Project Tourbillon

Exploring privacy, security and scalability for CBDCs

Final Report

November 2023

BIS Innovation Hub

SCHWEIZERISCHE NATIONALBANK
BANQUE NATIONALE SUISSE
BANCA NAZIONALE SVIZZERA
BANCA NAZIUNALA SVIZRA
SWISS NATIONAL BANK

Publication date: November 2023

ISBN 978-92-9259-711-5 (online)

Executive summary

The use of cash is in decline worldwide as digital payments continue to grow. Over the past decade, the number of cashless payments has grown at an annual rate of 16%, with more than one trillion transactions in CPMI countries alone.[1] In this context, concerns about the potential erosion of privacy are being raised.[2] Unsurprisingly, public consultations by central banks on retail central bank digital currencies (CBDCs) show that privacy is a fundamental user requirement.[3]

Privacy is the right to keep personal information secret or known only to a trusted group of people. This implies that there are different levels of privacy. According to Bank of Canada et al (2021), payments can be (i) *confidential*, where only trusted parties see personal information (such as credit card payments); (ii) *pseudonymous*, where identifiers or public addresses can be used to identify an individual (such as bitcoin transactions); or (iii) *anonymous*, where parties to a transaction cannot be identified (such as cash payments). However, privacy and data protection need to be balanced with other public policy objectives, in particular anti-money laundering and combating the financing of terrorism (AML/CFT) and countering tax evasion.

Project Tourbillon introduces a new privacy paradigm that balances user needs and public policy objectives: *payer anonymity*. For example, a consumer who pays a merchant using CBDCs does not disclose personal information to anyone, including the merchant, banks and the central bank. However, the identity of the merchant is disclosed to the merchant's bank (as part of the payment) but is kept confidential there. The central bank does not see any personal payment data but can monitor CBDC circulation at an aggregate level.

In addition to privacy, CBDCs must meet several other requirements from users and other stakeholders (such as banks and regulatory authorities) as well as its own public policy objectives.[4] Project Tourbillon addresses three features simultaneously:

- **privacy** – by enabling payer anonymity;

---

1 See Bank for International Settlements (BIS) Committee on Payments and Market Infrastructures (CPMI) Red Book statistics. The Red Book statistics on payments and financial market infrastructures in the 27 CPMI member jurisdictions are collected annually by BIS. For more information and the latest 2021 data, see www.bis.org/statistics/payment_stats.htm.

2 See Coy (2022).

3 The ECB (2021) notes that 43% of respondents to a public consultation on the digital euro ranked privacy as the most important aspect of the digital euro, well ahead of other features, such as security (18%). The Bank of England (2023b) notes that the majority of the 50,000 respondents to a public consultation on the digital pound in 2023 expressed concerns about privacy, programmability and the decline of cash.

4 As an example, the Bank of England (2023a) has highlighted the following design priorities for a digital version of the pound: privacy, security, resilience, performance, extensibility and energy usage. Additionally, the European Central Bank (2023b), in their third report, added that banks must provide core services to end users for offline functionality and should provide conditional payments (not programmable money).

- **security** – by implementing quantum-safe cryptography; and
- **scalability** – by testing the prototype's ability to handle a growing number of transactions using payment data.

In order to evaluate the trade-offs between these three features, two distinct prototypes based on the eCash design described by Chaum (1982) were built as part of project Tourbillon: eCash 1.0 (EC1)[5] and eCash 2.0 (EC2)[6]. Achieving this required meticulous calibration and precision, much like the inner workings of a tourbillon – a high-precision mechanical part in a watch.

Project Tourbillon shows that it is feasible to implement a design that provides payer anonymity. The project demonstrated that both prototypes are scalable and can handle a growing number of transactions. It also demonstrated that quantum-safe blind signatures, a cryptographic technique used to ensure anonymity, can be implemented. However, the implementation proved challenging. Quantum-safe cryptography exhibited slow performance and limited functionality, with throughput reduced by a factor of 200 compared to so-called classic cryptography, highlighting the need for further research and development. Finally, a comparison of the two prototypes illustrates the trade-offs between privacy and security: EC1 provides unconditional payer anonymity but EC2 has more resilient security features allowing for better protection against counterfeiting.

The Tourbillon project is a first step in exploring privacy, security and scalability in an eCash CBDC design. Future work can be categorised into three areas. First, further developing quantum-safe cryptography to make it easier to implement and deploy. Second, enhancing the design to improve speed and functionality, as well as cover more use cases. Third, addressing viability issues by exploring sustainable business models.

[5] See Chaum et al (2021).

[6] See Chaum and Moser (2022).

Acronyms, abbreviations and definitions

AML	Anti-money laundering.
BIS	Bank for International Settlements.
BISIH	BIS Innovation Hub.
Blind signatures	Cryptographic technique in which the content of a message is obscured (blinded) before it is signed.
CBDC	Central bank digital currency.
CBDC coin	A digital coin signed by the central bank.
Coin	A digital file with a serial number, not (yet) signed by the central bank.
CFT	Combating the financing of terrorism.
CNSA	Commercial National Security Algorithm.
CPMI	Committee on Payments and Market Infrastructures.
CPU	Central processing unit.
Digital signatures	The result of a cryptographic transformation of data that provides a mechanism for verifying origin authentication, data integrity, and signatory non-repudiation.
DLT	Distributed ledger technology.
E2EE	End-to-end encryption.
EC1	eCash 1.0.
EC2	eCash 2.0.
GPU	Graphics processing unit.
Hash	A one-way mathematical function that converts any digital data into a fixed number of alphanumerical characters
KYC	Know your customer.
NIST	National Institute of Standards and Technology.
PoS	Point of sale.
Public-key cryptography	Encryption system that uses a public-private key pair for encryption and/or digital signature.
rCBDC	A retail CBDC is a broadly available general purpose CBDC that can be used by the public, for day-to-day payments.
RSA encryption	Type of public-key cryptography widely used for data encryption over the internet, named for its inventors: Ronald Rivest, Adi Shamir and Leonard Adleman.
RTGS	Real-time gross settlement.
SIC	Swiss Interbank Clearing.
SNB	Swiss National Bank.
TPS	Transactions per second.
wCBDC	A wholesale CBDC is available to commercial banks and other financial institutions.
QR code	Quick response code.
QSC	Quantum-safe cryptography.

1 Introduction

A central bank digital currency (CBDC) is a digital payment instrument, denominated in the national unit of account, that is a direct liability of the central bank (Bank of Canada et al (2020)). In addition to the payment instrument, a CBDC system (or arrangement) includes the frontend, backend and communication infrastructure needed for a payer to transfer an amount of CBDC to a payee.

There are two types of CBDCs: *retail* and *wholesale*. A retail CBDC (rCBDC) is a broadly available general purpose CBDC that can be used by the public. A wholesale CBDC (wCBDC), on the other hand, is accessible only to financial institutions of significance to monetary policy implementation, financial stability and/or the smooth functioning of the payments and settlement infrastructure.[7] Currently, no country has yet fully implemented a wCBDC but 11 countries (Anguilla, Antigua and Barbuda, Bahamas, Dominica, Grenada, Jamaica, Montserrat, Nigeria, Saint Kitts and Nevis, Saint Lucia and Saint Vincent and the Grenadines) have implemented a rCBDC and China is testing the digital renminbi in several cities larger than many countries.[8] Additionally, the European Central Bank (ECB) and Bank of England (BoE) are exploring the digital euro and the digital pound, respectively.

Depending on the jurisdiction, introducing a CBDC is seen, inter alia, as a way to upgrade domestic payments rails (eg to complement cash or speed up government transfers), improve financial inclusion, counter currency substitution and/or enhance cross-border payments. But introducing a CBDC also poses many challenges, both in terms of ongoing operations and implications for the financial system. A rCBDC may, for example, lead to the disintermediation of banks if the populace substitutes commercial bank money for central bank money at scale. More expensive and less stable funding for banks could potentially spur financial instability.

In designing a rCBDC, central banks need to consider the requirements of users and other stakeholders (such as banks and regulatory authorities) as well as its own public policy objectives. The prioritisation of these requirements and objectives will determine the ultimate features included in the design of a CBDC and the wider CBDC system.[9] As an example, BoE (2023a) has highlighted the following design priorities for a digital version of the pound: privacy, security, resilience, performance, extensibility and energy usage. Additionally, the ECB (2023b), in its third report, added

[7] To distinguish wCBDC from the sight deposits or reserve balances that financial institutions currently hold with central banks in book entry form, it is often assumed that wCBDC are tokenised, ie, based on distributed ledger technology (DLT).

[8] Launched retail CBDCs can be explored via the Atlantic Council's CBDC tracker and Reuters reporting on the digital renminbi.

[9] Bank of Canada et al (2020) divides the features of a CBDC into three categories: (i) instrument features such as convertibility, convenience and acceptance; (ii) system features such as cyber security, resilience, instantaneous settlement, availability, scalability, throughput, privacy and interoperability; and (iii) institutional features such as robust legal frameworks and adherence to standards.

that banks must provide core services to end users for offline functionality and should provide conditional payments (not programmable money).

Not all requirements or objectives may be achievable given current technology, policy objectives or law. Moreover, some requirements and objectives may conflict with others and hence designing a CBDC may involve trade-offs between one requirement or policy objective and another.

Achieving privacy while combating illicit payments is one such challenge. Privacy is an important user requirement for rCBDCs. The majority of respondents to recent public consultations, cited in ECB (2021) and Cunliffe (2023), highlighted the importance of privacy in CBDCs. However, individual privacy protections need to be balanced with public policy objectives, in particular anti-money laundering and combating the financing of terrorism (AML/CFT) and countering tax evasion. Project Tourbillon explores the nexus of three important CBDC system requirements that current live implementations, pilots and studies have highlighted as particularly challenging: privacy, security and scalability.[10]

Privacy

Privacy is the right to keep personal information secret or known only to a trusted group of people. Payment systems provide different levels of privacy. Bank of Canada et al (2021a) posits three levels: confidential, pseudonymous, and anonymous.

- In c*onfidential* payments, an individual's identity is known only by a narrow set of trusted parties (eg involved banks or payment system providers in card payments).

- In *pseudonymous* payments, an individual's identity is not known, but there may be identifiers or other information that can be used to link the payment to an individual (eg as in bitcoin).

- *Anonymity* is the ability of individuals to remain unidentifiable in a payment transaction (eg as in cash).[11]

Project Tourbillon introduces *payer anonymi*ty to ensure anonymity for senders while combating illicit payments.[12]

---

[10] For example, the digital euro project (ECB 2023a) as well as China's digital renminbi (PBOC 2021) have mentioned privacy, cyber security and scalability as important features to be considered. The digital currency initiative summarises this by noting that it is part of a "larger CBDC initiative which combines technology research in security, privacy, and scalability with user research into the design of digital currency systems" (DCI).

[11] However, de Montjoye et al (2015) note, that even some anonymous payment data, with sufficient metadata, can be used to re-identify individuals.

