STARDUST

ISO8583 Blockchain Integration Deep Dive

Adit Patel (adit.patel@stardustfunds.com)

Seamless integration with traditional banking infrastructure, especially payment card transactions, would give distributed ledger users unparalleled versatility and control over their assets.

Several networks have attempted to adopt traditional banking data standards specifically ISO20022 as part of their strategies.

Stellar

IOTA

Hedera™

XinFin (XDC) Network

ripple

Algorand™

However, these networks all share a few key characteristics.

Centralized:

While the amount of centralization varies by project, all of these projects fundamentally differ from the fully decentralized models of traditional blockchains. Centralization ranges from networks relying on a few permissioned nodes to network that rely on a single, centrally coordinating node, like IOTA.

Permissioned:

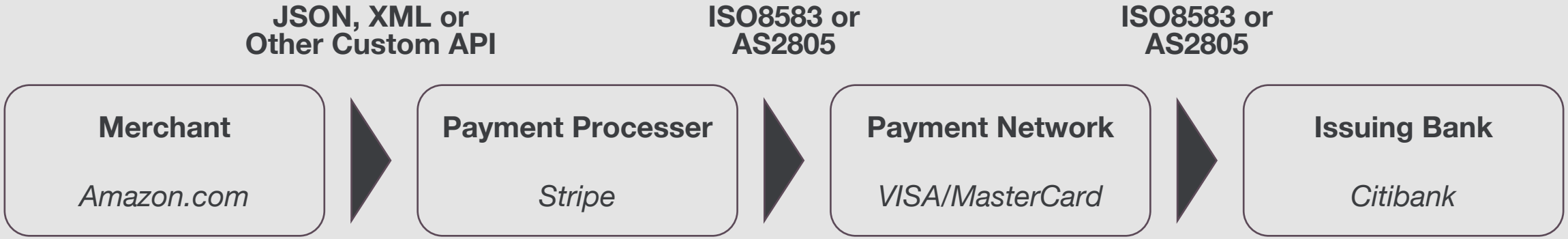
Even for networks that operate multiple nodes, individual user cannot simply host these nodes without first receiving permission from the parent organization.

Large Corporate/Government Backing:

Nearly all of these projects have a large incumbent backer. Whether that is networks like Stellar, which is supported by IBM and Deloitte, or Algorand which appears to have close ties to the Chinese Communist Party and had direct integration with China's state-backed Blockchain-based Service Network (BSN).

This report will focus specifically on integrating payment card transactions and their associated data messaging standard, ISO8583 and the closely related Australian Standard AS2805*.

ISO8583 messages are used to streamline and standardize information during a payment card transaction.



Merchant to Payment Processor

Any language can be used to accept transactions on the merchant site and communicate them to the payment processors. Today Stripe has client libraries in Ruby, Python, Java, Node, and many others. In addition, they also support JSON API calls over the internet or even mobile applications. Data standards at this rate are left up to the engineering team at the merchant.

Payment Processor To Network

As the user 'swipes' their card or enters transaction details, the payment processer converts the transaction request at the point of sale to an ISO8583 formatted message, or AS2805 format in Australia. We'll detail what this looks like on the next slide.

Intra-Financial Markets

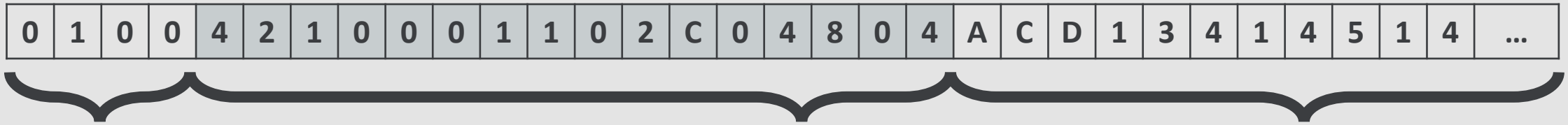
This ISO8583 is then sent through secure connections directly to the issuing bank who runs it through fraud models and either approves the transaction or declines it. This approval/decline message itself is return coded in ISO8583.

*AS2805 from a messaging perspective is nearly identical to ISO8583 and should be thought of as interchangeable for the purposes of this architecture.

The format of an ISO8583 message is relatively straightforward, and consist of a few key parts.

The MTI identifies the key information, the Bitmap specifies the metadata, and the Data Elements encode the actual data.

Example Message



Message Type Identifier (MTI)

This encodes who is sending the message, why, what they would like to do, and what standard this message conforms to.

Primary Bitmap

This is a 64 bit map indicating which fields have been provided in this message. These first 64 bits here have been encoded into a 16 character long hexadecimal string. These first bitmap is called the primary bitmap and is mandatory. Up to a total of 3 bitmaps can be contained in a single message though more than 2 is rare.

Data Elements

This is the actual payload associated with the metadata listed in the bitmap. It conforms to fields of fixed widths so the data can be verified for accuracy and that it was transmitted error free. This data can be decoded to match all of the data fields provided by the primary bit map.

The MTI arguably specifies the most important information, and makes it easy to identify the key classification of a transaction at a glance.

Each digit of the MTI indicates a vital piece of information about the message, from type to origination.

Example Message

0	1	0	0	…

ISO8583:1987 Coded Authorization Request by Acquirer

First Digit - Standard

This is the standard of ISO8583 this message conforms to. There are 3 standards, 1987, 1993, 2003.

- **0xxx – ISO8583:1987 (most common)**
- **1xxx – ISO8583:1993**
- **2xxx – ISO8583:2003 (rare)**

Second Digit - Class

At a high level this indicates what type of message it is, from approvals to reversals.

- **x0xx – Authorization**
- **x1xx – Authorization**
- **x2xx – Financial**
- **x3xx – File Action**
- **x4xx – Reversal**
- **x5xx – Reconciliation**
- **x6xx – Administrative**
- **x7xx – Fee Collection**
- **x8xx – Network Management**

Third Digit – Request Type

Indicates if the message is a response or needs a response or acknowledgement.

- **xx0x – Request**
- **xx1x – Request Response**
- **xx2x – Advice**
- **xx3x – Advice Response**
- **xx4x – Notification**
- **xx5x – Notification Acknowledgement**
- **xx6x – Instruction (2003 only)**
- **xx7x – Instruction Acknowledgement**
- **xx8x – Positive Acknowledgement**
- **xx8x – Negative Acknowledgement**

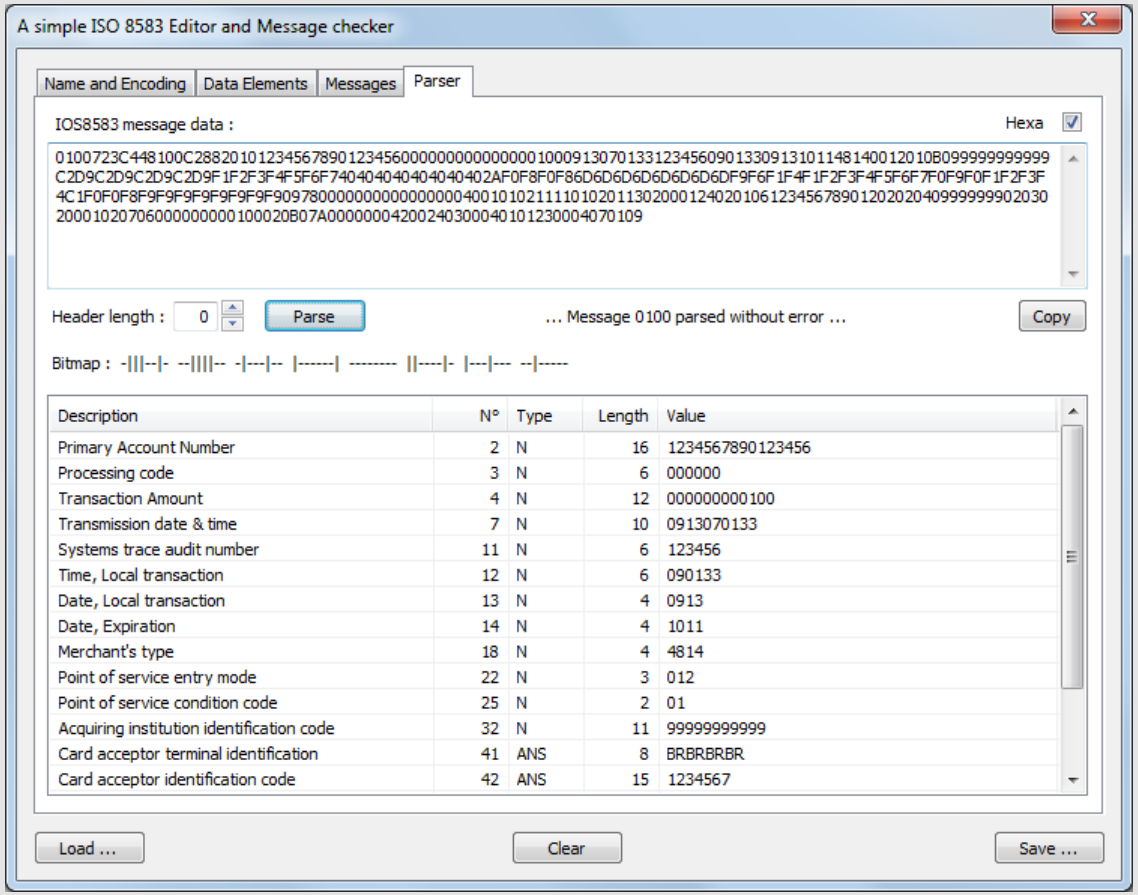
Fourth Digit - Source

Indicates who originated the message.

- **xxx0 – Acquirer**
- **xxx1 – Acquirer Repeat**
- **xxx2 – Issuer**
- **xxx3 – Issuer Repeat**
- **xxx4 – Other**

The primary bitmap and data elements are standardized, to allow anyone to decode the message contents easily as well as validate the message has been received in its entirety.

Standardizing this data allows anyone on the network to identify the type of transaction and easily decode the metadata fields and retrieve the data.



From this ISO8583 message, a bank can quickly identify the important information.

Identity:

The identification codes and primary account number allow a bank to quickly identify which cardholder swiped their card and which merchant account needs to be paid.

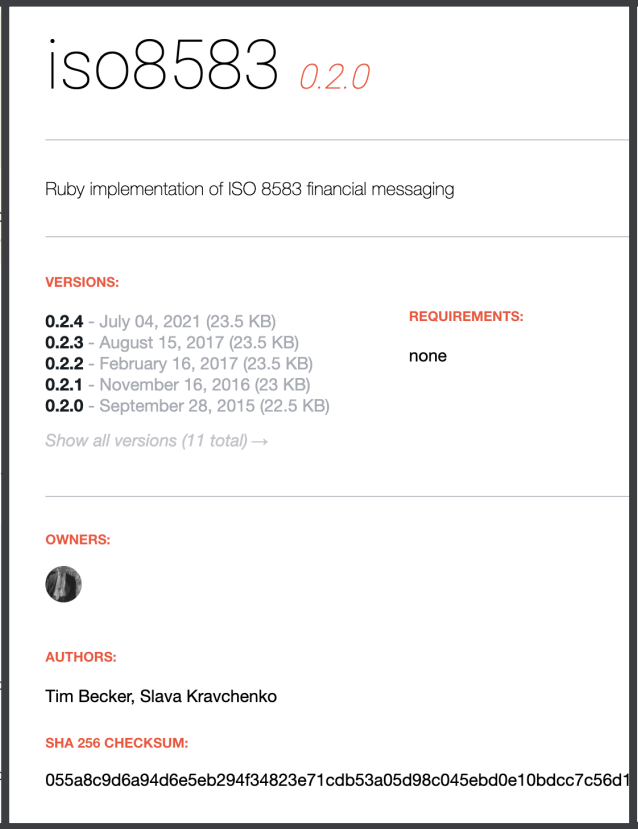
Transaction Details:

Under “Transaction Amount” we can see that this transaction was for $1 as it is denominated in pennies, the quanta unit of the specified currency. Fields #49 and #52 would provide us the ISO4217 currency code.

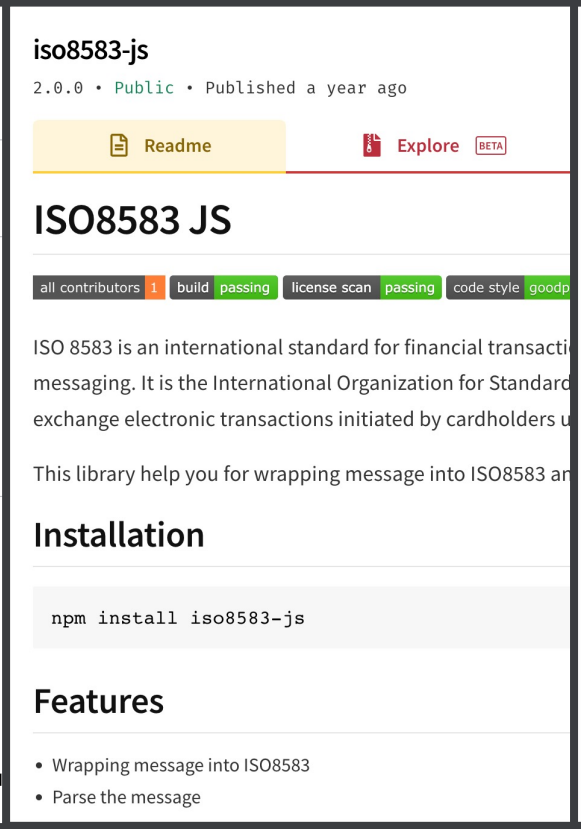
Several packages exist to help streamline the encoding and decoding of IS8583 messages. Seemingly making it easy to integrate and transmit ISO8583 transactions with Substrate.

These tools are open-source and production ready. They are capable of seamlessly ingesting ISO8583 data streams and decoding them into native, accessible objects in nearly any language including Rust.

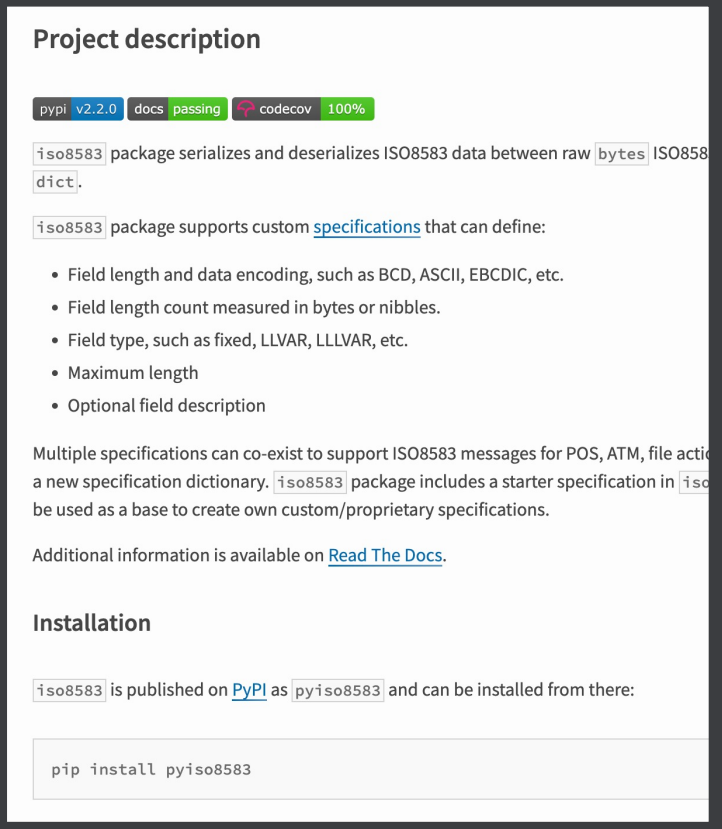
Rust



Ruby



Javascript



Python

Despite the ubiquity of libraries, market dominance of payment cards, and future-proofed entrenched position; no blockchain network has attempted to support ISO8583 transactions.

