Description of a Bitcoin Banking system

Lionel Dricot < ploum@ploum.net > - GPG 28405A38

ploum on bitcoin forum

http://ploum.net

v0.1.2 – changed the exchanges paragraph to make it easier – 29/06/2011

v0.1.1 – spelling and added paragraphs about market makers, generalization, differences with traditional banks and security concerns – 28/06/2011

v0.1 – initial draft - 26/06/2011

Currently, Bitcoin faces two main technical challenges that forbid it to gain popular acceptance. Those problems are usability and security.

Usability because using Bitcoin is still really complicated and way too different than what people know. For instance, people want to send a message with their payment. They also want to be able to pay someone without asking him for an address first. Last but not least, merchants want to be able to initiate payment by themselves, only asking the user for a confirmation.

Security because, in the real world, you have to consider the user's computer as being compromised. You cannot trust the user. Also, unless you use GPG signed messages, you cannot trust a bitcoin address on a website to be the real address. What if the website was defaced?

Those are technical problems. In this paper, we propose to solve those problems by the mean of a new protocol, a layer above bitcoin called “Bitcoin Banking”. Only a very high-level overview of this protocol is provided in this document.

We know that the term “Bank” might be frightening to a lot of people. But a Bitcoin Bank would have as much in common with a traditional bank than a mail server has with a post office. We also deliberately keep the loans and debts subject brief as we consider it as not mandatory in a bitcoin economy. It is the sole responsibility of each individual bitcoin bank to have a proper lending policy.

It is also assumed thorough this paper than users are able to use GPG. This is, sadly, not the case because there is no easy-to-use interface to GPG. We hope to see such an interface in the near future.

A Bitcoin only economy

In the first part of this paper, we will consider a bitcoin only economy. We consider that every user has bitcoins, earn them and spend them. The interaction with other currencies will be considered in the second part of this paper.

The end user and his bitcoin bank

What is a Bitcoin bank? To put it simply, it is a web based service who will store your bitcoins. Let's call that service foobank.com.

Users of this service will have an unique ID in the form of alice@foobank.com.

Your bank account ID will then have the same format as an email address or XMPP address.

This allows a bank account ID to be part of a GPG key. Using GPG, you can guarantee that a given bank account is owned by a given person.

The work of the bank will be to provide a web interface for its customers (or any other kind of interface). It will be also possible for anybody with the technical skills to open his own bank.

The situation might be compared to the mail server situation. You can currently choose between gmail, hotmail, yahoo, a local provider or even open your own mail server.

Some services might also offer mail, XMPP and banking with one single ID.

The handling of the security will be done by each bank separately and might be similar to what we experience currently with traditional online banking system, like two factor authentication (your bank providing you with a disconnected device containing a private key and allowing you to solve challenges).

Payer initiating a bitcoin payment to another bank

Take a typical situation where Alice wants to pay 5btc to Bob, as a gift for his birthday.

Bob has an account on barbank.net. When an user open a new account in a bank, he GPG-signs a certificate ensuring that he has well opened an account on barbank.net.

In his interface, Alice only set the recipient to “bob@barbank.net”, the amount to 5btc and the message to “Happy Birthday my dear Bob”.

Once the transaction is confirmed by Alice (probably by solving a security challenge), the following happens:

foobank.com find the server responsible for banking at barbank.net. Our proposed solution is to use Webfinger. A custom DNS entry might also be possible. Once the banking server for barbank.net is find, the following happens:

→ Open new transaction for bob@barbank.net

← bob@barbank.net user exists and has an account

← GPG certificate signed by bob

Foobank.com check that the signature of the certificate is not revoked. Foobank might even present the GPG signature to Alice to allow her to check that Bob is the bob she knows.

→ transaction from “Alice <alice@foobank.com>”

→ amount 5btc

→ message “Happy Birthday my dear Bob”

← Receiving address: 15SCCTDK9xcZyKFWXsPRXYS9s1m3Mikcxs

→ payment sent

This is of course only a high-level overview. The similarity with SMTP is striking.