[12] The Tourbillon prototypes cannot protect a consumer's anonymity against user behaviour or external tools. In particular, a consumer can always choose to reveal their identity to the merchant, to banks, or even to third parties. External tools like reward cards or facial recognition software in stores can also link consumers to payments.

Payer anonymity

Payer anonymity provides cash-like anonymity to payers in a payment, but not to payees. For example, a consumer who pays a merchant using CBDCs does not disclose personal information to anyone, including the merchant, banks and the central bank. However, the identity of the merchant is known to the payer and is only disclosed to the merchant's bank (as part of the payment) where it is kept confidential. The central bank does not see any personal payment data but can monitor CBDC circulation at an aggregate level.

Security

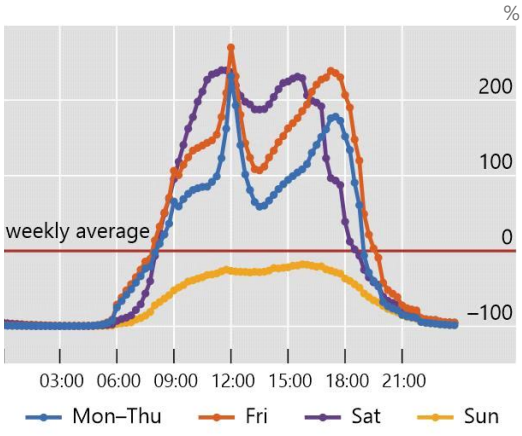
Security in digital payment systems aims to maintain, inter alia, the confidentiality and integrity of payments data as outlined by CPMI-IOSCO (2012). Cryptography can be used to uphold the confidentiality of the payment data (providing privacy) and the integrity of the payment system as a whole (preventing double spending or counterfeiting). Achieving robust security relies on secure cryptography that safeguards against both current and future cyber threats, such as attacks from quantum computers (Box A).

Scalability

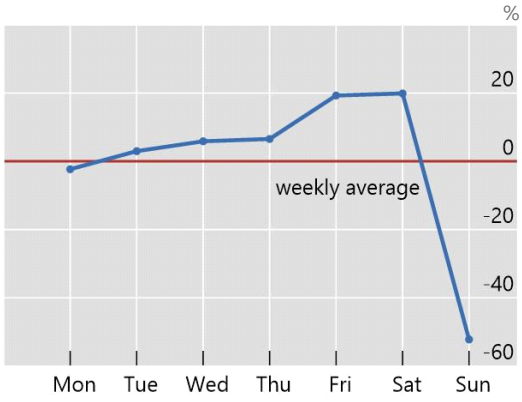
Scalability is the ability to adjust to changing demands without compromising performance, quality or cost. In the context of a payment system, scalability is about ensuring smooth functioning during peak surges and times of sustained elevated demand. Typically, payment systems operate on distinct daily and weekly cycles relative to the weekly average (illustrated in Graph 1). As such, it is important for a payment system to not only settle the average volume of payments quickly, but also its seasonal peaks and sustained loads too.

Payment seasonality — Graph 1

A. Intraday volume of payments relative to weekly average



B. Daily volume of payments relative to weekly average



Sources: Swiss National Bank; Worldline; PostFinance; Monitoring Consumption Switzerland.

Project objectives

The project assesses the degree to which privacy, security and scalability can each be achieved in the context of two CBDC designs proposed by Chaum and Moser (2022). Specifically, the objectives were:

- **Privacy** – enabling payer anonymity while countering illicit payments;
- **Security** – evaluating the implementation of quantum-safe cryptography (QSC); and,
- **Scalability** – testing the prototype's scalability using payment data.

Both designs are based on and extend the eCash design proposed by Chaum (1982).[13] The assessment was done by building a prototype for each design, and then analysing and testing how private, secure and scalable each prototype is.

This report summarises the work carried out in project Tourbillon. Section 2 describes the two eCash based prototypes and how they were tested. Section 3 presents the results and discusses privacy, security, scalability, trade-offs and implementation considerations. Section 4 concludes with an outlook on possible future work.

[13] The history of eCash can be found on David Chaum's website (at https://chaum.com/ecash/).

Box A: Threat of quantum computers to cryptography

Quantum computing makes use of the principles of quantum mechanics, which allow for a system to exist not only in a zero or one state, but rather in a composition of the two (Schumacher 1995). This property, known as superposition, enables quantum computers to solve certain computational problems faster than classical computers. Such computing may bring many benefits but it also poses threats.

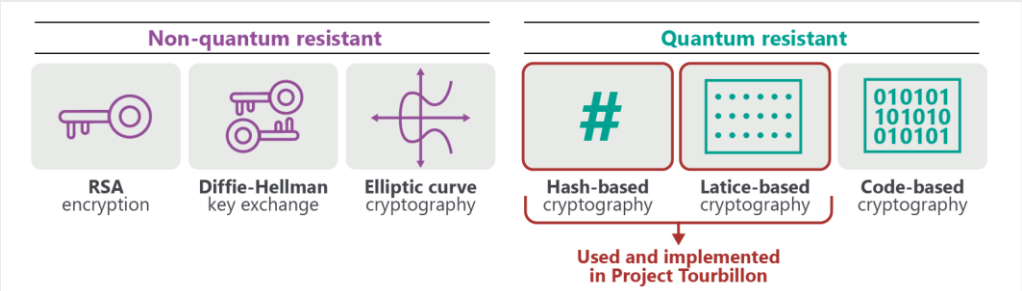
The threat of quantum computers to information security lies in their potential ability to break asymmetric cryptography (CNSS 2015), such as RSA-based encryption (Rivest et al 1978), Diffie-Hellman key exchange and digital signatures. These cryptographic schemes protect much of today's digital infrastructure and communications systems (Graph A.1). If this threat is realised, it could lead to data breaches and a profound loss of trust in all digital systems, including financial ones.

Quantum computers are still in the research and development phase, and there are many technical challenges to overcome before they become practical for widespread use. However, it is important that quantum-resistant algorithms are researched, implemented and deployed as soon as possible.

Project Tourbillon uses two approaches to achieve quantum-resistance. First, it uses hash functions that quantum computers cannot break. A hash function is the most fundamental primitive in cryptography; if hash functions were ever broken, it would dismantle all of cryptography because of hardness assumptions. Trying to reverse a hash function would be like trying to unscramble an egg.

Second, Project Tourbillon implements a lattice-based blind signature scheme, described in Beullens et al (2023). Lattice-based cryptography is a potentially quantum-resistant approach, used in NIST-standardised quantum-safe schemes, namely CRYSTALS-Kyber and CRYSTALS-Dilithium, but customised for blind signatures. It relies on the hardness of a lattice problem for quantum computers and is optimised for use in devices with limited resources such as smartphones.

Cryptography and quantum-resistance — Graph A.1



2 The prototypes

This section describes the two prototypes built as part of Project Tourbillon: **EC1**[14] and **EC2**.[15,16] It begins with a high-level solution architecture, followed by in-depth technical explanations of EC1 and EC2, including workflows, and concludes with a discussion of the implementation and testing setup.

The project focuses on consumer-to-merchant payments via mobile applications.[17] Extensions to other use cases, such as peer-to-peer payments, are in principle straightforward. However, for the sake of simplicity were not within the scope of this project.

2.1 High level solution architecture

Both prototypes leverage the existing two-tier banking system and involve four parties: a central bank, commercial banks (or simply banks), consumers and merchants (Graph 2). Consumers and merchants have deposit accounts with banks and banks have reserve accounts with the central bank.

Consumers and merchants must be initially onboarded by a bank, ensuring that know-your-customer (KYC) procedures are met. Once onboarded, consumers and merchants can install and use the Tourbillon app (Graph 3) on their mobile devices.

The interfaces are familiar to both consumers and merchants: the consumer Tourbillon app allows consumers to withdraw, hold CBDCs (via self-custody) and make payments. The merchant Tourbillon app allows merchants to request payments, receive payments and view the status of payments.[18]

The eCash-based design relies on unsigned digital coins and digital coins signed by the central bank – CBDC coins. An unsigned coin is a consumer generated digital file with a unique serial number that is not (yet) signed by the central bank. Once a coin is signed by the central bank, that coin becomes a CBDC coin.

The CBDC coin is designed as a single-use CBDC. Unlike banknotes, the CBDC cannot be reused by the recipient (merchant) for further payments. Instead, the CBDC must be redeemed with the central bank (via the merchant's bank). This has two

---

14 Chaum et al (2021).

15 Chaum and Moser (2022).

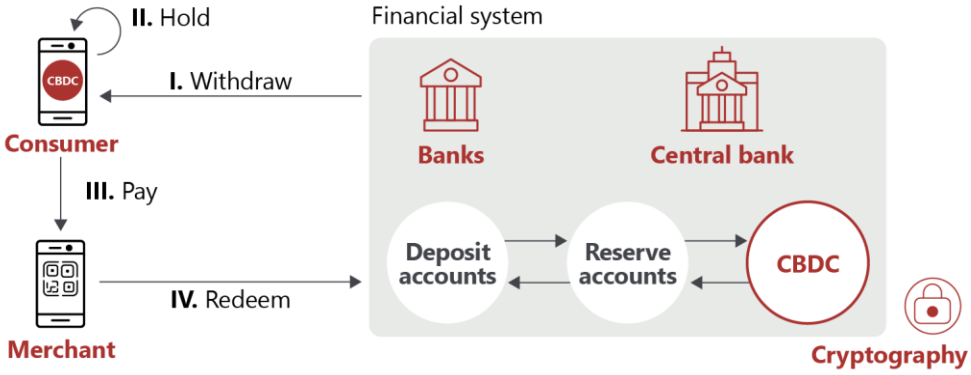
16 The development of the prototypes was supported by IBM and Currency Network.

17 Two apps were developed: a consumer app and a merchant app. It is possible to merge both apps for consumer-to-merchant payments as well as peer-to-peer payments.

18 The Tourbillon app for merchants does not contain a wallet because we focus on the consumer-to-merchant payment.

effects: (i) it provides protection against double-spending; and (ii) it ensures that a merchant's sale is recorded at their bank.

