Transparent Dishonesty: Front-running Attacks on Blockchain

Shayan Eskandari
ConsenSys Diligence
Concordia University

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

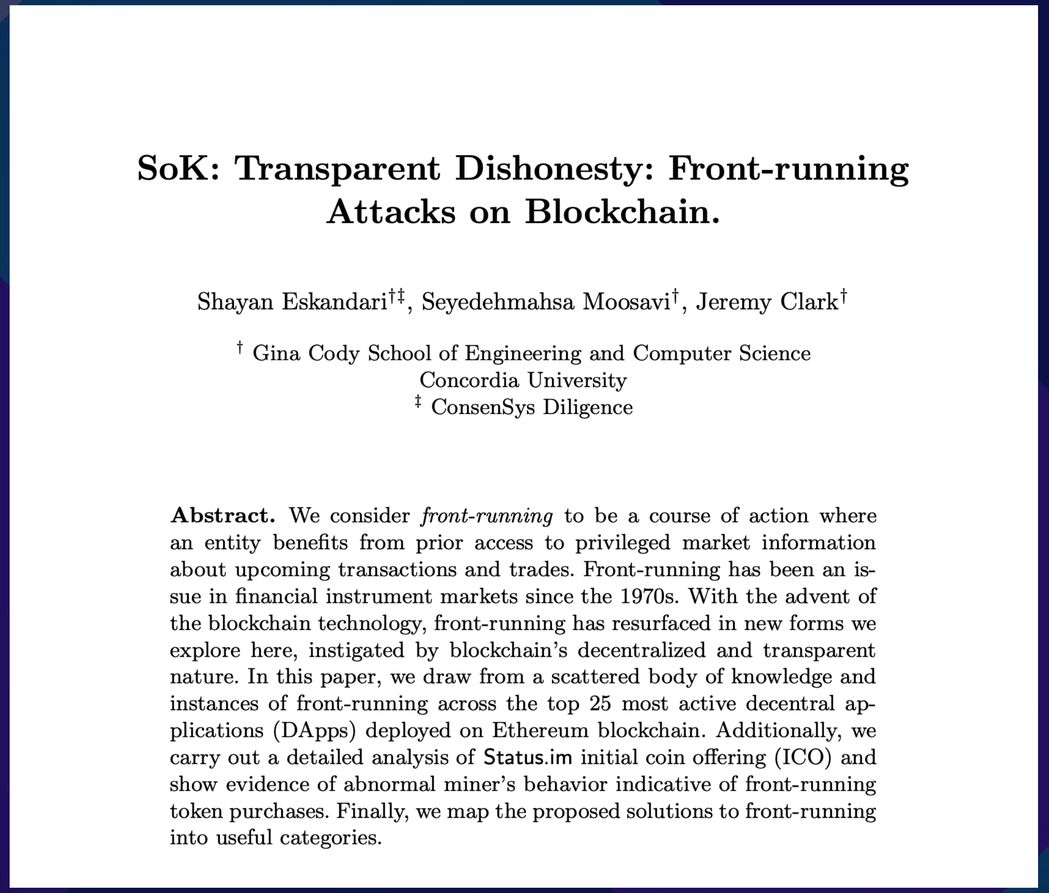
Introduction

- Shayan Eskandari
  - PhD Candidate at Concordia University, Canada, working with Jeremy Clark
  - Security Engineer and smart contract auditor at ConsenSys Diligence

https://shayan.es

Introduction

- Published at
  Workshop on Trusted Smart Contracts
  Financial Cryptography (FC) 2019 - St. Kitts
- Systematization of Knowledge (SoK)



SoK: Transparent Dishonesty: Front-running Attacks on Blockchain.

Shayan Eskandari[†‡], Seyedehmahsa Moosavi[†], Jeremy Clark[†]

[†] Gina Cody School of Engineering and Computer Science
Concordia University
[‡] ConsenSys Diligence

Abstract. We consider *front-running* to be a course of action where an entity benefits from prior access to privileged market information about upcoming transactions and trades. Front-running has been an issue in financial instrument markets since the 1970s. With the advent of the blockchain technology, front-running has resurfaced in new forms we explore here, instigated by blockchain's decentralized and transparent nature. In this paper, we draw from a scattered body of knowledge and instances of front-running across the top 25 most active decentral applications (DApps) deployed on Ethereum blockchain. Additionally, we carry out a detailed analysis of Status.im initial coin offering (ICO) and show evidence of abnormal miner's behavior indicative of front-running token purchases. Finally, we map the proposed solutions to front-running into useful categories.

https://arxiv.org/abs/1902.05164

Story Time

Borrowed from BBC - by Aart-Jan Venema [1]

Story 1: ICO

Story 1: ICO

The workflow “to the moon”:

Story 1: ICO

The workflow “to the moon”:

- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

The workflow “to the moon”:

- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

The workflow “to the moon”:

- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

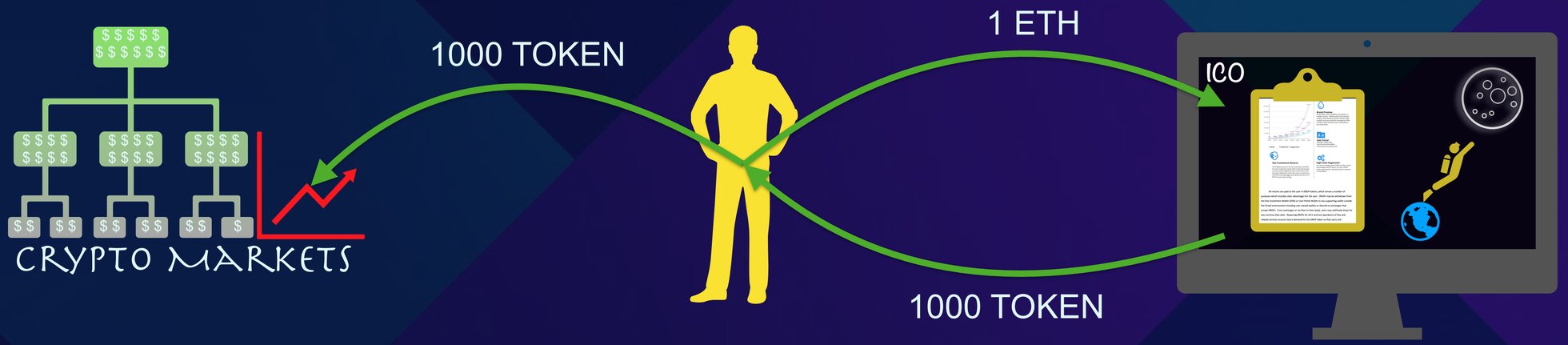
Story 1: ICO

The workflow “to the moon”:

- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

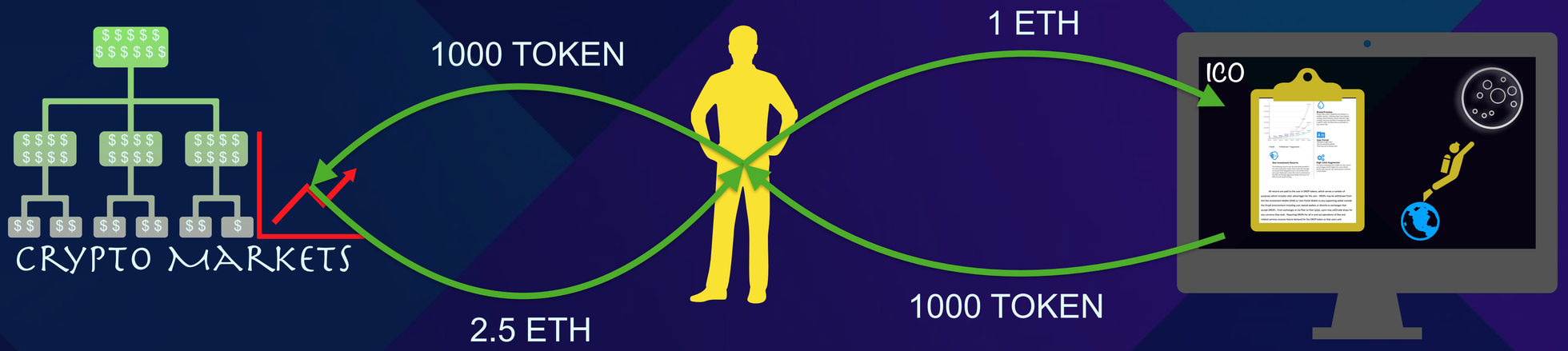
The workflow “to the moon”:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

The workflow “to the moon”:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

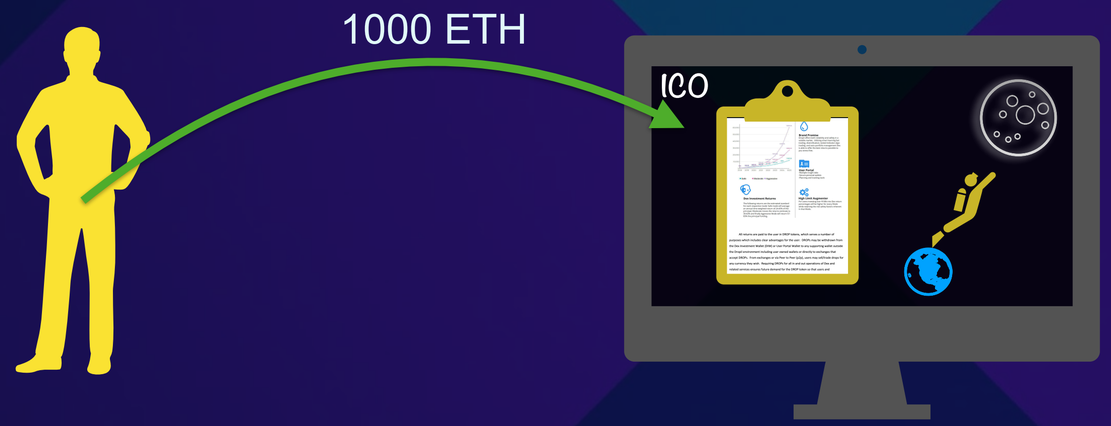
Story 1: ICO

The problem:

- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

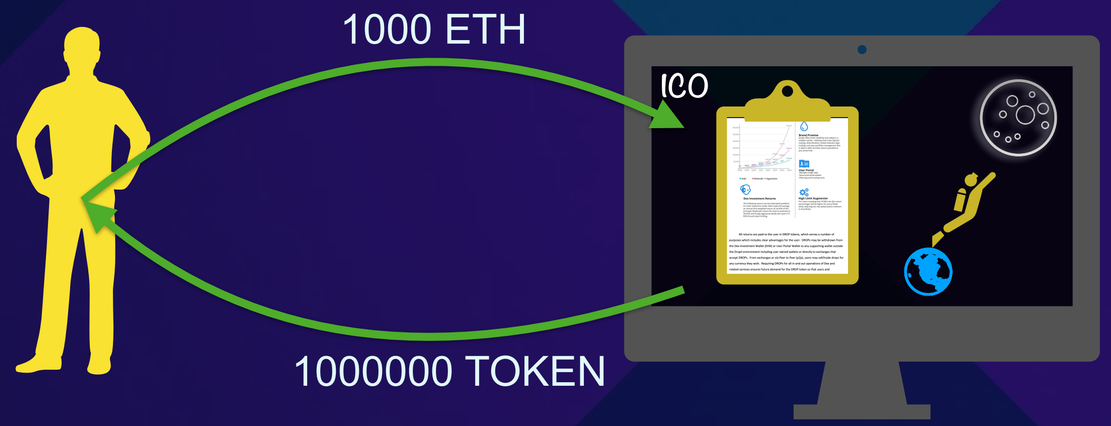
The problem:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

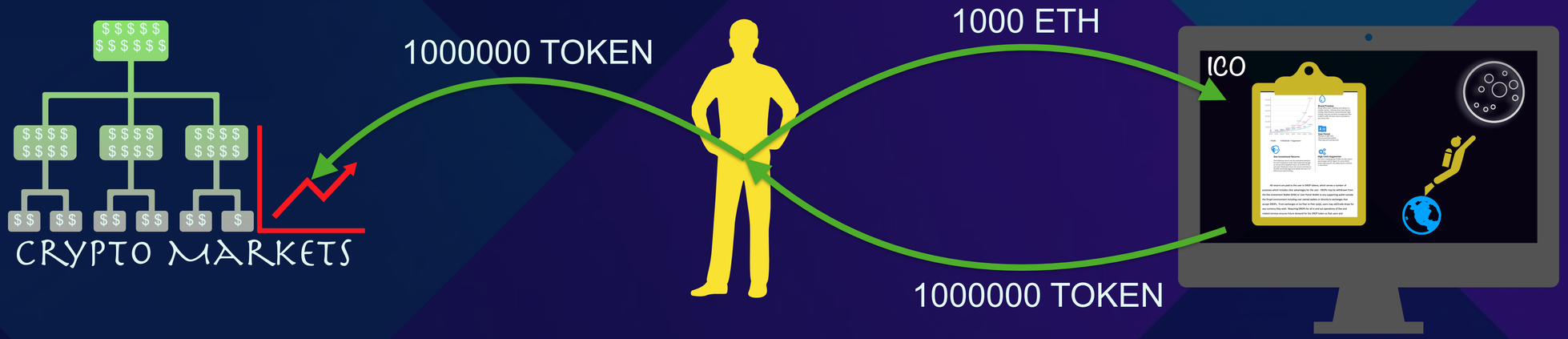
The problem:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

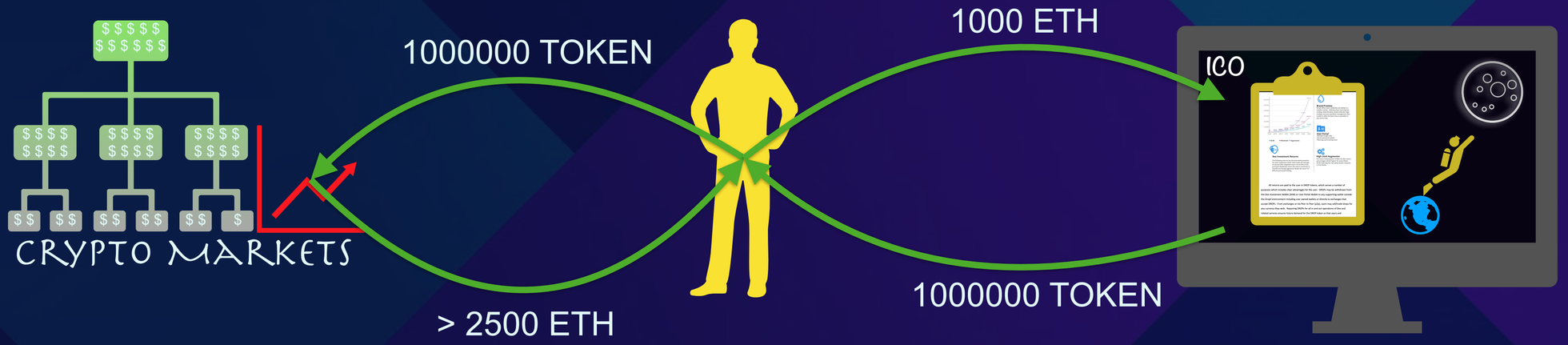
The problem:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: ICO

The problem:



- ERC20
  - Tradable
- Limited Time
- Soft/Hard Cap

Story 1: Status ICO

Story 1: Status ICO

Make ICOs fair again:

Story 1: Status ICO

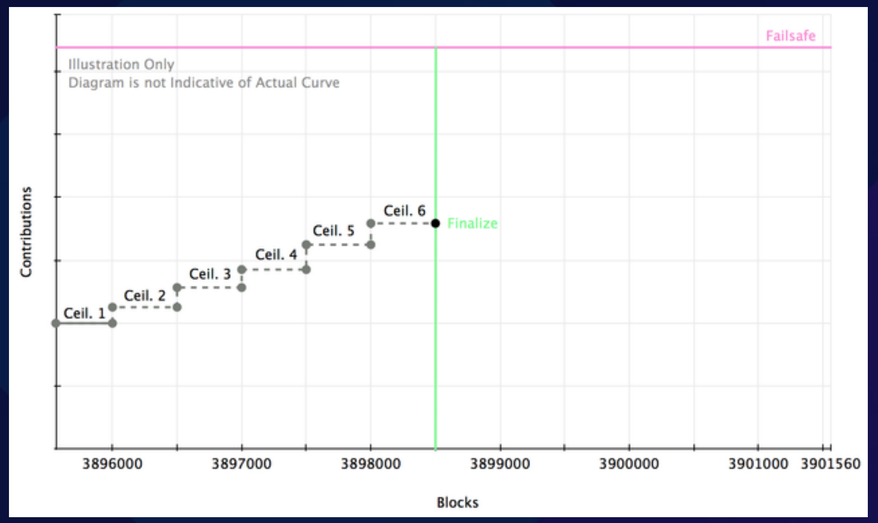
Make ICOs fair again:

- Limit GasPrice // require(gasPrice < 50 gwei)

Story 1: Status ICO

Make ICOs fair again:

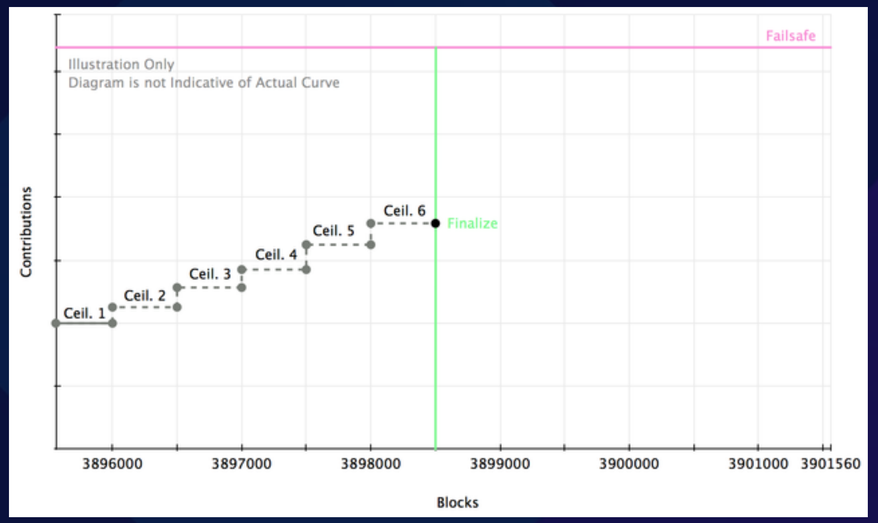
- Limit GasPrice // require(gasPrice < 50 gwei)
- Dynamic Cap/Ceiling // Maximum deposit amount per ceiling

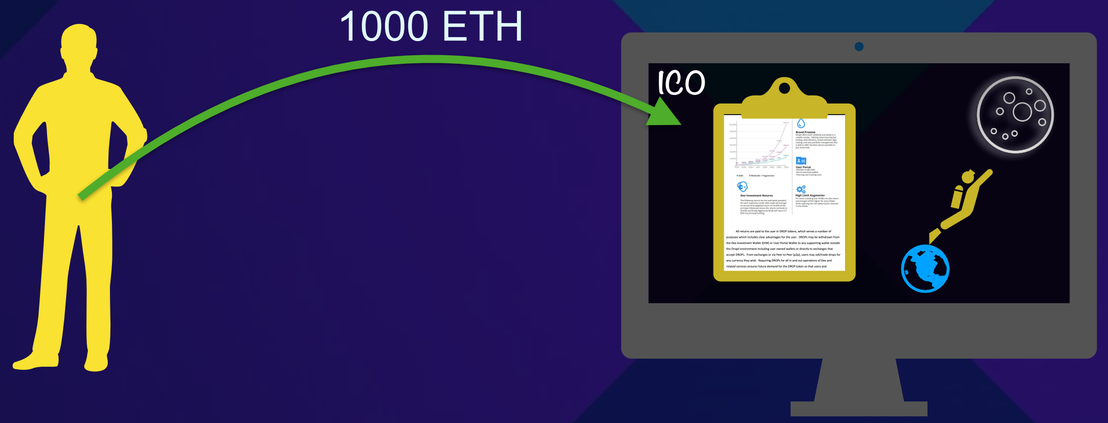


Story 1: Status ICO

Make ICOs fair again:

- Limit GasPrice // require(gasPrice < 50 gwei)
- Dynamic Cap/Ceiling // Maximum deposit amount per ceiling





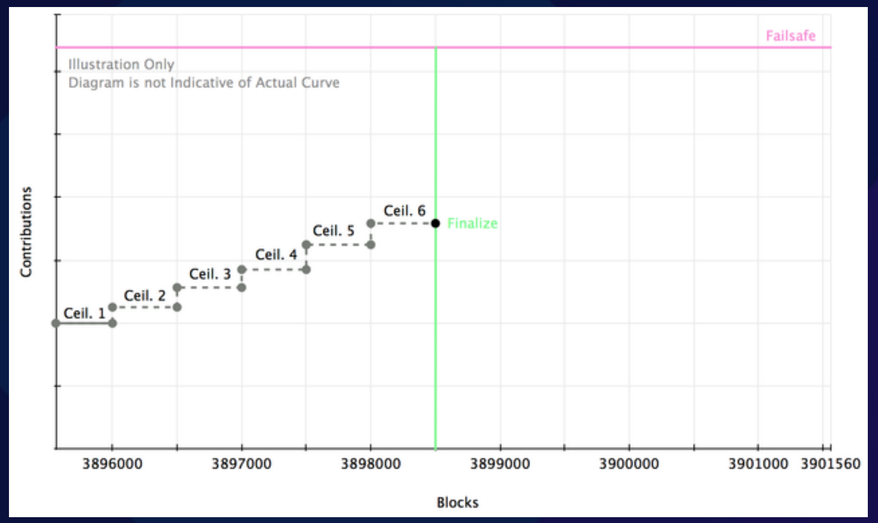
CONSENSYS Diligence

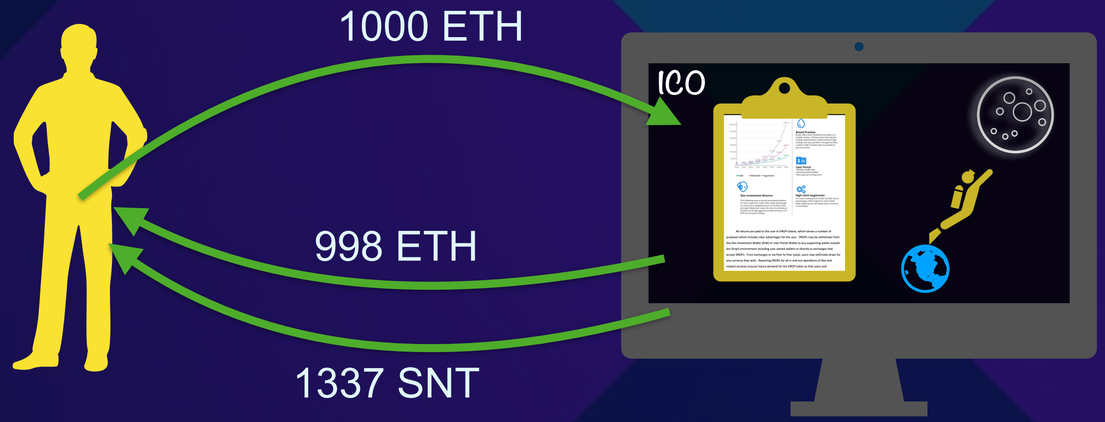
UNIVERSITÉ Concordia UNIVERSITY

Story 1: Status ICO

Make ICOs fair again:

- Limit GasPrice // require(gasPrice < 50 gwei)
- Dynamic Cap/Ceiling // Maximum deposit amount per ceiling





Story 1: Status ICO

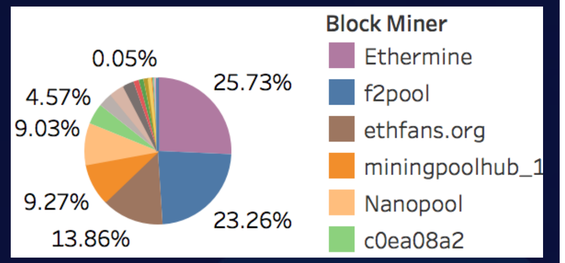
- June 2017
- Raised: ~300,000 Eth in 16 hours
- Refunded 111,161 attempts
  - Total of: 347,154 ETH

