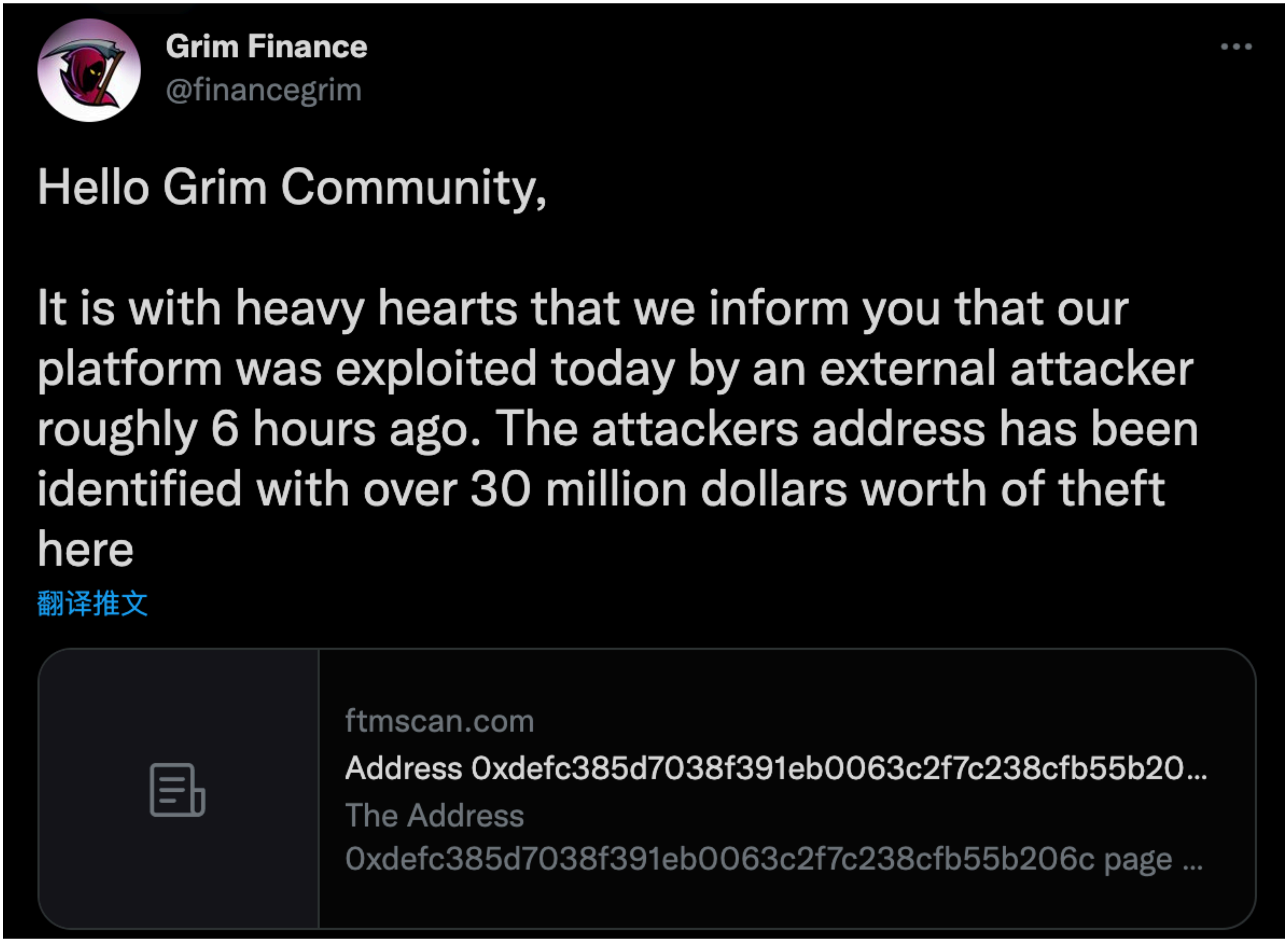
# 外部调用的不可控因素之 Grim Finance 被黑分析

Author: xxxeyJ

## 前言



Grim Finance
@financegrim

Hello Grim Community,

It is with heavy hearts that we inform you that our platform was exploited today by an external attacker roughly 6 hours ago. The attackers address has been identified with over 30 million dollars worth of theft here

ftmscan.com
Address 0xdefc385d7038f391eb0063c2f7c238cfb55b20...
The Address 0xdefc385d7038f391eb0063c2f7c238cfb55b206c page ...

