

Following Poly Network Attack: Beosin KYT/AML Keeps Tracing Stolen Funds and Unveils Hackers' Tactic



According to Beosin EagleEye security risk monitoring platform, on July 2, the Poly Network cross-chain bridge was suspected of being attacked due to a potential compromise of private keys or a multi-signature service attack. The hacker has exploited forged proofs to initiate withdrawal operations on the cross-chain bridge contracts across multiple chains.

It is not the first time that Poly Network got attacked. As early as August 10, 2021, Beosin EagleEye showed that Poly Network was attacked and nearly 600 million dollars in funds were stolen from the three chains of Ethereum, BSC, and Polygon.

This incident of the year also became the security incident with the largest amount of loss in 2021. Two years ago, the attacker used the logic flaw in the EthCrossChainManager contract to call the putCurEpochConPubKeyBytes function in the EthCrossChainData contract to change the Keeper to the hacker address and then used the address to sign the transaction of withdrawing tokens, thereby withdrawing a large number of tokens in the LockProxy contract. However, under the pressure of many parties, the attacker finally chose to return the stolen assets.

This time, how did the hacker attack Poly Network and deal with the stolen funds? The details are as follows.

Attack Analysis

Take the attack address 0x906639ab20d12a95a8bec294758955870d0bb5cc as an example:

Firstly, the attacker called the lock function on the LockProxy cross-chain bridge contract to lock a small amount of Lever Token.

From:0x906639ab20d12a95A8bec294758955870d0BB5CC

Interacted With (To):0x250e76987d838a75310c34bf422ea9f1AC4Cc906

ERC-20 Tokens Transferred:From 0x906639...0d0BB5CC To 0x250e76...AC4Cc906 For 0.0000000000000001 (\$0.00) Lever Token... (LEV...)

Value:0 ETH (\$0.00)

Transaction Fee:0.003060889868875704 ETH \$5.96

Gas Price:22.215132881 Gwei (0.000000022215132881 ETH)

Ether Price:\$1,924.30 / ETH

Gas Limit & Usage by Txn:206,676 | 137,784 (66.67%)

Gas Fees:Base: 17.11761545 Gwei

Burnt Fees:Burnt: 0.0023585335271628 ETH (\$4.59)

Other Attributes:Txn Type: 0 (Legacy) Nonce: 4 Position In Block: 135

Input Data:

#	Name	Type	Data
0	fromAssetHash	address	0xbc194e6f748a222754c3B8b9946922c09E7d4e91
1	toChainId	uint64	6
2	toAddress	bytes	0xe0afada1d93704761e9550f21a53de3468ba599
3	amount	uint256	1000

Switch Back

The toChainId 6 corresponds to the BNB chain, which can be viewed at <https://explorer.poly.network>. If a transaction is visible on the Poly Network explorer, it indicates that it has been validated through the relay chain.

Transaction

```
Hash 1b8f8a38895ce8375308c570c7511d16a2ba972577747b0ac7ace5cc59bbb1c4
```

Time: Jul-01-2023 17:55:47 UTC

Type: **cross chain transfer**

Duration: In Progress

Status: In Progress

Transfer Detail:

Asset: LEV

Amount: 1e,-15

From Chain: **Ethereum**To Chain: **BNB Chain**

Switching to the BNB chain, the attacker used the `verifyHeaderAndExecuteTx` function to initiate withdrawal operations, but the quantity involved does not match the original lock amount.

[illegible]

However, upon querying the relay chain network, no record of this transaction was found.

Transaction

Hash 5c70178e6dc882fba1663400c9566423f8942877a0d42bb5c982c95acc348e31

Time:

Type:

Duration:

Status:

Transfer Detail:

Asset:

Amount:

From Chain:

From Address:

To Chain:

To Address:

There is now reason to suspect that either the signatures have been leaked or the keepers have been modified.

