Reviews and Ratings Verified by Payments on Blockchains

Chlu

March 1, 2018

Abstract

We present a decentralised payments and reputation system that recognizes vendors for delivering goods and services. Payments made are accompanied by ratings and reviews and these can be verified by anyone the vendor grants permission to do so — in all other cases the reputation and payments remain private to the vendor. All ratings and reviews claimed by the vendor can be verified by cross checking data published on an immutable ledger.

All popular online marketplaces use closed reputation systems, where users can not take their reputation and sales history to another marketplace, if they want to. Chlu is an open, decentralised and freely accessible reputation platform built on top of public permissionless blockchain. Vendors and customers do not need to put their trust in any single entity. Anyone can use the Chlu protocols to make payments, track reputations and verify these reputations once they have the permission from the vendor to do so.

Chlu provides protocols and open source libraries for marketplaces and online shops to integrate their payments and reputation systems with Chlu. The only fees paid are paid to blockchain miners. Vendors on Chlu own their reputations and use them across any number of marketplaces, they are not confined to any single marketplace.

1 Introduction

Marketplaces need reputation to thrive. In real life, and even more so in online ecosystems the only way for people to do business is to look at each other's past behaviour and use that as a predictor of their future behaviour. All popular online marketplaces like eBay, Airbnb, Amazon marketplace and UpWork include a reputation system that provides trust in the vendors and in some cases customers too. However, these marketplaces keep the users' reputations locked to their platforms, denying users the chance to benefit from reputation they built. With immutable ledger systems provided by blockchains, users' reputations can be saved on publicly accessible systems while being kept private using advanced cryptographic solutions.

It is also important that a vendor's reputation is not publicly available if a vendor chooses so. Instead, a vendor can choose to reveal their reputation only to a select customer or on a marketplace. Vendors can also select marketplaces that can always display their reputations.

Alongside the reputation system, international payments remain a challenge. Bank transfers can take weeks and can cost as much as 3% in transaction fees. Blockchains, with their immutable ledgers, have enabled payment platforms where transfers are near instant with very low fees. Converting a cryptocurrency to fiat can be as low as 0.26% fee.

Chlu solves the above two problems thanks to blockchains. Using Chlu, customers can pay vendors within minutes and also record ratings and reviews associated with the transaction. These ratings, reviews and payments are saved on the public blockchain. Finally, vendors can keep their reputations private, such that until they choose to reveal it to a third party no one can find out a vendor's reputation. This allows for small vendors to preserve their privacy but reveal it to potential customers as and when needed.

Chlu provides a trustless and verifiable reputation and payments system that runs on a public permissionless blockchain. Chlu can be used by any marketplace to let users build their reputation backed by payments received, and let the vendors easily use the same reputation on multiple marketplaces. As vendors receive payments on any of the marketplaces, the changes to their reputations are visible across all other marketplaces as well.

Chlu provides a set of protocols that can be implemented by anyone to enable Chlu into their system. However, we provide reference implementations for these protocols as vendor and customer

wallets as well as libraries for marketplaces to use on their servers. Our goal is to drive the growth of an ecosystem around an open, trustless reputation system.

In the rest of this whitepaper, we demonstrate how important online reputation is in various industries and elaborate on the problems in international payments. We then describe how Chlu solves the problem and how the solution is possible only because of open ledgers provided by blockchains.

2 Reputation in Online Marketplaces

The value of reputation is now a well-established concept when it comes to eCommerce. The amount of data that we have from companies like Google, Amazon, eBay and others means that the evidence is irrefutable. Reviews from our peers are central to how we make purchasing decisions; this has been clearly established in the literature[1].

The numbers don't lie when trying to establish the scale of this opportunity. By 2020, global eCommerce will be a 4 trillion dollar industry[2]. Within that figure, somewhere around sixty-one[3] percent of consumers read reviews before deciding whether or not to buy. It could therefore be argued that by the end of the decade, well over two trillion dollars of eCommerce transactions will be guided by the current mish-mash of online reviews.

Reputation has been at the center of how we make socioeconomic decisions for millennia. However, we now live in an era where total strangers who could never attain information about each others' reputation are nonetheless interacting with each other and engaging in transactions on a daily basis. This is now commonplace, and yet we don't really have a good model for establishing the trust needed to have friction free engagement.

Some marketplaces have been very successful at establishing a proxy for this trust that over time establishes reputation; Amazon, eBay and others have created somewhat effective walled garden solutions to this problem with their reviews and ratings systems. We use the term 'walled garden' as the solutions are locked in a closed, managed ecosystem that is non transferable externally. In other words, your reputation on Amazon is useless on eBay. In an ideal world, this would not be the case.

Further distancing the current state of affairs from an ideal model is the fact that reputation is an umbrella term that doesn't always mean the same thing from marketplace to marketplace. A given user's excellent reputation for selling on Amazon may not be a good indicator of their reputation as an Airbnb user. Even within similar ecosystems, such as freelancer marketplaces, users can quickly become locked into a single marketplace due to the sunk cost fallacy. Rachel Botsman wrote in 2012 for Wired magazine:

"An aggregated online reputation having a real-world value holds enormous potential for sectors where trust is fractured: banking; e-commerce, where value is exponentially increased by knowing who someone really is; peer-to-peer marketplaces, where a high degree of trust is required between strangers; and where a traditional approach based on disjointed information sources is currently inefficient, such as recruiting"

The need for a universal, open, free-to-use reputation score is plain. Many businesses have tackled this problem over the last five years[4], and none have truly succeeded. Legit, TrustCloud, Confido, Scaffold, Reputate are some of the names on corporate headstones in the reputation economy graveyard.

