threats to the system in addition to privacy.

3.4 Project Tourbillon

The project Tourbillon [3] presents prototypes of a CBDC solution based on eCash. The goal is to balance three features simultaneously: privacy, security, and scalability. The project aims to assess to which degree an eCash-based CBDC system implements these features.

1. **Privacy** The Bank for International Settlements (BIS) wants to maintain payer anonymity for their CBDC prototypes. The concept of payer anonymity enables the protection of user privacy and the realization of public policy goals. From the customer's perspective, their payment does not reveal any personal information to any party involved, including the commercial bank, the central bank, and the merchant.

   Merchants have less privacy than customers because they must reveal their identity to the customer and their own commercial bank during a payment. The central bank has no access to any private information, as the commercial bank acts as an intermediary between users and the central bank. However, the central bank can observe the amount of currency in circulation on an aggregated level, as the central bank maintains a list of spent or unspent currency. The aim is to provide payer anonymity while making illicit payments as challenging as possible. It is important to note that the conservation of the consumer's privacy is only within the system itself. External factors can breach this confidentiality. A customer can, for example, choose to reveal their identity when showing an identity card to a merchant to purchase alcohol or using a Cumulus point card at Migros.

2. **Security** The goal is to use cryptography to ensure the confidentiality of payment data with respect to third parties such as the central bank, the commercial bank, and the merchant. The system should also provide integrity by preventing double spending and counterfeiting.

3. **Scalability** For a payment system to even be considered an alternative to cash, it needs to handle millions of concurrent payment transactions. The system needs to be able to adapt to sharp increases in demand, such as during lunch or events, without altering transaction time, quality, and cost.

Project Tourbillon introduces two prototypes, EC1 and EC2, to compare the tradeoffs of the three features against each other. The initial implementation uses eCash1.0 [10], and the latter uses eCash2.0 [11]. Both prototypes utilize a Tourbillon app and build upon existing two-tier banking systems. They developed two mobile apps, one for the consumer and one for the merchant. The consumer app contains a digital wallet, enabling the consumer to create payments, withdraw, and hold CBDC. For simplicity, the merchant's app does not contain a wallet. However, the app can request, receive, and view the status of a payment. The coins have distinct denominations in powers of two. Therefore, a coin can have a value of $1, 2, 4, 8, ..., 2^k$.

3.4.1 EC1

This section analyzes the proposed EC1 solution for Project Tourbillon [3]. It is similar to the eCash version discussed in Chapter 3.3, so we only look at an overview without going into the details. In the first paragraph, we examine how a customer can transform traditional money into CBDC, and in the second paragraph, we look at how a customer can use CBDC in a transaction.

Exchange traditional money into CBDC To show the workings of EC1, we consider a customer who wishes to withdraw 15 toubies, which is the unit of account in Project Tourbillon. To do so, the customer logs into the Tourbillon app and requests the withdrawal. The algorithm generates four coins of varying values, in this case, 1,2,4 and 8, which collectively total up to 15.[4] The customer hashes and blinds the coins, making the unique identifier unreadable, and sends the blinded coins to the commercial bank.

Subsequently, the commercial bank blocks 15 toubies of the customer's account and forwards the blinded coins to the central bank. The central bank debits 15 toubies from the commercial bank's central bank accounts and signs the blinded coins. It then sends the signed and blinded coins back to the consumer via the commercial bank. The commercial bank also unfreezes the previously blocked 15 toubies and debits the consumer's account by 15. The consumer's app unblinds the CBDC coins and stores them in their wallet. This self-custody ensures that neither the commercial nor the central bank is aware of the type of CBDC held by the customer, thus preserving customer privacy. The customer can spend the CBDC without anyone being able to link the coins to them.

Spending and Depositing CBDC With the 15 toubies in their account, the customer wishes to purchase an item worth 10 toubies from a merchant. The merchant initiates the payment by registering a pending transaction and generating a QR code containing all the relevant information, such as the amount and the deposit account number. The customer scans the QR code with the app and forwards the coins, in this case, 2 and 8, to the merchant's commercial bank. The commercial bank links the payment to the pending transaction and forwards the coins to the central bank. The central bank verifies the signature and checks that the coins are not on the list of spent coins. Once this verification is complete, the central bank redeems the coins, adds them to the spent coins list, and credits the commercial bank's reserves. Finally, it sends a confirmation to the commercial bank, which credits the merchant's deposit account. The remaining coins of the customer, in this case, 1 and 4, are rebalanced such that the coin of value 4 splits into one 2 and two 1 value coins. This allows the customer to spend any amount between 1 and 5. The algorithm minimizes the number of coins a holder holds while ensuring that sufficient change is always available for any transaction.

The EC1 prototype uses eCash1.0 technology. However, an updated version of eCash exists, eCash2.0. The following section examines the prototype that builds on it and compares it to EC1.

3.4.2 EC2

EC2 builds on EC1 and differs from it in only a few aspects. In EC1, the central bank records the unique identifiers associated with CBDC coins upon redemption. Consequently, the central bank maintains a register of spent CBDC coins. In EC2, the central bank records the coins upon issuance by maintaining a list of unspent CBDC coins, identifying them via the hashed coins. However, keeping a record of the unique identifier violates the goal of payer anonymity for the same reasons discussed in Chapter 2.8. To address this issue, EC2 proposes the usage of a mix network.

The mix network batches several hash function images supplied by different customers together. The respective commercial bank allows each customer to enter a batch. The first node processes the input batch, which decrypts and randomly orders them. The output batch forwards the remaining input batch to the next node, which takes it as its input batch. This goes on until the last node. EC2 utilizes teams of 5 nodes chosen at random. One team is responsible for only one batch, and all teams first process, mix, and then decrypt the data before handing it to the next team.

Analysis about the mix network The BIS argues that in their mix networks: "The teams are formed randomly on a blockchain". That is the only part where the paper mentions anything about blockchain.

[4] It is advantageous to create four coins because the consumer will always have the correct amount of change for any expense between 1 and 15.

We are interested in knowing what they mean by forming teams on a blockchain. Building their own blockchain seems highly likely because we believe that using existing blockchains like Ethereum would likely decrease speed and privacy while increasing transaction costs. However, we know that in the eCash2.0 proposal by Chaum and Moser [11], eCash can optionally be extended to a public blockchain, where users can check their coins via their hash on it. We believe it is possible to extend it and form teams on the blockchains because the hash is published on the public blockchain before it enters the unspent list of the central bank.

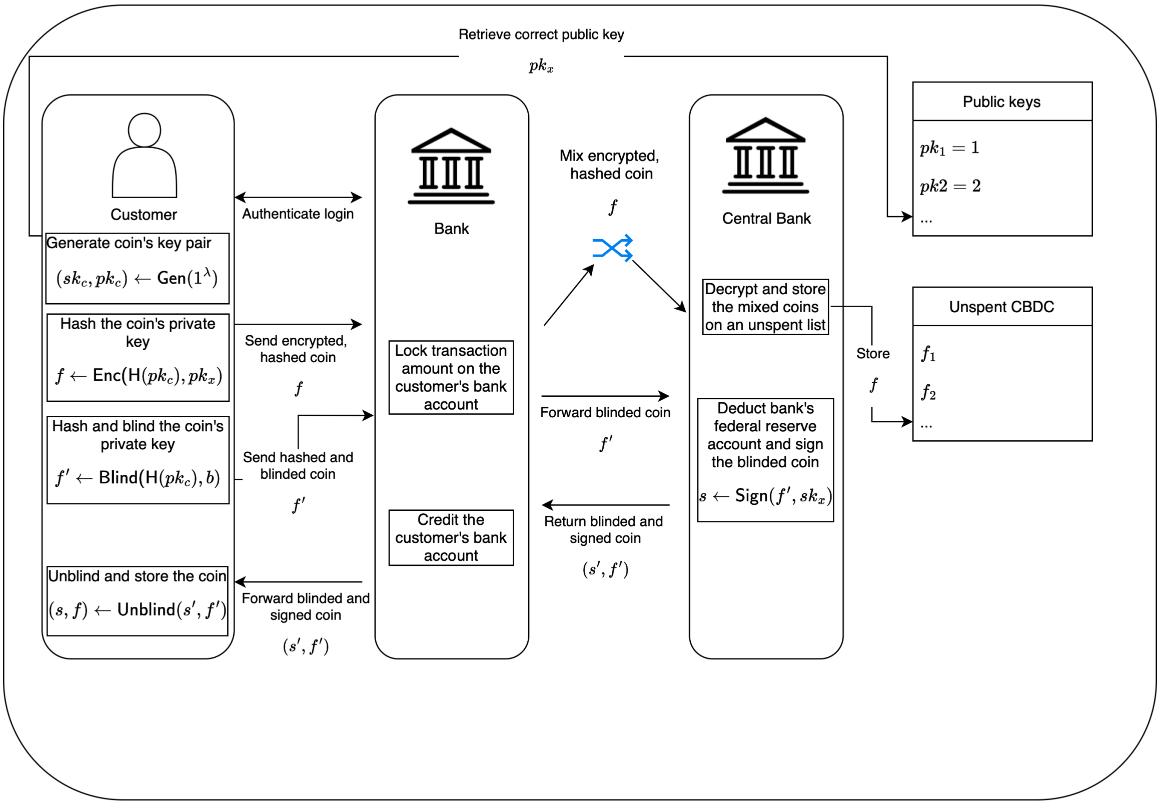
We have seen the importance of using mix networks in an eCash version that uses a list of unspent hashed coins. In the following two paragraphs, we examine the role of the mix network and unspent list more closely by looking at CBDC conversion and spending.

