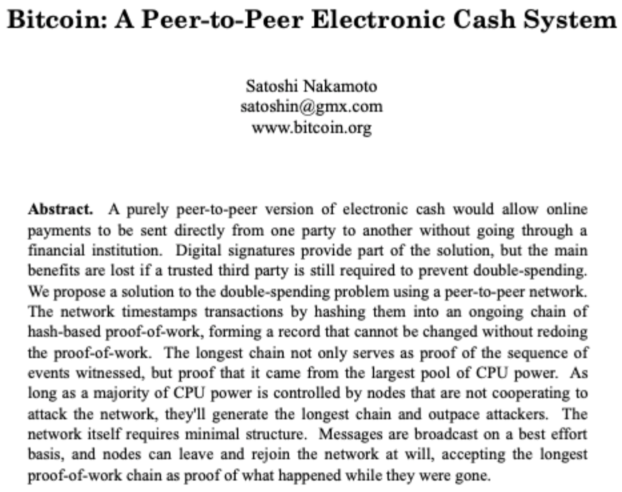
THE BLOCKCHAIN IMITATION GAME

Kaihua Qin, Stefanos Chaliasos, Liyi Zhou, Benjamin Livshits, Dawn Song, Arthur Gervais

Imperial College London

Berkeley UNIVERSITY OF CALIFORNIA

UCL



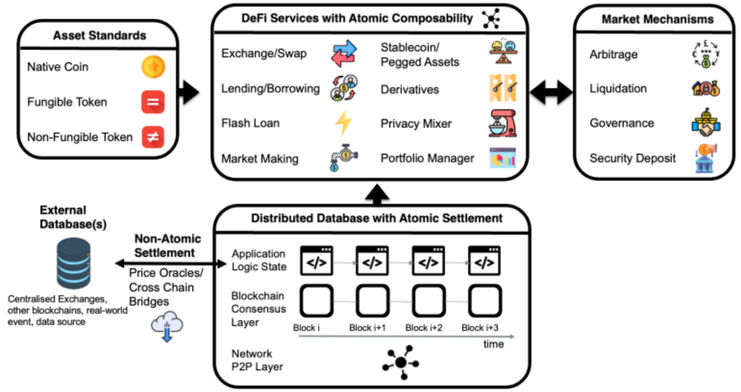
Bitcoin: A Peer-to-Peer Electronic Cash System

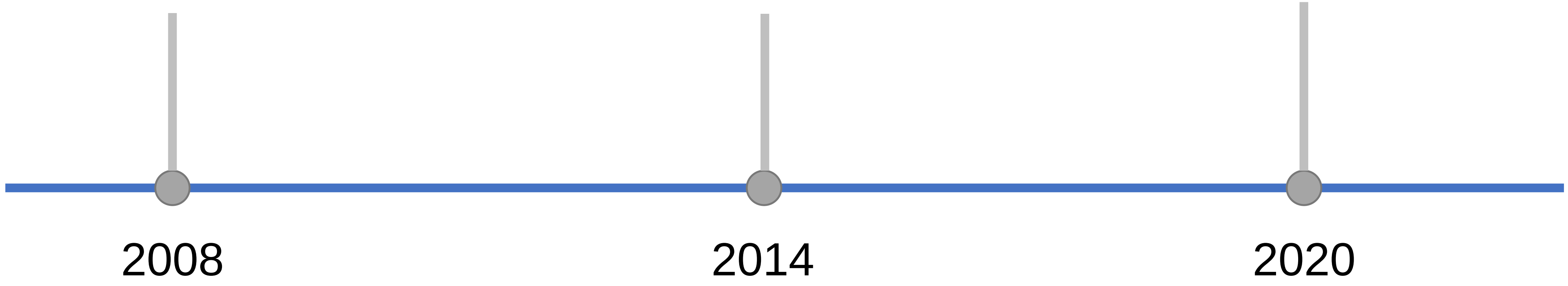
Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