Keepers are responsible for signing user withdrawals, so controlling a keeper would allow the attacker to initiate withdrawals with forged signatures. The attack on Poly Network in 2021 was caused by hackers using the attack to modify the keeper.

```
1448 * @param pubkeyList Poly chain consensus nodes public key list
1449 * @param sigList Poly chain consensus nodes signature list
1450 * @return true or false
1451 */
1452 function changeBookKeeper(bytes memory rawHeader, bytes memory pubkeyList, bytes memory sigList) whenNotPaused public returns(bool) {
1453     // Load Ethereum cross chain data contract
1454     ECCUtils.Header memory header = ECCUtils.deserializeHeader(rawHeader);
1455     IETHCrossChainData eccd = IETHCrossChainData(ethCrossChainDataAddress);
1456
1457     // Make sure rawHeader.height is higher than recorded current epoch start height
1458     uint64 curEpochStartHeight = eccd.getCurEpochStartHeight();
1459     require(header.height > curEpochStartHeight, "The height of header is lower than current epoch start height!");
1460
1461     // Ensure the rawHeader is the key header including info of switching consensus peers by containing non-empty nextBookKeeper field
1462     require(header.nextBookKeeper != bytes20(0), "The nextBookKeeper of header is empty");
1463
1464     // Verify signature of rawHeader comes from pubkeyList
1465     address[] memory polychainBks = ECCUtils.deserializeKeepers(eccd.getCurEpochConPubKeyBytes());
1466     uint n = polychainBks.length;
1467     require(ECCUtils.verifySig(rawHeader, sigList, polychainBks, n - (n - 1) / 3), "Verify signature failed!");
1468
1469     // Convert pubkeyList into ethereum address format and make sure the compound address from the converted ethereum addresses
1470     // equals passed in header.nextBookKeeper
1471     (bytes20 nextBookKeeper, address[] memory keepers) = ECCUtils.verifyPubkey(pubkeyList);
1472     require(header.nextBookKeeper == nextBookKeeper, "Nextbookkeepers illegal");
1473
1474     // update current epoch start height of Poly chain and current epoch consensus peers book keepers addresses
1475     require(eccd.putCurEpochStartHeight(header.height), "Save MC LatestHeight to Data contract failed!");
1476     require(eccd.putCurEpochConPubKeyBytes(ECCUtils.serializeKeepers(keepers)), "Save Poly chain book keepers bytes to data contract failed!");
1477
1478     // Fire the change book keeper event
1479     emit ChangeBookKeeperEvent(header.height, rawHeader);
1480     return true;
1481 }
1482
```

output: [0: "0x3dfc0b7b8a6972cde3b696d3c0c032514b0f3826", 1: "0x4c46e1f946362547546677bfa719598385ce56f2", 2: "0xf81e768932f6dfec4a5d0671bd2715642fcef98", 3: "0x51b7529137d34002c4ebd81a2244f0ee7e95b2c0"]

Upon analyzing the attacker's use of the verifyHeaderAndExecuteTx function for withdrawal operations, it was found that the keepers have not been modified.

Now, there is reason to believe that three keepers (0x4c46e1f946362547546677bfa719598385ce56f2, 0x51b7529137d34002c4ebd81a2244f0ee7e95b2c0, 0x3dfccb7b8a6972cde3b695d3c0c032514b0f3825) may have suffered private key compromise or a multi-signature service attack.



At the same time, according to Beosin KYT/AML, the hackers called the contract of Poly Network through a batch of addresses and used the loopholes in the two functions of UnlockEvent and verifyHeaderAndExecuteTx in the contract to attack the project party and transfer funds to your deposit address.

The fund flow

On the Ethereum

The addresses calling the attacked contract obtained gas fees through a common address.