Exchange traditional money into CBDC First, the consumer decides on the value x to transform into CBDC. Therefore, the consumer looks up the correct public key pk_x, generates a key pair $(sk_c, pk_c) \leftarrow \mathsf{Gen}(1^\lambda)$ of the coin and a blinding factor b. The customer makes two copies of the coin, hashes and encrypts $f \leftarrow \mathsf{Enc}(\mathsf{H}(pk_c), pk_x)$ the first and hashes and blinds $f' \leftarrow \mathsf{Blind}(\mathsf{H}(pk_c), b)$ the second copy. The consumer sends both versions separately to the commercial bank, which debits the customer's account. After the debiting process, the commercial bank sends the hashed and blinded coin directly to the central bank, while it sends the encrypted, hashed coin to a mix network.

- In the Mix network, the encrypted, hashed coin f goes through five nodes before it reaches the central bank. The central bank receives the encrypted, hashed coin f, but the origin is well hidden since it went through five mixing nodes. The central bank can then decrypt the hash with the corresponding private key sk_x and put it onto the unspent CBDC list.
- The hashed and blinded coin is verified and signed $s' \leftarrow \mathsf{Sign}(f', sk_x)$ by the central bank using its private key sk_x of the corresponding public key pk_x, and returns the signed and blinded coin (s', f') back to the commercial bank.

The commercial bank receives the signed and blinded coin (s', f') and forwards it directly to the customer, who can unblind the coin's signature $(s, f) \leftarrow \mathsf{Unblind}(s', b)$. After this stage, the coin is ready to be used for payment. Figure 3.3 shows these steps visually where a customer gains CBDC from the central bank through the commercial bank.

Spending and Depositing CBDC Once a customer and a merchant agree on a deal, the merchant initiates a pending transaction order at their commercial bank and generates a QR code. The customer scans this code using the Tourbillon app, signs the transaction, and sends the CBDC coin (s, f) to the merchant's commercial bank. The bank links the incoming coin to the pending transaction order and forwards it to the central bank for verification. The central bank validates the signature of the coin using its public key $v \leftarrow \mathsf{Vrfy}(s, f, pk_x)$ and also checks that the coin is on the unspent list by searching the registry for the correct hash f. If successful, it deletes the entry of the hash f on the unspent list, credits the commercial's reserve, and notifies the commercial bank of successful completion. The commercial bank finally credits the merchant's account, and the merchant hands over the goods to the customer. Figure 3.4 visually shows these steps involving the customer, the merchant, the commercial bank, and the central bank.



Retrieve correct public key
pk_x
Customer
Authenticate login
Bank
Mix encrypted, hashed coin
f
Central Bank
Public keys
$pk_1 = 1$
$pk2 = 2$
...
Generate coin's key pair
$(sk_c, pk_c) \leftarrow \mathsf{Gen}(1^\lambda)$
Hash the coin's private key
$f \leftarrow \mathsf{Enc}(\mathsf{H}(pk_c), pk_x)$
Send encrypted, hashed coin
f
Lock transaction amount on the customer's bank account
Decrypt and store the mixed coins on an unspent list
Store
f
Unspent CBDC
f_1
f_2
...
Forward blinded coin
f'
Hash and blind the coin's private key
$f' \leftarrow \mathsf{Blind}(\mathsf{H}(pk_c), b)$
Send hashed and blinded coin
f'
Deduct bank's federal reserve account and sign the blinded coin
$s \leftarrow \mathsf{Sign}(f', sk_x)$
Return blinded and signed coin
(s', f')
Credit the customer's bank account
Unblind and store the coin
$(s, f) \leftarrow \mathsf{Unblind}(s', f')$
Forward blinded and signed coin
(s', f')

Figure 3.3. Withdraw CBDC eCash 2.0

Analysis about the unspent list The central bank uses its private keys to sign the coins, transforming them into CBDC. If these private keys are compromised, an adversary could create their own CBDC by using them to sign their coins. However, the unspent list presents a challenge to the adversary. While a valid signature confirms a coin's authenticity, they are only usable if the hash f of that coin is on the unspent list. This requires an adversary to cooperate with a malicious commercial bank, which would need to send the adversary's coin hashes to the mix network without deducting their account balance. While this scenario is unlikely, it shows that the private keys of the central bank are an integral part of the system's security. We believe the unspent list offers limited additional protection against illicit money creation compared to EC1. The counterfeited CBDC would be extremely hard to detect in such a scenario.

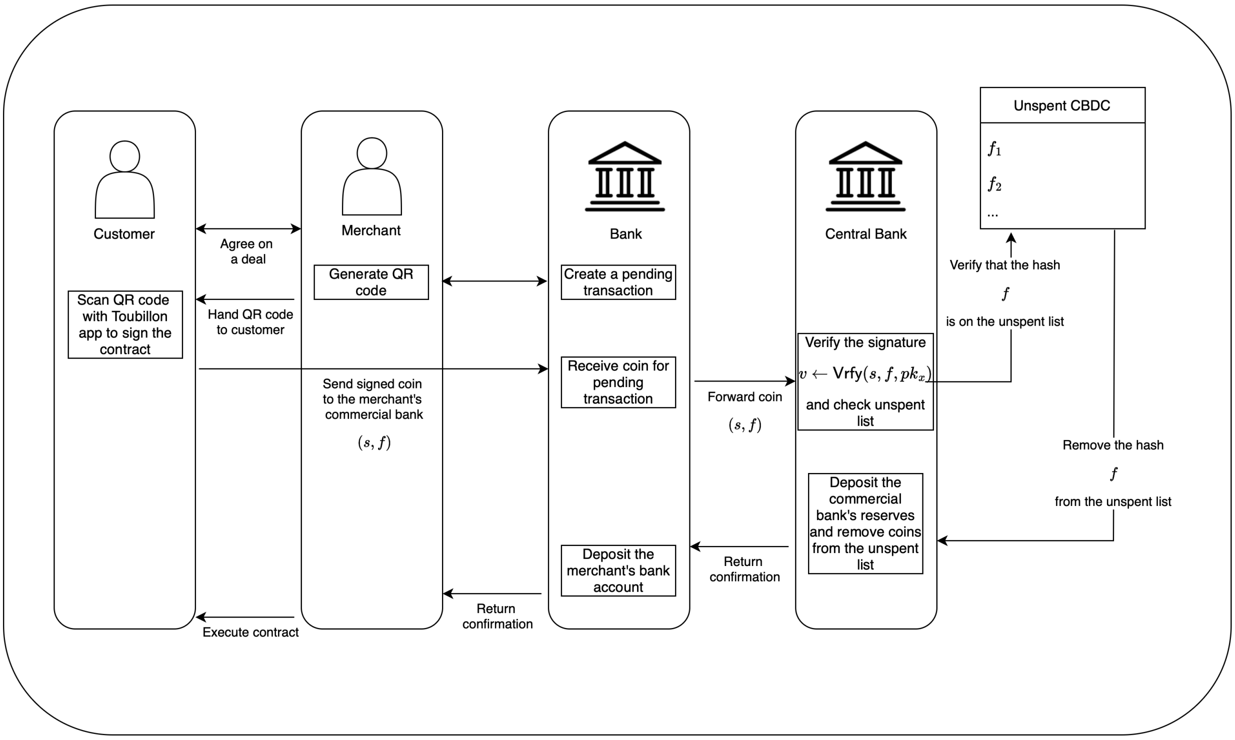


Figure 3.4. Spending and depositing CBDC eCash 2.0

Analysis about the coin's value We do not know how, apart from the customer, any other party can determine the value of a coin. The central bank, for example, signs the coin with the correct private key sk_x according to the customer's chosen public key pk_x. It would be interesting to know how the central bank knows which private key sk_i for $i \in \{1...n\}$ to use or how a merchant can verify the coin's value. We believe that one potential way would be metadata.

3.4.3 Comparing EC1 and EC2

While double spending is detectable in EC1, the growth of the unspent list is much smaller than the registry of spent coins. This also makes the total amount of CBDCs in circulation transparent, and therefore, statistical outflows should be detectable much quicker. For example, if the list suddenly has, on average, 10% more entries and eliminations per day, then the central bank knows that the signing keys might have been compromised. However, EC2 introduced a potential vulnerability in payer anonymity. An attacker with access to information on payments and withdrawals on the mix network batches might be able to identify a consumer's identity and spending patterns.

Another difference is the approach to reaching quantum resistant cryptography. EC1 uses quantum resistant hash functions, while EC2 uses lattice-based blind signature, which relies on the believed hardness of lattice problems.

The prototypes have a trade-off between privacy and security. On the one hand, EC1 has better privacy features, but checking for counterfeiting is only indirectly possible through an investigation into suspiciously large redemptions. However, the central bank has to guess whether redemptions are suspicious or not. EC2 is more complex, and thus, there are more possibilities for potential vulnerabilities. It is also new and thus has been potentially less audited when compared to EC1. EC2 has reduced privacy compared to EC1 because of the stored hash f. A connection between the customer's identity and the hash entry of the unspent coins would lead to a similar privacy problem, as discussed in Chapter 2.8 with the privacy-less digital payment system.

3.4.4 Discussion of Project Tourbillon

The goal of the prototypes was to make a comparative analysis of the three different points over two prototype models: (1) The preservation of privacy with regards to payer anonymity, (2) security by implementing quantum-safe cryptography and making the system counterfeit resistant, and (3) scalability.

The initial goal, maintaining payer anonymity, is possible. Designing a system capable of doing so has been achieved in both prototypes EC1 and EC2. Moreover, they have demonstrated that scaling is possible. However, using quantum-safe cryptography for security is challenging due to the computing intensive process, which leads to significant time constraints. Utilizing quantum-safe technology resulted in a reduction of performance by a factor of 200 from conventional classical cryptography and, therefore, could threaten the scalability of the system.

The BIS identifies three areas where further research and development is needed. Firstly, further development has to be achieved in the field of quantum-safe cryptography to make it easier to implement and deploy such systems. Secondly, the current suggested implementations require more efficient designs, and there is a lack of coverage of sufficient use cases. Lastly, researchers have not yet studied the viability of these systems, and they should be tested by investigating sustainable business models.