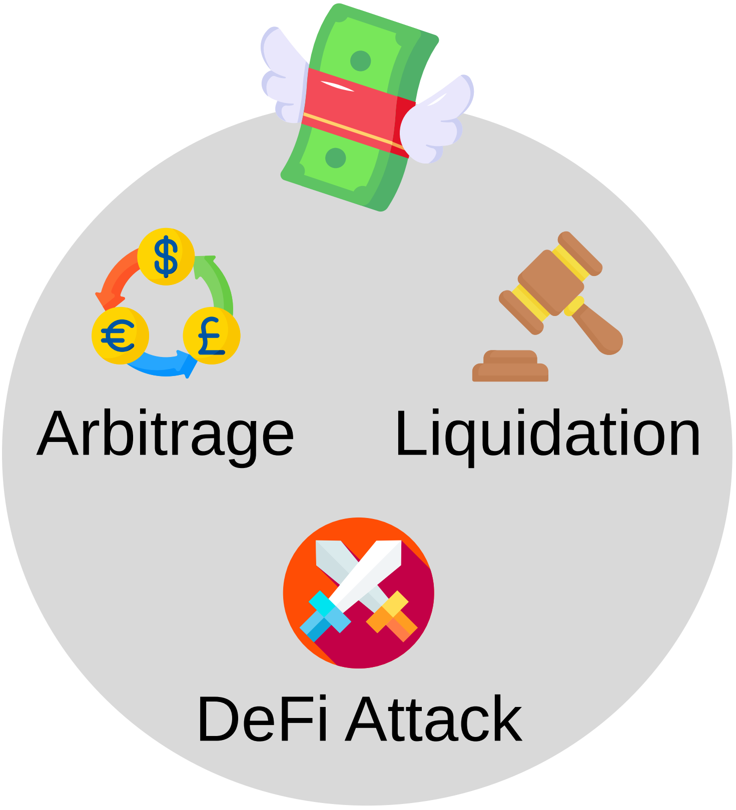
ethereum

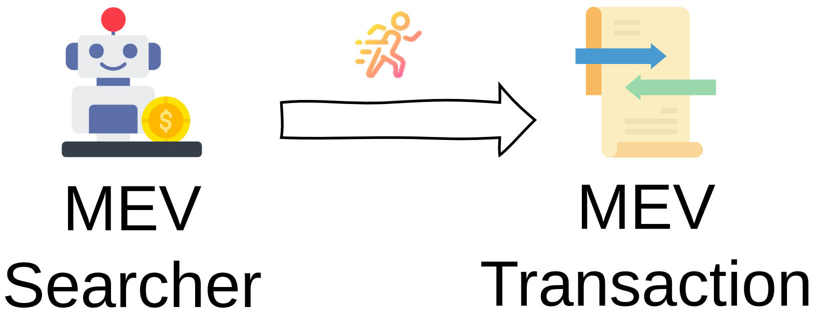
Decentralized Finance





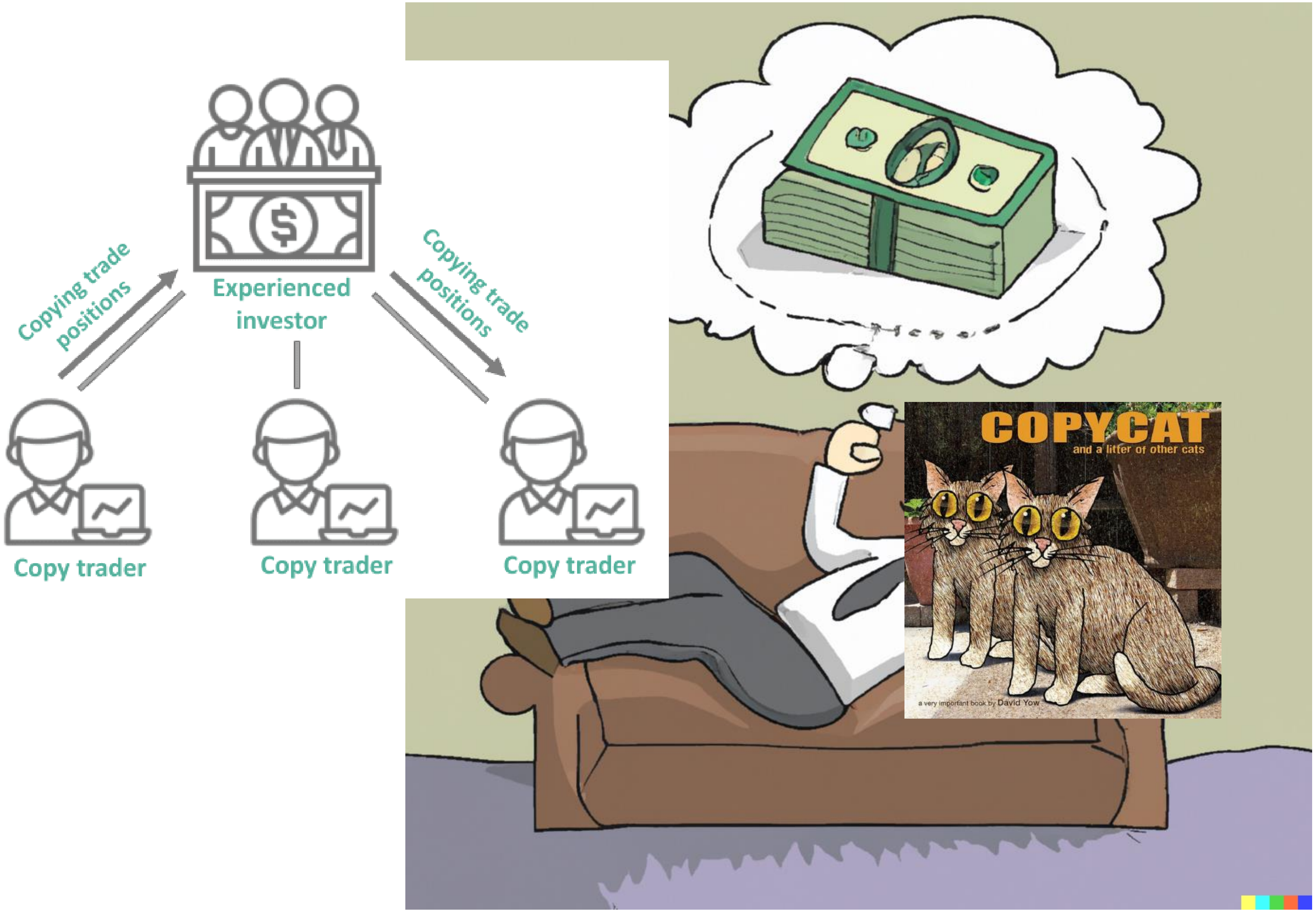
MEV Extraction



