Bitcoin Subscription Payments

How to implement recurring payments in Bitcoin

Jason Bourne · Follow
Published in Coinmonks
8 min read · Dec 10, 2018

Pay for your Netflix in Bitcoin?

Traditional reoccurring / subscription payments should be familiar to most. Companies such as Netflix and your electric company bill you monthly for services. The money automatically gets charged to your credit card or pulled from your bank account. This service costs you nothing, though the seller pays transaction fees up to approximately 3% for credit card payments. People like

subscription payments because its convenient. Sellers like them because it eliminates collection issues and people often forget to unsubscribe. In fact, some companies may make it hard for subscribers to cancel by hiding the cancellation process or in a contract you didn't read when you signed up.

For these reasons, subscription payments are popular. However, it is not straight forward to implement in Bitcoin as there are several technical hurdles. The first is that Bitcoin is push technology. Spending is always initiated by the buyer. Traditional credit cards and banking subscriptions use pull technology to work. The charges are initiated by the seller on a person's behalf and sometimes without even their actual permission.

Second, Bitcoin does not have a mechanism to authorize transactions of an unknown amount. You can setup the electric company to debit your account each month even though neither of you know the amount. Committing to a transaction with an unknown amount is something credit cards and banks can do, but cannot be directly reproduced in the Bitcoin protocol. With Bitcoin, the amount of the transaction always has to be specified when a transaction is made.

So seller initiated variable amount subscriptions are not directly possible in the Bitcoin protocol. But, buyer initiated, fixed amount subscriptions payments are. In this article I explore the various ways to implement reoccurring subscription payments using Bitcoin.

Manual Subscription Payments

The first way to do a reoccurring subscription payment is to do it completely manually. While not sexy, it is probably how must reoccurring payments are done today and understanding this case helps set a baseline for any improvements.

Here are the steps for implementing a manual Bitcoin subscription:

1. The seller offers a product or service for a monthly subscription denominated in Bitcoin.
2. When the buyer signs up, seller provides buyer with bitcoin address to send funds
3. Buyer remembers to send payments at the right time
4. Seller ensures payments are received by the right time

5. If payment is not received, seller needs to follow up to determine whether buyer forgot or intended to cancel.

This process is obviously not convenient for either buyer or seller. Both must do work on a reoccurring basis which increases the friction of the transaction and the likelihood of an error. Further, it creates ambiguity when the payment is not received as the buyer's intent is not clear. The seller must follow up.

In short, this version of Bitcoin subscriptions sucks in almost every way compared to existing credit card / bank implementations. Thankfully we can do much better with just a little bit of automation.

Automated Bitcoin Payments

The obvious improvement is to add some automation to the above workflow. We can automate processes for both the buyer and the seller side to make the Bitcoin subscription more set-and-forget.

Here are the steps in an automated Bitcoin subscription process:

1. The seller offers a product or service for a monthly subscription denominated in Bitcoin.
2. When the buyer signs up, seller provides buyer with bitcoin address to send funds.
3. Buyer's hot wallet has the ability to send reoccurring payments. Buyer sets up the payment to seller and they will occur automatically until canceled or until funds run out.
4. Seller has software to monitor for incoming transactions. This software would alert the seller to a missing payment.
5. If payment is not received, seller still needs to follow up to determine whether buyer forgot (or their wallet went offline) or they intended to cancel.

Steps 1,2 & 5 are the same as the manual example. Step 3 & 4 are where the automation happens. The improvement is that the solution is largely set-and-forget as long as the buyer wants to keep paying AND they have enough funds in their wallet. So its clearly an improvement over the manual case.

Coinbase is one wallet that has reoccurring transactions capability and there are likely others. The major downside of this solution is that it requires a hot wallet with a Bitcoin balance. This is a security risk. Also, this solution does require some maintenance because eventually the buyer's wallet will need to be depleted by the subscription.

Finally, this solution doesn't eliminate the ambiguity when a payment is missed. Communication is still required to determine whether the buyer intended to cancel or there was just a payment error. Despite these downsides, you could imagine this being an actual usable solution.