From a privacy definition perspective of Section 3.1, we believe that payer anonymity fulfills this definition. Like in eCash1.0, public policy objectives are possible because of the service provided by commercial banks. Payer anonymity does not interfere with the goal but provides good privacy protection for the customer. In the next section, we look at the offline based eCash system, which can offer several advantages, including expanding the financial inclusion for those without access to any PSP, enhancing resilience against network disruptions, and potentially increasing privacy standards.

3.5 Offline eCash 2.0

Chaum's offline eCash2.0 addresses the limitations of online CBDCs in three areas: phone unavailability, no internet connection, and payment systems disruptions, for example, during an infrastructure attack [9]. According to the paper, existing approaches are unsuitable for these three scenarios, and thus, offline CBDCs need a completely new solution. For instance, methods relying on trusted chips are susceptible to systematic security vulnerabilities and demand an electronic device. Software-only solutions do not provide strong enough assurance that a completed transaction cannot be reversed or altered. The proposed solution introduces a low-cost physical card. In the next section, we take a closer look at this card.

Physical Card Chaum's approach utilizes a physical card containing micro-glitter, making it nearly impossible to clone. The size of the physical card is comparable to a credit card. To verify the card's authenticity, the merchant can scan the micro-glitter pattern and check if they are identical to the digitally signed pattern stored by the card issuer. The card has scratch-off tiles similar to those found on lottery tickets. On each tile is a number inscribed, representing the value beneath the scratch-off area. Scratching off a tile reveals a barcode, which acts as a cryptographic key. The barcode can be authenticated by scanning it and verifying the corresponding digital signature for its value. To prevent double-spending, a merchant should not accept revealed barcodes for payments. A crucial security feature is that a barcode hidden beneath the scratch-off material is not readable without damaging (removing) the coating. It is possible to spend less than the value indicated on the tiles, with the remaining change automatically being retained on the account associated with the specific card. While the cards must be prepaid, it is possible to charge them with less money than the total value indicated by the tile's numbers. A user can increase the prepaid value online at any time, and any remaining change is transferable to the card's new balance. The system reserves the final tile of a card for closing out any unspent balance. Figure 3.5 shows an illustration of a physical card. A user can use the card for two types of payments. The first

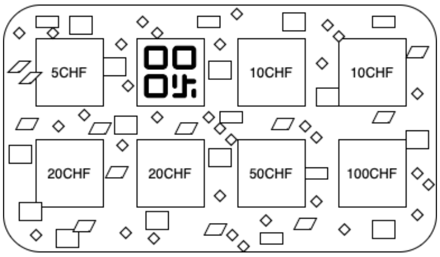


Figure 3.5. Offline eCash 2.0 payment card

payment type leaves the coated tiles on the card, with payments initiated in two different ways called offline-phone mode and card-only mode. The second option involves physically removing the tiles from the card and handing them over to the merchant and is called breakout-tile mode.

Offline-phone Mode Customers and merchants use the offline-phone mode when both have a functional smartphone but cannot establish an internet connection. The tiles remain on the card and the phones communicate over a local network, such as Bluetooth. A customer wishing to purchase an item from a merchant hands them their physical card. To verify the card's authenticity, the merchant scans the card containing a glitter pattern and obtains a digital signature from the card provider via the payer's phone over a local network. This received digital signature contains the following three elements: Firstly, it contains a Merkle tree root of the hashed barcode images, ensuring the legitimacy of each tile's barcode as a component of the signed cryptographic structure. A Merkle tree is an inverted branching structure where the root represents the hash at the top level. This root is computed from the hashes of the nodes directly below it, with each of those nodes calculated from the hashes at the level below them. This process continues until the leaf nodes are reached, representing each barcode's individual hashed values. Reversing the barcode hash to the original barcode is computationally challenging, requiring the inversion of the tree's one-way hash function. Consequently, the barcode can only be obtained by removing the coating from the tile and then scanning the barcode. Secondly, the merchants use the glitter pattern on the card to verify its authenticity by scanning the pattern with a device and checking if it is digitally signed. Finally, the signature contains a public key associated with the payer's phone, which enables the generation of a digital signature that guarantees that the signature has been signed on behalf of the payer.

The cardholder receives five digitally signed data elements when purchasing the physical card or when a user decides to increase the card's balance.

a) The public key of the card owner.

b) The glitter pattern.

c) The Merkle tree of the barcodes.

d) The card's net balance at the latest reload.

e) The total amount of the tile's face value at the time of the latest reload.

The card issuer supplies all these data elements to the card owner. Two further data elements are created during a transaction when:

f) A customer and merchant agree on the payment amount.

g) The merchant's device detects the value of all the already used tiles while scanning the card.

A merchant uses these data points during a transaction. By scanning both the barcode and the glitter pattern, the merchant not only prevents the possibility of card cloning because of data point b) but also detects the unused coated tiles with a face value on them. This allows the merchant to ensure that the card has sufficient balance by calculating whether the transaction amount f) is less than or equal to the most recent card balance upload d), minus all the scratched-off tiles at the time of payment g), and minus all the scratched-off tiles at the time of the latest reload e), resulting in the equation f $\leq$ d-g-e.

After verifying the card's authenticity and ensuring sufficient balance is available, the merchant can remove the coating from the tile, revealing the barcode. The merchant's device scans and hashes the barcode to verify its signature by the card issuer as part of the Merkle tree c). Finally, the merchant can remove the barcode from the card by applying a wet tissue onto it.

An alternative to physically wiping away the barcode is to use a non-interactive zero-knowledge proof of the barcode, the agreed-upon payment amount, and the merchant's private key. This ensures that the tile's value is reserved, even if the merchant has not yet processed the transaction online. This procedure helps both the merchant and the customer to prevent payment fraud. This procedure safeguards a customer from a merchant attempting to redeem a sum exceeding the agreed-upon amount, as the signature on the agreed-upon sum f) with the private key corresponding to the public key a) must be submitted during an online redemption. Similarly, the procedure protects a merchant from a customer trying to claim a smaller amount than the agreed-upon price, as the zero-knowledge proof authenticates the agreed sum f) by a merchant who knows the barcode.

In the next paragraph, we examine the card-only mode, focusing on the key differences in data points delivery because the rest is the same.

Card-only Mode Customers and merchants use the card-only mode when the payer's phone is inaccessible but the merchant's phone is operational and connected online. The merchant scans the barcode corresponding to the agreed-upon value and verifies the card's correctness by getting the data points online instead of the customer's phone, as with the offline-phone mode.

Analysis In the offline-phone or card-only mode, the merchant must wipe away the used barcode to prevent others from using it before the merchant can go online and redeem the funds. Alternatively, merchants can use a non-interactive zero-knowledge proof to "reserve" a tile for the merchant. Whipping away the barcode may be inconvenient and time consuming. Therefore, we suggest using non-interactive zero-knowledge proofs as the standard option.

Breakout-tile Mode Customers and merchants use the breakout-tile mode when a connection is unavailable for an extended period or when a functioning phone is inaccessible. The customer provides a tile with the correct value on it to the merchant, which contains the original unaltered scratch-off coating on it. Customers and merchants can use these tiles just like cash, as the coating on the tile guarantees their security. A user who received such a tile can redeem its value by scratching off the coating and scanning it while having an internet connection.

On the reverse side of the Figure 3.5, viewable and hidden security features, such as holograms and special ink, provide sufficient counterfeit resistance for the tiles and the denominations, even if the card owner removes the tiles from the card.

One use case of the breakout-tile mode is during widespread infrastructure failure. In such an instance, a central bank could simultaneously activate some or all tiles as spendable. These tiles could function as emergency loans, ration tickets, or other forms. Depending on the circumstances, the fixed value printed onto the tile could represent another meaning, like a ration ticket.

Analysis In the breakout-tile mode, tiles are detached from the card. To ensure that the tile is authentic, the merchant can check the security features on the backside of the tile. However, unlike physical cash, the cards do not have to be fully loaded by the cardholder. We would like to understand how the merchant can verify that sufficient balance exists on the card.

One potential approach is to allow cards to have negative balances. This would enable the enforcement of broken-out tile redemption, resulting in a negative balance on the card if insufficiently loaded. However, this would require the card issuer to give credit to the cardholder, which carries the associated risk of default and is costly.

Privacy features When tiles remain attached to the card, each tile is directly linked to the card. This implies that if a person is associated with a card, all payments can be traced back to that person. To maintain privacy, a card could be purchased anonymously. However, with extensive use, an owner's identity could be linked to the card. Detaching the tiles from the card enhances anonymity because the security features on the reverse side of the tiles are identical, and there is no unique serial number on them unless the user removes the coating during redemption, revealing the unique barcode hidden beneath it.

Analysis about obtaining a card A customer can buy the card from a card issuer. We do not know much about this relationship, but they most likely have to conduct some form of public policy goals, such as KYC checks. However, other public policy objectives like AML and CFT are very hard, if not impossible, to implement because a card issuer will not receive any form of data post-purchase of the card. This implies that the privacy definition of 3.1 is not fulfilled. However, we believe that the public policy objectives might differ for an offline CBDC because it resembles cash.

3.6 Summary between the eCash versions

We have seen how eCash works and how it guards customer privacy. Figure 3.6 summarizes the most important points. Throughout all versions, customer privacy is, in our opinion, strong. However, one open question is whether person-to-person (p2p) payments are allowed. In such a scenario, the receiving party does not have privacy. In the reports that we analyzed, p2p was never mentioned, so we do not analyze this scenario.

	Transaction privacy	Receiver privacy	Payer anonymity	Unspent list
SNB eCash	Yes	No	Yes	No
Tourbillon eCash2.0	Yes	No	Yes	Yes
Offline-phone mode	Yes	No	Yes	No
Card-only mode	Yes	No	Yes	No
Breakout-tile mode	Yes	Sometimes[5]	Yes	No

Figure 3.6. Summary of the eCash findings

Before introducing the Digital Euro, we want to explore how eCash can improve the four-party model discussed in Chapter 2.1.

