Lesson 7

PolyNetwork Hack

Encoding the function signatures and parameters

Example

```
pragma solidity ^0.8.0;

contract MyContract {

  Foo otherContract;


  function callOtherContract() public view returns (bool){
      bool answer = otherContract.baz(69,true);
      return answer;
  }
}


contract Foo {
    function bar(bytes3[2] memory) public pure {}
    function baz(uint32 x, bool y) public pure returns (bool r)
{
    r = x > 32 || y;
    }
    function sam(bytes memory, bool, uint[] memory) public pure
{}
}
```

The way the call is actually made involves encoding the function selector and parameters

If we wanted to call ***baz*** with the parameters **69** and **true** , we would pass 68 bytes total, which can be broken down into:

1. the Method ID. This is derived as the first 4 bytes of the Keccak hash of the ASCII form of the signature baz(uint32,bool).

   ```
   ***0xcdcd77c0:***
   ```

2. the first parameter, a uint32 value 69 padded to 32 bytes

   0×0000000000000000000000000000000000000000000000000000000000000045:

3. the second parameter - boolean true, padded to 32 bytes

   0×0000000000000000000000000000000000000000000000000000000000000001:

In total

0xcdcd77c0000000000000000000000000000000000000000000

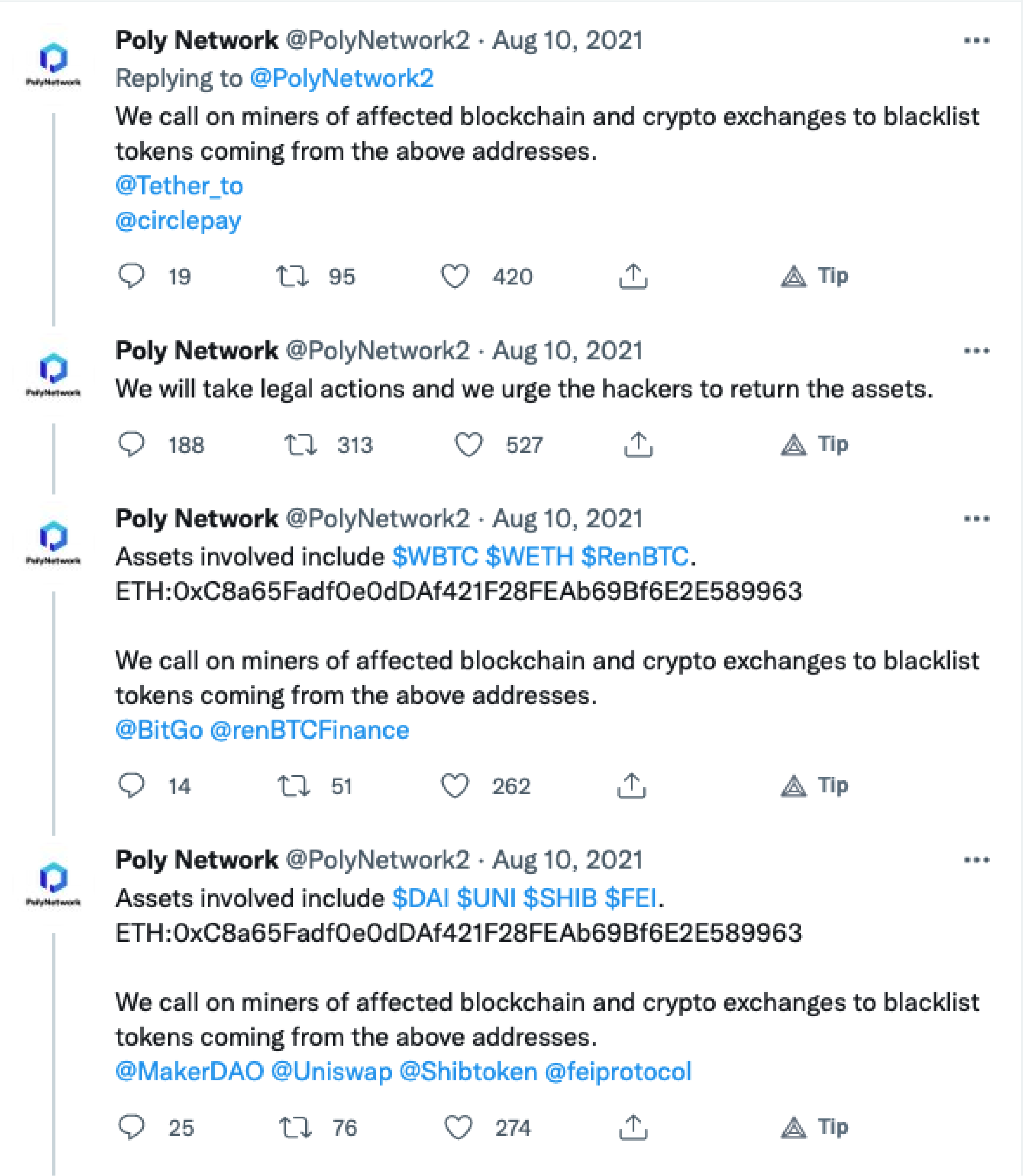
0000000000000000000004500000000000000000000000000000