Tourbillon high-level architecture Graph 2



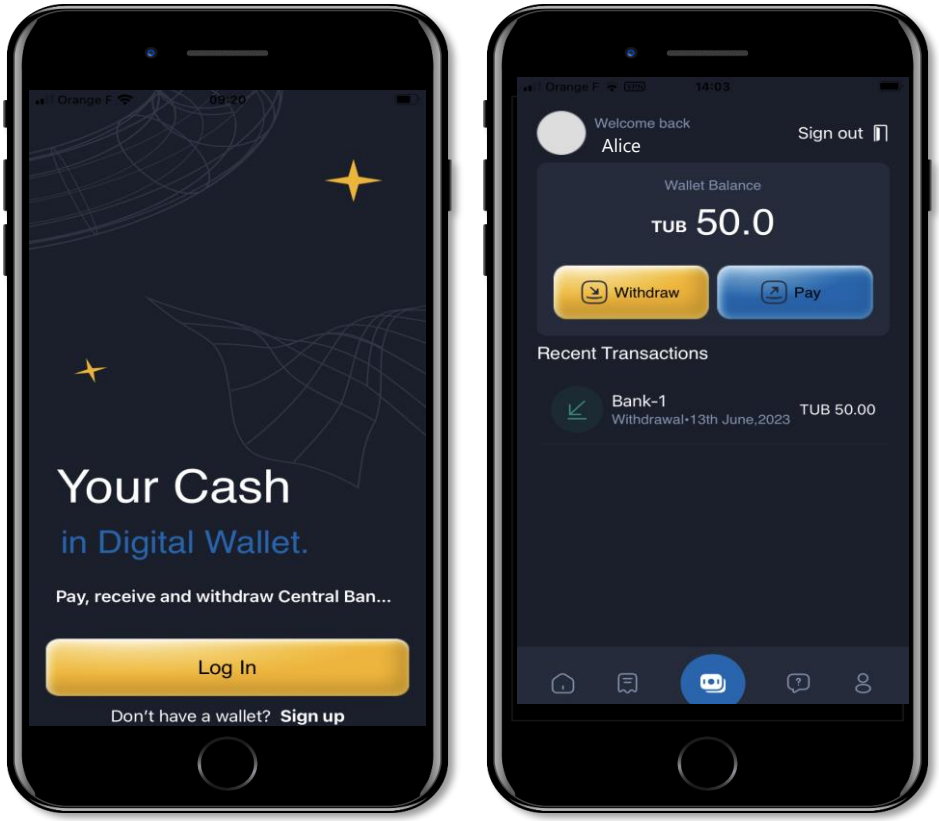
eCash-based designs use coins with fixed denominations. Project Tourbillon established four distinct denominations that follow the power of two, meaning there are coins with a value of 1, 2, 4 and 8.[19] For each withdrawal request, an algorithm is used to optimise the denomination of the CBDC coins used, minimising their quantity while ensuring that the consumer always has the correct change to pay any amount (Annex A). Whenever the consumer spends CBDC coins, the algorithm assesses the optimal denomination of the remaining coins. If this is not the case, a background process is initiated to rebalance the denominations (Annex B).

An overview of the withdraw, hold, pay and redeem processes is illustrated in Graph 2. To use CBDCs, consumers must first request the withdrawal (I) of CBDCs through their bank. The bank debits the consumer's deposit account and forwards the withdrawal request to the central bank. Upon receipt, the central bank issues CBDCs, debits the bank's reserve account and forwards the CBDCs to the consumer. The consumer can then hold (II) the CBDCs and later use them, possibly with other previously withdrawn CBDCs, to pay (III) various merchants for goods and/or services.

Upon receiving CBDCs as payment, the merchant submits (IV) the single-use CBDCs to her bank, which then forwards them to the central bank. After verifying the authenticity and validity of the CBDCs, the central bank credits the reserve account of the merchant's bank. The merchant's bank then credits the merchant's deposit account. In this process, the consumer can remain anonymous, similar to a cash payment, while the merchant's sale (income) is recorded with her bank (providing payer anonymity, as described in Section 1).

[19] Van Hove (2001) discusses a broad set of similar denomination optimization problems and shows that the power of two denomination yields one such optimal denomination.

Tourbillon application Graph 3



2.2 eCash 1.0 (EC1)

EC1 starts with the withdrawal of CBDC by the consumer (Graph 4). In Project Tourbillon, toubie [TUB] is the unit of account. The consumer (1) logs into her Tourbillon app and requests to withdraw a certain amount of CBDC – 15 toubies, in the example shown. For 15 toubies, the app generates four specific coins with a unique identifier of different denominations[20] and uses cryptography to (2) hash and then blind them.[21] Blinding allows the consumer to obtain a signature on each coin without revealing the unique identifier to the bank or central bank (Box B).[22]

The blinded coins are then sent to the bank, which (3) blocks 15 toubies on the consumer's deposit account. The bank forwards the blinded coins to the central bank. The central bank (4) debits 15 toubies from the bank's reserve account and signs

[20] To withdraw 15 toubies, the app generates four coins (1, 2, 4 and 8). With these four coins, the consumer always has the correct change to spend any whole amount between 1 and 15 toubies.

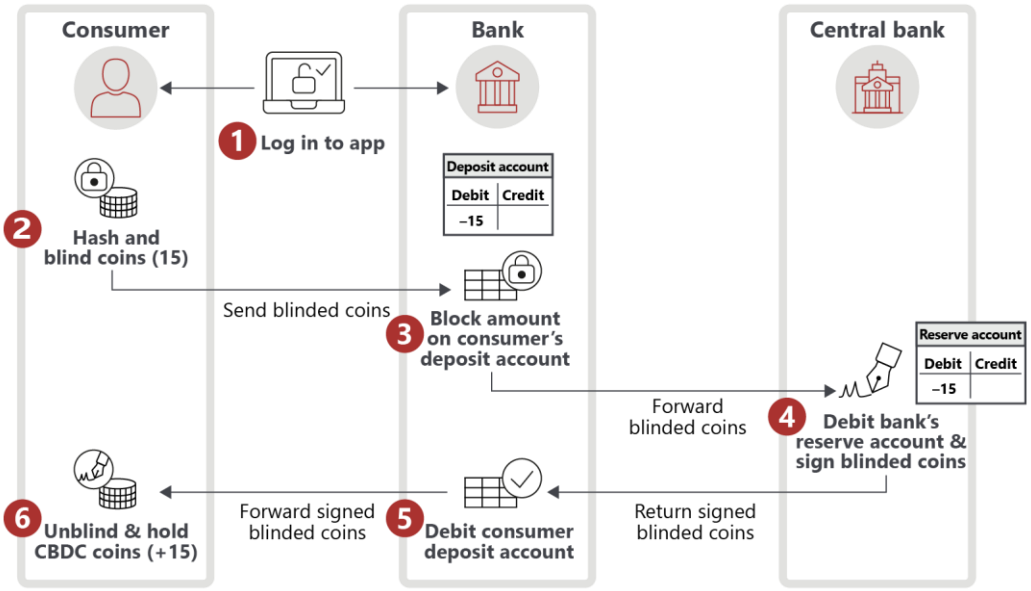
[21] A hash (function) is a one-way mathematical function that converts any digital data into a fixed number of alphanumerical characters.

[22] Note that it is the consumer who blinds the coins, not the bank or the central bank, and that it is therefore only the consumer who can unblind the coins. In other words, neither the bank nor the central bank get to see the unblinded coin at the time of withdrawal, nor does the consumer have to trust the bank or the central bank to do the blinding correctly.

the blinded coins with the central bank's private key for the respective denominations, at which point it issues CBDCs.[23] The central bank sends the signed but still blinded CBDC coins to the bank, which (5) debits the consumer's deposit account.

Finally, the bank forwards the blinded CBDC coins to the consumer. Upon receipt, the app (6) unblinds and stores the CBDC coins in the wallet, where they are stored together with any CBDC coins already held in self-custody. As a result, neither the bank nor the central bank know which specific coins the consumer owns. The consumer can use the coins to pay anonymously, as no one can link the coin to the consumer's identity.

Withdrawal in eCash 1.0 (EC1) Graph 4



Once the consumer holds CBDC coins in her wallet, she can make payments to a merchant at the point-of-sale (Graph 5). After the consumer (1) selects an item to purchase and agrees with the merchant to pay the price – 10 toubies, for example – the merchant uses his app (2) to create a pending transaction at his bank and to generate a quick response (QR) code that contains all relevant payment information (eg amount, deposit account of the merchant and a transaction number).

The consumer uses her app (3) to scan the QR code, which transfers all the information to the consumer's app. The consumer's app selects the required CBDC coins from the wallet[24] and (4) sends them to the merchant's bank, which links them

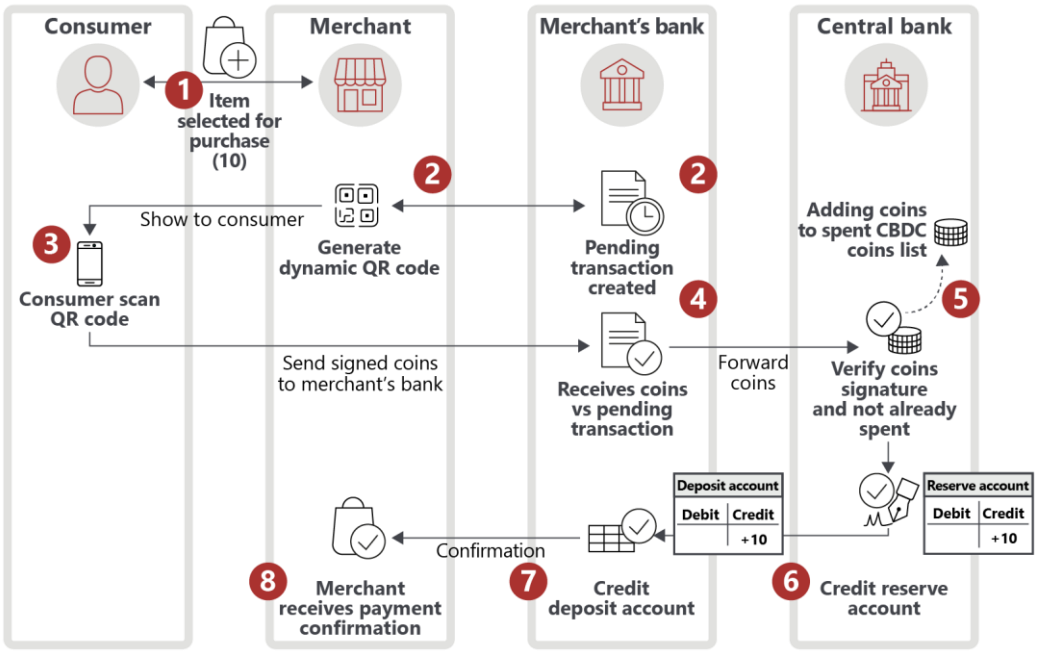
[23] The central bank has a different public / private key pair for each denomination of value. The private key is used for signing the blinded coins and the public key is used by the consumer as input for blinding the coin and by the central bank to check the validity of a CBDC coin when used for payment.

[24] In our case, the wallet selects two CBDC coins with denominations 2 and 8 to be sent to the merchant's bank. Two CBDC coins with denominations 1 and 4 remain in the wallet. The coin with denomination 4 needs to be rebalanced later into one coin of denomination 2 and two coins of denomination 1 to ensure the user can spend any whole amount between 1 and 5.

to the pending transaction and forwards the CBDC coins to the central bank. The central bank (5) verifies the signature. The central bank then checks that the CBDC coins have not already been spent against the list of spent CBDC coins.[25] If everything is in order, the central bank redeems the CBDC coins and immediately adds them to the spent list to ensure that the CBDC coins cannot be spent again, then (6) credits the bank's reserve account, and sends a confirmation to the bank. Next, the bank (7) credits the merchant's deposit account and (8) notifies the merchant.

