About cryptocurrencies and blockchains – part 3

Jyväskylä 24th of April 2018

Henri Heinonen (henri.t.heinonen@jyu.fi)

Digital signing

- Asymmetric key cryptography is using key pairs, which have a mathematical relationship between them.
- One of the keys is called a private key and the another one is a public key.
- The private key is a huge random number.
- The public key is created from the private key using, for example, the mathematics of elliptic curves.
- The private key is kept secret, the public key is usually given to other people.

Digital signing

- A message encrypted with the private key can be decrypted using the public key.
- A message encrypted with a public key can be decrypted using the private key.

Digital signing

- How to sign with keys?
- Usually encrypting is done with the public key, so the receiver of the message can decrypt with his/her private key.
- With digital signatures, encrypting is done with the private key, so that ”everybody” can decrypt with the public key. The message is not ”secret”, but one can be sure who is the sender of the message.
- https://8gwifi.org/rsafunctions.jsp

Digital signing

- Alice has a message to Bob ”I, Alice, will pay 1 bitcoin to Bob.”
- Alice encrypts the SHA256 hash of the message using her private key.
- Alice sends the message, the encrypted message, and her public key for everybody to see.
- If the SHA256 hash of the message and the encrypted message decrypted with Alice’s public key are the same, the message is coming from Alice.
- The nodes of Bitcoin network are using the same principle to accept transactions.

Script language

- Bitcoin has its own Script language, which can be used for simple smart contracts. Bitcoin transactions themselves are a form of smart contracts.
- Script is Forth-like, stack-based, non-Turing-complete language, which is using reverse Polish notation.
- Examples of reverse Polish notation,
  3+4 is 34+
  5*3+4 is 534+*

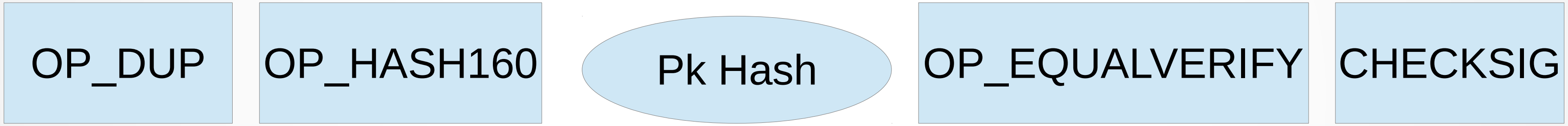
Script language

- The idea of a stack is LIFO (Last In First Out), so that a new book is usually pushed on the top of a book stack, and the topmost book is usually popped out from the stack first.
- 2+3=5 in Bitcoin Script language:

  Reverse Polish notation: 23+5=
  OP codes: OP_2 OP_3 OP_ADD OP_5 OP_EQUAL

Pay to the Public Key Hash (P2PKH) transaction

- Instructions and data, that Alice will give in the pubkey script of transaction #1.



OP_DUP

OP_HASH160

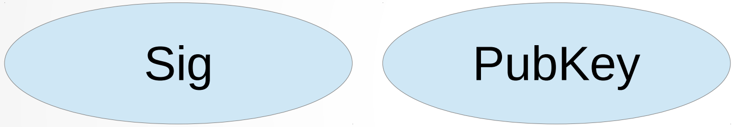
Pk Hash

OP_EQUALVERIFY

CHECKSIG

Pay to the Public Key Hash (P2PKH) transaction

- Data, that Bob will give in the signature script of transaction #2.

