SUERF
The European Money and Finance Forum

SUERF Policy Brief
No 622, June 2023

Practical Offline Payments Using One-Time Passcodes*

By Priscilla Huang (EFREI Paris), Emmanuel Benoist (Bern University of Applied Sciences), Christian Grothoff (Bern University of Applied Sciences, Taler Systems), and Sebastian Javier Marchano (Taler Systems)

Keywords: offline, digital payment, usability, retail CBDC
JEL codes: E42

Modern buyers enjoy the convenience of digital payments, but not all points of sale always have an Internet connection. Trusting the buyer's device to honestly report that a payment was definitively made puts merchant's revenue at risk. We present an inexpensive and usable solution for merchants to verify that a buyer correctly completed a payment even when the point of sale is offline.

*Authors' acknowledgements: We thank Stefan Kügel for doing research on prior art and writing the patent application LU103081 based on the presented design. We thank NGI ZERO ASSURE for funding part of the work on GNU Taler for regional currencies.

1. Introduction

While more and more humans are online, some points of sale (PoS) still have no Internet connectivity. Furthermore, even PoS that generally have Internet access may experience network outages from time to time. Consequently, offline payments are frequently cited by central banks as a requirement for central bank digital currencies [11, 2, 5, 3] and various commercial digital payment systems [13, 9, 12] offer modes in which offline payments are possible.

A recent report on offline payments by the Bank of International Settlements [1] cites financial inclusion, resilience and cash resemblance as primary reasons why central banks desire offline support in central bank digital currencies (CBDCs). Our work addresses these concerns, as financial inclusion is also always a question of cost, we improve resilience as we handle situations where the PoS is offline, and by integrating our solution with GNU Taler [4] we preserve cash-like privacy for buyers.

We consider a *point of sale* (PoS) to be any physical location where a *merchant* operation is using PoS *machines* operated by *sellers* to distribute goods for money to *buyers*.

The CAP theorem [6] puts a fundamental limit on the security of digital offline payments when neither buyer nor seller can communicate with the database that ensures consistency for the payment data: a payment system cannot be available and committing transactions while the network is partitioned and also guarantee consistency at all times. In this article, we focus on a practially relevant corner case where the seller is offline but the buyer has Internet connectivity.

If only the seller were online, the buyer either doesn't have a digital device or could generally use the seller's Internet connection. Similarly, when the seller is offline it could use the buyer's Internet connection. However, practically speaking, contemporary devices do not make it easy for the buyer to quickly bring the seller online, and moreover, the offline seller may not even have a smart phone.

In this article we present a practical system which allows an offline PoS to validate a transaction made by an online buyer, and compare this new approach with previous designs. Our design assures the seller that a payment was made with high probability and without requiring the PoS to have a general purpose computer, phone or access to SMS. The proposed solution is deliberately *simple*: such solutions have a better chance of providing security in practice when widely deployed to non-expert users.

2. Background

In traditional commerce, many small merchants have purely offline systems to limit costs. With physical cash, this is not a problem. Buyers pay directly and the seller can collect their money physically. Consequently, physical cash is generally the most inclusive offline payment option.

In terms of digital payments, the limitations imposed by the CAP theorem [6] are generally resolved by lowering the security assurances provided by the system. While this seems inherent when both seller and buyer are offline, this even happens in cases where the CAP theorem does not strictly apply, such as the Twint system.

2.1 Twint-style Digital Offline Payments

For sellers who use Twint [14] (a payment system used in Switzerland), they can also accept offline payments. [13] For offline payments, the system generates a static QR-code which is printed and put on display for buyers. It contains the bank details of the seller with or without the price to pay. To verify that the buyer has paid, the seller checks the screen shown on the buyer's device. If it is showing a certain green dialog, the payment has been made. The security of this system is imperfect, as an attacker only has to simulate scanning a QR code to then display the validation dialog to trick the seller into believing that the transaction completed. The seller has no offline way of verifying whether the transaction has actually taken place.

Twint's design is not an exception: PayPal offers essentially the same design with its static QR code payments [9]. Again, merchants are expected to display a static QR code, have the buyer enter the amount into their PayPal application, and believe the buyer's device that the payment was actually made. We have anecdotal reports of merchants being shown false confirmations with both Twint and PayPal. WeChat also supports off-line payments [12]. Again, buyers scan a QR-code presented by the merchant. The buyer is connected to a page, where they can enter the price to pay. Then a confirmation appears on their phone. The confirmation contains the price plus an identifier specific for each merchant (usually the name of the merchant and a logo). However, this identifier can easily be faked. WeChat recommends therefore: "*However, with no device or frontend system at all, the merchants will not receive notifications after users make payments. Institutions are thus advised to develop some result notification channels for merchants, such as an Official Account message template or an app dedicated for result notification*." [12].

