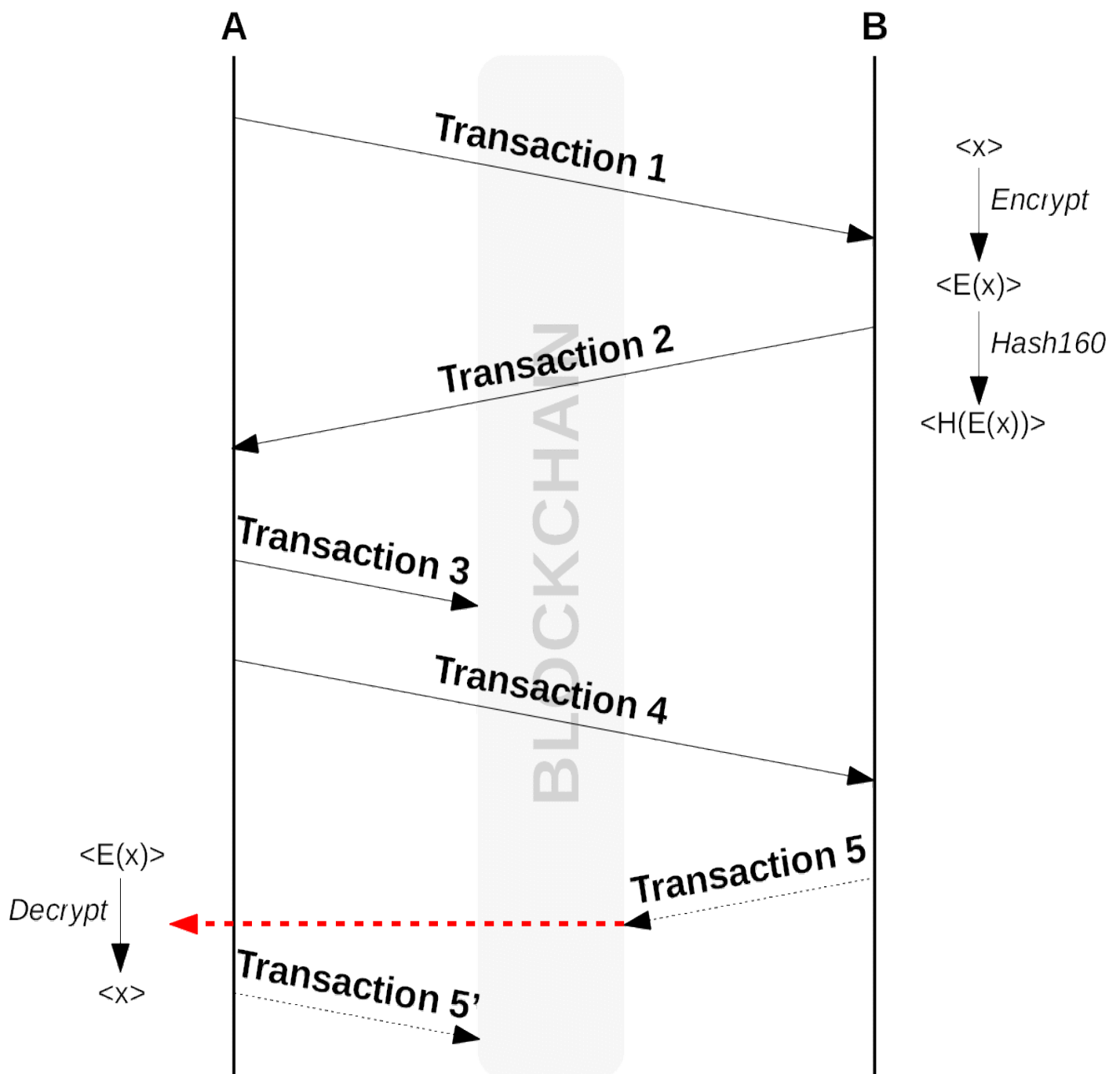


# Purchase coupon in the Blockchain

## I. Example of protocol

Let's say that B wants to sell a coupon  $\langle x \rangle$  for N smly and that A would like to buy it ...



Transaction1 and Transaction2 prepare the information necessary for Transaction3:

### Transaction1

amount: 0

type: TX\_NULL\_DATA

scriptPubKey: OP\_RETURN <I want to buy a coupon>

### Transaction2

amount: 0

type: TX\_NULL\_DATA

scriptPubKey: OP\_RETURN <H(E(x))>

Transaction3 is the protocol's main transaction:

### Transaction3

amount: N

type: TX\_NONSTANDARD

scriptPubKey: OP\_IF  
                  | <PubKeyA> OP\_CHECKSIGVERIFY  
                  OP\_ELSE  
                  | OP\_HASH\_160 <H(E(x))> OP\_EQUAL <PubKeyB> OP\_CHECKSIGVERIFY  
                  OP\_ENDIF

Transaction4 gives B the Transaction3's "txid":

### Transaction4

amount: 0

type: TX\_NULL\_DATA

Now, either B spend Transaction3 and so A will have access to <E(x)> and can decrypt it to obtain <x>...