Story 1: Status ICO

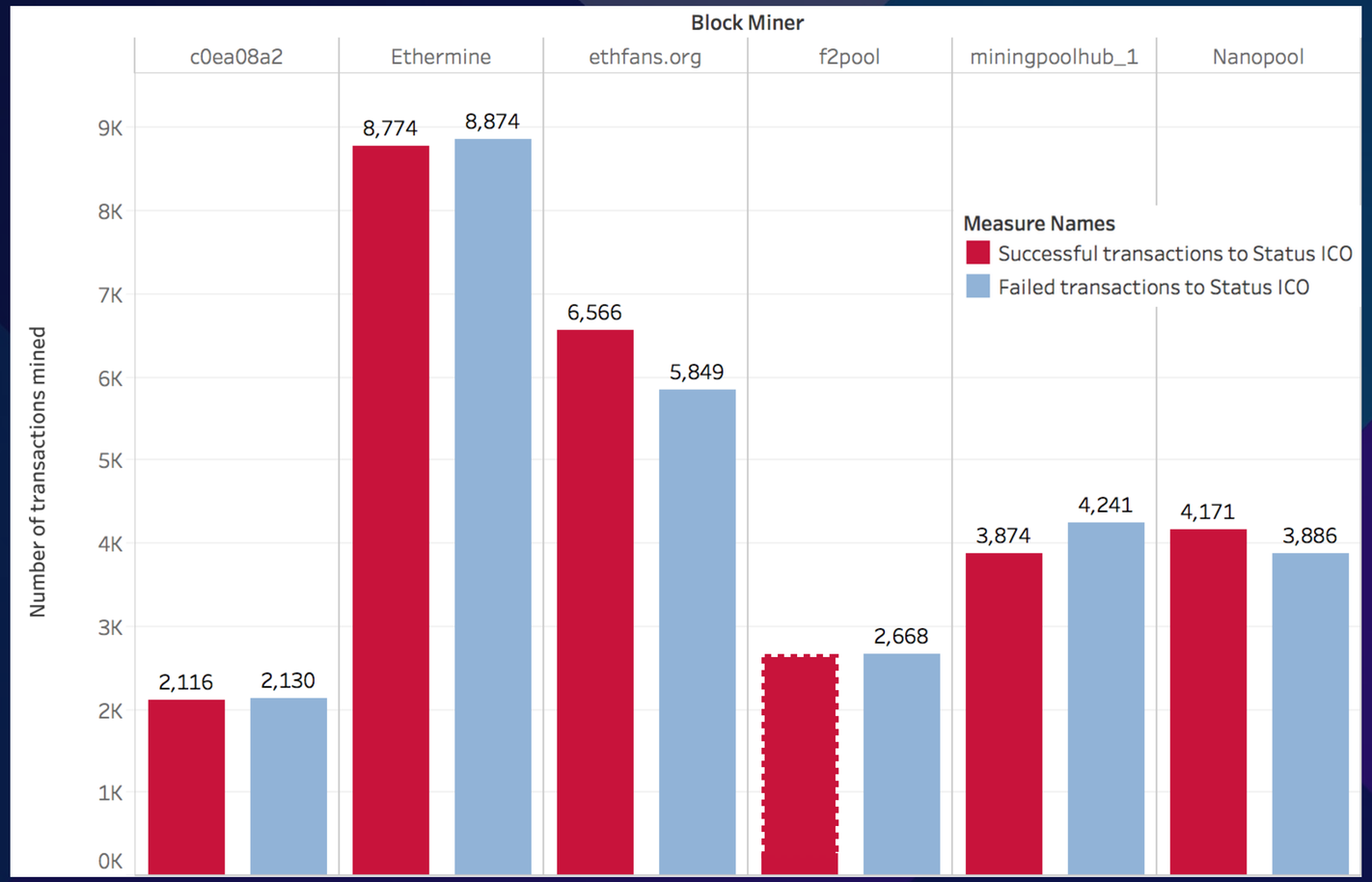
- June 2017
- Raised: ~300,000 Eth in 16 hours
- Refunded 111,161 attempts
  - Total of: 347,154 ETH

- We define:
  - ***Successful transaction***: **resulted in token purchase**
  - ***Failed transaction***: **failed** to purchase any tokens (high gasPrice, etc)