Several networks have attempted to adopt the ISO20022 as part of their strategies.

Stellar

IOTA

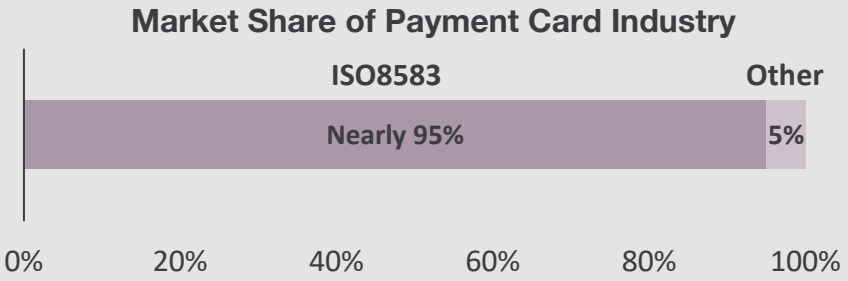
Hedera™

XinFin (XDC) Network

ripple

Algorand™

But despite its ubiquity and utility, none have attempted to adapt ISO8583.



Universal Ubiquity

ISO8583 is the dominant messaging format for retail and payment cards, one of the largest segments of the financial industry and will be for the foreseeable future. Despite this and the large number of blockchains attempting to integrate 20022, no blockchain has attempted to integrate ISO8583.

Entrenched Standard

Even though ISO20022 can support payment cards, it is very unlikely the industry will adopt the new standard. The rich metadata of ISO20022 has no relevance for payment cards, and the current business case indicates the net effect of migration would result in a $1B loss to the industry.*

*https://www.apb.it/wp-content/uploads/2016/09/iso-8583-vs-20022-cards-mandatory-migration-required-yet-anzellotti.pdf

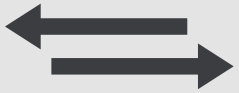
This is likely because data standards like ISO8583 were not designed for blockchain applications and integrating them poses significant challenges.

Information Security

Traditional financial transactions simply require a credit card number to execute. This means anyone who accesses an ISO8583 message has all the information they need to execute fraudulent transactions.

Authority

Nodes in a blockchain are peers with no node having authority over the others. Payment card systems have centralized players, some of whom have unilateral control over key network decisions or disputes.

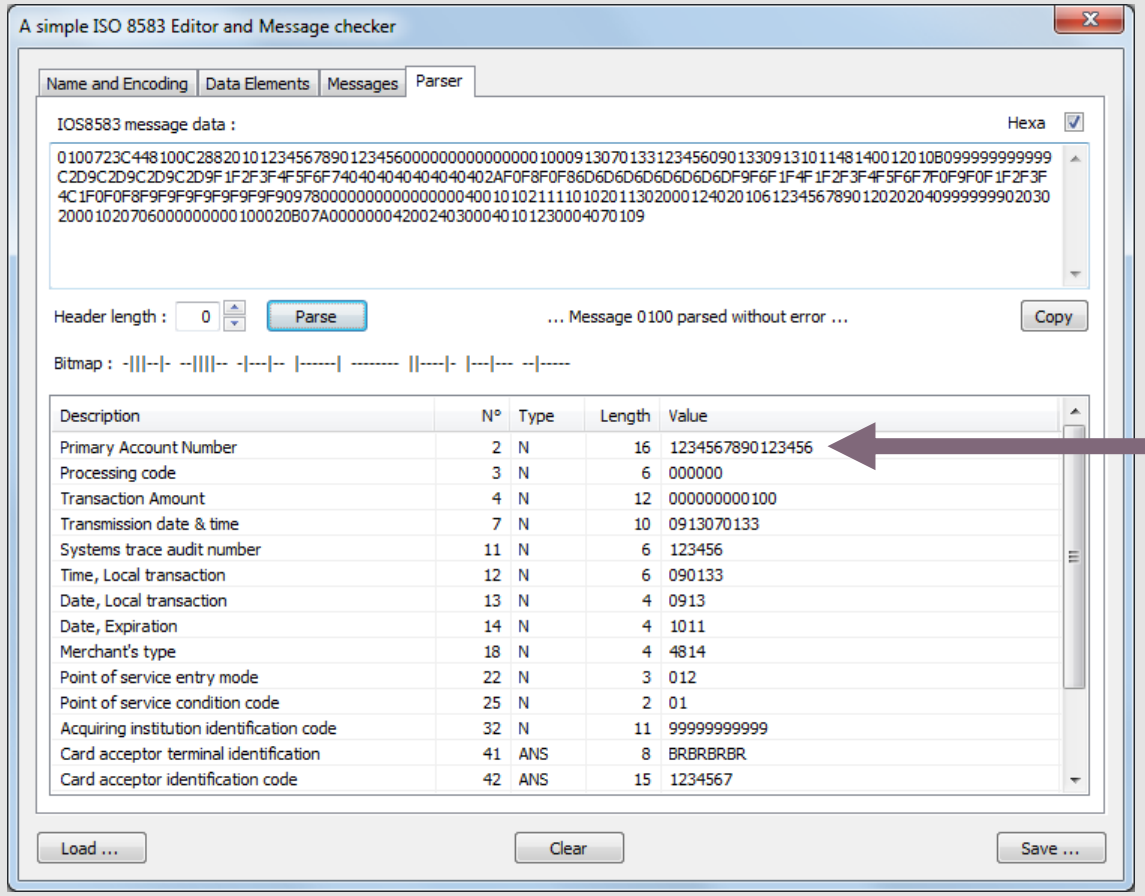


Reversals

Blockchain technology relies on finality of transactions as a key component of its consensus mechanism. Payment cards transactions meanwhile are legally required to be reversable in the case of fraud.

As mentioned earlier, anyone on the network can unpack ISO8583 messages, which means they have complete access to the details held within.

This includes all of the sensitive financial information you would need to issue and authorize false payments.



Primary Account Numbers (PANs), more readily recognized as a credit card number, are a required field for most messages on the network.

Anyone who has access to this message would be easily be able to perform fraudulent purchases with this information

All ISO8583 messages are extremely sensitive and it's extremely important to limit access to them. Banks avoid this through careful control of network access privileges.

Even innocent Stackoverflow questions can easily revel sensitive banking information inadvertently.

The essence of the whole ISO message, contain information about the transaction such as ...

- transaction type,
- amount,
- customerid

and so on.

So, After i reading these two web references, I want to make divide my ISO messaging log as MTI, bitmap, and Data Element.

For example.

```
 (0800 2020000000800000 000000 000001 3239313130303031)
MTI: 0800 (1987 version, Network Management Message, Request, Acquirer)
Bitmap: 20 20 00 00 00 80 00 00 (eg. 20 = 0010 0000 ,so position 3 is on)
DataElement:(by seeing Bitmap , we can defined data element as follow)
     field 03:000000 (Processing Code)
     field 11:000001 (Systems trace audit number)
     field 41:3239313130303031 (Card acceptor terminal idenfication)
```

But my problem is, I already have ISO 8583 messaging log from my ATM Machine. This actual output messaging log is not very clear like this upper example. So I cannot divide this message to MTI, Bitmap and Data element like upper example.

Here are my Example of data

```
00 14 5e 47 2e d8 00 1a d4 0c 32 0f 08 00 45 00
00 7b b2 ec 40 00 80 06 e5 29 ac 11 05 37 ac 11
05 0d 1a 78 1a 78 bf 1c 66 c8 8f 11 b5 a9 50 18
3f b6 c8 f6 00 00 00 51 31 31 1c 30 30 32 1c 1c
1c 31 3b 1c 3b 35 32 36 34 30 32 31 37 30 33 32
36 34 30 32 34 3d 31 34 30 35 32 32 31 31 30 30
```

Here a user is asking for help processing the bitmap output of their ATM Machine.

That bitmap data has revealed their credit card number, expiry date and other information.

What you have there as a sample is just the representation of the transaction info as it's transmitted over the wire. This is effectively the way all data transmission looks like at the transport layer, regardless of application.

Depending on the terminal management application/switch you're using (Postilion and Base24 are good examples), there should be a translation of that hex payload into ASCII text somewhere in your logs.

For the sample you have, you should first convert it to binary and then convert the binary result to ASCII. Using those steps, I can tell you the Institution Identifier Number (or Bank Identifier Number) in that sample is ***526402***. The snippet you've posted contains the Track 2 data, which also has the PAN in it. I'm not posting that here for obvious reasons (I'm not even going to apply the masking to it)

Share Edit Follow

answered Jan 23, 2014 at 8:39

kolossus

20.3k • 3 • 50 • 100

Unpacked payload data this user is referencing:

```
h: ";526402******4024=1405221100" # Track 2 Data // masked and expired
```

Not only can we extract the card number, the card number itself tells us a large amount of information itself. For example, we also know that this transaction is using a Mastercard Debit Card and is referencing an account at Banque Du Caire in Egypt. Track 1 would have revealed his full name.

This is because the ISO8583 messages and authentication system is very vulnerable to fraud and false authorizations, and the industry expects to lose more than $400B USD over the next decade.

Due to the relatively insecure, and archaic infrastructure used to authorize and transmit card information, a myriad of attack vectors on ISO8583 messages exist, from “man in the middle” intercept attacks to placing malware on point of sale units.

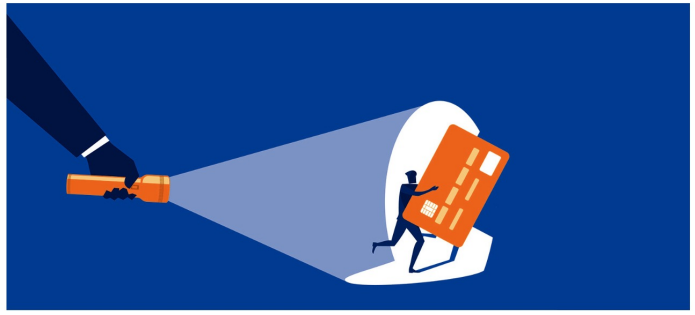
First contact: An introduction to credit card security

Written by Timur Yunusov

I bet you have several cards issued by international payment systems (e.g. Visa or MasterCard) in your wallet. Do you know what algorithms are used in these cards? How secure are your payments? People pay with such cards every day but know very little about them. Numerous myths accompany card payments. But to understand what tricks can be used to steal money from a card, you must first get an idea of the payment mechanisms used in it.

First contact: How hackers steal money from bank cards

Written by Timur Yunusov



Network fraudsters and carders continuously invent new ways to steal money from cardholders and card accounts. This article discusses techniques used by criminals to bypass security systems protecting bank cards.

INFO

See an overview of card payment security mechanisms in our previous article.

First contact. Attacks against contactless cards

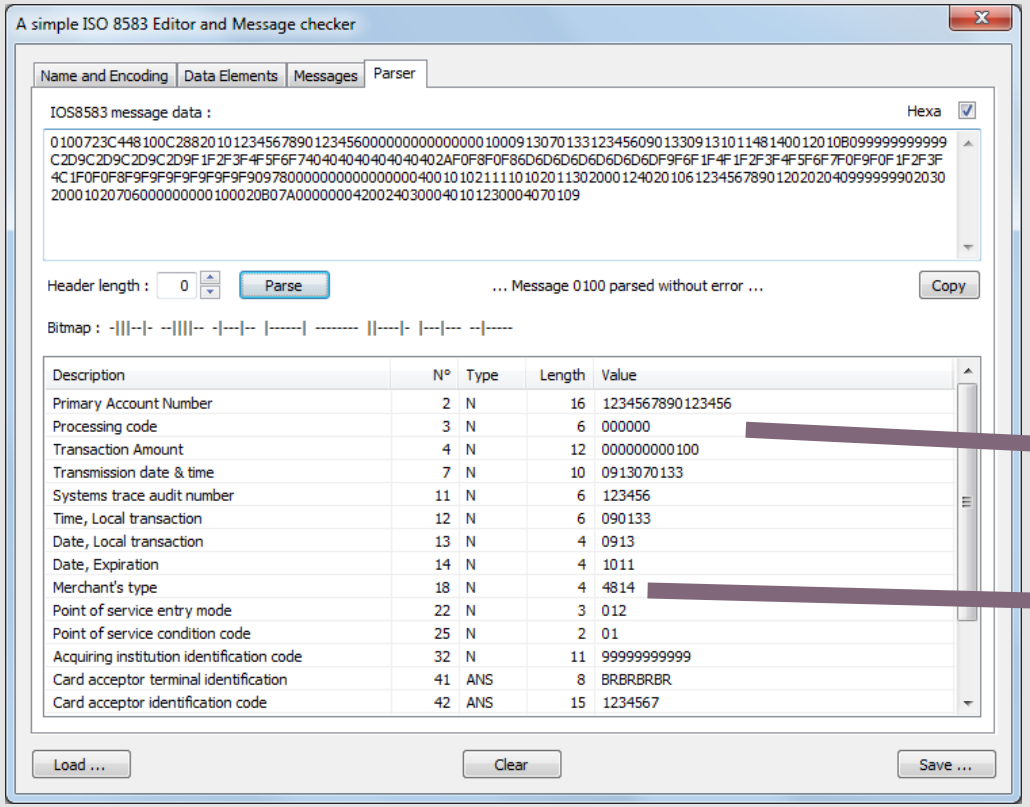
Written by Timur Yunusov

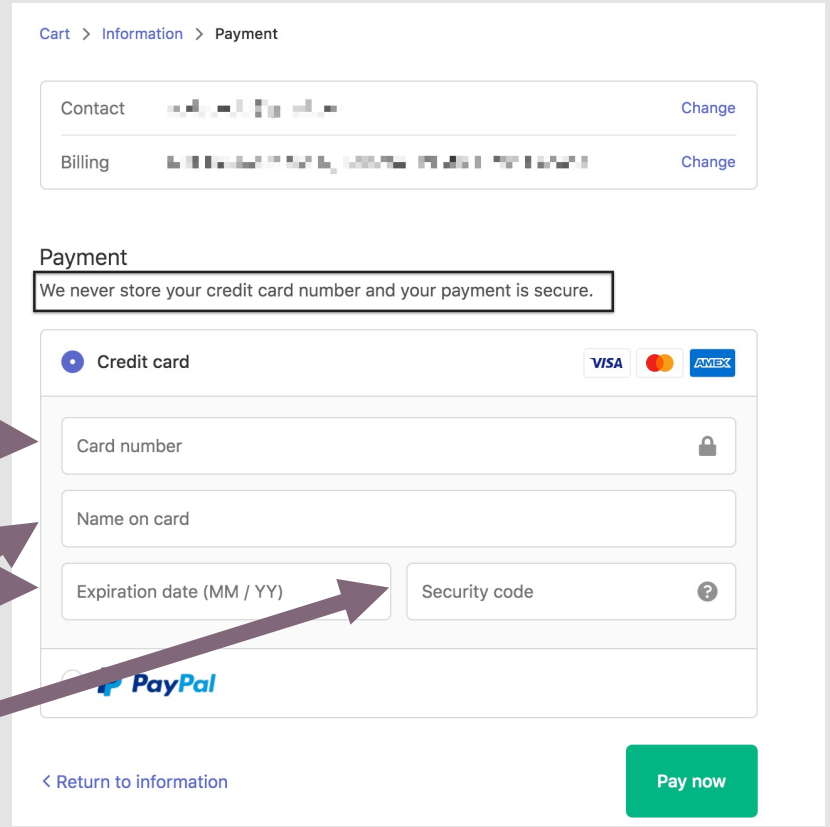
Contactless payment cards are very convenient: you just tap the terminal with your card, and a few seconds later, your phone rings indicating that the transaction is completed. But this convenience has a downside: malefactors can steal money from such cards. This article provides an overview of methods used to hack NFC (near-field communication) payment cards.