0xf64d5843d96634dfe...	Transfer	17144348	2023-04-28 11:32:59	0x0dFeb4...8Cb9701C	OUT	0x58c999...953e5201	0.2 ETH	0.0007039
0x3d823911011f0f99e...	Transfer	17144155	2023-04-28 10:53:23	0x0dFeb4...8Cb9701C	OUT	0x080C23...48575f6E	0.2 ETH	0.00068272
0xea19dae4f164539bd...	Transfer	17144118	2023-04-28 10:45:59	0x0dFeb4...8Cb9701C	OUT	0x8c43d3...797d8968	0.2 ETH	0.00103371
0x9672ab880619f163...	Transfer	17143814	2023-04-28 9:44:35	0x0dFeb4...8Cb9701C	OUT	0xB03BAE...f85bFe9A	0.2 ETH	0.00069639
0xc18eec9ad15d3d62...	Transfer	17143711	2023-04-28 9:23:35	0x0dFeb4...8Cb9701C	OUT	0x0Cb079...2c337905	0.2 ETH	0.00112659
0xf6761e7ac4a644090...	Transfer	17143644	2023-04-28 9:10:11	0x0dFeb4...8Cb9701C	OUT	0x1c1B69...5231081A	0.2 ETH	0.00069091
0x60ca19097c27252b...	Transfer	17143605	2023-04-28 9:02:23	0x0dFeb4...8Cb9701C	OUT	0x29A232...f2E3353B	0.2 ETH	0.00074891
0x6ac8fa4a853933cdf...	Transfer	17143549	2023-04-28 8:51:11	0x0dFeb4...8Cb9701C	OUT	0xa59118...2e5fB049	0.2 ETH	0.00066676
0x7c54d6adc8fa8f916...	Transfer	17143518	2023-04-28 8:44:59	0x0dFeb4...8Cb9701C	SELF	0x0dFeb4...8Cb9701C	0 ETH	0.00076047
0xf47a1fc85eabaf408...	Transfer	17143518	2023-04-28 8:44:59	0x0dFeb4...8Cb9701C	OUT	0x906639...0d0B85CC	0.2 ETH	0.00076026
0x2e58724fb6beccf88...	Transfer	17143372	2023-04-28 8:15:11	0x0dFeb4...8Cb9701C	OUT	0xa052A3...A9a0dDf1	0.2 ETH	0.00066785
0xccb5c63af8c3a42a8...	Transfer	17143290	2023-04-28 7:58:23	0x0dFeb4...8Cb9701C	OUT	0x879ab1...D8180261	0.2 ETH	0.0007663
0x69923ea025e76845...	Transfer	17143252	2023-04-28 7:50:47	FixedFloat	IN	0x0dFeb4...8Cb9701C	6.7569414 ETH	0.000735

Transactions to distribute fees

The address(0x0dfb429166e629204aca66467484cd88cb9701c) to distribute fees obtained the fee through an exchange called FixedFloat.

There are three sources of fees in total for the hackers' consolidated address.

1. A fee comes from Tornado.Cash

0x97e0cd3da10be440...	17599609	2023-07-01 13:50:23	Tornado.Cash: 1 ETH	PolyNetwork Exploiter 3	0.98437579 ETH
-----------------------	----------	---------------------	---------------------	-------------------------	----------------

The fee for the consolidated address 0xe0afadad1d93704761c8550f21a53de3468ba599

2. Through Bybit, it flows into the address through a layer of transit.

The address to transfer fees: 0x4FbC021742A4664D1cf8e9d2730b8519B9Dcc523
transaction

hash0xb8b0626b86ed336c9c0fff56b20761438535aa06461dcca9cdc39dc10ec1c620

3. Use the stolen assets to swap for ETH as fees

There are two hacker addresses that swap the stolen USDT/USDC into ETH in DEX, and then use it as fees for subsequent addresses. The following are the initial fee transfer and hashes of the two addresses.

Hacker address 1: 0xddddE20a5F569DFB11F5c405751367E939ebC5886

Fee transfer address: 0xD475747a4937a66Cc7D4a2c7eA7F6e827D0f7390

Transaction hash:

0x853b75b1b8a7f56c51fcb9b996af8d132b784cfa0da7162c20a48a5994d8a06

Hacker address 2: 0x8E0001966e6997db3e45c5F75D4C89a610255b2E

Transaction hash:

0x0f3cf1fe16052223e091e87c2a6f7a9a94e53a565dfac7b83eb0b9b79458ad8f

On the BSC chain

0x0b0aa0d438e4f15c91...	Transfer	27513038	73 days 6 hrs ago	0x635308e731a878741b...	IN	0x1634bf68e6b3bb8d79...	6.6796 BNB	0.000063
0xa724cc4ef5cd2a5db24...	Transfer	27505953	73 days 12 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0x3def2aeee007d5ee5c...	0.5 BNB	0.000063
0x8b4b2b600db7a73a0a...	Transfer	27501185	73 days 16 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0xe8abb679704397c1e4...	0.5 BNB	0.001176
0xeff2d1bc44db7171e7...	Transfer	27499813	73 days 17 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0xfe295bf0b7da255c1dc...	0.5 BNB	0.000063
0x44b49bd2f9c3065765...	Transfer	27477844	74 days 12 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0xbff33ba0fd3fc718e132...	0.3 BNB	0.000063
0x2c434d8efd70b1e443...	Transfer	27472235	74 days 16 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0x3c18c88170056520e2...	0.3 BNB	0.000063
0x84501f82af4d0bd5b...	Transfer	27470970	74 days 17 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0x8e0001966e6997db3e...	0.3 BNB	0.000063
0x307f81487aa3a450a8...	Transfer	27444972	75 days 15 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0xe8a374c8f361b42bd3...	0.2 BNB	0.000063
0x3dadd5a53ad3569a2d...	Transfer	27444828	75 days 15 hrs ago	0x1634bf68e6b3bb8d79...	OUT	0x6a58b1a947fc4b84e5...	2 BNB	0.000063
0x6db6c128960b7268f2...	Transfer	27128643	86 days 17 hrs ago	0x975d9bd9928f398c7e...	IN	0x1634bf68e6b3bb8d79...	5.93290184 BNB	0.00021

The address(0x1634Bf68e6b3Bb8D79388EfB3d1A5215506FBbEd) to distribute fees

The address to distribute fees obtained fees and distributes them through Kucoin and ChangeNow exchanges.

1. Kucoin

Transaction hash:

0x0b0aa0d438e4f15c919e55148c87890ae0d089d036cadbdc6b87afa9e19f747b

2、ChangeNow

Transaction hash:

0x6db6c128960b7268f2bf8c199b2c0c017b3bee29bbefac0bf5d31c63b6373075

There is no fee for the consolidated addresses.

On the Polygon chain



0xc6857adf33758b3b77...	Transfer	44518708	2 days 4 hrs ago	0x09f92edce2e46c399bf...	OUT	0x55aba51912e14dde79...	15 MATIC	0.011550559693
0x0aad0ab472d7deaf29...	Transfer	44459701	3 days 16 hrs ago	0x09f92edce2e46c399bf...	OUT	0x9450c9131e4ca94672...	250 MATIC	0.003038403646
0xc7a25eb840718028c0...	Transfer	44456280	3 days 18 hrs ago	FixedFloat	IN	0x09f92edce2e46c399bf...	374,484842 MATIC	0.0038501

The address(0x09F92eDce2E46C399BFE7881a7619598AF8436d5) to distribute fees

The address calling the attacked contract obtained fees through a common address.

This address gets the transaction fee through the FixedFloat exchange.

Transaction hash:

0xc7a25eb840718028c0d8f402d1293dcb479755d77609a7dfb616c10e90176dec

The source of the hacker's consolidated addresses' fees is

0x09F92eDce2E46C399BFE7881a7619598AF8436d5.

Beosin KYT/AML keeps tracing the flow of funds

On the Ethereum

Beosin KYT/AML tracked and found that the stolen funds on the ETH chain are as follows:

The fees of these 20 addresses all came from the address 0x0dfeb429166e629204aca66467484cd88cb9701c, and the fees of this address were transferred in through Fixedfloat.

The hacker called LOCK in the Poly Network contract, locked the funds, and then called the two functions of UnlockEvent and verifyHeaderAndExecuteTx to attack the project. The case is as follows:

123	<div> <div>Address</div> <div>0x250e76987d838a75310c34bf422ea9f1ac4cc906</div> <div>🔍</div> </div>
	<div> <div>Name</div> <div>UnlockEvent (address toAssetHash, address toAddress, uint256 amount) View Source</div> </div>
	<div> <div>Topics</div> <div> <div>0</div> <div>0xd90288730b87c2b8e0c45bd8226fd22478aba30ae1c4d578bdaba9261604df</div> </div> </div>
	<div> <div>Data</div> <div> <div>toAssetHash: 0x00</div> <div>toAddress: 0xe0Afadad1d93704761c8550F21A53DE3468Ba599</div> <div>amount: 1592518181684320000000</div> </div> <div>DecHex</div> </div>

It can be seen that in the UnlockEvent, the variable toAddress has become the hacker consolidated address, and the amount has also been modified to the amount of stolen funds (1,592.51818168432 ETH here).