2.2 Public-Key Cryptography to the Rescue?

The obvious solution to providing an offline merchant with a secure payment confirmation via an untrusted buyer's device involves the use of digital signatures. Here, the payment system could simply generate a cryptographically signed message for the buyer to show to a PoS machine. The PoS machine would have been configured with the public key of the payment system and would need a computer to verify the cryptographic signature. Furthermore, the digital signature would have to be transmitted from the buyer's device to the merchant's computer. The probably most likely method for this in practice would be to scan a QR code displayed by the buyer's device.[1]

The main drawbacks of this method are thus the need for a moderately powerful computer by the merchant that can communicate with the buyer's device and run public key cryptography, and the need to exchange a non-trivial binary message. This work seeks to provide an alternative using only simple symmetric cryptography which also does not need any communication hardware (no camera, no digital transmission).

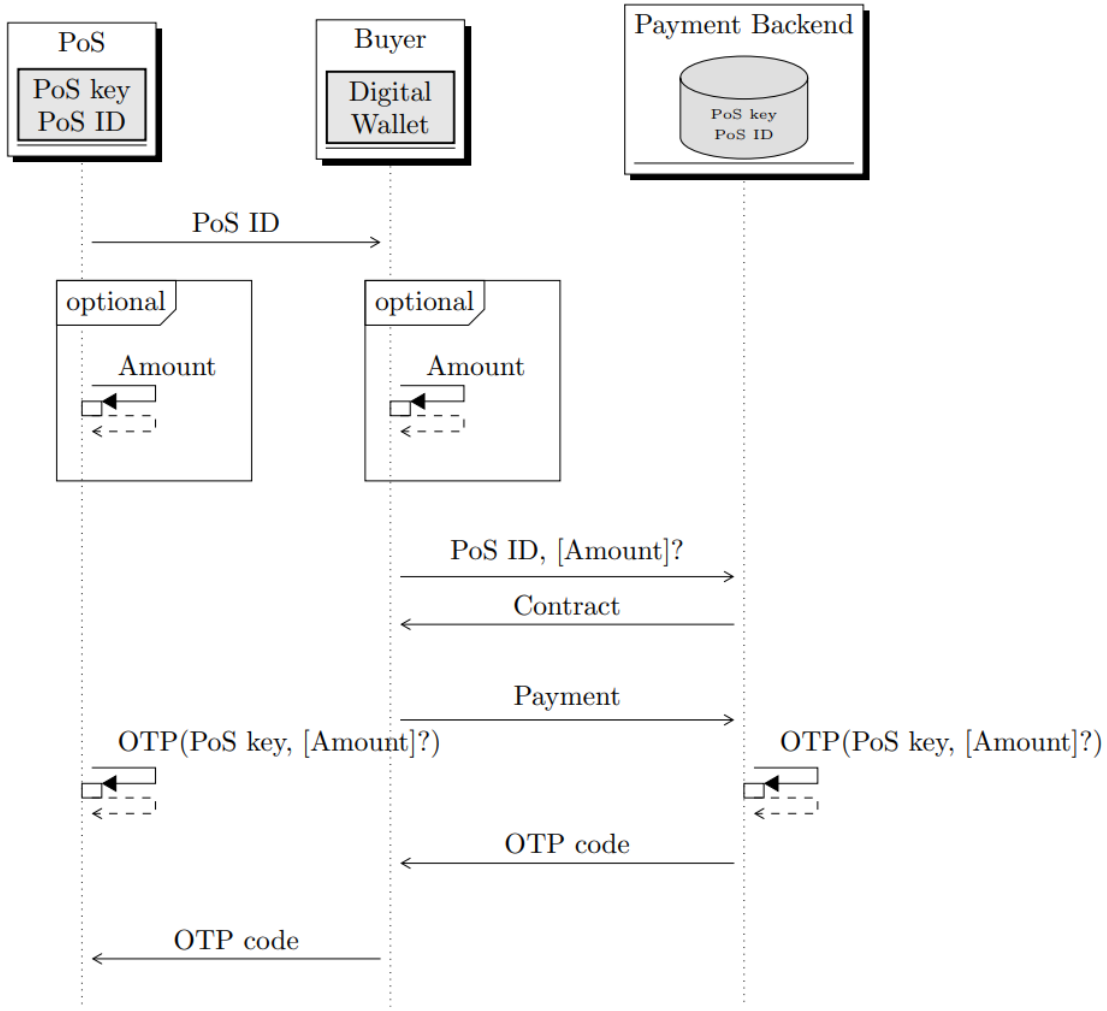
2.3 Time-based One-Time Passwords

Time-based one-time passwords (TOTP) [8] allow users to authenticate to an Internet service using a shared symmetric key and the current time. One-time passwords are core to our solution as they inherently prevent replay attacks due to their ephemeral nature. With TOTP, both the user and the server generate a unique 4 to 8 digit passcode by hashing the symmetric key and the current time rounded to a multiple of a key rotation frequency, typically 30s. Additionally, the server typically compares the client's to multiple codes generated using timestamps around the current server time to compensate for clock-skew, network delays and user data input delays.