Looking at some of the current players in this space, from established players like TrustPilot to new entrants like Feefo, and everyone in between (TrustedShops, Reevoo, and of course even Amazon themselves) the problem is evident: the current systems are prone to being gamed. Even a solution like TrustedShops, where 'proof of purchase' is necessary to leave a review, the reality is that ownership of the data in the purchasing chain is broken several times along the way, and fake reviews abound. Google uses a long list of independent vendors[5] that provide inputs to establish their seller ratings, and still it appears that approximately 15[6] to perhaps 35[7] percent of these are

[1] https://econsultancy.com/blog/9366-ecommerce-consumer-reviews-why-you-need-them-and-how-to-use-them
[2] https://www.emarketer.com/Article/Worldwide-Retail-Ecommerce-Sales-Will-Reach-1915-Trillion-This-Year/1014369
[3] https://econsultancy.com/blog/9366-ecommerce-consumer-reviews-why-you-need-them-and-how-to-use-them/
[4] http://www.wired.co.uk/article/welcome-to-the-new-reputation-economy
[5] https://support.google.com/partners/answer/2375474
[6] http://www.gartner.com/newsroom/id/2161315
[7] https://www.cs.uic.edu/~liub/publications/WWW-2012-group-spam-camera-final.pdf

still fake. The importance of these ratings cannot be understated: going from a three-star rating to a five-star rating gets a business twenty-five percent more clicks from Google Local Pack[8].

Ninety five percent of users doubt the validity of reviews when they don't see bad scores, and yet Google only shows scores of three-and-a-half stars or more. Something is broken; customers know it, and have to resort to personal, highly imperfect hacks to sort the real review wheat from the fake review chaff.

2.1 So why will Chlu be any different? What has changed?

The difference today is the emergence of blockchains providing distributed ledgers, where:

- data is stored in a public immutable database; and
- payments are publicly visible.

These two key features provide us with tools where trust can be established in a much more robust manner.

Customers trust other real customers, but existing systems are fraught with the problem of fake reviews. Using the two essential attributes of blockchains we can eliminate fake reviews. Customers can only leave a review and a rating for a vendor if they have made a purchase from that vendor. Equally importantly, we provide the infrastructure where payments have to be supported by publicly verifiable requests for payments from the vendor.

Furthermore, blockchain based payment backed reviews establish a clear financial inhibitor against fake reviews, and the more the network grows, the larger this inhibitor becomes.

There can be zero doubt that a truly global, open, free and trustless reputation system, where you control access to your reputation, in what context that access is granted, with no third party in the middle, would revolutionize eCommerce on a global scale. We believe that the solution that Chlu provides is one of the key components necessary to take online trade from its current phase into the coming century where it will inevitably eclipse traditional commerce.

3 Problems with Online Payments

In this section we describe the current state of the online payments market, and why there is a need for a better solution.

3.1 Current Solutions

As mentioned above, more and more financial transactions are moving online with each passing year. E-Commerce in general continues to grow at double digit rates, and mobile payments are just starting to take hold in mainstream transactions. You can more than likely pay for your groceries with your mobile device in your local supermarket today. In the US, mobile payments rose thirty nine percent in 2016 to reach $112bn, according to global firm Forrester Research. The Chinese market dwarfs these numbers by a factor of almost fifty; third party mobile payments in China were valued at approximately $5.5 trillion last year, which was a turning point for physical retailers accepting mobile payments in supermarkets, clothing stores and restaurants leading the way.

Even though there is significant competition in the payments space, established players, national and international regulators, and other incumbent factors mean that innovation in payments is uncommon and generally lags the pace of innovation in technology and the overall online marketplace. This is especially true for international payments, where cross border regulation and processes are cumbersome, costly and ineffective.

Making an international transfer is far from a frictionless process. The number of steps involved is problematic, often reaching double digits in both the number of parties involved and the number of associated fees. This quickly becomes expensive to users of the system while adding virtually no value other than facilitating the transaction itself.

[8] https://www.entrepreneur.com/article/295233

3.2 Fees

"The current industry standard allows for opaque pricing, which obscures unfavourable currency conversion rates, hides the real cost to the consumer and makes it almost impossible for the consumer to 'comparison-shop'" (Consumers International)

Transaction fees are typically between $2 - 6\%$ even for intra-bank transfers, and flat fees are common for international transfers no matter the transaction size. Typical ranges are between $35 to $65 per transfer for international outgoing wire transfers initiated online. Initiating a transfer by other means (e.g. over the phone) will incur further charges on top of this. This pricing may seem exorbitant, but in reality it is necessary to provide a healthy profit on top of the long list of fees imposed by the many parties involved along the way.

The full range of fees that can be applied to standard transactions is eye-opening: basic transaction fees, chargeback fees, retrieval request fees, terminal fees, Payment Card Industry fees (of which there are many), anti-money laundering fees, address verification service fees, and many, many more. Ultimately, the accumulation of these hidden fees mean that a given bank may charge anything from four percent to an incredible fifteen percent for sending money across international borders.

3.3 Transaction Times

Transaction times vary depending again on the type of transfer. For local transfers, anything from two to five days is typical, and for international transfers it can take several weeks to receive funds at the destination account. Even for in-country transfers, some eCommerce payment processors will hold merchants' money for several additional days due to the greater possibility of chargebacks. This of course causes real cash flow problems, especially for small businesses.

Of course, there are well known alternatives should speed be of the essence, but they come at a cost. Users of a service such as Western Union should expect to pay somewhere between seven to twelve percent of the total transaction value, making it a very expensive option.

Finally, most current solutions involve trusting a third party to an extent that adds significant risk to the users of the system. Paypal is a perfect example of a major payment solution provider that has repeatedly broken trust with many of its users, locking account access for seemingly arbitrary reasons, and effectively seizing funds with no arbitration or review process in place. They have often left businesses and individuals without access to their funds for six months or more with zero recourse, and zero oversight. Again, the current systems are broken, and the world needs a more effective solution.