Bitcoin Subscription Time Lock Escrow (STiLE) Contracts

A third way to implement Bitcoin Subscriptions would be to use time-locked outputs. For this to work, the subscription between the buyer and seller needs to be fixed in duration and funded up front in an special Bitcoin contract I'm calling STiLE. The limitation of this solution is that the subscription cannot be open-ended or pay as you go like credit card subscriptions. STiLE contracts must be funded up front, pay the seller over time, yet allow the buyer to cancel and redeem all unspent payments at anytime.

For example, the buyer would agree to to a one-year subscription paid monthly. There would be a STiLE Contract for each agreed monthly payment. The Bitcoin script for the STiLE Contract for the payment one month from now would be:

```
IF
     <now + 1 month > CHECKLOCKTIMEVERIFY DROP
     <Seller's pubkey> CHECKSIGVERIFY
ELSE
     <Buyer's pubkey> CHECKSIGVERIFY
ENDIF
```

This places all subscription payments in an escrow. The logic prevents the seller from retrieving the payment until the payment date (one month from now) while allowing the buyer to retrieve the payment at anytime.

This creates a payment structure that makes the buyer's intent clear to the seller throughout the entire relationship. The buyer does nothing when they want to

continue with the subscription. This allows the seller to retrieve the payments according to the payment schedule. When the buyer wants to cancel the service, they retrieve all unspent payments.

All of these STiLE Contracts can be combined into a single on-chain Bitcoin transaction making it more efficient compared to separate monthly payments (if implemented more efficiently using pubkey hashes).

When the monthly payment is due, the seller can retrieve the funds using

```
<Seller's signature> 1
```

The seller will need to remain on top of their redemptions and redeem the payments promptly when they become available. If they do not, the buyer could redeem payments that are owed to the seller.

And the buyer can retrieve any funds not retrieved by the seller at any time using:

```
<Buyer's signature> 0
```

Here are the steps in for a Time-Locked Bitcoin Subscription:

1. The seller offers a product or service for a monthly subscription denominated in Bitcoin. The seller will require a minimum commitment of, say, 12 months.
2. When the buyer signs up, seller provides buyer with public key to use to create the STiLE Contract.
3. Buyer publishes a single Bitcoin transaction with a STiLE Contract for each payment. The transaction would escrow the full contract amount.
4. Seller has software to verify and monitor the Buyer's transaction on chain. The software will sweep each payment when the payment becomes due.
5. If the buyer redeems the contract, the seller knows the buyer canceled.
6. The seller will need to reach out to the buyer towards the end of the contract

to renew.

As you can see, there is no ambiguous case. The seller always knows the status of the subscription. It is in effect as long as there un-redeemed subscription outputs. It is canceled if the buyer redeems future payments. It is expired when there are no more outputs left.

Just like the other Bitcoin solutions the buyer has completely control over whether a seller gets a payment. There are no cancellation hoops to jump through like with a credit card subscription. The buyer can just redeem all unspent payments at any time and the subscription would be canceled.

Additionally, the fixed length subscriptions creates more of an “opt-in” transaction versus the traditional indefinite subscriptions which are “opt-out”. The seller must explicitly convince the buyer to take action at the end of the contract to renew. Whereas with indefinite subscriptions, the buyer must explicitly cancel and it is common for the buyer to forget to cancel. This, as well as strong cancellation rights give the buyer more power in the relationship and strengthen consumer rights.

Once the subscription transaction is added to the blockchain it is set-and-forget for the buyer for the duration of the subscription. No hot wallet or third party is needed. The seller would need to be online periodically to redeem payments and monitor for cancellations.

This STiLE Contract Transaction can have lower fees for the Buyer, especially if they do not cancel. The reason is that a single transaction with multiple outputs (one for each payment) is smaller than multiple transactions each with one output. The details of this is a future article.

Related to this, STiLE reduces the buyer’s fee risk. The buyer cannot predict how expensive it will be to send payments in the future and this would be a disincentive to committing to a Bitcoin subscription. By bundling all payments in one transaction that published immediately, there is not future fee uncertainty for the buyer.

This type of subscription does increase costs to the seller. First, the seller now needs to issue 12 transactions to sweep each output after the payment is

available. Additionally the the fees for these transactions are uncertain and could possibly be much higher.