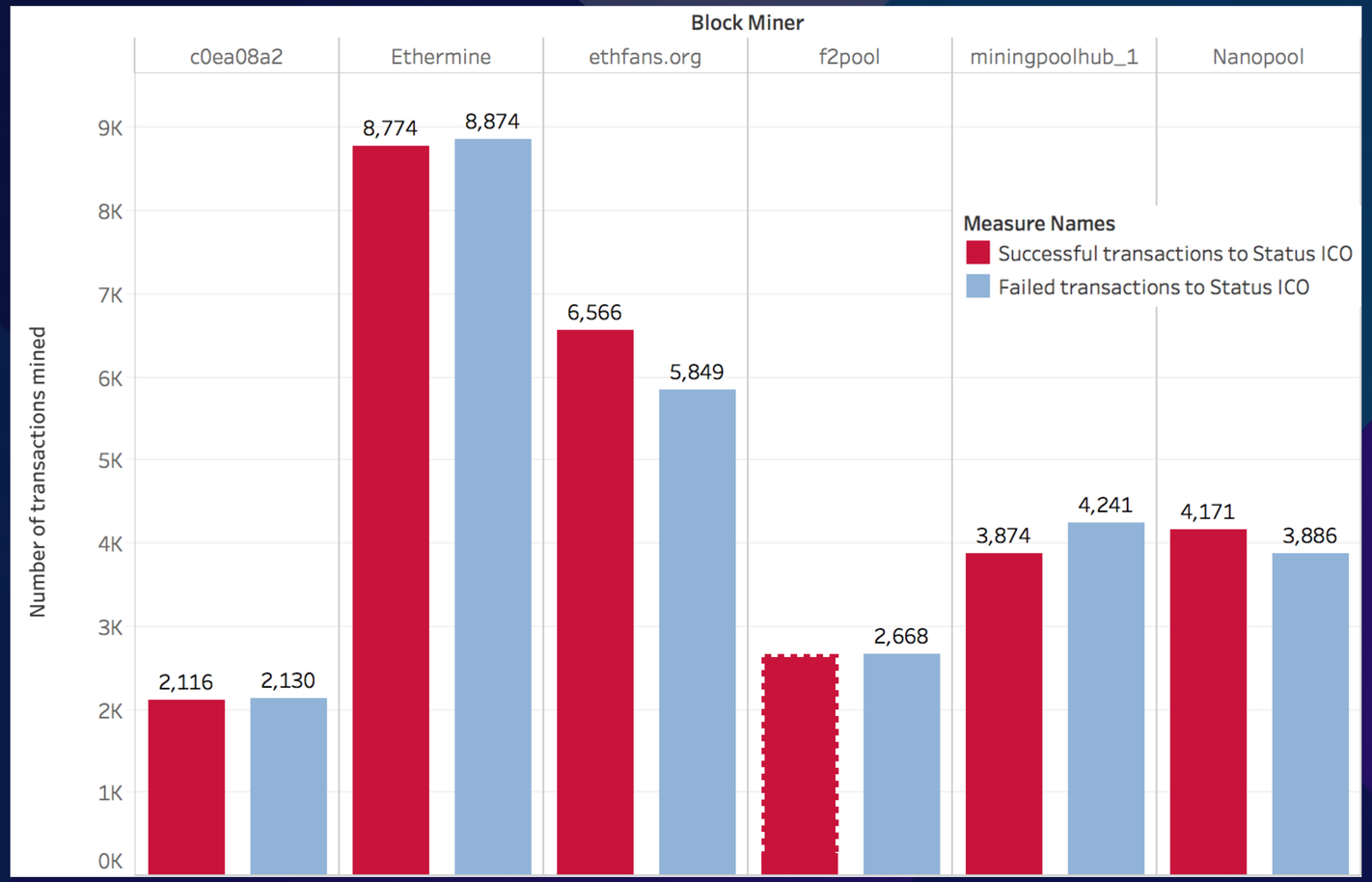
Story 1: Status ICO



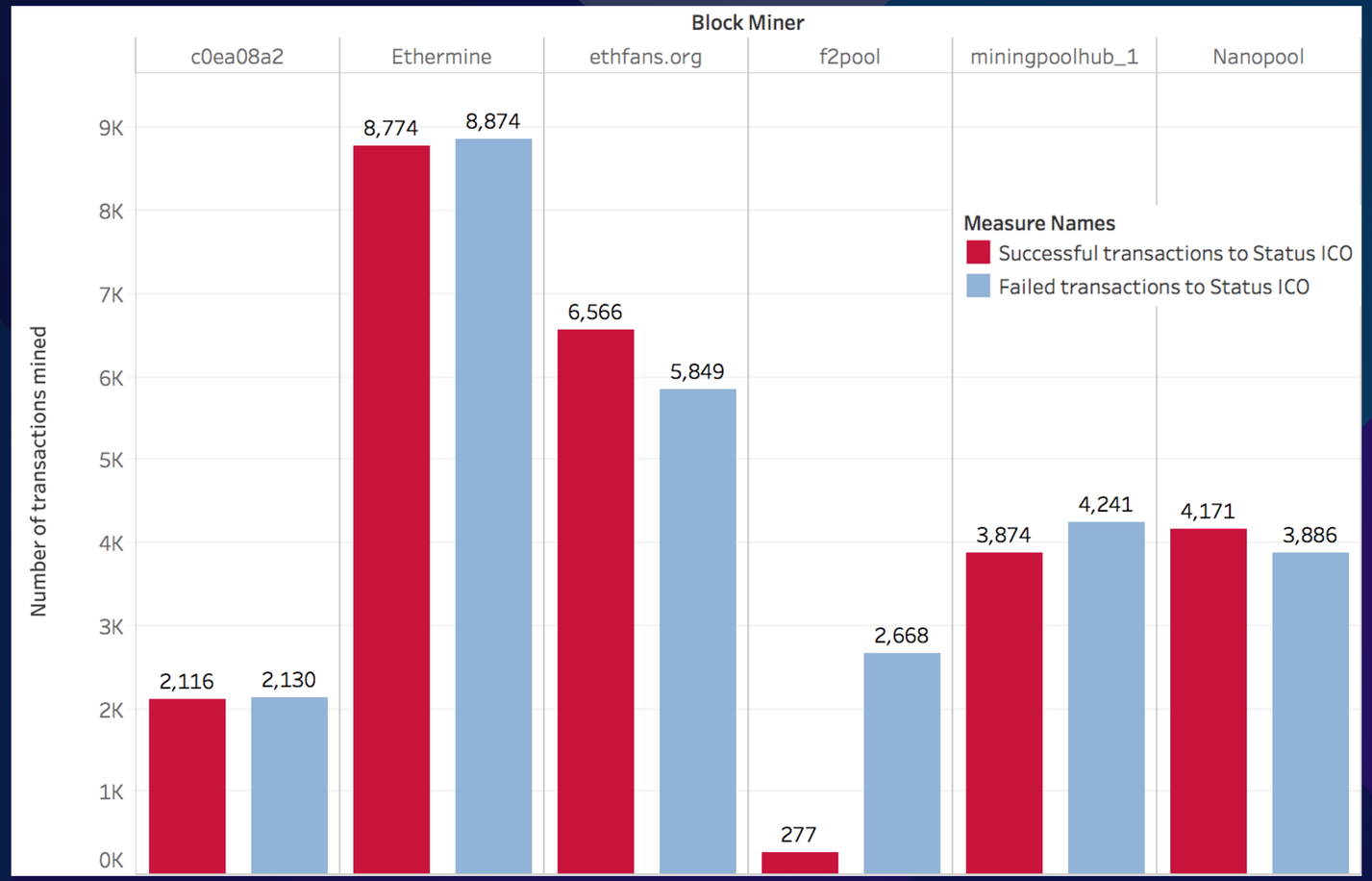
Miners in the network at the time



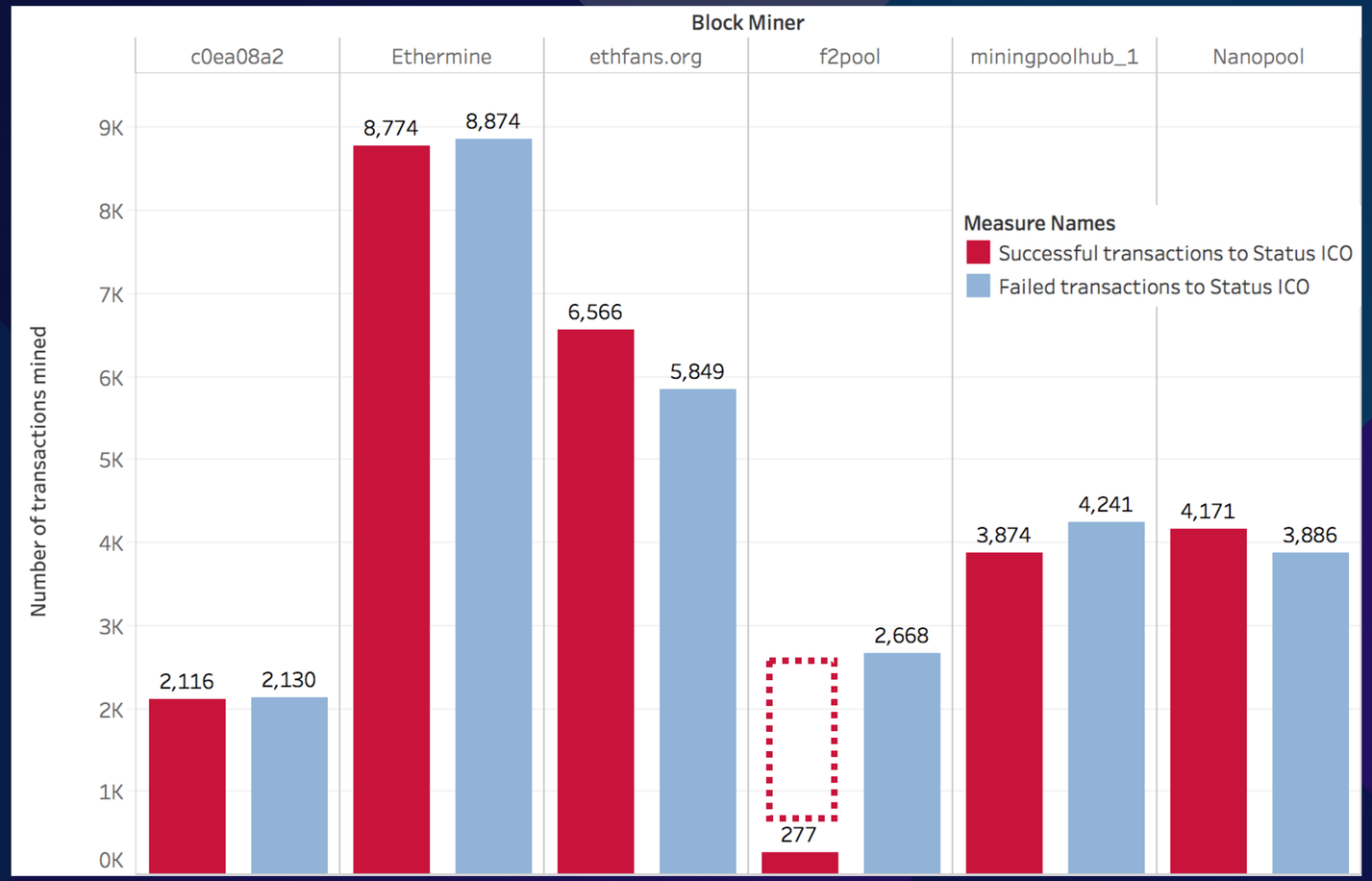
Story 1: Status ICO



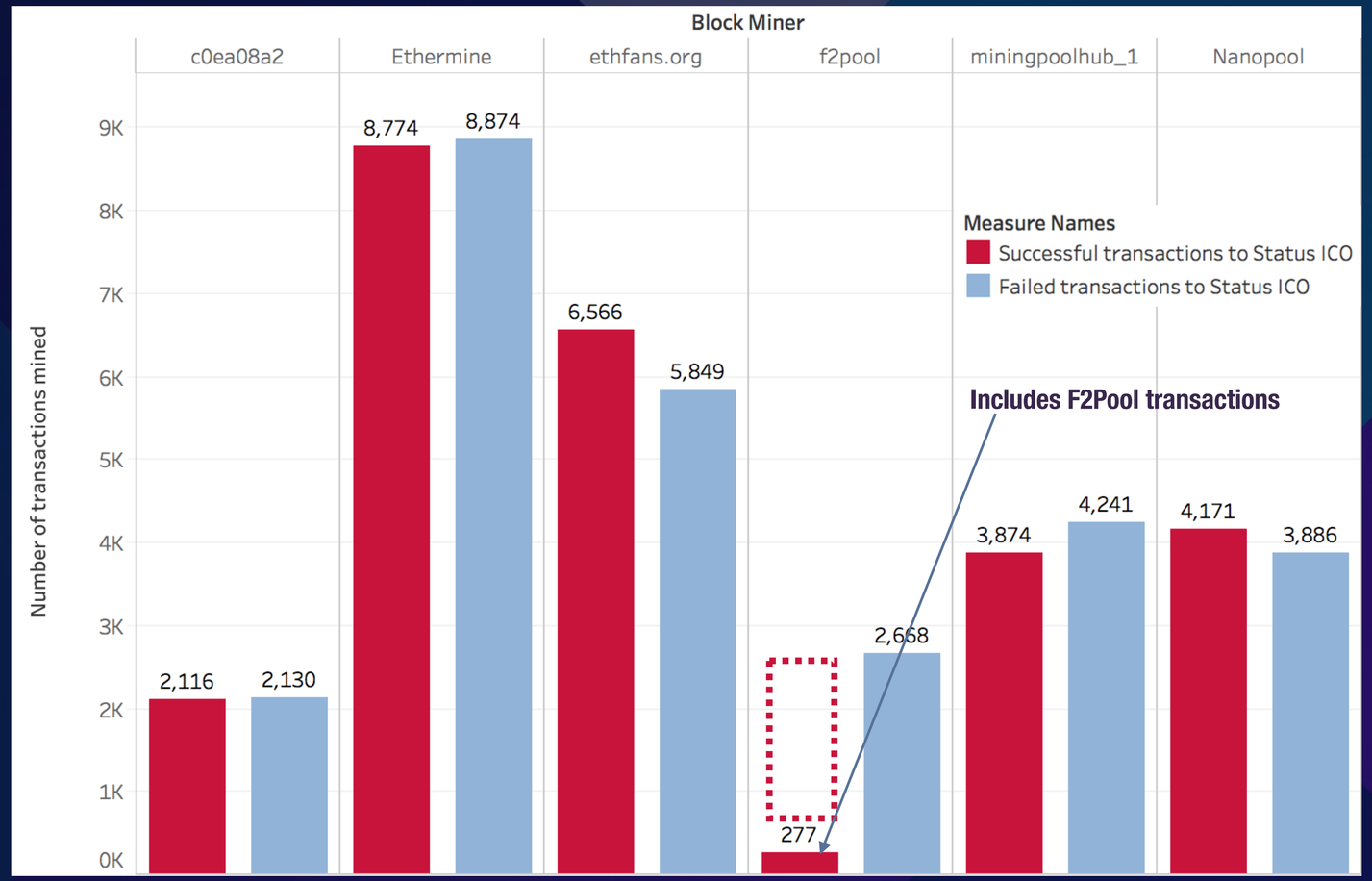
Story 1: Status ICO



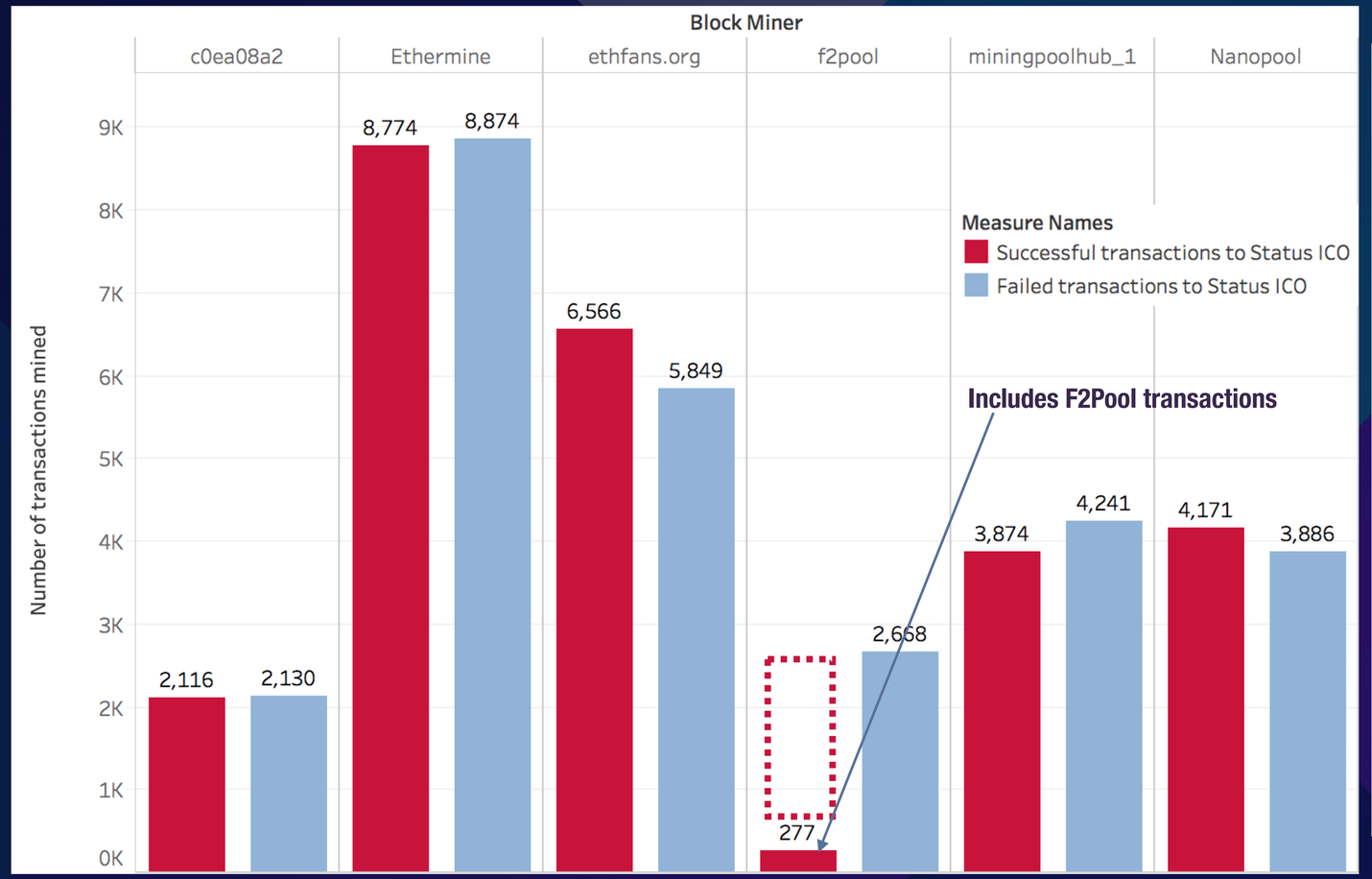
Story 1: Status ICO

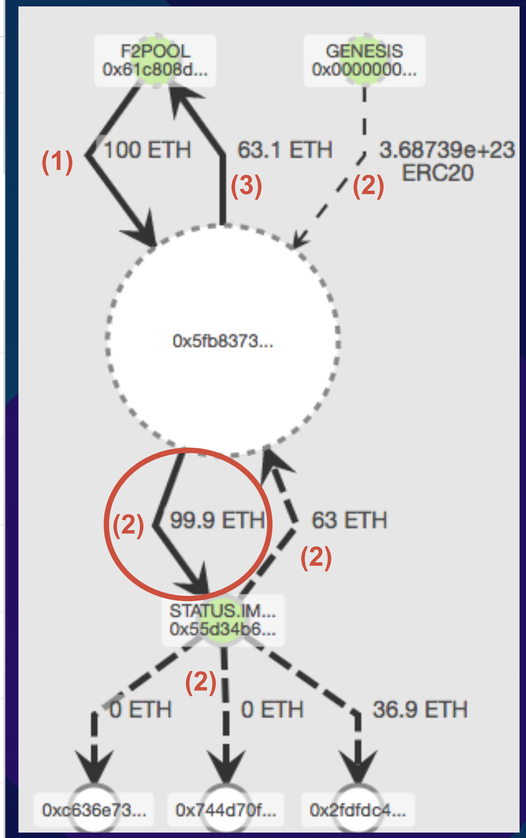


Story 1: Status ICO



Story 1: Status ICO



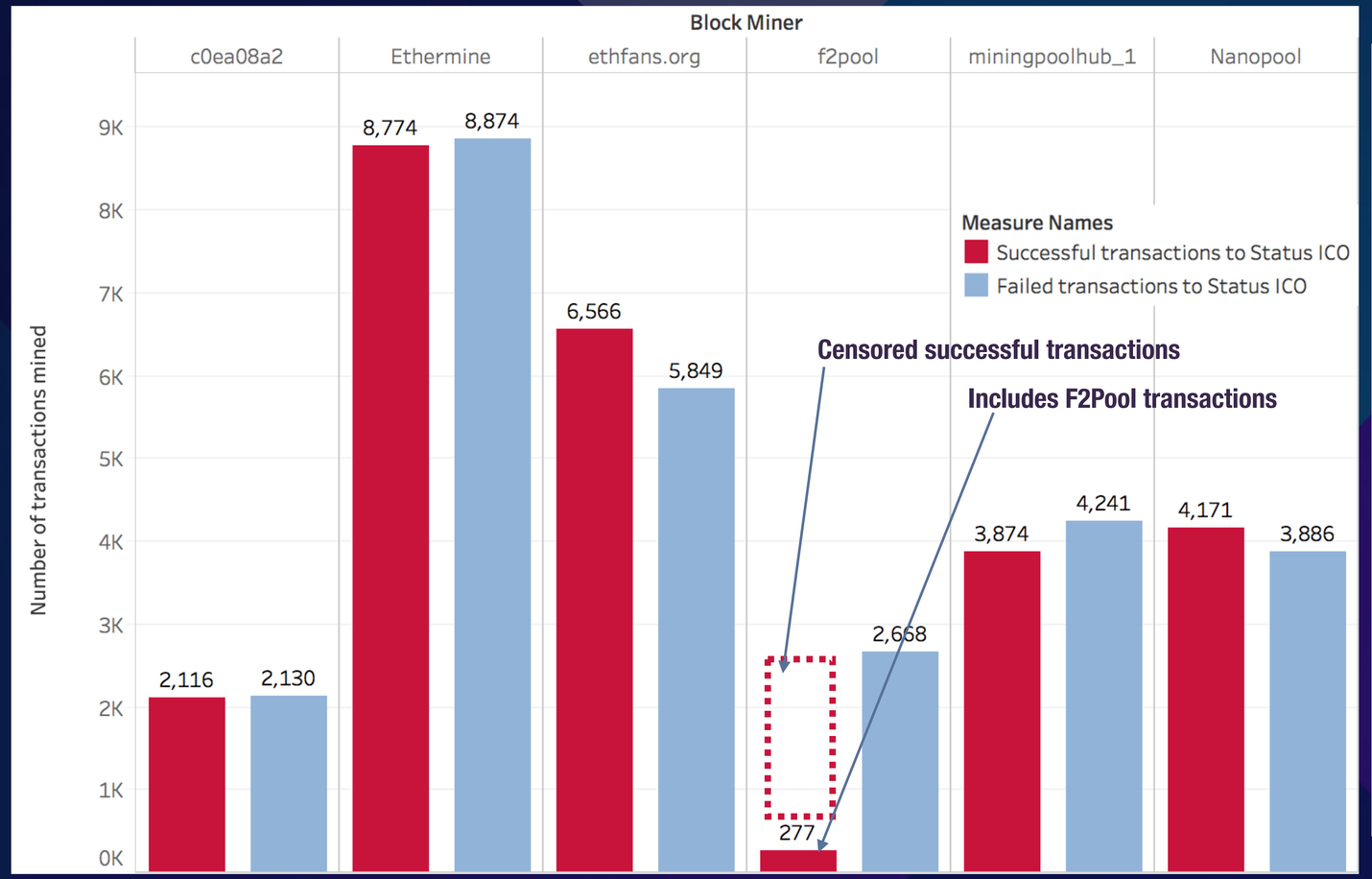


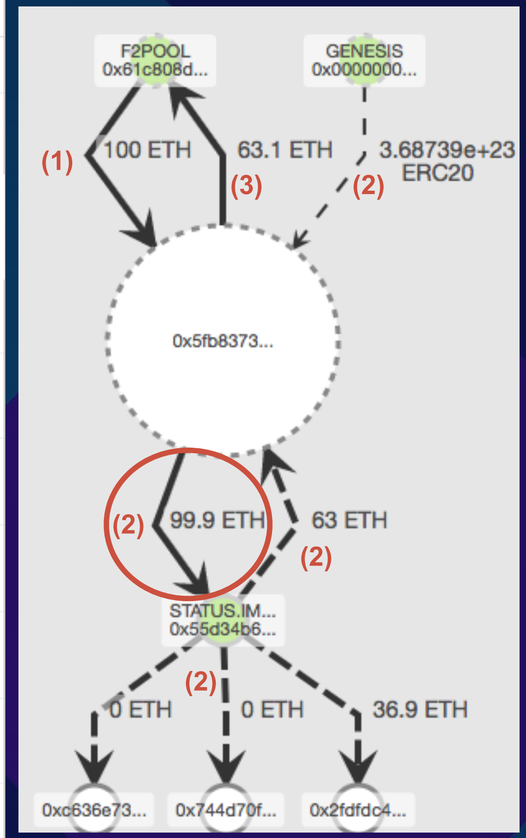
CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Transparent Dishonesty: Front-running Attacks on Blockchain

Story 1: Status ICO





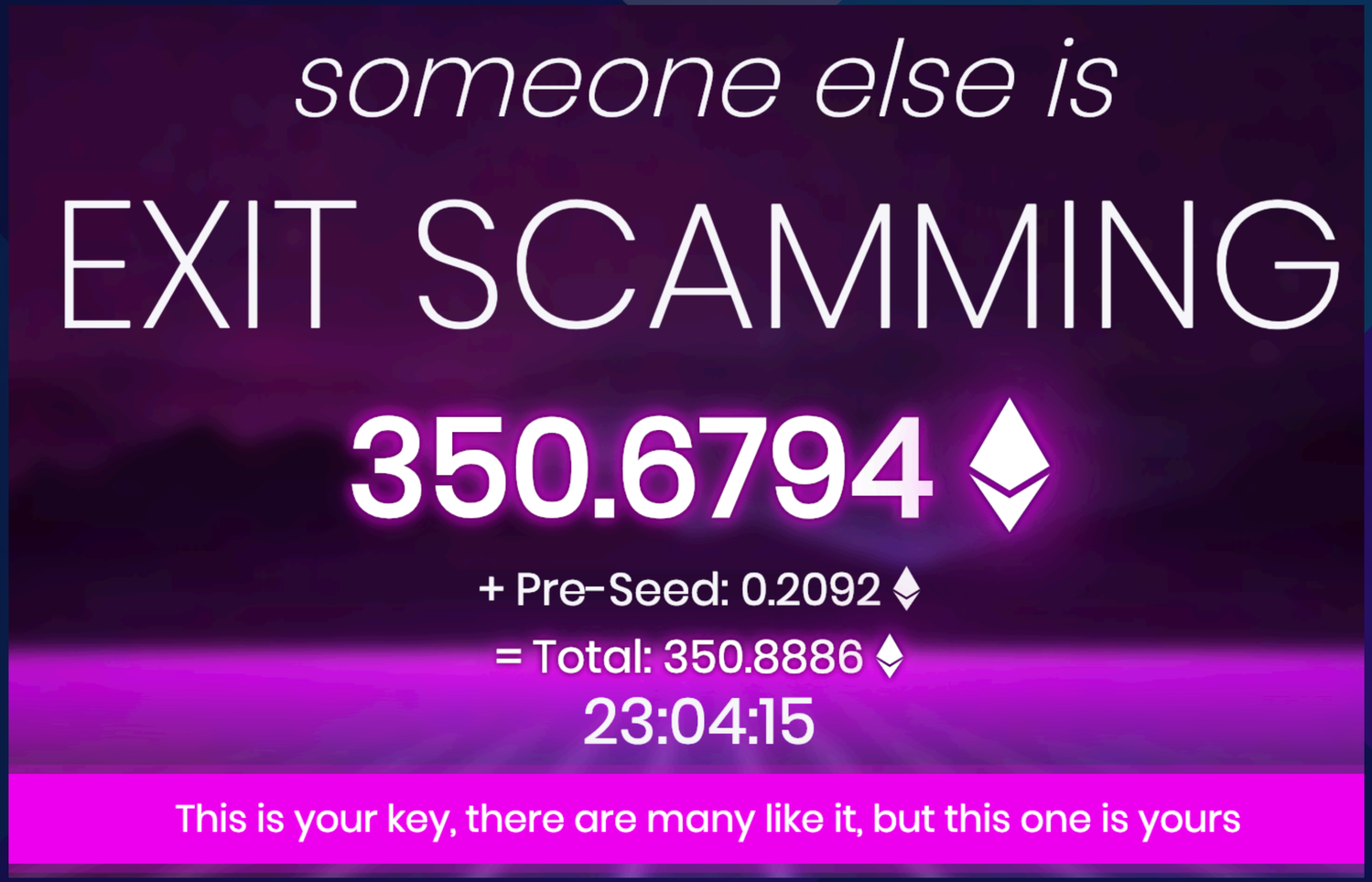
CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

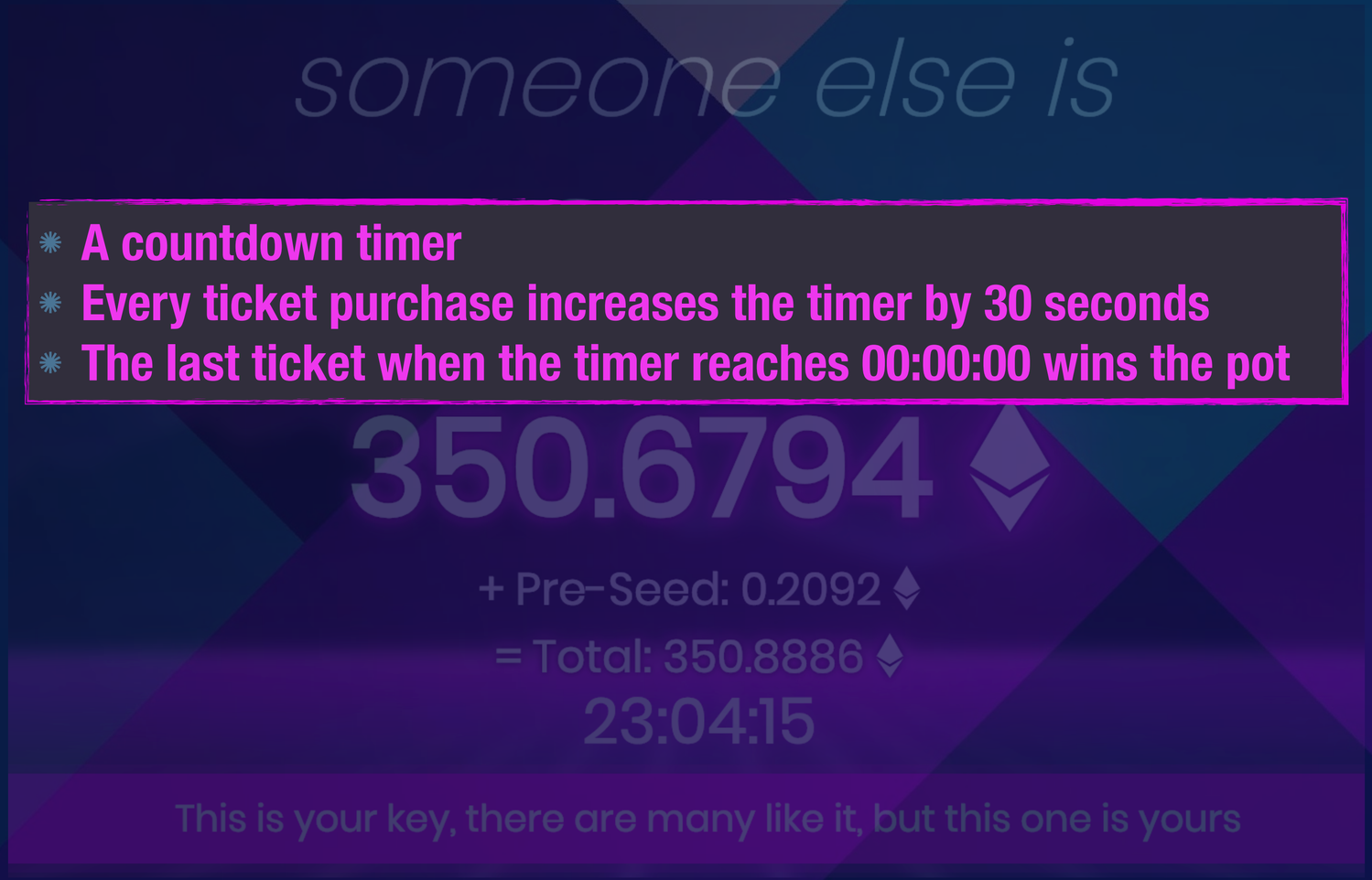
Transparent Dishonesty: Front-running Attacks on Blockchain

Story 2: FOMO3D

Story 2: FOMO3D

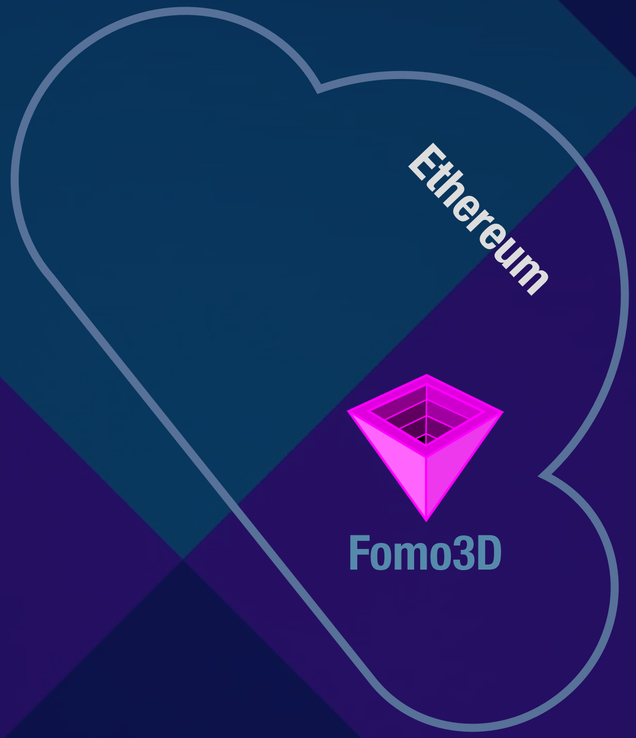


Story 2: FOMO3D



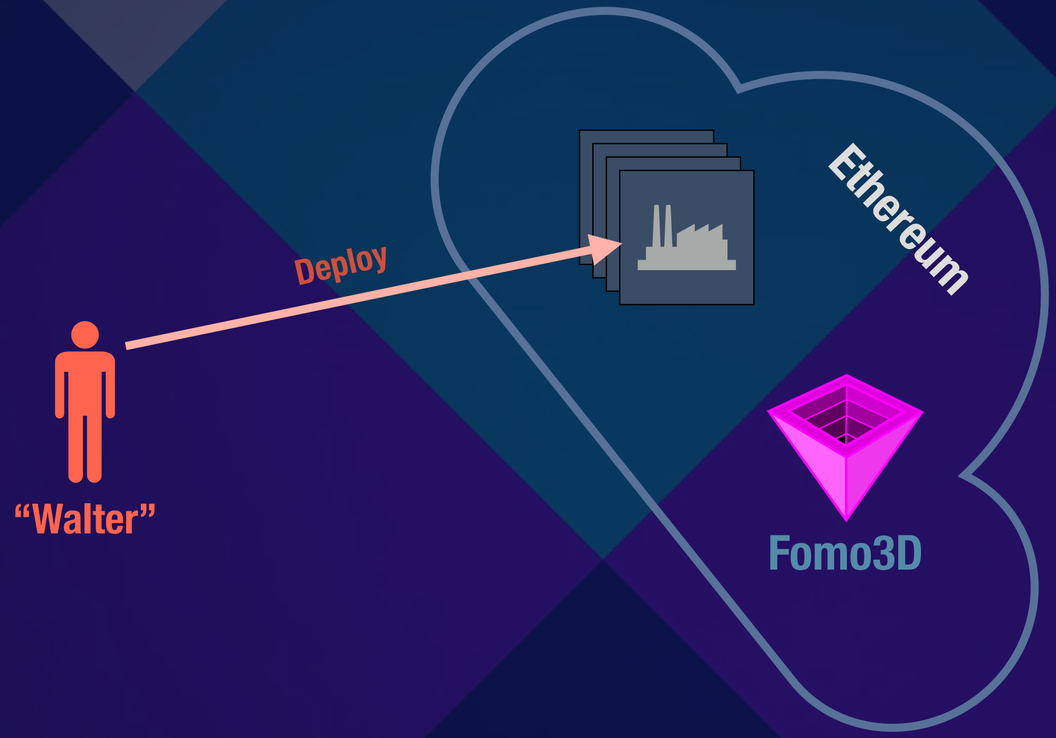
- A countdown timer
- Every ticket purchase increases the timer by 30 seconds
- The last ticket when the timer reaches 00:00:00 wins the pot

Story 2: FOMO3D

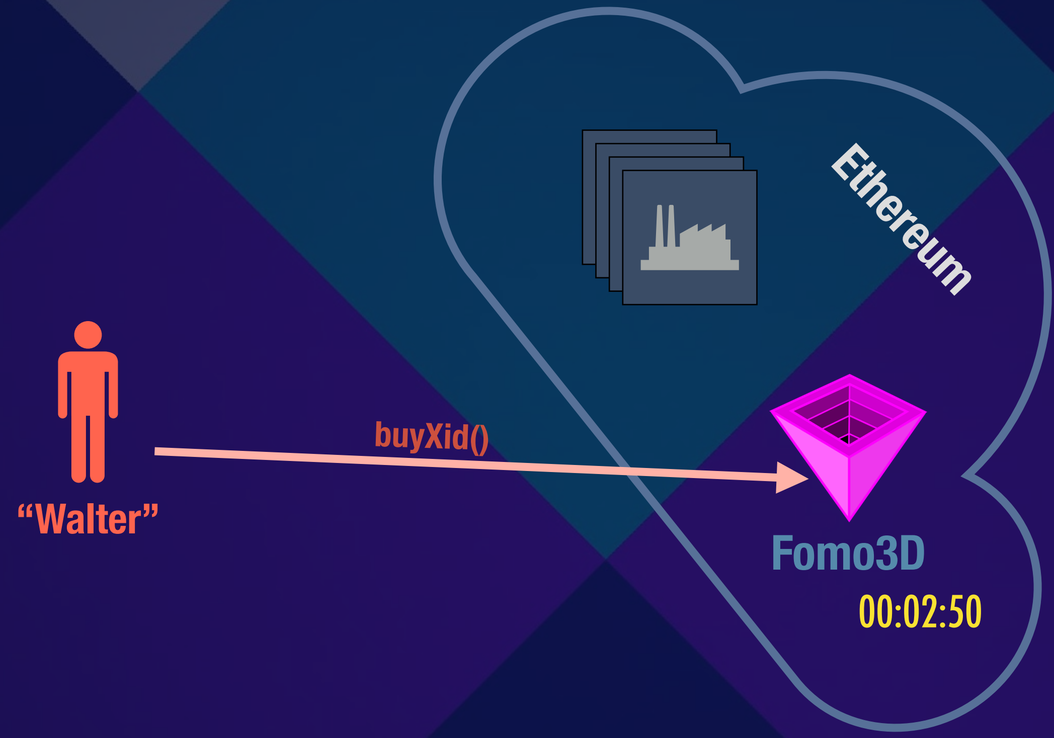


Story 2: FOMO3D

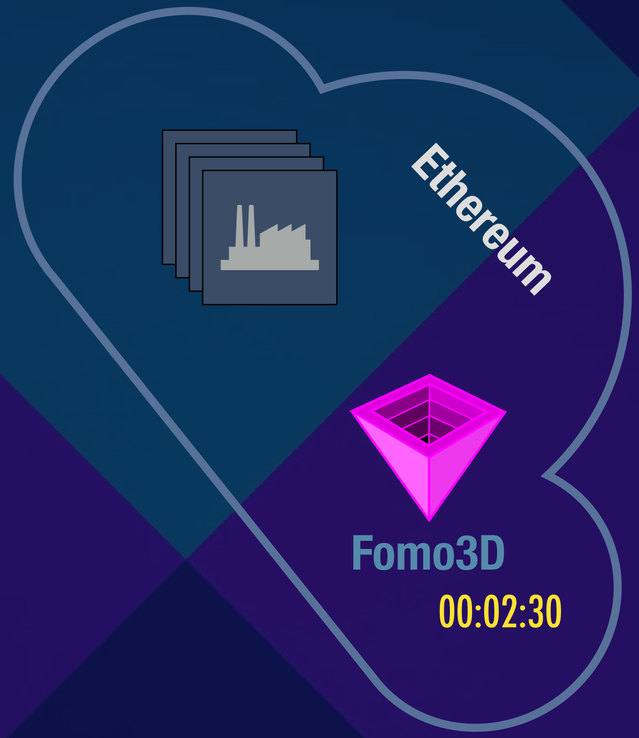
- “Walter” deploys contracts that have a “sophisticated” high gas consuming function