[illegible]

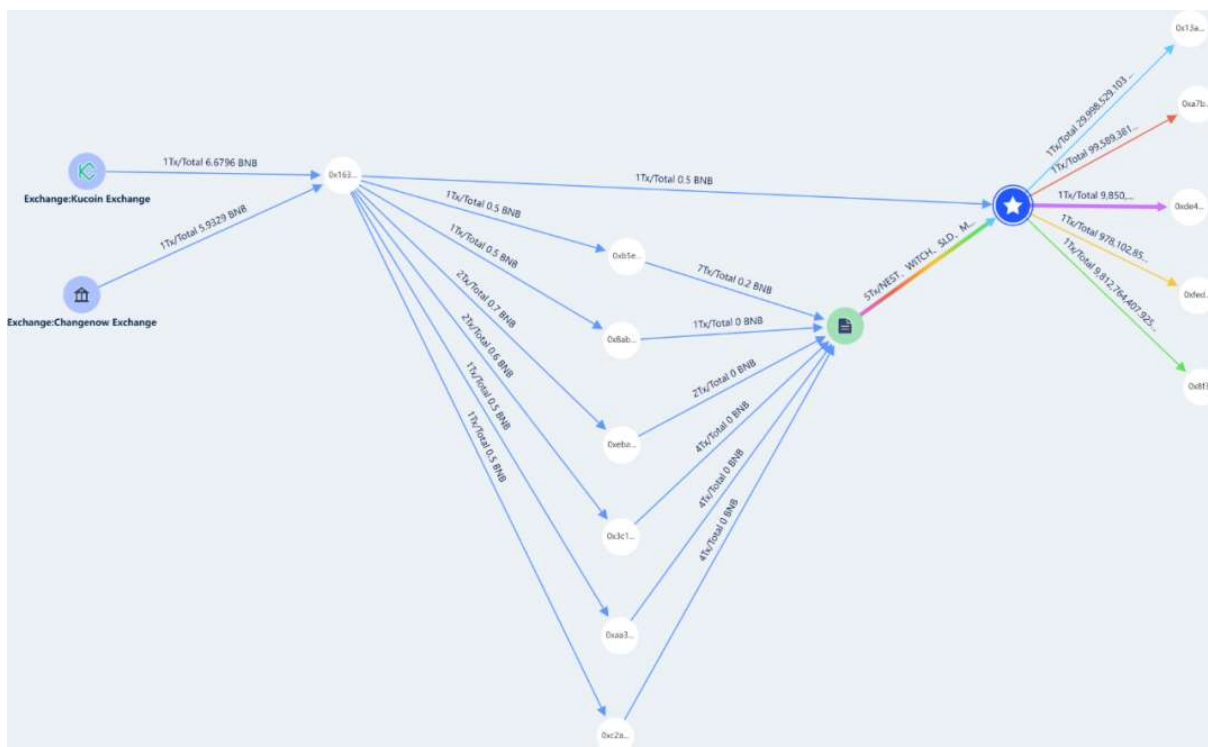
From here, we can see that the Proof item in the input data has been replaced with the content containing the hacker consolidated address. The fee of the hacker's address involved in the case is mainly obtained from five ways:

1. Transfer to ETH through Tronado.cash
2. Transfer to ETH through Bybit exchange
3. Transfer to ETH through KuCoin Exchange
4. Transfer to ETH through the FixedFloat exchange
5. Using stolen ETH

Hackers began to attack on July 1, 2023. Up to now, only some virtual currencies have been exchanged for ETH through Dex and ETH and some other virtual currencies have been transferred to other addresses.