4 Chlu

In this section we describe the goals of the Chlu protocols and how we are enabling an ecosystem where anyone can provide a payments and reputation on their platform. We also explain new ideas about Proof of Payment (`PoP`) and Proof of Payment Requested (`PoPR`).

4.1 Goals

Chlu is a decentralised platform for making payments and storing vendor reputations. Chlu enables vendors to build their reputation on a decentralised blockchain so they are not confined to a walled garden of a single marketplace, and customers benefit from a transparent reputation of vendors backed by payments made for goods and services. The key goals of a decentralised reputation platform are to provide:

Proof of Payment Customers can leave reviews and ratings only if they made a payment to the vendor.

Proof of Payment Requested Every payment made can be verified to be in response to a payment request by the vendor.

Zero Fees Chlu does not charge any fees. The only fees in the system are those charged by blockchain miners.

Private Feedback Reviews and ratings are private to the vendor.

All or nothing Vendors can share their reputation with any marketplace, including their own websites.

Multiple Reputation Scores Vendors cannot hide a review. Once they share their reputation with a marketplace their entire history is available to the marketplace.

Extensible Marketplaces are free to compute a reputation score for a vendor using the reputation data available.

Open Access Anyone can use the reputation system as a vendor, customer or marketplace.

4.2 Proof of Payment

All user feedback for a vendor is tied to payments made by the customers leaving the rating and review. This is an important aspect of a trustless decentralised reputation system. *There is no third party involved.* Instead of trusting a payment gateway or the vendors themselves, Chlu ties the rating and review to a payment made through a trustless payment system.

A customer can leave feedback for a vendor only if there is a verifiable payment made by the customer to the vendor. Further still, the customer can leave only one feedback per purchase made. The customer can update this feedback for upto two months after the purchase.

4.3 Proof of Payment Requested

All payments made to a vendor have to be backed by a payment request signed by the vendor or by the marketplace on behalf of the vendor. Chlu protocols require that this proof of payment requested or `PoPR` is verified before a review is included in the vendor's reputation.

The requirement of checking for a `PoPR` avoids an attack by a vendor's competition to leave negative reviews for the vendor. Even if the competitor is willing to pay the price of spamming a vendor, the vendor can put a "flow control" on the reviews received by controlling the inventory on sale.

4.4 Zero Fees

The Chlu protocols are designed so that no third party can directly profit from facilitating a trustless reputation system. That is, there are no fees charged for providing the libraries that together make up Chlu. Even the founders of Chlu can not charge any fees for use of the system.

The only fee charged is that by miners to save the payment on the blockchain.

4.5 Private Feedback

The reviews and ratings left for a vendor are saved after encrypting it with the vendor's public key, so that only the vendor can read the feedback. The vendor can enable sharing of their reviews and ratings with one or more marketplaces as they see fit.

This requirement is important to avoid marketplaces listing vendors as members of the marketplace without permission from the vendor as well as providing privacy to the vendor in case they want to disable any reviews and ratings for their services.

This feature also allows a vendor to leave a marketplace and thus stop sharing their reputation with the marketplace, even though the reputation already shared could have been saved by the marketplace.

4.6 All or Nothing Reputation

A vendor can not selectively hide any of the reviews and ratings received. Once a vendor shares their reputation with a marketplace all the reviews and ratings are made available to the marketplace. The marketplace can detect if the vendor is not revealing any of the reviews and ratings and thus refuse to show the vendor's reputation on the marketplace, or even further, block the member from the marketplace outright.

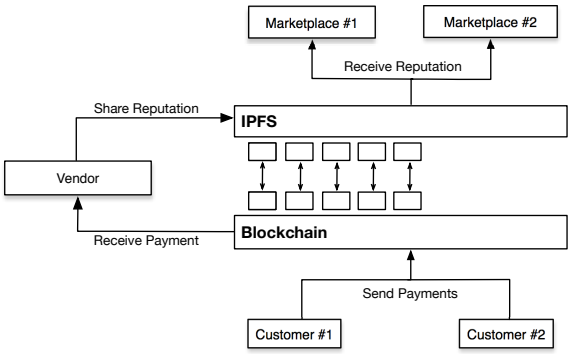

Figure 1: Linking reputation data to payments on blockchain

4.7 Multiple Reputation Scores

Marketplaces are free to compute vendors' reputation scores based on their own algorithms. This is important so that marketplaces can encourage vendor behaviour they choose. For example, some marketplaces want to track a vendor's average ratings over the lifetime of the vendor, while others give additional weight to more recent ratings received.

4.8 Open Access

Anyone can join the system as a customer, vendor or a marketplace. No vetting by third party is required, no minimum balances in a bank account, or any such old world measures. In this truly open and decentralised reputation system, there will be no central party that controls access to the system.

4.9 Extensible

Different marketplaces will need to store details of a review as part of the review created by the customer. For example Airbnb[9] and Homeaway[10], both property rentals marketplaces, ask their users to review the host and the property using different criteria. Chlu supports storing any details a marketplace wants to include in the reviews details by describing an open reviews structure as part of the Chlu protocols.

5 Architecture

The Chlu platform uses a blockchain to make payments and saves the `Proof of Payment` along with a `Proof of Payment Request` in IPFS[2], a decentralised immutable storage for objects and files. The key idea behind Chlu is that we can save a review on IPFS and save the review's IPFS content address in the blockchain transaction.

Once we have a way to find an immutable review associated with a payment, we then use cryptography to make sure of the following:

1. The review is only readable by the vendor
2. The review and the payment associated with it are in response to a payment request created by the vendor or on behalf of the vendor by a marketplace
3. Vendor can't hide any of the reviews they have received

In the following sections we explain in detail how these characteristics are provided for by the Chlu protocols. First, we describe the structure of the payment record first and then describe the IPFS record structure as well.