This subscription transaction offers the buyer some exchange rate hedging capability. The seller would presumably be on the hook to deliver the services regardless of Bitcoin's fiat price as long as the contract is in effect. However, since the buyer can cancel anytime, they would be incentivized cancel and renegotiate a new contract if the Bitcoin price appreciates.

Almost all of these benefits are for the buyer and often at the expense of the seller. That said, there are two benefits to the seller. First, the seller knows the funds are available as the subscription was funded up front. Second, once the seller redeems a payment, there is no risk of a charge-back or reversal like with credit cards.

Conclusion

STiLE Contracts make Bitcoin Subscriptions feasible. They require the term and amount of the subscription to be fixed. However, it offers the following benefits:

- Strong cancellation guarantees
- More secure — no hot wallet or trusted third parties needed
- Can be set-and-forget for the buyer and fully automated for the seller
- Less expensive for buyer than sending individual Bitcoin payments
- Reliable income stream for seller in Bitcoin(!) with no risk of payment failure

Bitcoin | Cryptocurrency | Subscription Economy | Payments | Tutorial

Follow

Written by Jason Bourne

47 Followers · Writer for Coinmonks

More from Jason Bourne and Coinmonks

Jason Bourne

How Government Will Try to Kill Bitcoin

Bitcoin cannot be “banned” but hyper bitcoinization can be stopped

13 min read · Apr 11, 2018

Johnwege in Coinmonks

Bitcoin is about To SHOCK the World

Bitcoin is about to shock the world, and it will be sooner than you think. This market will test your conviction, unlike anything you have...

· 6 min read · Jan 27, 2024

1K 7

Ava in Coinmonks

My Fastest Way To Make Money with Python Web Scraping

Hey there, fellow programmers! Today, I'm going to let you in on one of my favorite secrets for making money with Python: web scraping...

· 3 min read · Nov 1, 2023

795 5

Jason Bourne

Bitcoin's Taproot

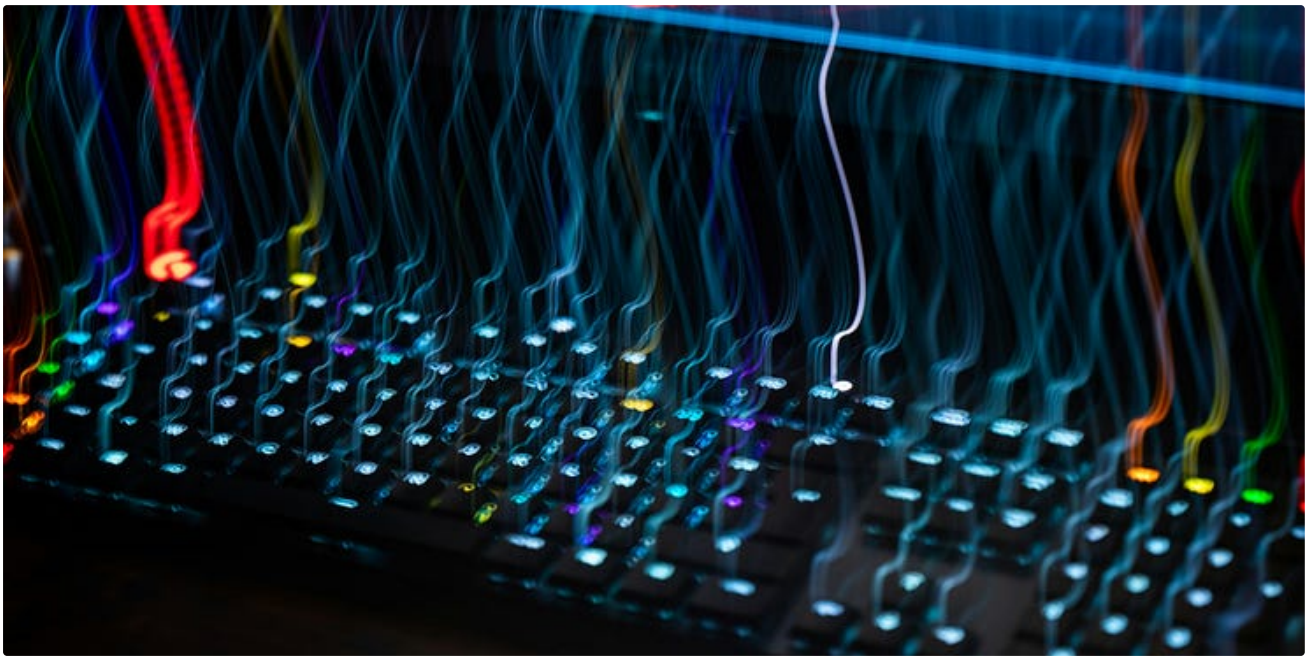
Greg Maxwell's Taproot proposal is a novel way to create complex Bitcoin transactions that are indistinguishable from simple payment...