On the BSC chain

The flow on the BSC chain is similar to that on the Ethereum. The hacker continued to use some of the same addresses for the stealing operation, and used the contract bugs to transfer the assets to the hacker's addresses.



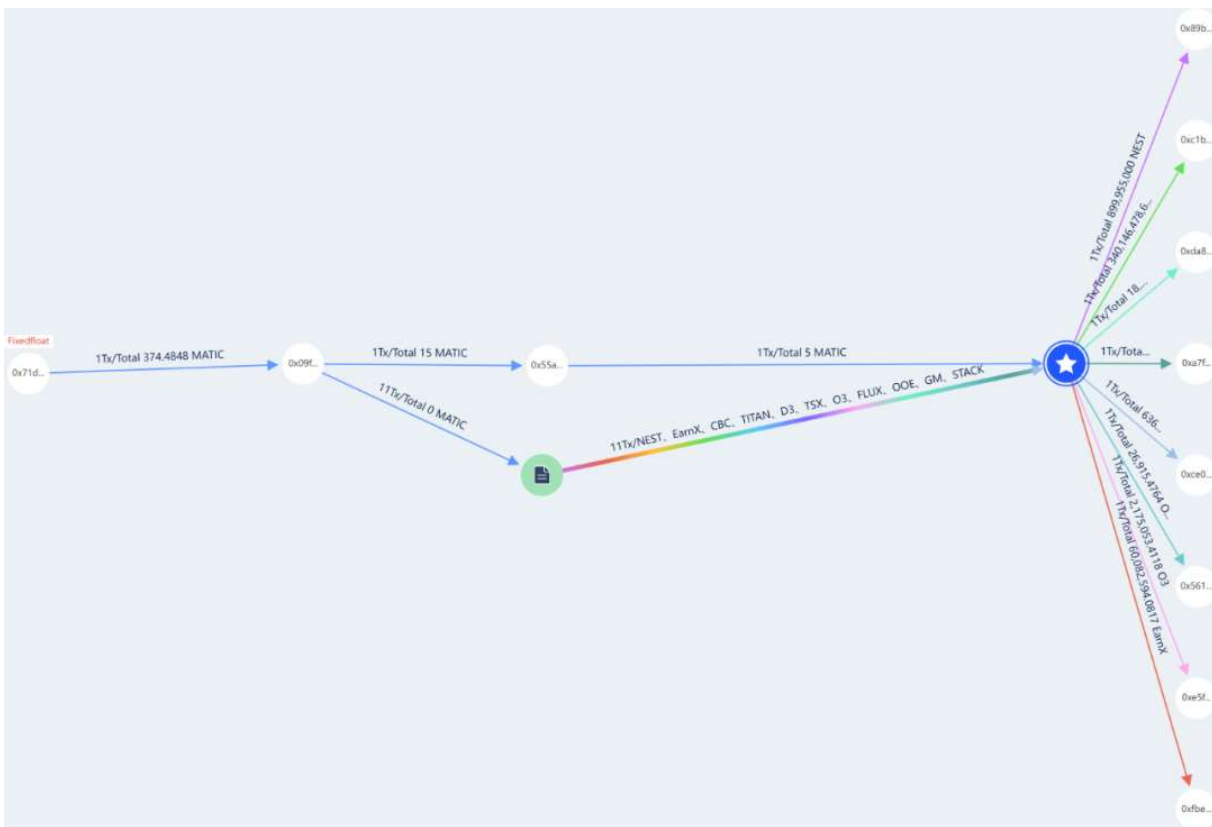
First, the hacker called the attacked contract through a batch of addresses to exploit the vulnerability, using more than 30 addresses.

The fees of these addresses all came from the address 0x1634Bf68e6b3Bb8D79388EfB3d1A5215506FBbEd and the fee of this address was transferred through the Kucoin and ChangeNow platforms.

Then, by using the same vulnerability to attack the contract, the stolen funds were transferred to the hacker address, and then part of the funds were transferred to multiple consolidated addresses.

