Lightning Network Part III

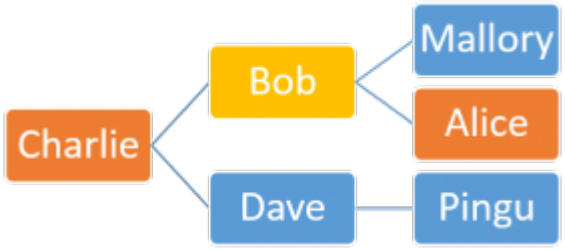
scritto da Alberto De Luigi | 26 Maggio 2016
< Back to Part II

Users trade goods and services in exchange of bitcoin in a permanent way, through a network of users, where each couple of users is connected by a bilateral channel, but the network as a whole allows to exchange bitcoin with everybody without loading any other data on the blockchain

The network of channels

Assume it is not Bob to sell Alice the smartphone, but Charlie.

Charlie has an open channel with Bob and Bob has the same open channel with Alice we analysed in part II.



Charlie sells to Alice. Charlie's software (or Alice's) search for a possible route among the open channels. It finds out Bob as intermediary. But the intermediate steps could be infinite

Alice pays Bob through the open channel (exactly as we already seen), knowing that Bob will let her bitcoins «flow» directly to Charlie. Alice is sure Bob is the correct intermediary because Bob can receive the payment from Alice exclusively if he knows a secret key given from Charlie.

In fact Charlie creates a random string called «pre-image». He hashes the string, producing the «image» and sends it to Alice. Without the «pre-image» only Charlie originally owns,

no one can receive Alice’s payment.

The procedure (summary)

Alice owns the **Image** given from Charlie. Then she creates the **C**ommitment transaction in the open channel with Bob (suppose it is **C2**), which as before if broadcasted allows Bob and Alice to send back the funds in their respective wallets/outputs what previously allocated in **F**.

But there’s a difference: in the new **C**ommitment, there is a new output of 1btc (the smartphone price) which Bob can receive only if he knows the **pre-image**. If he doesn’t know, that 1btc goes back available to Alice (n. blocks later).

Bob can know the **pre-image** only if he already created a new **C**ommitment in the channel with Charlie, where he is committed to send 1btc to Charlie.

That means Charlie opened the «sluice gate» with Bob, consequently, when Bob open the «gate» with Alice, inevitably that 1btc output flows like water directly from Alice to Charlie.

More technically:

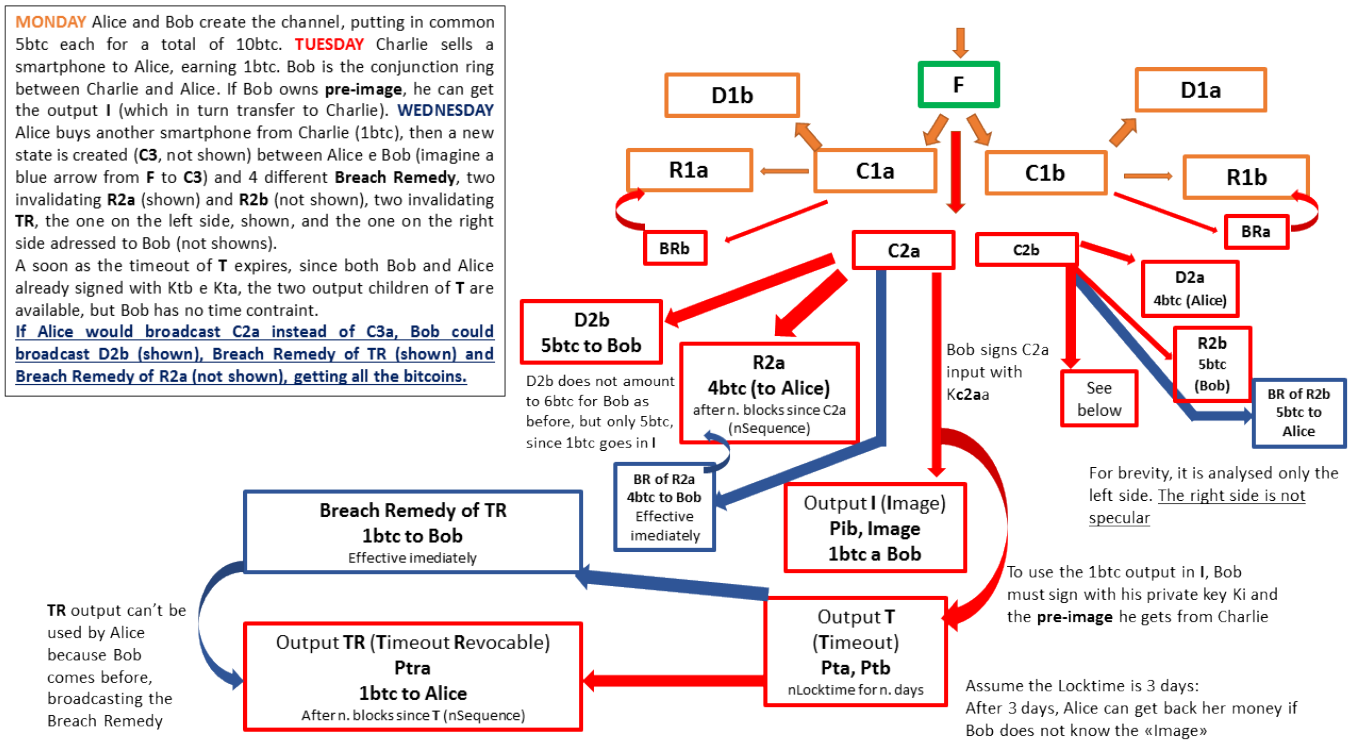
Alice owns the **image**

Alice uses **image** to create the signature script of a new **C2** output, making available that output only if signed with the following «signs» (otherwise it goes back to Alice): Bob’s sign (Kib in the next graph) and the **pre-image**.

Bob uses the **image** given from Alice to create the sigscript of the output of a new transaction **C** with Charlie, available only with Charlie’s private key and **pre-image**.

If it’s true that Charlie sold the smartphone to Alice, giving her the **image**, then Charlie actually owns the **pre-image** corresponding to that **image**. He uses the **pre-image** to receive

Bob's payment, opening the «gate» downstream. In this way the **pre-image** becomes available even to Bob, who then can sign **C2** Alice created, opening the «gate» upstream. Bitcoins «flow» from Alice to Charlie (or better, *the possibility to broadcast to the blockchain the transaction that brings those bitcoins in his/her own output/wallet «flows» from Alice to Charlie).*

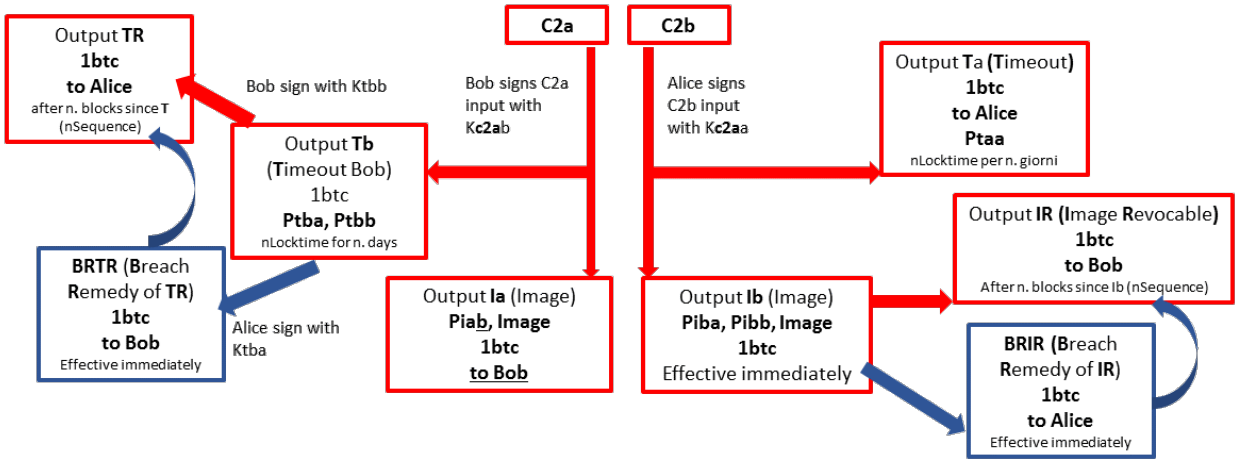


The two branches are not specular. If Alice broadcasts C2a, Bob can gets bitcoins in **Ia** immediately if he knows **pre-image**. If he doesn't, Alice broadcast **Tb** and consequently the child **TR** (Bob already signed with Ktbb at the beginning) and then she can get back 1btc. Later (wednesday) **TR** can be invalidated thanks to **BRTR.**

If Bob broadcast **C2b** and knows the **pre-image**, he gets 1 btc, because he broadcasts directly **Ib** and the child **IR**. If he doesn't know **pre-image**, he can't broadcast **Ib** and Alice gets back 1btc thanks to **Ta** after n. days of Locktime. Also **IR** at a later time is revocable thanks to **BRIR.**

Neither Bob nor Alice have convinience in first broadcasting C2 because in that case they have to wait n. days (n. blocks)

before to get the ouptut spendable.



The two branches are not specular. If Alice broadcasts C2a, Bob can gets bitcoins in **Ia** immediately if he knows **pre-image**. If he doesn't, Alice broadcast **Tb** and consequently the child **TR** (Bob already signed with Ktbb at the beginning) and then she can get back 1btc. Later (wednesday) **TR** can be invalidated thanks to **BRTR.**

If Bob broadcast **C2b** and knows the **pre-image**, he gets 1 btc, because he broadcasts directly **Ib** and the child **IR.** If he doesn't know **pre-image**, he can't broadcast **Ib** and Alice gets back 1btc thanks to **Ta** after n. days of Locktime. Also **IR** at a later time is revocable thanks to **BRIR.**

Neither Bob nor Alice have convinience in first broadcasting C2 because in that case they have to wait n. days (n. blocks) before to get the ouptut spendable.