### Transaction5

scriptSig: <sigB><E(x)>

Or B does not spend Transaction3. A will never know <x> but can get his N smly back.


### Transaction5'

scriptSig: <sigA>

## II. New tools for Smileycoin

The code is available on github: <https://github.com/guighienne/PurchaseInBlockchain>

### 1. Encrypt/Decrypt a message



Encryption Hash

Enter a message here...

Enter a PublicKey (to encrypt) or a PrivateKey (to decrypt)...


Encrypt Decrypt

☐ Automatically convert to hexadecimal

To encrypt a message, enter the message and provide the recipient's Public Key. Click on "Encrypt". At any time, you can switch from plain text to hexadecimal (and vice versa) string by checking / unchecking the box.

To decrypt, enter the encrypted message and provide the Private Key corresponding to the Public Key for encryption. Click on "Decrypt".

### 2. Hash a message



Encryption Hash

Enter hexadecimal string...

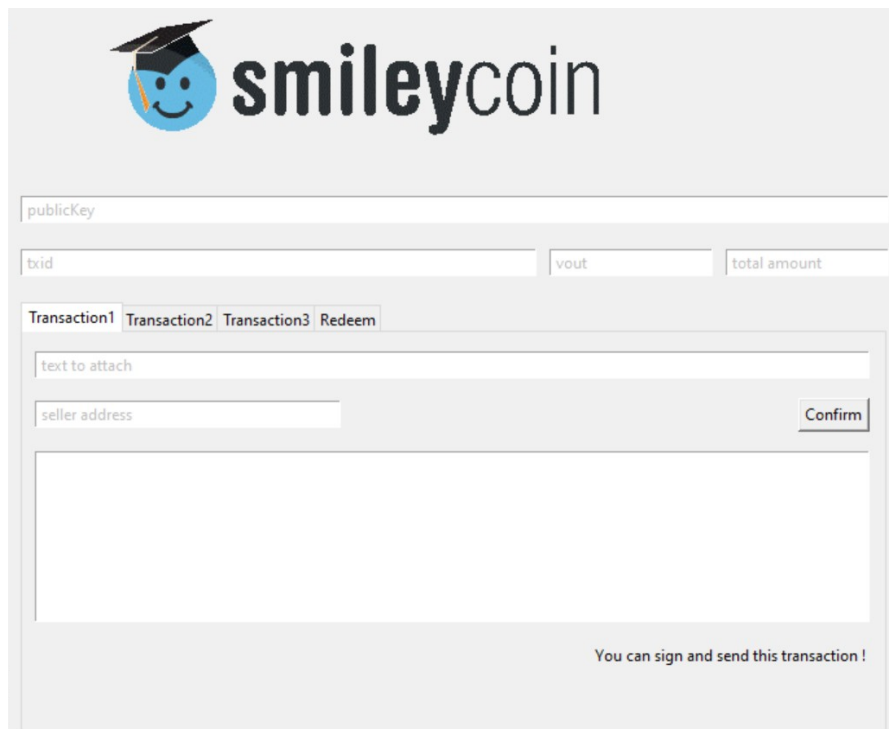
Choose an algorithm: SHA-256 Hash

To hash a message, enter the message (in hexadecimal format), choose an algorithm and click on "Hash".

The algorithms available are the 5 algorithms recognized by Bitcoin Script:

- Ripemd-160
- Sha-1
- Sha-256
- Hash-160 (Ripemd-160 [Sha-256 [x]])
- Hash-256 (Sha-256 [Sha-256 [x]])

### 3. Create transactions

The image shows a web interface for creating transactions on the Smileycoin platform. At the top, there is a logo featuring a blue smiley face wearing a graduation cap, followed by the text "smileycoin". Below the logo, there is a form with several input fields: "publicKey", "txid", "vout", and "total amount". There are four tabs: "Transaction1", "Transaction2", "Transaction3", and "Redeem". The "Transaction1" tab is currently selected. Below the tabs, there are two more input fields: "text to attach" and "seller address". A "Confirm" button is located to the right of the "seller address" field. At the bottom of the form, there is a message: "You can sign and send this transaction !".

This tool allows you to easily create the transactions described in the protocol. Whether you are seller or buyer, you must provide an unspent transaction (txid, vout, total amount), a Public Key as well as various information depending on the transaction you are creating (previous transaction, coupon, or coupon price for example ...).

Apart from the Redeem Transaction, the transactions are easy to sign (`smileycoin-cli signrawtransaction 'TX'`). The Redeem Transaction is automatically signed in the tool. You just have to send it (`smileycoin-cli sendrawtransaction 'TX-signed'`).

## III. Continuation...

- Incorporate the new tools directly into the Smileycoin Wallet
- Modify `IsStandard ()` to be able to send transaction 3 in the blockchain