The full breath and history of possible attack vectors is beyond the scope of this report. For the curious, we’d recommend the Hackmag series on payment card attacks. For the purposes of this proposal, it is important to note that any actor worldwide can easily conduct fraudulent transactions and wreak havoc, by just intercepting a single ISO8583 message or compromising a single terminal.

Payment cards don't use public/private key cryptography. Instead, Authorization is dependent on only a few fields, all of which are present that ISO8583 message we showed earlier.

The card number, expiration date, and CVV are oftentimes the only information a sophisticated attacker needs to commit fraud.

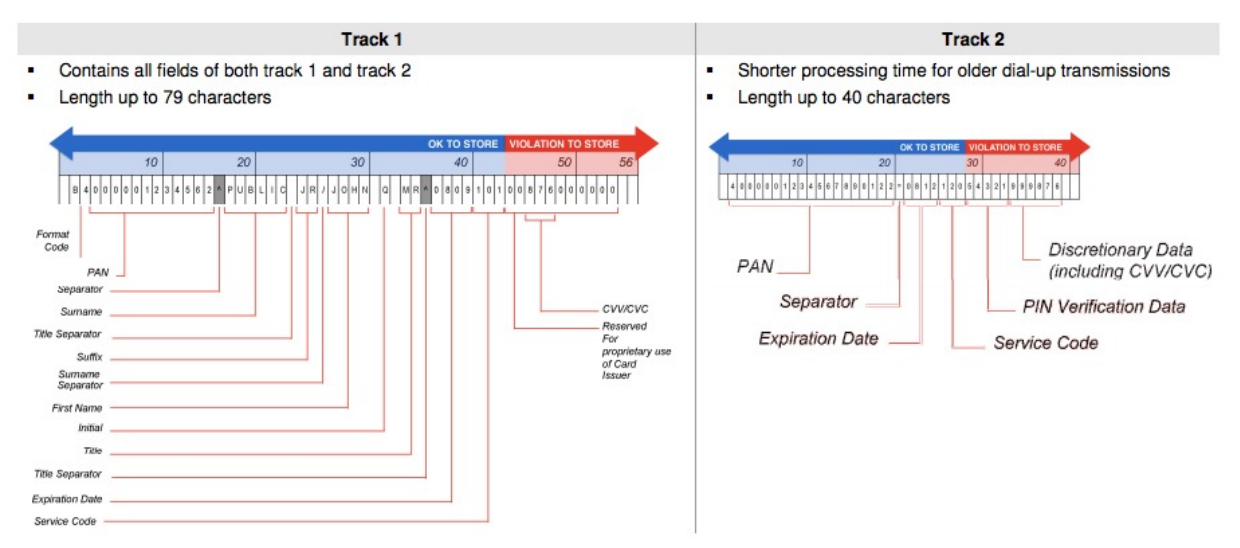




Track 1 data fields (including full name) are stored in field 45
PIN numbers are usually stored in field 52
CVV number or the security code is usually stored in field 61

Even your browser won’t save the CVV/CVC/PIN number in autofill as it’s against the TOS for any company outside of the issuing bank to store that information.

Intermediaries can only store certain, limited information related to the transaction.



While the user’s name and card number itself can be saved for archiving transactions, companies are banned by the terms of service from saving the PIN or CVC/CVV number associated with a card.

Google itself never saves that information, that’s why security codes are never auto-populated in payment forms.

Comment 2 by jsaul@google.com on Mon, Aug 3, 2020, 3:49 PM EDT

Project Member

Status: WontFix (was: Assigned)

Ah, yep, I can explain this one. Hi there, thanks for reporting this. This is actually the bank's/American Express's doing, not ours, so it's unfortunately out of our hands. Essentially:

1) Google doesn't store CVC ever, so in order to verify you're the cardholder, we send the card and the CVC you provide to the issuer/network and ask them to auth it.
2) In this case, AmEx says A) they looked at the CVC and B) the auth is successful. We treat that as a success and return the full card number through Chrome Autofill.

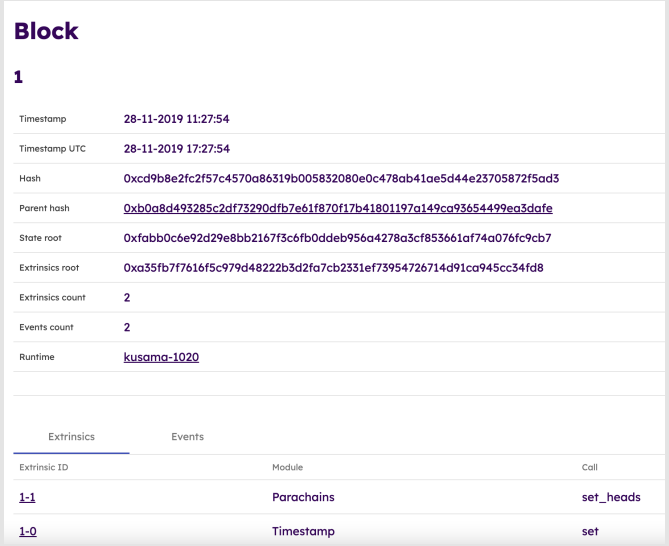
Now on AmEx's side, they'll sometimes allow one digit to be incorrect or slightly off, based on their own risk models of how certain they think you are who you say you are, and in what ways the digit was incorrect. From your report, it sounds like they're more lenient with the first and last digit, but not the middle two. But if they say it was a successful CVC check and you're the cardholder, we have no way to dispute that, because Google doesn't actually know the true CVC.

I'll mark this as "working as intended" since there's nothing we can do here, but I'm happy to explain further if you have any questions! Thanks.

That being said, the actual way browsers populate credit cards is to authorize your card when you autofill them. It’s likely that the browser also provides additional location, site, and other usage fields to the bank to help them distinguish between legitimate and fraudulent uses of a payment card. As a tangent, the reason for Amex’s “wiggle room” isn’t due to a fraud model, it’s likely for compatibility. Amex CVVs are 4 digits while industry standard is 3.

The open, eternally persistent, permissionless nature of blockchains possibly makes it the worst network architecture for transmitting or processing ISO8583 messages, even if they were encrypted.

Data on blockchains can be retrieved at anytime through an RPC call and is is permanently stored in order to prove the provenance of the chain.



Block

1

Timestamp	28-11-2019 11:27:54
Timestamp UTC	28-11-2019 17:27:54
Hash	0xcd9b8e2fc2f57c4570a86319b005832080e0c478ab41ae5d44e23705872f5ad3
Parent hash	0xb0a8d493285c2df73290dfb7e61f870f17b41801197a149ca93654499ea3dafe
State root	0xfabb0c6e92d29e8bb2167f3c6fb0ddeb956a4278a3cf853661af74a076fc9cb7
Extrinsics root	0xa35fb7f7616f5c979d48222b3d2fa7cb2331ef73954726714d91ca945cc34fd8
Extrinsics count	2
Events count	2
Runtime	kusama-1020

Extrinsics | Events

Extrinsic ID	Module	Call
1-1	Parachains	set_heads
1-0	Timestamp	set

Here is the Polkadot genesis block including all of the raw data. For obvious reasons, any ISO8583 information included into the ledger would be a potential liability and security hazard. A robust implementation needs to prevent even encrypted ISO8583 messages on chain.

The reason blockchains cannot employ an encrypted data standard is that all nodes need to be able to execute the message's payload in order to maintain consensus. Even if you set up a private network with a secure key (also referred to as a permissioned network), if that key was ever leaked, a malicious actor would be able to retroactively decrypt all of the ISO8583 messages, retrieve the account and authorization information, and use the PAN numbers to commit widespread fraud.

Placing any ISO8583 message on the chain is a significant security risk.

Permanent Storage:

By its very nature, blockchains are immutable and permanently stored, meaning that sensitive transactions will persist on-chain for as long as the network exists.

No Access Control:

Permissionless networks and RPC protocols allow anyone in the world to directly access the details without any controls or restrictions.

Lack of Encryption:

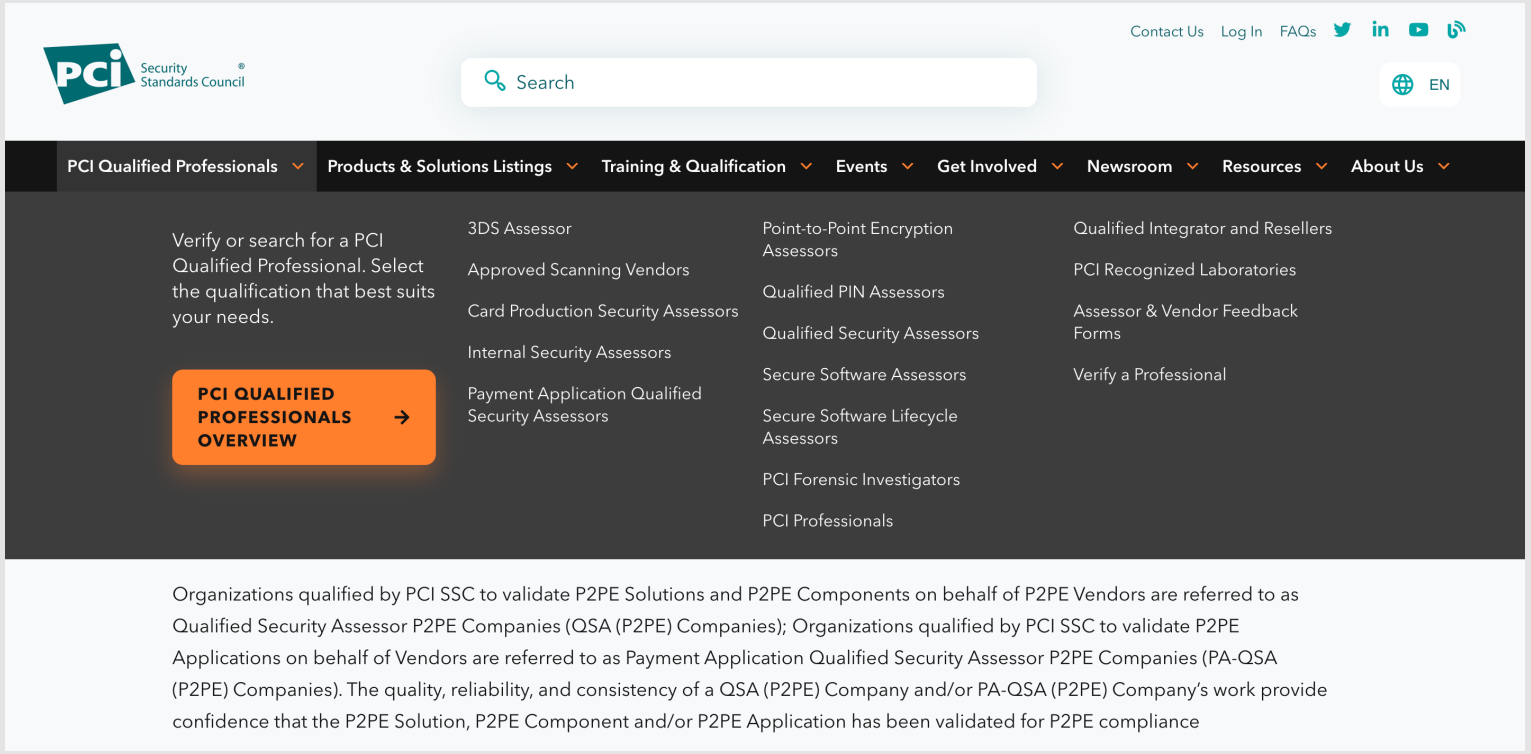
All peer nodes need to be access and validate the information of the blockchain in order to update their own ledgers and arrive at consensus about the state which limits encryption techniques.

Legal Liabilities:

Some US and European States legally obligate companies to expunge all financial card data after a given time period.

Compounding these challenges are the strict requirements of Payment Card Industry Data Security Standard (PCIDSS) compliance required for most financial applications.

PCIDSS is designed to reduce fraud and improve security. Compliance is rather arduous to achieve and must be evaluated ever quarter.



Though PCIDSS is not required by US Federal law, it is de facto necessary.

State Laws:

Several states have language specifying that every firm in the card payments chain be PCIDSS certified or at least conform to its specification.

Business Insurance:

In the case of a breach, PCIDSS compliance has a material impact in many jurisdictions on an indemnitee's liability. In some cases, PCIDSS may even provide prosecution immunities or other legal protections.

Terms of Service:

Many if not most networks and financial firms mandate PCIDSS compliance in their terms of service for all partners, vendors, and suppliers.