But how the output «goes to» Charlie?

Bob created with Charlie a new **C**ommitment exactly like he did with Alice, and created a signature script for the output **I** adressed to Charlie, presenting the same **image** Alice used to create the signature script of the output **Ia** adressed to Bob (remember the **image** has been originally sent to Alice from Charlie).

Once all transaction in both channels have been created, if Charlie signs with **pre-image**, 1btc goes from Bob to Charlie and Bob uses **pre-image** to sign **Ia** and **Ib** with Alice, in this way obtaining from her 1btc.

If Charlie doesn't sign with **pre-image** and then Bob can't use **pre-image** in the channel with Alice, she can gets back 1btc thanks to transaction **Tb->TR** or the transaction **Ta** (it depends on who first broadcasts **C2**) and Bob, in the same way, retrieve 1btc from Charlie.

Lightning Network: problems

Motivational problems: freezing funds and broadcasting too late

There’s a «problem» of time lag, for which each actor involved sees his/her money «freezed» in the channel.

If one wants to close the channel, in this way bringing bitcoins from the multisignature funding transaction to one’s wallet, he/she can’t do it instantly. In fact the first who broadcasts is always subject to a time span before he/she can use the output as a new input in another transaction.

For example, Charlie closes the channel open with Bob after having sold, through Bob, a smartphone to Alice. Bob now wants retrieve 1btc from Alice and broadcast to the blockchain. He can’t do immediately unless Alice broadcast **C2**, but she could not broadcast first. Then Bob is forced to broadcast **C2** and wait for the time span set by the time constraint of the output **IR**.

Moreover, Bob could not monitor the blockchain as frequently as he should, risking Charlie gets his money from Bob while Alice still holds the 1btc payment, taking advantage of Bob’s inattention, who broadcast too late.

If Bob wants to secure his money from this risk, he should use a software (will it be fully reliable?) or an intermediary who broadcast Bob’s transaction. He is then forced to bear a cost. But it is not rational to bear a cost only to intermediate Alice and Charlie’s businesses.

Is it thus necessary a commission paid to each intermediary? In this scenario, it is more likely that some users (or «banks») will act as big intermediaries: the majority of users will keep their money in a funding transaction with a single big intermediary which has many open bilateral channels, one for each user. In this way, all the users will trade through

the same intermediary, like a node of the network full of traffic. This big «bank» make easier to find a route in the network when users want to pay off-chain, and will check if some users try to cheat broadcasting past commmitment transactions. For this reason, the big intermediary has to be trusted.

However, we should say that there is a case when LN client might also work without a third party monitoring: it’s the case of unilateral payments. In fact, if the channel is used only to do payments, and not receive money, to broadcast onchain past transactions could only result in a discount of some payments, in the interest of the user!

Too small funding for the payment? Intermediation required

Often users could face problems in finding the «route» throught the intermediaries in order to make off-chain transaction with the Lightning Network, because of too small funding transactions.

For example, Charlie and Alice find Bob as the only intermediary available in the network, but he has too few funds available on the funding shared with Alice to bear the exchange between Alice and Charlie, who are then forced to open a new funding (or create a transaction on-chain).

It’s likely that users in future will keep substantial parts of their bitcoins in fundings with very big intermediaries (like we now hold bank accounts) to make possible payments off-chain of any amount.

Freezing money and Network liquidity

The network might have or not have «liquidity». If there is liquidity, it is because big intermediaries ensure that the funding are enough large to cover the majority of the common transactions.

The problem of freezing bitcoins in the funding transactions is not a serious problem only if:

1)The network is full of liquidity

2)The network is wide

In this case the majority of trades will happen off-chain and users won’t need to «retrieve» on the blockchain what they previously shared in the multisignature funding transaction.

Nevertheless, if the network is not wide or there’s not enough liquidity, users will often need to do many on-chain transactions, then less likely they will leave a large amount of money in the funding. In this case the Lightning Network won’t be very effective.

Use «banks» as intermediaries

For sure, the spread in the population of the Lightning Netowork will solve the scalability problem of the blockchain. Then there is a collective benefit. But users individually have benefits using the Network?

The user of the Network, to avoid risks of «being late» (in broadcasting) might be willing to pay an intermediary who constantly checks cheting attempts. In any case, in order to have a wide network with enough liquidity, it will probably necessary to call on big intermediaries.

On the other side, without the Network, users pay miners’ commission, and in a very near future a «controller» who advise when the output are already spent (when the blockchain will be too large to be downloaded by each single user).

Are these benefits given by the Network perceived by the single users greater than the costs?

If to the «public» or collective benefit (the solution to the scalability problem) doesn’t correspond an adequate «private»

benefit, a problem of «free riding» will raise and this could threaten the effectiveness of the Lightning Network.

< Back to Part II