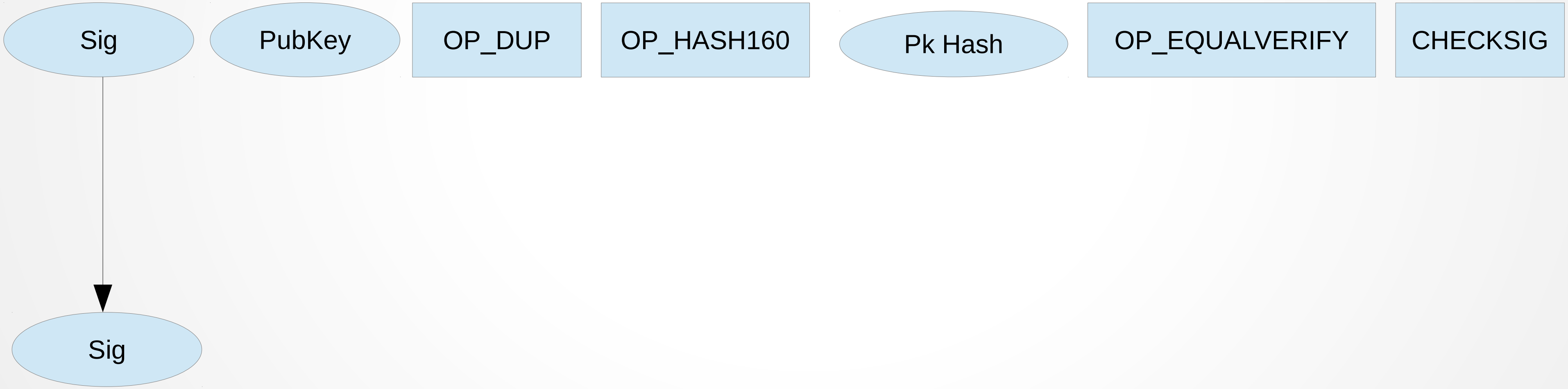


Sig

PubKey

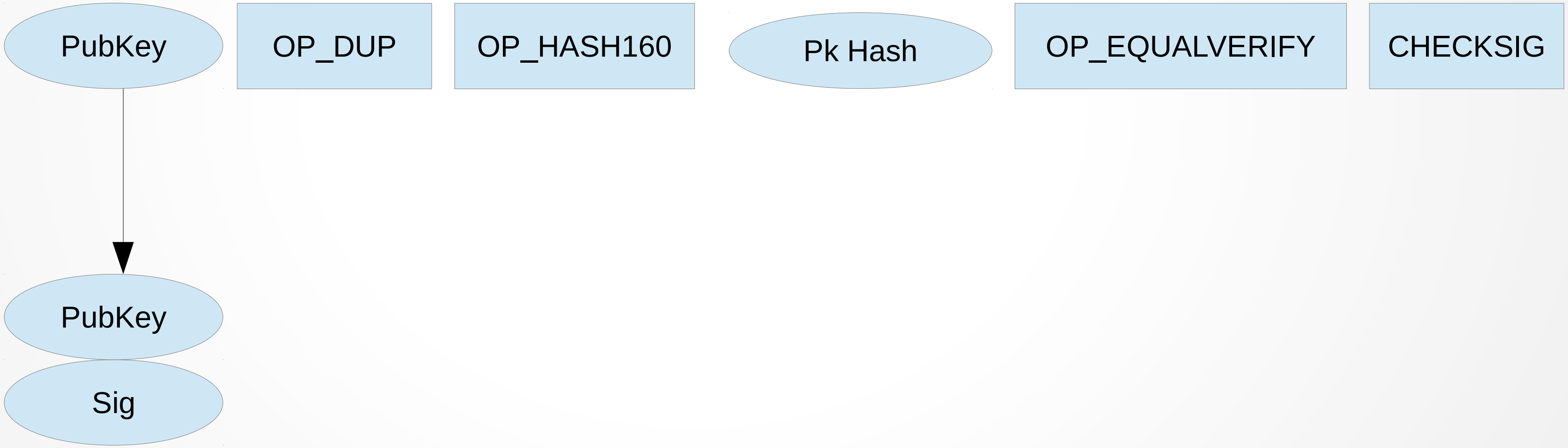
Pay to the Public Key Hash (P2PKH) transaction

- The signature of Bob’s signature script will be pushed to an empty stack. Because it is only data, nothing will be done to it.



Pay to the Public Key Hash (P2PKH) transaction

- The public key will be pushed on the signature.