To protect against man-in-the-middle attacks, this would have to be encrypted. It also have to be signed in order for barbank.net to ensure that the transaction is coming well from foobank.com.

Even if it is not strictly necessary, Banks might want to build a web of trust between themselves. For well trusted banks, it might be decided to wait for less bitcoin confirmations or even for no bitcoin confirmation at all, allowing very fast payment processing.

Now, as for SMTP, you realize it is possible to send payment using a fake name. There is two solutions to this. Either you consider it as allowed by the protocol and forbid transactions beneath a certain threshold to avoid spam, either you add one more step and ask barbank.net to verify that foobank.com really hold an account for alice@foobank.com, confirming that with the GPG signed certificate. If it might be not suitable to allow a fake name for the sender (allowing Eve to send money to Bob as she was Alice, confusing the poor Bob), it is highly recommended to accept anonymous transactions.

This system also allows you to use your personal DNS as your banking address. For example, Eve owns myserver.com and her email is eve@myserver.com.

She opens an account on foobank.com, requesting eve@myserver.com to be an alias to her account. She signs the certificate and, on her website, she make sure that a webfinger request for banking returns “foobank.com”.

If she changes and open an account on barbank.net, she has to revoke the foobank.com certificate, sign the barbank.net certificate and change her webfinger configuration to return “barbank.net”.

Making a bitcoin payment on an online webshop

In most transactions, the payment is now initiated by the seller.

The buyer will enter his bank account ID in a web form: bob@barbank.net.

The webshop software instructs his bank (foobank.com) to initiate the transaction.

When bob buys a tshirt at tshirt.com, the following happens between foobank.com and barbank.net.

→ Open new transaction from bob@barbank.net

← bob@barbank.net user exists and has an account

← GPG certificate signed by bob

Foobank.com check that the signature of the certificate is not revoked.

→ transaction to “Tshirt <thirtcom@foobank.com>”

→ amount 0.77btc

→ message “Tshirt.com transaction 42”

→ Receiving address: 15SCCTDK9xcZyKFWXsPRXYS9s1m3Mikcxs

← confirmation page: http://barbank.net/012345

Foobank.com then send this confirmation page URL to the webshop which displays it to Bob.

He's presented with a challenge to confirm the transaction. This challenge and confirmation page might be different for every bank.

As soon as the transaction is confirmed, the payment is sent.

← Payment sent

Depending on its trust for barbank.net, foobank.com might confirm immediately the transaction or wait for a given amount of confirmations on the bitcoin network.

When foobank.com decides to confirm the transaction, it calls the webshop API.

From the webshop point of view, there is nothing new compared to existing online payment solutions.

An anonymous alternative could be done where the webshop displays a bitcoin address given by foobank.com. From the tshirt.com perspective, this doesn't change anything.

Instant bitcoin payment in a physical shop

The physical transaction follows exactly the same protocol. But, instead of being asked to enter a mail address, the user could be allowed to simply present his NFC compatible phone. The confirmation page and the challenge would be displayed on the user phone screen.

But, for such transactions, waiting twenty minutes for confirmation of the bitcoin network is not an option.

Instead, foobank.com might decides to trust barbank.net and confirm the transaction immediately. This is doable for small sum (the vending machine scenario).

The next level would be to wait for the transaction to hit the network, even with 0 confirmation. That would add a few second latency if banks are well connected, which is nothing more than what we are used to with current solutions.

It would be up to the bank to decide the quickness of the confirmation, based on the amount of the transaction and the trust of the other bank.

It is well accepted that, for larger sum, you might have to wait longer. When you

buy a car or a house, you are generally not in a hurry and don't mind waiting a few minutes.