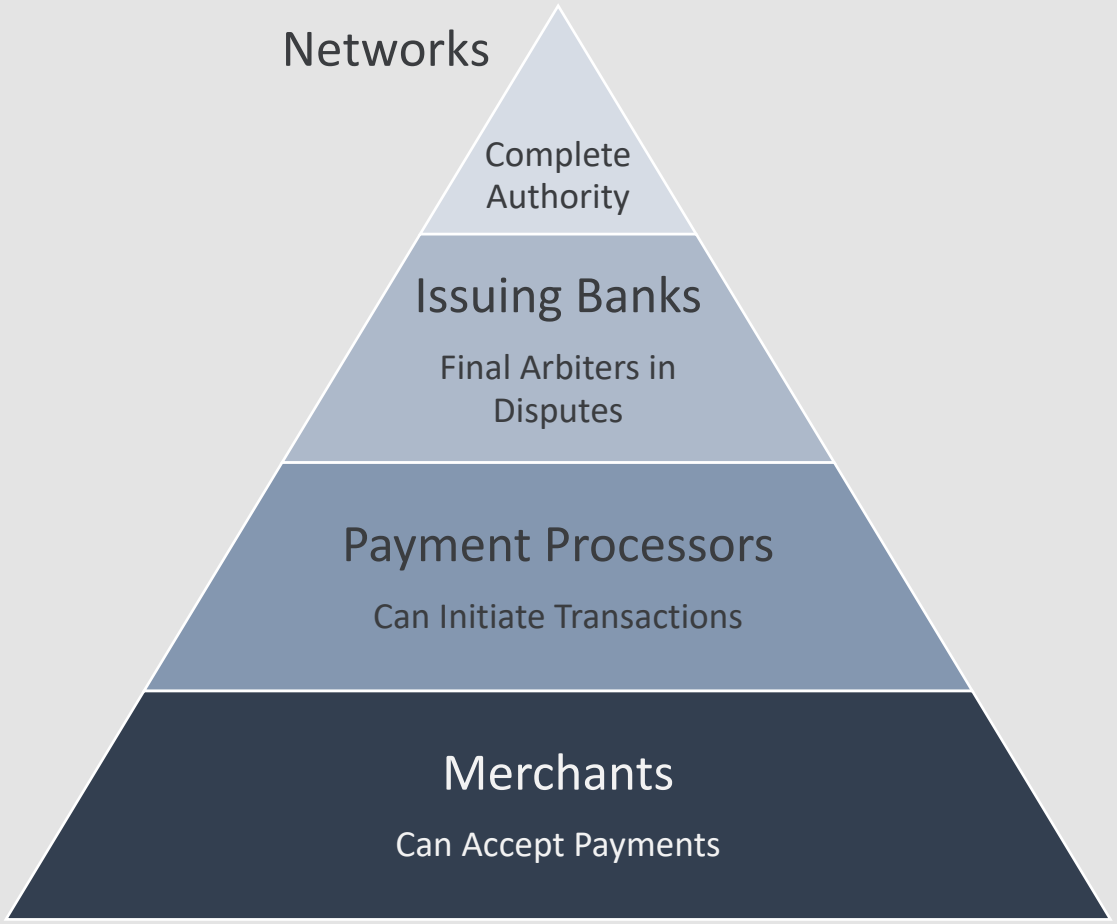
Public Perception:

Despite the fact that consumers in the US and EU are largely protected against unauthorized use of their cards, it isn't perfect, and consumers are very sensitive to the security of their financial information.

Unlike blockchains, traditional payment networks are explicitly centralized and have several privileged actors who can make and impose unilateral decisions governing transactions or the network.

Blockchains have famously flat topology where every node is an equal peer in the network with equal privileges.

In traditional payment networks however, MasterCard and Visa are king and the merchants have limited recourse.



This is a challenge to adopt on modern blockchains architectures through cryptography as a breach of the centralized player's privileged private keys might collapse the entire network.

Assigning account privileges on-chain is a poor solution as the financial industry regularly get's breached and is certain to lose control of keys.

The First American Corporation

885M Credit Card Applications
A simple oversight on webpage access control.

EQUIFAX

147M Credit Profiles
Failed to patch a well-known vulnerability Apache Strusts

Heartland

130M Credit Card Numbers
SQL Injection and Physically Stealing Computers

Capital One

100M Credit Card Applications
Cloud data stolen by an AWS employee

Historic attacks have ranged from Russian hackers, disgruntled AWS employees, to attackers simply stealing physical hardware. There are simply too many attack vectors to assume even the most secure banks in the world can maintain operational security of their private keys. It is better to assume every network participant will lose their keys at some point and will require a recovery mechanism.

Depending on which key is breached, the impact could be catastrophic.

Network's Private Keys:

A breach in the network key on-chain would be catastrophic. It would basically allow the attacker to gain full control over the network and issue fraudulent transactions unilaterally causing complete network collapse.

Issuing Bank's Private Keys:

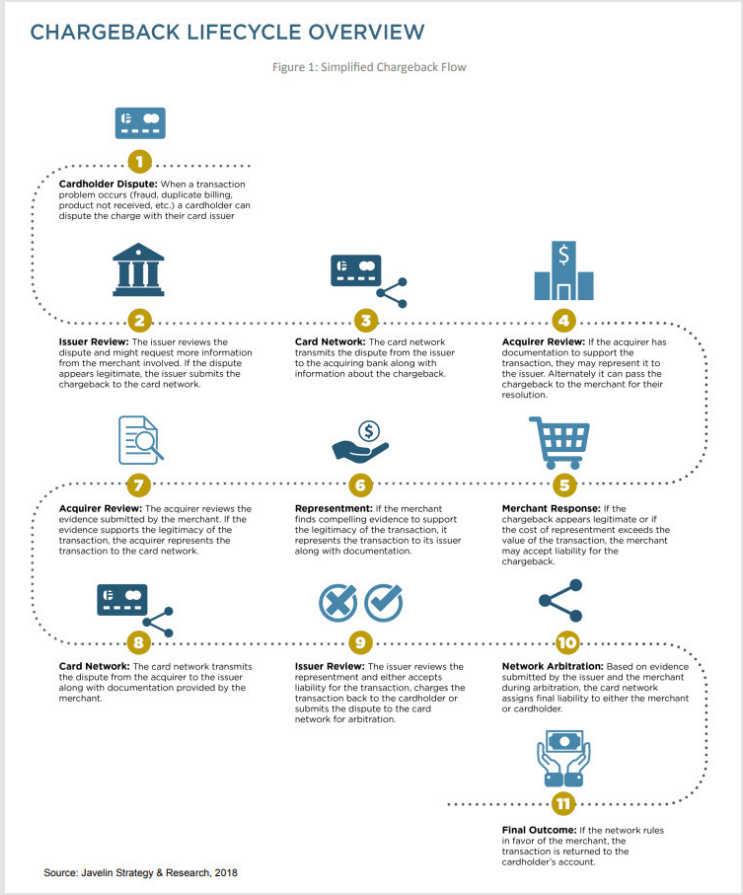
Losing control of an issuing bank's key is slightly less problematic, however it would allow a user to reverse transactions to their own personal benefit. This would likely come with significant random reversals as well to disguise their identity quickly eroding confidence in the network.

Payment Processor's Private Keys:

Even if data is stored in an encrypted form on-chain, a breached payment processer key would allow the attacker to read all the PAN numbers since inception. If data is kept off-chain this key could allow an attacker to initiate and possibly perform fraudulent transactions.

Adapting these responsibilities to a peer to peer structure doesn't work as members of the network must respond to a unilateral, centrally mediated fraud process when it comes to reversals.

Fraud disputes are a complex process that involves both manual reviews and network arbitrations.



This chargeback process was something that Satoshi was attempting to solve in the original Bitcoin implementation.

Transaction Finality:

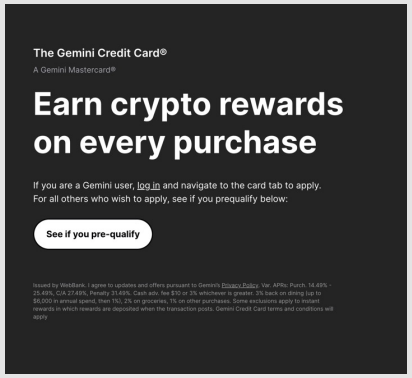
Transactions on every blockchain are final as the computed hash is what allows the network to maintain consensus. One of the key principles of Bitcoin's original implementation that it has maintained to the present is that transactions are final and there is no way for the initiating wallet or any wallet other than the receiving user to reverse the transactions.

Challenges with Escrow:

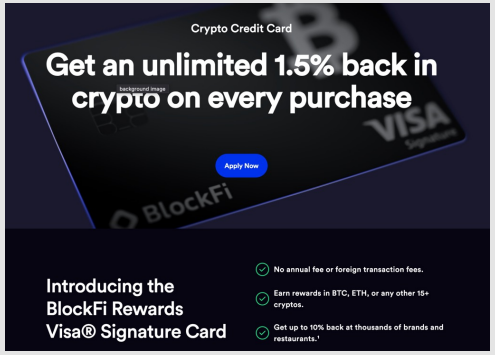
Satoshi envisioned Escrow accounts being used to manage a reversal process, however the use of escrow accounts is itself problematic as it would lock up otherwise productive capital for significant amounts of time. In addition, it still falls on a centralized entity to mediate whether or not the terms of the escrow payment were fulfilled if the transaction involved off-chain information.

In terms of attempted and in-market commercial products, most “crypto-related” payment cards have no direct or indirect exposure to the blockchain.

These cards range from regular credit cards with cryptocurrency based rewards issued by traditional banks..



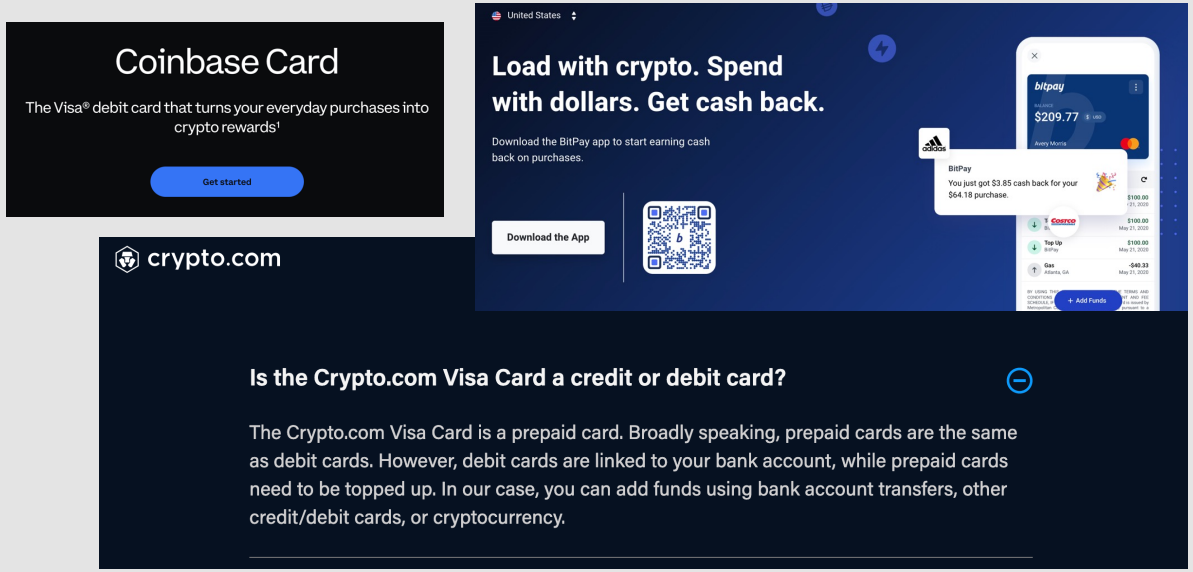
Gemini Credit Card



Blockfi Credit Card

These credit cards are traditional products, white labeled, underwritten, and administrated by a traditional chartered bank. The main blockchain connection, is the exchange provides the rewards in tokens instead of cash. This benefits all parties as the banks profit off of underwriting the loans, and the exchanges gain additional users. In addition for every transaction, the issuing bank and exchange receives ~2% of the purchase as a fee, called interchange. This interchange funds the reward, and any additional excess is direct profit for the bank and exchange. These accounts are completely separate from the blockchain, and do not reference any ledger addresses.

..To simple pre-paid debit cards, where the exchange automatically sells your crypto and loads the cash balance to a debit card.



These debit cards are even more straight forward. Best though of as a gift card, these payment cards are simply loaded with cash by the exchanges by selling your crypto currency. This balance is then available to spend as it’s held in cash by MasterCard or Visa under their prepaid accounts. There is no direct connection to the blockchain as the exchange has all of the assets pooled under their own addresses.

These products are simply traditional products with a blockchain veneer, and lack any integration. They serve primarily as a way for incumbent networks to capitalize on blockchain's brand.

VISA and Mastercard are attempting to push these traditional white label cards in an effort to prevent users from adopting truly native blockchain based card payment systems that can be used independently of their network.

We are also providing on-ramps for crypto players creating connectivity with fiat economies. There are over 65 crypto platforms and exchanges that have partnered to issue Visa credentials. This quarter, Visa credentials in crypto wallets had ***more than $2.5 billion in payments volume****, which is already 70% of the payments volume for all of fiscal 2021.*

In addition to embedding credentials in crypto platforms, we continue to innovate around our settlement and crypto API capabilities which have been key differentiators for us for fintechs and financial institutions that are looking to extend crypto capabilities to their customers. We will continue to lean into the crypto space and our strategy is to be a key partner to provide the connectivity, scale, consumer value propositions, reliability and security that is needed for crypto offerings to grow.

Earlier this month, we previewed CBDC payment APIs currently in development which would enable central banks to connect their Ethereum-based CBDCs with Visa rails through a wallet with digital issuance capabilities enabling consumers to spend with CBDCs at any Visa merchant. We partnered with ConsenSys to develop this concept which was selected as one of the winning entries out of 300 ideas from 50 countries at the global CBDC challenge as part of the Singapore FinTech Festival judged by representatives from the IMF, the World Bank, the Bank of International Settlements (sic) [Bank for International Settlements] (00:07:34), and the central banks of Brazil, India, Kenya and Indonesia.

VISA Q1 2022 Earnings Call

These incumbent networks and financial institutions have patented most applications in the space, though these patents seem overbroad and their validity and protections remain untested.

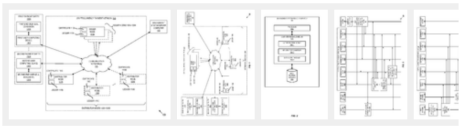
For example, VISA owns the patent for all cryptocurrency infrastructure systems in general in the US.

Cryptocurrency infrastructure system

Abstract

Embodiments of the present invention are directed to methods and systems for managing a cryptocurrency payment network comprising one or more issuer nodes and one or more distributor nodes. Issuer nodes may be granted different rights from distributor nodes with respect to the issuance and distribution of digital currency within the cryptocurrency payment network. A management system server computer may generate unique node verification key pairs for each node in the cryptocurrency payment network, where the node verification key pairs may be used to identify and authenticate issuer nodes and distributor nodes.

Images (8)



Classifications

G06Q20/3829 Payment protocols; Details thereof insuring higher security of transaction involving key management

View 3 more classifications

US20150371224A1
United States

Download PDF | Find Prior Art | Similar

Inventor: Phaneendra Ramaseshu Lingappa

Current Assignee : Visa International Service Association

Worldwide applications

2015 · US 2021 · US

Application US14/749,573 events

2014-06-24	Priority to US201462016556P
2015-06-24	Application filed by Visa International Service Association
2015-06-24	Priority to US14/749,573
2015-07-24	Assigned to VISA INTERNATIONAL SERVICE ASSOCIATION
2015-12-24	Publication of US20150371224A1
2021-07-06	Application granted
2021-07-06	Publication of US11055707B2
Status	Active
2037-10-22	Adjusted expiration

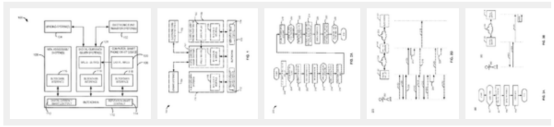
And American Express owns the patent for operating a payment network on a blockchain based ledger.