000000000000000000000000000000000001

The Polynetwork Hack - $600M stolen

"LatestCourse/img/Screenshot 2022-07-28 at 10.06.23.png" is not created yet. Click to create.

Polynetwork react to attack



Poly Network @PolyNetwork2 · Aug 10, 2021

Replying to @PolyNetwork2

We call on miners of affected blockchain and crypto exchanges to blacklist tokens coming from the above addresses.
@Tether_to
@circlepay

19 95 420 Tip

Poly Network @PolyNetwork2 · Aug 10, 2021

We will take legal actions and we urge the hackers to return the assets.

188 313 527 Tip

Poly Network @PolyNetwork2 · Aug 10, 2021

Assets involved include $WBTC $WETH $RenBTC.
ETH:0xC8a65Fadf0e0dDAf421F28FEAb69Bf6E2E589963

We call on miners of affected blockchain and crypto exchanges to blacklist tokens coming from the above addresses.
@BitGo @renBTCFinance

14 51 262 Tip

Poly Network @PolyNetwork2 · Aug 10, 2021

Assets involved include $DAI $UNI $SHIB $FEI.
ETH:0xC8a65Fadf0e0dDAf421F28FEAb69Bf6E2E589963

We call on miners of affected blockchain and crypto exchanges to blacklist tokens coming from the above addresses.
@MakerDAO @Uniswap @Shibtoken @feiprotocol

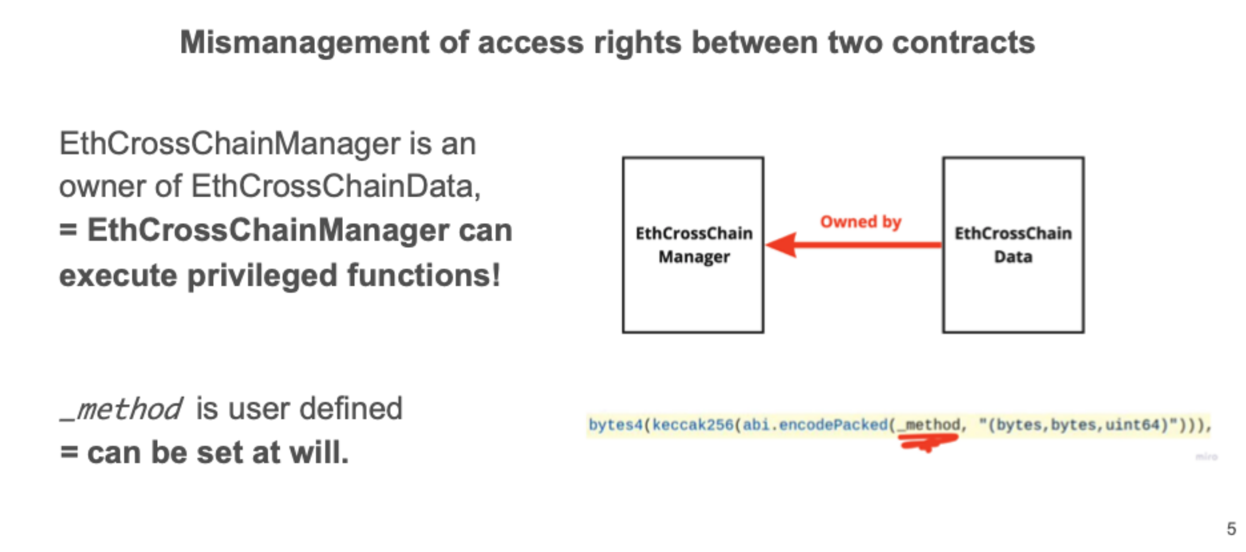
25 76 274 Tip

Example messages to the attacker

- Can I have some ETH? i been losing a lot of money Thank you
- Consider donating to public goods after all the pleasure you've got from using public infra 🌱