3.7 eCash and the four-party model

We make two analysis, firstly cost and secondly privacy. We keep the privacy part shorter and refer the reader to Chapter 3.4.2 for the corresponding part of the work.

[5] The merchant has receiver privacy in the breakout-tile mode if they use the received tile from a customer in a payment itself, where the merchant hands the tile to another merchant. Only the merchant who uploads the tile to redeem the funds does not have receiver privacy.

Cost analysis Firstly, the payment owner is the central bank. This service would most likely be free, eliminating the payment scheme's fees. The acquirer and issuer, which would be payment service providers in eCash, might charge some fees, but this depends on the regulation. The ECB, for example, wants basic features of the Digital Euro to be free of charge. Therefore, acquirer and issuer charges are also potentially eliminated. But even if this was not the case, the merchant receives the CBDC from the customer at the time of payment. An acquirer does not need to prepay this transaction (which is costly to the consumer or the merchant, depending on who bears the costs because of the time-value of money). There is no need for an issuer to charge many costs to the customer for the credit card because of the limited service they need to provide. Competitive markets, therefore, will push the costs down to incentivize the customers to switch the PSP until the infrastructure costs of running those services create a lower barrier. The infrastructure costs associated with running those services are very high. It will be interesting to see how the ECB handles this problem to make the service free of cost.

Privacy analysis The payment issuer is the central bank, which cannot collect transaction data because of the unlinkability between a blinded and unblinded coin. The issuer and acquirer are both commercial banks. However, the issuer only knows the customer requested a CBDC conversion of amount x. The issuer knows nothing about a future transaction, including time, place, destination, and amount. On the other hand, the acquirer might still be able to collect data about customer behavior. We can make two assumptions: Firstly, the acquirer possesses a category code. With this information, the commercial bank can match incoming CBDC to a purchase and can thus collect customer behavior within a shop. In our second assumption, the acquirer does not have a category code. This strongly decreases the opportunity to map customer behavior. However, it is still doable, albeit really expensive, because everyone can create a handmade category code by visiting the store daily and writing down the new products and price changes.

We believe that eCash can bring more benefits to the payment ecosystem than privacy. In the next chapter, we will examine how the Digital Euro works, the ECB's goals, and how it plans to fulfill them.

Chapter 4

The Digital Euro

This chapter analyzes the Digital Euro as a potential addition to the current payment ecosystem made by the European Central Bank (ECB) [17]. The report presents the findings of the investigation phase and concludes that the ECB can design a solution that satisfies customer needs from both a product and distribution standpoint. The legislative framework has yet to be determined. Therefore, the ECB is currently unable to provide precise definitions of, for example, the level of privacy or to make a decision regarding a potential launch of a Digital Euro.

4.1 Motivation of the Digital Euro

In a blog update by the ECB, Cipollone, a member of the ECB's executive board, provided an insightful explanation of the reasoning behind wanting to introduce a Digital Euro [12]. One of the fundamental principles of the European Union is the freedom to work, study, and live in any EU member state. One part that makes this possible is the Euro because it can be used as a medium to exchange goods and services for a monetary value. The ECB is firmly committed to cash, so the goal of the Digital Euro is not to replace it. They would like to develop a digital alternative that offers the same benefits as cash and that exists digitally.

During the investigation phase [17], the ECB formed a focus group with the goal of evaluating the potential benefits of a Digital Euro. The focus group identified the increase in life quality because of the simplification of payments as a key point that could thrive a Digital Euro adoption. It would provide Europeans with an additional form of payment across the European area. Customers can use the Digital Euro for various forms of payment like online shopping and person-to-person (P2P) payments.

The ECB has identified three potential positive effects of the introduction of the Digital Euro. Firstly, it would establish a new Eurozone-wide payment solution under European control, complementing the existing European private payment solutions. Secondly, the Digital Euro would depend solely on its underlying infrastructure. Finally, a pan-European platform will be made available to European Payment Service Providers (PSPs), who can build their own services on this platform. We will dive deeper into the topic of PSPs later in this work. These effects will reinforce the European payment system and reduce dependencies on the retail payment layer of other nations. This increases the autonomy and resilience against technological disruptions and cyber attacks. Furthermore, payment costs will decrease while encouraging innovation and reducing dependencies on private external providers.

There are some restrictions on the Digital Euro. The holding of the Digital Euro will have a maximum limit for two reasons. The first is to comply with regulations such as Anti-Money Laundering (AML), as the goal of the Digital Euro is to provide similar levels of privacy as cash and, therefore, is susceptible to being used for illicit activities. The second reason is to avoid bank runs. Nevertheless, a customer can buy goods and services with a higher price than the Digital Euro limit because the Digital Euro account is linkable to a commercial bank account. These precautions result in a Digital Euro having little or no material impact on financial stability or the transmission of monetary policy. Business users, public authorities, and governments have a zero holding limit but are allowed to make certain types of payments over the Digital Euro. This is possible because any Digital Euro those parties receive will be immediately sent to their commercial bank account and converted to Euro. Additionally, a linked bank account can fund any Digital Euro payment from their side at the time of payment.

We elaborated on the motivation for a Digital Euro and now take a closer look at the ECB's goals regarding its core features.

4.2 Background

The first phase of launching a Digital Euro is the investigation phase [17]. During this phase, the Eurosystem, which is the central banking system of the eurozone and comprises the ECB and all the central banks of the member states that use the Euro, tested the feasibility of a digital payment system that could supplement physical cash. Before diving into the report, it is important to understand the goals of the ECB regarding a Digital Euro so that we can later verify whether these goals are achievable using an eCash based CBDC in Chapter 5.

4.2.1 Goals of the ECB regarding a Digital Euro

The ECB envisions some goals for the Digital Euro:

1. A digital currency that is widely accepted and easy to use.
2. Suitable for any payment in the eurozone.
3. Usable offline and online.
4. It should offer the highest possible protection of privacy.
5. Settling payments instantly.
6. The use of basic features would be free of charge.

If a Digital Euro is implemented that meets these objectives, users will have another payment option that respects their privacy and settles payments instantly with minimal or no transaction costs.

The ECB's goal of providing the highest possible level of privacy protection is vague. In the following subsection, we examine this goal in more detail to better understand the possibilities and limitations of privacy that the report sets out.

4.2.2 Privacy of the Digital Euro

The aim is to provide the highest possible levels of privacy. However, it is up to the legislators to decide what that entails. Cipollone explains in his blog post that the Digital Euro would offer improved privacy features compared to existing commercial solutions [12]. Online payments will use the latest privacy-enhancing technologies. In offline payments introduced in the next section, only the parties involved will have access to the transaction details. The data generated from utilizing the Digital Euro will be pseudonymized and stored within the European jurisdiction, leading to high privacy standards. In addition, independent data protection authorities will monitor the ECB's data privacy standards to ensure that the data protection rules are respected.

The ECB has no intention of monitoring the user's payments and has no desire to engage in any commercial activities. In addition, the ECB will not store or have access to any personal data that would directly identify a user. All actors, including the Eurosystem, are subject to the General Data Protection Regulation (GDPR) and EUDPR [17].

Daman, a data protection officer in the ECB's Legal Service department, published a blog post describing the privacy of the Digital Euro [13]. In this blog post, Daman noted, "Will it be as private as cash? Not quite, but close. The Digital Euro promises you better privacy and data protection than other current electronic means of payment".

The ECB wants to give end-users the power to protect their privacy through opt-in privacy features, where a user has to consent if a payment service provider wants to collect additional customer data [17]. It is strictly forbidden for a PSP to restrict access to the Digital Euro in any manner if an end-user decides not to opt in. Pseudonymization and a clear separation of the user's personal data between the PSP, the Eurozone, and any other potential third-party provider segregate any personal information between the user and the Eurosystem. It only receives the data necessary to carry out its tasks in accordance with the regulatory framework, and even this data is pseudonymized.

Analysis While the Eurosystem aims to provide a level of privacy similar to that of physical cash, this is not decided by the Eurosystem. The legislators decide what level of privacy is possible without interfering with other policy objectives. The ECB is in a difficult situation because it has to argue for the privacy aspect of the Digital Euro without having a clear framework.

We know that the ECB intends to use a pseudonymized list of all the user's individual holdings. We believe this is a significant privacy concern, especially if it is a two-way pseudonymized function. If the list were ever to get stolen and published, it would most likely be possible for a small group of people to figure out the two-way pseudonymization function by comparing their Digital Euro Account Number (DEAN) to the pseudonymized account number. The DEAN is a unique identifier for a person and can be used, among other things, to receive payments. Each Digital Euro user has one DEAN, which can act as proof of ownership as well as a PSP identifier if the central alias lookup system fails.

.A one-way pseudonymization function would offer more privacy, but we do not understand the advantages of monitoring individual Digital Euro holdings. We believe that monitoring the coins instead of the user's holdings is more advantageous. A maximum holding limit is still enforceable by setting a limit on the wallet.

We briefly mentioned the Digital Euro's online and offline capabilities. In the next section, we examine both versions more closely.

4.2.3 Online and offline Digital Euro

The Digital Euro payment system consists of two modes: an online mode and an offline mode. The online mode covers the majority of use cases. As the name suggests, an online connection is needed to transmit payments from a customer to a merchant. The goal of the online mode is to reduce the reliance on bearer instruments such as cash. The problem with such forms of payment is that they give ownership to the bearer, but there is no record of that ownership. If the funds were to get lost or stolen, there is no way for the user to recover the funds because there is no way to prove ownership in the first place. A PSP is needed to validate the payments and ensure that they comply with the payment regulations. From a privacy point of view, the PSP only receives the data transaction points necessary for the basic functionalities and regulation.