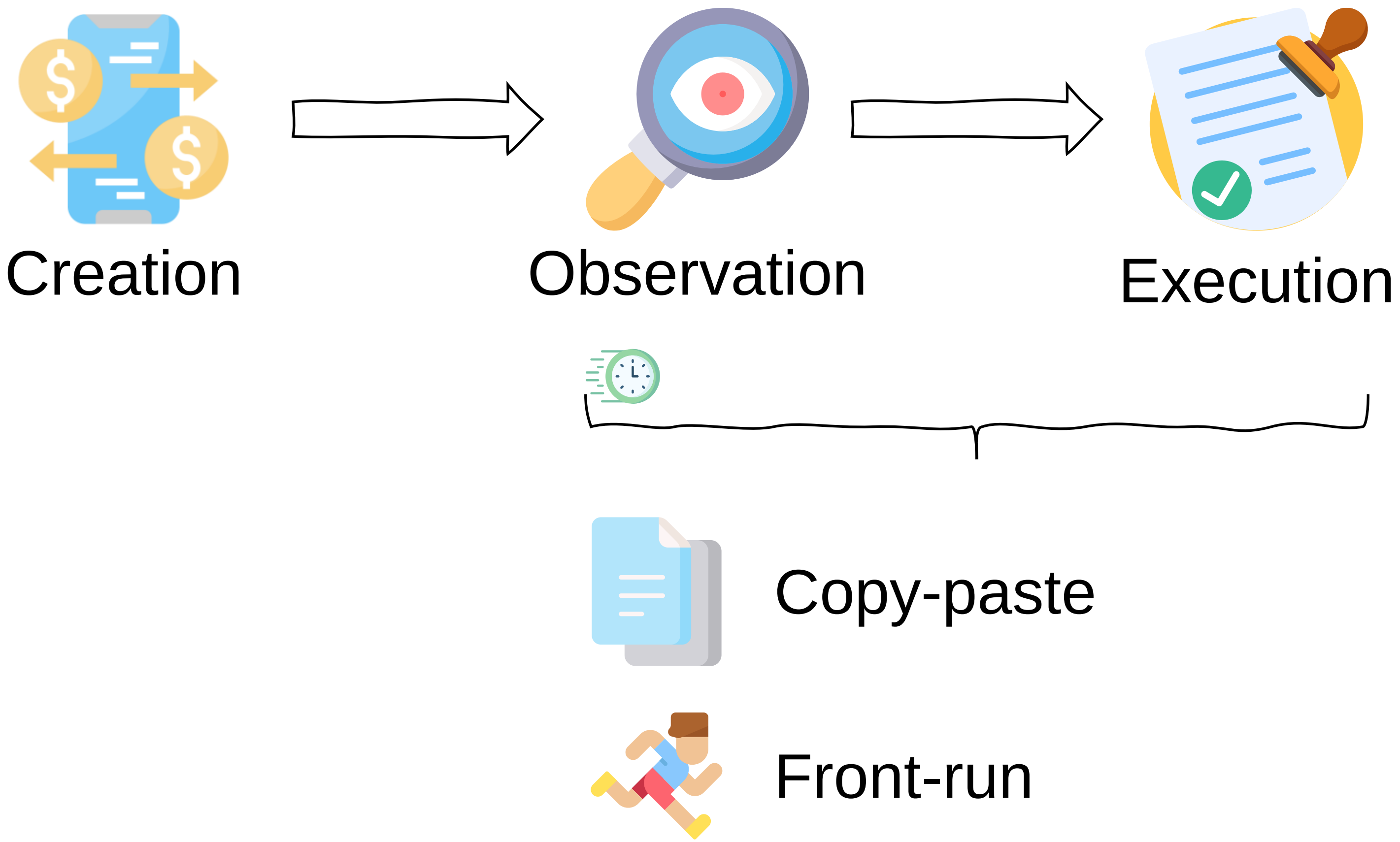


```
contract MEV {

 function arb(uint x, uint y) public {

  swapETHtoUSDC(x);

  swapUSDCtoETH(y);

  msg.sender.transfer(profit);
 }
}
```