Story 2: FOMO3D

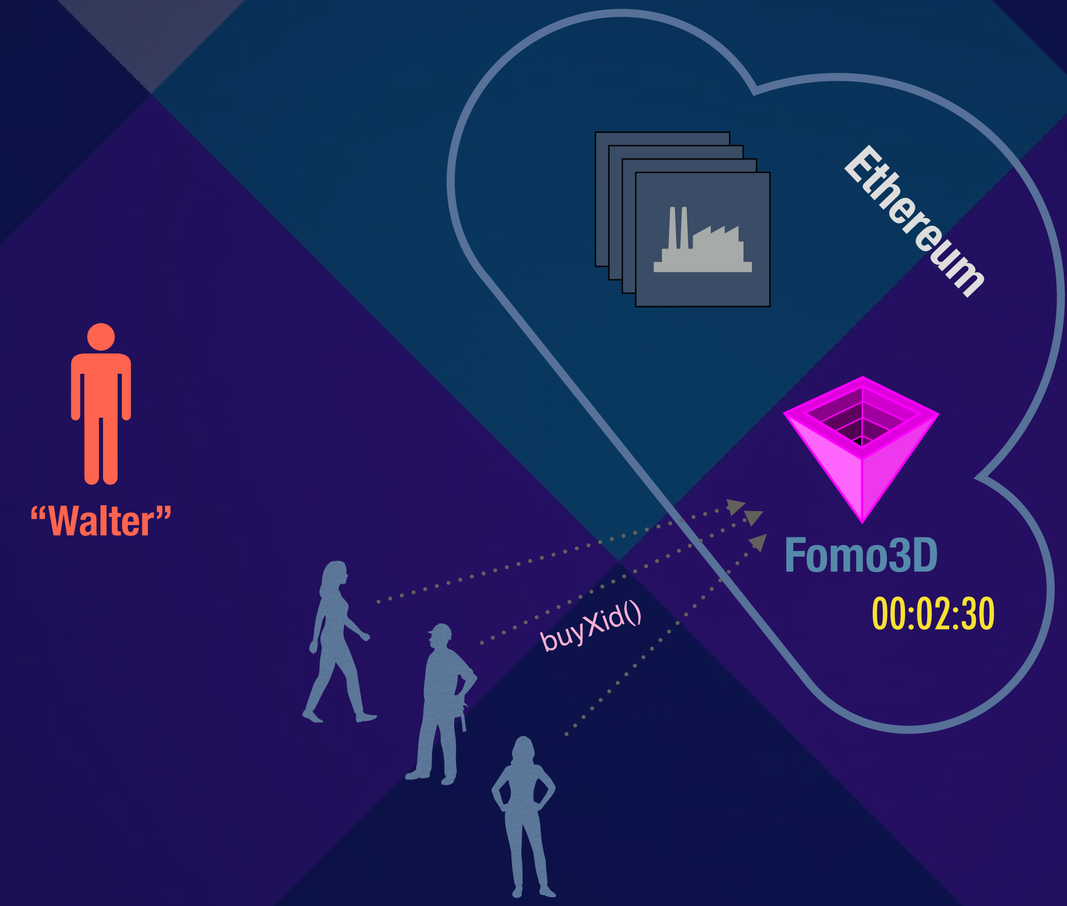


Story 2: FOMO3D

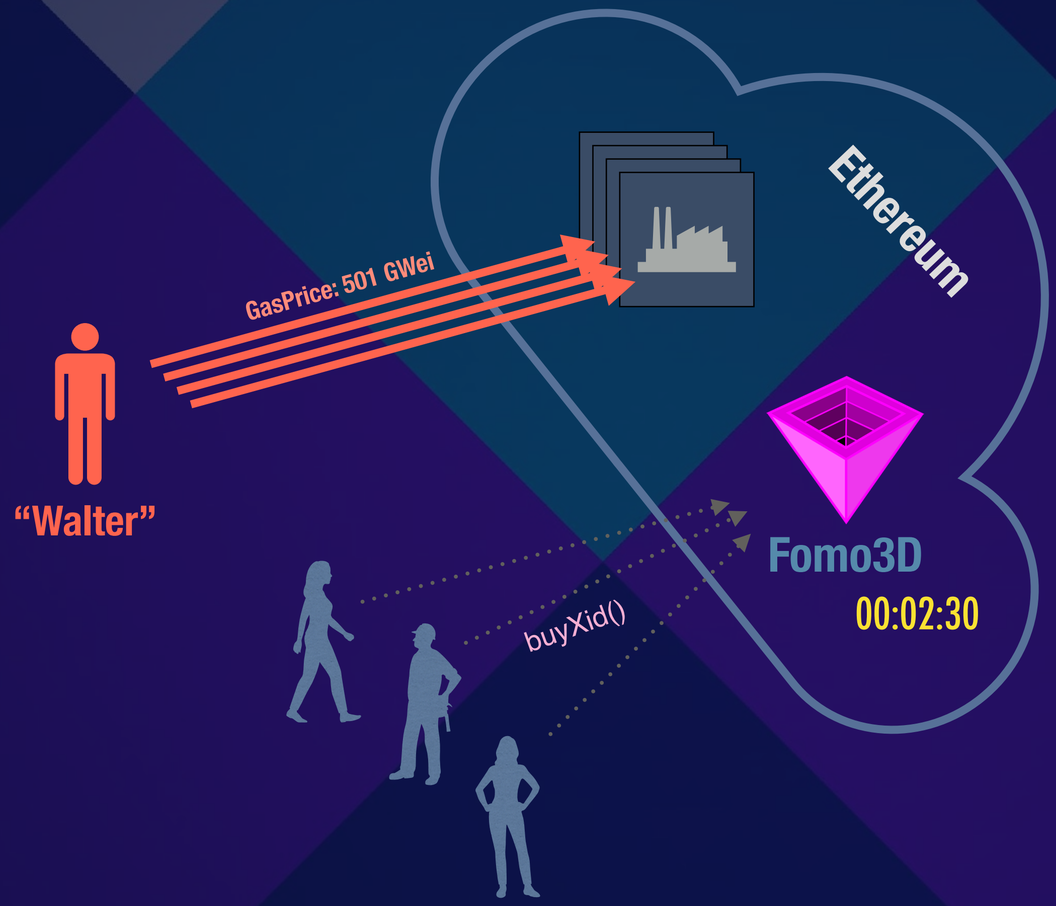


“Walter”

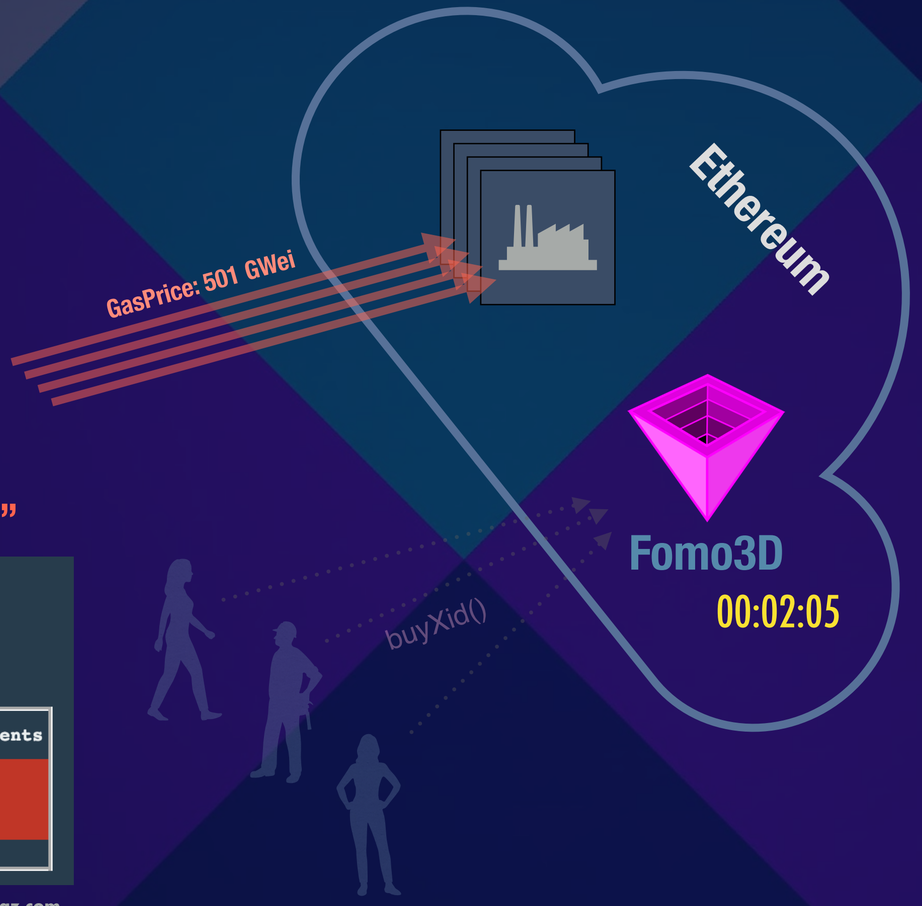
Story 2: FOMO3D



Story 2: FOMO3D



Story 2: FOMO3D



Block 6191904

2018-08-22 06:49:57, ts:1534920597

Average gas price: 190.0 Gwei

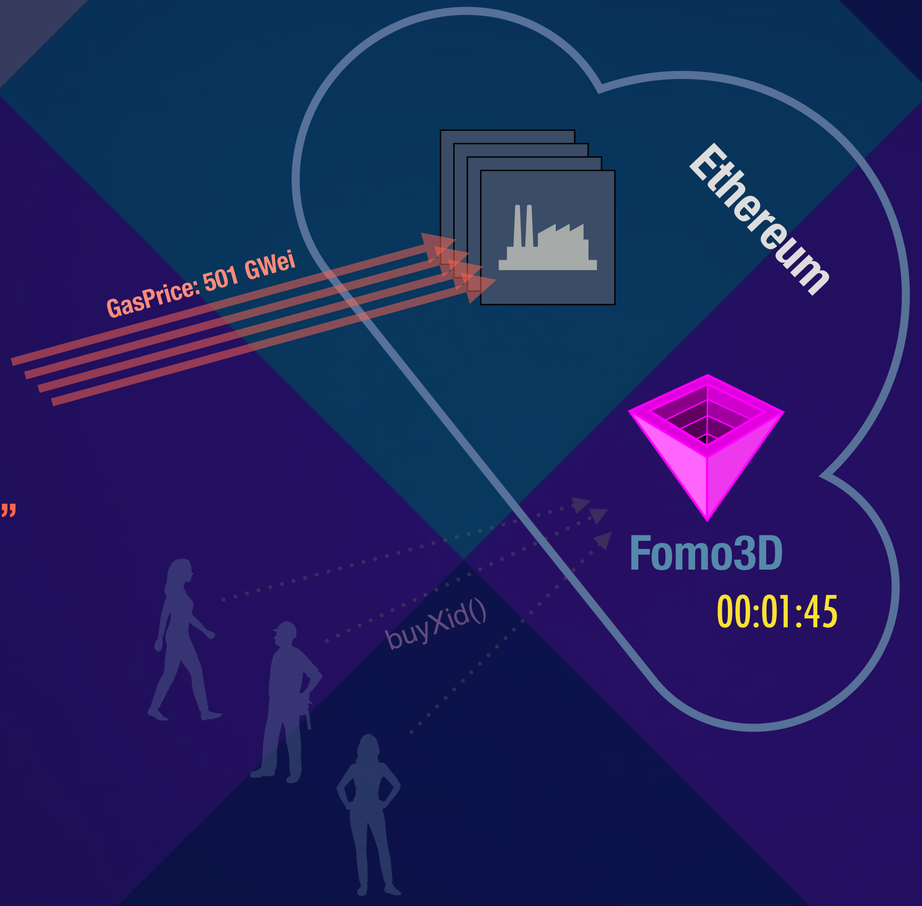
Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0xF03…1f2	0x18e…801	0x7d1…4cf	0	190.0	4,200,000	4,200,000	0.798018		
1	0x87C…4eF	0x18e…801	0x8db…9d2	0	190.0	3,600,000	3,600,000	0.684013		
2	0xf6E…059	0x18e…801	0x79a…1aa	0	190.0	200,000	200,000	0.038		
				0	570.008	8,000,000	8,000,000	1.52003		

osolmaz.com

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Story 2: FOMO3D



Block 6191905

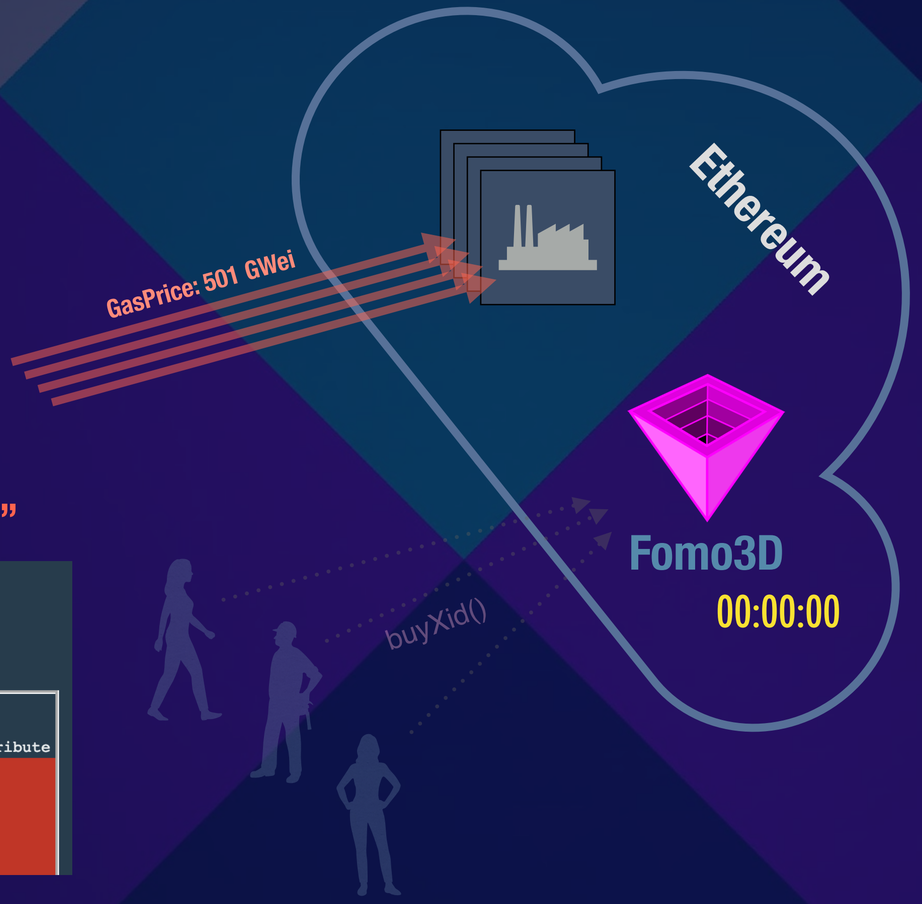
2018-08-22 06:50:36, ts:1534920636

Average gas price: 228.3 Gwei

Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0xb29…347	0x359…B23	0xd54…a47	0.0396017	5.0	100,000	21,000	0.000105		
1	0xb29…347	0xF65…Fa8	0xe76…d52	0.0195326	5.0	100,000	21,000	0.000105		
2	0xb29…347	0x48d…aFe	0x78b…f5b	0.0095638	5.0	100,000	21,000	0.000105		
3	0x9DA…0cF	0x18e…801	0xbc8…319	0	501.0	4,800,000	4,800,000	2.40482		
4	0x7Dd…c4c	0x18e…801	0xd9d…2b7	0	501.0	2,700,000	2,700,000	1.35271		
5	0x00c…776	0x18e…801	0x7c3…2e0	0	501.0	400,000	400,000	0.2004		
6	0xA10…e25	0xb9e…9b0	0xb27…c2e	6.99832	80.0	21,000	21,000	0.00168		
				7.06702	**1598.01**	**8,221,000**	**7,984,000**	**3.95993**		

A few more blocks…

Story 2: FOMO3D



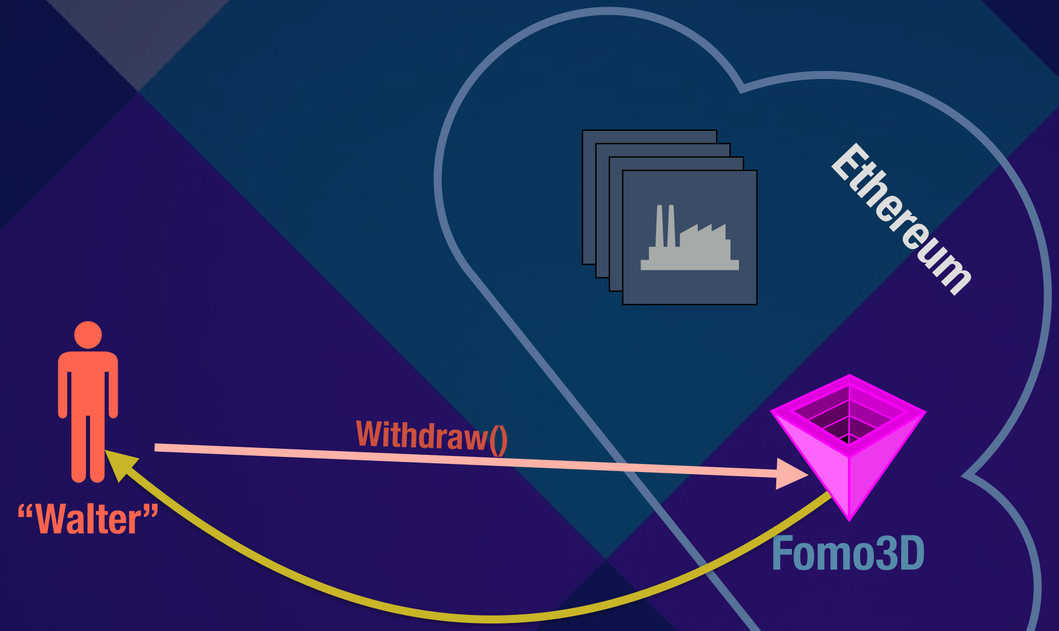
Block 6191909

2018-08-22 06:51:17, ts:1534920677

Average gas price: 93.0 Gwei

Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0x32A…370	0xA62…Da1	0xa14…012	0.00560162	5562.2	379,000	304,750	1.69508	buyXaddr	onBuyAndDistribute
1	0xC96…590	0x18e…801	0xf47…9ca	0	501.0	2,200,000	37,633	0.0188542		
2	0xb1D…aEF	0x18e…801	0xe4c…edb	0	501.0	1,400,000	37,633	0.0188542		
3	0x18D…A9A	0x18e…801	0xf3a…995	0	501.0	800,000	37,633	0.0188542		
4	0x00c…776	0x18e…801	0xeb2…100	0	501.0	400,000	37,633	0.0188541		
5	0xf6E…059	0x18e…801	0x8c2…b23	0	501.0	200,000	37,633	0.0188541		

Story 2: FOMO3D



Contract 0xa62142888aba8370742be823c1782d17a0389da1 (Fomo3D:Long)

TRANSFER 10,469.660003123933104565 Ether From 0xa62142888aba8370742… To 0xa169df5ed3363cfc4c92…