PubKey

OP_DUP

OP_HASH160

Pk Hash

OP_EQUALVERIFY

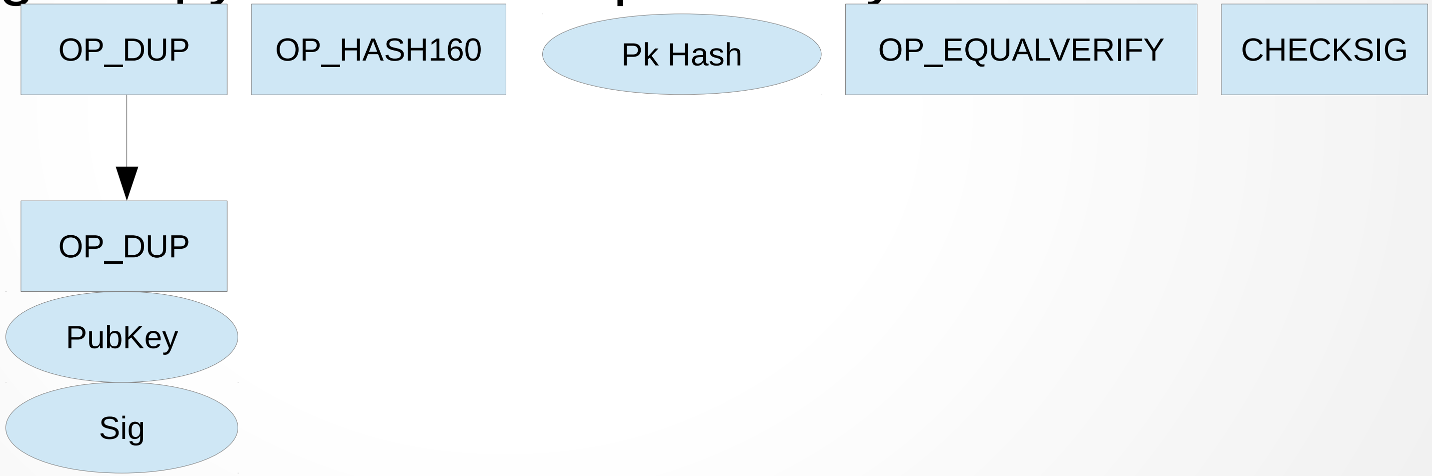
CHECKSIG

PubKey

Sig

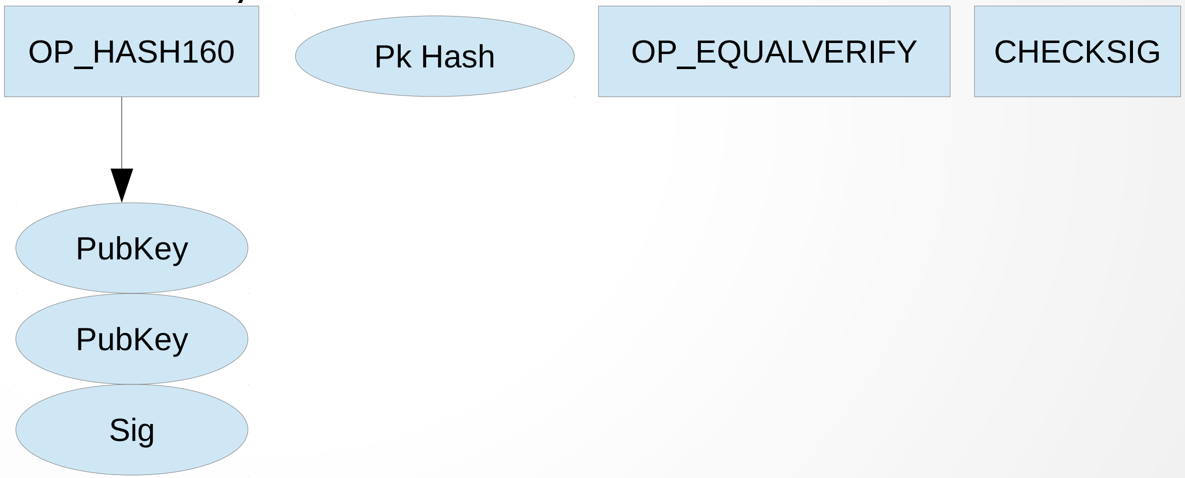
Pay to the Public Key Hash (P2PKH) transaction

- The OP_DUP operation of Alice’s pubkey script will be executed. OP_DUP will push a copy of the data located on the stack, making a copy of the Bob’s public key in this case.