Escrow policies

Unlike traditional bitcoin transactions, the proposed banking system allows to recover from error. If you sent by mistake to bobby@barbank.net instead of bob@barbank.net, you can ask your bank to revert the transaction. The bank will itself contact barbank.net. If bobby@barbank.net agrees that it is an error, the transaction is easily reverted. This happens routinely in the traditional banking world.

But what if bobby says that he provided a service. What if you did not received what you bought on a webshop?

This is were an escrow is needed. In the current world, this job is done by credit cards company like VISA or MasterCard.

In the Bitcoin world, that would be a job for your bitcoin bank. As the market is very open, escrow policies might be completely different from one bank to another, might be charged for or not.

For instance, foobank.com could provide extensive escrow policies. Your first complain will immediately be reimbursed without questions. They will investigate every case.

On the other hand, barbank.net will not provide any escrow service. But they are a lot cheaper.

But another tool could be provided: reputation. For every bank account address, a reputation could be computed. There are two kind of reputation: the internal reputation, computed by the bank, and the external, decentralized reputation.

When initiating a transaction to bobby@barbank.net, foobank.com could add a new step in the protocol and ask for the reputation of bobby@barbank.net.

The reputation will be computed with the total of litigious bitcoins received and the total number of bitcoins received. Both number are know by the bank. The exact formula is still to be determined.

That internal reputation has several problems:

- you have to trust the bank to compute an accurate reputation.
- when you change your bank provider, you reputation is reset to 0.

Both problems might be acceptable. A real solution would be to provide an independent decentralized reputation mechanism but it falls out of the scope of this paper.

Anonymous transactions

Some bitcoin users care a lot about the anonymous aspect of Bitcoin. It should be underlined that, in most transaction, anonymity is not wanted. You want to know who you are paying or from who you receive money.

But we recognize that there are a few cases were anonymity is needed.

Bitcoin banking doesn't really break you anonymity if you don't want to. Most of the mail providers allow you to create a new mail address without providing any identification. It is expected that it would be the same for Bitcoin banks. A Bitcoin bank that you would access only with the TOR network would never know who you are.

The protocol being open, we also expect to see the apparition of open source implementations, allowing people to run their own bank if they want it.

Being anonymous would then be only a matter of finding a bank in a friendly country or to open your own bank.

The current official bitcoin client could be easily adapted to talk with banks that should accept anonymous transaction. You would enter “alice@foobank.com” in you bitcoin client, foobank.com will return a bitcoin address and the payment would be processed as any bitcoin spending. As we saw previously, we highly recommend that bank accept anonymous transactions.

Bank should also offer in their interface the possibility to send bitcoins to a plain bitcoin address.

Such transaction will of course come with a lot of confirmation and warning, explaining to the user that, after confirmation, this transaction could not be reverted nor could the receiver be found.

Last but not least, direct bitcoin transactions between two bitcoin clients are of course still possible.

Interacting with other currencies

Indeed, bitcoin doesn't exist alone and, at least in the short-term and mid-term, it is required to allow people to exchange any currency for bitcoins.

Built-in exchanges

Currently, most bitcoin exchanges hold your money. In Euros, dollars or bitcoins. As such, they act as a bank.

It make sense for a Bitcoin bank to do the same. Note that this is not mandatory for a Bitcoin bank. Barbank.net might be bitcoin only. On the other hand, Foobank.com accepts euro payment via SEPA transfers. After the transfer, you will see euros on your account.

This is now up to the bank to determine which kind of exchange interface they want to provide but it could be transparent enough to allow people non-used to trade to place orders. For instance, you could have a line saying “convert euros to bitcoin if btc value under 12.4€”.

Foobank could try to fulfill its customers orders internally but, also by connecting to external exchange services.

This would allow them to offer a “quick buy” features who will ensure the best

rate on the market, buying 2btc at a good rate on one exchange and buying 3btc on another to transparently buy 5btc.

A huge decentralized exchange

It is often considered that, in order to build a decentralized exchange, all players have to agree on a protocol and implement it.