Grim Finance 是一家位于 Fantom 链上的复合收益率优化器；据 Grim Finance 在 Twitter 发布的消息报道称 2021 年 12 月 19 日协议遭到黑客攻击，此次攻击事件预估总损失约为 $3000w - $4000w。

## 涉事资产

Grim Finance Exploiter 1: 0xdefc385d7038f391eb0063c2f7c238cfb55b206c

Balancer Vault: 0x20dd72ed959b6147912c2e529f0a0c651c33c9ce

Spirit LPs - SLP Token: 0x279b2c897737a50405ed2091694f225d83f2d3ba

GB-BTC-FTM: 0x660184ce8af80e0b1e5a1172a16168b15f4136bf

攻击合约 1: 0xb08ccb39741d746dd1818641900f182448eb5e41

攻击合约 2: 0x059c989a18990572a0537341903e0254857464b3

攻击合约 3: 0x1f8d2e68da1e44035864f08a698b02cb87cdd013

攻击合约 4: 0x577edc8d679ca0da67d95294e40d0685e871b468

攻击合约 5: 0xfecd91fe44868d795fddf347a0cd71330baab4a2

攻击合约 6: 0x360b65da2930ab6ffc801ff1ef365c92b75e878e

攻击交易 1: 0x19315e5b150d0a83e797203bb9c957ec1fa8a6f404f4f761d970cb29a74a5dd6

获利: 11.78381727 WBTC / 362,770.590819615422480573 WFTM

攻击交易 2: 0x497e6d97cb7ae75b33113d9bd793f9d977be3dd2d361006304ea80871dc6d9a0

获利: 755,621.556845439477817809 DAI / 755,107.037958 USDC

攻击交易 3: 0xa5413918bf6ab8757264437db33eae0763394010d139b48ab157925eb1c20406

获利: 14.66651695 WBTC

攻击交易 4: 0xf14ca312081996d84c5add7351c29d3e5e803eadf7d2db3d1b9a011c8da5ec94

获利: 1,951,895.698279216364199401 WFTM

攻击交易 5: 0x7b9ea370949331aadb99530eead580b4134e08dc0d90a9c7c8556de9ddf74233

获利: 170,249.993528467658920507 WFTM

攻击交易 6: 0x753551a7d98834621071d20a42978123f7fdaeb468a8801f0ad155bd7516bbb6

获利: 104,237.994398486028272678 BOO

攻击交易 7: 0x1e71556800bf607261384452ac86cd99d69f08cb5ff2d8ed11272671aa950d54

获利: 32,412.076801684098281814 BOO

攻击交易 8: 0xce6a505a6da960a8ce43659aa9a039092e4e4e682d0707866c6a78f2a28be2f1

获利: 15,454,992.738574061358888122 WFTM

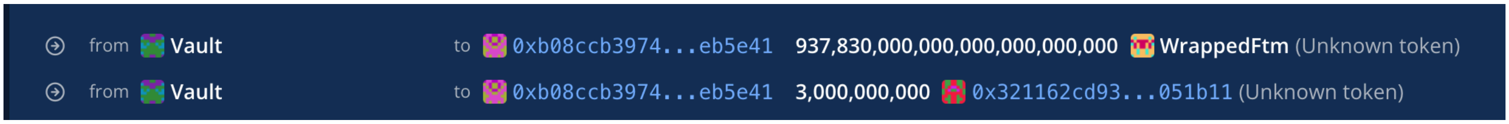
攻击交易 9: 0xb5e99eff2a3f37bf3da9e119e4b8544fdad38d5058e7e794a7810f2942f00b23

获利: 728,018.806854597056664641 WFTM

## 攻击流程

本次攻击分析将以 攻击交易1: 0x19315e5b150d0a83e797203bb9c957ec1fa8a6f404f4f761d970cb29a74a5dd6 为切入口举例说明。