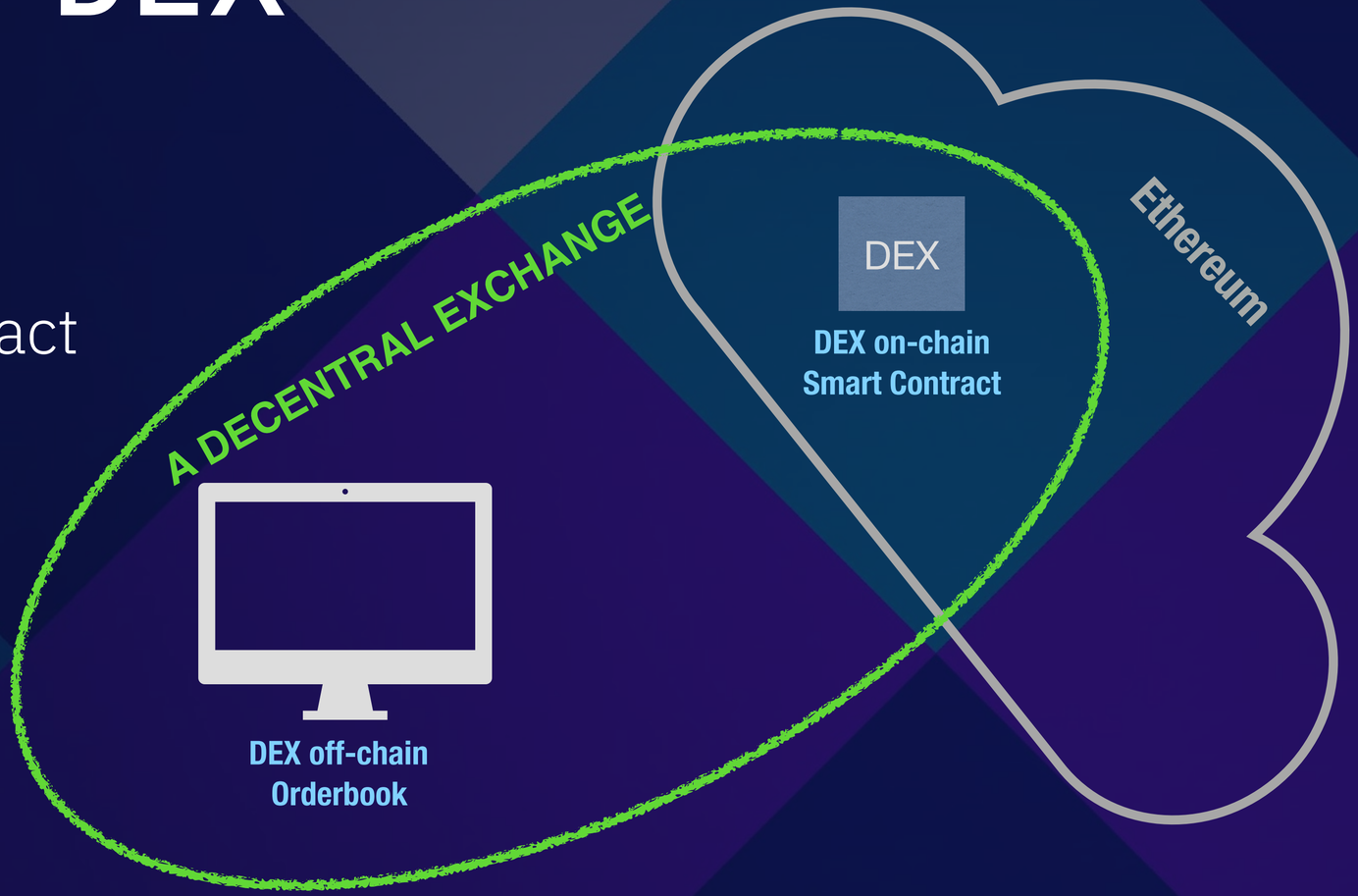
Story 3: DEX

Story 3: DEX

- Decentralize Exchange
  - On-chain Smart Contract
  - Off-chain order book

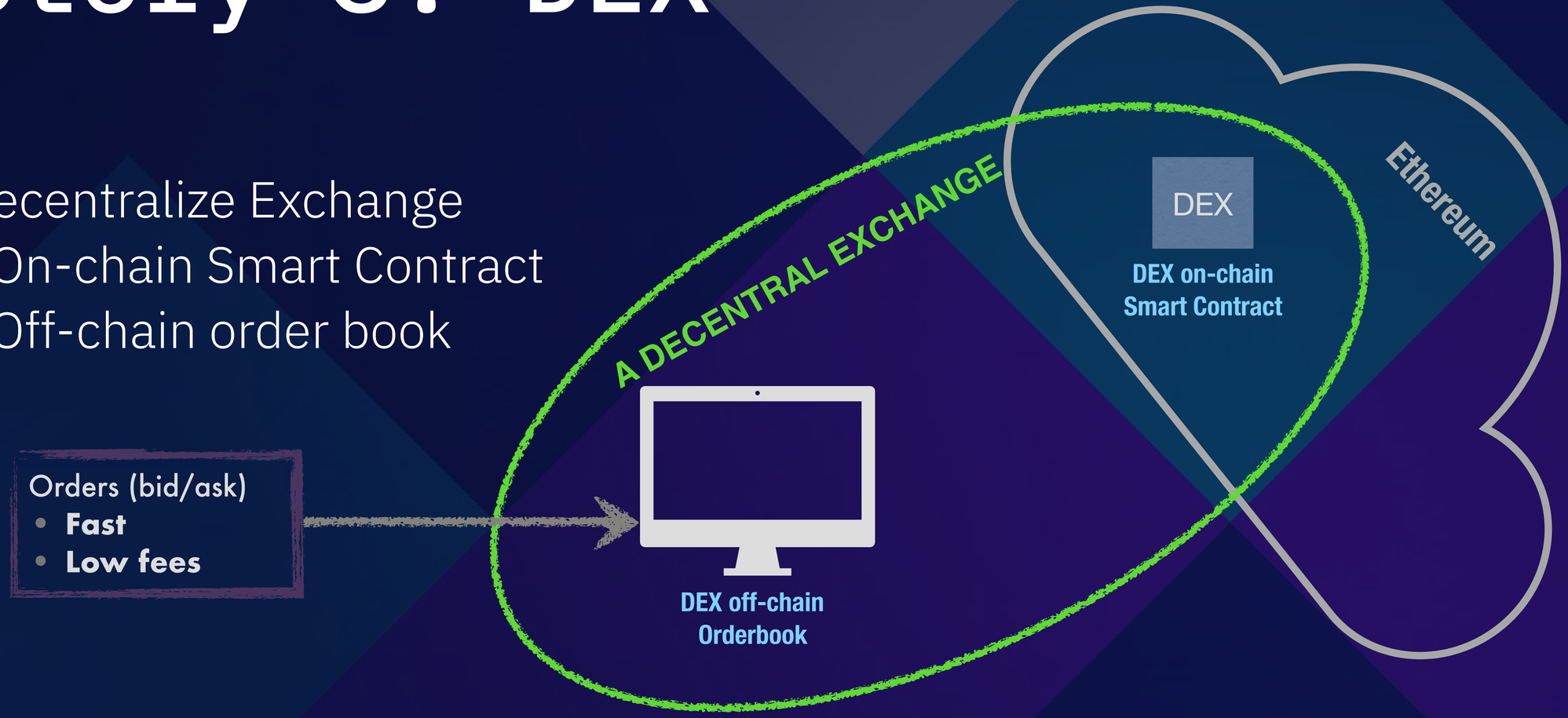
Story 3: DEX

- Decentralize Exchange
  - On-chain Smart Contract
  - Off-chain order book



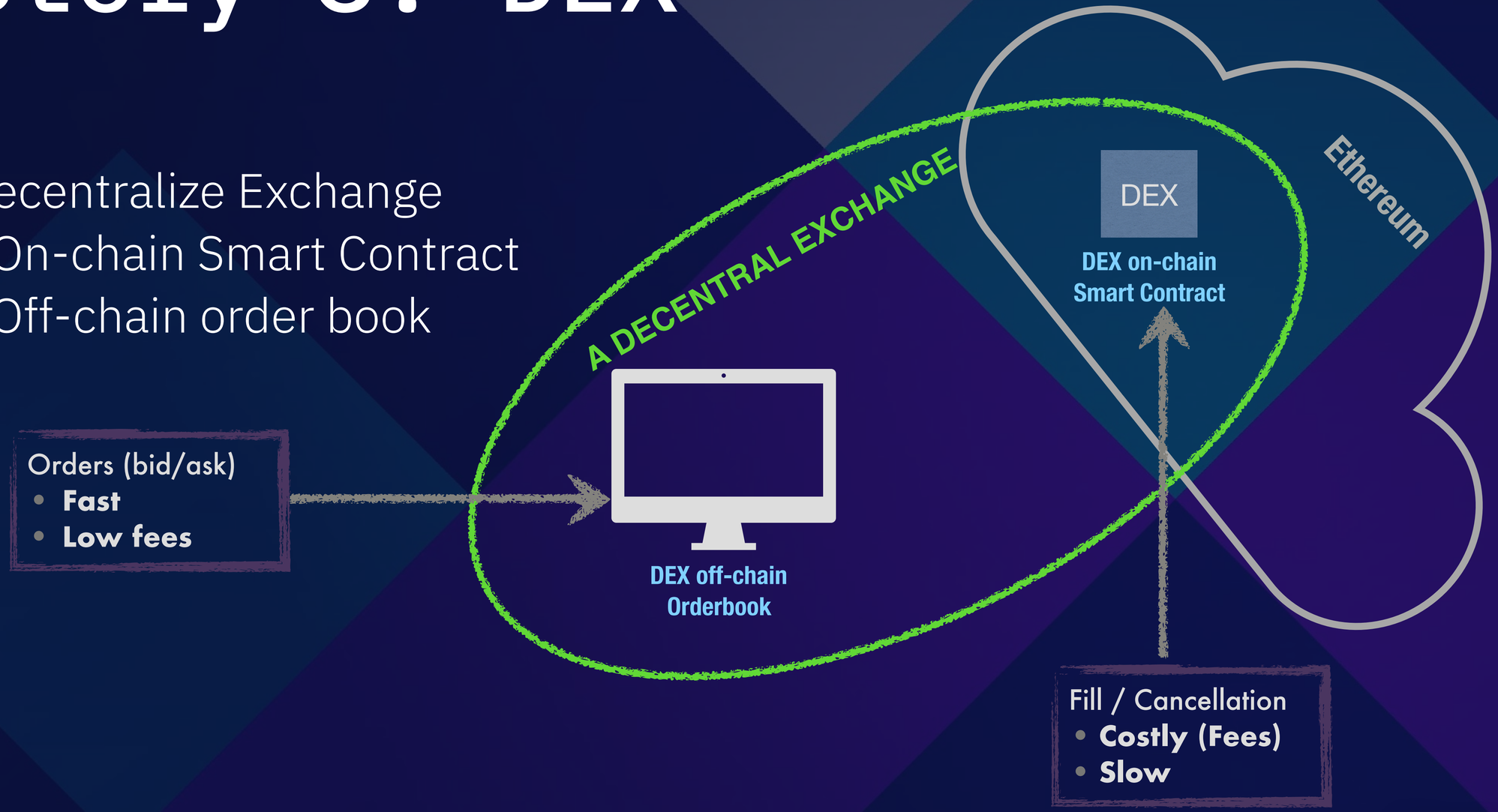
Story 3: DEX

- Decentralize Exchange
  - On-chain Smart Contract
  - Off-chain order book



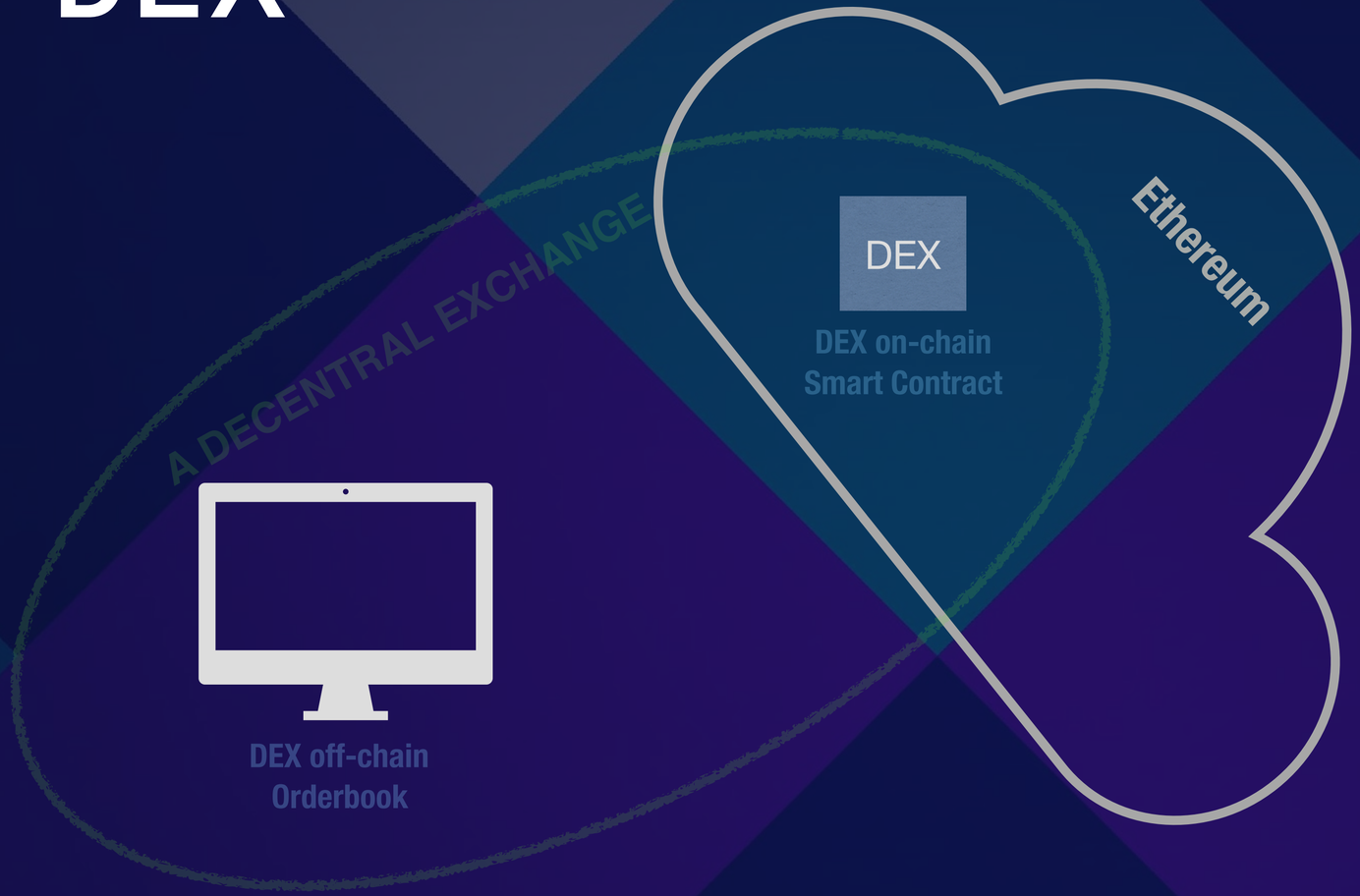
Story 3: DEX

- Decentralize Exchange
  - On-chain Smart Contract
  - Off-chain order book

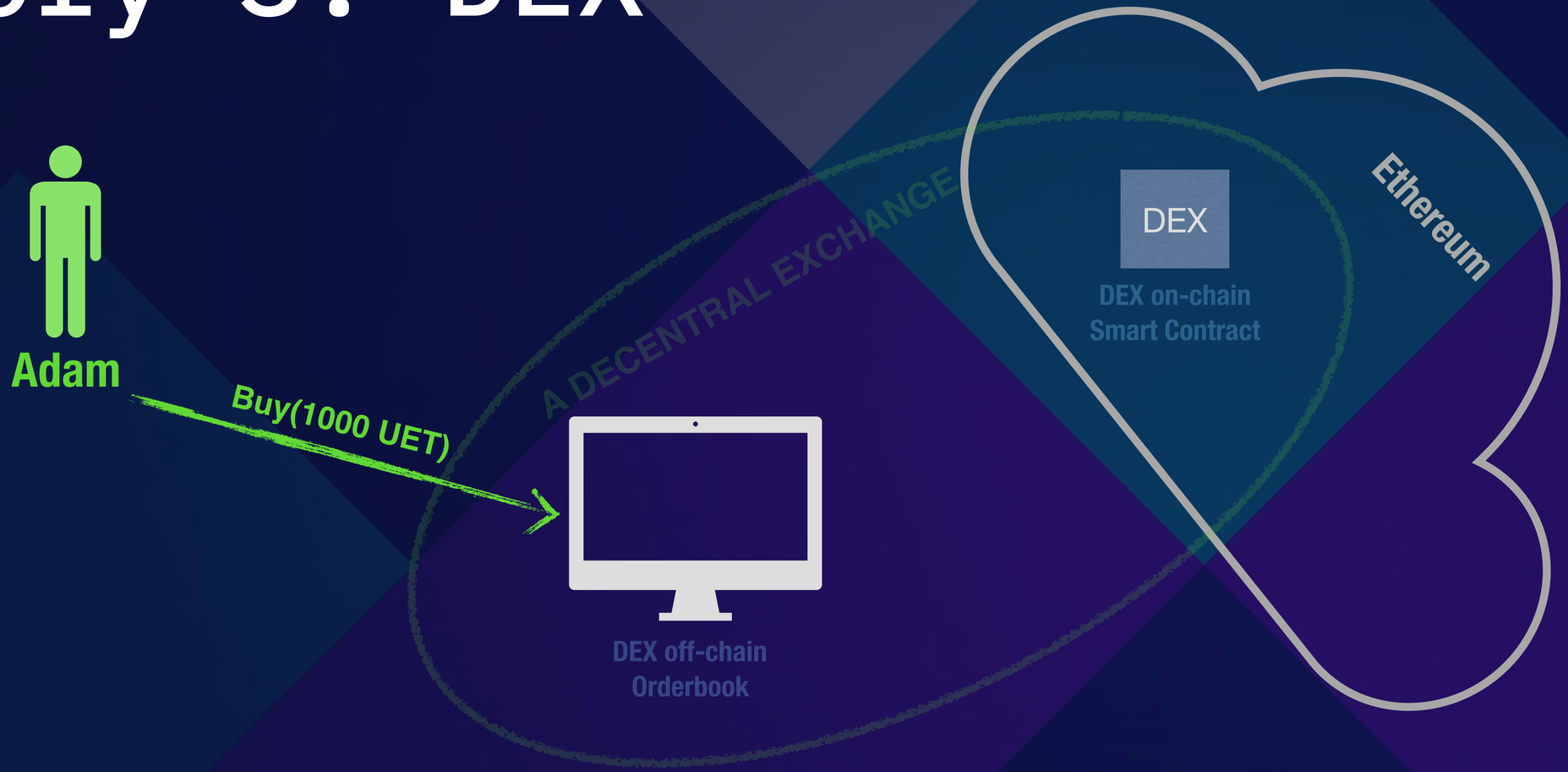


Story 3: DEX

Adam



Story 3: DEX


Adam
Buy(1000 UET)
A DECENTRAL EXCHANGE
DEX off-chain
Orderbook
DEX
DEX on-chain
Smart Contract
Ethereum

Story 3: DEX


Adam
Buy(1000 UET)
Buy(1000 UET)
DEX off-chain
Orderbook
DEX
DEX on-chain
Smart Contract
Ethereum

Story 3: DEX

Adam

DEX off-chain
Orderbook

CONSENSYS Diligence

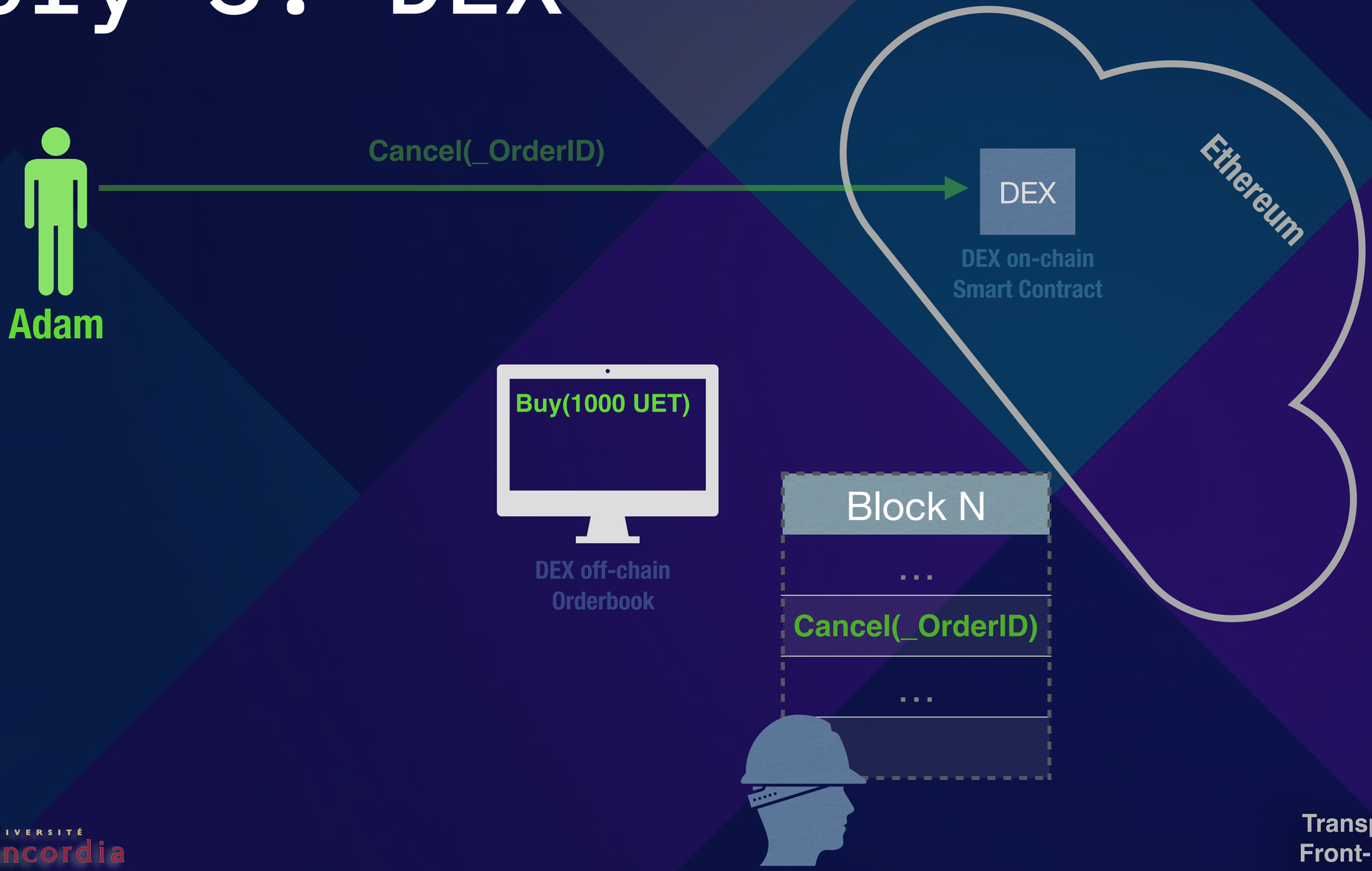
UNIVERSITÉ Concordia UNIVERSITY

Story 3: DEX

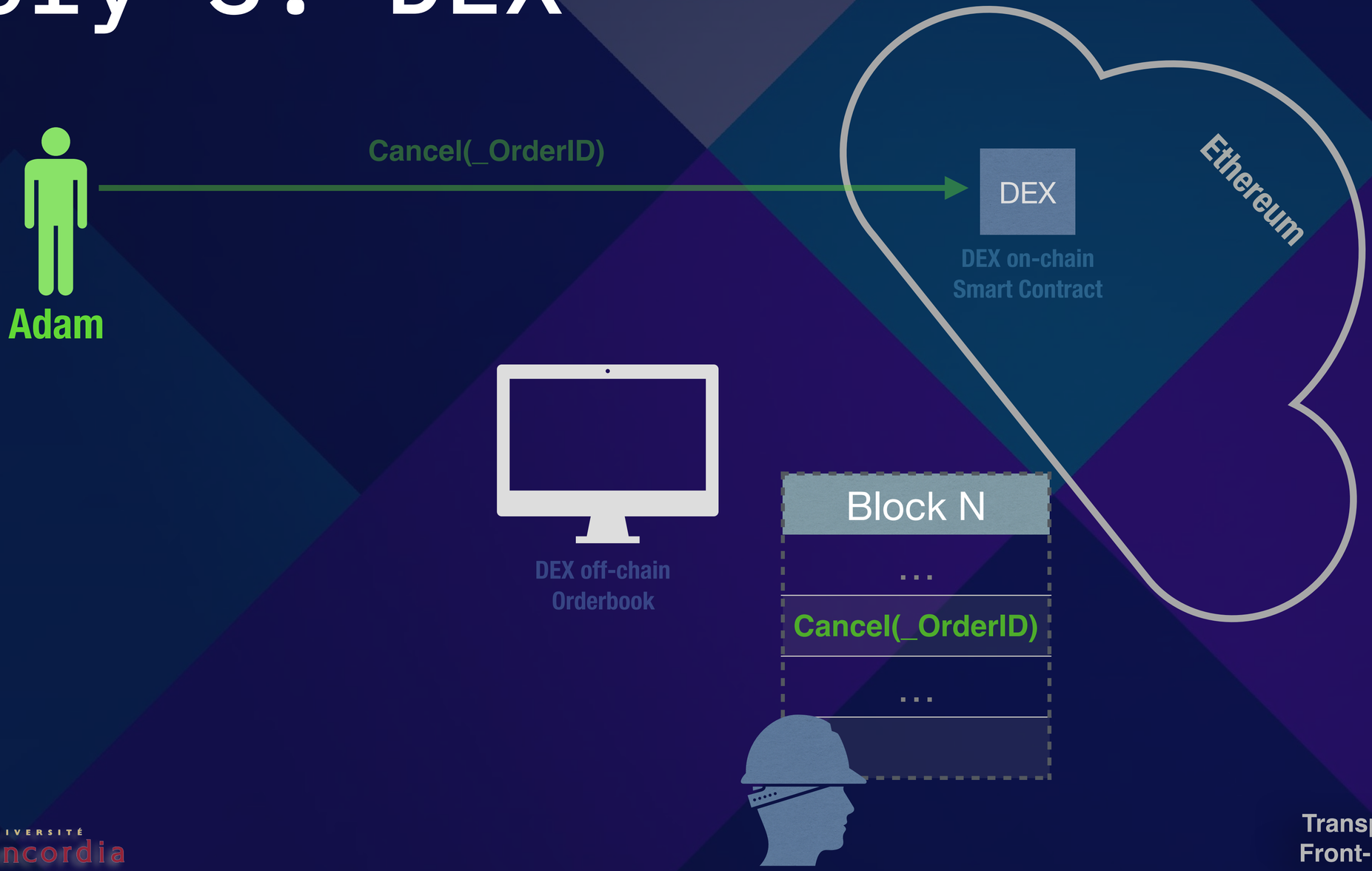

Cancel(_OrderID)
Adam
DEX
DEX on-chain
Smart Contract
Ethereum
Buy(1000 UET)
DEX off-chain
Orderbook


Useless Ethereum Token Charts
Zoom 1d 7d 1m 3m 1y YTD ALL
$1 000K
$750K
$500K
$250K
$0
Market Cap
200K
0
24h Vol
27. Jan
28. Jan
29. Jan
30. Jan
31. Jan
1. Feb
2. Feb

Story 3: DEX


Cancel(_OrderID)
Adam
DEX
DEX on-chain
Smart Contract
Ethereum
Buy(1000 UET)
DEX off-chain
Orderbook
Block N
...
Cancel(_OrderID)
...

Story 3: DEX


Cancel(_OrderID)
Adam
DEX
DEX on-chain
Smart Contract
Ethereum
DEX off-chain
Orderbook
Block N
...
Cancel(_OrderID)
...