One question that immediately comes to mind when looking at CBDC is where the CBDC actually comes from, or more precisely, how a user funds their Digital Euro holdings. The most convenient way is to establish a permanent commercial bank connection. A transfer exceeding the Digital Euro limit could still be received and immediately converted via the commercial bank to Euros. Prefunding is not necessary under these circumstances, as the linked bank account can be the source of funds. This connection to the commercial bank is called waterfall/ reverse waterfall. This system combines funding, defunding, and payment processing into a single action while keeping the processing time relatively constant. A key point is that funding and defunding can be done on a 24/7 basis, 365 days a year via digital payments. As a result, a user can facilitate any transaction below the holding limit at any time.

However, this is not the only way to obtain or get rid of Digital Euros. Cash is another way because not everyone wants to link a bank account to their Digital Euro account.

The Offline mode is not dependent on an internet connection but requires prefunding and is limited to the need for physical proximity in order to make payments. The user must manually fund and defund the wallet while staying within the holding limit. Digital Euro coins must be stored safely on a secure device because if the apparatus is compromised or lost, retrieving them will not be possible. The goal of the offline version is to provide a similar level of privacy to that of physical cash. There is no need for a PSP to act as a validator, and since the customers make the payment locally, there is no validation or record of the transaction. Therefore, there is no need to share transaction data with a PSP, the Eurosystem, or other providers.

The offline mode is a peer-to-peer validation model, and prefunding and defunding of offline holdings are only possible with a functioning internet connection. Additionally, the device storing the offline holdings would still need to go online periodically to verify the security features. It is not clear how the offline solution will look like and which parts will be provided by the Eurosystem. However, the ECB envisions that the mode settles transactions instantly and that the Eurosystem cannot see the user's information or payment patterns. It must be possible to spend Digital Euros received via an offline transaction without going online again. Figure 4.1 summarizes the differences between the online and offline modes.

	Online	Offline
PSP involved	Yes	No
Prefunding needed	No	Yes
Close proximity needed	No	Yes
Waterfall features available	Yes	No
Connection needed	Continuously	Periodically
ECB can identify the users	No	No

Figure 4.1. Differences between Online and Offline version

Analysis While connecting the commercial bank account makes sense from a practical perspective, it harms the customer's privacy if the PSP is not the commercial bank of the customer.

In this section, we have seen the differences between the online and the offline modes. In the next section we discuss the role of PSPs in greater detail, as they are a substantial part of the Digital Euro.

4.2.4 Payment service provider

A payment service provider enables a connection between a user and the underlying infrastructure of the Digital Euro. PSPs can be commercial banks, but they can also be financial technology companies (fintechs) and payment processors that want to build their own services on top of the infrastructure. The goal is to improve the collaboration between banks and other PSPs, thereby increasing competition, promoting innovation, and lowering costs.

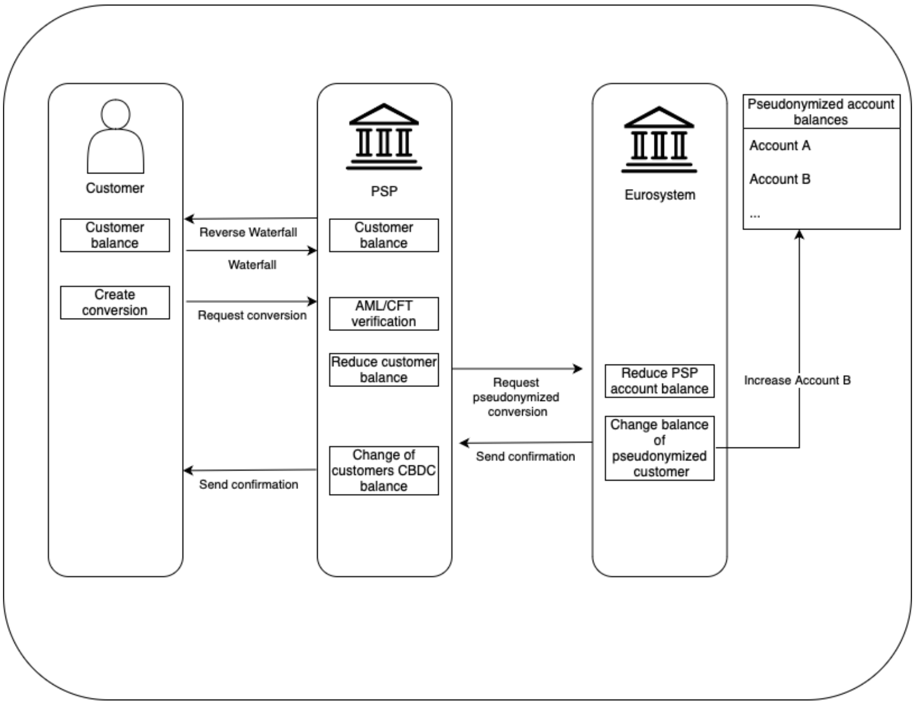


Figure 4.2. Exchanging traditional Euro for Digital Euro

The PSP is responsible for providing onboarding services to users. This process must be as simple as possible to achieve the goal of ease of use and accessibility to anyone. As part of the onboarding process, the PSP is responsible for carrying out know-your-customer (KYC) checks according to the regulations. The two main parts of a KYC check are i) collecting the relevant data on the customer's identity and ii) ensuring that the identity provided is correct. Distributing licenses to PSPs is the responsibility of the Payment Service Directive. While the role of the PSP is to open and manage Digital Euro accounts, the Digital Euros are not the PSP's liability but the Eurosystem's. The PSP is responsible for creating and distributing payment instruments, such as a mobile app or a credit card. The PSP also initiates and validates payments and follows the regulation of anti-money laundering (AML) and combating financing of terrorism (CFT) procedures.

A PSP could not perform its tasks without a functioning backend infrastructure. The Eurosystem creates this structure, which we examine in the next section.

4.2.5 Eurosystem

The role of the Eurosystem is to provide the backend infrastructure for the PSP. The PSP needs to provide secure and synchronized services to users, and the Eurosystem would support these functions by providing the necessary features. In this section, we take a closer look at what the Eurosystem does and what kind of customer information it has access to.

One feature is the settlement of online Digital Euro transactions. As the Digital Euro is a liability of the Eurosystem, it also carries the risk of undue central bank money creation when an error happens. As a solution, the Eurosystem keeps a record of all the Digital Euro liabilities and maintains the ledger that determines the ownership of all the money it issues. This has advantages for users because a PSP cannot falsify the user's holdings since the Eurosystem records them. Privacy is protected because the system only gives the data that is absolutely necessary to the Eurosystem in a pseudonymized form.

The Eurosystem maintains a pseudonymized register of all Digital Euro accounts. The stored data is limited to contain the smallest amount of information needed to ensure that users follow the rules, such as respecting the holding limit. The Eurosystem will also launch a mobile app for the Digital Euro, which will make it easy for a user to interact with a PSP, but also to provide the basic features that the ECB envisions. The idea is to lower the barriers to entry in order to increase competition. Using this app by the Eurosystem is only possible if a user has gone through an onboarding process by a PSP. Figure 4.2 shows how various actors could collaborate to exchange traditional Euros for Digital Euros. In this example, the customer enabled the waterfall and reverse waterfall features, so they do not have to manage the funding manually. The PSP conducts all the necessary regulatory steps and deducts the customer's account balance. Subsequently, a pseudonymized conversion request is submitted to the Eurosystem, resulting in a reduction of pseudonymized Account A and a corresponding reduction in the PSP's account balance. The Eurosystem then sends a confirmation to the PSP, which updates the Digital Euro app holdings and notifies the customer of successful conversion.

We have seen all the relevant parties involved in a Digital Euro transaction. We have mentioned, but not explained, the basic functionalities that the ECB envisions for the Digital Euro. These basic features should be available to every user of the Digital Euro, are free of cost and are the merit of the whole system.

4.3 Basic features

The ECB has stated that one of its goals is to provide basic features free of charge. These features can be divided into three categories. The first is user management. As the name suggests, this category contains everything related to the user, like onboarding and offboarding. The second is liquidity management, which verifies that a user has enough Digital Euros but not too many to exceed the limit. The last category is transaction management, where the system carries out all the administrative and processing tasks during a transaction, such as payment initiation and authentication.

4.3.1 User management

To interact with Digital Euros, users must first establish an account through a PSP. To promote inclusion, users can complete this process remotely or in person. The rights of merchants differ from those of ordinary users. Therefore, the onboarding process for individuals is different than for merchants.

Individuals receive credentials, including a DEAN. A customer can create an alias, which is data that uniquely identifies a user, such as a telephone number. This alias is linked to the DEAN and can be used to receive payments. Proof of ownership is only required when the PSP's IT system fails. The PSP will provide a customer with a platform to access and interact with their funds and a physical card that can be requested. While it is possible for a customer to change the PSP, having several Digital Euro accounts is impossible. Therefore, if a user wishes to change the PSP, they must transfer their holdings via a portability request. The DEAN remains the same when changing providers. Users who wish to terminate their interaction with the Digital Euro can invoke an offboarding process. During this process, the user must first transfer the funds to a bank account and then deactivate it.

Merchants can only use a limited version of the Digital Euro. They can have one or several DEANs and a special user application for businesses to receive and send Digital Euros but not to hold them because of the strict no-holding rule. To accept and spend Digital Euros, they must connect their bank account to their Digital Euro account and allow the waterfall/reverse waterfall features. The following chapter explains these features.

4.3.2 Liquidity management

Users can track their liquidity and not rely on any automated service that converts fiat into Digital Euro or vice versa. However, users need to know before making any payment whether they have sufficient

Digital Euro in their wallet. This is generally an inconvenience that could hinder adoption. Therefore, the ECB suggests that the PSP should provide some basic functionalities to automate this process and make these services available 24/7. Cash withdrawals must always be possible, for example, via an ATM.