[9] https://www.airbnb.com/help/article/13/how-do-reviews-work
[10] https://help.homeaway.com/articles/How-do-I-submit-a-review

5.1 Proof of Payment Request, `PoPR`

The `Proof of Payment Request` is stored on IPFS and the IPFS content address for this record is stored with the `Review Record`. The record has the following structure:

I, **Item ID** A unique ID for the item from the marketplace or the vendor's site

In, **Invoice ID** The ID of the invoice generated by the marketplace or the vendor's site

C, **Customer ID** The user ID of the customer on the marketplace if available

T, **Timestamp** A unix timestamp for when the payment request was generated

Cr, **Currency** The currency symbol of the blockchain

A, **Amount** In blockchain currency, e.g. Satoshis

M, **Marketplace** The domain name of the marketplace

K, **Key location** The IPFS content address where the public key to verify PoPR is published. See Section 5.5 for details on how vendor gets this key.

K_v, **Vendor Key location** The IPFS content address where the vendor's public key can be found to validate that the vendor signed the key at location K. See Section 5.5 for details on how vendor gets this key.

K_{ve}, **Vendor Encryption Key location** The IPFS content address where the vendor's encryption key can be found to encrypt the `Review Record`. The key at this location is signed by the vendor's public key at location K_v. See Section 5.5 for details on how vendor gets this key.

S, **Signature** Signature of all of the above, signed by the marketplace's private key for the vendor. Section 5.5 describes how this key should be generated and stored by the marketplace

$$\texttt{Proof of Payment Request}, \texttt{PoPR} = (I, In, C, T, Cr, A, M, K, K_v, K_{ve}, Sig(I, In, C, T, Cr, A, M, K, K_v, K_{ve})).$$

The signature is created using S_{vm}, the vendor's signing key generated by the marketplace m for vendor V.

5.2 Payment Record

The payment record stored on blockchain contains the following details:

V, **Vendor address** The address of the vendor on the blockchain

A, **Amount paid** In the blockchain currency

RR, **Review Record** The IPFS content address where the review is stored

$$\texttt{Payment record}, \texttt{PR} = (V, A, RR)$$

5.3 Review Record stored on IPFS

The reviews stored on IPFS include the following details:

R, **Rating** An integer in the range 1 to 5

Rv, **Review** A plain text review

Dr, **Detailed Review** A JSON formatted review; the structure of the review is defined by the marketplace

$PoPR$, **Proof of Payment Request** This is the request for payment signed by the secret key so that it can be verified by the vendor's public key. See Section 5.5 for the details of when these keys are generated and stored

$Prev$, **Previous Review** This is a hash pointer to the most recent review written by the customer. See Section 5.9 for details on why we need this and how it is used

H, **Hash** : The hash of all of the above, $H(R, Rv, Dr, PoPR)$

`Review Record`, `RR` $= (PoPR, H, E_v(Pve, (R, Rv, Dr)))$

All the elements of the RR are encrypted using the Vendor's public key, P_{ve}. The hash along with the encrypted data is stored on IPFS. The content address of the `Review Record` is stored in the payment transaction.

5.4 Keys

Before we describe the protocols, we identify the keys that need to be generate and distributed by the various participants in the network.

We use IPFS to distribute keys. The idea is that marketplaces generate a root key pair. Later, when vendors signup to the marketplace, it generates a key pair for each vendor, keeps the private key safe, signs the public key for the vendor and sends it to the vendor. At the moment, the signature does not include any other meta data. We describe the details of this scheme in the next section, first need to defined the key pairs that need to be generated and distributed.

S_m **and** P_m — **root signing key for marketplace** m The key pair generated by marketplace when it starts using Chlu. The public key P_m is published in a `.well-known` location and the secret key S_m is used to sign public keys for vendors who register with the marketplace.

S_{vm} **and** P_{vm} — **signing key pair for vendor** v **at marketplace** m When a vendor registers at a marketplace, it creates a key pair for the vendor, keeps the secret key, S_{vm}, safe, signs the public key, P_{vm}, using its own root key, S_m, and sends the public key to the vendor. This public key is later used validate that the marketplace requested a payment on behalf of the vendor.

S_v **and** P_v — **root key for vendor** v The key pair generated by a vendor when it starts using Chlu. The public key P_v is published on IPFS and the secret key S_v is used to cross sign vendor public keys P_{vm} generated by marketplaces. See Section 5.5 for details.

There is yet another key that is required by the vendor to receive reviews that are encrypted so that they remain private to the vendor. The vendor can later decrypt the received reviews and share them with any marketplace.

S_{ve} **and** P_{ve} — **vendor's encryption key pair** The secret key, S_{ve}, is retained by the vendor and the public key, P_{ve}, is signed using S_v and published on IPFS[2]. The vendor is able to control who can view the review details. To share the review details, the vendor can decrypt the reviews and use a marketplace API to send them to the marketplace servers.

5.5 Key Distribution

There are a number of systems proposed in literature that describe a decentralised PKI so that anyone can publish their public keys, associate the keys with their identity and let them be easily discoverable. Some excellent proposals that will solve our problem of key distribution are under development[1, 8, 11, 3, 12, 14]. As of now, only Blockstack is in production use, however a standalone API or specifications are not available yet. We will update the Chlu protocol once a widely adopted decentralised PKI is available. For Chlu to adopt a decentralised PKI, it will be important that it (i) allows any third parties to provide an implementation without requiring a permission from the developers, and (ii) allows that Chlu can work outside their ecosystem

Until decentralised PKIs are available for production use and can be easily deployed by marketplaces, Chlu specifies the use of `.well-known`[15] location for publishing the marketplace root certification keys. The access to the `.well-known` will be made using SSL and therefore our solution depends on the centralised PKI based on Certification Authorities. We have chosen this approach to provide a pragmatic workable solution that can be deployed today. However, as we said earlier, once decentralised PKIs are available in production, Chlu will switch to one with the easiest adoption curve for marketplaces.