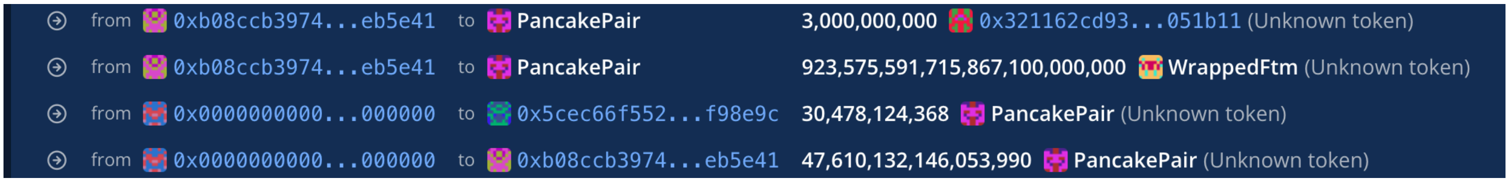
1）首先通过闪电贷为攻击合约借入 30 个 WBTC 和 937,830 个 WFTM。



from Vault to 0xb08ccb3974...eb5e41 937,830,000,000,000,000,000,000 WrappedFtm (Unknown token)

from Vault to 0xb08ccb3974...eb5e41 3,000,000,000 0x321162cd93...051b11 (Unknown token)

2）向 SpiritSwap Pair 注入 WBTC 和 WFTM 的流动性并以此铸造 SLP Token。



from 0xb08ccb3974...eb5e41 to PancakePair 3,000,000,000 0x321162cd93...051b11 (Unknown token)

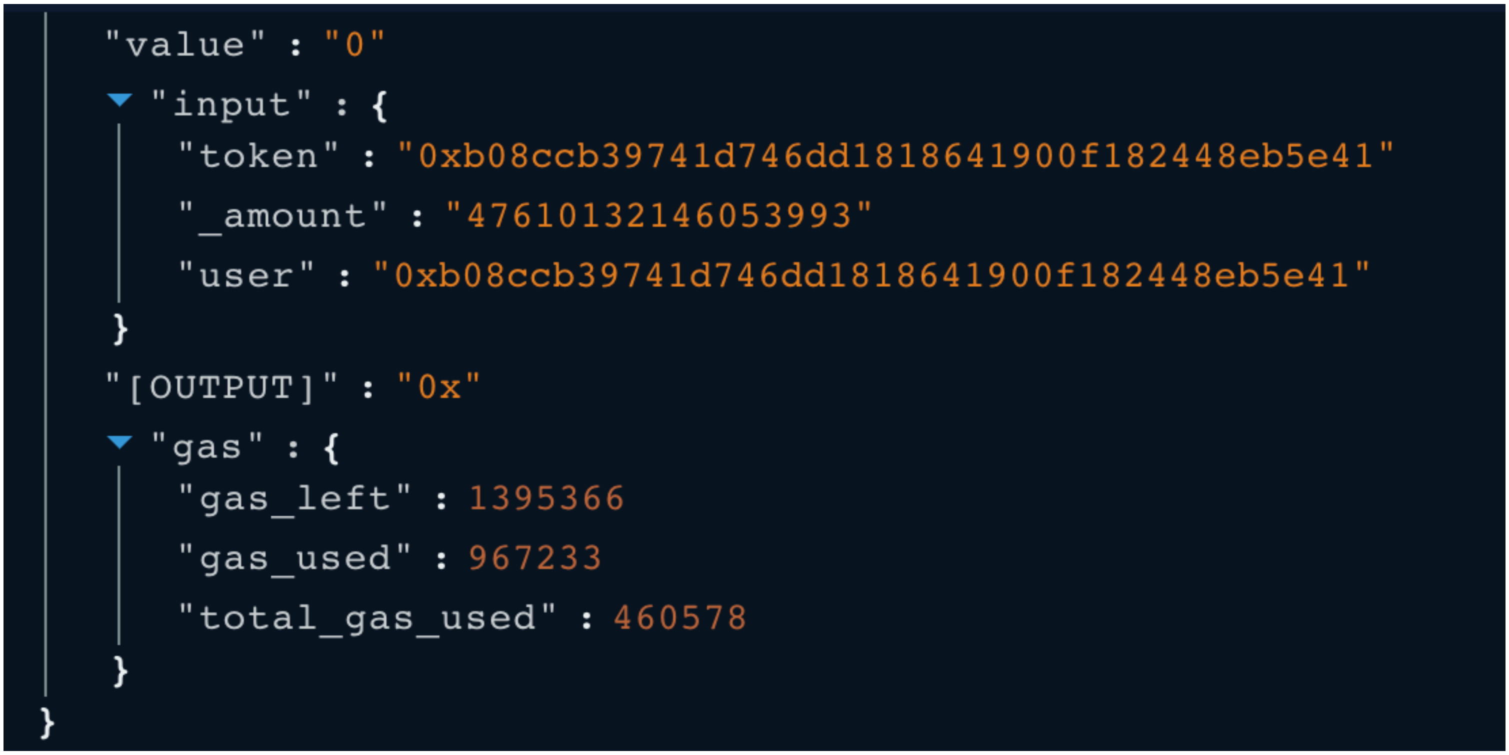
from 0xb08ccb3974...eb5e41 to PancakePair 923,575,591,715,867,100,000,000 WrappedFtm (Unknown token)