An important feature of liquidity management is the waterfall and reverse waterfall concept, which removes the need for a customer to keep track of their Digital Euro liquidity with respect to being able to receive/spend them. With the reverse waterfall feature activated, users connect their commercial bank accounts to their Digital Euro account. If the payment exceeds the current Digital Euro holdings, the commercial bank will automatically top up this balance by converting fiat into Digital Euros. The counterpart of this feature is called waterfall, where Digital Euros get converted to fiat and sent to the commercial bank as soon as the user reaches the limit. This allows a user to receive payments that would otherwise exceed the limit without the transaction failing. These features are also important for businesses such that they can send and receive Digital Euros.

Analysis Customers should be aware that waterfall/reverse waterfall features harms their privacy because a commercial bank can monitor their Digital Euro inputs and outputs.

4.3.3 Transaction management

To initiate a payment, users log into their Digital Euro app or use a payment card provided by the PSP. The PSP initiates the payment, conducting necessary checks for verification, AML/KYC compliance, and fraud detection. Once approved, the PSP routes the payment for settlement. During this process, the PSP sends settlement instructions to the settlement infrastructure.

Refunds are an important aspect of any payment system. Many industries require this concept. Zalando, a clothing retailer company, has a business model where customers can order goods, and if they do not meet their expectations, they can return them and get a refund. In the Digital Euro, refunds from merchants are possible, regardless of the size or category of the merchant, using the reverse waterfall feature to send Digital Euro. In order to prevent money laundering, the refund must have the same form as the payment. If a customer uses Digital Euros, the refund will also be in Digital Euros. The refund is independent of the device and infrastructure used during the payment. A customer receives the refund without linking the commercial bank account. However, if the refund exceeds the Digital Euro holding limit and a conversion from Digital Euro to another form is not possible at that time, the refund will fail. An alternative payment instrument will have to be used.

Analysis There is a contradiction regarding the market practices of sending a refund in the same form back. The report argues that: “A Digital Euro payment could only be refunded via Digital Euro, in line with market practice, to prevent, for example, money laundering. ” (ECB 25), while also arguing from a customer perspective that: “If the waterfall functionality is not activated and there is not enough room within the holding limit to receive the incoming refund, this refund process would fail and an alternative payment instrument would need to be used for the refund”(ECB 26). This implies that users could circumvent the refund type.

Figure 4.3 illustrates a potential Digital Euro payment scenario where the waterfall and reverse waterfall features are disabled. After the customer and merchant agree on a deal, the customer funds their Digital Euro account using the app provided by the PSP. The customer sends a payment request, including the merchant’s DEAN, to their PSP to initiate a payment. The PSP reduces the customer’s Digital Euro holdings, conducts AML/CFT checks, and forwards a pseudonymized spending request to the Eurosystem. The Eurosystem debits Account B and transfers the Digital Euros to the designated DEAN. However, since the merchants have a strict no-holding policy, the central bank automatically destroys the Digital Euro and credits the PSP’s bank account. The PSP, in turn, credits the merchant’s bank account and sends a deposit confirmation to them, who then hands over the purchased goods to the customer.

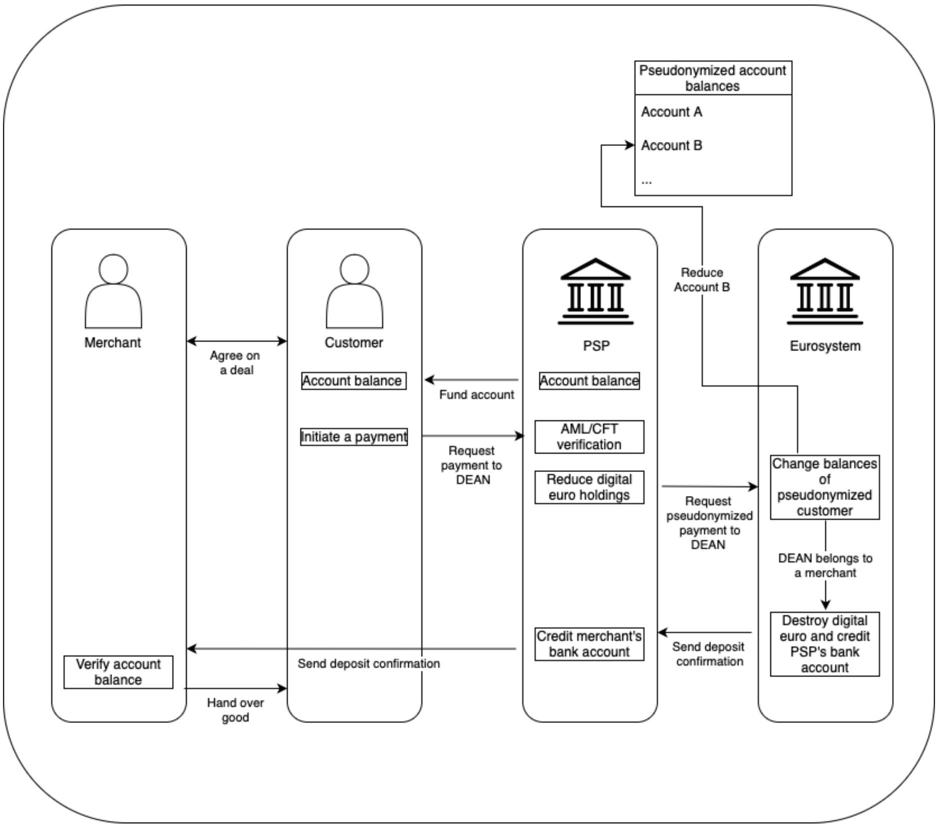

Pseudonymized account balances
Account A
Account B
...
Reduce Account B
Merchant
Customer
PSP
Eurosystem
Agree on a deal
Account balance
Fund account
Account balance
Initiate a payment
Request payment to DEAN
AML/CFT verification
Reduce digital euro holdings
Request pseudonymized payment to DEAN
Change balances of pseudonymized customer
DEAN belongs to a merchant
Destroy digital euro and credit PSP's bank account
Credit merchant's bank account
Send deposit confirmation
Send deposit confirmation
Verify account balance
Hand over good

Figure 4.3. Spending Digital Euro

4.4 Discussion about the Digital Euro

The Digital Euro can enable significantly more privacy than existing solutions. Sillaber and Eggen [23] analyzed existing real-time payment systems like TARGET Instant Payment Settlement (TIPS) and the Digital Euro. In this analysis, they identified privacy concerns regarding the user's relationship with the PSP and the TIPS system because of broad data processing agreements, unilateral contractual terms, and consent as the only basis for data processing. This allows the participants of the system to process and store more data than is needed for the task. While user consent is still the basis for data processing in the Digital Euro, the consent is limited by the Digital Euro's legislation[1]. Therefore, contrary to existing solutions, stricter legal boundaries will limit the processing and storage of user data, thereby improving privacy.

While the online and offline Digital Euro versions offer more privacy than traditional online payment systems, the online version has less privacy than the offline version. This is because the ECB wants the online Digital Euro to combat AML and CFT effectively, whereas the privacy of the user plays a more important role in the offline Digital Euro [22].

We believe that by making some systematic changes to the proposal, the system's privacy can be improved while also following public policy goals like AML and CFT. In the next chapter, we evaluate our recommendations and explain why we believe these changes benefit the Digital Euro. We also evaluate whether eCash and the Digital Euro are compatible and if it is possible to implement the Digital Euro with eCash.

[1] While the legislative framework does not exist yet, there is a proposal by the ECB.

Chapter 5

Digital Euro implementation through eCash

In this chapter, we want to answer the research question of whether or not it is feasible to use eCash to implement the Digital Euro. We will put the most emphasis on privacy, as this is the feature of a CBDC that citizens are most concerned about. During this process, we make systematic change suggestions to the Digital Euro proposal and explain why we believe these changes benefit the customer.

5.1 Compatibility of eCash and the Digital Euro

We assess the four actors: the central bank, the PSP, the customer, and the merchant. We analyze each of the entities where it makes sense.

5.1.1 Account versus Token-based CBDC

The ECB has not yet decided on the specific type of CBDC to implement. Therefore, we want to analyze whether a token-based CBDC is viable for the Digital Euro.

Digital Euro We know that the ECB intends to use a pseudonymized list of all Digital Euro users. This could indicate that the ECB envisions an account-based CBDC because such a registry is necessary, while such a list is not beneficial for a token-based CBDC. Eggen and Sillaber also conclude that the online version is an account-based CBDC [22]. However, the Digital Euro prototype developed during the prototyping exercise, which spanned from July 2022 to February 2023, used an unspent transaction output (UTXO) data model combined with a settlement engine [16]. In such a model, the UTXO holdings are recorded as discrete units with fixed values, with each unit only involved in two transactions. Firstly, the creation of the coin where a user transforms fiat currency into a coin and, secondly, the spending of the coin in exchange for a good or service. Interestingly, one of the key results is that: "the prototype showed that the Eurosystem would be able to perform the settlement tasks without being able to know the balance or to infer the payment patterns of any user." The ECB noted that the prototypes were developed only for learning experiences and as a research tool. However, not strictly using an account-based system for the prototyping shows that the ECB is still experimenting and has not settled on a specific CBDC design so far. For the offline Digital Euro, a token-based CBDC has to be used because no settlement engine is involved in those transactions.

eCash The eCash proposal is a token-based system where a customer creates tokens by first computing the blinded coin and then sending it to the central bank. Subsequently, the central bank authenticates the coin by applying its private key to the blinded coin, thereby generating the signed and blinded coin. The central bank returns the signed and blinded coin to the customer, where the customer unblinds it, making the coin spendable.