Algorithm 1 Root certification key pair generated by Marketplace at "domain"

Marketplace $\mathbb{M}$ generates a root certification key pair (P_m, S_m)
$\mathbb{M}$ safely stores the secret key S_m
$\mathbb{M}$ publishes the public key S_m at `https://domain/.well-known/chlu.txt`

Algorithm 2 Key pair generated by Marketplace for vendor

Marketplace $\mathbb{M}$ generates a key pair (P_{vm}, S_{vm})
$\mathbb{M}$ signs the vendor's public key P_{vm} with its secret key S_m
$\mathbb{M}$ securely sends the signed key P_{vm} to the vendor in response to vendor's registration request
The vendor signs P_{vm} with their root key S_v and publishes it on IPFS
The vendor sends the IPFS content location address of P_{vm} and P_v back to the marketplace

Our approach of using `.well-known` as a source of verifying that the entity in charge of the domain are the same as those who own the root certification key pair. This approach is also being tried in the Keybase project[10].

The Algorithm 2 results in a key pair for vendor where the marketplace has the signing key and the vendor has the verification key stored on IPFS.By signing the vendor marketplace key with the root key of the vendor, the vendor is able to assert ownership of all `PoPR`s signed by the marketplace on behalf of the vendor.

Finally, anyone can verify that the vendor's public key is signed by the marketplace's root certification key that is available at the marketplace's `.well-known` location.

The vendor's public key, P_{vm} — required to verify if the signed payment request is valid — is saved in a publicly accessible location by the vendor's wallet. Early drafts of the Chlu protocol required that this key is saved under `/ipns/vendorname/chlu/keys/pubver/mi/`. By using IPNS the vendor can add new keys under the pubver directory and change the IPNS name to point to the updated Chlu directory under the `vendorname` IPNS entry. However, we have started work on publishing a DID Method[6] for Chlu. Meanwhile, the Chlu protocol is implemented using a bespoke scheme where the key P_{vm} is published on IPFS after signed by both the marketplace and the vendor and the location is included in the `PoPR`. We are also working a IPFS pubsub based discovery mechanism which will be replaced by a DID Method service endpoint instead.

The section shows how the key distribution scheme used by Chlu demonstrates trust between the marketplace and the vendor. In the next section we show how the marketplace or vendor can remove the trust demonstration by removing the respective keys.

5.5.1 Key Revocation

If a marketplace wants to remove its demonstrated trust in a vendor, it can do so by publishing a signature revocation in the same `.well-known` location where it published the root certification public key. This will require a folder structure under the well known directory and we are in the process of drafting a request for registering our well-known URI according to RFC 5785[15].

On the other hand, if vendors want to remove the demonstrated trust in a marketplace, the vendor publishes a revocation request and publishes it again using IPFS pubsub.

The revocation process described above is also a bespoke solution being implemented and will be replaced by our DID Method.

5.5.2 Sharing the Vendor Public Encryption Key

In this section we describe how the vendor generates an encryption key pair so that customers can encrypt a review so that only the vendor can read it.

A vendor's wallet generates an encryption key pair (S_{ve}, P_{ve}) and publishes the public key P_{ve} signed by the vendor root key S_v. The signed key is published on IPFS and broadcasted using IPFS pubsub.

By requiring that the vendor encryption public key, P_{ve}, and the signature verification keys, P_{vi}, both be signed in the same vendor root key, we are able to guarantee that the entity who signs the `Proof of Payment Request` is the only one that can decrypt the `Review Record`.

Once the `Review Record` is decrypted by vendors, they can then share the reviews with any marketplaces they want. This guarantees that the `Review Record` are private to the vendor until

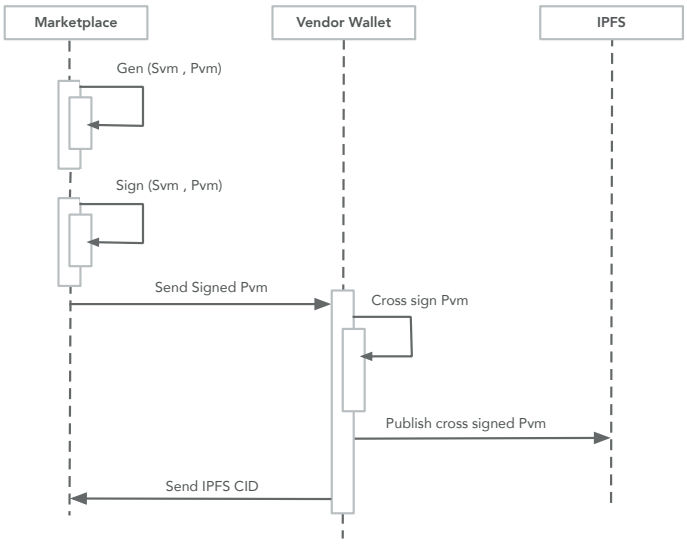

Figure 2: Key distribution: Vendor

the moment the vendor decides to share them.

Figure 2 shows the sequence diagram for generating vendor's signing and encryption keys and publishing them on IPFS.

5.6 Making Payments and Saving Review Records

We now describe how payment processing works so that the each published review is supported by a `Proof of Payment` and each payment received is in turn supported by a `Proof of Payment Requested`.

When a customer selects an item and wants to make a payment, the marketplace first generates a `PoPR` and shares it with the customer. The customer opens the payment request in a customer wallet and this payment request includes the `PoPR`. The customer then chooses to make the payment, and includes review and a rating as part of the payment. The customer's wallet makes a blockchain transaction to make the payment to the vendor and embeds the following two IPFS content hash of `Review Record` in the payment record.

RR The `Review Record` includes the review and rating sent by the customer and saved on IPFS.The content address of the `Review Record` is also saved inside the blockchain payment record allowing anyone to find the review left for the vendor with the payment. The `RR` also contains a content hash in plaintext, and the vendor has to include this hash whenever they publish the review to a marketplace. This allows anyone to verify that a review is backed by a `Proof of Payment`, which in turn is backed by `Proof of Payment Requested` as shown earlier, and the review has not been tampered with.