The Blockchain Imitation Game



Imitation



Creation

Observation

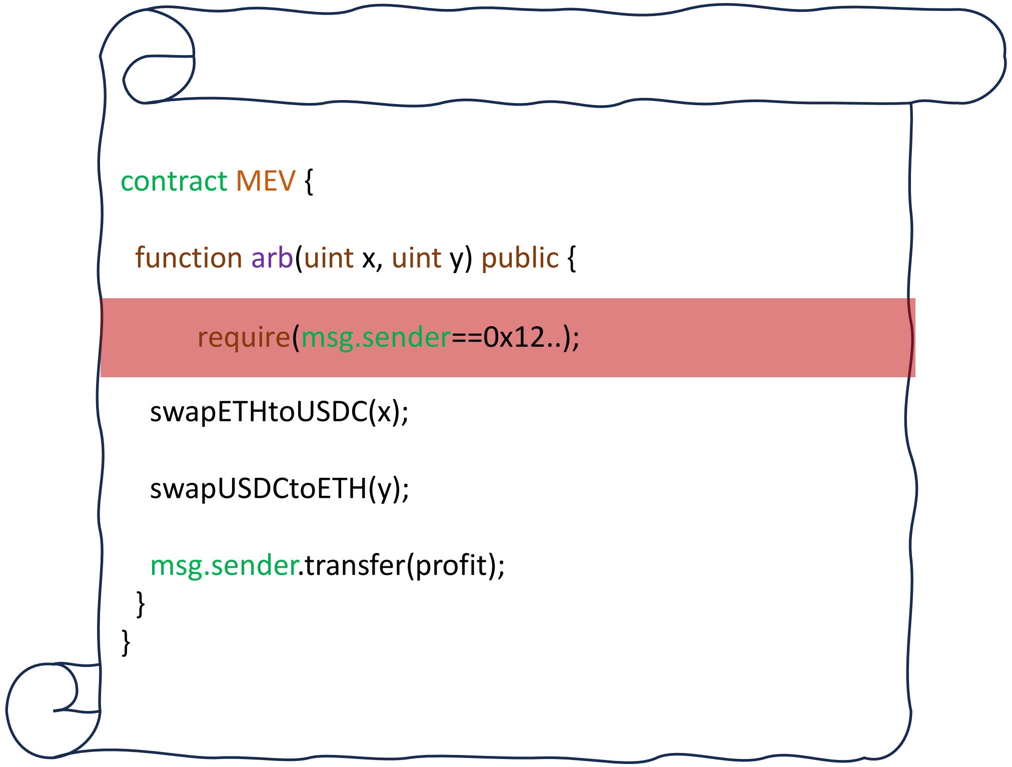
Execution

Copy-paste

Front-run

Naive Imitation

- Blind duplicate & string replacement
- Verify locally & front-run
- Simple but effective
- 35M USD (December 2018 – August 2021) on Ethereum
- Easy to prevent



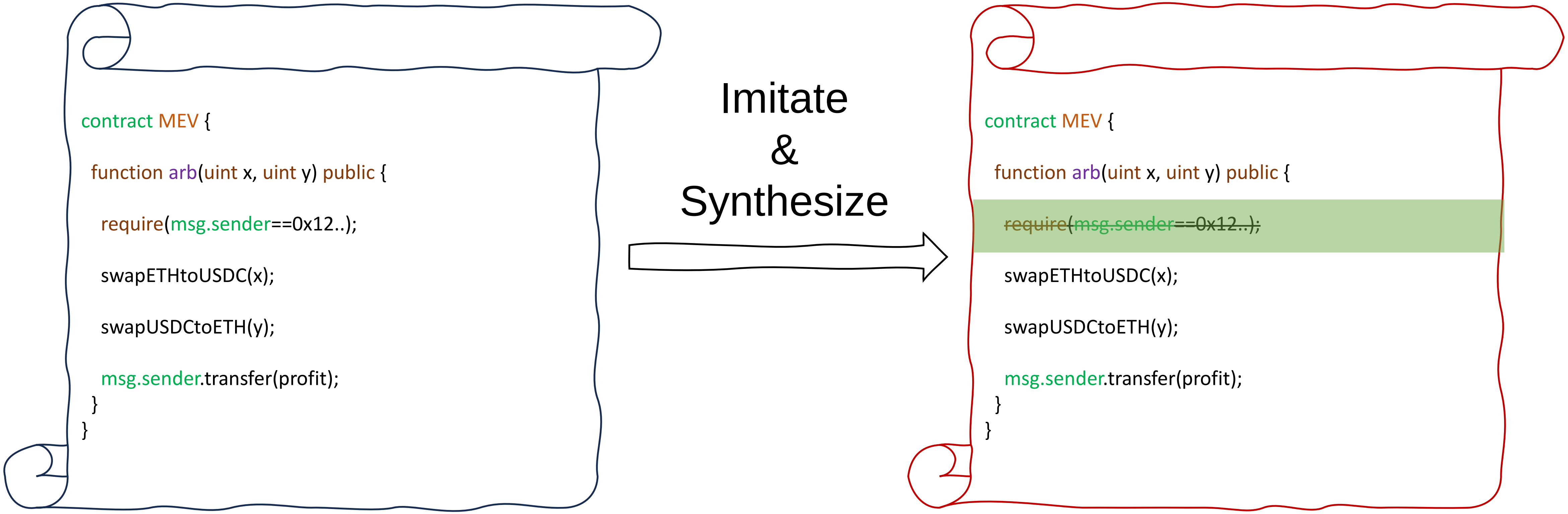
```
contract MEV {
 function arb(uint x, uint y) public {
      require(msg.sender==0x12..);
   swapETHtoUSDC(x);
   swapUSDCtoETH(y);
   msg.sender.transfer(profit);
 }
}
```