from 0x0000000000...000000 to 0x5cec66f552...f98e9c 30,478,124,368 PancakePair (Unknown token)

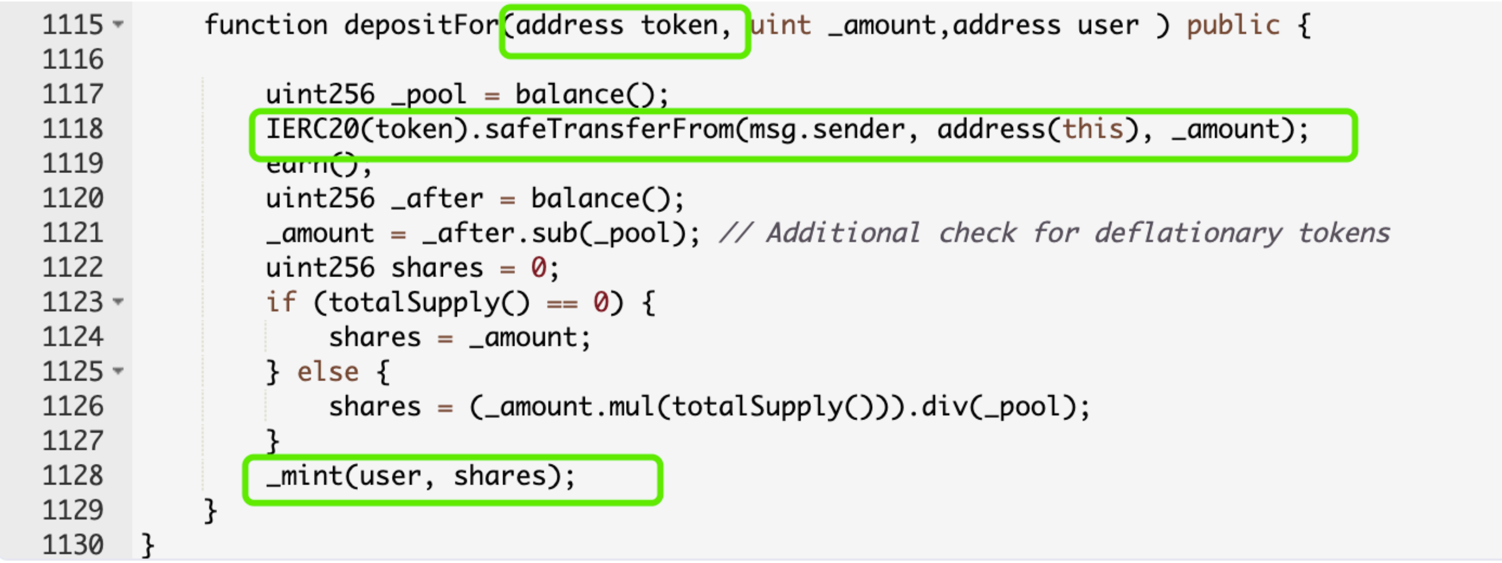
from 0x0000000000...000000 to 0xb08ccb3974...eb5e41 47,610,132,146,053,990 PancakePair (Unknown token)