Following a successful transaction, the consumer's app employs an algorithm to evaluate the denominations of the remaining CBDC coins. This evaluation aims to minimise the number of coins while guaranteeing the consumer always possesses the exact change required for any transaction. Should this assessment reveal suboptimal coin denominations, a rebalancing process is initiated, as outlined in Annex B.

Payment and redemption in eCash 1.0 (EC1) Graph 5



The central bank has the ability to monitor, at an aggregate level, the issuance, payment and redemption of CBDCs, without knowing the identity of the consumer or merchant for two main reasons. First, during withdrawal, a CBDC coin must be signed by a central bank to be considered valid. Second, once a CBDC coin has been used for a payment, it must subsequently be redeemed at the central bank.

In EC1, the existence of the spent coins list presents a potential security issue with regard to counterfeiting. That is, if an attacker managed to steal the private key

[25] In EC1, checking against the spent CBDC coins list is necessary because, unlike with DLT, there is no record of coin ownership, not even pseudonymously. In EC2, the same goal is accomplished by listing the unspent CBDC coins at withdrawal and removing them from the list once they are spent.

of the central bank and started to sign their own CBDC coins, then the central bank may not necessarily notice. Thus, counterfeit resistance would rely on the overall cyber security and cyber resilience of the central bank's public and private keys.[26] However, EC2 addresses this issue with the introduction of an unspent coins list.[27]

[26] The private key is the sensitive component of the key pair that must be kept secret — central banks already protect sensitive keys. Thus, the existing cyber security of central banks must extend to the signing keys as well.

[27] Since EC2 only allows payments corresponding to pre-published hash values, so-called "fail-stop signatures" can be easily and efficiently incorporated, further enhancing protection against counterfeiting (Pfitzmann 1991).

Box B: Blind signatures and security

Blind signatures, first introduced by Chaum (1982), provide privacy by allowing a user to obtain a signature from a signer without the signer knowing the contents of the message being signed. Blind signatures have since found numerous applications, beyond the original eCash, in electronic voting and privacy technology in general.

The blind signature concept is akin to the dated system of notary embossing stamps. Essentially, a person can create any document, such as a banknote, with their own unique randomly selected serial number on it and protect the confidentiality of the document's content by enclosing it in a sealed paper envelope. The envelope is then given to the notary, usually along with a fee for "signing". When the envelope is returned, inspection confirms that the special embossing has been applied to it and that it remains sealed. So, when the person removes and discards the envelope, they have an unforgeable embossed document with a unique serial number that no one else has ever seen.

In a digital cash system, blind signatures allow users to obtain valid coins signed by a central authority (eg a central bank) while keeping their ownership of the specific coins – and thus payments made with these coins – private and preventing third parties, including the central bank, from tracing individual spending patterns.

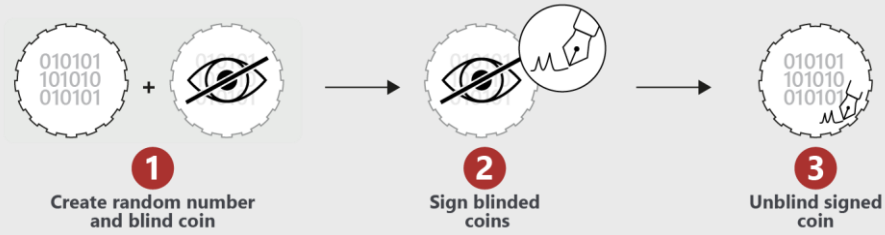
Blind signatures follow a three-step process (illustrated in Graph B.1). First, the payer creates a coin by choosing a random number and blinds it (1) using a random blinding factor. Second, the central bank receives the blinded coin and applies its digital signature (2). Since the coin is blinded, the central bank has no knowledge of the actual random number of the coin. Third, the payer unblinds (3) the received signed blinded coin by removing the blinding factor but keeping the signature on the original coin. The payer can now use the unblinded coins to pay digitally.

Key benefits of using blind signatures are:

1. **Privacy:** the central bank (or any third party) cannot link the coin's spending history back to the payer because nobody has ever seen the unblinded random number.

2. **Counterfeit resistance:** the payer cannot tamper with the signature when blinding/unblinding and the central bank can verify the signature's validity. Spent coins are registered on a spent coins list and cannot be spent again.

Blind signatures — Graph B.1

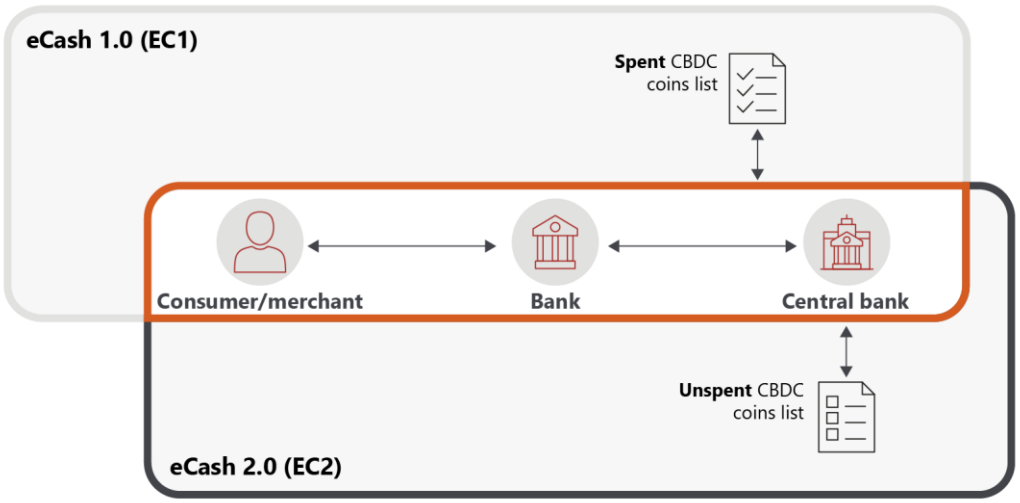


2.3 eCash 2.0 (EC2)

EC1 and EC2 have similar designs but differ in how CBDCs are recorded by the central bank (Graph 6). In EC1, unique identifiers from CBDC coins are recorded upon redemption; thus, EC1 keeps a *spent CBDC coins list*. In contrast, in EC2, unique identifiers from CBDC coins are recorded upon issuance; thus, the central bank maintains an *unspent CBDC coins list*.

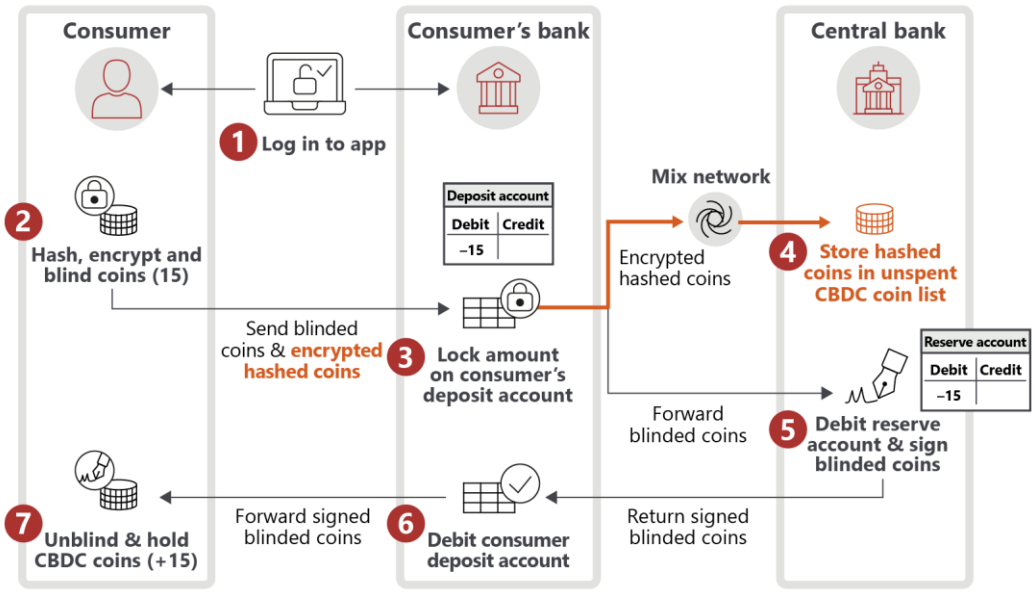
Recording unique CBDC coin identifiers at the time of withdrawal introduces privacy concerns, potentially compromising the anonymity of consumers by linking their identity to specific CBDC coins. To prevent this, EC2 uses a mix network (Box C). The following description focuses on the main differences between EC1 and EC2.

Comparison of eCash 1.0 and eCash 2.0 Graph 6



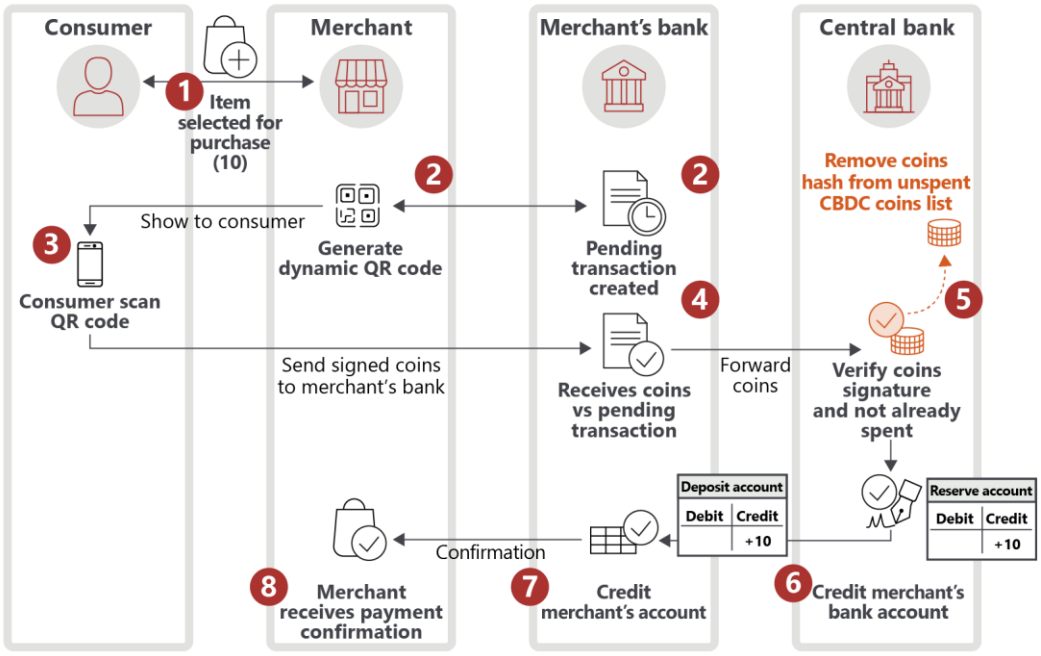
The first difference is that at the time of withdrawal (Graph 7), the consumer's wallet not only (2) blinds hashed CBDC coins as in EC1 but also sends the hashed CBDC coins to a mix network. After the mixing, the hash CBDC coins are received by the central bank (4) and added to the unspent CBDC coins list. The rest of the withdrawal is the same as in EC1.