Systems and methods for blockchain based payment networks

Abstract

A system for operating a payment network with a blockchain-based ledger may prepare a request to complete a transaction from an account associated with a payer digital wallet for entry on the blockchain. The request may include an amount and payee address associated with a payee digital wallet. The system may also send the request to the blockchain using a blockchain interface. The system may approve or decline the request. The system may further adjust a balance of the payer a balance of the payee to reflect approval of the request. The adjustment may include writing the transaction to the blockchain.

Images (13)



Classifications

G06Q20/4016 Transaction verification involving fraud or risk level assessment in transaction processing

View 5 more classifications

US20180075453A1
United States

Download PDF | Find Prior Art | Similar

Inventor: Sastry Durvasula, Andras Ferenczi, Upendra Mardikar, Keshav Narsipur, Vishnuvajhala Subrahmanyam

Current Assignee : American Express Travel Related Services Co Inc

Worldwide applications

2016 · US 2020 · US

Application US15/266,350 events

2016-09-15	Application filed by American Express Travel Related Services Co Inc
2016-09-15	Priority to US15/266,350
2016-09-15	Assigned to AMERICAN EXPRESS TRAVEL RELATED SERVICES COMPANY, INC.
2018-03-15	Publication of US20180075453A1
2020-02-03	Assigned to AMERICAN EXPRESS TRAVEL RELATED SERVICES COMPANY, INC.
2020-11-10	Application granted
2020-11-10	Publication of US10832247B2
Status	Active
2038-09-18	Adjusted expiration

Disclaimer this is not legal advice, for full clarity please consult a lawyer in your jurisdiction. In our non-legal opinion, these patents are overly broad and likely unenforceable. Cryptocurrency infrastructure has been around well before VISA “developed” it 2014, and users have been using the blockchain to facilitate payments prior to America Express’s ”innovation” in late 2016. Any competent legal team should be able to find prior art of both use cases, and invalidate these patents. It is likely this is why neither of these patents has ever been attempted to be enforced despite the large number of “infringing” firms and projects. Actual patent litigation in the blockchain space has been incredible rare, and these patents seems to be largely untested.

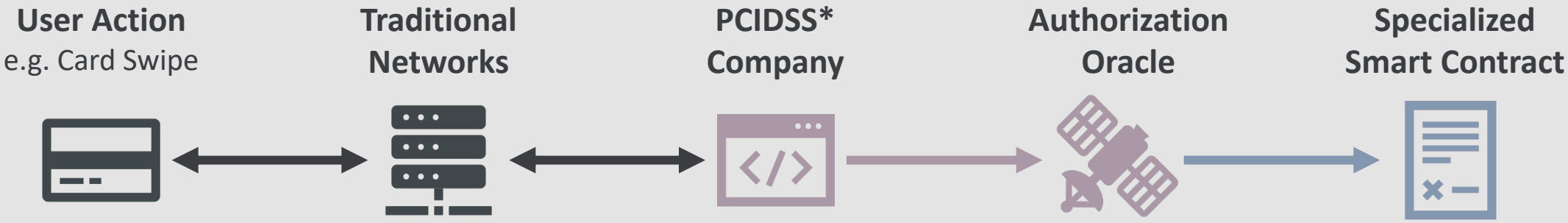
While directly sending or processing ISO8583 messages on an open P2P blockchain is impossible from a security perspective, we may be able to clear them elsewhere and settle the transactions on-chain.

Ultimately, the goal of any successful ISO8583 integration is to allow users to access their on-chain balance, and spend it at any merchant that accepts payments cards worldwide.

We may be able to enable this by centralizing key functions, however this centralized entity will also have to solve the key problems of information security, authority and its recovery, and reversals.

We propose using an oracle to simultaneously allow users to manage their funds like any ERC20 token while accepting and processing ISO8583 messages.

This architecture allow users to perform wallet-to-wallet transfers on chain like any ERC20 token with their private key, while allowing a centralized company to securely process ISO8583 messages and authorize them on the chain.

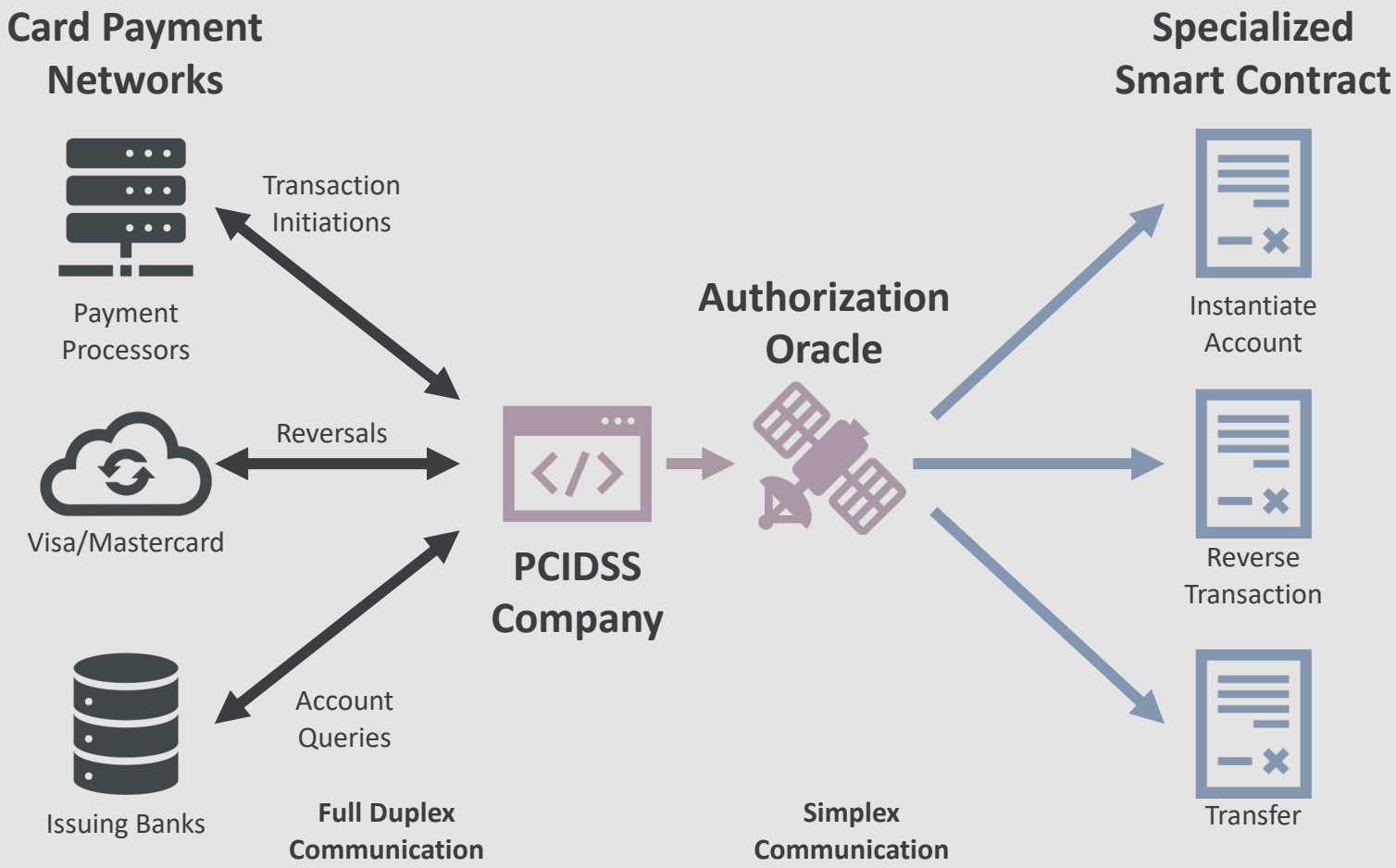


Here traditional networks send their ISO8583 messages to a specially setup PCIDSS-compliant company. This company processes the transaction and issues it for settlement through an authorization oracle. This allows the company to recover from breaches, arbitrate reversals, comply with payment regulations, and withhold sensitive ISO8583 details from being sent directly to the blockchain.

* PCIDSS is the set of standards that any company storing or processing payment card information is required to comply with.

This allows both payment networks using ISO8583 and users to issue transactions out of their accounts and conduct on-chain actions safely.

The company accepts ISO8583 formatted transactions off-chain, verifies the authorization, and issues on-chain actions through an authorization oracle.



This allows us to maintain information control and provides recovery options.

PCIDSS Company:

The card networks and financial firms send their ISO8583 messages to a specially setup PCIDSS company for processing. This company is responsible for processing ISO8583 messages, issuing responses, controlling data security, preserving PCIDSS compliance, and maintaining the registered bin number for message routing on the payment networks.

API Driven Oracle:

On-chain smart contract transactions can be initiated by a user for any account balances that isn't possibly reversable, or the oracle. This oracle is controlled by the PCIDSS company and provides control for reversals and on card network payments.

Recovery:

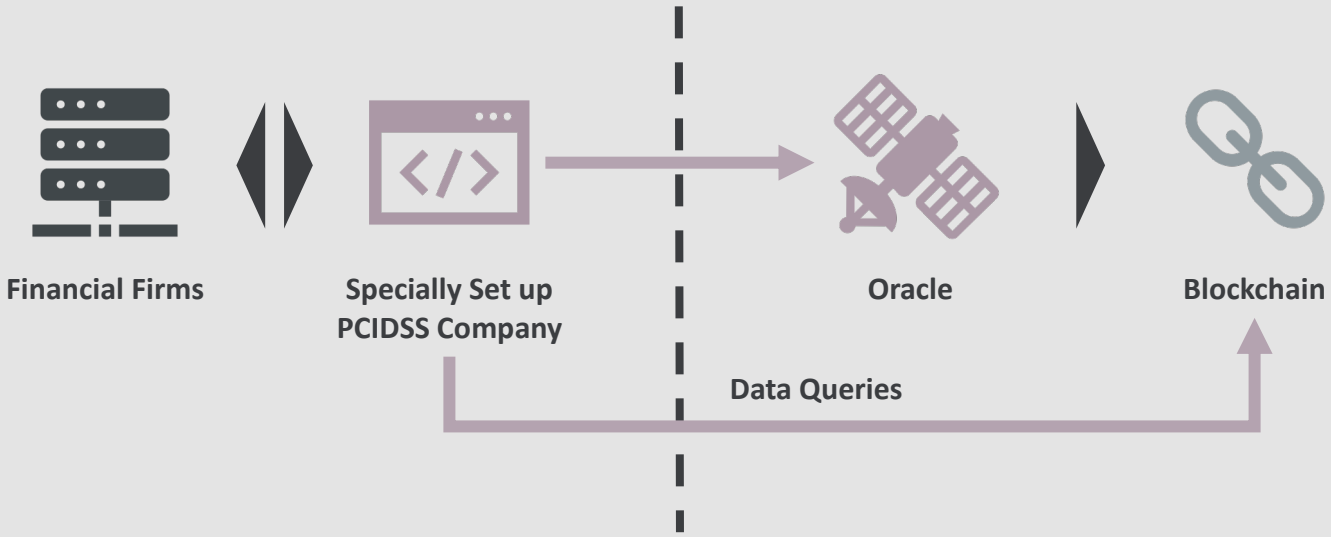
This eliminates the reliance on private keys. If the holding company's infrastructure is breached, there are several established ways to prove identity to the cloud server provider to recover control of the oracle's domain.

Agency:

As a blockchain native solution, users have address and keys and as such can directly manage their funds without needing to route payments through traditional financial networks.

The high degree of centralization of an oracle allows us to; hide ISO8583 information behind a privacy barrier, comply with PCIDSS requirements, and manage transaction authorizations in real time.

Payment gateways and networks ultimately route ISO8583 messages to the financial company that controls the oracle just as they would any other firm.



The only way the blockchain receives ISO8583 transaction information is through the oracle. This allows us to hide sensitive details like account numbers and authorization codes that are contained in the payload of an ISO8583 transaction. The PCIDSS compliant company also allows us to perform data queries on-chain to keep an up to date record of the ledger to perform balance checks necessary for real-time authorizations.

The PCIDSS Company also helps smooth transaction and data flows.

Information Security:

The most important functions of the PCIDSS company are to censor ISO8583 transactions and perform authorization checks before it sends the transaction to the oracle to be placed on-chain. This allows us to hide all authorization info and account numbers, as well as destroy the information after performing the transaction. This is needed to comply with current state and future federal government regulations on payment card information security.

Realtime State Information:

As the PCIDSS Company has control over the authorizing oracle, pending transactions and insight into the current state of the blockchain, it's able to verify account balances and perform real-time authorizations of funds, a requirement for use on payment card networks.

Sole Source of Truth:

The company also maintains a real-time record of the blockchain's state along with pending transactions to ensure an accurate understanding of a user's financials at any given time.

At a high level, here is the comparison between traditional networks and through our proposed oracle-based authorization flow.

Authorization requests are routed by BIN number and are sent to either the issuing bank or our company for a response.

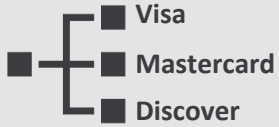
Merchant
Amazon.com

A user presents their card or card details at an authorized accepting merchant.

The merchant sends it to their payment processor

JSON, XML or Custom API

Traditional Network
Stripe/MasterCard



The payment processer formats the message into ISO8583 and forwards it through the appropriate network to the card's issuing bank.

ISO8583 or AS2805

Issuing Bank
Chase

The issuing bank verifies the card details and authorization information and makes an approve/decline decision. It then sends this message back through the network.

ISO8583 or AS2805

Payment Processer
Stripe

After the response reaches the payment processer, they are forwarded to the merchant, but all the approvals are stored for the settlement process.

Merchant
Amazon.com

These cards would be specially issued by the Oracle company to tie their PAN number to a blockchain address.

It is then processed normally.

JSON, XML or Custom API

Traditional Network
Stripe/Mastercard/Substrate

The payment processer formats the message into ISO8583 and forwards it through the network which routes to the PCIDSS company.

ISO8583 or AS2805

Oracle Company
PCIDSS Company

Here the specially setup company checks the balances on-chain and the pending transactions to verify that a user has sufficient funds. It issues it's own ISO8583 formatted approve/decline message and a substrate transaction.