Figure 3 shows how marketplace and the customer wallet interact to save the `PoPR`, `PR` and `RR` when a payment is made to a vendor selling products or services on the marketplace.

5.7 Discovering Reviews

Marketplaces can observe all the blockhchains that support Chlu payments react to any Chlu transactions, discovering new reviews as they are are authored and linked to payments.

Chlu also provides for customers to update a review, and since review updates are not linked to a blockchain transaction, we provide an alternative way for marketplaces and any other interested third parties to receive an event when a review is updated. We have developed a bespoke solution to achieve the same.

Our current solution uses the IPFS pubsub system to provide a solution of discovering and replicating reviews. Currently we are using orbit-db[16] to provide event subscription services. Every time a review is updated by a customer, it is first validated and if found to be valid is broadcast to the entire network and replicated on IPFS by Chlu and ideally by any other interested

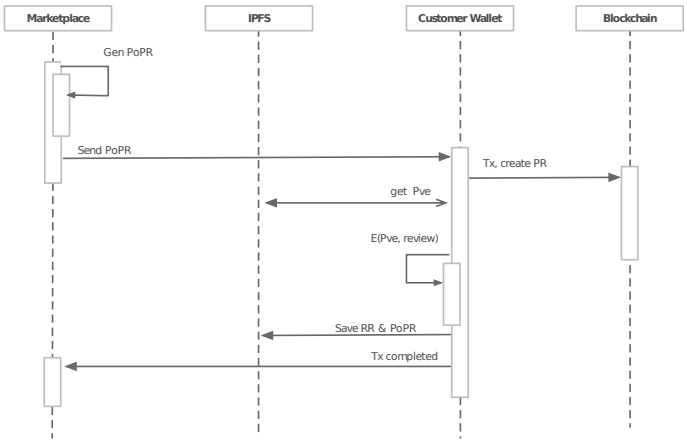

Figure 3: Payment Processing

third parties, for example marketplaces who have an interest with access to this data with low latency.

Apart from orbit-db, the IPRS[9] specifications developed by IPFS team is also a promising avenue, and we are also exploring implementing peer to peer storage network.

5.8 Authenticating Reviews

Authenticating reviews requires that anyone can validate that reviews and review updates are authored by the same entity.

Chlu provides the same by requiring that the customer sign the review record with a private key and include the corresponding public key to verify the signature in the review record. When the customer publishes an update to the review, the customer includes a signature using the same private key, so that the public key published in the original review record can be used to verify that the update to the review record is written by the entity that controls the private key.

Customer wallets then publish all reviews and review updates as normal IPFS objects and anyone can then pin these objects.

5.9 Timestamping Reviews

We provide a means to provide virtual timestamps on reviews so that there is a way for marketplaces to determine the elapsed time between when a payment was made and when a review is written. The goal of providing these timestamps is not to provide a precise timestamp but to limit a customer from accumulating reviews and sell the right to edit them to an attacking vendor. Using the timestamping scheme, as a customer makes purchases from a Chlu enabled wallet, marketplaces can determine this history of purchases and disregard reviews being created in response to a payment made in the past. The time limit beyond which a marketplace wants to disregard reviews is set by individual marketplaces. Chlu suggest this limit be set to 2 months, but each marketplace can set this limit independently.

We now describe how the review timestamping scheme works and later describe further how marketplace can use these timestamps.

5.9.1 Hash Pointers of Reviews

When a customer writes a review, the customer wallet embeds a hash pointer inside the `Review Record`, shown as attribute `Prev` in 5.3. As shown in figure 4, each `Review Record` contains a hash pointer to the previous record created by the customer. Each `Review Record`, R_i can be resolved to a payment P_i which includes the timestamp T_i. The reviews $R_1, R_2, R_3 \ldots$ can be for different payments $P_1, P_2, P_3 \ldots$ created at times $T_1, T_2, T_3 \ldots$ and they don't need to be reviews for the same purchase.

By requiring that customers store a chain of reviews connected by hash pointers, we enable any marketplace or independent user to conclude that an update, $R_1^{'}$, to review R_1 is at least more recent than the time payment P_3 was made.

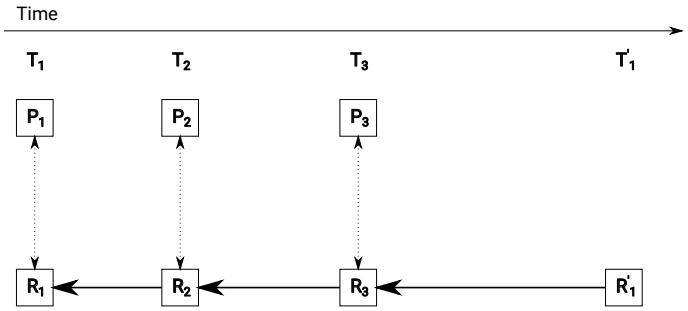

Figure 4: Hash pointers in customer reviews

By providing a hash chain of reviews and being able to cross reference payment timestamps, marketplaces can determine if a customer has created a review for a purchase made long in the past. If they find a long time has passed since the purchase, the marketplace can choose to ignore the review.

5.10 Verifying Reviews

Vendors are free to share their review history with marketplaces and send an update to the marketplace when a review is available. We show how a marketplace can verify that the `Review Record` submitted by a vendor has not been modified by the vendor since it was published.

There are a few important elements of `Payment Record` and `Review Record`:

1. `Payment Record` contains the IPFS content address of a `Review Record`.
2. Each `Review Record` is encrypted using P_{ve}, the public encryption key of the vendor. This key is signed by of S_v which was also used to sign P_{vm}, which was in turn used to sign the `PoPR`.
3. Each `Review Record` has two parts to it, the hash of the plaintext and the encrypted review.