[1] NFC may also be an option, but it is less certain to be available on the buyer's device.

On the client-side, TOTP codes can be generated using TOTP applications or specialized TOTP devices. If two systems share the same configuration in formation (key, hash function and number of digits), they will provide exactly the same number at the same time without having to be connected.

Figure 1



Notes: The PoS is offline while the buyer is online and can talk to the payment backend over the network. The seller only has an **offline** device merely capable of computing an OTP based on a *PoS key* shared apriori with its payment backend. First, the buyer obtains the PoS ID (for example, by scanning a QR code). If there is more than one price per PoS and the amount paid is to be verified, both merchant and buyer enter the amount into their device. The buyer pays and in return receives the *OTP code* returned by the payment backend. The buyer shows the OTP code to the merchant, who can *visually* compare the payment confirmation OTP code of the buyer with the computation by its own offline device.

3. Our Approach

We propose a payment-system independent method that allows an offline vendor to verify that a payment has been made by a buyer to their wallet. We assume here that the buyer has a digital payment application on their device (typically a smartphone) and that the buyer's device has access to the Internet. Using the Internet, the digital payment application must be able to communicate with the payment service provider and a trusted online service run by or on behalf of the seller.

The main idea behind this paper (Figure 1) is to invert the authentication mechanism provided by TOTP, confirming to a human that they are talking to the right server — with the implied message being that the merchant received the expected payment.

As usual, the TOTP protocol uses a secret shared between the PoS's device and the trusted server of the seller. When a buyer successfully completes an online digital payment, the seller's trusted online service returns a set of TOTP codes based on the time of the purchase and the secret shared with the PoS to the buyer. The seller can easily create the same TOTP code(s) using a cheap device that is not connected to the Internet, and can verify that their code(s) match those that the buyer presents to the seller from their smartphone. As usual with TOTP, the server should typically return multiple codes to compensate for minor disagreements about the current time. This also addresses the scenario where a merchant is slow to generate their TOTP code, as then they may simply find the code at a slightly different position in the list of codes shown by the buyer. The exact number of codes being generated and the width of the time interval for each code can be adjusted to the needs of the merchant. Our implementation defaults to 5 codes with 30 second intervals. The specific choices made here must be conservative, as once the time interval has lapsed the merchant must go online to verify that the buyer made the payment.

A cheating buyer has no good way to predict the TOTP code without actually performing a payment. While the TOTP security level at 4–8 digits is at best 10^{-8}, this should be more than enough to deter a buyer from attempting to deceive a merchant in person.

The system described so far can only attest that a transaction has taken place, but does not provide any a assurance that this transaction corresponded to the desired amount. A forged application on a smartphone could display a transaction validation with the current TOTP code obtained by paying a tiny amount instead of the actual price.

A small extension of the TOTP protocol can be used to make this attack impossible: We simply need to include the price when computing the hash. Then, a seller can simply type the price into their device to generate a price-dependent code. A modified TOTP algorithm on the seller's trusted server would equally include the price paid by the buyer, allowing the seller to verify that the buyer paid the correct price at the current time. Our approach thus has the huge advantage of not requiring any modifications to existing PoS systems, as a stand-alone TOTP device can be used by the seller when interacting with buyers making digital payments.

Table 1: Comparison of the various ways of securing digital payments for merchants without Internet.

	Security	Convenience	HW cost
Twint	- -	+++	0
Signature	+++	-	≈ $100
TOTP	+	+	≈ $10
Ext. TOTP	++	+	≈ $15

4. Discussion

Digital payments at a PoS where the buyer is online but the PoS is offline are an important use-case. Using TOTP is a cheap solution to secure digital payments against buyers faking payments. Table 1 summarizes the advantages and disadvantages of the various approaches for semi-offline digital payments.