Story 3: DEX


Cancel(_OrderID)
Adam
DEX
DEX on-chain
Smart Contract
Ethereum
Buy(1000 UET)
DEX off-chain
Orderbook

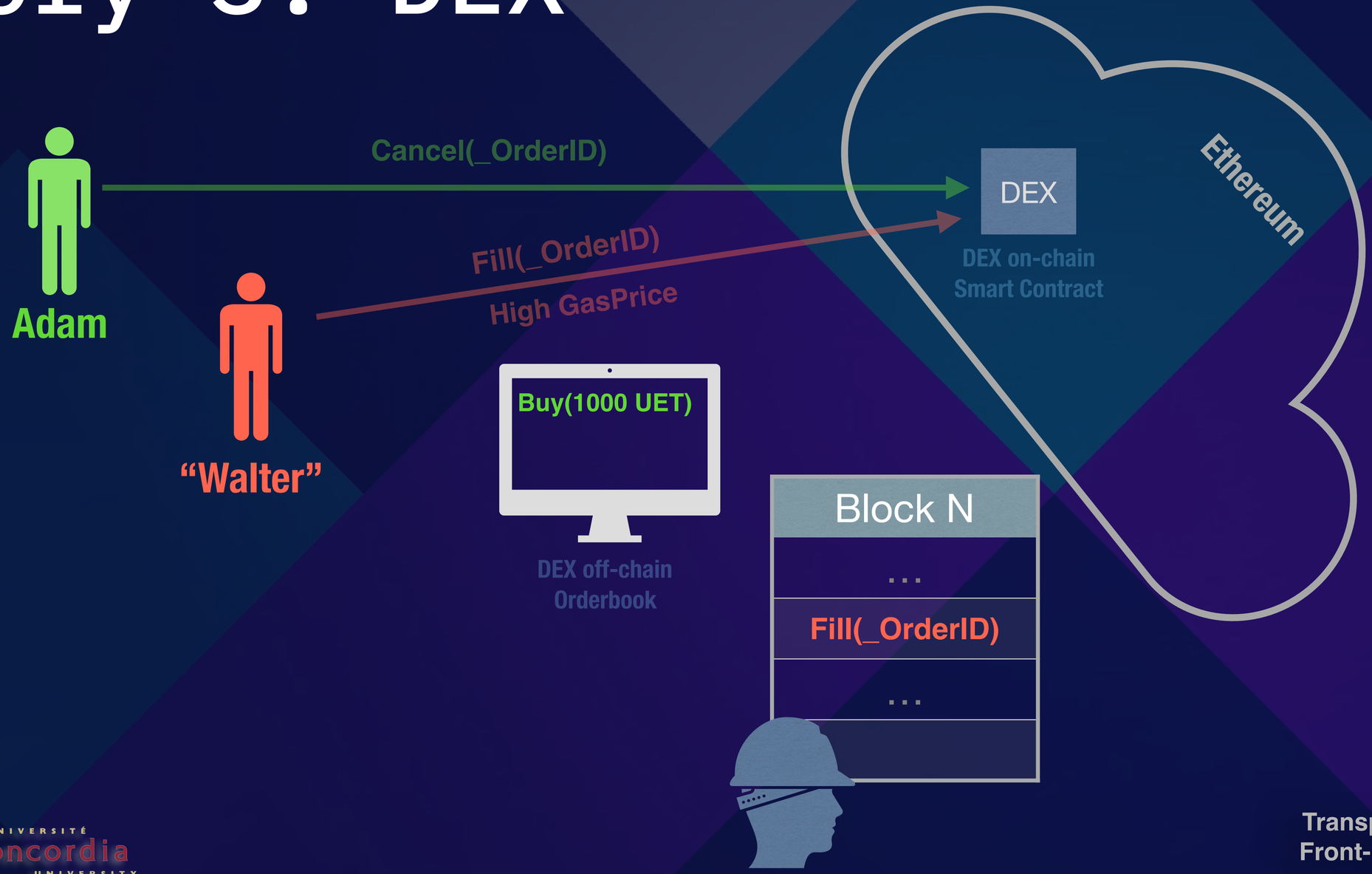
Story 3: DEX


Cancel(_OrderID)
Adam
"Walter"
Buy(1000 UET)
DEX off-chain
Orderbook
DEX
DEX on-chain
Smart Contract
Ethereum

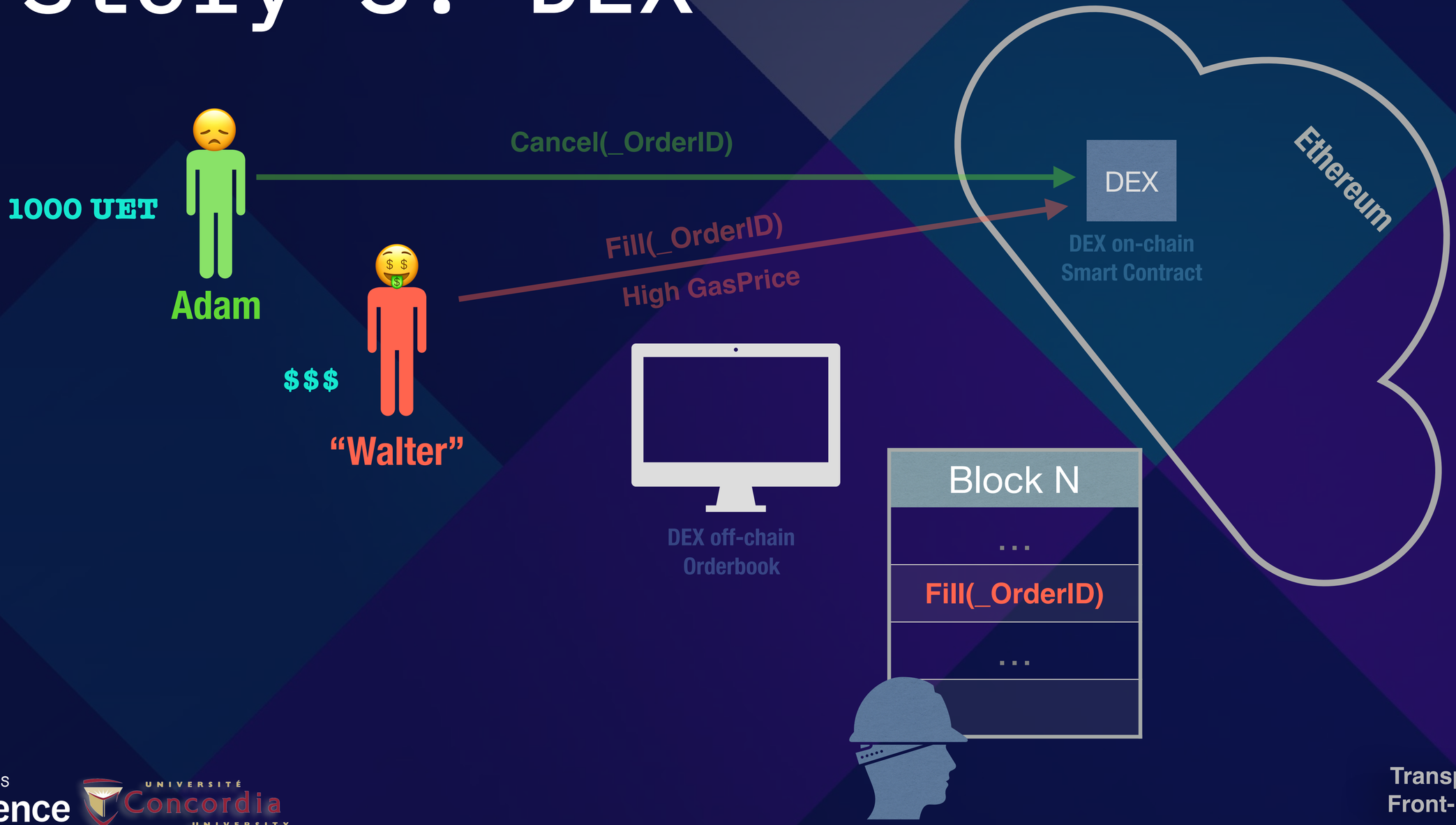
Story 3: DEX


Cancel(_OrderID)
Fill(_OrderID)
High GasPrice
Adam
"Walter"
Buy(1000 UET)
DEX off-chain
Orderbook
DEX
DEX on-chain
Smart Contract
Ethereum

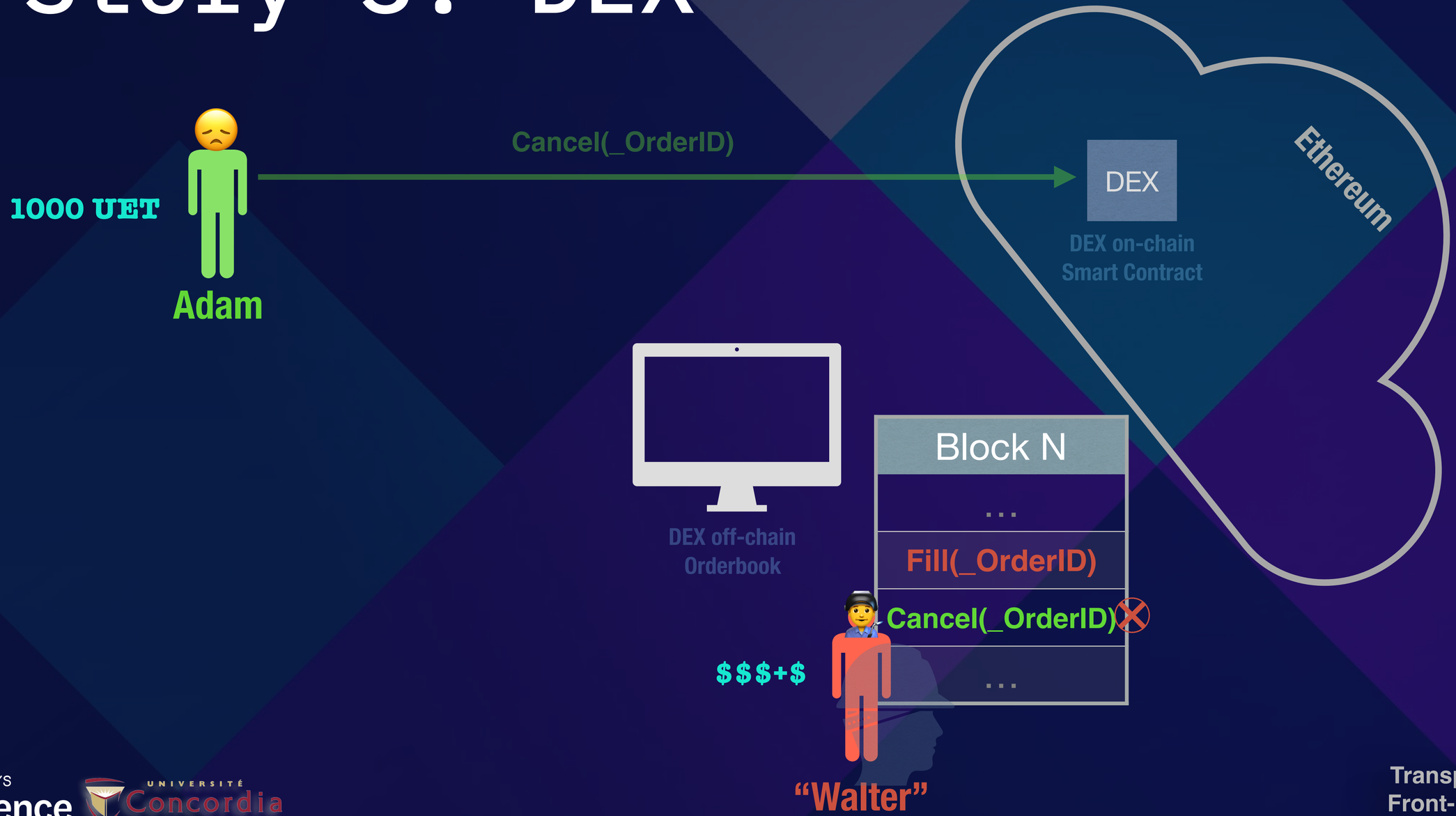
Story 3: DEX


Cancel(_OrderID)
Fill(_OrderID)
High GasPrice
Adam
"Walter"
DEX
DEX on-chain
Smart Contract
Ethereum
Buy(1000 UET)
DEX off-chain
Orderbook
Block N
...
Fill(_OrderID)
...

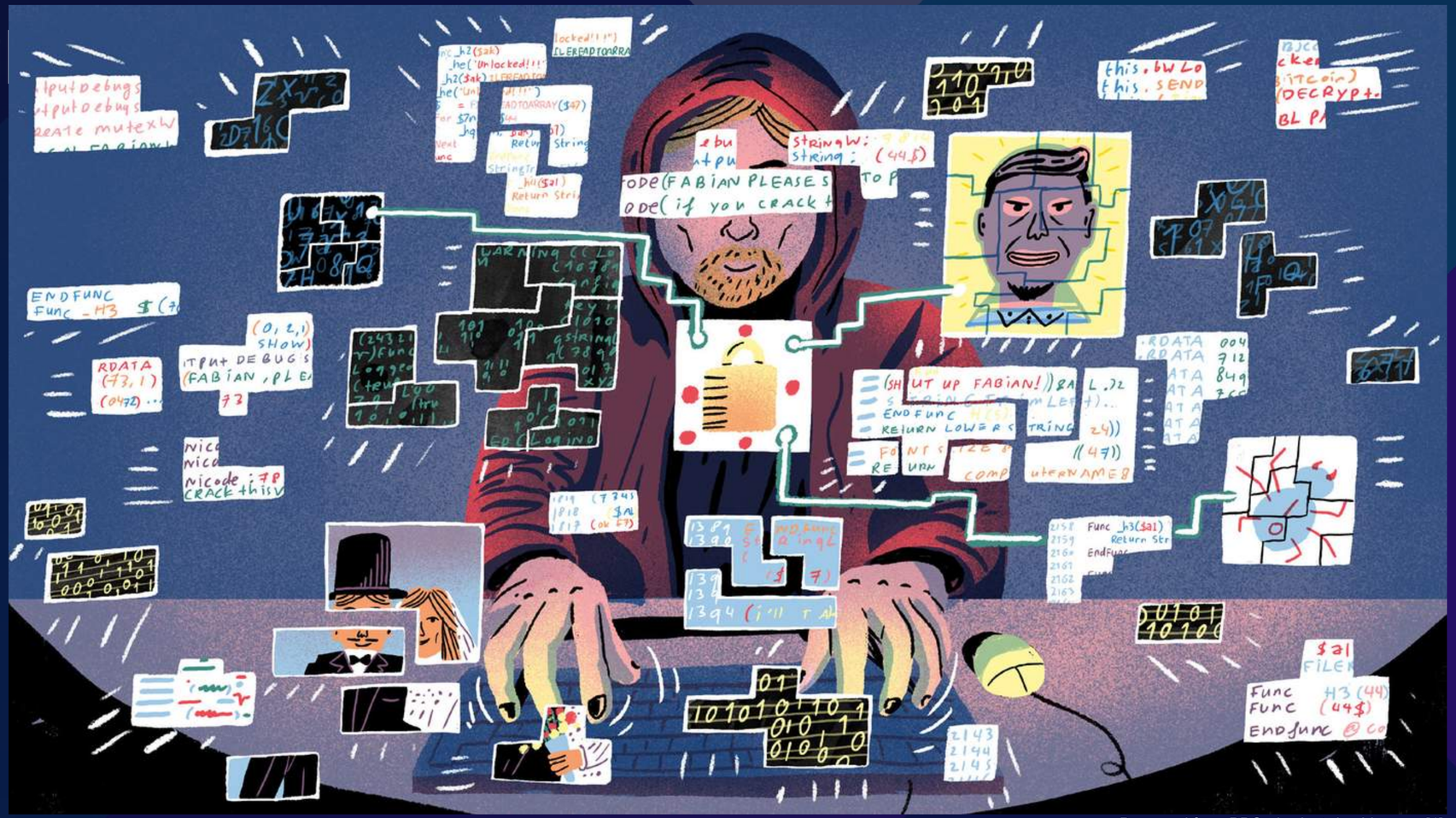
Story 3: DEX


1000 UET
Adam
Cancel(_OrderID)
Fill(_OrderID)
High GasPrice
$$$
"Walter"
DEX
DEX on-chain
Smart Contract
Ethereum
DEX off-chain
Orderbook
Block N
...
Fill(_OrderID)
...

Story 3: DEX


1000 UET
Adam
Cancel(_OrderID)
DEX
DEX on-chain
Smart Contract
Ethereum
DEX off-chain
Orderbook
Block N
...
Fill(_OrderID)
Cancel(_OrderID)
...
$$$+$
"Walter"

What do these stories have in com



Borrowed from BBC - by Aart-Jan Venema [1]

Front-running

They are all called “Front-running” attacks within the community.
But are they all the same attack?

Traditional Front-running

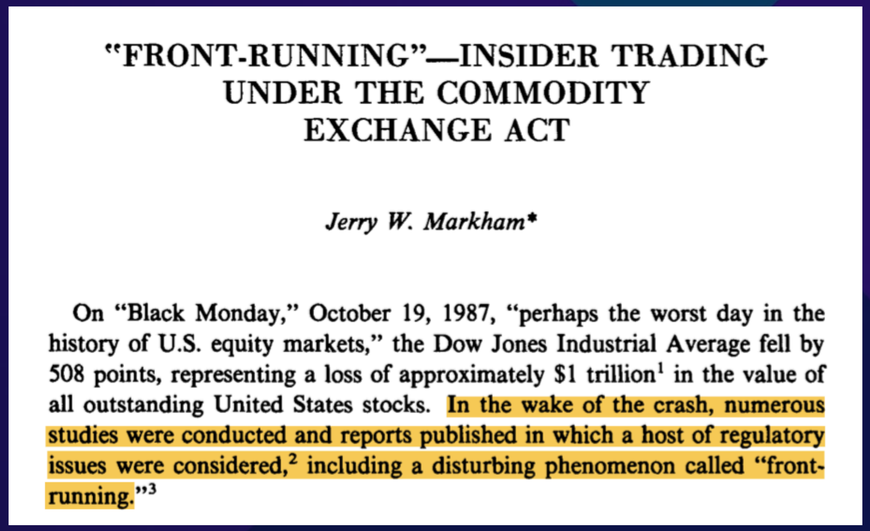
Front-running is a course of action where someone:

- Benefits from early access to market information about upcoming transactions and trades
  - typically because of a privileged position along the transmission of this information

Traditional Front-running

Front-running is a course of action where someone:

- Benefits from early access to market information about upcoming transactions and trades
  - typically because of a privileged position along the transmission of this information



"FRONT-RUNNING"—INSIDER TRADING UNDER THE COMMODITY EXCHANGE ACT

*Jerry W. Markham**

On "Black Monday," October 19, 1987, "perhaps the worst day in the history of U.S. equity markets," the Dow Jones Industrial Average fell by 508 points, representing a loss of approximately $1 trillion[1] in the value of all outstanding United States stocks. In the wake of the crash, numerous studies were conducted and reports published in which a host of regulatory issues were considered,[2] including a disturbing phenomenon called "front-running."[3]

Blockchain Front-running

Blockchain Front-running

- Everyone (Full Nodes) in the network have access to “Privilege Information”

Blockchain Front-running

- Everyone (Full Nodes) in the network have access to “Privilege Information”
  - Miners are in a more privileged position
    —> Order of transactions in a block they mine

Blockchain Front-running

- Everyone (Full Nodes) in the network have access to “Privilege Information”
  - Miners are in a more privileged position
    —> Order of transactions in a block they mine
- Miners can be “bribed” by transaction fee / gasPrice A.K.A *Gas Auction*

Taxonomy of Front-running attacks

Attack Type	Description	Example
Displacement	**Not important** to the adversary for original function call **to run after her function**.	Domain Name Registration
Insertion	**Important** to the adversary for original function call to **run after her function**.	Buy(_price), _price > Best offer
Suppression*	Run Function and **delay original function** call (Or any other)	Fomo3D

* A.K.A Block Stuffing

Variants	Description	Example
Asymmetric	**Different function than the original caller**	Cancellation Griefing
Bulk	**Run Large set of functions**	Capped ICO

Story 1: Status ICO

emansipater 189 points · 2 years ago

I assume that (unlike all the price discussion here which is totally offtopic) you are referring to the transaction issues which have led several exchanges to pause ETH withdrawals. Here is what happened:

The badly designed Status ICO clogged up the network yesterday with a huge number of high gas fee transactions, most of which are failing but still filling up the blocks and preventing normal tx's from getting in.

In addition, dwarfpool and perhaps others have set bad defaults on their client software that both actually cost themselves money and also prevent the network from automatically adapting to larger gas volumes the way it's supposed to.

Furthermore, evidence is accumulating that f2pool was actively manipulating transactions bound for the Status ICO, which they participated in themselves, exacerbating the problem. Experts explained weeks ago that bad ICO designs are vulnerable to such attacks, but this appears to be the first time it was actually executed in the wild.

So now, even though the Status ICO is over, there are still a huge number of transactions clogging up the network and the only way to get transactions in is to pay huge fees (which most of the exchanges probably don't want to do). Until it clears out, people are going to be missing ENS auctions, unable to withdraw from many wallets and exchanges, etc. etc. etc.

TL;DR: badly designed ICOs, plus selfish and foolish miners = major delays and maybe even substantial losses for everyone else.

Give Award Share Report Save

Story 1: Status ICO

emansipater 189 points · 2 years ago

I assume that (unlike all the price discussion here which is totally offtopic) you are referring to the transaction issues which have led several exchanges to pause ETH withdrawals. Here is what happened:

The badly designed Status ICO clogged up the network yesterday with a huge number of high gas fee transactions, most of which are failing but still filling up the blocks and preventing normal tx's from getting in.

In addition, dwarfpool and perhaps others have set bad defaults on their client software that both actually cost themselves money and also prevent the network from automatically adapting to larger gas volumes the way it's supposed to.

Furthermore, evidence is accumulating that f2pool was actively manipulating transactions bound for the Status ICO, which they participated in themselves, exacerbating the problem. Experts explained weeks ago that bad ICO designs are vulnerable to such attacks, but this appears to be the first time it was actually executed in the wild.

So now, even though the Status ICO is over, there are still a huge number of transactions clogging up the network and the only way to get transactions in is to pay huge fees (which most of the exchanges probably don't want to do). Until it clears out, people are going to be missing ENS auctions, unable to withdraw from many wallets and exchanges, etc. etc. etc.

TL;DR: badly designed ICOs, plus selfish and foolish miners = major delays and maybe even substantial losses for everyone else.

Give Award Share Report Save

Posted by u/emansipater 2 years ago

43

Collecting information about f2pool

Yesterday in my widely read comment explaining what was going on in the network, I suggested that there was a possibility f2pool had actively manipulated transactions bound for the Status ICO.

I'm now less convinced of that than I was, and wanted to make a thread where we could aggregate all the available information to make a better assessment.

What I knew when I wrote the comment:

Suspicion was originally cast on f2pool by blocks such as 3903912 and 3903913, the first blocks f2pool successfully mined after the Status ICO began accepting contributions at block height 3903900. The Status ICO contract permitted a maximum of 50 Gwei gas price for non-whitelisted transactions, however many users appear to have been not properly aware of this as even before the proper block height the network was flooded with transactions paying a higher gas price than this (example). At activation height most other blocks being mined (such as 3903901, 3903902,3903903,3903904,3903909,3903910,3903911) were flooded with these transactions since the default strategy of mining nodes is to prefer higher fee transactions. You can see that in blocks 3903901 to 3903907 the only transactions without a red exclamation mark going to the Status ICO are the ones who were whitelisted to pay above 50 Gwei fees, such as this one. There are also some non-Status-related transactions paying a >50 Gwei price. Compared with these blocks, the f2pool blocks are visibly different. Every transaction in them successfully participates in the crowdsale despite not being whitelisted and using a fee below 50 Gwei. Most of the amounts descend smoothly in size: 100 ETH, 99.99, 99.98, 99.97, 99.96, 99.95, 99.93 etc. down to 99.7 (presumably this is due to how the "dynamic ceiling" part of the crowdsale works at that point?), and then there are a few smaller transactions. **ALL** of the ~100 ETH transactions originate directly from f2pool itself or an account which f2pool funded with 100 ETH ~1 hr before the crowdsale. **ALL** of the smaller transactions originate from IBAN-compatible accounts, probably created by Parity. There are NO >50 Gwei transactions included in

Story 1: Status ICO

emansipater 189 points · 2 years ago

I assume that (unlike all the price discussion here which is totally offtopic) you are referring to the transaction issues which have led several exchanges to pause ETH withdrawals. Here is what happened:

The badly designed Status ICO clogged
fee transactions, most of which are failin
from getting in.

In addition, dwarfpool and perhaps othe
actually cost themselves money and also
larger gas volumes the way it's suppose

Furthermore, evidence is accumulating
for the Status ICO, which they participat
explained weeks ago that bad ICO desig
the first time it was actually executed in

So now, even though the Status ICO is o
up the network and the only way to get
exchanges probably don't want to do). U
auctions, unable to withdraw from many

**TL;DR: badly designed ICOs, plus selfis
substantial losses for everyone else.**

Give Award Share Report Save

UPDATE: F2Pool Manipulates $1.2 Million on the Ethereum Blockchain During the Status.im ICO

dhumphrey (61) in ethereum • 3 years ago

A redditor (/u/blueseeker) recently uncovered evidence that F2Pool is manipulating the Ethereum blockchain to ensure that they'll be the first (and also nearly the only people) to invest in the new Status.im ICO. This piqued my interest and so I dug deeper. This is what I found:

- First they made a ton of different addresses (30, to be exact) and sent 100 ETH to each of those. Examples include: this one and this one. There are many more to be found if you look at their transactions.
- Then they stopped including transactions in their blocks a half an hour before the ICO was due to launch.
- In their first blocks discovered after the ICO, they included only the

Posted by u/emansipater 2 years ago

43

Collecting information about f2pool

n my widely read comment explaining what was going on in the network, I suggested that
a possibility f2pool had actively manipulated transactions bound for the Status ICO.

s convinced of that than I was, and wanted to make a thread where we could aggregate all
le information to make a better assessment.

ew when I wrote the comment:

was originally cast on f2pool by blocks such as 3903912 and 3903913, the first blocks f2pool
y mined after the Status ICO began accepting contributions at block height 3903900. The
contract permitted a maximum of 50 Gwei gas price for non-whitelisted transactions,
any users appear to have been not properly aware of this as even before the proper block
network was flooded with transactions paying a higher gas price than this (example). At
eight most other blocks being mined (such as 3903901,
903903,3903904,3903909,3903910,3903911) were flooded with these transactions since the
tegy of mining nodes is to prefer higher fee transactions. You can see that in blocks
3903907 the only transactions without a red exclamation mark going to the Status ICO are
ho were whitelisted to pay above 50 Gwei fees, such as this one. There are also some non-
ted transactions paying a >50 Gwei price. Compared with these blocks, the f2pool blocks are
rent. Every transaction in them successfully participates in the crowdsale despite not being
and using a fee below 50 Gwei. Most of the amounts descend smoothly in size: 100 ETH,
8, 99.97, 99.96, 99.95, 99.93 etc. down to 99.7 (presumably this is due to how the "dynamic
t of the crowdsale works at that point?), and then there are a few smaller transactions. **ALL**
ETH transactions originate directly from f2pool itself or an account which f2pool funded
H ~1 hr before the crowdsale. **ALL** of the smaller transactions originate from IBAN-
accounts, probably created by Parity. There are NO >50 Gwei transactions included in

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Story 1: Status ICO

emansipater 189 points · 2 years ago

I assume that (unlike all the price discussion here which is totally offtopic) you are referring to the transaction issues which have led several exchanges to pause ETH withdrawals. Here is what happened:

The badly designed Status ICO clogged
fee transactions, most of which are failin
from getting in.