Withdrawal in eCash 2.0 (EC2) Graph 7



The second difference occurs during payment and redemption (Graph 8). The central bank's task is (5) to verify that the submitted CBDC coins are indeed on the unspent CBDC coins list. If so verified, the CBDC coins are removed, rendering them invalid for future payments.

Payment and redemption in eCash 2.0 (EC2) Graph 8



A major advantage of EC2 compared with EC1 is that CBDC coins can only be spent if they are registered by the central bank in the unspent CBDC coins list. This

renders counterfeiting highly difficult.[28] Therefore, if a counterfeiter spends any one of the CBDC coins listed, it can rapidly be detected. Such detection can occur, for example, when the holder of the legitimate CBDC coin unsuccessfully attempts to spend it.

Another practical advantage of EC2's approach to preventing double spending is that the list of unspent CBDC coins does not keep growing, as with the EC1 spent CBDC coins list. A more fundamental advantage of the "ready to spend" list of EC2 compared with the "already spent" list of EC1 is that it makes the total amount of outstanding CBDC coins transparent. This allows for the quick detection of counterfeiting, as explored in Annex C.

Overall, the design change in EC2 weakens privacy assumptions, which could compromise consumer anonymity. Although the mix network anonymises withdrawals, an attacker with access to information on withdrawals and payments across mix network batches might be able to link the identity of consumers to payments using statistical techniques.[29]

[28] An attacker could try to steal unspent CBDC coins by finding the preimage of a hashed unique identifier placed on the unspent CBDC coins list and attempt to pay with it. However, finding the preimage of a hashed unique identifier is virtually impossible, even for quantum computers. In the unlikely event that an attacker is successful, the affected consumer holding the original CBDC coins would detect the theft as soon as she tries to spend them and realises that the CBDC coins are invalid.

[29] However, this potential vulnerability could be eliminated with the Mix network design described in Chaum and Yaksetig (2023).

Box C: Mix networks

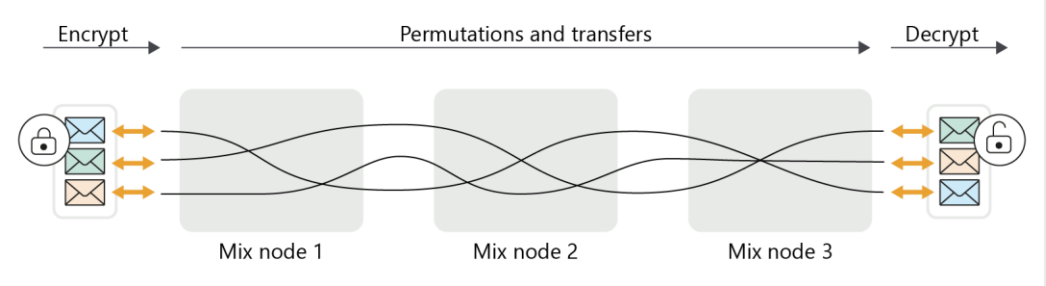
Mix networks, first introduced by Chaum (1981), are cryptographic protocols designed to enhance privacy in digital communications. They are a basic building block of digital privacy, with applications in a wide range of areas, including secure communications and anonymous internet browsing.

The assumption in mix networks that there is at least one honest party is akin to a parlour game in which each person sitting around a table takes the deck of cards from the person on their right and shuffles it hidden from view. If all players were to secretly record their shuffle re-ordering, the final order could be deduced, but if even one of the players is honest, and so does not record the random shuffle order they used, the result is a perfectly unknowable re-ordering. Instead of playing cards, mix networks route communications and payments so that they cannot be traced back to particular users.

The EC2 mix network processes batches of hash function images, each supplied by a different payer (or consumer) and allowed into the batch to be entered by their respective bank. The input batch is initially processed by a node that decrypts the items and randomly orders them. The next node takes this output as its input, processes and forwards it in the same way, and so forth. If there is at least one honest mix node, the cryptography hides the order of the inputs completely in the order of the outputs. (Graph C.1).

EC2 mixing has ephemeral teams of five nodes. Each team is chosen for just a single batch. The teams are formed randomly on a blockchain. Teams independently agree to process all messages beforehand, then mix and decrypt them, delivering the results to the recipients. Following completion, the team disbands and member nodes become available to be placed on a new random team.

Mix network (schematic) Graph C.1



2.4 Implementation and testing setup

To assess the prototypes' ability to withdraw, hold, spend and redeem CBDCs, a mobile application was developed for customers and merchants using four fictional banks and one central bank. Within this setup, withdrawals and payments can be executed end to end. They were conducted in the BIS cloud environment in combination with mobile devices.

To assess the feasibility of QSC, both prototypes have the capability to switch between so-called classical cryptography and QSC. Using the QSC setup, withdrawals and payments were executed end to end.

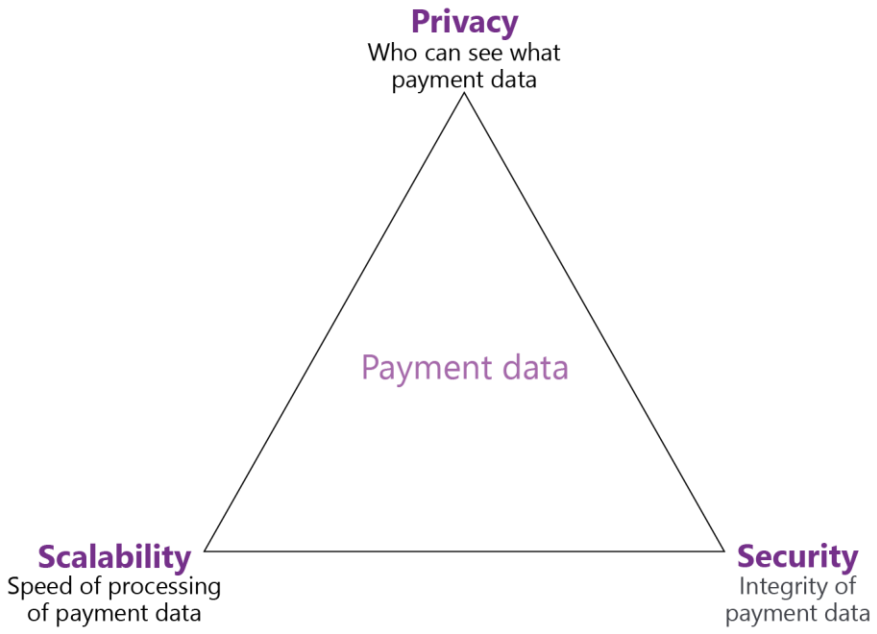
Finally, to assess scalability (ie performance), calibrated payment data was used to measure latency (in seconds) and throughput (in transactions per second (TPS)).[30] Moreover, scalability was measured in each prototype using both classical and QSC setups.

3 Results and considerations

This section discusses the results and considerations of building EC1 and EC2, emphasising privacy, security, scalability and their trade-offs. The prototyping has shown that these issues deal with different aspects of how payment data – consisting of information about the consumer, the merchant, the amount and additional metadata – is handled in digital payments (Graph 9).

Privacy is concerned with who can see what payment data: Tourbillon demonstrates payer anonymity. *Security* is about how to ensure the integrity of payment data: Tourbillon looks at counterfeiting resistance, even against future quantum computers. Finally, *scalability* is concerned with the speed and volume of payment data processing: Tourbillon evaluates throughput and latency for both designs, each tested with both classical cryptography and QSC.

Payment data: privacy, security and scalability — Graph 9



[30] Anonymised payment data were taken from retail transactions settled in the Swiss Interbank Clearing (SIC) system. SIC is a payment system that settles both large value as well as retail transactions.

3.1 Payer anonymity and other policy objectives

Project Tourbillon shows that it is feasible to implement a CBDC that provides payer anonymity while combating illicit transactions. In particular, a consumer paying a merchant with CBDCs is anonymous to all parties, including the merchant, banks and the central bank. However, the identity of the merchant is known to the payer and is only disclosed to the merchant's bank as part of the payment and is kept confidential thereafter. The central bank does not see any personal payment data but can monitor CBDC circulation at an aggregate level.

Commercial banks use existing procedures to combat illicit transactions. First, all users – consumers as well as merchants – of CBDCs must undergo a KYC process at a commercial bank to ensure that their identity is known and verified. Only users who have gone through this process can withdraw, hold, pay and redeem CBDCs. Second, the merchant's bank is responsible, as in today's two-tier financial system, for ensuring that transactions comply with regulatory requirements such as AML, CFT and combating tax evasion, as well as for taking necessary action if they do not.[31]

Tourbillon shows that central banks can monitor the aggregate use of CBDCs in real time without seeing any personal information. As part of the project, a dashboard was designed and implemented to allow central banks to view the amount of CBDCs in circulation and the aggregated rate of CBDC withdrawals and redemptions.

3.2 Quantum-safe cryptography

Project Tourbillon uses conventional RSA-based cryptography, which is vulnerable to future quantum computers.[32] The project assesses the feasibility of replacing the RSA-based cryptography with a quantum-safe, lattice-based scheme proposed by Beullens et al (2023) in both designs, detailed in Annex D. The implementation proved challenging, and QSC exhibited slow performance and limited functionality. This underscores the need for further research and development related to such algorithms generally.

Implementing quantum-safe blind signature schemes was difficult for two main reasons. First, the quantum-safe blind signature scheme is significantly more complex than its traditional RSA-based counterpart. This increased complexity has made it difficult to adapt to a protocol designed for classical cryptography. The full integration of quantum-safe blind signature schemes requires extensive integration work that would have been too time-consuming for project Tourbillon. As a result, the quantum-safe prototypes have two functional limitations: (i) they only allow

---

[31] The recommendations of the FATF (2023) for cash or electronic payments may apply to CBDCs. If so, the so-called "travel rule" requires the collection and sharing of the payer's transaction data along a payment chain, which could create barriers to the implementation of payer anonymity. A legal assessment of the applicable regulatory and compliance rules is beyond the scope of this project.

[32] RSA-based cryptography is used for blinding coins.

consumers to withdraw or pay with one CBDC coin at a time (making withdrawals and payments slow); and (ii) they do not support rebalancing.

Second, the quantum-safe blind signature scheme has higher computational and memory requirements compared with the RSA equivalent. This led to a significant drop in performance in scalability tests as shown in Section 3.3.

Further research and development are needed to ready QSC for end-to-end operational usage. In Tourbillon, for instance, the mix network is an additional system in which quantum resilience is needed.[33] Another important area of research is how the replacement of current cryptography with a quantum-safe equivalent could be safely implemented in a live system. This highlights both the critical need for effective quantum resilience in all aspects of digitisation (eg CBDCs and communications), and the importance of achieving such resilience without degrading the service.