We consider vanilla TOTP and SMS to be about equally secure: SMS messages can be intercepted and faked [7], and with vanilla TOTP the amount paid would not be validated. Thus, in both case a determined attacker would succeed in defrauding a merchant, making neither solution viable for large amounts. Digital signatures offer a much higher security level even compared to the extended TOTP protocol, as we would expect to stick to 8 digits to preserve the convenience when checking that the TOTP codes match.

In terms of convenience, Twint's method of checking a green screen shown by the buyer is simpler than any of the other methods. SMS is generally also pretty easy, as it does not even require the merchant to interact with the buyer. However, SMS can also be difficult as phone networks cannot always guarantee timely delivery. In terms of setup convenience, both the SMS and TOTP solutions require sharing information between the PoS equipment and the payment backend: for SMS the phone number must be configured, while for TOTP the PoS key must be entered into the payment backend. Thus, we would rank SMS about equal with the TOTP approaches in terms of convenience.

In contrast, merchants scanning a QR code shown by a buyer's device for signature validation is likely to be the most cumbersome interaction of all presented approaches.

5. Conclusion

Digital Inverted TOTP authentication can be a practical method for buyers to prove that they made a payment to a cost-sensitive offline merchant. Using TOTP we can establish a secure channel between the payment backend and the seller operating an offline PoS. Our solution around the limitations of the CAP theorem [6] is thus to securely heal the network partition by establishing cryptographically secured communication via an untrusted buyer's device. This allows the payment system to assure consistency despite the merchant not having a connected endpoint under their control.

In practice, it may suffice for sellers to only selectively check the TOTP code to still create an effective deterrent against buyers using fake payment applications. Even merchants that are generally operating online could use our design as part of an inexpensive contingency plan to continue secure operations during network outages.

References

[1] Anonymous. Project polaris: A handbook for offline payments with cbdc. Technical report, Bank of International Settlements, May 2023.

[2] R. Auer and R. Böhme. The technology of retail central bank digital currency. https://www.bis.org/publ/qtrpdf/r_qt2003j.pdf, March 2020.

[3] U. Bindseil, F. Panetta, and I. Terol. Central bank digital currency: functional scope, pricing and controls. https://www.ecb.europa.eu/pub/ pdf/scpops/ecb.op286~9d472374ea.en.pdf, December 2021.

[4] F. Dold. *The GNU Taler system: practical and provably secure electronic payments*. PhD thesis, Universit´e Rennes 1, 2019.

[5] ECB. Report on a digital euro, October 2020.

[6] S. Gilbert and N. Lynch. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. *SIGACT News*, 33(2):51–59, jun 2002.

[7] R. P. Jover. Security analysis of sms as a second factor of authentication, 2020.

[8] D. M'Raihi, S. Machani, M. Pei, and J. Rydell. Totp: Time-based one-time password algorithm. Technical report, IETF, 2011.

[9] PayPal Inc. Pay in a flash with QR. https://www.paypal.com/ us/digital-wallet/ways-to-pay/pay-with-qr-code, 2023. Accessed: March 13th, 2023.

[10] RedHat. FreeOTP. https://freeotp.github.io/, 2023.

[11] S. Schönborn and C. Rudelt. BDI: Digitaler Euro. Industriebedarfe bei Etablierung nicht vernachlässigen. https://bdi.eu/publikation/news/ digitaler-euro-innovation-digitalisierung-waehrung/, 2022.

[12] Tenpay. Offline Store without POS Devices. https://pay.weixin.qq. com/wiki/doc/api/wxpay/en/guide/OfflineStoresWithoutPOS.shtml, 2023. Accessed: March 14th, 2023.

[13] Twint AG. Collect payments simply – even without any infrastructure. https://www.twint.ch/en/bausiness-customers/our-solutions/ qr-code-sticker/, 2023. Accessed: March 3rd, 2023.

[14] Twint AG. Simply TWINT it. https://www.twint.ch/en/, 2023. Accessed: March 3rd, 2023.

About the authors

Priscilla Huang *is a postgraduate student at EFREI Paris, she worked as an intern at the Bern University of Applied Sciences for the project "Deploying GNU Taler for a Regional Currency in Basel" where she developed solutions for using Taler based on requirements of merchants from Basel.*

Emmanuel Benoist *is professor for computer science at the Bern University of Applied Sciences, his research is aimed at privacy protection for the Internet. The two main areas of research are e-health and Darknet markets since for both topics, privacy protection is a central asset.*

Christian Grothoff *is professor for computer network security at the Bern University of Applied Sciences, researching future Internet architectures. His research interests include compilers, programming languages, software engineering, networking, security and privacy.*

Sebastian Javier Marchano *is core developer of GNU Taler from Argentina. Since 2005 he covered roles such as software engineer, team management and chief technology officer.*