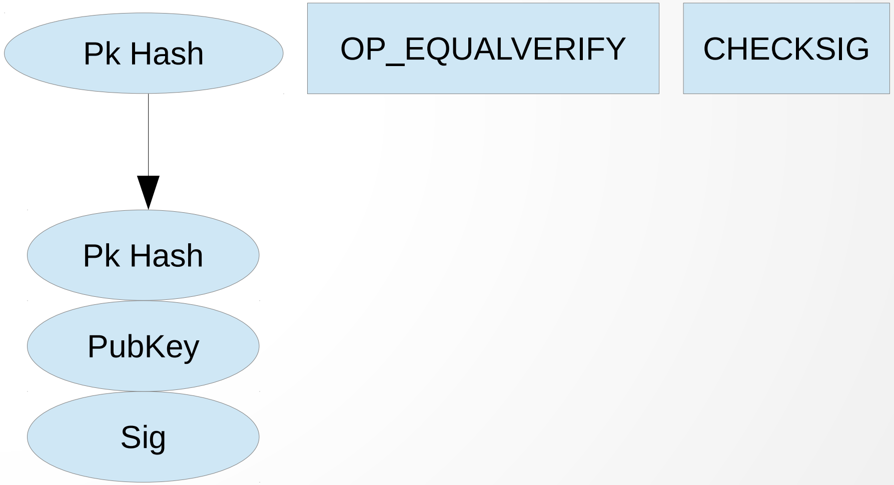
Pay to the Public Key Hash (P2PKH) transaction

- OP_HASH160 operation hashed the input twice: first using SHA-256, then using RIPEMD-160. The public key of Bob, which was on top of the stack, is now hashed.



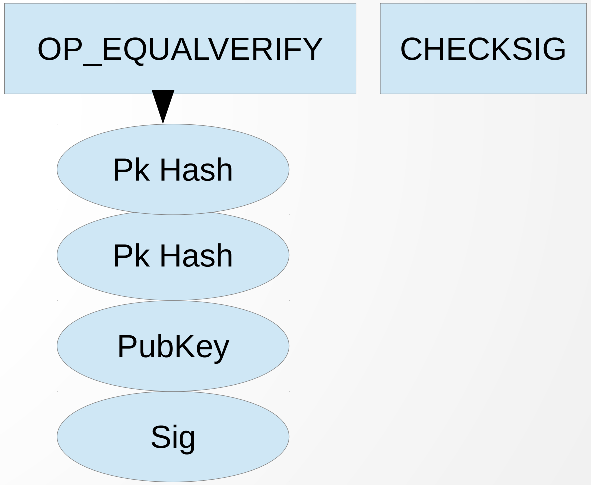
Pay to the Public Key Hash (P2PKH) transaction

- Now Alice’s pubkey script pushes pubkey hash, which Bob gave, on the stack. After this there should be two pieces of Bob’s pubkey hashes on the top of the stack.



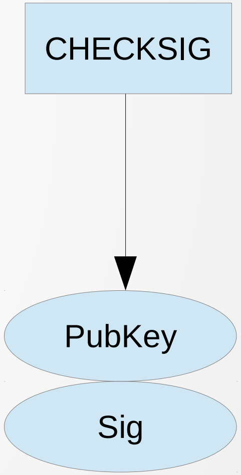
Pay to the Public Key Hash (P2PKH) transaction

- Now Alice's pubkey script will execute the OP_EQUALVERIFY operation, which is the same as executing the OP_EQUAL and OP_VERIFY operations in a row.
- OP_EQUAL compares the two topmost values in stack. It pops the values out of the stacks and replaces them with the result of the comparison: 0 (false) or 1 (true).
- OP_VERIFY checks the topmost value in stack. If value is false, the execution end and the validation of the transaction fails. Otherwise, the value of true is popped out of the stack.



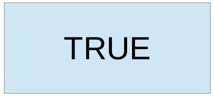
Pay to the Public Key Hash (P2PKH) transaction

- Finally Alice’s pubkey script will execute the OP_CHECKSIG operation, which checks Bob’s signature and Bob’s publick key. If the value of this comparison is true, that truth value will be pushed on the stack.



Pay to the Public Key Hash (P2PKH) transaction

- If the value False is not on the stack after the execution of the pubkey script, the transaction is valid.



TRUE