3.3 Scalability

Tourbillon measures the performance of EC1 and EC2 and demonstrates that both prototype designs are scalable using the RSA-based blind signature scheme (Graph 10), but also that the quantum-safe equivalents slowed performance significantly (Graph 11).

RSA-base blind signature scheme

Tourbillon applies the following mechanisms to test system scalability of the RSA-based blind signature scheme:

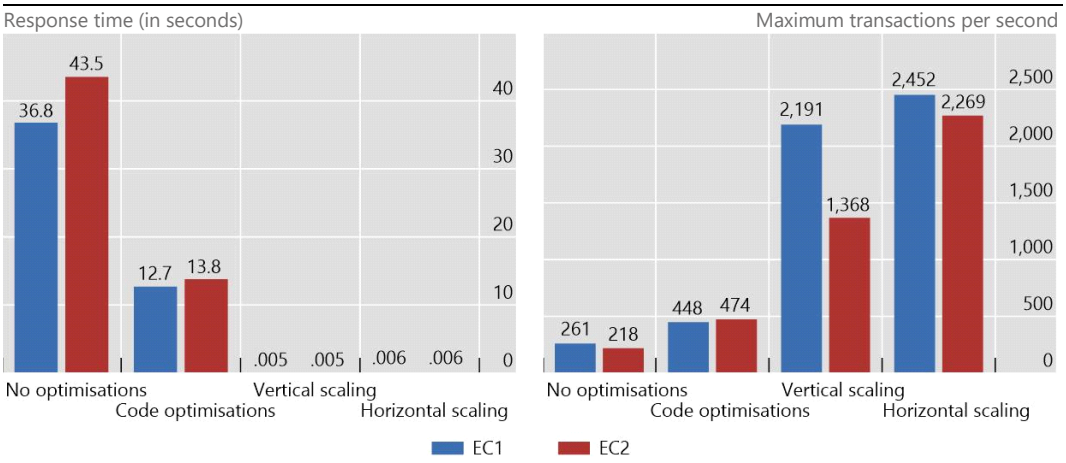
- *Code optimisation* for central and commercial banks with the objective of parallelising withdrawals, payments and redemptions.
- *Vertical scaling* involved increasing the performance – in terms of GPUs, CPUs and memory – of virtual machines in the system.
- *Horizontal scaling* involved duplicating the central bank's servers, increasing the capacity to balance a high volume of transactions.

Testing shows that scaling mechanisms improve throughput and latency. The measures taken – code optimisation, horizontal and vertical scaling – increased the maximum TPS from about 250 to over 2,450 for EC1 and over 2,250 for EC2. Additionally, the average response time of the central bank fell from around 40 seconds to almost five milliseconds (Graph 10).

[33] Quantum-safe mix networks, as proposed by Boyen et al (2020) for instance, could be implemented in the future.

A similar trend holds for the latency of withdrawals and end-to-end payments from consumer to merchant, both of which were executed within one second.[34]

Scalability of payments using RSA cryptography — Graph 10



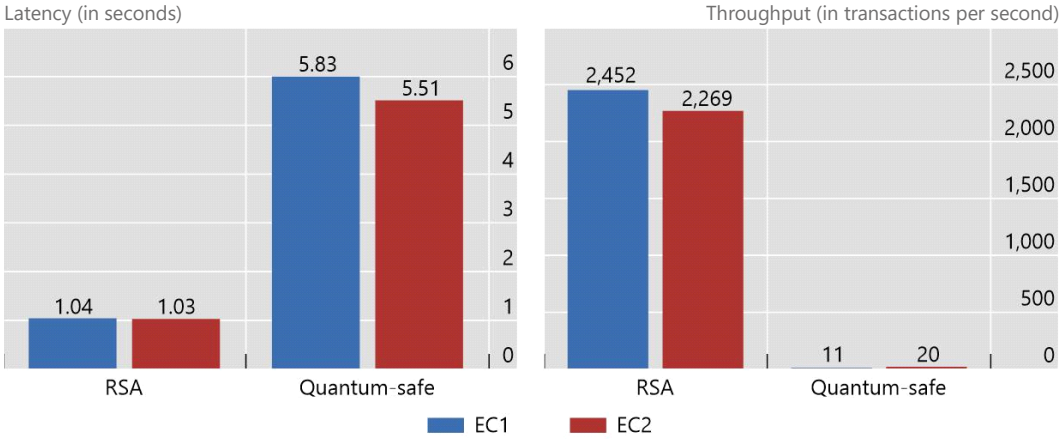
Quantum-safe cryptography

Introducing QSC increases the duration of end-to-end payments from consumer to merchant from just over 1 second to more than 5 seconds (Graph 11). At the same time, maximum throughput drops significantly from 2,452 TPS for EC1 and 2,269 for EC2 to 11 TPS for EC1 and 20 for EC2 – a reduction by a factor of more than 200. This is reflective of the less than straightforward nature of implementing and using the quantum-safe blind signature scheme.[35]

[34] Note, that denominations may have an impact on performance. Given the design of the eCash protocol, denominations have a direct correlation to the number of coins that need to be held. In particular, having larger denominations will reduce the number of coins needed during withdrawals or payments. Conversely, smaller denominations will increase the number of coins needed.

[35] Note that the scalability test with synthetic payment data measured the withdrawal of four CBDC coins. Since the quantum-safe blind signature scheme currently only allows consumers to withdraw one CBDC coin at a time, withdrawals are very slow. Functional improvements that allow parallel withdrawals would reduce the duration by a factor of four.

RSA vs quantum-safe cryptography Graph 11



3.4 Privacy and security trade-offs

A comparison of the two prototypes illustrates the trade-offs between privacy and security. EC1 provides very strong payer anonymity. However, its resistance to counterfeiting depends on the secrecy of the central bank's private signature. In addition, counterfeiting, however unlikely, can only be detected indirectly by investigating suspicious (large) redemptions.

EC2 offers a very high level of resistance to counterfeiting as well as rapid detection of it, overcoming the weakness of EC1. This comes at the cost of a more complex design introducing mix networks to ensure payer anonymity. Adding design complexity could introduce potentially exploitable vulnerabilities.

A threat analysis of EC1 and EC2 is needed to better understand potential vulnerabilities and the measures required to address them. A rudimentary threat analysis is presented in Annex C which highlights potential vulnerabilities.

3.5 Implementation considerations

Both EC1 and EC2 provide a payment process between consumer and merchant that builds on existing infrastructure and leverages the two-tier financial system. However, Tourbillon is a prototype focused on specific use cases and ideal payment scenarios. Further work is needed to include other use cases and to deal with exceptional scenarios.

In principle, Tourbillon's payment process is easy to integrate into today's payment landscape.[36] It uses existing technology such as QR codes, builds on existing

[36] The payment process between consumer and merchant provides a simple user experience and is almost instantaneous in both prototypes. The merchant's app generates a QR code that transmits all relevant

infrastructure such as PoS systems and uses existing account relationships between consumers, merchants, banks and central banks. However, EC1 has a simpler design and is likely to be easier to implement than EC2, which requires banks and the central bank to operate the nodes of a mix network.

The Tourbillon prototype is a first step in understanding privacy, security and scalability of EC1 and EC2 for a limited set of use cases. Further work will be required, for example to extend use cases and the handling of exceptional scenarios, on the journey towards a potential rCBDC based on the eCash design.

4 Conclusions and next steps

Project Tourbillon explores three important features of CBDC systems that have proven challenging in current implementation studies and pilots: privacy, security and scalability. It has developed and tested two distinct CBDC prototypes based on eCash-designs that use cryptography, such as blind signature schemes and mixing. The aim was to provide:

i) strong *privacy* in the form of payer anonymity that protects the payer's personal payment data while countering illicit payments;

ii) strong *security* that is counterfeit-resistant both to current and potential future threats, including those from quantum computers; and

iii) adequate *scalability* validated against payment data.

Project Tourbillon is a first step in exploring privacy, security and scalability in CBDC designs. It successfully demonstrates the feasibility of the proposed design, provides new insights into the potential of eCash as a basis for future CBDC systems and, most importantly, highlights three areas where further work is needed.

First, the project highlighted the challenges of implementing and deploying QSC; mainly the reduction in transaction processing speed. Further research and experimentation are needed to improve functionality and efficiency, and to better understand how to safely transition from current cryptography to QSC.

Second, although Tourbillon demonstrates the feasibility of the eCash design, privacy, security or scalability may be improved as requirements or objectives

payment information to the consumer's app on the mobile device and initiates the payment immediately after authorisation by the consumer. Both the merchant and the consumer can check the outcome of the payment in the app. The entire process, from scanning the QR code to confirming the payment with the merchant, takes only one second, although the design using QSC takes longer due to higher computing and memory requirements.

change. Thus, modelling the trade-offs between privacy, security and scalability and the extent to which they impact each other is useful for advancing the prototypes.

Third, looking ahead, further work is needed to explore how an eCash-based design could be implemented. This includes considering additional use cases – such as offline payments – and exploring economic viability with a sustainable business model.

References

Amoroso, E (1994): *Fundamentals of computer security technology,* AT&T Bell Labs, Prentice-Hall: Upper Saddle River.

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 (2020): *Central bank digital currencies: foundational principles and core features*, October.

——— (2021): *Central bank digital currencies: system design and interoperability*, September.

Bank of England (BoE) (2023): *The digital pound: technology working paper*, February.

Beullens, W, V Lyubashevsky, N K Nguyen and G Seiler (2023): "Lattice-based blind signatures: short, efficient and round-optimal", *Cryptology ePrint Archive*, no 77.

Beullens, W and G Seiler (2022): "LaBRADOR: compact proofs for R1CS from module-SIS", *Cryptology ePrint Archive*, no 1341.

Bootle, J, V Lyubashevsky, N K Nguyen and A Sorniotti (2023): "A framework for practical anonymous credentials from lattices", *Cryptology ePrint Archive*, no 560.

Bos, J, L Ducas, E Kiltz, T Lepoint, V Lyubashevsky, J Schanck, P Schwabe, G Seiler and D Stehlé (2018): "CRYSTALS – Kyber: a CCA-secure module-lattice-based KEM".

Boyen, X, T Haines, J Müller (2020): "A verifiable and practical lattice-based decryption mix net with external auditing", in: L Chen, N Li, K Liang and S Schneider (eds), *Computer security – ESORICS 2020*, Springer.

Chaum, D (1981): "Untraceable electronic mail, return address, and digital pseudonyms", *Communications of the ACM*, vol 24, pp 84–88.

——— (1982): "Blind signatures for untraceable payments", in D Chaum, R Rivest and A Sherman (eds), *Advances in cryptology: proceedings of Crypto 82*, Springer-Verlag, pp 199–203.

Chaum, D, C Grothoff and T Moser (2021): "How to issue a central bank digital currency", *SNB Working Papers*, no 3.

Chaum, D and T Moser (2022): *eCash 2.0: inalienably private and quantum-resistant to counterfeiting*.