> - ngmi
> - Welcome to the crypto world. Fee Tips: Use tornado.cash to laundry your money ASAP.
> - Forget your USDT. Forget your USDC.
> - Swap tokens to ETH then deposit to tornado.cash . Good luck.
> - Hi. Boos. Can. You. Give me eth thanks
> - You can use Tornado for currency mixing
> - DONT USE YOUR USDT TOKEN YOU VE GOT BLACKLISTED
> - You can buy every pudgy penguin on opensea :)
> - Dad, this is my only asset. Please accept it
> - JOINING FOR EPIC SCREENCAP
> - Please gice me some eth
> - DONT USE YOUR USDT TOKEN
>   YOU VE GOT BLACKLISTED,god bless you
> - Remove liquidity from curve pool in form of DAI,
>   and exchange it for Eth and launder with Tornado
>
> ---

Exploit Details



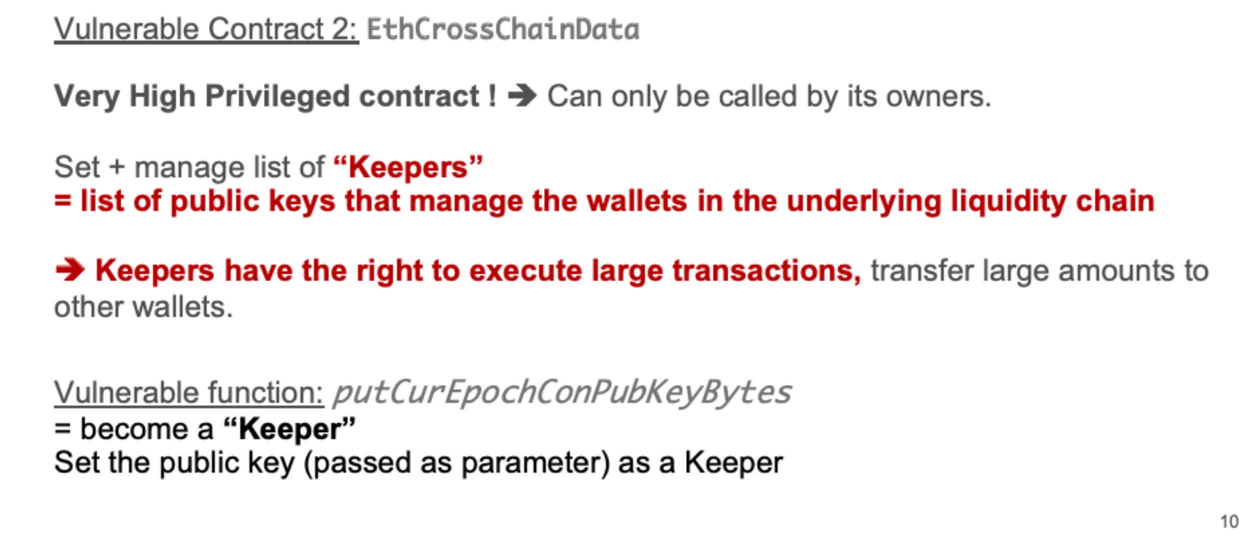
Mismanagement of access rights between two contracts

EthCrossChainManager is an owner of EthCrossChainData,
= EthCrossChainManager can execute privileged functions!