3）随后攻击合约 `call` GrimBoostVault::depositFor() 并指定 `Token` 参数为攻击合约 1 自身 0xb08ccb39741d746dd1818641900f182448eb5e41

如下图所示，GrimBoostVault::depositFor() 并未校验 `token` 的正确性导致参数可控，从而形成了外部调用的空间，而后通过传递 `token=0xb08ccb39741d746dd1818641900f182448eb5e41,` `user=0xb08ccb39741d746dd1818641900f182448eb5e41` 以此调用 depositFor() 函数，利用攻击合约中的 `transferForm()` 回调 `GrimBoostVault::depositFor()` 从而实现了重入攻击，最终因重入攻击为攻击者铸造出了大量的 `GB-TOKEN0-TOKEN1`。

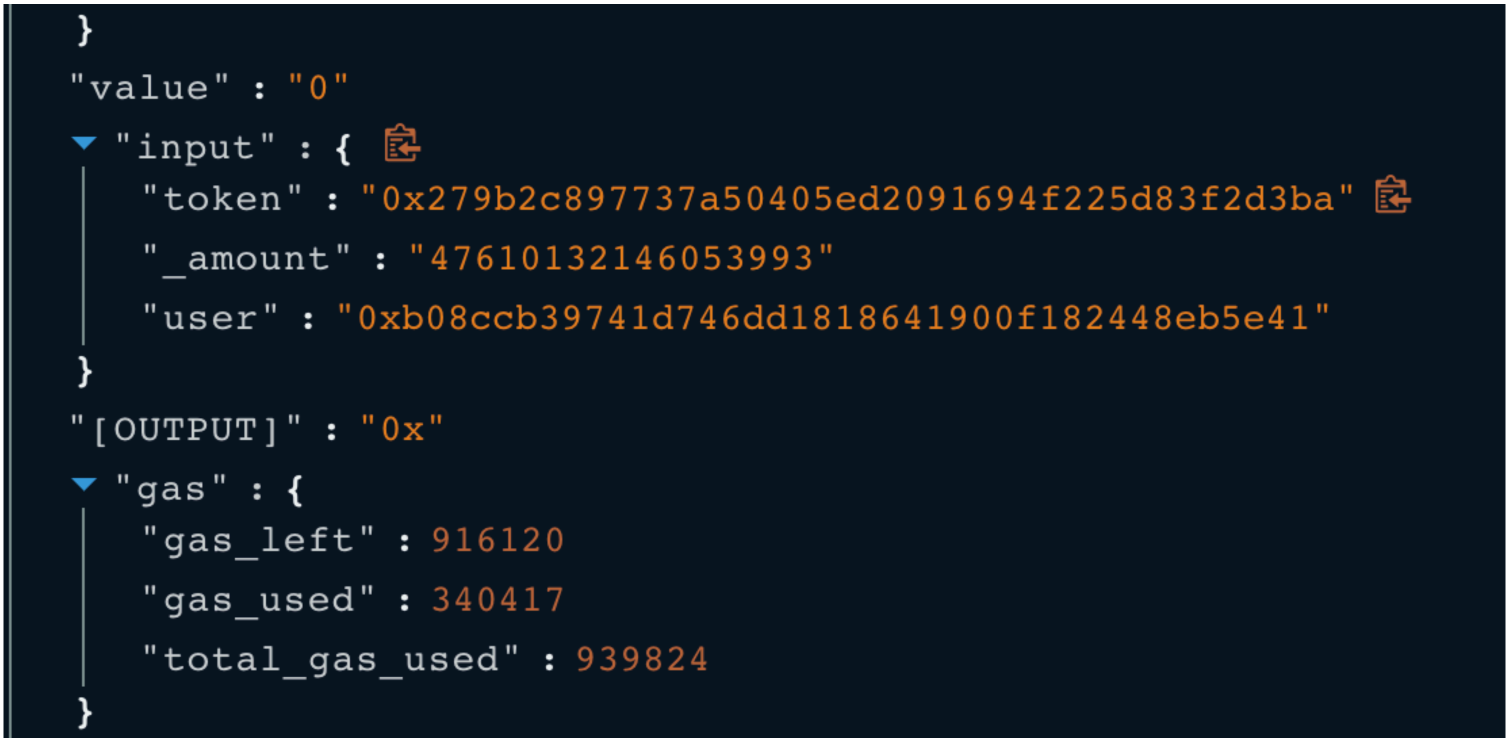


```
    "value" : "0"
    "input" : {
      "token" : "0xb08ccb39741d746dd1818641900f182448eb5e41"
      "_amount" : "47610132146053993"
      "user" : "0xb08ccb39741d746dd1818641900f182448eb5e41"
    }
    "[OUTPUT]" : "0x"
    "gas" : {
      "gas_left" : 1395366
      "gas_used" : 967233
      "total_gas_used" : 460578
    }
}
```