Analysis One of the objectives of the Digital Euro is to enable the highest level of privacy possible, which, in our opinion, entails transaction privacy. Chaum, Grothoff, and Moser argued: “In an account-based system, the assets (accounts) are associated with transaction histories that include all of the credit and debit operations involving the accounts. In a token-based system, the assets (tokens) carry information about their value and the entity that issued the token. The only possibility of attaining the transaction privacy property of cash, therefore, lies in token-based systems.” [10] While we do not fully agree that token-based systems are the only possibility of attaining transaction privacy, we believe it is the easiest way to achieve this property. Therefore, we propose that we focus on the token-based type for the Digital Euro for this thesis. However, we believe that a combination of account and token-based CBDC would also work and that account-based CBDC can offer the same levels of privacy as a token-based solution.

We do not believe an online token-based system hinders public policy objectives because parties can conduct them during CBDC creation and redemption. Commercial banks, for example, achieve these objectives in eCash. In this thesis, we propose a token-based system for the Digital Euro. As a next step, we compare the role of the Eurosystem with that of the central bank in an eCash implementation.

5.1.2 The central bank

Citizens are concerned about giving too much information to the central bank. Therefore, it makes sense to create a CBDC solution in which the central bank knows as little as possible about the users while still being able to do its job properly according to the legal framework.

Digital Euro The role of the Eurosystem is relatively limited because the PSP covers most services. If we look at a transaction, the only task that the Eurosystem has to perform is the settlement of the transaction. The PSP initiates, validates, and implements the payment post-settlement. After the validation by the PSP, it sends settlement instructions to the settlement infrastructure provided by the Eurosystem, which performs the transaction according to this note.

eCash The central bank has two tasks. Firstly, it authenticates the blinded coin by signing and returning the signed and blinded coin, and secondly, it maintains an unspent CBDC list.

Analysis We do not know much about the settlement infrastructure. However, we assume that the goal of the Eurosystem is to provide the core transaction process so that the PSPs can maximize the customization of their service. The Eurosystem could, for example, provide an API where the PSP sends the pseudonymized Digital Euro addresses and the requested amount to the Eurosystem. The Eurosystem changes the balances and returns a confirmation.

We believe that maintaining a list of the individual holdings of each user can be a privacy vulnerability even if the accounts are pseudonymized. Suppose a malicious actor breaches the pseudonymity, for example, by leaking the list and working together collectively to develop the pseudonymization function. In that case, the whole transaction history of all its users can become visible. In our opinion, such a list hinders the adoption and violates the goal of being as private as possible because of the risks of de-pseudonymization. Therefore, we agree with the concept of eCash and believe that maintaining a list of unspent CBDC makes more sense from a privacy perspective than keeping a list of each user’s Digital Euro holdings. However, if we propose a new approach using eCash, it must be consistent with the goals of the Digital Euro that we mentioned in Chapter 4.2.1. An unspent list will affect the fourth goal, which is to be as private as possible, and the fifth goal of settling payments instantly. For the reasons explained above, we believe that an unspent list will enhance privacy, so goal four is fulfilled. The fifth goal is more difficult to fulfill because the BIS has identified a scaling problem when using quantum-safe encryption. Depending on the size of the simultaneous transactions, payments may not be settled immediately, which conflicts with the objective of settling transactions immediately. Nevertheless, we think a trade-off between better privacy and time taken per payment is worthwhile, as the BIS survey identified privacy as the biggest concern of citizens [18].

We recognized the differences between the roles of central banks in the Digital Euro and eCash. For privacy reasons, we propose to maintain an unspent list rather than a global pseudonymized list for each user. As a next step, we examine the role of payment service providers interacting with the central bank in creating and settling CBDC transactions.

5.1.3 Third party providers

Both systems rely on third-party providers to act as the link between the customer and the central bank.

Digital Euro The role of a PSP is similar to that of a regular commercial bank. It provides onboarding for new users, initiates transactions, and completes the necessary regulatory measures. To complete a transaction, it sends a transaction report to the Eurosystem. Importantly, a Payment Service Directive grants and removes PSP licenses to companies that want to take on the role of a PSP, so not exclusively banks but also fintechs and payment processor companies can be PSPs.

eCash With eCash, the commercial banks are the third-party providers, acting as intermediaries between the consumer and the central bank. A customer initiates a payment at their commercial bank, which performs all the necessary regulatory measures. It then forwards the hashed and blinded coin to the central bank and the hashed coin to a mix network. After the central bank has completed its job and signed the transaction, the commercial bank simply returns the signed and blinded coin to the customer. When a merchant receives a payment, they forward it to the commercial bank, which itself forwards the coin to the central bank. The commercial bank credits the merchant once the central bank returns a confirmation of the transaction.

Analysis We believe there is no technological barrier for fintechs and payment processor companies to act as commercial banks in eCash. If we look at the goals of the Digital Euro, there is no conflicting objective with the use of PSPs based on an eCash system other than the immediate settlement of payments. If we compare the scalability problem of quantum-safe eCash identified by the BIS in the Tourbillon paper [3] with the ECB's prototype summary [16], we find that both prototypes have scalability problems, and both papers conclude that further research and development needs to be done in this direction. Therefore, we leave out the transaction speed objective when comparing PSP compatibility between the Digital Euro and eCash. We conclude that it is possible to base PSPs on eCash technology.

5.1.4 Consumer and merchant

From a customer's perspective, the two solutions are essentially identical. In order to obtain CBDC, both solutions involve a PSP, and in both cases, the customer has a holding limit. Using eCash, a user initiates the conversion of money into CBDC by first logging into their bank account. Once the bank has authenticated the login, the customer can hash and blind the coins and transmit the blinded coins and the hashed and blinded coins to the commercial bank. If the central bank completes the conversion, the customer receives the signed hashed and blinded coin back and can unblind it to make it spendable. If we compare this process with the ECB's objectives for a Digital Euro, we conclude that there is no conflict of interest and that an eCash-based model can be used from a customer's perspective.

From a merchant's point of view, they have a strict no-holding rule. However, if we look at eCash, this restriction is not a problem. Let us assume Alice buys a good from Bob. Therefore, Alice scans a QR code with her mobile phone and signs the contract. To make the payment, Alice sends her CBDC directly to Bob's commercial bank, which interacts with the central bank. If the payment is accepted, Bob's bank account is credited.

The merchant never holds any CBDCs because they are sent directly to the central bank via the commercial bank and converted into currency. Thus, the goals of the ECB regarding a Digital Euro are not violated, and we conclude that an eCash-based model is also possible for the Digital Euro from a merchant's perspective. We do not need a comparative analysis in this section because of the compatibility between eCash and the Digital Euro.

One of the ECB's objectives regarding a Digital Euro is an offline mode. The next section compares the compatibility between the offline eCash mode and the offline Digital Euro mode.

5.1.5 Offline mode

We analyze the compatibility between the offline eCash version discussed in Chapter 3.5 and the offline Digital Euro.

Offline Digital Euro The offline Digital Euro requires prefunding and relies on close proximity. There is no need for a PSP to act as a validator, and no transaction data is stored. A device is still required to go online from time to time in order to complete security feature verification.

Offline eCash Chaum suggests using a physical card containing scratch-off tiles with a numerical value on them [9]. Merchants can remove this coating to scan the underlying QR code containing the funds. If a phone is unavailable, the customer can break out the tiles and hand them to the merchant, which acts in the same way as physical cash.

Analysis It is difficult to conclude whether offline eCash is compatible with the offline mode of the Digital Euro. If we compare the ECB's objectives for the Digital Euro, we see that the offline version of eCash offers really strong privacy, and we think this may be too strong for the ECB. For example, the breakout-tile mode provides nearly the same level of privacy as physical cash. However, according to the Digital Euro report, the goal of the ECB is to offer a similar level of privacy to cash, but still less than that of physical cash. Another issue is the holding limit. When using the breakout-tile mode, no maximum holding limit is enforceable. This is a clear violation of the ECB's limited holding rule.

Offline eCash challenges the ECB's goals for an offline Digital Euro. We believe that the offline-phone and the card-only mode are compatible with the ECB's objective, especially since these modes can improve inclusivity. In our opinion, the breakout-tile mode is incompatible because it offers too high privacy standards and a holding limit cannot be effectively enforced. Additionally, the ECB does not intend to replace the physical Euro. The breakout-tile mode is, however, close to physical cash in many ways. We do not see the benefit of having a breakout-tile mode available when physical cash is still in circulation. Therefore, we propose an offline eCash version that does not include the breakout-tile mode.

5.2 Architectural proposals

We want to propose two suggestions regarding the implementation of the Digital Euro. First, we propose making the Digital Euro a token-based CBDC. Second, we propose removing the list, which contains pseudonymized holding information about each user. The following two paragraphs examine how we can use these proposals to create an eCash-based Digital Euro.

Exchanging traditional Euro for Digital Euro To make this exchange happen, we can examine the moving parts in Figure 4.2 and replace each entry with an eCash equivalent term.

1. The customer creates a conversion on their app and sends that request to the PSP. This can be facilitated in eCash by first taking the public key pk_x of the central bank for the intended exchange value x. The customer also generates a key pair $(pk_c, sk_c) \leftarrow \mathsf{Gen}(1^\lambda)$ of the coin and creates two values: the hashed public key of the coin, encrypted with the public key of the central bank $f \leftarrow \mathsf{Enc}(\mathsf{H}(pk_c), pk_x)$ and the hashed and blinded public key of the coin $f' \leftarrow \mathsf{Blind}(\mathsf{H}(pk_c), b)$ with the blinding factor b. The customer sends the encrypted, hashed coin f and the hashed and blinded version f' to the PSP.

2. The PSP then conducts all necessary verification, such as AML and CFT, and reduces the customer's account balance. Using our proposal, we change the next step from requesting a pseudonymized conversion to sending the encrypted, hashed coin over a mix network to the Eurosystem and the hashed and blinded coin directly to it.

3. The Eurosystem decrypts the hash using the corresponding private key sk_x, reduces the account balance of the PSP, and adds the hashed coin f to the proposed unspent list. The hashed and blinded coin is signed $s' \leftarrow \mathsf{Sign}(f', sk_x)$ with the correct private key sk_x. Instead of sending a confirmation to the PSP in the next step, the Eurosystem sends the signed coin (s', f') back to the PSP.