Qin K, Zhou L, Gervais A. Quantifying blockchain extractable value: How dark is the forest?. In 2022 IEEE Symposium on Security and Privacy (SP) 2022 May 22 (pp. 198-214). IEEE.

the CAFE

Cartoon-Box #13

Ape — Generalized Imitation



```
contract MEV {

 function arb(uint x, uint y) public {

  require(msg.sender==0x12..);

  swapETHtoUSDC(x);

  swapUSDCtoETH(y);

  msg.sender.transfer(profit);
 }
}
```

Imitate

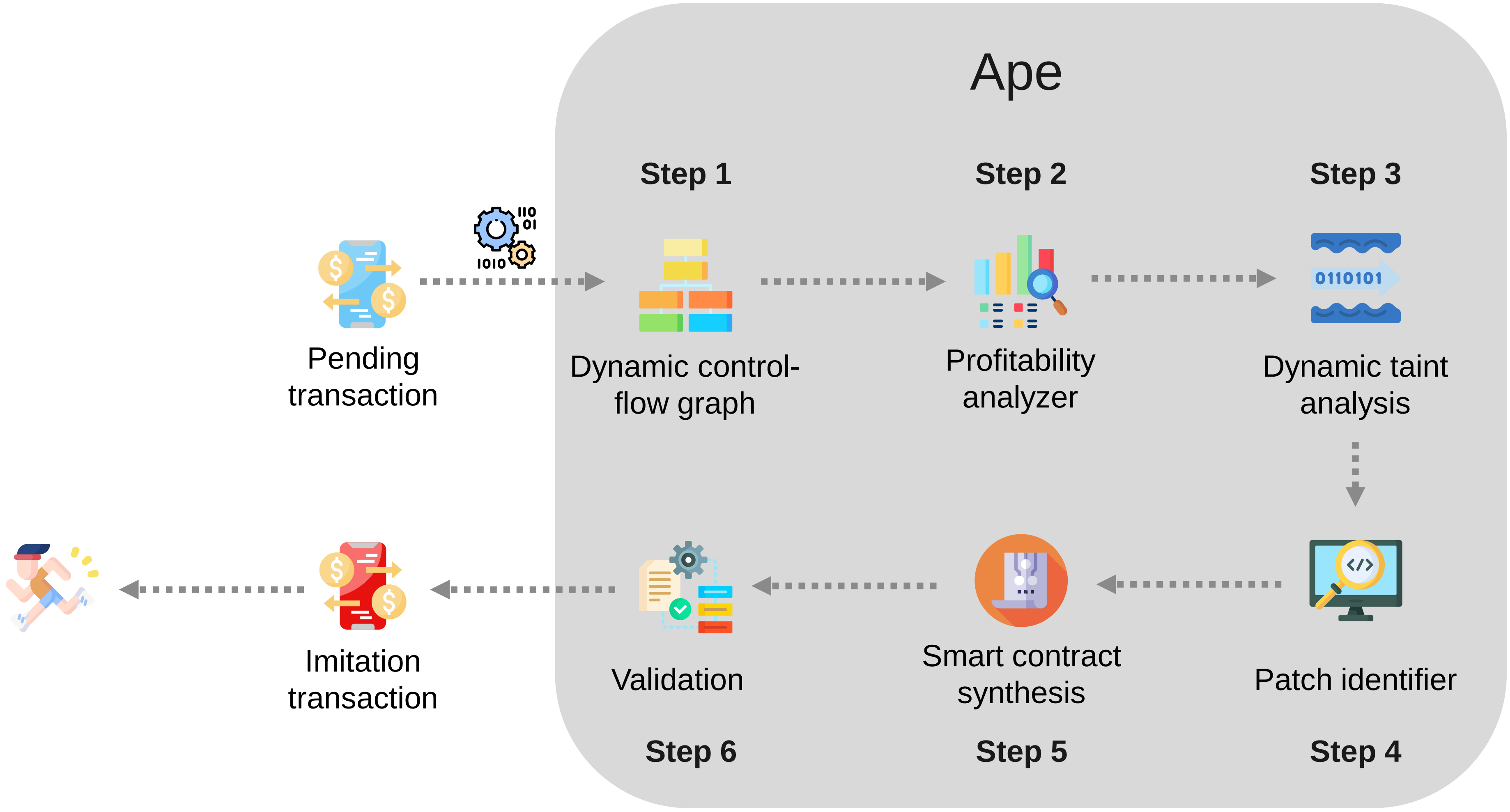
&

Synthesize

```
contract MEV {

 function arb(uint x, uint y) public {

  ~~require(msg.sender==0x12..);~~

  swapETHtoUSDC(x);

  swapUSDCtoETH(y);

  msg.sender.transfer(profit);
 }
}
```

Qin K, Chaliasos S, Zhou L, Livshits B, Song D, Gervais A. The blockchain imitation game. (2023). In 32nd USENIX Security Symposium (USENIX Security 23) (pp. 3961-3978).