ISO8583 or AS2805

+

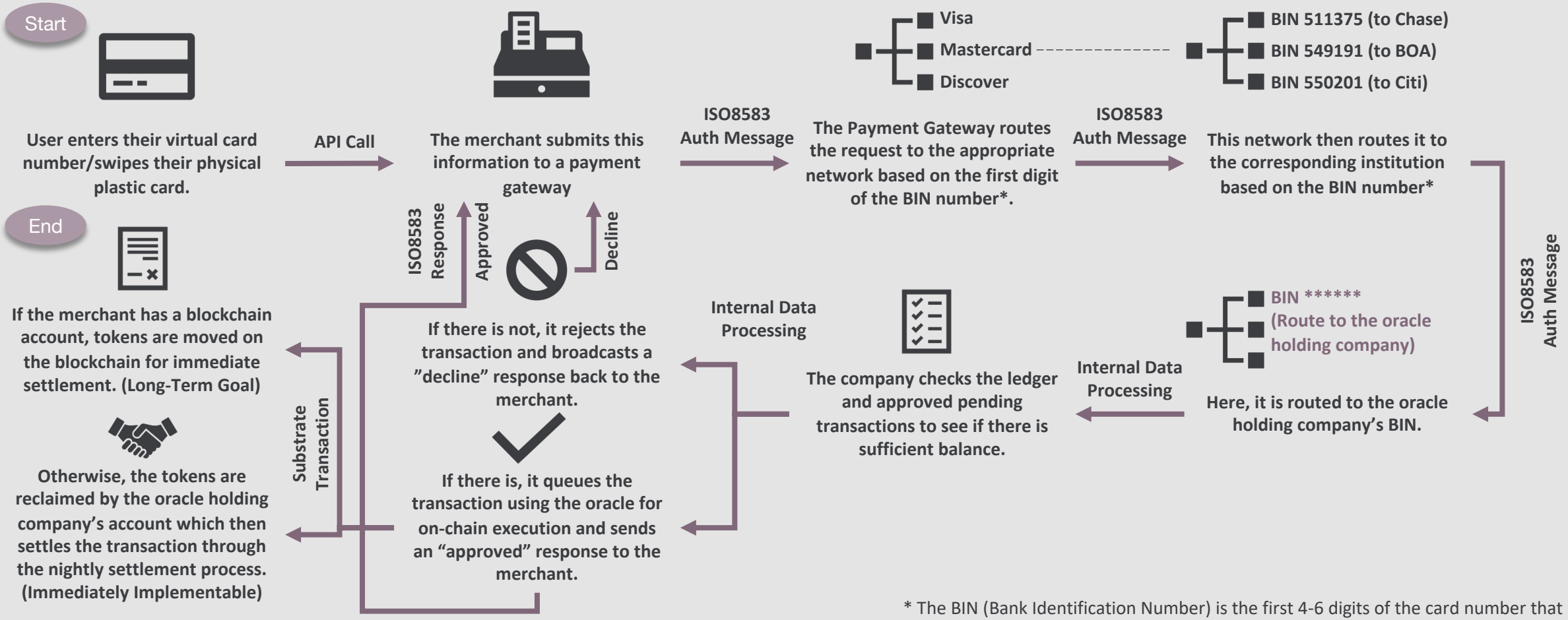
Substrate Transaction

Substrate Blockchain
Polkadot

The response is not only sent to the payment processer and merchant, but the transaction is also queued on the oracle for settlement on the blockchain.

Here is that same flow for our proposed oracle solution in deeper technical detail.

The full flow is significantly more complex, however despite this backend complexity, the process is seamless from the user's and merchant's point of view and the customer experience is exactly the same as using a generic bank issued payment card.



* The BIN (Bank Identification Number) is the first 4-6 digits of the card number that indicates key information such as the network that issued it, and the authorizing bank.

In order to minimize currency risks, it's ideal if the smart contract tokens are some form of custodied stablecoins.

For the simplest implementation, the smart contract's tokens would ultimately be backed by custodied assets (USD ideally) held by the oracle holding company, this allows them to easily be liquidated and used in the nightly settlement process.

VISA Individuals Businesses Innovators Everyone

Visa Becomes First Major Payments Network to Settle Transactions in USD Coin (USDC)

3/29/2021

With the direct acceptance of payments in USD Coin, Visa forges new connections between digital and traditional currencies.

SAN FRANCISCO--(BUSINESS WIRE)--Mar. 29, 2021-- Visa (NYSE: V) today announced a major industry first in bridging the worlds of digital and traditional fiat currencies: the use of USD Coin (USDC), a stablecoin backed by the US dollar, to settle a transaction with Visa over Ethereum—one of the most actively used open-source blockchains.[1] Visa is piloting the capability with Crypto.com, a Visa partner and one of the world's largest crypto platforms, and plans to offer the USDC settlement capability to additional partners later this year.

This press release features multimedia. View the full release here: https://www.businesswire.com/news/home/20210329005171/en/

Visa Becomes First Major Payments Network to Settle Transactions in USD Coin (USDC) (Graphic: Business Wire)

Support for digital currencies as a new type of settlement currency marks an important step forward for Visa's network of networks strategy, which is designed to enhance all forms of money movement, whether on the Visa network, or beyond. By harnessing its global presence, partnership approach, and trusted brand, Visa is focused on adding differentiated value to the ecosystem and making cryptocurrencies more secure, useful, and applicable for payments.

Visa has spent the last year establishing a pathway for digital currency settlement within Visa's existing treasury infrastructure, a platform that moves billions of dollars each day across thousands of institutions in more than 200 markets and 160 currencies. Working with Anchorage, the first federally chartered digital asset bank and an exclusive Visa digital currency settlement partner, Visa has launched a pilot that allows Crypto.com to send USDC to Visa to settle a portion of its obligations for the Crypto.com Visa card program.

There is precedent for using 3rd party stablecoins themselves to settle on VisaNet, VISA's inter-firm settlement network. In 2021, VISA piloted the ability to settle in USDC by partnering with Crypto.com, though it didn't see widespread adoption, and ultimately relies on a combination of Anchorage and Crypto.com to facilitate the redemption to USD to settle the debt. In our view, settling on VisaNet in crypto adds additional complexity and headache to the receiving partners on the network as they must now have separate systems to accept the settlement currency.

In our architecture, the oracle holding company simply reclaims the issued tokens and the custodied USD released is sent through VisaNet. This is **significantly** easier to implement than it probably seems at first glance as the settlement process is a batched process and is firm-to-firm for overall account imbalances and not performed on an individual transaction basis. The holding company only needs to audit the smart contract's overall token balance to the custodied assets after every settlement process instead of auditing accounts and transactions individually.

The oracle controlling company can also issue reversals as is required by law, though blockchain architectures and the ERC20 standard were not designed with this capability natively.

All payment card transactions, and therefore all ISO8583 transactions, must be reversable.

§1666. Correction of billing errors

(a) Written notice by obligor to creditor; time for and contents of notice; procedure upon receipt of notice by creditor

If a creditor, within sixty days after having transmitted to an obligor a statement of the obligor's account in connection with an extension of consumer credit, receives at the address disclosed under section 1637(b)(10) of this title a written notice (other than notice on a payment stub or other payment medium supplied by the creditor if the creditor so stipulates with the disclosure required under section 1637(a)(7) of this title) from the obligor in which the obligor—

(1) sets forth or otherwise enables the creditor to identify the name and account number (if any) of the obligor,
(2) indicates the obligor's belief that the statement contains a billing error and the amount of such billing error, and
(3) sets forth the reasons for the obligor's belief (to the extent applicable) that the statement contains a billing error,

the creditor shall, unless the obligor has, after giving such written notice and before the expiration of the time limits herein specified, agreed that the statement was correct—

(A) not later than thirty days after the receipt of the notice, send a written acknowledgment thereof to the obligor, unless the action required in subparagraph (B) is taken within such thirty-day period, and
(B) not later than two complete billing cycles of the creditor (in no event later than ninety days) after the receipt of the notice and prior to taking any action to collect the amount, or any part thereof, indicated by the obligor under paragraph (2) either—
(i) make appropriate corrections in the account of the obligor, including the crediting of any finance charges on amounts erroneously billed, and transmit to the obligor a notification of such corrections and the creditor's explanation of any change in the amount indicated by the obligor under paragraph (2) and, if any such change is made and the obligor so requests, copies of documentary evidence of the obligor's indebtedness; or
(ii) send a written explanation or clarification to the obligor, after having conducted an investigation, setting forth to the extent applicable the reasons why the creditor believes the account of the obligor was correctly shown in the statement and, upon request of the obligor, provide copies of documentary evidence of the obligor's indebtedness. In the case of a billing error where the obligor alleges that the creditor's billing statement reflects goods not delivered to the obligor or his designee in accordance with the agreement made at the time of the transaction, a creditor may not construe such amount to be correctly shown unless he determines that such goods were actually delivered, mailed, or otherwise sent to the obligor and provides the obligor with a statement of such determination.

After complying with the provisions of this subsection with respect to an alleged billing error, a creditor has no further responsibility under this section if the obligor continues to make substantially the same allegation with respect to such error.

(b) Billing error

For the purpose of this section, a "billing error" consists of any of the following:

(1) A reflection on a statement of an extension of credit which was not made to the obligor or, if made, was not in the amount reflected on such statement.
(2) A reflection on a statement of an extension of credit for which the obligor requests additional clarification including documentary evidence thereof.
(3) A reflection on a statement of goods or services not accepted by the obligor or his designee or not delivered to the obligor or his designee in accordance with the agreement made at the time of a transaction.
(4) The creditor's failure to reflect properly on a statement a payment made by the obligor or a credit issued to the obligor.
(5) A computation error or similar error of an accounting nature of the creditor on a statement.
(6) Failure to transmit the statement required under section 1637(b) of this title to the last address of the obligor which has been disclosed to the creditor, unless that address was furnished less than twenty days before the end of the billing cycle for which the statement is required.
(7) Any other error described in regulations of the Bureau.

(c) Action by creditor to collect amount or any part thereof regarded by obligor to be a billing error

For the purposes of this section, "action to collect the amount, or any part thereof, indicated by an obligor under paragraph (2)" does not include the sending of statements of account, which may include finance charges on amounts in dispute, to the obligor following written notice from the obligor as specified under subsection (a), if—

(1) the obligor's account is not restricted or closed because of the failure of the obligor to pay the amount indicated under paragraph (2) of subsection (a), and
(2) the creditor indicates the payment of such amount is not required pending the creditor's compliance with this section.

Nothing in this section shall be construed to prohibit any action by a creditor to collect any amount which has not been indicated by the obligor to contain a billing error.

(d) Restricting or closing by creditor of account regarded by obligor to contain a billing error

Pursuant to regulations of the Bureau, a creditor operating an open end consumer credit plan may not, prior to the sending of the written explanation or clarification required under paragraph (B)(ii), restrict or close an account with respect to which the obligor has indicated pursuant to subsection (a) that he believes such account to contain a billing error solely because of the obligor's failure to pay the amount indicated to be in error. Nothing in this subsection shall be deemed to prohibit a creditor from applying against the credit limit on the obligor's account the amount indicated to be in error.

(e) Effect of noncompliance with requirements by creditor

Any creditor who fails to comply with the requirements of this section or section 1666a of this title forfeits any right to collect from the obligor the amount indicated by the obligor under paragraph (2) of subsection (a) of this section, and any finance charges thereon, except that the amount required to be forfeited under this subsection may not exceed $50.

(Pub. L. 90–321, title I, §161, as added Pub. L. 93–495, title III, §306, Oct. 28, 1974, 88 Stat. 1512; amended Pub. L. 96–221, title VI §§613(g), 620, Mar. 31, 1980, 94 Stat. 177, 184; Pub. L. 111–203, title X, §§1087, 1100A(2), July 21, 2010, 124 Stat. 2086, 2107.)

EDITORIAL NOTES

CODIFICATION

Pub L. 111–203, §1100A(2), which directed the substitution of "Bureau" for "Board" wherever appearing in title I of Pub. L. 90–321, was executed to this section, which is section 161 of title I of Pub. L. 90–321. Section 1087 of Pub. L. 111–203, which directed the making of an identical amendment in title III of Pub. L. 93–495, which added this section to title I of Pub. L. 90–321, has not been executed.

US, EU laws, the terms and service of VISA and MasterCard, and several regulatory bodies have stated in no uncertain terms that payment card fraud is not the liability of the consumer and those transactions must be reversible. A large portion of ISO8583 messages are related to transaction reversals.

Blockchains are not reversible by design, chargebacks were viewed as a design flaw Satoshi attempted to address.

1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers from fraud, and routine escrow mechanisms could easily be implemented to protect buyers. In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes.

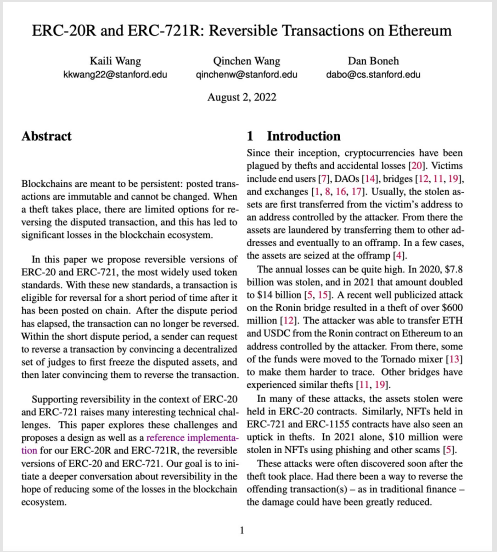
Reversals and their effects compose nearly the entire intro of the original Bitcoin white paper. Chargebacks introduce uncertainty into the payment system and require a centralized entity to resolve. Decentralization and finality are key components of blockchains from their inception.

Luckily, reversibility is an area of active research and Stanford has just released its reversable ERC20R smart contracts. Though, these implementations are still on the untested cutting edge of technology.

The idea of reversable transactions on the blockchain isn't a new idea and has been thrown around for years.

Reversible ERC-20 or ERC-721 transactions are still at the cutting edge of blockchain technology. While seemingly simple, reversals actually introduce several significant engineering and economic challenges. Most of these challenges are still unsolved in decentralized trustless networks.

ERC20 implementations with reversal functions have recently been outlined in the latest research by Stanford.



ERC-20R and ERC-721R: Reversible Transactions on Ethereum

Kaili Wang kkwang22@stanford.edu
Qinchen Wang qinchenw@stanford.edu
Dan Boneh dabo@cs.stanford.edu

August 2, 2022

Abstract

Blockchains are meant to be persistent: posted transactions are immutable and cannot be changed. When a theft takes place, there are limited options for reversing the disputed transaction, and this has led to significant losses in the blockchain ecosystem.

In this paper we propose reversible versions of ERC-20 and ERC-721, the most widely used token standards. With these new standards, a transaction is eligible for reversal for a short period of time after it has been posted on chain. After the dispute period has elapsed, the transaction can no longer be reversed. Within the short dispute period, a sender can request to reverse a transaction by convincing a decentralized set of judges to first freeze the disputed assets, and then later convincing them to reverse the transaction.

Supporting reversibility in the context of ERC-20 and ERC-721 raises many interesting technical challenges. This paper explores these challenges and proposes a design as well as a reference implementation for our ERC-20R and ERC-721R, the reversible versions of ERC-20 and ERC-721. Our goal is to initiate a deeper conversation about reversibility in the hope of reducing some of the losses in the blockchain ecosystem.

1 Introduction

Since their inception, cryptocurrencies have been plagued by thefts and accidental losses [20]. Victims include end users [7], DAOs [14], bridges [12, 11, 19], and exchanges [1, 8, 16, 17]. Usually, the stolen assets are first transferred from the victim's address to an address controlled by the attacker. From there the assets are laundered by transferring them to other addresses and eventually to an offramp. In a few cases, the assets are seized at the offramp [4].

The annual losses can be quite high. In 2020, $7.8 billion was stolen, and in 2021 that amount doubled to $14 billion [5, 15]. A recent well publicized attack on the Ronin bridge resulted in a theft of over $600 million [12]. The attacker was able to transfer ETH and USDC from the Ronin contract on Ethereum to an address controlled by the attacker. From there, some of the funds were moved to the Tornado mixer [13] to make them harder to trace. Other bridges have experienced similar thefts [11, 19].