In addition, dwarfpool and perhaps othe
actually cost themselves money and also
larger gas volumes the way it's suppose

Furthermore, evidence is accumulating
for the Status ICO, which they participat
explained weeks ago that bad ICO desig
the first time it was actually executed in

So now, even though the Status ICO is o
up the network and the only way to get
exchanges probably don't want to do). U
auctions, unable to withdraw from many

**TL;DR: badly designed ICOs, plus selfis
substantial losses for everyone else.**

Give Award Share Report Save

Posted by u/emansipater 2

43

Collecting infor

n my widel
a possibilit

s convince
le informat

ew when I

was origina
y mined aft
contract pe
any users a
network wa
eight most
903903,390
ategy of mi
3903907 t
ho were wh
ted transac
rent. Every
and using
8, 99.97, 99
t of the cro
ETH transa
H ~1 hr be
accounts,

UPDATE: F2Pool Manipulates $1.2 Million on the Ethereum Blockchain During the Status.im ICO

dhumphrey (61) in ethereum • 3 years ago

A redditor (/u/blueseeker) recently uncovered evidence that F2Pool is manipulating the Ethereum blockchain to ensure that they'll be the first (and also nearly the only people) to invest in the new Status.im ICO. This piqued my interest and so I dug deeper. This is what I found:

- First they made a ton of different addresses (30, to be exact) and sent 100 ETH to each of those. Examples include: this one and this one. There are many more to be found if you look at their transactions.
- Then they stopped including transactions in their blocks a half an hour before the ICO was due to launch.
- In their first blocks discovered after the ICO, they included only the

Altcoin

Status.im ICO Accused of Shady Practices

June 22, 2017 Shivam Chawla 3 Comments ethereum, ico, status.im

Status.im ICO took place on June 20,2017 raising more than $270 million making it the most successful ICO of the history of cryptocurrencies. Despite being such a successful ICO, Status.im has been accused of using shady practices in the ICO.

Issues with Status.im ICO

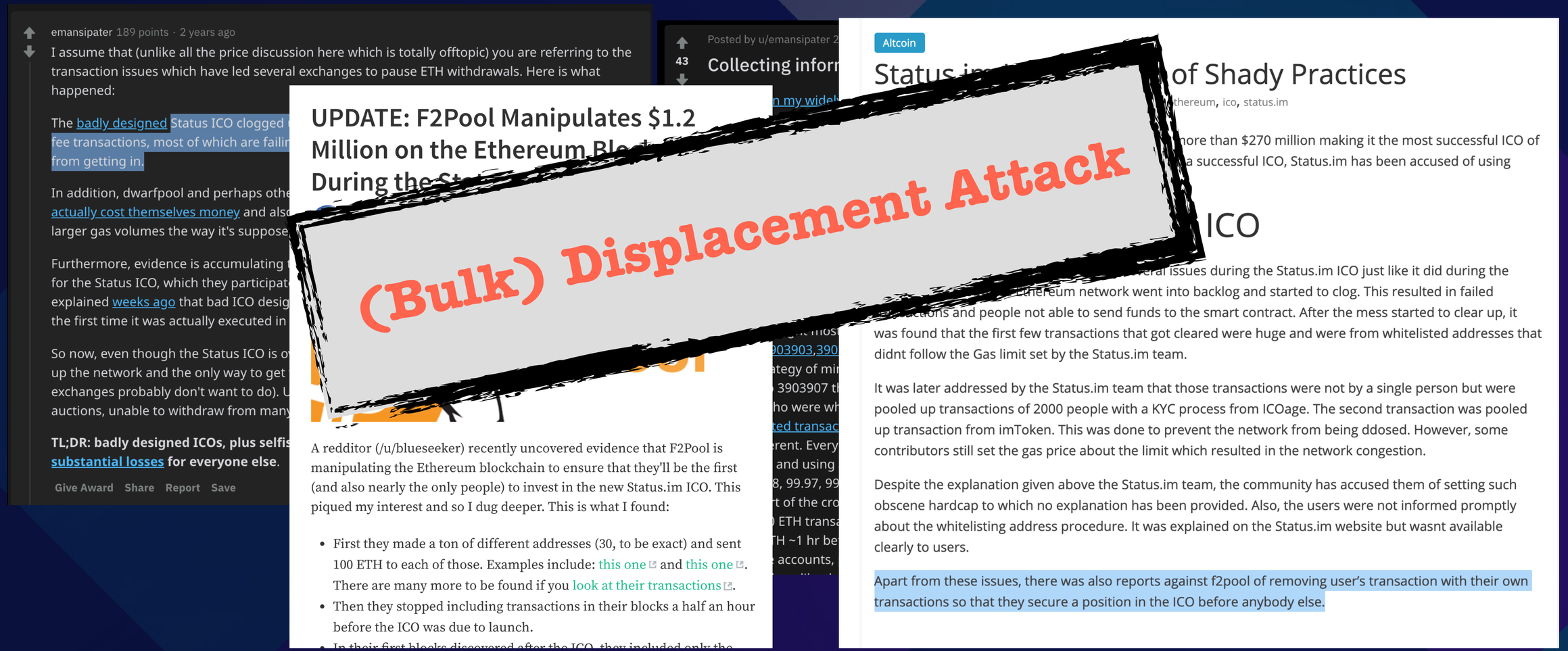
The Ethereum network once again faced several issues during the Status.im ICO just like it did during the Bancor ICO. The whole Ethereum network went into backlog and started to clog. This resulted in failed transactions and people not able to send funds to the smart contract. After the mess started to clear up, it was found that the first few transactions that got cleared were huge and were from whitelisted addresses that didnt follow the Gas limit set by the Status.im team.

It was later addressed by the Status.im team that those transactions were not by a single person but were pooled up transactions of 2000 people with a KYC process from ICOage. The second transaction was pooled up transaction from imToken. This was done to prevent the network from being ddosed. However, some contributors still set the gas price about the limit which resulted in the network congestion.

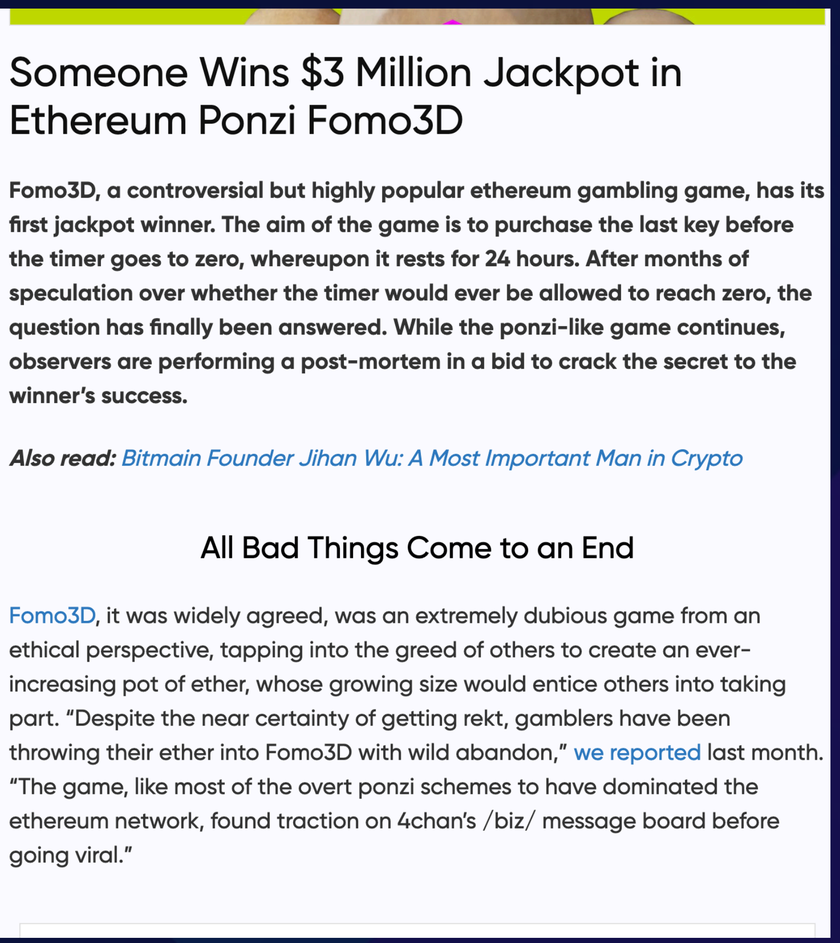
Despite the explanation given above the Status.im team, the community has accused them of setting such obscene hardcap to which no explanation has been provided. Also, the users were not informed promptly about the whitelisting address procedure. It was explained on the Status.im website but wasnt available clearly to users.

Apart from these issues, there was also reports against f2pool of removing user's transaction with their own transactions so that they secure a position in the ICO before anybody else.

Story 1: Status ICO



Story 2: FOMO3D



Someone Wins $3 Million Jackpot in Ethereum Ponzi Fomo3D

Fomo3D, a controversial but highly popular ethereum gambling game, has its first jackpot winner. The aim of the game is to purchase the last key before the timer goes to zero, whereupon it rests for 24 hours. After months of speculation over whether the timer would ever be allowed to reach zero, the question has finally been answered. While the ponzi-like game continues, observers are performing a post-mortem in a bid to crack the secret to the winner's success.

Also read: *Bitmain Founder Jihan Wu: A Most Important Man in Crypto*

All Bad Things Come to an End

Fomo3D, it was widely agreed, was an extremely dubious game from an ethical perspective, tapping into the greed of others to create an ever-increasing pot of ether, whose growing size would entice others into taking part. "Despite the near certainty of getting rekt, gamblers have been throwing their ether into Fomo3D with wild abandon," we reported last month. "The game, like most of the overt ponzi schemes to have dominated the ethereum network, found traction on 4chan's /biz/ message board before going viral."

Story 2: FOMO3D

Someone Wins $3 Million Jackpot in Ethereum Ponzi Fomo3D

Fomo3D, a controversial but highly popular ethereum gambling game, has its first jackpot winner. The aim of the game is to purchase the last key before the timer goes to zero, whereupon it rests for 24 hours. After months of speculation over whether the timer would ever be allowed to reach zero, the question has finally been answered. While the ponzi-like game continues, observers are performing a post-mortem in a bid to crack the secret to the winner's success.

Also read: *Bitmain Founder Jihan Wu: A Most Important Man in Crypto*

All Bad Things Come to an End

Fomo3D, it was widely agreed, was an extremely dubious game from an ethical perspective, tapping into the greed of others to create an ever-increasing pot of ether, whose growing size would entice others into taking part. "Despite the near certainty of getting rekt, gamblers have been throwing their ether into Fomo3D with wild abandon," we reported last month. "The game, like most of the overt ponzi schemes to have dominated the ethereum network, found traction on 4chan's /biz/ message board before going viral."

The first round of Fomo3D has ended and the winner's address is 0xa169, with a prize of 10,469.66 ethers.

You may think that the winner is just an ordinary participant.

SECBIT Labs first found that the Fomo3D winner played a special attack trick to sharply decrease the number of transactions packed by miners near the end of the game (multiple blocks were involved), accelerating the approach of the game end and increasing the winning probability. SECBIT Labs also observed several similar abnormal blocks and transactions regarding to the last round of Last Winner.

A Series of Abnormal Blocks and Transactions

6191908	21 hrs 35 mins ago	5	0	0x2a5994b501e6a5...	7991000 (99.94%)	7996106	499.95 Gwei	6.99509 Ether
6191907	21 hrs 35 mins ago	4	0	BitClubPool	7979000 (99.83%)	7992222	741.29 Gwei	8.91475 Ether
6191906	21 hrs 35 mins ago	3	0	Nanopool	8000000 (100.00%)	8000029	501.00 Gwei	7.00803 Ether
6191905	21 hrs 35 mins ago	7	0	MiningPoolHub_1	7984000 (99.80%)	8000029	495.98 Gwei	6.95992 Ether
6191904	21 hrs 36 mins ago	3	0	Nanopool	8000000 (100.00%)	8000029	190.00 Gwei	4.52003 Ether
6191903	21 hrs 36 mins ago	6	0	Special Tricks	7984000 (99.80%)	8000029	188.34 Gwei	4.50374 Ether
6191902	21 hrs 37 mins ago	46	0	Ethermine	7978342 (99.83%)	7992259	19.48 Gwei	3.15541 Ether
6191901	21 hrs 37 mins ago	15	0	SparkPool	7979663 (99.94%)	7984489	22.07 Gwei	3.1761 Ether
6191900	21 hrs 37 mins ago	10	0	Nanopool	7979192 (99.84%)	7992259	22.87 Gwei	3.18251 Ether
6191899	21 hrs 37 mins ago	34	0	0xd9580260be45c3...	7975461 (99.89%)	7984464	18.45 Gwei	3.14713 Ether
6191898	21 hrs 37 mins ago	25	0	SparkPool	7980081 (99.99%)	7980567	15.85 Gwei	3.12648 Ether
6191897	21 hrs 37 mins ago	103	0	bw	3948328 (45.67%)	7988343	8.74 Gwei	3.03188 Ether

Normal block

Story 2: FOMO3D

Someone Wir
Ethereum Pon

**Fomo3D, a controversial
first jackpot winner. The
the timer goes to zero, w
speculation over whethe
question has finally beer
observers are performin
winner's success.**

Also read: *Bitmain Found*

All B

Fomo3D, it was widely a
ethical perspective, tapp
increasing pot of ether, w
part. "Despite the near c
throwing their ether into
"The game, like most of t
ethereum network, found
going viral."

MOTHERBOARD
TECH BY VICE

A Wildly Popular Ethereum Gambling Game Just Paid Out 3 Million Dollars

'Fomo3D' is a controversial and popular Ethereum application, and the lottery's winner may have clogged the blockchain itself to win.

By **Jordan Pearson**

Aug 23 2018, 3:45pm Share Tweet Snap

SCREENGRABS: FOMO3D, KNOWYOURMEME. COMPOSITION: AUTHOR

The **most popular application for Ethereum right now** isn't **digital kitty collectibles** (such innocent days)—it's a depraved gambling game called Fomo3D that describes itself as "a psychological social experiment in greed." On Wednesday, the first round of the game ended and paid out a jackpot worth roughly $3 million USD in ether to a player.

The first round of Fomo3D has ended and the winner's address is 0xa169, with a prize of 10,469.66 ethers.

You may think that the winner is just an ordinary participant.

SECBIT Labs first found that the Fomo3D winner played a special attack trick to sharply decrease the number of transactions packed by miners near the end of the game (multiple blocks were involved), accelerating the approach of the game end and increasing the winning probability. SECBIT Labs also observed several similar abnormal blocks and transactions regarding to the last round of Last Winner.

A Series of Abnormal Blocks and Transactions

6191908	21 hrs 35 mins ago	5	0	0x2a5994b501e6a5...	7991000 (99.94%)	7996106	499.95 Gwei	6.99509 Ether
6191907	21 hrs 35 mins ago	4	0	BitClubPool	7979000 (99.83%)	7992222	741.29 Gwei	8.91475 Ether
6191906	21 hrs 35 mins ago	3	0	Nanopool	8000000 (100.00%)	8000029	501.00 Gwei	7.00803 Ether
6191905	21 hrs 35 mins ago	7	0	MiningPoolHub_1	7984000 (99.80%)	8000029	495.98 Gwei	6.95992 Ether
6191904	21 hrs 36 mins ago	3	0	Nanopool	8000000 (100.00%)	8000029	190.00 Gwei	4.52003 Ether
6191903	21 hrs 36 mins ago	6	0		7984000 (99.80%)	8000029	188.34 Gwei	4.50374 Ether
6191902	21 hrs 37 mins ago	46	0	Ethermine	7978342 (99.83%)	7992259	19.48 Gwei	3.15541 Ether
6191901	21 hrs 37 mins ago	15	0	SparkPool	7979663 (99.94%)	7984489	22.07 Gwei	3.1761 Ether
6191900	21 hrs 37 mins ago	10	0	Nanopool	7979192 (99.84%)	7992259	22.87 Gwei	3.18251 Ether
6191899	21 hrs 37 mins ago	34	0	0xd9580260be45c3...	7975461 (99.89%)	7984464	18.45 Gwei	3.14713 Ether
6191898	21 hrs 37 mins ago	25	0	SparkPool	7980081 (99.99%)	7980567	15.85 Gwei	3.12648 Ether
6191897	21 hrs 37 mins ago	103	0	bw	7948328 (45.67%)	7988343	8.74 Gwei	3.03188 Ether

Special Tricks

Normal block

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Story 2: FOMO3D

Someone Wir
Ethereum Pon

**Fomo3D, a controversial
first jackpot winner. The
the timer goes to zero, w
speculation over whethe
question has finally beer
observers are performin
winner's success.**

Also read: *Bitmain Found*

All B

Fomo3D, it was widely a
ethical perspective, tapp
increasing pot of ether, v
part. "Despite the near c
throwing their ether into
"The game, like most of t
ethereum network, found
going viral."

MOTHERBOARD
TECH BY VICE

A Wildly Popular Ethereum Gambling Game Just Paid Out 3 Million Dollars

'Fomo3D' is a controversial and popular Ethereum application, and the lottery's winner may have clogged the blockchain itself to win.

By **Jordan Pearson**

Aug 23 2018, 3:45pm Share Tweet Snap

SCREENGRABS: FOMO3D, KNOWYOURMEME. COMPOSITION: AUTHOR

The **most popular application for Ethereum right now** isn't **digital kitty collectibles** (such innocent days)—it's a depraved gambling game called Fomo3D that describes itself as "a psychological social experiment in greed." On Wednesday, the first round of the game ended and paid out a jackpot worth roughly $3 million USD in ether to a player.

The first round of Fomo3D has enc
with a prize of 10,469.66 ethers.

You may think that the winner is ju

SECBIT Labs first found that the F
trick to sharply decrease the numb
the end of the game (multiple bloc
approach of the game end and incr
Labs also observed several similar
regarding to the last round of Last

A Series of Abnormal Bl

6191908	21 hrs 35 mins ago	5	0	0x2a5994b501e6
6191907	21 hrs 35 mins ago	4	0	BitClubPool
6191906	21 hrs 35 mins ago	3	0	Nanopool
6191905	21 hrs 35 mins ago	7	0	MiningPoolHub_1
6191904	21 hrs 36 mins ago	3	0	Nanopool
6191903	21 hrs 36 mins ago	6	0	Special Tricks
6191902	21 hrs 37 mins ago	46	0	Ethermine
6191901	21 hrs 37 mins ago	15	0	SparkPool
6191900	21 hrs 37 mins ago	10	0	Nanopool
6191899	21 hrs 37 mins ago	34	0	0xd9580260be45
6191898	21 hrs 37 mins ago	25	0	SparkPool
6191897	21 hrs 37 mins ago	103	0	bw

Normal block

80081 (99.99%)	7980567	15.85 Gwei	3.12648 Ether
48328 (45.67%)	7988343	8.74 Gwei	3.03188 Ether

The Anatomy of a Block Stuffing Attack

How the first round of FOMO3D was won

Onur Solmaz Following
Oct 17, 2018 · 7 min read

Block stuffing is a type of attack in blockchains where an attacker submits transactions that deliberately fill up the block's gas limit and stall other transactions. To ensure inclusion of their transactions by miners, the attacker can choose to pay higher transaction fees. By controlling the amount of gas spent by their transactions, the attacker can influence the number of transactions that get to be included in the block.

The attack is like constantly cutting in line using your money.

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Story 2: FOMO3D

Someone Wir
Ethereum Pon

**Fomo3D, a controversial
first jackpot winner. The
the timer goes to zero, w
speculation over whethe
question has finally beer
observers are performin
winner's success.**

Also read: *Bitmain Found*

All B

Fomo3D, it was widely a
ethical perspective, tapp
increasing pot of ether, v
part. "Despite the near c
throwing their ether into
"The game, like most of t
ethereum network, found
going viral."

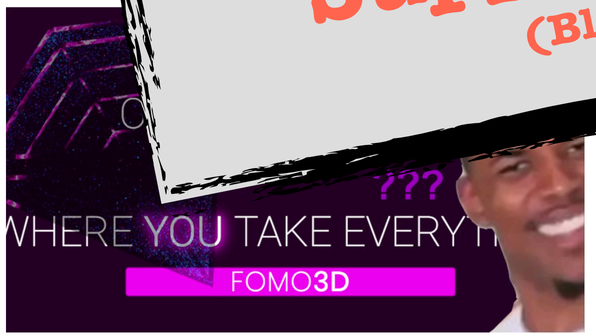
MOTHERBOARD
TECH BY VICE

A Wildly Popular Ethereum Gambling Game Just Paid Out 3 Million Dollars

'Fomo3D' is a controversial and popul
the lottery's winner m

By Jordan Pearso

Aug 23 2018, 3:45pm



SCREENGRABS: FOMO3D, KNOWYOURMEME. COMPOSITION: AUTHOR

The **most popular application for Ethereum right now** isn't **digital kitty collectibles** (such innocent days)—it's a depraved gambling game called Fomo3D that describes itself as "a psychological social experiment in greed." On Wednesday, the first round of the game ended and paid out a jackpot worth roughly $3 million USD in ether to a player.

Suppression Attack
(Block Stuffing attack)

The first round of Fomo3D has enc
with a prize of 10,469.66 ethers.

You

A Series of Abnormal Bl

6191908	21 hrs 35 mins ago	5	0	0x2a5994b501e6
6191907	21 hrs 35 mins ago	4	0	BitClubPool
6191906	21 hrs 35 mins ago	3	0	Nanopool
6191905	21 hrs 35 mins ago	7	0	MiningPoolHub_1
6191904	21 hrs 36 mins ago	3	0	Nanopool
6191903	21 hrs 36 mins ago	6	0	Special Tricks
6191902	21 hrs 37 mins ago	46	0	Ethermine
6191901	21 hrs 37 mins ago	15	0	SparkPool
6191900	21 hrs 37 mins ago	10	0	Nanopool
6191899	21 hrs 37 mins ago	34	0	0xd9580260be45
6191898	21 hrs 37 mins ago	25	0	SparkPool
6191897	21 hrs 37 mins ago	103	0	bw

Normal block

7980567 15.85 Gwei 3.12648 Ether
7988343 8.74 Gwei 3.03188 Ether

The Anatomy of a Block Stuffing Attack

O3D was won

blockchains where an attacker submits
p the block's gas limit and stall other
f their transactions by miners, the
ansaction fees. By controlling the
transactions, the attacker can influence the
of transactions that get to be included in the block.

The attack is like constantly cutting in line using your money.