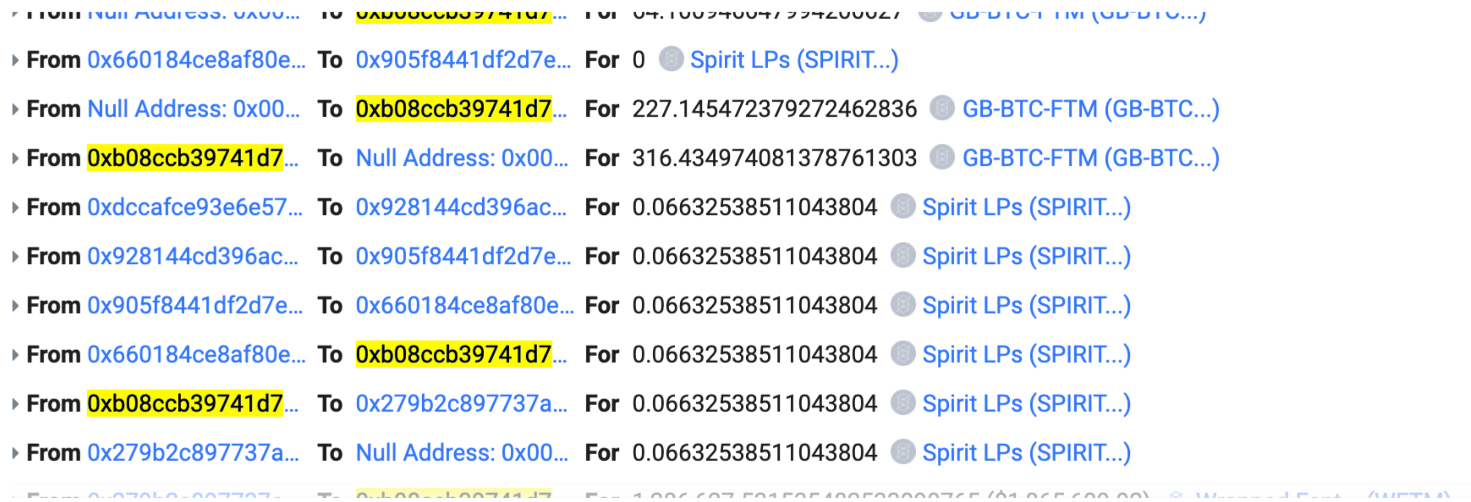
```
    function depositFor(address token, uint _amount,address user ) public {

        uint256 _pool = balance();
        IERC20(token).safeTransferFrom(msg.sender, address(this), _amount);
        earn();
        uint256 _after = balance();
        _amount = _after.sub(_pool); // Additional check for deflationary tokens
        uint256 shares = 0;
        if (totalSupply() == 0) {
            shares = _amount;
        } else {
            shares = (_amount.mul(totalSupply())).div(_pool);
        }
        _mint(user, shares);
    }
}
```