EthCrossChain Manager ← Owned by — EthCrossChain Data

_method is user defined
= can be set at will.

```
bytes4(keccak256(abi.encodePacked(_method, "(bytes,bytes,uint64)"))),
```



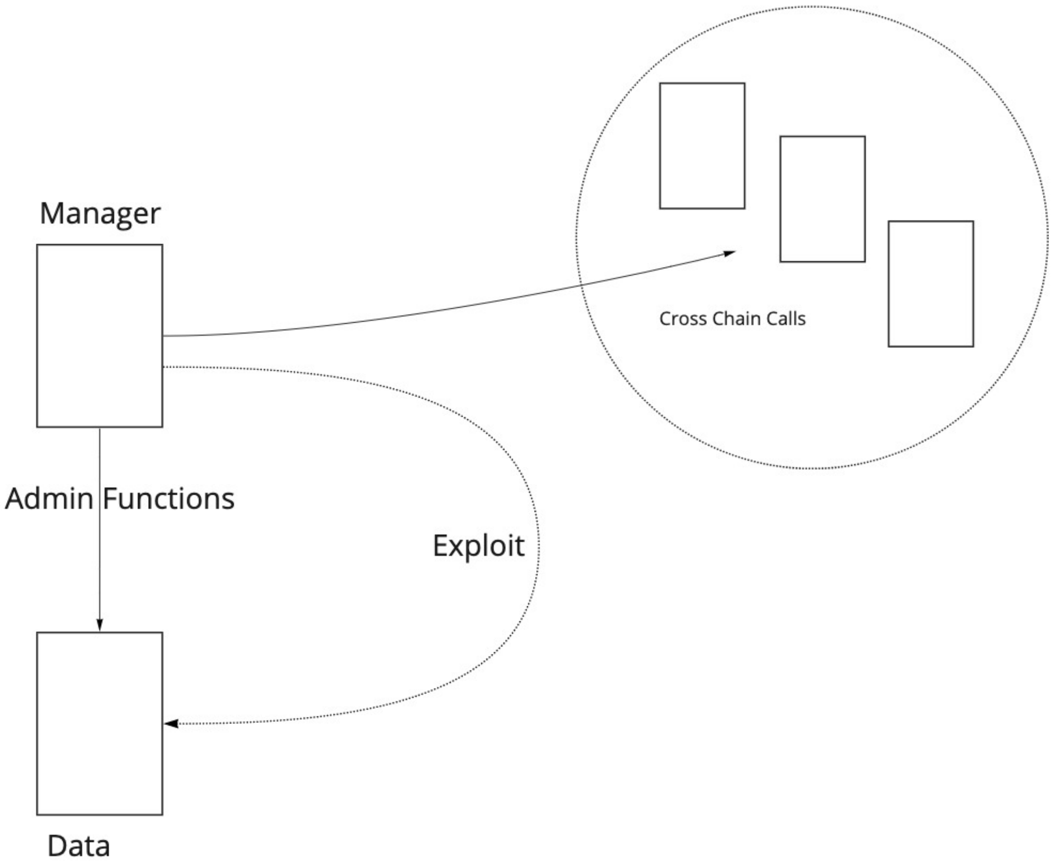
Vulnerable Contract 2: `EthCrossChainData`

Very High Privileged contract ! ➔ Can only be called by its owners.

Set + manage list of **"Keepers"**
= list of public keys that manage the wallets in the underlying liquidity chain

➔ **Keepers have the right to execute large transactions,** transfer large amounts to other wallets.

Vulnerable function: *putCurEpochConPubKeyBytes*
= become a **"Keeper"**
Set the public key (passed as parameter) as a Keeper


Manager
Cross Chain Calls
Admin Functions
Exploit
Data

> The attacker could change the contract address, the tricky part was to make the method ID in the call match the target
>
> The target is
> Hash(putCurEpochConPubKeyBytes,bytes) = 0×41973cd9
>
> The manager contract will create a call using a method ID calculated as Hash(X,bytes,bytes,uint64)
>
> where X can be supplied by the user calling the contract.
>
> So
>
> The attacker was aiming to find X such that
>
> Hash(X,bytes,bytes,uint64)
>
> =
>
> Hash(putCurEpochConPubKeyBytes,bytes) = 0×41973cd9
>
> The Brute Force solution was
>
> X = f1121318093
>
> ---

The Attacker reconsiders

The attacker received a rather cryptic message
” Dont instant tornado funds, dont instant move blacklistable tokens to DAI/ETH? Insider confirmed!”