Ape Overview



Ape

Step 1

Dynamic control-flow graph

Step 2

Profitability analyzer

Step 3

Dynamic taint analysis

Step 4

Patch identifier

Step 5

Smart contract synthesis

Step 6

Validation

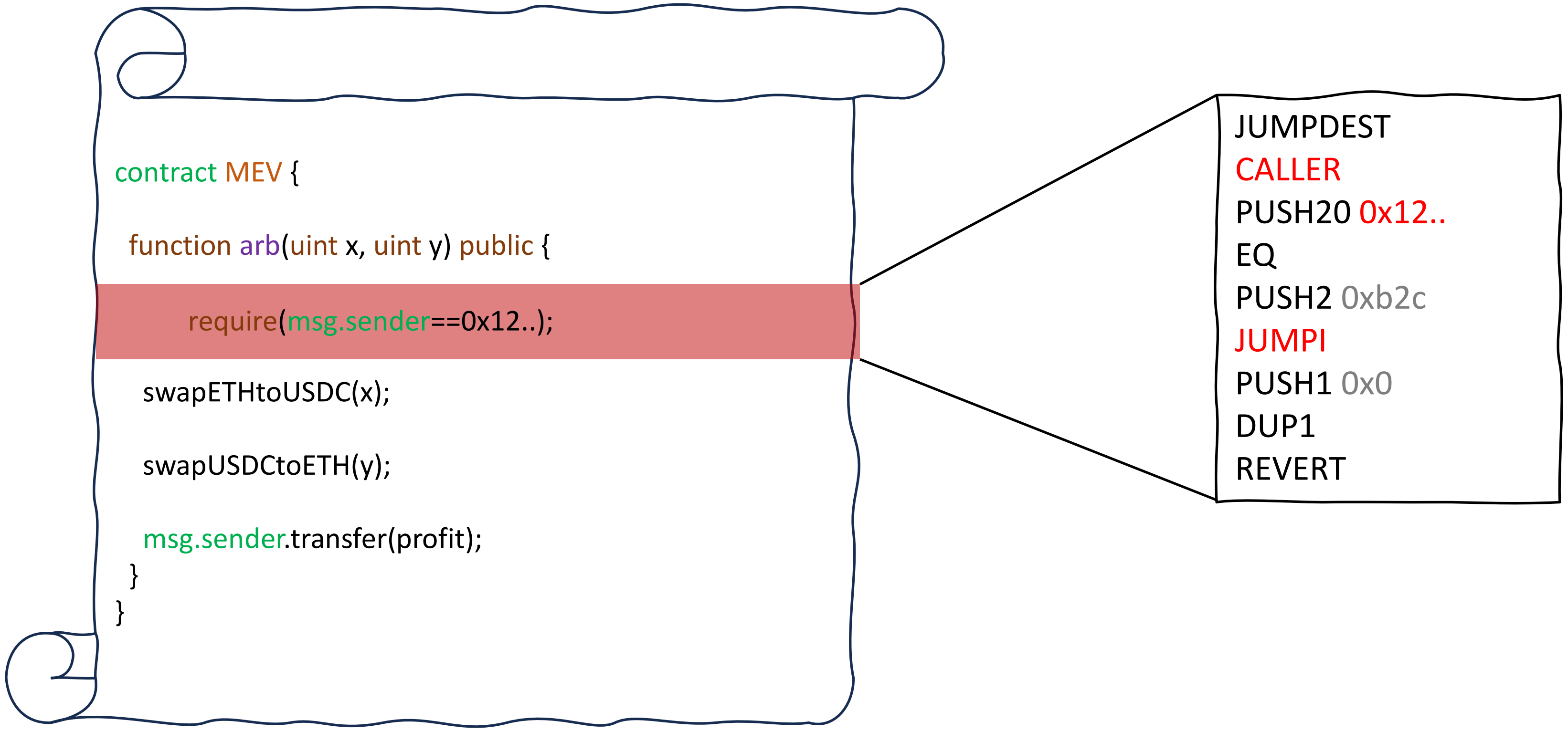
Pending transaction

Imitation transaction

Dynamic Taint Analysis

Why does a naive imitation fail?

difference (e.g., transaction sender) → conditional jump → different execution path