4）最后在重入攻击的最后一步 (step 5) 传入真实的 SLP Token，并利用巨大差额为其铸造了大量的 `GB-BTC-FTM`。



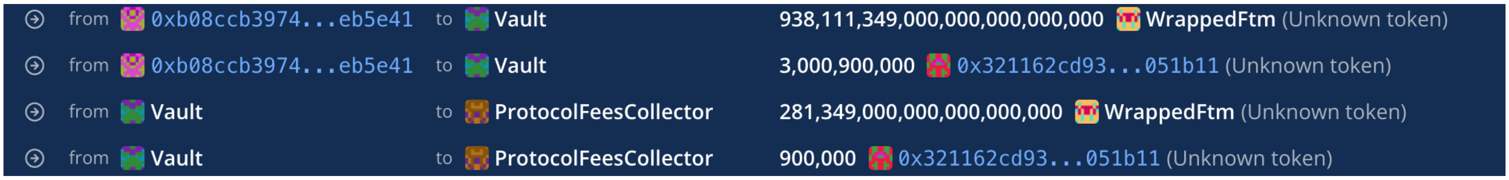
```
  }
  "value" : "0"
  ▼ "input" : {
      "token" : "0x279b2c897737a50405ed2091694f225d83f2d3ba"
      "_amount" : "4761013214605399З"
      "user" : "0xb08ccb39741d746dd1818641900f182448eb5e41"
  }
  "[OUTPUT]" : "0x"
  ▼ "gas" : {
      "gas_left" : 916120
      "gas_used" : 340417
      "total_gas_used" : 939824
  }
```

5）利用先前获取到的 `GB-TOKEN0-TOKEN1` 拿到了预期之外 `SLP Token`，通过 `SLP Token` 移除流动性并获取到了大量的 WBTC / WFTM。



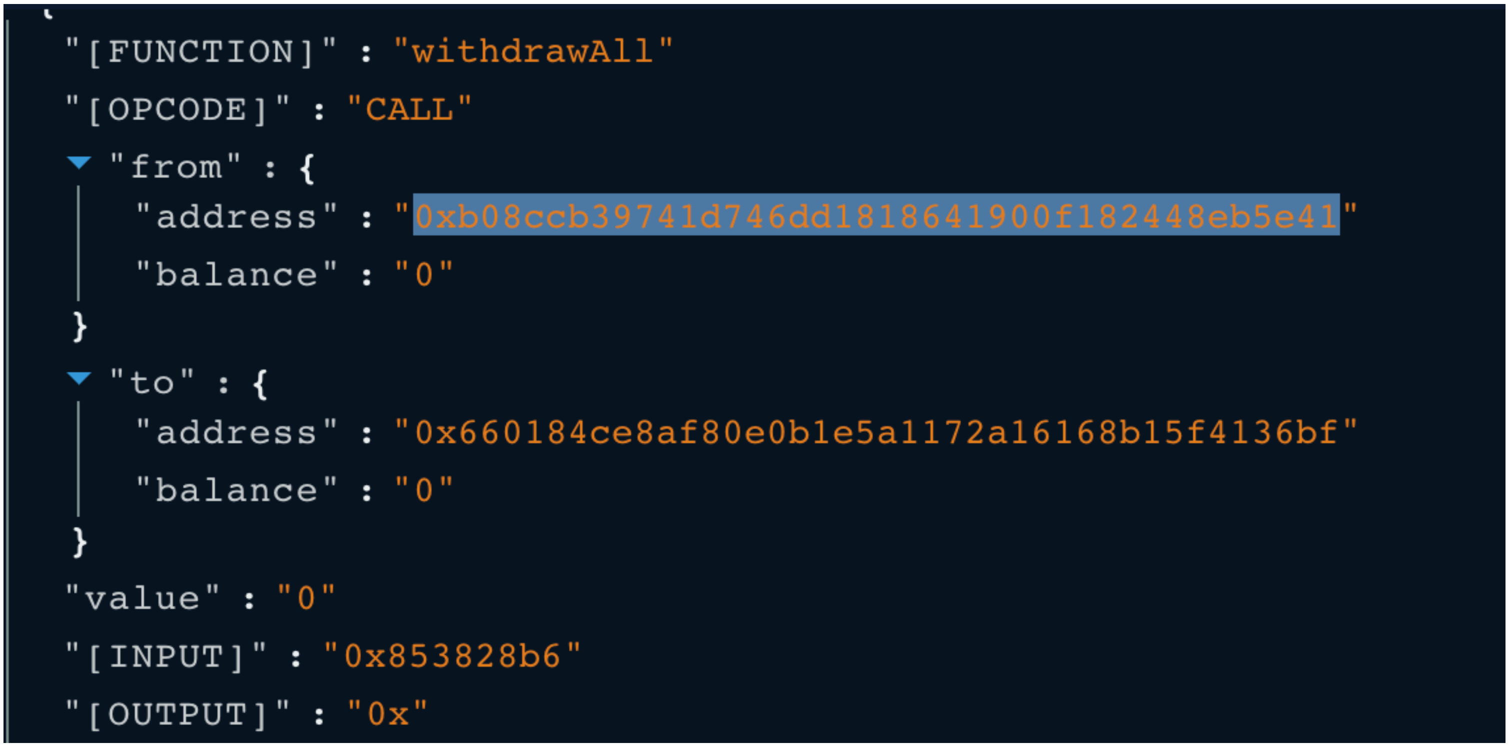
‣ From 0x660184ce8af80e... To 0x905f8441df2d7e... For 0 Spirit LPs (SPIRIT...)
‣ From Null Address: 0x00... To 0xb08ccb39741d7... For 227.145472379272462836 GB-BTC-FTM (GB-BTC...)
‣ From 0xb08ccb39741d7... To Null Address: 0x00... For 316.434974081378761303 GB-BTC-FTM (GB-BTC...)
‣ From 0xdccafce93e6e57... To 0x928144cd396ac... For 0.06632538511043804 Spirit LPs (SPIRIT...)
‣ From 0x928144cd396ac... To 0x905f8441df2d7e... For 0.06632538511043804 Spirit LPs (SPIRIT...)
‣ From 0x905f8441df2d7e... To 0x660184ce8af80e... For 0.06632538511043804 Spirit LPs (SPIRIT...)
‣ From 0x660184ce8af80e... To 0xb08ccb39741d7... For 0.06632538511043804 Spirit LPs (SPIRIT...)
‣ From 0xb08ccb39741d7... To 0x279b2c897737a... For 0.06632538511043804 Spirit LPs (SPIRIT...)
‣ From 0x279b2c897737a... To Null Address: 0x00... For 0.06632538511043804 Spirit LPs (SPIRIT...)