The attacker was looking at Tornado Cash and sent themselves a message

”Wonder why Tornado? Will miners stop me? Teach me please”

Then someone found a link between an address used by the attacker and some exchanges and tweeted

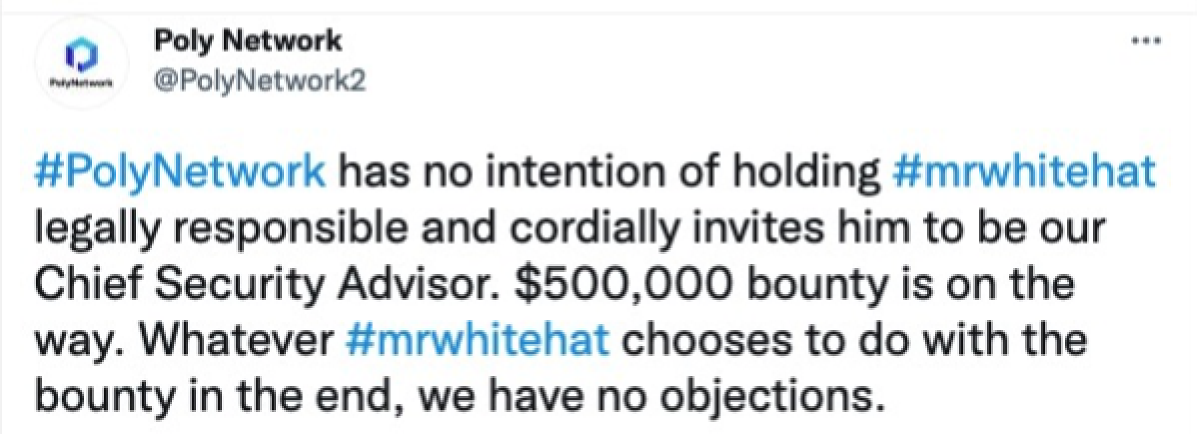
“Did the PolyNetwork Exploiter accidentally use the wrong sender address for this tx 0xb12681d9e? The sender address is tied to FTX, Binance, Okex accounts.”

The hacker’s attitude started to change, he suggested he could return "some tokens" or even abandon them, saying that they were "not so interested in the money".

Followed by a suggestion : “What if I make a new token and let the DAO decide where the tokens go”

Finally the attacker messaged ”Ready to return the fund !”

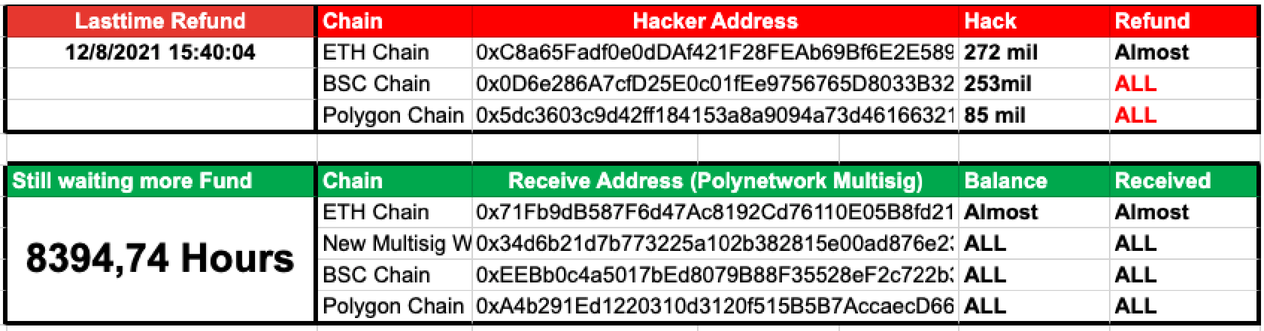
See this spreadsheet for all communications
The attacker starts to return the funds
Polynetwork starts to refer to the attacker as ‘Mr White Hat’ and offer him a job and bounty



Poly Network
@PolyNetwork2

#PolyNetwork has no intention of holding #mrwhitehat legally responsible and cordially invites him to be our Chief Security Advisor. $500,000 bounty is on the way. Whatever #mrwhitehat chooses to do with the bounty in the end, we have no objections.