With Bitcoin banks, it might not be needed. Indeed, if the bank open an account on every major exchange and use it through its API, it means that customers from that bank will always be able to have the best possible rate without having an account on any of those exchanges.

The bank should only care about keeping sufficient funds (both in bitcoins and fiat money) on all those exchanges and execute its customers orders.

If a customer want to buy 10btc, it might be the best solution to buy 5 on MtGox, 3 on TradeHill and 2 on Bitcoin7. That will be done in a complete transparent way by the bank.

This is achieved without any agreement between exchanges.

Ideally, the API of exchanges should also be standardized to make the job of banks easier but we consider it as out of the scope of this paper.

Guaranteed exchange rate for merchants

A big blocking point for merchant right now is that they don't have any guarantee regarding the value of a bitcoin. Accepting bitcoin is thus seen as a risk.

But a bitcoin bank could play the role of a market maker and offer a guaranteed rate for a given time.

Let's take the example of a webshop tshirt.com, selling t-shirts. The merchant wants 9.5€ per t-shirt. The website state 10€ because of the VISA fees.

With a market maker API, Bob, a simple visitor, would be allowed to switch the whole site to BTC instead of euros. The computation is done at rendering time using a ticker provided by tshirts.com's bank. The ticker says that 1btc = 12.4€. The 9,5€ price is thus rendered as 0.76btc on the website. It is said that prices are informative and subject to change.

When Bob goes to the checkout state and enter her account ID, the following happens between tshirt.com and foobank.com

→ Initiate transaction 042 for 9.5€ from bob@barbank.net. (042 being the webshop internal representation of the transaction).

← You have to ask 0.77btc and add the message “code1234” to the transaction, it should be completed in the following 10 minutes else it might not be accepted.

Tshirt.com thus initiate a transaction of 0.77btc. Bob see that the price was increased and can cancel the transaction. If he confirms, the transaction happens as usual.

As soon as foobank.com, the bank of tshirt.com, receives the payment, bitcoins are sold and 9.5€ are credited on tshirt.com account.

Using its merchant API, foobank.com confirm to tshirt.com that the payment for transaction 042 was received and that the order should be fulfilled.

An anonymous alternative could be provided where foobank.com send a full bitcoin address.

Both solutions provide a full guarantee for tshirt.com. There is no risk involved for them.

The way foobank.com should handle the risk is its own responsibility. They could try to ask a lower selling rate. But the risk can be completely minimized by considering the actual offers on the market.

Such a service would offer a strong incentive for merchant to accept bitcoins. The burden will come from the work needed to implement that kind of solution in every website. We hope to see plugins for the most popular webshop platforms.

Converting it to any currency

A direct consequence of such a system is allowing you to accept any currency.

Tshirt.com would be able to accept Qatari riyals transparently. The only thing required to make it works is to have some offers for bitcoins against Qatari riyals somewhere in one of the exchanges. Tshirt.com would only see euros on its account, not even knowing that someone paid with Qatari riyals.

As we see, this system allows you to transfer money from anywhere to anywhere, in whatever currency you want.

Let say that Alice want to send 1000€ to Bob. Alice make the request on her foobank.com interface. Foobank.com is currently able to sell bitcoins at 12.5€. Barbank.net answer that they are able to buy euros at 12.4€.

The total cost of the transaction will then be 8€. Bob will see 992€ on his barbank.net account.

This is of course more expensive than SEPA but this is 0.8% and is considered as not bad at all in most international transactions.

Also, the bigger the Bitcoin economy is, the thinner the margin between buy and sell will be, inducing a decreasing fee over time.

Conclusion

Bitcoin is a wonderful technology and it gives the foundations of a ground breaking innovation. But a foundation alone is not enough. As such, we consider that a banking system on top of bitcoin is needed.

As we have seen in this paper, there is nothing technically complicated in such a banking system and it could be realized with current standard technologies.