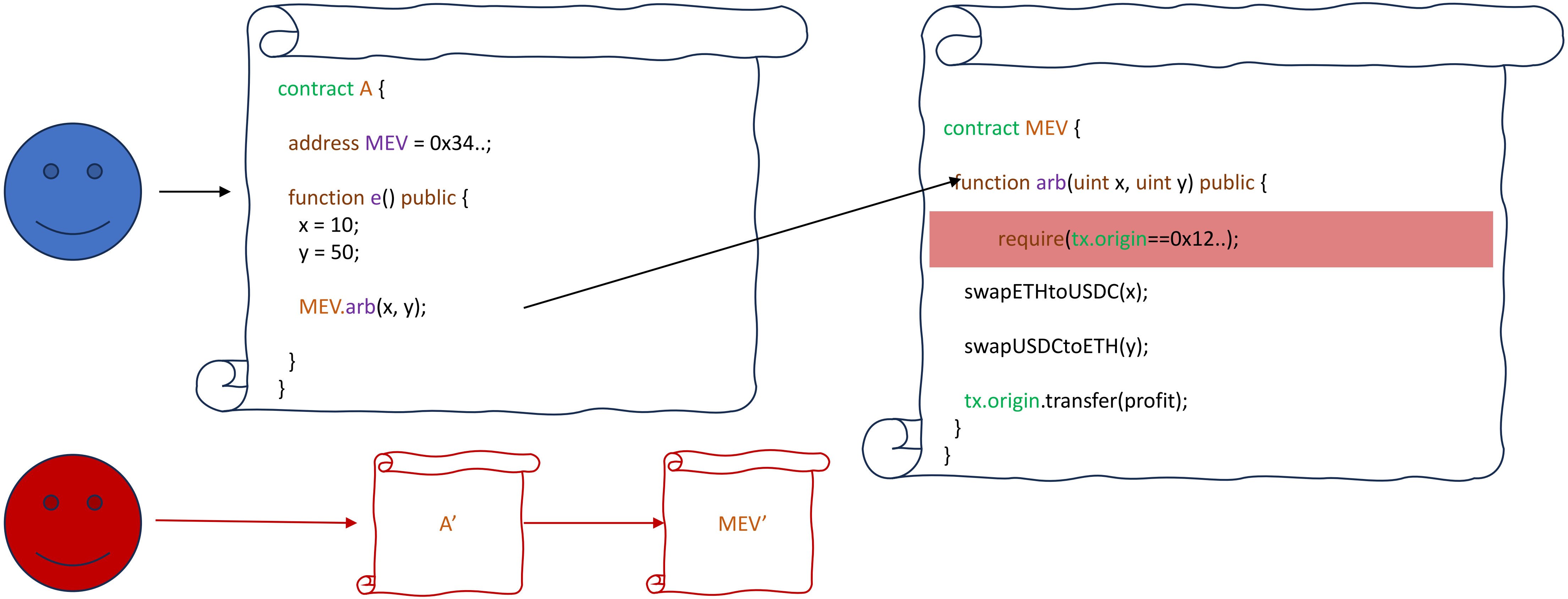


```
contract MEV {

 function arb(uint x, uint y) public {

     require(msg.sender==0x12..);

   swapETHtoUSDC(x);

   swapUSDCtoETH(y);

   msg.sender.transfer(profit);
 }
}
```

```
JUMPDEST
CALLER
PUSH20 0x12..
EQ
PUSH2 0xb2c
JUMPI
PUSH1 0x0
DUP1
REVERT
```

Patch Identifier

Ensure synthesized contract(s) are probably invoked

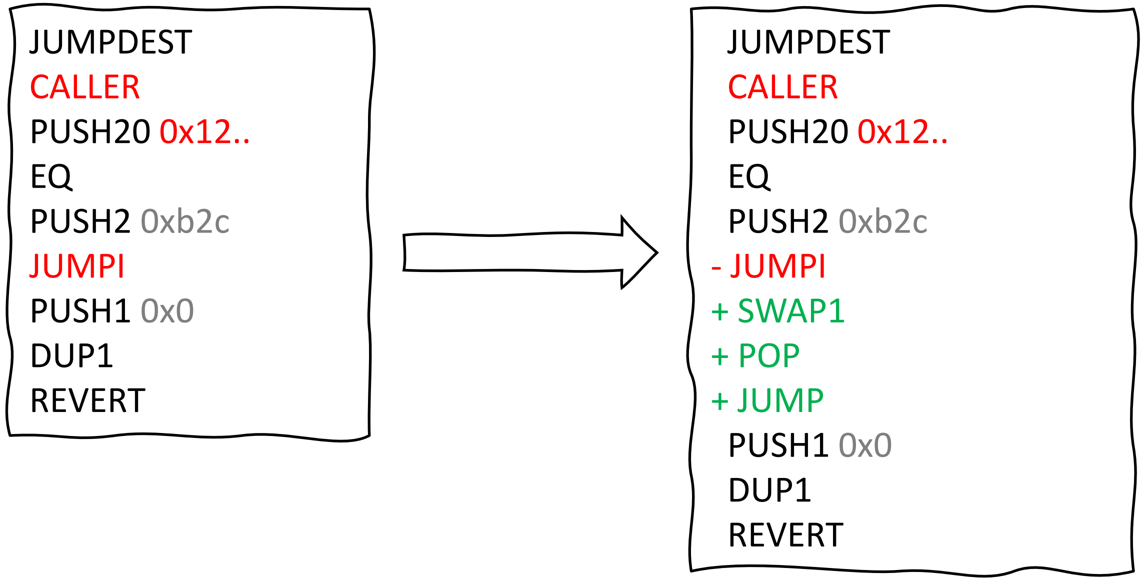


```
contract A {

  address MEV = 0x34..;

  function e() public {
    x = 10;
    y = 50;

    MEV.arb(x, y);

  }
}
```

```
contract MEV {

 function arb(uint x, uint y) public {

     require(tx.origin==0x12..);

   swapETHtoUSDC(x);

   swapUSDCtoETH(y);

   tx.origin.transfer(profit);
 }
}
```

Contract Synthesis

Copy executed bytecode with amendments



```
JUMPDEST
CALLER
PUSH20 0x12..
EQ
PUSH2 0xb2c
JUMPI
PUSH1 0x0
DUP1
REVERT
```

```
  JUMPDEST
  CALLER
  PUSH20 0x12..
  EQ
  PUSH2 0xb2c
- JUMPI
+ SWAP1
+ POP
+ JUMP
  PUSH1 0x0
  DUP1
  REVERT
```