'We appreciate you sharing your experience and believe your action constitutes white hat behavior. But we can't touch user assets and Poly Network doesn't have its own token. Since , we believe your action is white hat behavior, we plan to offer you a $500,000 bug bounty after you complete the refund fully. Also we assure you that you will not be accountable for this incident.
We hope that you can return all tokens as soon as possible. You can reserve the equivalent value of 500,000 USD in any assets to the current owner address. We will make up this part of the assets to Poly Network users.
Your contribution is very helpful to us. Again, we think this behavior is white hat behavior, therefor this 500,000 USD will be seen as completely legal bounty reward. We will also ensure that you will not be held accountable for this incident, and we will publicly express our gratitude to you.'



Lasttime Refund	Chain	Hacker Address	Hack	Refund
12/8/2021 15:40:04	ETH Chain	0xC8a65Fadf0e0dDAf421F28FEAb69Bf6E2E589	272 mil	Almost
	BSC Chain	0x0D6e286A7cfD25E0c01fEe9756765D8033B32	253mil	ALL
	Polygon Chain	0x5dc3603c9d42ff184153a8a9094a73d46166321	85 mil	ALL

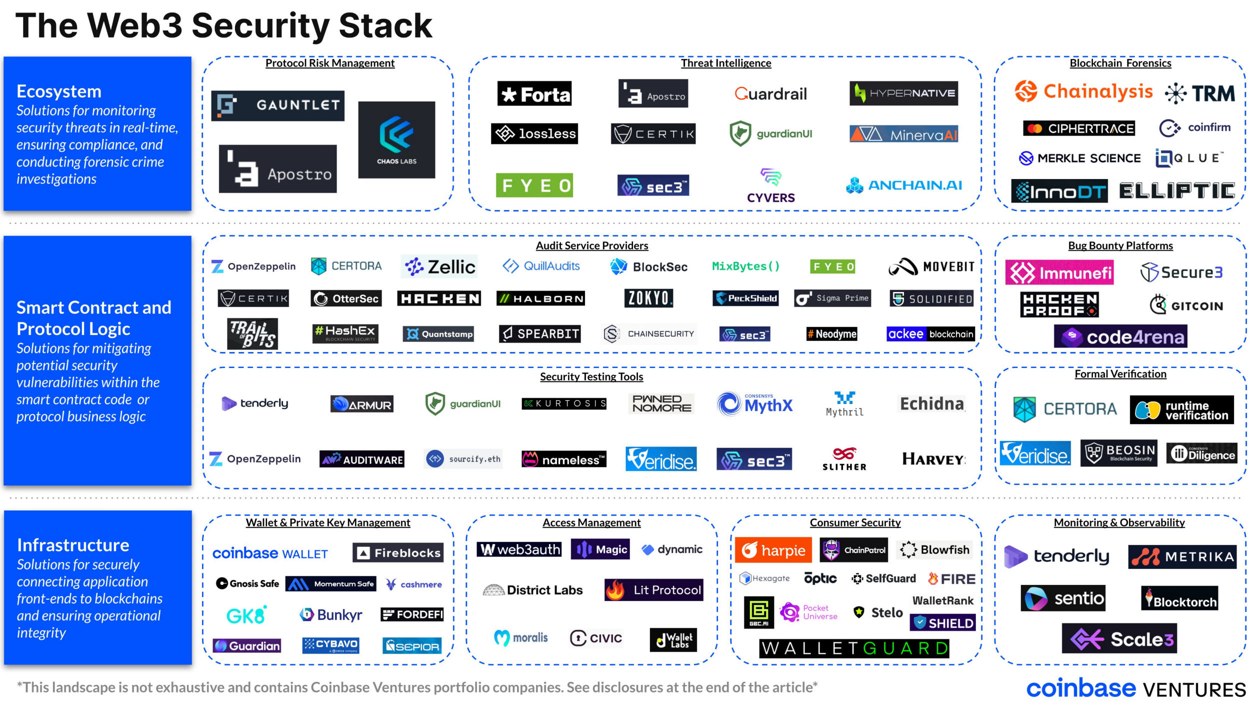
Still waiting more Fund	Chain	Receive Address (Polynetwork Multisig)	Balance	Received
8394,74 Hours	ETH Chain	0x71Fb9dB587F6d47Ac8192Cd76110E05B8fd21	Almost	Almost
	New Multisig W	0x34d6b21d7b773225a102b382815e00ad876e2	ALL	ALL
	BSC Chain	0xEEBb0c4a5017bEd8079B88F35528eF2c722b	ALL	ALL
	Polygon Chain	0xA4b291Ed1220310d3120f515B5B7AccaecD66	ALL	ALL