6）归还闪电贷并支付手续费。

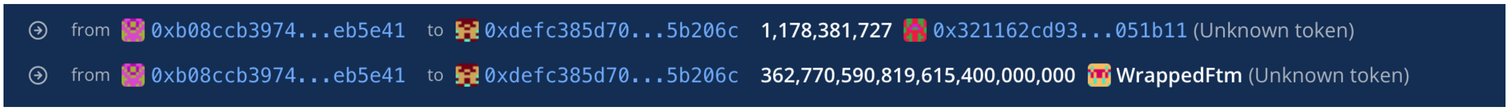


| | | | | | |
|---|---|---|---|---|---|
| from | 0xb08ccb3974...eb5e41 | to | Vault | 938,111,349,000,000,000,000,000 | WrappedFtm (Unknown token) |
| from | 0xb08ccb3974...eb5e41 | to | Vault | 3,000,900,000 | 0x321162cd93...051b11 (Unknown token) |
| from | Vault | to | ProtocolFeesCollector | 281,349,000,000,000,000,000 | WrappedFtm (Unknown token) |
| from | Vault | to | ProtocolFeesCollector | 900,000 | 0x321162cd93...051b11 (Unknown token) |

7）最终攻击者调用 `GrimBoostVault::withdrawAll()` 提取资金以此获利。



```
  "[FUNCTION]" : "withdrawAll"
  "[OPCODE]" : "CALL"
  "from" : {
     "address" : "0xb08ccb39741d746dd1818641900f182448eb5e41"
     "balance" : "0"
  }
  "to" : {
     "address" : "0x660184ce8af80e0b1e5a1172a16168b15f4136bf"
     "balance" : "0"
  }
  "value" : "0"
  "[INPUT]" : "0x853828b6"
  "[OUTPUT]" : "0x"
```



```
from 0xb08ccb3974...eb5e41  to 0xdefc385d70...5b206c  1,178,381,727 0x321162cd93...051b11 (Unknown token)
from 0xb08ccb3974...eb5e41  to 0xdefc385d70...5b206c  362,770,590,819,615,400,000,000 WrappedFtm (Unknown token)
```

## 总结

归根结底，本次攻击事件的核心问题还是在于并未对关键函数建立完备的访问控制机制。因而形成了关键参数可控的情况，这也就导致了此次攻击事件的产生。