3 min read · Feb 28, 2018

See all from Jason Bourne

See all from Coinmonks

Recommended from Medium



Mukund Bhuva

How I Hacked the Dutch Government: Exploiting an Innocent Image for Remote Code Execution

Pwning the Dutch Government with RCE

5 min read · Feb 12, 2024

356 3

Acurast

Become an Acurast Ambassador — Applications Are Open Now

2023 just came to an end. We hope you are ready to set goals for this new year. As we kickstart into 2024, we'd like to start with some...

3 min read · Feb 2, 2024

228 1

Lists

Tech & Tools
16 stories · 152 saves

General Coding Knowledge
20 stories · 932 saves

Icon Design
36 stories · 227 saves

Generative AI Recommended Reading
52 stories · 742 saves

Golden Trades

Farcaster — HOLD ON TO YOUR HATS FOR SOME SERIOUS FOMO!

I'm confident that most of you have heard about Farcaster, at least because it's been covered quite extensively all over X. However, many...

4 min read · Feb 11, 2024

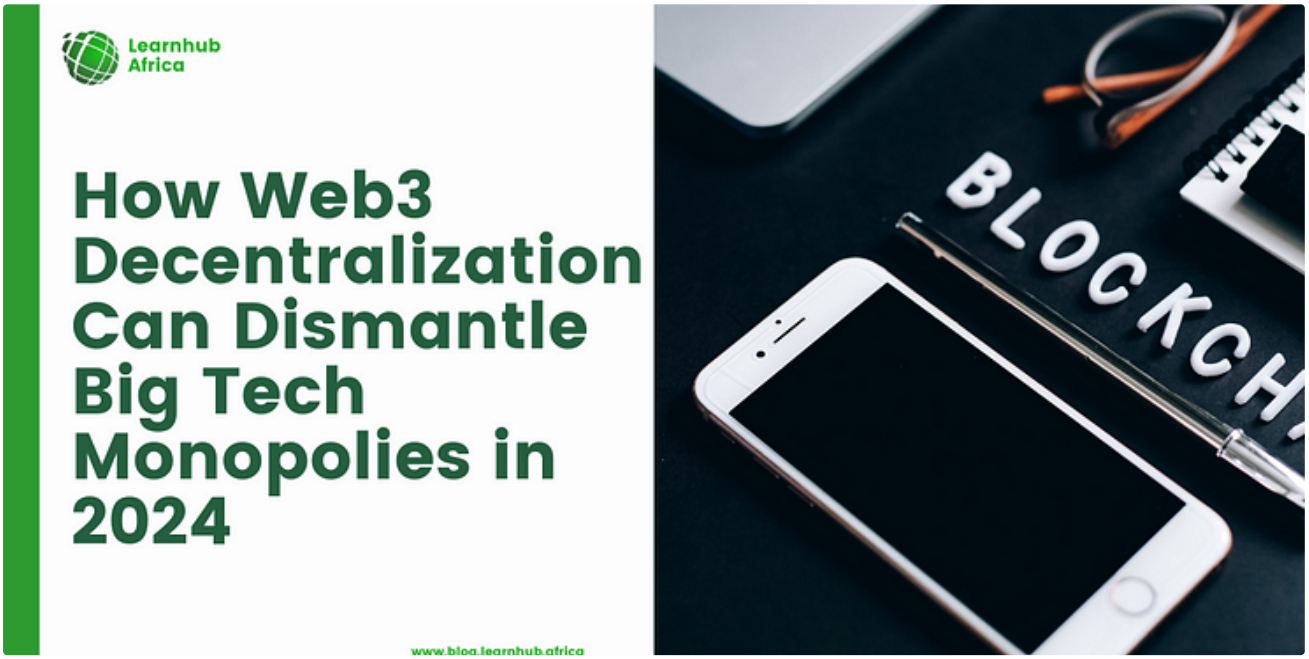
Florian Walter

The Easiest Way to Find CVEs at the Moment? GitHub Dorks!

In this article, I will demonstrate how I used GitHub dorks to find 24 vulnerabilities in popular open-source projects in just a few weeks...

7 min read · Feb 9, 2024

271 3



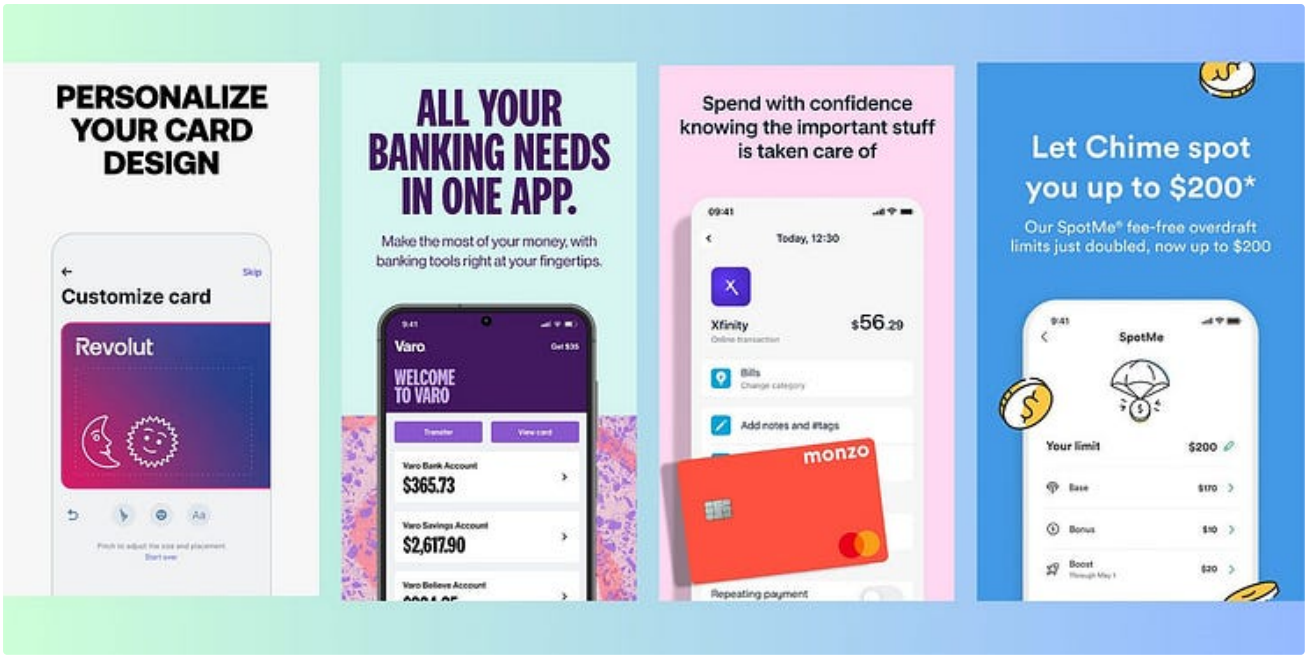
Scofield O. Idehen

How Web3 Decentralization Can Dismantle Big Tech Monopolies in 2024

The staggering market dominance of Big Tech gatekeepers like Google, Amazon, Facebook, Apple, and Microsoft (GAFAM) has raised growing...

7 min read · Feb 14, 2024

1



Sam Boboev

Deep Dive: Neobanking

In this edition:

8 min read · Oct 22, 2023

54

See more recommendations