CONSENSYS Diligence

UNIVERSITÉ Concordia UNIVERSITY

Taxonomy of Front-running attacks

Attack Type	Description	Example
Displacement	**Not important** to the adversary for original function call **to run after her function**.	Domain Name Registration
Insertion	**Important** to the adversary for original function call to **run after her function**.	Buy(_price), _price > Best offer
Suppression*	Run Function and **delay original function** call (Or any other)	Fomo3D

* A.K.A Block Stuffing

Variants	Description	Example
Asymmetric	**Different function than the original caller**	Cancellation Griefing
Bulk	**Run Large set of functions**	Capped ICO

Case Studies

DApp Category	Names	Rank
Exchanges	IDEX	1
	ForkDelta, EtherDelta	2
	Bancor	7
	The Token Store	13
	LocalEthereum	14
	Kyber	22
	0x Protocol	23
Crypto-Collectible Games (ERC-721 [26])	**CryptoKitties**	3
	Ethermon	4
	Cryptogirl	9
	Gods Unchained TCG	12
	Blockchain Cuties	15
	ETH.TOWN!	16
	0xUniverse	18
	MLBCrypto Baseball	19
	HyperDragons	25
Gambling	**Fomo3D**	5
	DailyDivs	6
	PoWH 3D	8
	FomoWar	10
	FairDapp	11
	Zethr	17
	dice2.win	20
	Ether Shrimp Farm	21
Name Services	**Ethereum Name Service**	24

- **Top 25 DApps**
  - Based on recent user activity
    - DAppRadar.com
    - September 2018
- A few ICOs as well

- **See the paper for detailed case studies**

Key Mitigations

1. Transaction Sequencing
2. Confidentiality
3. Design Practices

Transaction Sequencing

Transaction Sequencing

- Blockchain itself removes the (miner's) ability to arbitrarily order transactions

Transaction Sequencing

- Blockchain itself removes the (miner's) ability to arbitrarily order transactions
- First-in-first-out (FIFO) is generally not possible on a distributed network

Transaction Sequencing

- Blockchain itself removes the (miner's) ability to arbitrarily order transactions
- First-in-first-out (FIFO) is generally not possible on a distributed network
  - Go-Ethereum implementation prioritizes transactions based on their gas price and nonce

Transaction Sequencing

- Blockchain itself removes the (miner's) ability to arbitrarily order transactions
- First-in-first-out (FIFO) is generally not possible on a distributed network
  - Go-Ethereum implementation prioritizes transactions based on their gas price and nonce
  - Off-chain (e.g. Order books in 0x or EtherDelta)

Transaction Sequencing

- Blockchain itself removes the (miner's) ability to arbitrarily order transactions
- First-in-first-out (FIFO) is generally not possible on a distributed network
  - Go-Ethereum implementation prioritizes transactions based on their gas price and nonce
  - Off-chain (e.g. Order books in 0x or EtherDelta)
- Pseudorandom Sorting (e.g. Canonical Transaction Ordering Rule (CTOR) by Bitcoin Cash ABC)

Confidentiality

Limit the visibility of transactions

Confidentiality

Limit the visibility of transactions

- DApp interaction includes the following components:

1	**Code** of the DApp
2	Current **state** of the DApp
3	Name of the **function** being invoked
4	**Parameters** supplied to the function
5	**Address** of the contract the function is being invoked on
6	Identity of the **sender**.

Confidentiality

- **Privacy-Preserving Blockchains, similar to Dark pools in HFT**
  - (2,3,4)-confidential

1	Code of the DApp
2	**Current state of the DApp**
3	**Name of the function being invoked**
4	**Parameters supplied to the function**
5	Address of the contract
6	Identity of the sender.

Confidentiality

- **Privacy-Preserving Blockchains, similar to Dark pools in HFT**
  - (2,3,4)-confidential
- **Commit and Reveal.**
  - (3,4)- or (4)-confidentiality
  - Namecoin, ENS
  - Collateralized? —> Leaks information

1	Code of the DApp
2	Current state of the DApp
3	**Name of the function being invoked**
4	**Parameters supplied to the function**
5	Address of the contract
6	Identity of the sender.

Confidentiality

- **Privacy-Preserving Blockchains, similar to Dark pools in HFT**
  - (2,3,4)-confidential
- **Commit and Reveal.**
  - (3,4)- or (4)-confidentiality
  - Namecoin, ENS
  - Collateralized? —> Leaks information
- **Enhanced Commit and Reveal: Submarine Sends**
  - (3,4,5)-confidentiality+

1	Code of the DApp
2	Current state of the DApp
3	**Name of the function being invoked**
4	**Parameters supplied to the function**
5	**Address of the contract**
6	Identity of the sender.

Design Practices

Design Practices

- Assume front-running is unpreventable —> Remove any benefit from it

Design Practices

- Assume front-running is unpreventable —> Remove any benefit from it
  - remove the importance of transaction ordering or time

Design Practices

- Assume front-running is unpreventable —> Remove any benefit from it
  - remove the importance of transaction ordering or time
- **Call market** design instead of a time-sensitive order book

Design Practices

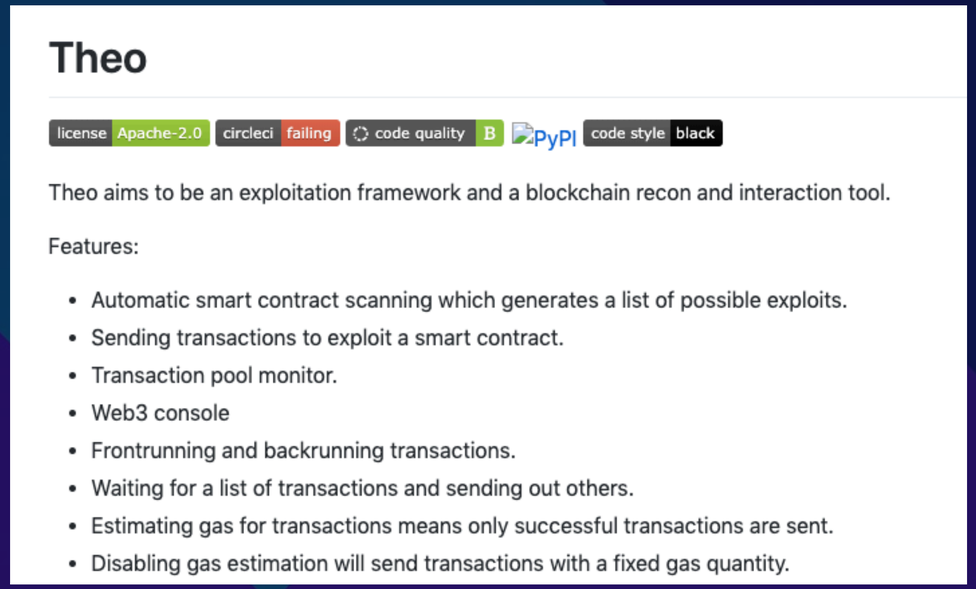
- Assume front-running is unpreventable —> Remove any benefit from it
  - remove the importance of transaction ordering or time
- **Call market** design instead of a time-sensitive order book
- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.

Concluding Remarks

- Front-running is a pervasive issue in Ethereum DApps
- Increase awareness of these type of attacks
- Usable DApp layer & Blockchain-level solutions
  - We highlight this as an important research area.

Future Works

- Theo: Front-running (and back-running) transactions
  https://github.com/cleanunicorn/theo
  - Coming up next: Front-running flash loans?



Theo

license Apache-2.0 | circleci failing | code quality B | PyPI | code style black

Theo aims to be an exploitation framework and a blockchain recon and interaction tool.

Features:

- Automatic smart contract scanning which generates a list of possible exploits.
- Sending transactions to exploit a smart contract.
- Transaction pool monitor.
- Web3 console
- Frontrunning and backrunning transactions.
- Waiting for a list of transactions and sending out others.
- Estimating gas for transactions means only successful transactions are sent.
- Disabling gas estimation will send transactions with a fixed gas quantity.

- Mythx: Smart contract code analyzer
  Modules to detect different front-running attacks

MythX

CONSENSYS
Diligence

Thank You

Get in touch:

Shayan Eskandari (@sbetamc)

shayan.eskandari@consensys.net

https://shayan.es

UNIVERSITÉ Concordia UNIVERSITY

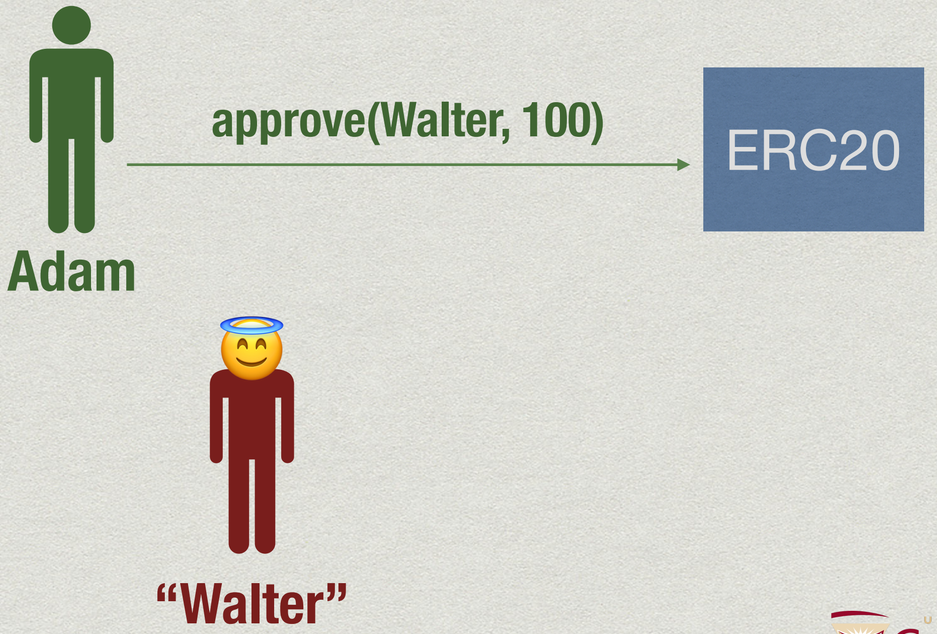
SOK: TRANSPARENT DISHONESTY FRONT-RUNNING ATTACKS ON BLOCKCHAIN.

Take-home readings:

- The paper itself : https://arxiv.org/abs/1902.05164
- Front-Running - Insider Trading under the Commodity Exchange Act (1988) - Jerry W. Markham
- Flash Boys: A Wall Street Revolt
  - Youtube: "Brad Katsuyama - The Stock Market had become an Illusion”
- Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges: https://arxiv.org/abs/1904.05234

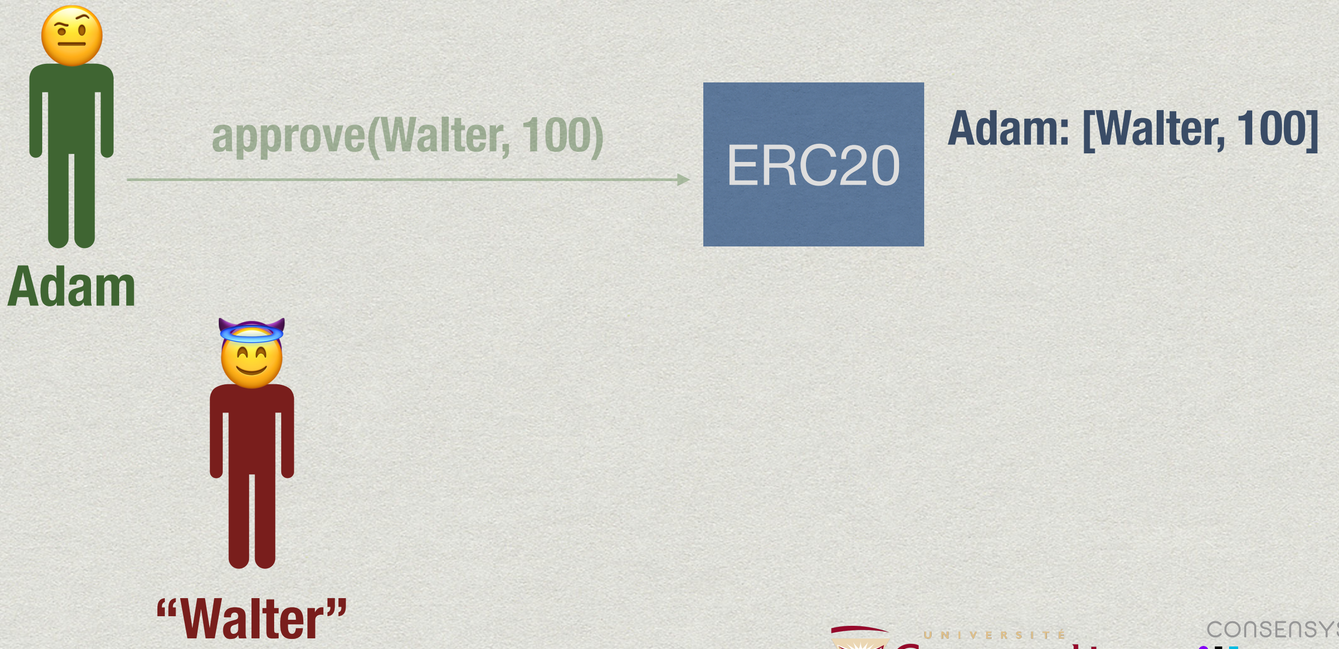
Design Practices (cont.)

- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



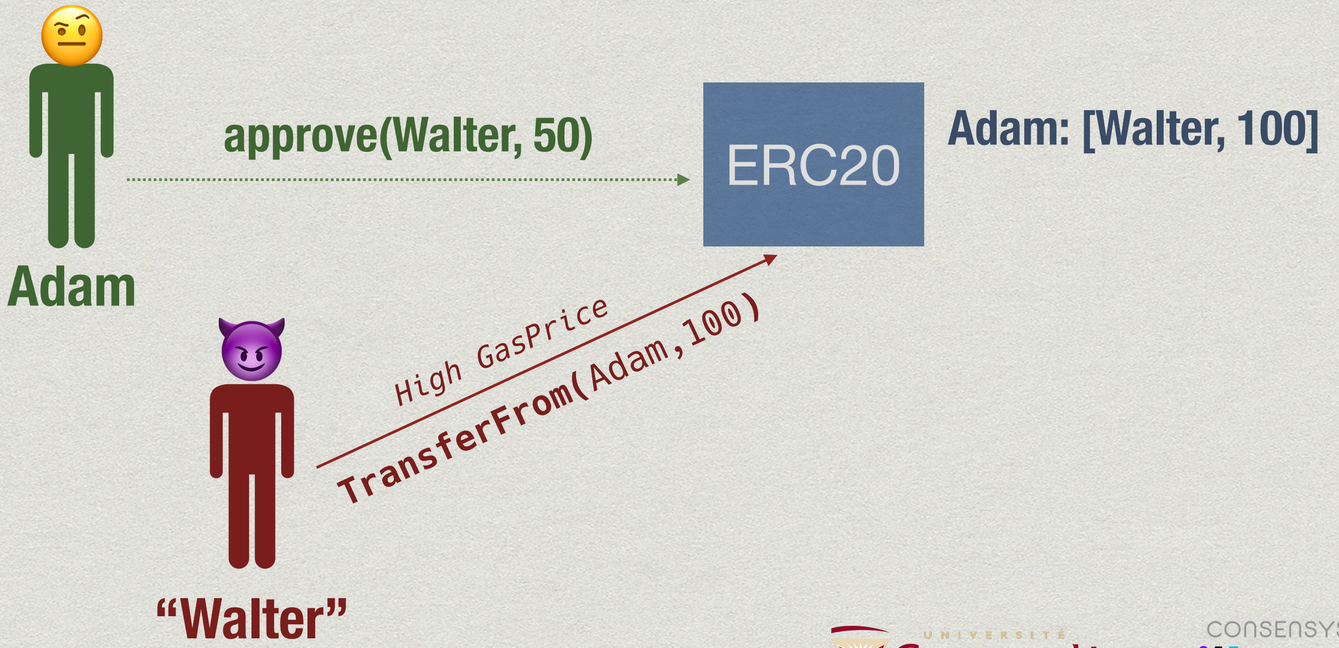
Design Practices (cont.)

- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



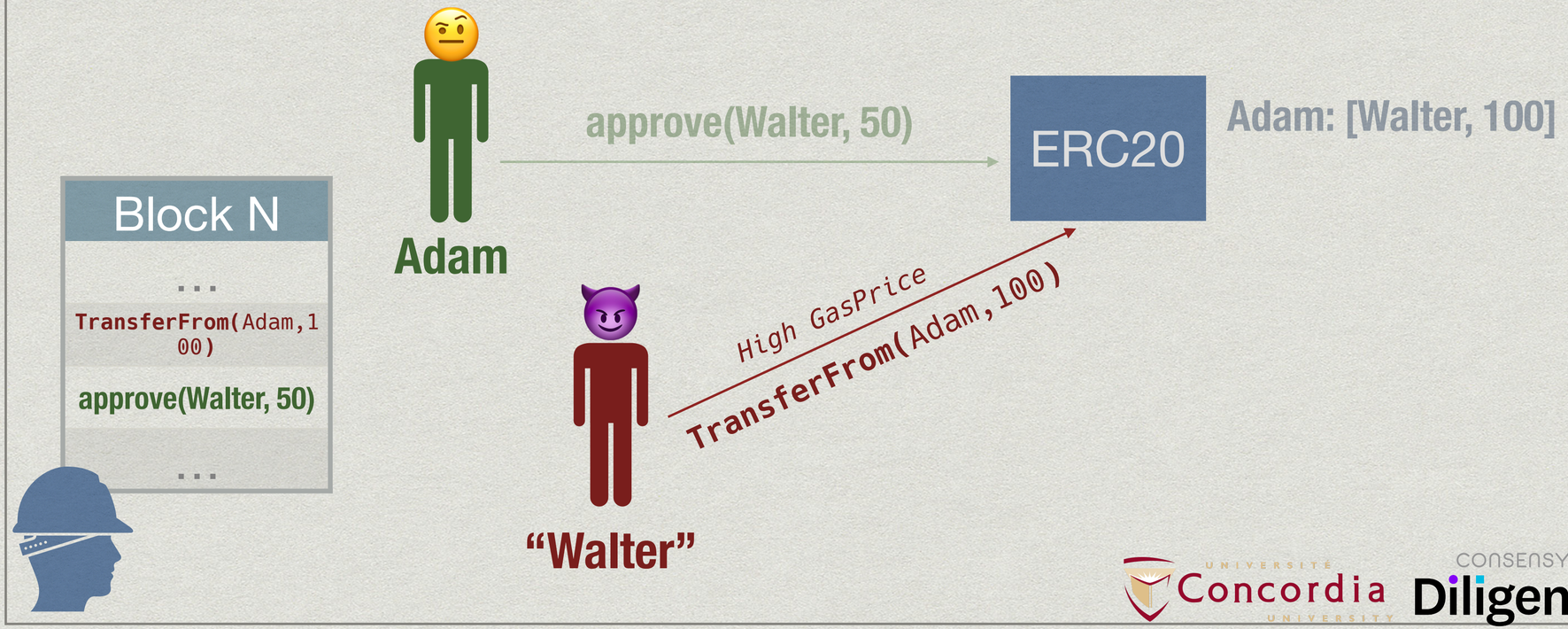
Design Practices (cont.)

- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



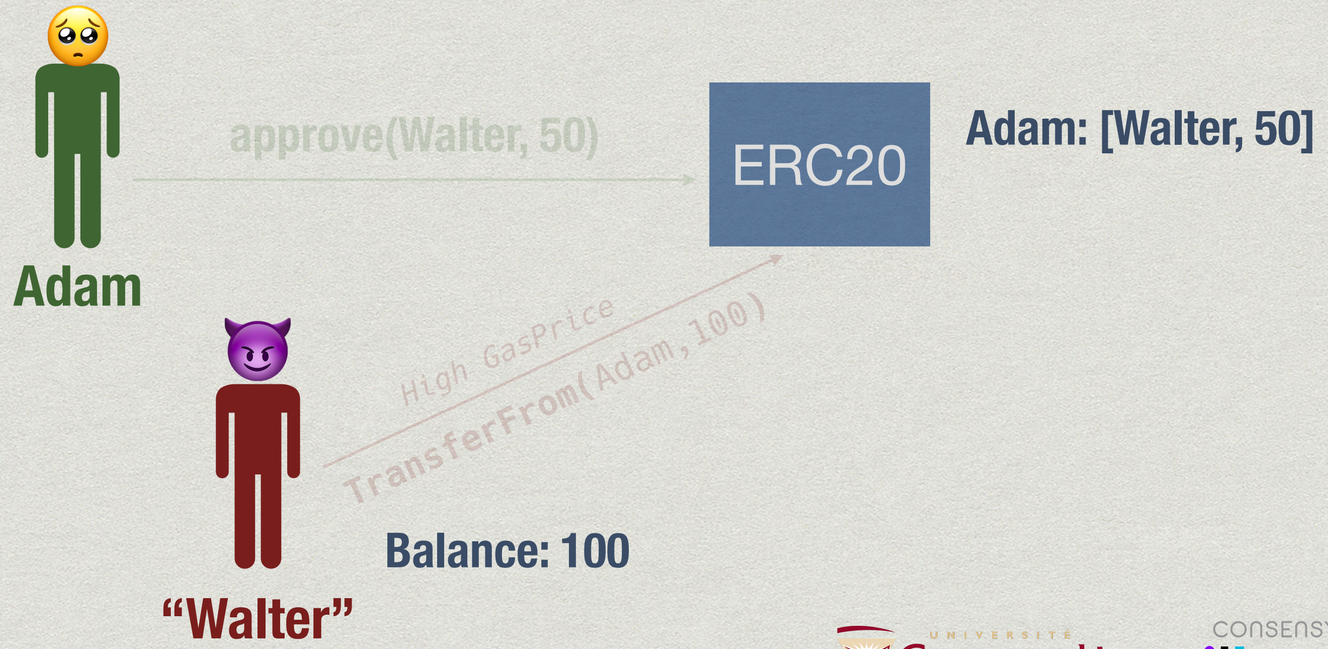
Design Practices (cont.)

- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



Design Practices (cont.)

- ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.
  - ***decreaseApproval()*** / ***increaseApproval()*** were proposed



Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run
Buy a domain name	Get the domain before her
Trade on a DEX	Sandwich her trade
Cancel a DEX transaction	Fill the order before it’s canceled
Buy a lottery ticket	Prevent her from buying the ticket

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run	Don’t care if Alice’s transaction runs after attacker’s
Buy a domain name	Get the domain before her	
Trade on a DEX	Sandwich her trade	
Cancel a DEX transaction	Fill the order before it’s canceled	
Buy a lottery ticket	Prevent her from buying the ticket	

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run	Don't care if Alice's transaction runs after attacker's
Buy a domain name	Get the domain before her	**Displacement**
Trade on a DEX	Sandwich her trade	
Cancel a DEX transaction	Fill the order before it's canceled	
Buy a lottery ticket	Prevent her from buying the ticket	

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run
Buy a domain name	Get the domain before her
Trade on a DEX	Sandwich her trade
Cancel a DEX transaction	Fill the order before it’s canceled
Buy a lottery ticket	Prevent her from buying the ticket

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run
Buy a domain name	Get the domain before her
Trade on a DEX	Sandwich her trade
Cancel a DEX transaction	Fill the order before it's canceled
Buy a lottery ticket	Prevent her from buying the ticket

Alice's transaction should run after attacker's

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

Alice's transaction should run after attacker's

Insertion

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run
Buy a domain name	Get the domain before her
Trade on a DEX	Sandwich her trade
Cancel a DEX transaction	Fill the order before it’s canceled
Buy a lottery ticket	Prevent her from buying the ticket

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

Run transaction and delay Alice's transaction

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

Run transaction and delay Alice's transaction

Suppression

Front-running on DApps

Alice sends a transaction	Transaction lives in mempool, All nodes can front-run
Buy a domain name	Get the domain before her
Trade on a DEX	Sandwich her trade
Cancel a DEX transaction	Fill the order before it’s canceled
Buy a lottery ticket	Prevent her from buying the ticket

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

- Don't care about Alice's transaction

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

- Don't care about Alice's transaction

Displacement

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before her
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

- Don't care about Alice's transaction
- Different transaction as Alice's transaction

Displacement

Front-running on DApps

Alice sends a transaction

- Buy a domain name
- Trade on a DEX
- Cancel a DEX transaction
- Buy a lottery ticket

Transaction lives in mempool, All nodes can front-run

- Get the domain before
- Sandwich her trade
- Fill the order before it's canceled
- Prevent her from buying the ticket

- Don't care about Alice's transaction
- Different transaction as Alice's transaction

Asymmetric

Displacement