A number of rather lengthy messages follow the return of the funds

```
Q & A, PART TWO:

Q: WHAT REALLY HAPPENED 30 HOURS AGO?
A: LONG STORY.

BELIEVE IT OR NOT, I WAS _FORCED_ TO PLAY THE GAME.

THE POLY NETWORK IS A SOPHISTICATED SYSTEM, I DIDN'T MANAGE TO BUILD A LOCAL TESTING ENVIRONMENT. I FAILED TO
PRODUCE A POC AT THE BEGINNING. HOWEVER, THE AHA MOMEMNT CAME JUST BEFORE I WAS TO GIVE UP. AFTER DEBUGGING ALL
NIGHT, I CRAFTED A _SINGLE_ MESSAGE TO THE ONTOLOGY NETWORK.

I WAS PLANNING TO LAUNCH A COOL BLITZKRIEG TO TAKE OVER THE FOUR NETWORK: ETH, BSC, POLYGON & HECO. HOWEVER THE HECO
NETWORK GOES WRONG! THE RELAYER DOES NOT BEHAVE LIKE THE OTHERS, A KEEPER JUST RELAYED MY EXPLOIT DIRECTLY, AND THE
KEY WAS UPDATED TO SOME WRONG PARAMETERS. IT RUINED MY PLAN.

I SHOULD HAVE STOPPED AT THAT MOMENT, BUT I DECIDED TO LET THE SHOW GO ON! WHAT IF THEY PATCH THE BUG SECRETLY
WITHOUT ANY NOTIFICATION?

HOWEVER, I DIDN'T WANT TO CAUSE _REAL_ PANIC OF THE CRYPTO WORLD. SO I CHOSE TO IGNORE SHIT COINS, SO PEOPLE DIDN'T
HAVE TO WORRY ABOUT THEM GOING TO ZERO. I TOOK IMPORTANT TOKENS (EXCEPT FOR SHIB) AND DIDN'T SELL ANY OF THEM.

Q: THEN WHY SELLING/SWAPPING THE STABLES?
A: I WAS PISSED BY THE POLY TEAM FOR THEIR INITIAL REPONSE.

THEY URGED OTHERS TO BLAME & HATE ME BEFORE I HAD ANY CHANCE TO REPLY! OF COURSE I KNEW THERE ARE FAKE DEFI COINS,
BUT I DIDN'T TAKE IT SERIOUSLY SINCE I HAD NO PLAN LAUNDERING THEM.

IN THE MEANWHILE, DEPOSITING THE STABLES COULD EARN SOME INTEREST TO COVER POTENTIAL COST SO THAT I HAVE MORE TIME
TO NEGOTIATE WITH THE POLY TEAM.
```

Security Best Practices

Ecosystem



The Web3 Security Stack

Ecosystem
Solutions for monitoring security threats in real-time, ensuring compliance, and conducting forensic crime investigations

Smart Contract and Protocol Logic
Solutions for mitigating potential security vulnerabilities within the smart contract code or protocol business logic

Infrastructure
Solutions for securely connecting application front-ends to blockchains and ensuring operational integrity

This landscape is not exhaustive and contains Coinbase Ventures portfolio companies. See disclosures at the end of the article

Consensys Best Practices

General

- Prepare for Failure
- Stay up to Date
- Keep it Simple
- Rolling out
- Blockchain Properties
- Simplicity vs. Complexity

Precautions

- General
- Upgradeability
- Circuit Breakers
- Speed Bumps
- Rate Limiting
- Deployment

Solidity Specific

Token Specific

Documentation

> ## Attacks
>
> - Reentrancy
> - Oracle Manipulation
> - Frontrunning
> - Timestamp Dependence
> - Insecure Arithmetic
> - Denial of Service
> - Griefing
> - Force Feeding