In many of these attacks, the assets stolen were held in ERC-20 contracts. Similarly, NFTs held in ERC-721 and ERC-1155 contracts have also seen an uptick in thefts. In 2021 alone, $10 million were stolen in NFTs using phishing and other scams [5].

These attacks were often discovered soon after the theft took place. Had there been a way to reverse the offending transaction(s) – as in traditional finance – the damage could have been greatly reduced.

1

ERC-20R and ERC-721R: Reversible Transactions on Ethereum

Published August 2, 2022 by Kaili Wang, Qinchen Wang, and Dan Boneh.

https://arxiv.org/pdf/2208.00543.pdf

This proposed implementation from Stanford is largely in response to the recent large-scale bridge hacks, and has been designed to facilitate reversal disputes through a decentralized panel of randomly chosen judges. If the judges approve, the transaction(s) are reversed through a transaction. With a payment card reversal, the authority is relatively simplified, but the economics of reversing largely legitimate transactions down chain of the transaction is far more complicated.

There are a few key challenges a reversible ERC20 token must address. The Stanford implementation addressed authority and downstream impacts, but not capital efficiency.

Downstream Impacts

Reversing a transaction up to 90 days later has significant logistics challenges if those funds have been spent in the general transaction pool. Generally, funds can be pulled from an account as long as there is sufficient balance, however if that account has then spent the funds in legitimate transactions reversing those downstream has a strong negative impact on everyday users.

Authority

As fraudulent users, honest users, merchants and banks all have conflicts of interest and unaligned incentives, there is an outstanding challenge as to who holds the ultimate authority to reverse transactions.

Capital Efficiency

As any account can hold ERC20 tokens, one possible attack vector is to have a new fraudulent account that immediately cashes out into a non-reversable token. Some solutions might be requiring users to post collateral or a holding period before which moving tokens out of the smart contract is not possible, however both solutions are very capital inefficient.

Traditional Financial Market Solutions

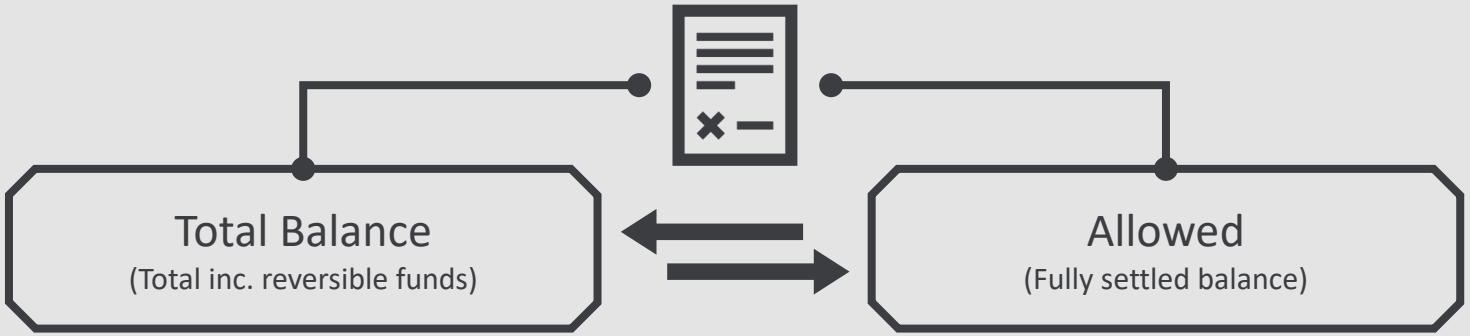
As transactions are not settled on a 1:1 basis, but rather aggregated into a nightly settlement, any reversals are also aggregated and the account holder is required to keep a sufficient balance in their account to cover any potential debits and are liable for 100% of the reversal regardless of any further transaction they have made.

In traditional finance, the network itself, such as Visa and Mastercard act as the ultimate arbiters and retain the ultimate authority. They coordinate a complex fraud dispute process that allows the merchant, issuer bank, and cardholder to identify and reverse fraudulent transactions.

Traditional finance has the benefit of known identities which allows firms to operate with a high level of trust without posting collateral or requiring lockup periods.

The custodied stablecoin smart contract that underpins the system on the blockchain is proposed at this stage to be an ERC20R contract modified to include oracle based approvals.

As a quick recap, an ERC20 token has two main databases for balances and permitted allowance. We extend this with just one function for reversals.



ERC20 Standard Functions

+

ERC20R Extended Functionality (optional)

function totalSupply() public view returns (uint256);
function balanceOf(address tokenOwner) public view returns (uint);
function allowance(address tokenOwner, address spender) public view returns (uint);
function transfer(address to, uint tokens) public returns (bool);
function approve(address spender, uint tokens) public returns (bool);
function transferFrom(address from, address to, uint tokens) public returns (bool);

function reversal(address from, address to, uint tokens) public returns (bool); (New Function)

Here we've placed reversals as their own function that extends the standard, a reversal is similar to a traditional transfer except it can withdraw from the total balance, not just the allowed portion, and it relies on authentication and approval by the oracle, not based on the users ownership of the private key.

ERC20 is a powerful standard and adapts well to our use case.

Oracle Based Approve Function:

In addition to the traditional private key/public key to transfer allowed balances, we also code a separate approval function that allows the oracle to approve reversals.

Balances/Allowance:

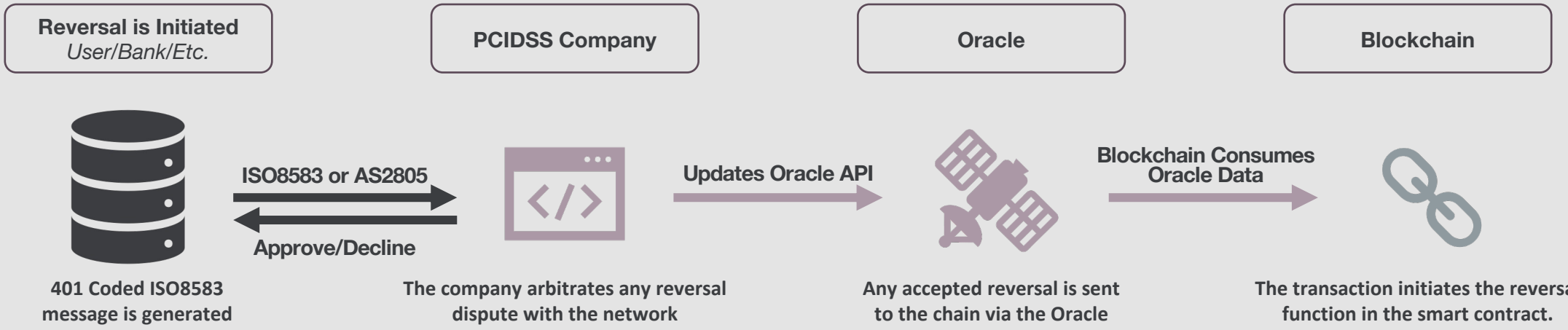
Users are able to freely transfer their allowed balances while the smart contracts keeps potentially reversible transactions sequestered until they clear the dispute window.

Reversal Function:

To create the ERC20R standard, Stanford researchers extended the ERC20 standard with a reversal function. We can use this in order to execute any reversal ISO8583 message the network receives.

An oracle based solution simplifies reversal rights in the traditional financial way, though at the cost of significant centralization.

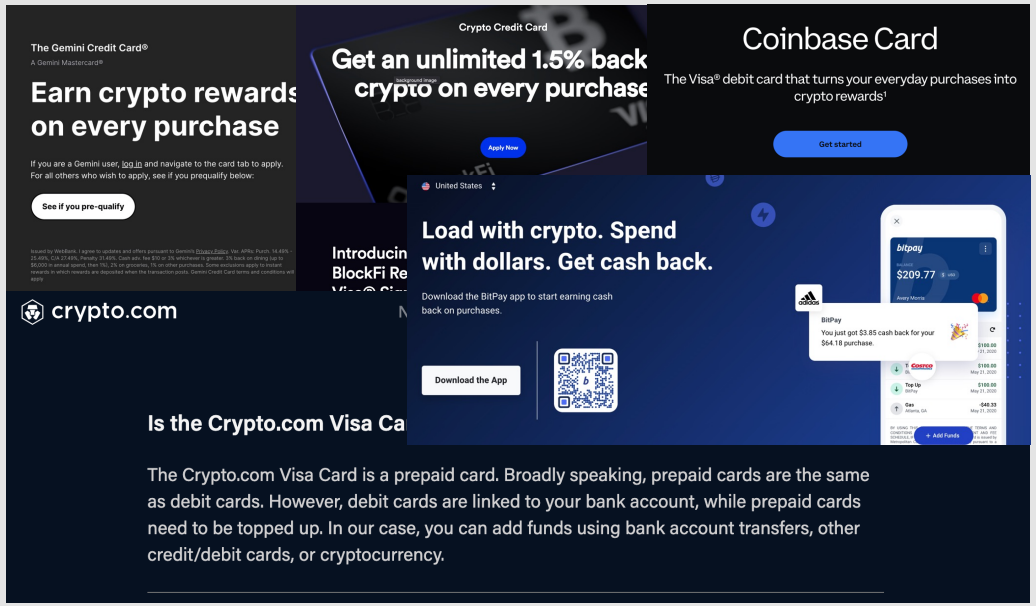
Reversals can be routed through the oracle to the smart contract for reconciliation on the blockchain.



We can centralize and process reversals by using the centralized company as the final arbiter of the dispute. Any accepted reversal is forwarded by the oracle to the blockchain for reconciliation where the smart contract calls it's reversal function.

There are many advantages to this architecture over existing commercial solutions. Today, users largely have to choose between the traditional payment cards or blockchain transactions.

Today's commercially available "crypto" payment cards give users no control over the assets on chain.



US, EU laws, the terms and service of VISA and MasterCard, and several regulatory bodies have stated in no uncertain terms that payment card fraud is not the liability of the consumer and those transactions must be reversible. A large portion of ISO8583 messages are related to transaction reversals.

Meanwhile, blockchain native payments are not widely supported and requires specialized infrastructure.

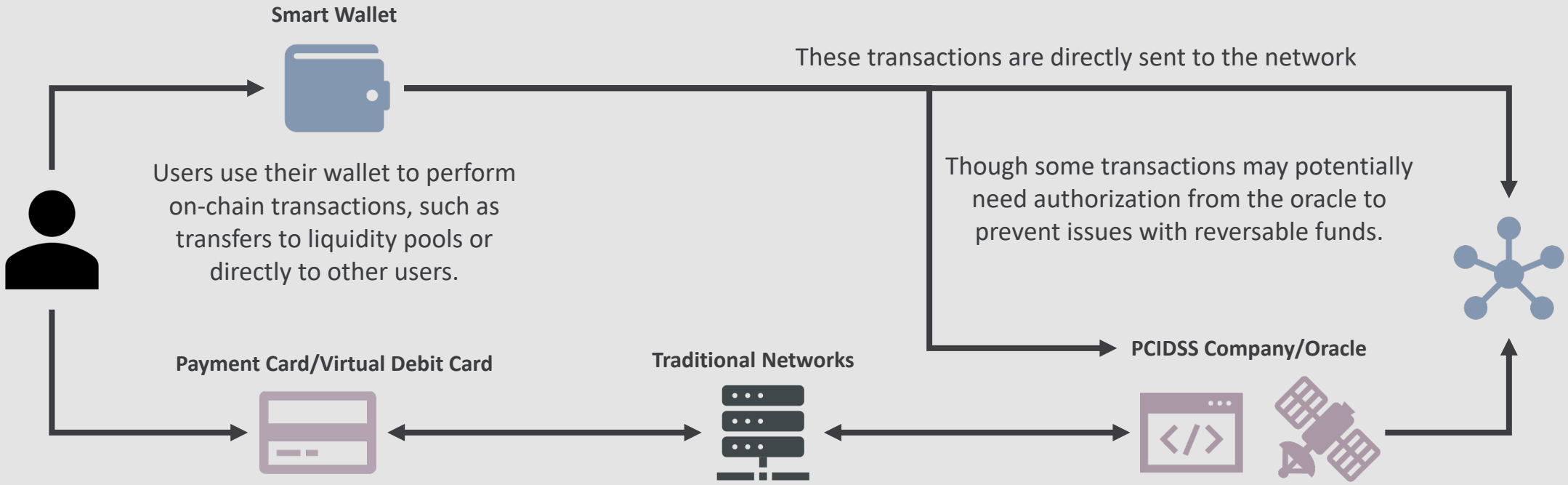
There is very strong demand for crypto payments by both merchants and customers, leading to several crypto payment providers. These providers have no connection with traditional financial networks and must be manually integrated each time and require a user to have a smart wallet.

For readers curious as to how and why merchants are gearing up to accept crypto and how much they are investing in the space, we'd recommend reading this public report by Deloitte.

https://www2.deloitte.com/content/dam/Deloitte/us/Documents/technology/us-cons-merchant-getting-ready-for-crypto.pdf

The proposed system allows users to utilize the full breadth of functionality of the blockchain as well as payment card infrastructure for everyday payments.

In comparison to today's commercial crypto card products, users can take full advantage of DeFi while still using their accounts and payment cards for existing merchants.



It also improves on crypto payment gateways by simplifying the user experience, automatically increasing merchant adoption, and allowing for increased protection for both merchants and users.

An oracle based architecture improves on many of the key weaknesses of modern competing crypto payment solutions such as Coinbase Commerce, Coingate, Bitpay and others.

User Experience

The user experience for payments is dramatically simplified and seem less risky, even for users with zero exposure to crypto, as users transact in the exact manner they are already familiar with. All of the major changes related to transaction routing are behind the scenes.

Merchant Adoption

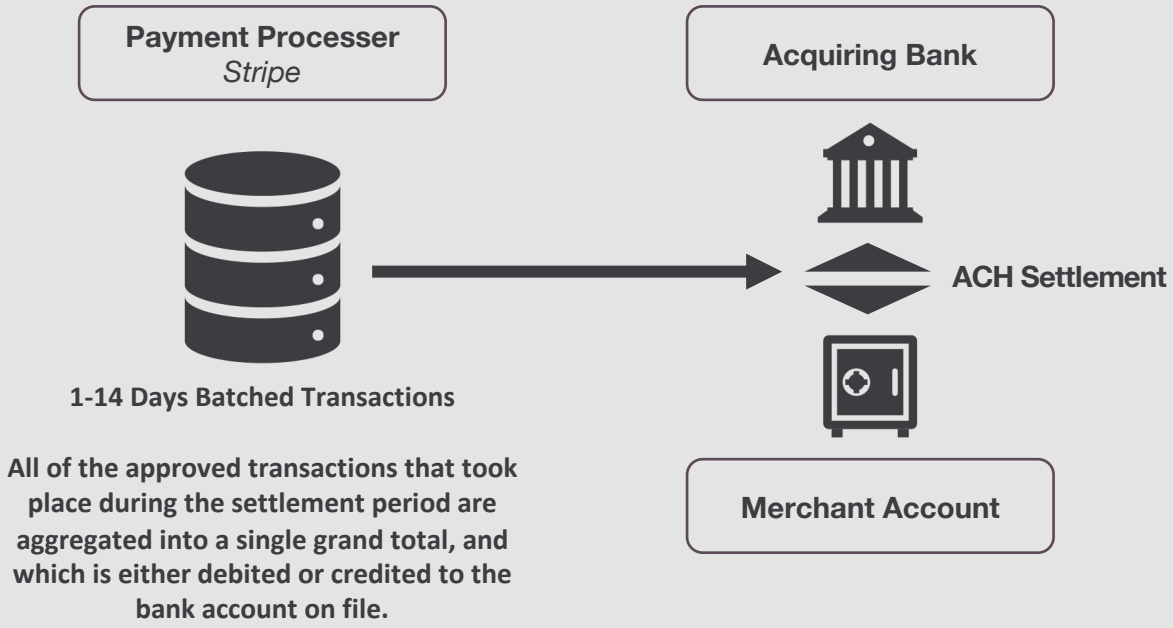
Integrating with payment card networks means merchants no longer need to worry about setting up integrating new crypto payment gateways with their front end store or accounting software. Furthermore, the dispute process for fraud is familiar and routine.