Once a marketplace has the above information, it can take a hash of the plaintext review submitted by the vendor and compare it to the hash included in the `Review Record`.

This is, however, not enough to make sure the review is valid. We need to further verify that the `Proof of Payment Request` is also signed by the vendor's public key P_{vm} and that P_{vm} is signed by P_v and P_m. Algorithm 3 shows how marketplaces verify that the review has not been changed by the vendor and is a valid review created in response to a request for payment made by a marketplace on behalf of the vendor.

Algorithm 3 Verifying reviews

Marketplace fetches the `PoPR`
Marketplace fetches P_{vm}, P_v and P_{ve} linked from the `PoPR`
Marketplace verifies that P_v is the same as the P_v sent to the marketplace at the time of vendor registration
Marketplace validates that the signature for `PoPR` can be verified using P_{vm}
Marketplace validates that a signature on P_{vm} and P_{ve} can be verified using P_v
Marketplace validates that a signature on P_{vm} can be verified using P_m
Marketplace then fetches the `Review Record`, `RR` that it wants to verify
Marketplace finds the plaintext `RRv` with the same hash that vendor has submitted
Marketplace takes a hash of the plaintext part of `RRv` and compares it to the hash in the `RR`
If the above two hashes are the same, the marketplace is sure that vendor is in control of the private key (S_v) whose public key (P_v) is used to sign the P_{vm} that was in turn used to sign the `PoPR`

Figure 5 above shows the sequence of actions for review verification.

The above process assures the marketplace that the same vendor is in control of S_v and S_{ve} and that the vendor has not altered the review since it was encrypted by the customer wallet using P_{ve}.

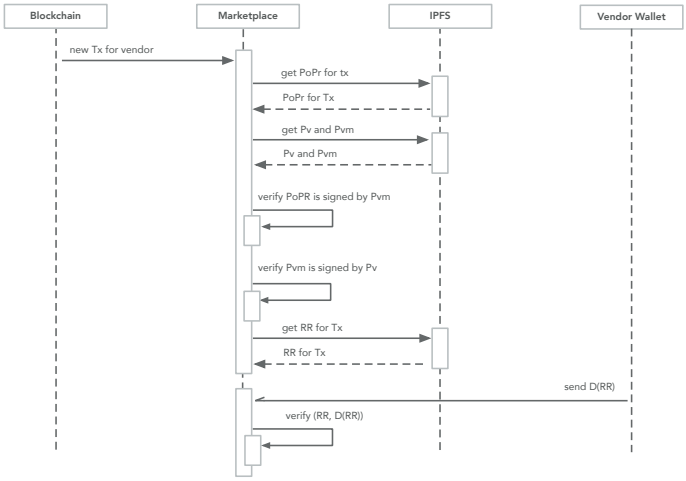

Figure 5: Validating reviews found from blockchain transactions

This section described how vendors can share their reviews with marketplaces directly using their wallet. It should be noted, however, that this functionality can be provided by any service that watches the blockchain for transactions including payments to a vendor address.

5.10.1 Review Completeness

A vendor receives payment to an address they submit to a marketplace. Once we show that review completeness property is provided for a single vendor address it is easy to see how the protocol can be extended to support multiple vendor addresses. Multiple addresses and review completeness is easily provided for once we add a requirement on the marketplaces that they publish all the vendor addresses that vendors provide them for receiving payments.

To see how review completeness is provided for a single vendor address. We note that marketplaces can monitor a blockchain for any payments to that address and can find all the `Review Record`s for all of those payments.

5.10.2 Reviews After Product Delivery

The above solution shows how a review that is created when a customer makes a payment is shared with marketplaces in a secure way. However, we still need to allow customers to pay for a product and create a review after the product has been received, which means separating the events of payment and writing a review. While trying to address this separation we can also provide support for customers updating reviews that they have already created for a given purchase.

As described up till now, the `Payment Record` stored on the blockchain stores a reference to the IPFS content address of the `Review Record` authored by the customer. To support the separation of payments and reviews, and to allow customers to update a review, we change the requirement of the IPFS content address to instead be an IPNS name of a directory.

Once the marketplace has access to an IPNS name of a directory from a `Payment Record`, it can poll the directory to check if there is any new data. If new data is available under the directory, IPNS guarantees that it has been created by the customer who left the first review. This is achieved by the IPNS requirement that only the holder of the private key that created the IPNS name can add content to the location pointed to by the IPNS name.

The above requires a customer wallet initialization similar to the vendor wallet initialization so that the customer wallet creates a key pair to be used for saving reviews and publishing an IPNS name.

In future, the requirement from the marketplace to poll an IPNS directory can be replaced by IPFS pubsub once it is ready for production release and, more importantly, supports authentication of pubsub events[11].

[11] https://ipfs.io/blog/25-pubsub

5.11 Sybil Attacks

Any reputation system needs to be resistant to Sybil attacks, so that no vendors can create multiple customer accounts and give themselves high ratings through these sockpuppet accounts. A reputation system can be made resistant to Sybil attacks by requiring high costs for for using the system or requiring a network of identities within the system. Since Chlu is free to use it would first appear that it would be highly vulnerable to Sybil attacks. But there are two important features that make Chlu resistant to Sybil attacks:

1. Marketplaces are free to choose how they use a vendor's reputation history
2. Marketplaces charge vendor a commission to sell items through their service

Keeping the above two points in mind, let us see how a vendor can conduct a Sybil attack and how Chlu remains resistant to the attacks thanks to well behaving marketplaces.

Say a vendor launches a Sybil attack by creating numerous fake customer accounts and tries to gain a positive reputation history. In this case, the vendor creates a dummy marketplace, that charges no commissions, generates `PoPR` for the sockpuppet customers and makes payments to the vendor's account backed by these `PoPR`, attaching the best ratings to each payment.