Chaum, D and M Yaksetig (2023): "cMixx^{+}", https://xxfoundation.org/wp-content/uploads/2023/11/cmixx.pdf.

Committee on National Security Systems (CNSS) (2015): *Glossary,* April.

Committee on Payment and Settlement Systems (CPSS) and International Organization of Securities Commissions (IOSCO) (2012): *Principles for financial market infrastructures*, April.

Coy, P (2022): "Does the End of cash mean the end of privacy?", *New York Times,* March.

Cunliffe J (2023): "Money and payments: a "black ships" moment?" remarks given at a conference titled "Economics of payments XII" at the Federal Reserve Board, Washington DC, 26 October.

de Montjoye, Y-A, L Radaelli, V Singh and A Pentland (2015): "Unique in the shopping mall: on the reidentifiability of credit card metadata", *Science,* vol 347, no 6221, pp 536–39.

Digital Currency Institute (DCI), "Project Hamilton: building a hypothetical central bank digital currency".

Ducas, L, E Kiltz, T Lepoint, V Lyubashevsky, P Schwabe, G Seiler and D Stehlé (2018): "CRYSTALS-Dilithium: a lattice-based digital signature scheme", *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol 1.

European Central Bank (ECB) (2021): *Eurosystem report on the public consultation on a digital euro*, April.

——— (2023a): *Digital euro: the next step in the advancement of our currency*.

——— (2023b): *Progress on the investigation phase of a digital euro – third report*.

Financial Action Task Force (FATF) (2023): *International standards on combating money laundering and the financing of terrorism and proliferation – the FATF recommendations*, February.

Fischlin, M (2006): "Round-optimal composable blind signatures in the common reference string model", in C Dwork (ed), *Advances in cryptology – CRYPTO 2006*, Springer.

Fouque, P, J Hoffstein, P Kirchner and V Lyubashevsky (2018): *Falcon: fast-fourier lattice-based compact signatures over NTRU*.

People's Bank of China (PBOC) (2021): *Working group on E-CNY research and development*.

Pfitzmann, B (1991): "Fail-stop signatures; principles and applications", in the proceedings of Compsec 91: the 8th world conference on computer security, audit and control.

Rivest, R, A Shamir and L Adleman (1978): "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM,* February.

Schumacher, B (1995): "Quantum coding", *Physical Review A,* April.

Van Hove, L (2001): "Optimal denominations for coins and bank notes: in defense of the principle of least effort," *Journal of Money, Credit, and Banking*, vol 33, no 4.

Annex A: Optimal denomination algorithm

Project Tourbillon allows a consumer to make withdrawals and payments of any amount, but it aims to facilitate these transactions with the lowest quantity of individual coins possible. The eCash design uses coins that are encrypted, recorded and destroyed individually. As such, efficiency in processing individual coins leads to better scalability. A binary numeral system for denominations (eg 1, 2, 4 and 8) is an optimal solution as explained by Van Hove (2001). However, maintaining an ideal set of denominations in a wallet requires an optimisation algorithm.

Withdrawal

This algorithm takes certain trivial cases as its starting point. These are the cases where one of each denomination would achieve the desired outcome: specifically, withdrawals of 1, 3, 7 or 15. Beginning from 1, adding one coin of the next highest denomination will net the next trivial case: 1 + 2 = 3 and 1 + 2 + 4 = 7 and so on. Henceforth, these trivial cases will be referred to as increments. These increments are the basis of an optimal account balance because different subsets of these increments can sum to any value less than or equal to the increment. The following is a description of the algorithm with accompanying pseudocode.

Now, say a user wants to withdraw a certain amount W. In phase 1, the algorithm determines the largest increment that is smaller than or equal to W. Henceforth, refer to this increment as D. Add the coins for increment D into the withdrawal basket. If W = D, the withdrawal is trivial for the reasons previously mentioned, and the algorithm is complete. Otherwise, there is an outstanding amount of the withdrawal that must be calculated, equal to W – D. Call this value S.

Phase 1

```
W = withdrawal amount
D = maximum increment <= W
Withdrawal basket = [empty list]
Add coins for increment D to withdrawal basket
If W = D:
  algorithm is complete. Terminate.
Else:
  S = W - D
```

Next, determine the largest coin denomination that is less than or equal to S. Call this denomination N. Allowing for a remainder, divide S by N, determining how many whole coins of denomination N are needed in phase 2 of the withdrawal. Add these coins to the withdrawal basket.

Phase 2

```
If S != 0:
```

```
  N = largest denomination <= S
  Remainder = S % N
  Number of coins of denomination N = (S – Remainder) / N
  Add coins to the withdrawal basket
Else:
  algorithm is complete. Terminate.
```

At this stage in the algorithm, it has determined the largest increment that is still smaller than W and it has used as few coins as possible to make up the difference between W and that increment. The only remaining question is: what happens if S is not perfectly divisible by denomination N? The answer is to repeat phase 2, but first take the value of the remainder and assign it to S. Repeat phase 2 until the value of S is 0. Then the algorithm is complete, at which point the contents of the withdrawal basket will sum to W using the fewest number of individual coins that still allow for payments of any value less than or equal to the withdrawal amount.

Payment

The payment algorithm is far simpler as a result of the guarantees provided by the withdrawal algorithm.

First, sort the contents of the wallet in descending order to prioritise using the fewest coins. Subtract the largest possible denomination from the amount to be paid. Repeat until sufficient funds have been allocated.

```
C = coins in descending order
P = payment amount
While P > 0 and C is not empty:
  P = P – (highest value in C <= P)
algorithm is complete. Terminate.
```

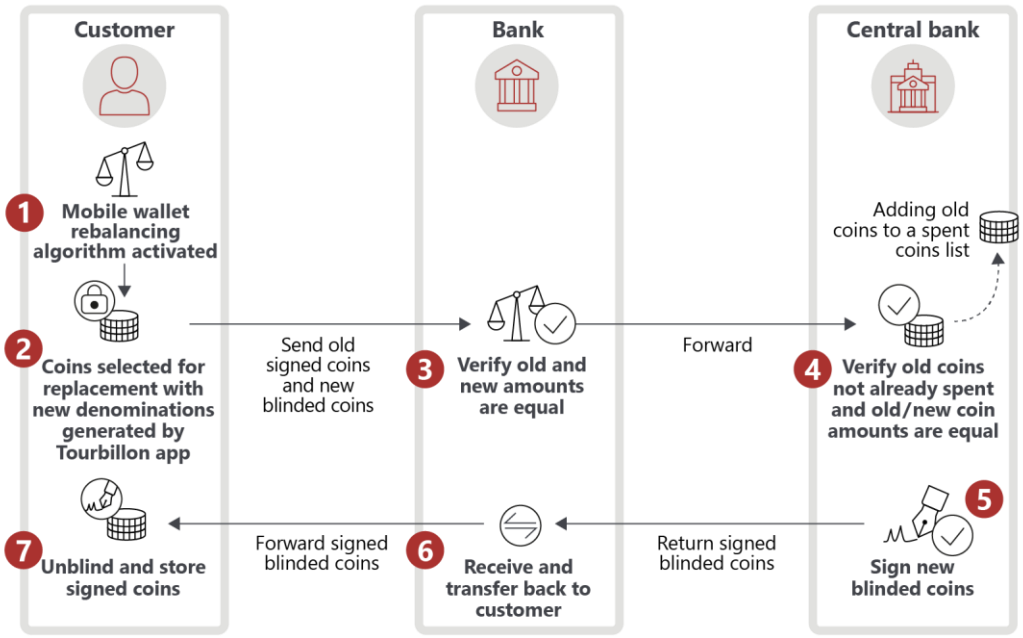
However, it may be the case that the payment results in an imbalance of denominations, where the current wallet balance is composed of coins that cannot be combined to sum any value required. To remedy this, a rebalancing algorithm is required before the next payment can be made.

Rebalancing

A rebalance is triggered if there are no coins of denomination 1, 2 or 4 remaining in the account, considering whether the balance is greater than or equal to the value of the missing denomination. First, remove a coin of the largest value present in the wallet and destroy it. Then initiate the withdrawal algorithm to replace that amount. The result will be an account balance composed of coins that can be combined to make payments of any amount.

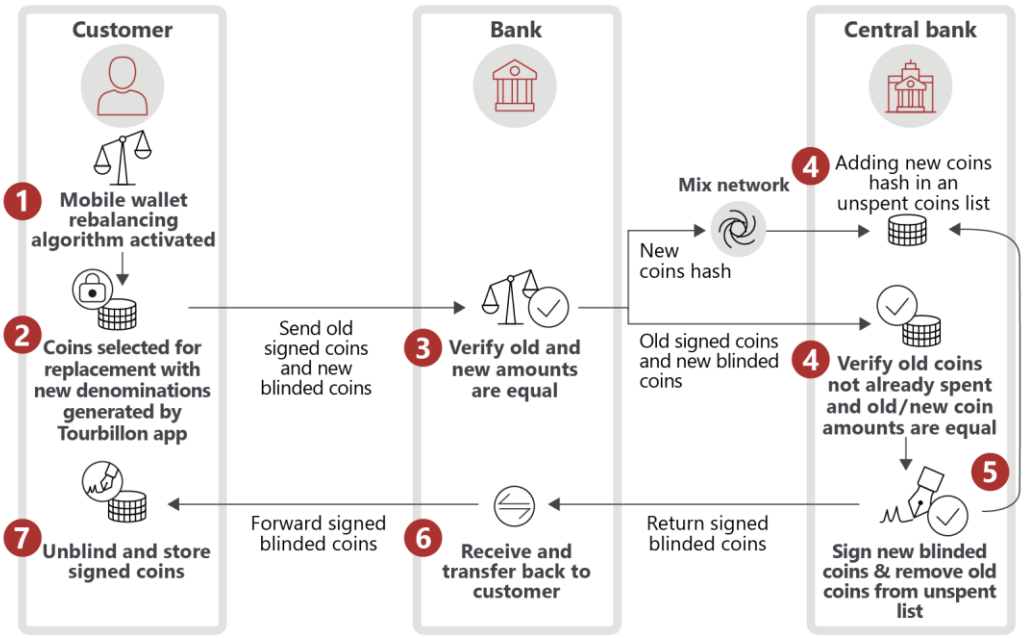
Annex B: Sequence diagrams for rebalancing

Rebalancing for eCash 1.0 (EC1) Graph B.1



The Tourbillon app uses a rebalancing algorithm to ensure that the customer can always pay any amount between 1 and their CBDC balance. The algorithm runs each time the customer's balance changes and calculates the smallest combination of coin denominations that the customer should hold.

Rebalancing for eCash 2.0 (EC2) Graph B.2



The Tourbillon app uses a rebalancing algorithm to ensure that the customer can always pay any amount between 1 and their CBDC balance. The algorithm runs each time the customer's balance changes and calculates the smallest combination of coin denominations that the customer should hold.