- JUMPI Forcing
- Invocation Redirection
- Storage Recovery
- Asset Transfer Redirection

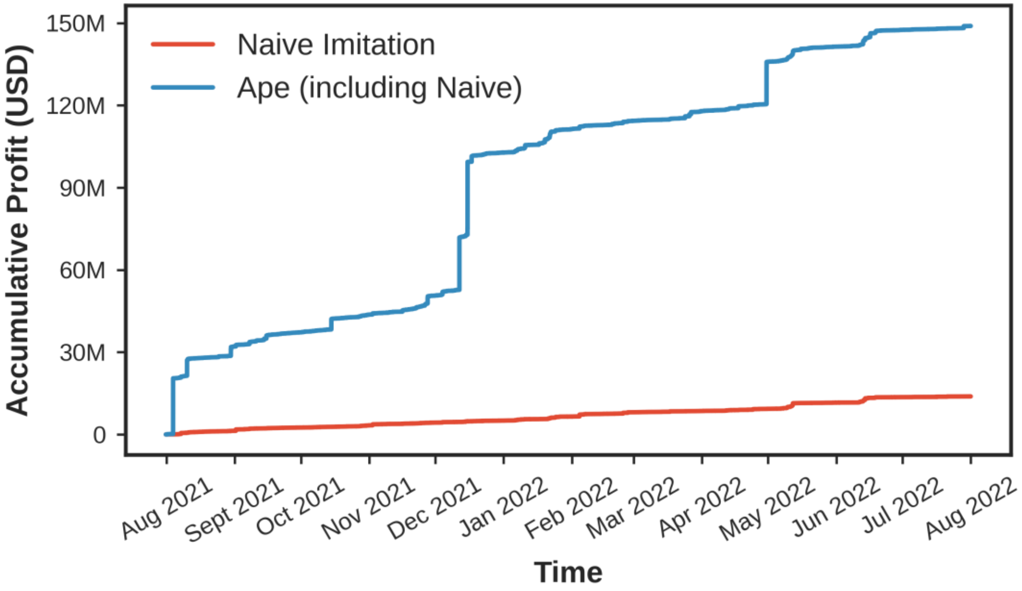
Ape Evaluation

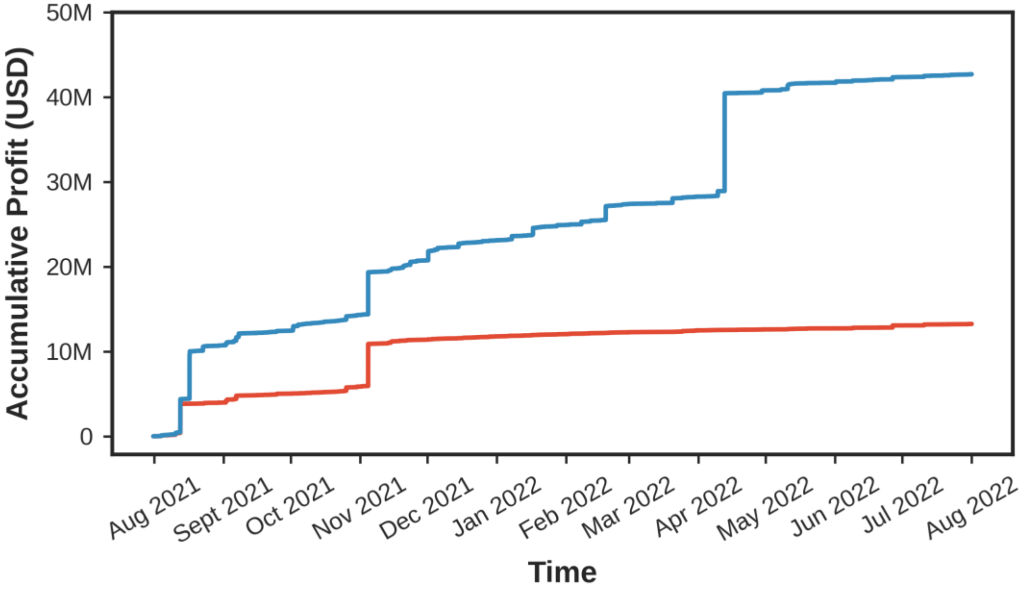
August 1, 2021 – July 31, 2022 (1 year)

0.07 second

Ethereum 148.96M USD

BSC 42.70M USD





Imitation as Whitehat

DeFi Attacks

Ethereum

29 73.74M USD

Protocol	Loss (USD)	Date
Popsicle Finance	20.25M	Aug-03-2021
Saddle Finance	9.71M	Apr-30-2022
Indexed Finance	3.58M	Oct-14-2021
…	…	…

BSC

40 22.39M USD

Protocol	Loss (USD)	Date
Elephant Money	11.52M	Apr-12-2022
XSURGE	5.17M	Aug-16-2021
CollectCoin	1.06M	Dec-01-2021
…	…	…

Questions

Kaihua Qin

https://qin.ac/