On the Polygon chain

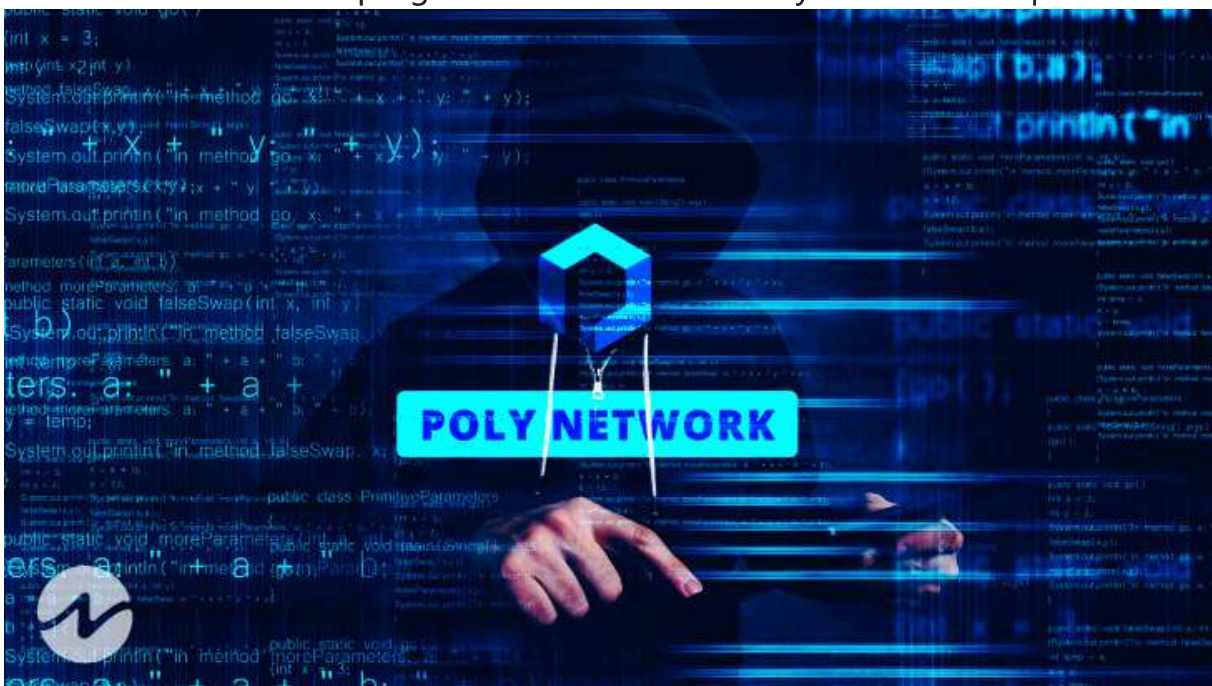
The flow on the Polygon is similar to that on the Ethereum and BSC. Hackers continued to use some of the same addresses to carry out the stealing operations, and used contract bugs to transfer assets to the hacker's addresses.



First, the hacker called the attacked contract through a batch of addresses to exploit the vulnerability. The hacker only used one address 0x09F92eDce2E46C399BFE7881a7619598AF8436d5 and the fee was transferred through Fixedfloat.

Then, by using the same vulnerability to attack the contract, the stolen funds were transferred to the collection address, and then part of the funds were transferred to multiple consolidated addresses.

Currently, the Beosin security team is working with Poly Network to deal with this security incident and the latest progress will be shared with you as soon as possible.



Beosin is a leading global blockchain security company co-founded by several professors from world-renowned universities and there are 40+ PhDs in the team. It has offices in Singapore, Korea, Japan, and other 10+ countries. With the mission of "Securing Blockchain Ecosystem", Beosin provides "All-in-one" blockchain security solution covering Smart Contract Audit, Risk Monitoring & Alert, KYT/AML, and Crypto Tracing. Beosin has already audited more than 3000 smart contracts and protected more than \$500 billion funds of our clients.

Contact

If you need any blockchain security services, welcome to contact us:

[Official Website](#) [Beosin](#) [EagleEye](#) [Twitter](#) [Telegram](#) [Login](#) | [InCareer](#)