Security

This architecture offers consumer protection in line with traditional payment methods. Furthermore, the oracle controlling company can authorize transaction in real-time allowing merchants to have more certainty around payments instead of competing solutions which require confirmation times of several minutes to hours.

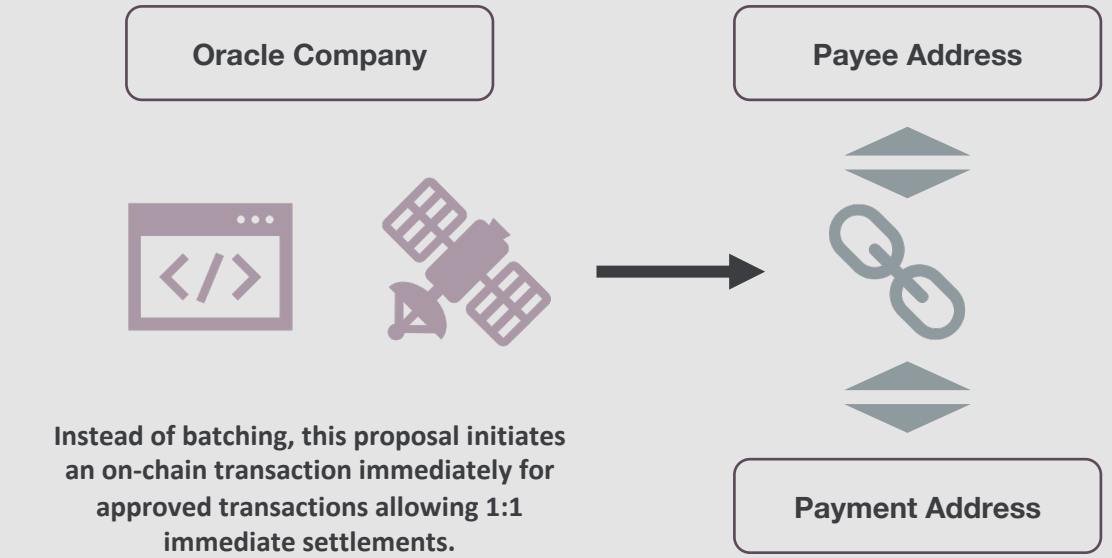
In the future, blockchain based settlement might also allow for immediately settlement on a 1:1 basis instead of merchants and banks waiting for a weekly or, at best, nightly batch process.

Traditionally approved ISO8583 messages are batched and sent for settlement in a single aggregated transaction.



All of the approved transactions that took place during the settlement period are aggregated into a single grand total, and which is either debited or credited to the bank account on file.

With a 24/7 blockchain, this settlement process would be 1:1 for transactions and immediate.



Instead of batching, this proposal initiates an on-chain transaction immediately for approved transactions allowing 1:1 immediate settlements.

The exact process varies by payment gateway, but settlement periods range from overnight to a week. The general processes is similar in that all the transactions are batched into a single grand total that is then settled at an aggregate level as a single lump sum.

On the ledger, transactions are relatively low cost compared to 1960s settlement technology which allows us to perform 1:1 immediate settlements instead of batching the transactions to reduce network demand. Though depending on network capacity, we can also batch transactions to reduce the fees, similar to a layer 2 solution.

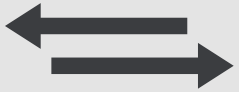
In conclusion, we are advocating a centralized oracle based solution, as it seems to offer the best solutions to perform reversals, keep information secure, and recover in case of an authority breach.

Information Security

Data is securely accepted by and stored behind a centralized PCIDSS compliant corporation. This corporation initiates transactions through a public Oracle. This prevents any ISO8583 messages from ever being in advertently placed on the blockchain and allows us to comply with local state and country regulations.

Authority

The reason we use a centralized oracle over a simple master account in the smart contract is to establish authority with recoverable access control. If the worst case scenario happens and the centralized oracle's domain is hacked, the domain owner can easily recover control of the account from the domain name provider.

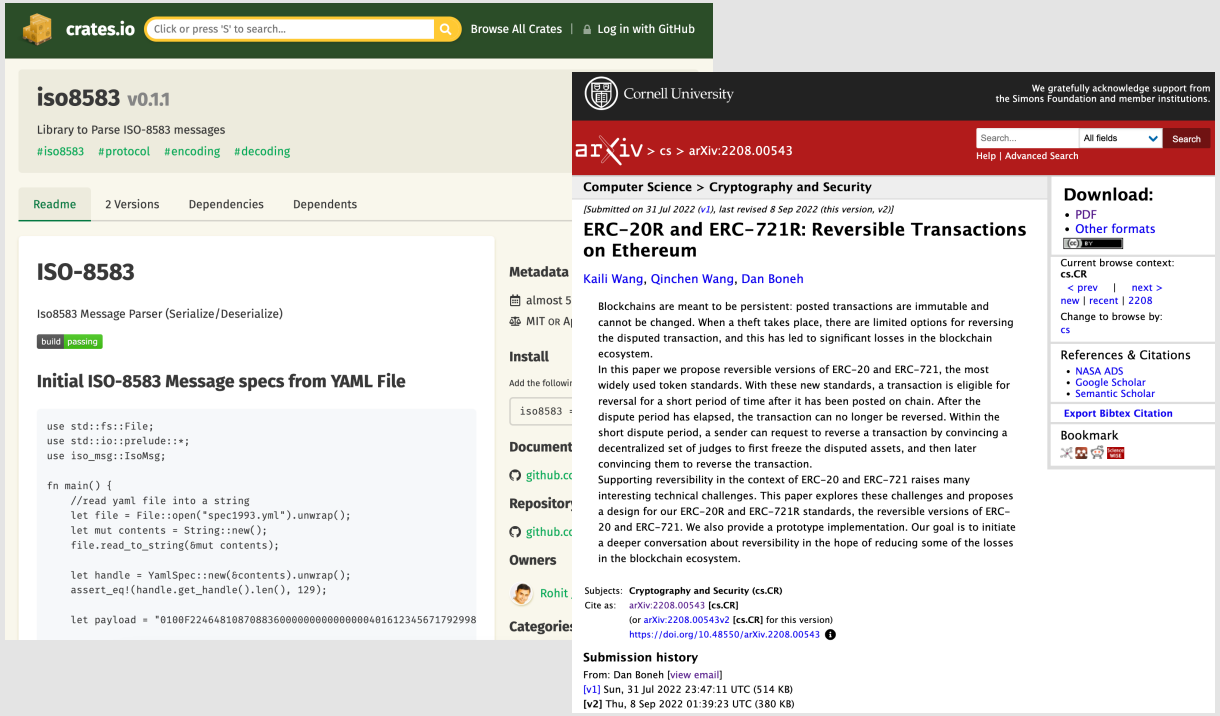


Reversals

A specialized smart contracts extends the ERC20 standard to support reverse transactions based on the input of the centralized oracle. The smart contract also handles tracking and segregating reversable funds from a user's liquid balances.

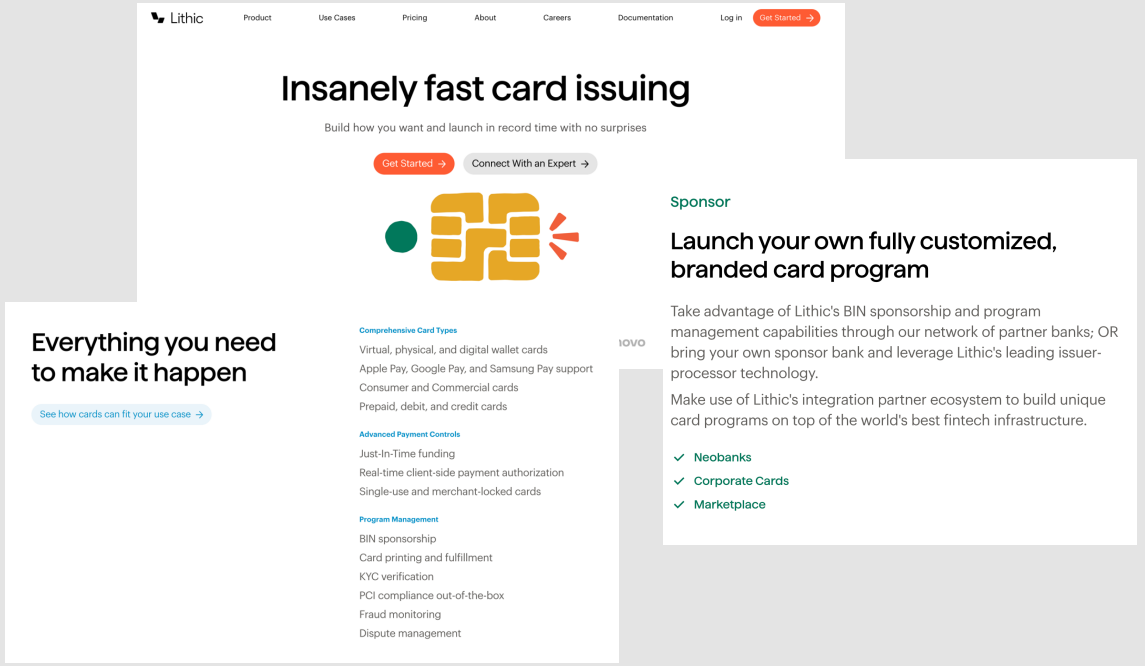
The technical build is straightforward compared to the financial infrastructure. However, the complexity can be reduced by leveraging open source libraries and partners.

The technical implantation is relatively straightforward and can leverage several open source libraries.



The next steps would be to build a prototype of the oracle and smart contract and deploy them to a substrate based testnet. The smart contract would adopt the open source ERC20R standard as a start, while the oracle can leverage open source ISO8583 Rust crates to speed development.

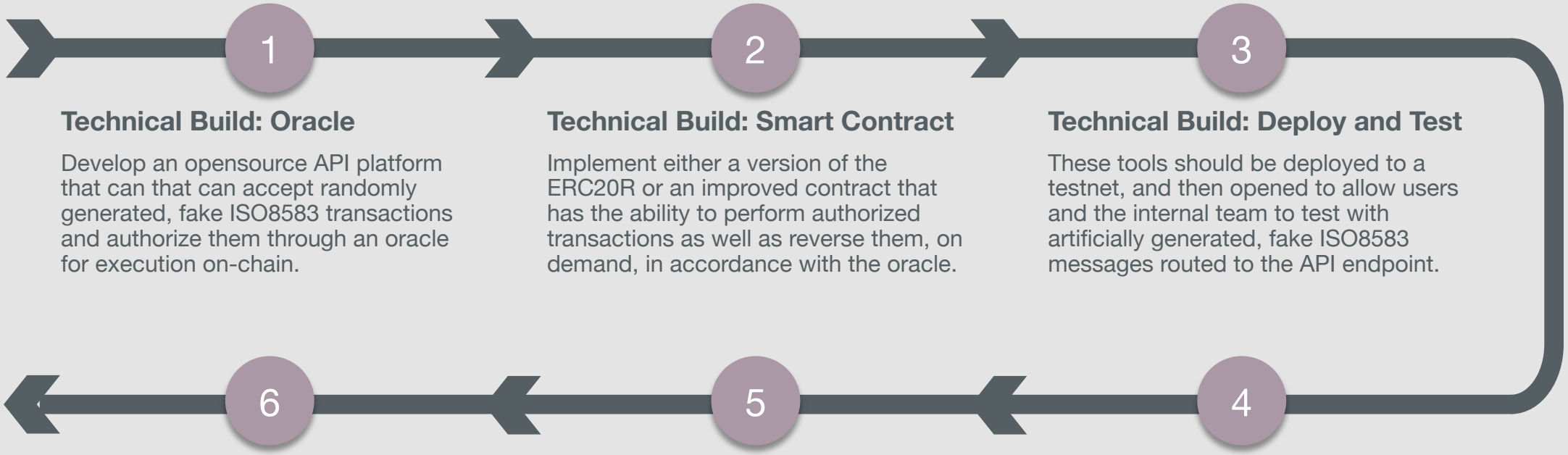
Setting up the financial infrastructure is the more difficult challenge, but has been simplified thanks to modern tools.



While at first glance it might seem intimidating to set up financial infrastructure, the very recently launched banking-as-a-service infrastructure companies can dramatically simplify the process. Lithic, for example, provides payment card issuance services primarily for “neobanks”, however these tools are the exact ones we require and can easily be repurposed for our purposes, dramatically reducing the build out required to reach MVP.

The recommended next step would be to prove the technical feasibility by constructing an oracle that can accept ISO8583 messages and push them to a smart contract on a testnet.

We recommend tackling the technical implantation first, then using banking-as-a service firms to issue a very small number of physical plastics that integrate directly with the oracle and testnet as a very limited initial test.



1 Technical Build: Oracle

Develop an opensource API platform that can that can accept randomly generated, fake ISO8583 transactions and authorize them through an oracle for execution on-chain.

2 Technical Build: Smart Contract

Implement either a version of the ERC20R or an improved contract that has the ability to perform authorized transactions as well as reverse them, on demand, in accordance with the oracle.

3 Technical Build: Deploy and Test

These tools should be deployed to a testnet, and then opened to allow users and the internal team to test with artificially generated, fake ISO8583 messages routed to the API endpoint.

4 Infrastructure: Issue the First Few Test Plastics

The next test would be to build and test infrastructure connections to that API endpoint. The fastest way, would be to partner with a banking-as-a-service provider that issue debit cards and route messages while serving as a BIN sponsor.

5 Infrastructure: Leverage Fast Track Programs

Leverage programs like VISA's Fintech Fast Track to quickly develop the ability to issue cards with greater control. While BaaS will work well for a very limited card test, it is unlikely to suffice for commercial blockchain applications.

6 Infrastructure: 2nd Alpha Test

Move towards another test in partnership with a payment network. The aim is to deliver cards issued by the holding company's BIN number that reflect a real balance of custodied stablecoins and that are cleared through the payment networks' settlement layer overnight (e.g. VisaNet).