Annex C: Threat model and trust assumptions

Threat models and trust assumptions serve as pillars of a solid cyber-secure design, equipping professionals with the capacity to predict, comprehend and counteract potential threats proactively, whilst safeguarding the system's resilience and reliability (Amoroso 1994).

Trust assumptions embody both implicit and explicit views that specific components of a system are secure. These principles include the belief that these components will perform as anticipated under varying circumstances and that certain security protocols will provide an expected level of protection.

Threat modelling is a systematic method aimed at pinpointing, understanding, and methodically addressing potential threats to a system or an application. This approach empowers security experts to visualise a system's attack surface, list possible threats, evaluate associated risk levels and strategise appropriate proactive countermeasures.

System participants

The system participants identified in the context of the Tourbillon prototype are not all-encompassing, yet they represent the primary participants – those that are indispensable for functionality. A catalogue of system participants is found in Box C.1.

Box C.1: System participants

Central bank: The central bank is the sole issuer of CBDC. It maintains an account of bank reserves, CBDC balances for banks and changes to reserves or CBDC balances. The central bank is also responsible for validating CBDCs when payments are made. The central bank maintains a direct relationship with banks.

Bank(s): A bank facilitates the exchange of CBDC against the bank money of its customers and vice versa. The bank is responsible for maintaining its own ledger of customer balances and changes to those balances. The bank maintains a direct relationship with end users.

Payer(s): A user that holds CBDC and is exchanging their CBDC for goods or services.

Payee(s): The user receiving CBDC as payment in exchange for goods or services.

Threat and trust assumptions

The threat and trust assumptions (explored in Table C.1) are not mutually exhaustive, yet they capture a substantial series of circumstances that may result in systemic susceptibilities.

Threat and trust assumptions	Table C.1

Central bank

(1) The central bank is trusted to issue valid CBDCs that will be accepted upon payment.

(2) The central bank is trusted to reject invalid CBDC deposits.

(3) The central bank will monitor the amount of CBDC in circulation (issuance and redemption).

(4) The central bank is trusted to always correctly debit and credit bank reserves and balances.

Bank(s)

(1) Banks are trusted to debit and credit the accounts of their clients correctly.

(2) Banks may collude with each other for their benefit (to falsely increase / decrease reserves) by tricking the central bank to:

   a. accept coins already spent;

   b. create coins the value of which is larger than the value being debited from their reserves; and

   c. credit to their reserves a value that is larger than the original value of the deposited coins.

Payer(s)

(1) Payers are not trusted to perform withdrawals or payments correctly.

(2) Payers will attempt to pay using already spent or counterfeited coins.

(3) A payer may attempt to trick the commercial or central bank to receive more CBDCs than was requested (debited from their account).

Payee(s)

(1) Payees are not trusted to perform deposits or payments correctly.

(2) Payees will attempt to deposit more CBDC than was intended.

(3) Payees will attempt to deposit already spent coins.

(4) A payee will attempt to trick payers by claiming that successful payments failed.

Colluding parties

(1) Banks, payers and payees may collude with each other to achieve malicious goals.

(2) The central bank, commercial banks, payers and payees will attempt to use any data gained from system processes (ie withdrawals, payments and deposits) to deduce private information like personal transaction patterns.

(3) There is at least one honest mix network node.

Security requirements

Table C.2 lists the security requirements of Tourbillon; the conditions which must be met for the system to be secure.

Tourbillon security conditions Table C.2

Security requirements

(1) *Counterfeiting*: The central bank must be the sole issuer of CBDC; the CBDC system must be counterfeit-proof and capable of detection.

(2) *Double spending*: A CBDC can only be spent once (directly deposited upon receipt).

(3) *Balances*: The total value of central bank money is preserved after each withdrawal, payment and deposit.

(4) *Privacy*: Participants should not learn any information about payment amounts or the identities of parties, even via collusion.

(5) *Unlinkability*: Payments by a user must not be linked to any of the withdrawals made by the same user, nor to any other possible payments, even via collusion from participants.

(6) *Quantum resistance:* Payer privacy and resistance to counterfeiting must hold up against possible future quantum computer attacks, including "harvest now and decrypt later" attacks.

Threats and limitations

While both designs (EC1 and EC2) share components, they diverge primarily in the recording of issued and redeemed CBDCs at the central bank. In EC1, CBDCs only appear on the list of spent coins once they are redeemed. Conversely, in EC2, an unspent coins list is used to keep track of the issued coins. These differences have implications for the CBDC design and threat models.

EC1

EC1 provides unconditional privacy but has no transparency on the issuance of CBDCs. Specifically, only the central bank knows how many coins exist; there is no third party (or public) mechanism to check which coins, or how many, are in circulation. As a result, merchants can manipulate consumers by denying receipt of coins upon valid payment (for example, by claiming bad connectivity). A consumer would find it challenging to dispute such a claim without a mechanism to prove CBDC ownership and a link to the spent coins list.

Critically, in the presence of a quantum computer, it would be impossible to distinguish central bank issued CBDCs from counterfeit ones. Given the existence of the spent coins list, a valid coin must simply not appear on a list. To put it another way, there would be no mechanism to document which CBDCs have been issued by the central bank and are ready to be spent.

EC2

EC2 is transparent on the issuance of new coins, but no longer provides unconditional privacy. With the introduction of the unspent coins list, transparency is increased, yet

posting to the list presents potential vulnerabilities to unlinkability. Specifically, how does a consumer update the unspent money list without disclosing their identity? The mix network mitigates this risk.

Quantum computers may break unlinkability and privacy. A mix network alleviates the unlinkability condition, but anonymity is proportional to the size of mix network messages. In the presence of quantum computers, anyone can reduce the anonymity size to one. Using quantum secure end-to-end encryption (E2EE), or a quantum secure mix, minimises susceptibility to quantum computer attacks.

The central bank and commercial bank may maliciously collude to reveal a consumer payment, thereby breaking unlinkability. A bank knows who submitted withdrawals and has some information about the withdrawal (ie denominations and the time of day). In a malicious collusion with the central bank, this information may be used to learn if the CBDC has been spent and when – linking identity to payment.

Additional threats

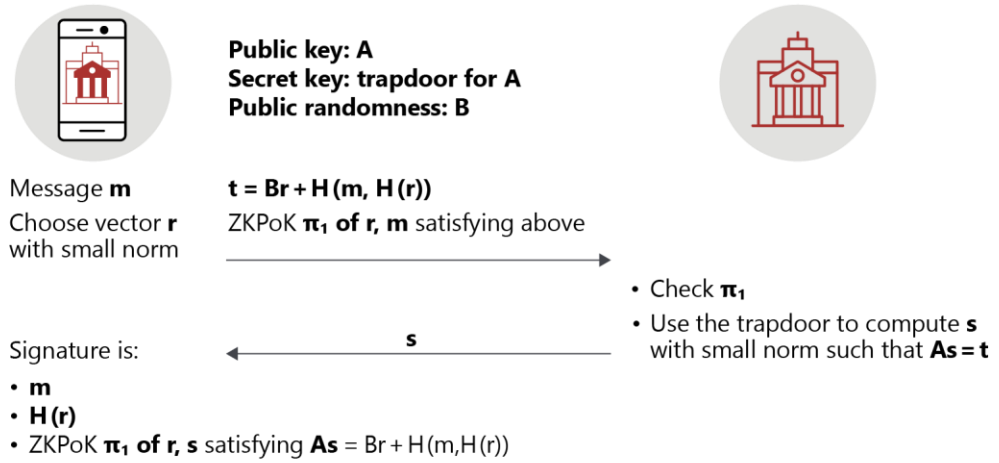
Linkability concerns are also found via different permutations of colluding parties. Notably, commercial banks and merchants can collude to identify consumers and their habits. For example, a supermarket and commercial bank can collude to reveal consumer identities. A commercial bank can identify customers who (i) live in the area of the supermarket and (ii) have performed withdrawals of CBDC that meet or exceed the value of a payment to the supermarket (via collusion with the supermarket). The result of this may reveal a consumer's withdrawal, payment and ultimately their habit.

Annex D: Quantum-safe blind signature scheme

The quantum-safe blind signature scheme is based on the hardness of problems over module lattices. The hardness of these problems underlies the recently selected NIST and CNSA 2.0 standards for quantum-safe encryption (Bos et al (2018)) and digital signature schemes (Ducas et al (2018), Fouque et al (2018)). For added efficiency, an additional assumption introduced in Bootle et al (2023) may also be used. The blind signature schemes are based on the general framework of Fischlin (2006), using the signature scheme of Fouque et al (2018) and instantiated using the recent designs of Bootle et al (2023) and Beullens et al (2023).

The high-level idea in the construction of Beullens et al (2023) is that the user sends a commitment to a message *m* that the user wants the signer to sign and gives a zero-knowledge proof that the commitment is of the right form (Graph D.1).

Quantum-safe blind signature scheme — Graph D.1



The signer then creates a hash-and-sign type signature of the commitment, possibly after adding some randomness – using Fouque et al (2018) – and sends it to the user. The user's signature of *m* then contains *m* and a zero-knowledge proof of knowledge of the commitment and the signature.

The main tool in these constructions are the zero-knowledge proof systems in the first round of the signing process and the user's signature. One could use the more general scheme of Beullens and Seiler (2022), which would then instantiate the signature as in Beulens et al (2023). It would have its security based on the same problems that underlie the NIST standards.

Contributors

Steering Committee	Thomas Moser, SNB, Alternate Member of the Governing Board
	Morten Bech, BIS Innovation Hub, Swiss Centre Head
Project Team[37]	Mike Alonso, BIS Innovation Hub, Adviser
	Olaf Keller, SNB, Deputy Chief Information Security Officer
	Robert Oleschak, BIS Innovation Hub, Adviser
Technical Advisers	David Chaum, Cryptographer and inventor of eCash
	Mario Yaksetig, Cryptographer

Contributors from external vendors or third parties

IBM Research Lab	Elli Androulaki
	Ilie Circiumaru
	Angelo De Caro
	Kaoutar El Khiyaoui
	Vadim Lyubashevsky
	Gregor Seiler
	Patrick Steuer
IBM Consulting	Antoine Arnould
	Clément Berti
	Labib Farag
	Joris Huynh
	Saoussen Marouani
IBM Technology	Fabio Keller
Currency Network	Animesh Ghosh
	Banasree Ghosh
	Rajib Ghosh
	Sibabrata Banerjee
Expert review	Jean-Philippe Aumasson, Taurus Group
	Srdjan Čapkun, ETH Zurich

[37] Special thanks to Andreas Adriano, Jiwoo Bae, Rudolf Biczok, Emma Claggett, Josiah Friesen, Martin Hood, Nataliia Luchyn, Lilli Menti, Christoph Meyer, Darko Micic, Nicole Neumann, Katie Ranger, Esther Rey Losada, Nanette Roth, Maximilian Schrader, Sigrid Sulcebe, Marino Vollenweider, Violeta Vuletic, Fabio Wieser and many others for their valuable administrative, communication, IT, legal and statistical support.

Bank for International Settlements (BIS)

ISBN 978-92-9259-711-5 (online)