4. The PSP forwards the coin to the customer's app and informs them of successful conversion.

5. Lastly, the customer unblinds $(s, f) \leftarrow \mathsf{Unblind}(s', b)$ and stores the newly minted Digital Euro.

Using Digital Euro in a transaction We again identify the moving parts in the spending Figure 4.3 and replace each entry with an eCash equivalent term.

1. Once the merchant agrees to a deal with a customer, they generate a QR code and hand it over to the customer. The customer scans this code with an application provided by the PSP, signs the transaction, and sends the signed coin (s, f) to the merchant.

2. To verify the payment, the merchant forwards it to the PSP, which conducts all the necessary checks like AML and CFT and forwards the coin to the Eurosystem.

3. The Eurosystem verifies that the signature is valid $v \leftarrow \mathsf{Vrfy}(s, pk_x)$ and searches the unspent list for a matching hash f and removes it. Lastly, it credits the PSP's central bank account and informs them.

4. The PSP credits the merchant's bank account over the reverse waterfall feature and notifies them.

5. The merchant verifies their bank account holdings and hands over the goods to the customer.

With these architectural proposals, we believe that implementing the Digital Euro using eCash is possible. In the next chapter, we draw a definite conclusion and define future work.

Chapter 6

Conclusion

Discussion In this thesis, we took a closer look at eCash. To study eCash's privacy standards, we examined the information a merchant, a commercial bank, and a central bank can gather about a customer during a transaction. A merchant receives a payment without learning anything about the customer because the only information received is a coin with the central bank's signature on it. The commercial bank can track when and how much a customer converts into CBDC but cannot link transactions to a specific customer. Tracking when and how much is necessary to perform public policy objectives and fund the transaction. Due to the blinded coins, a central bank knows nothing about the customer's identity. The only relevant information that a central bank receives is the commercial bank origin of the coins because it has to deduct their central bank account holdings.

Project Tourbillon concludes that payer anonymity is achievable, and the SNB's objective of transaction privacy is feasible. The privacy definition of section 3.1, which details the level of privacy a CBDC can have according to many central banks, let's conclude that eCash does not interfere with public policy goals. This is because the commercial bank implements and verifies transactions when a conversion or redemption of CBDC happens. Based on these findings, we believe that eCash provides strong privacy features for CBDC customers and is suitable from a privacy standpoint while not compromising public policy goals. However, we believe that in the offline eCash version of Chapter 3.5, the Breakout-tile mode does not fit the objectives of the Digital Euro, and therefore, we propose to remove the Breakout-tile mode.

The second part of this thesis explored the potential of eCash as a foundation for the Digital Euro. We believe that eCash is a promising technology for the Digital Euro because of its privacy features, which should reduce public concern regarding the privacy of CBDC. Nonetheless, we believe that the privacy standards are not strong enough intervene with the public policy objectives, because PSP's can conduct these policies in eCash. In Chapter 5, we proposed to make some adjustments to the Digital Euro to improve its compatibility with eCash. We justified those suggestions in order to increase the privacy and security standards.

Future work Quantum resistance for eCash is challenging because of the potential performance loss, which could compromise scalability issues. The BIS concluded that further research and development is necessary to address this issue.

Regulators need to make a clear legal framework before we can make a final decision on the compatibility of eCash and the Digital Euro. The privacy features of eCash might conflict with certain public policy goals of the EU. Additionally, the quantum-safe implementation of eCash has shown to have performance issues, which conflicts with the ECB's goal of settling payments instantly.

Bibliography

[1] S. Allen, S. Čapkun, I. Eyal, G. Fanti, B. Ford, J. Grimmelmann, A. Juels, K. Kostiainen, S. Meiklejohn, A. Miller, E. S. Prasad, K. Wüst, and F. Zhang, "Design choices for central bank digital currency: Policy and technical considerations." NBER Working Paper 27634, National Bureau of Economic Research , August 2020.

[2] Z. Amsden, R. Arora, S. Bano, M. Baudet, S. Blackshear, A. Bothra, G. Cabrera, C. Catalini, K. Chalkias, E. Cheng, A. Ching, A. Chursin, G. Danezis, G. D. Giacomo, D. L. Dill, H. Ding, N. Doudchenko, V. Gao, Z. Gao, F. Garillot, M. Gorven, P. Hayes, J. M. Hou, Y. Hu, K. Hurley, K. Lewi, C. Li, Z. Li, D. Malkhi, S. Margulis, B. Maurer, P. Mohassel, L. de Naurois, V. Nikolaenko, T. Nowacki, O. Orlov, D. Perelman, A. Pott, B. Proctor, S. Qadeer, Rain, D. Russi, B. Schwab, S. Sezer, A. Sonnino, H. Venter, L. Wei, N. Wernerfelt, B. Williams, Q. Wu, X. Yan, T. Zakian, and R. Zhou, "The Libra Blockchain." [Online], Available: https://diem-developers-components.netlify.app/papers/the-diem-blockchain/2020-05-26.pdf, May 2020.

[3] Bank for International Settlements, "Exploring privacy, security and scalability for CBDCs." Final report, [Online], Available: https://www.bis.org/publ/othp80.pdf, November 2023.

[4] Bank of Canada, European Central Bank, Bank of Japan, Sveriges Riksbank, Swiss National Bank, Bank of England, Board of Governors of the Federal Reserve System and Bank for International Settlements, "Central bank digital currencies: foundational principles and core features." Bank for International Settlements, Report no 1, October 2020.

[5] D. Boneh and V. Shoup, "A Graduate Course in Applied Cryptography." Unpublished, January 2023.

[6] C. Cachin, "Cryptographic protocols." Lecture 3, Unpublished lecture notes, University Bern, 2023.

[7] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Commun. ACM*, vol. 24, no. 2, pp. 84–88, 1981.

[8] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology: Proceedings of CRYPTO '82, Santa Barbara, California, USA, August 23-25, 1982* (D. Chaum, R. L. Rivest, and A. T. Sherman, eds.), pp. 199–203, Plenum Press, New York, 1982.

[9] D. Chaum, "Offline eCash 2.0 robust in-person payments later onlineable." [Online], Available: https://chaum.com/wp-content/uploads/2022/11/Offline_eCash2.0_8-31-22.pdf, August 2022.

[10] D. Chaum, C. Grothoff, and T. Moser, "How to issue a central bank digital currency." Swiss National Bank Working Papers, January 2021.

[11] D. Chaum and T. Moser, "eCash 2.0: Inalienably private and quantum-resistant to counterfeiting." [Online], Available: https://chaum.com/wp-content/uploads/2022/11/eCash_2.0_9-7-22-.pdf, September 2022.

[12] P. Cipollone, "Maintaining the freedom to choose how we pay." European Central Bank Blog, Opinion piece,[Online], Available: https://www.ecb.europa.eu/press/blog/date/2024/html/ecb.blog240625~78525e0d5c.en.html, June 2024.

[13] M. G. Daman, "Making the digital euro truly private." European Central Bank Blog, Opinion piece,[Online], Available: https://www.ecb.europa.eu/press/blog/date/2024/html/ecb.blog240613~47c255bdd4.en.html, June 2024.

[14] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[15] European Central Bank, "Eurosystem report on the public consultation on a digital euro." [Online], Available: https://www.ecb.europa.eu/pub/pdf/other/Eurosystem_report_on_the_public_consultation_on_a_digital_euro~539fa8cd8d.en.pdf, April 2021.

[16] European Central Bank, "Digital euro – Prototype summary and lessons learned." [Online], Available: https://www.ecb.europa.eu/pub/pdf/other/ecb.prototype_summary20230526~71d0b26d55.en.pdf, 2023.

[17] European Central Bank, "A stocktake on the digital euro." [Online], Available: https://www.ecb.europa.eu/euro/digital_euro/timeline/profuse/shared/pdf/ecb.dedocs231018.en.pdf, October 2023.

[18] A. D. Iorio, A. Kosse, and I. Mattei, "Embracing diversity, advancing together – results of the 2023 BIS survey on central bank digital currencies and crypto." Bank for International Settlements, June 2024.

[19] J. Katz and Y. Lindell, *Introduction to modern cryptography*. CRC Press, third ed., 2020.

[20] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system." https://bitcoin.org/bitcoin.pdf, 2008.

[21] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, p. 120–126, feb 1978.

[22] C. Sillaber and M. Eggen, "The Digital Euro: An (almost) perfect equivalent to cash?," *SSRN Electronic Journal*, December 19 2023. The German version of this paper was first published in Recht Digital 2023, 501.

[23] C. Sillaber and M. Eggen, "Privacy in payments: What a CBDC can do better than commercial bank money," *Zeitschrift für Bankrecht und Bankwirtschaft*, vol. 36, no. 4, pp. 267–276, 2024.

[24] SIX Group, "Four-Party Model, Fee Structure and Interchange in Debit Card Business." [Online], Available: https://www.six-group.com/dam/download/banking-services/debit-and-mobile-services/en/learning-nugget/learning-nugget-vier-parteien-modell-en.pdf, 2021.

Erklärung

Erklärung gemäss Art. 30 RSL Phil.-nat. 18

Ich erkläre hiermit, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe r des Gesetzes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.

Für die Zwecke der Begutachtung und der Überprüfung der Einhaltung der Selbständigkeitserklärung bzw. der Reglemente betreffend Plagiate erteile ich der Universität Bern das Recht, die dazu erforderlichen Personendaten zu bearbeiten und Nutzungshandlungen vorzunehmen, insbesondere die schriftliche Arbeit zu vervielfältigen und dauerhaft in einer Datenbank zu speichern sowie diese zur Überprüfung von Arbeiten Dritter zu verwenden oder hierzu zur Verfügung zu stellen.

Zürich, 28.01.2025
Ort/Datum

Unterschrift