It is also believed that decentralized exchanges and market makers will help to

stabilize the price of Bitcoins or, at least, to make it move slower. This has the side-effect of answering one major concern that people have regarding bitcoins.

The rise of bitcoin banks would also completely change the world of national currencies. In a first stage, bitcoin could be completely invisible. People might use it without even knowing it, exchanges with national currencies being made on the fly. In the long term, it will allow currencies to be completely decorrelated from a geographical place and make them mostly irrelevant. Bitcoin banks might be extremely popular in countries with weak currencies. It should also be noted than, if the popularity of Bitcoin raises, it might have a non-negligible impact on the exchange rates of those currencies themselves.

The success of such a system may have implications way beyond our actual understanding, from the credit card issuer companies to the governments themselves. Studying such possible implications falls outside of the scope of this document.

Implementation

There is currently no known implementation of a bitcoin bank as described in this paper. If there are no major technical difficulties anticipated, this is anyway an important software project that would require important ressources.

Many other aspects have to be investigated, including the legal and the commercial one.

Generalization

The attentive reader would have remarked that Bitcoin is only used as a backend, a way to enter and transmit value through the system.

The author strongly believe that the existence of a sustainable decentralized currency cannot be avoided. It is possible and will happen.

This decentralized currency might not be Bitcoin. Bitcoin could be killed or disappear. A competitor might see the light.

It is important to realize that, for an existing Bitcoin bank, this would not be a real problem. It would only be a matter of extending the protocol to support a new currency.

The direct consequence is that investing in a Bitcoin bank might be less risky than investing directly in Bitcoin. A Bitcoin bank might easily become a Whatever bank if there is a need for it.

Differences with traditional banks

After seeing all of this, the logical next question is to ask in what way a bitcoin bank would be different from a traditional bank.

The biggest problem with the current banks is that the system allows them to lend money they don't have. People think that loans are coming from spare accounts. This is not true. Banks can just create the amount in their database without

thinking more about it. In most countries, laws try to somewhat regulate this and limit the creation of money but, in the end, the bank is still creating money out of nothing.

A Bitcoin bank will not be able to do that, even if a given Bitcoin bank decides to enter in the mortgage business (which is not mandatory for a bank after all).

If Foobank.com decides to lend bitcoins, they know that those lent bitcoins might be retrieved in a few seconds. It means that a sparer should be notified and should agree that his money might be lent to somebody else.

Imagine that Alice want to buy a car from bob for 1000btc. Bob already have 1000btc in his account on the same bank as alice.

Alice contracts a loan with the bank. The bank take the BTC from bob's account and give them to Alice. She immediately send them to Bob's account.

At that point, Bob's account shows 2000btc. Money has been created from nothing and we are back to the good old system.

With one big exception: the bank perfectly knows that there is only 1000BTC in their wallet. There is no way to cheat with that.

A Bank might even offer to display publicly its internal deposit, in order to gain trust from customers. This value could be proven by moving the money and using blockexplorer.

Also, the bank takes a huge risk if Bob simply withdraw all of his money suddenly. That's why he should have agreed first on lending his money. In fact, the big change with the traditional system is that bank will not lend “their” money anymore. They will only act as intermediates to allow people lending to other people. A bank could even offer his customer to fix themselves the rate at which they want to lend.

Instead of being special businesses with a huge power, banks become paying services. People stay in control of their money and only use Banks as a mean to help them.

Security concerns

Making it easy for people to spend their money is also very dangerous. Scam and theft will be made easier. What if someone puts a gun on your head and ask you to empty your account to an anonymous Bitcoin address? What if your phone and your laptop are stolen.

Those concerns are very important and could be answered by allowing users to set a daily spending limit which would be independent for each terminal (one for your phone, one for your laptop). There is a lot of field for innovation there but we consider it as being outside of the scope of this document.

Acknowledgements and tips might be sent to
15SCCTDK9xcZyKFWXsPRXYS9s1m3Mikcxs