The vendor will succeed in creating a reputation history where each `PoPR` is signed by the dummy marketplace, and the vendor could use this history to show top ratings on their own website. However, any other marketplace will chose to leave out ratings from this dummy marketplace. The vendor's ratings on other marketplaces will only show ratings received on other marketplaces that they accept ratings from.

If vendors try to create sockpuppet accounts on a legitimate marketplace, they will have to pay commissions of up to 15% charged by these marketplaces, the vendor will not be able to leave too many positive reviews without burning a lot of money.

6 Which Blockchain

The system described above can work on Bitcoin and all the blockchains derived from bitcoin that support the `OP_RETURN` operator. The content addresses for `Review Record` and `Proof of Payment Request` can be saved in the `OP_RETURN` field, as they will together fit into the 80 bytes limit for `OP_RETURN`.

We provide reference implementations for Bitcoin, Litecoin and ZCash, all of which support `OP_RETURN`. As well as on Ethereum[12] where we store IPFS content addresses on the blockchain using a thin smart contract.

7 Related Work

Chlu is not a marketplace. This immediately differentiates it from all other efforts to provide decentralised marketplaces. Instead, Chlu is a service that can be used by any marketplace, decentralised or not.

Chlu is an openly accessible and decentralised service for making payments and building online reputation as a vendor. We contrast Chlu with the other decentralised marketplaces to make it clear where Chlu stands in comparison. In the case where the marketplace does not yet exist, but is only described on a website, we compare Chlu with the planned features offered by the marketplace.

The most important features we compare are whether the review and ratings systems are walled gardens or not, and whether they require proof of payment for reviews. We also include extensibility as a property because without it, the reputation system can not be used by marketplaces that encourage different user behaviours.

If the specifications of the various systems don't provide a clear answer for these properties, we leave it as unknown.

[12] https://www.ethereum.org/

System	Walled Garden	Proof of Payment	Extensible
Chlu	No	Yes	Yes
OpenBazaar[17]	Yes	Yes	No
Ethlance[7]	Yes	Yes	No
District0x[5]	Yes	Unknown	No
Colony[4]	Yes	Unknown	Unknown
Monetha[13]	Yes	Yes	No

The above table shows how Chlu's goal of decentralised reputation is not being provided by any other system.

8 Conclusion

We describe Chlu, a decentralised reputation system where the vendors are in control of the reputation they have earned by selling products and services on various marketplaces across the internet. Chlu defines protocols that do not require any trusted third party to run services or authorise any transactions. Chlu protocols use a blockchain for making payments, save reviews and ratings on IPFS, and use cryptography to provide a secure system of verifying claims made by vendors.

In a global economy with products and services bought and sold between strangers, reputation systems are instrumental in driving business. However, until now, reputation systems have been enclosed in walled gardens. Even if vendors have excellent reputations in one marketplace they are unable to successfully sell products and services on another marketplace.

Marketplaces also have to constantly fight vendors trying to game their reputation systems, with Chlu's reputation history backed by a proof of payment, and no need for a trusted third party required to verify transactions, all marketplaces will benefit by adopting Chlu.

References

[1] Muneeb Ali, Jude Nelson, Ryan Shea, and Michael J. Freedman. Blockstack: A global naming and storage system secured by blockchains. In *2016 USENIX Annual Technical Conference (USENIX ATC 16)*, pages 181–194, Denver, CO, 2016. USENIX Association.

[2] Juan Benet. Ipfs — content addressed, versioned, p2p file system. https://github.com/ipfs/papers/raw/master/ipfs-cap2pfs/ipfs-p2p-file-system.pdf, 2014.

[3] Vitalik Buterin Jon Callas Duke Dorje Christian Lundkvist Pavel Kravchenko Jude Nelson Drummond Reed Markus Sabadello Greg Slepak Noah Thorp Christopher Allen, Arthur Brock and Harlan T Wood. Decentralized public key infrastructure. Rebooting the Web of Trust I: San Francisco, November 2015.

[4] Colony. https://colony.io, 2017.

[5] A collective of decentralized marketplaces and communities. https://district0x.io/docs/district0x-whitepaper.pdf, 2017.

[6] Dave Longley Christopher Allen Ryan Grant Markus Sabadello Drummond Reed, Manu Sporny. Decentralized identifiers (dids) v0.9. Technical report.

[7] Ethlance. https://ethlance.com.

[8] Conner Fromknecht, Dragos Velicanu, and Sophia Yakoubov. A decentralized public key infrastructure with identity retention. 2014.

[9] Iprs - interplanetary record system spec. https://github.com/ipfs/specs/tree/master/iprs.

[10] The "keybase" well-known resource identifier. https://keybase.io/docs/keybase_well_known, 2014.

[11] Bogdan Kulynych, Marios Isaakidis, Carmela Troncoso, and George Danezis. Decentralizing public key infrastructures with claimchains.

[12] S. Matsumoto and R. M. Reischuk. Ikp: Turning a pki around with decentralized automated incentives. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 410–426, May 2017.

[13] Monetha. https://www.monetha.io/static/media/Monetha_White_Paper.72d6c2dc.pdf.

[14] Ruggero Morselli, Bobby Bhattacharjee, Jonathan Katz, and Michael Marsh. Keychains: A decentralized public-key infrastructure. Technical report, University of Maryland, College Park College Park United States, 2006.

[15] M. Nottingham and E. Hammer-Lahav. Defining well-known uniform resource identifiers (uris). RFC 5785, RFC Editor, April 2010. http://www.rfc-editor.org/rfc/rfc5785.txt.

[16] orbit-db. https://github.com/orbitdb/orbit-db.

[17] Dr. Washington Sanchez. Decentralized reputation in openbazaar. https://blog.openbazaar.org/decentralized-reputation-in-openbazaar, 2015.