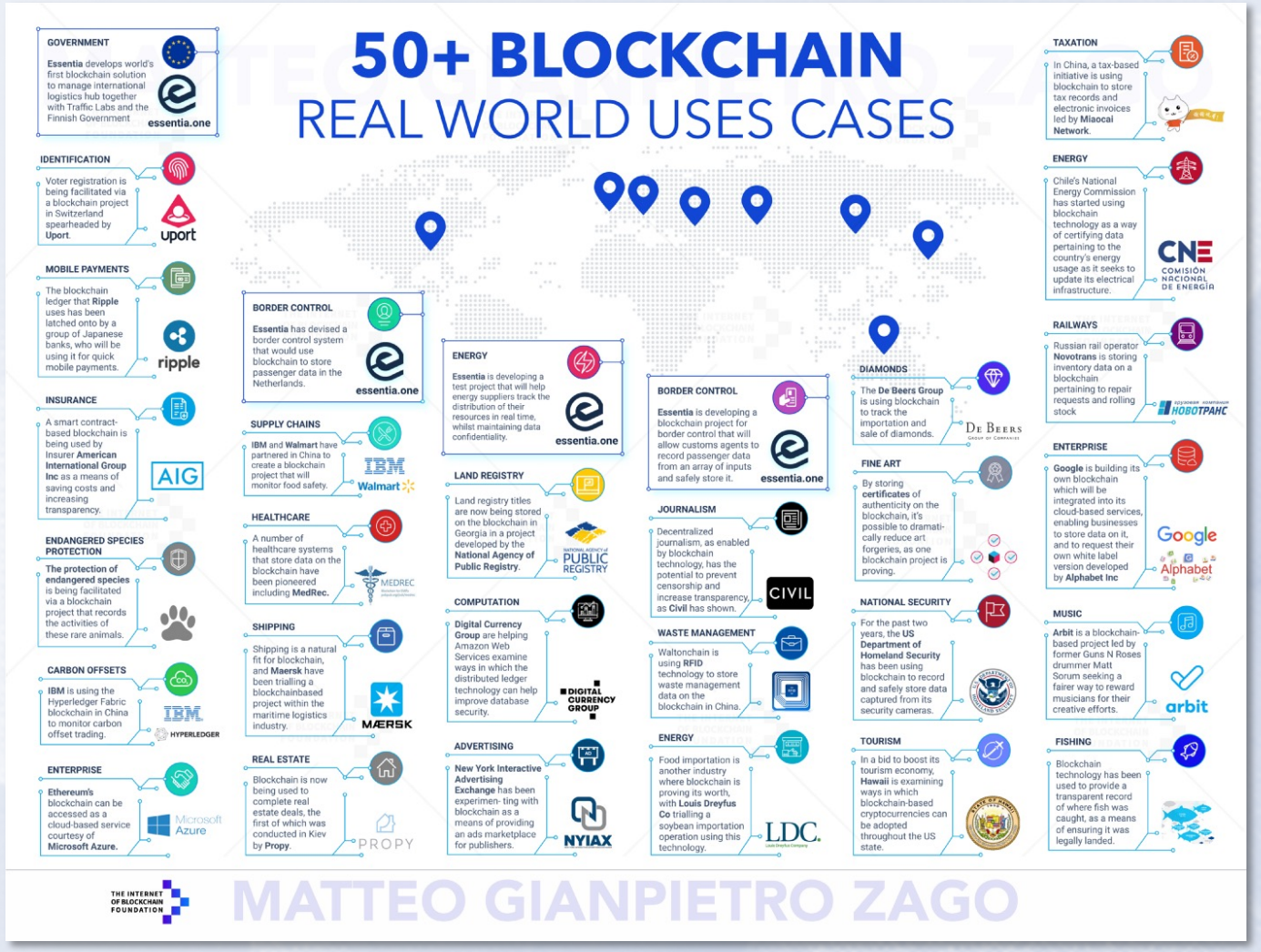
CS 190: DeFi Security

Yanju Chen

UCSB

Blockchain Real-World Use Cases



- Government
- Waste Management
- Identification
- Border Control
- Healthcare
- Enterprise
- Medical
- Music
- Carbon Offsets
- Supply Chains
- Diamonds
- Real Estate
- Fishing Industry
- Fine Art
- Public Utilities
- LGBT Rights
- ...

source: https://medium.com/@essentia1/50-examples-of-how-blockchains-are-taking-over-the-world-4276bf488a4b

Different Blockchains

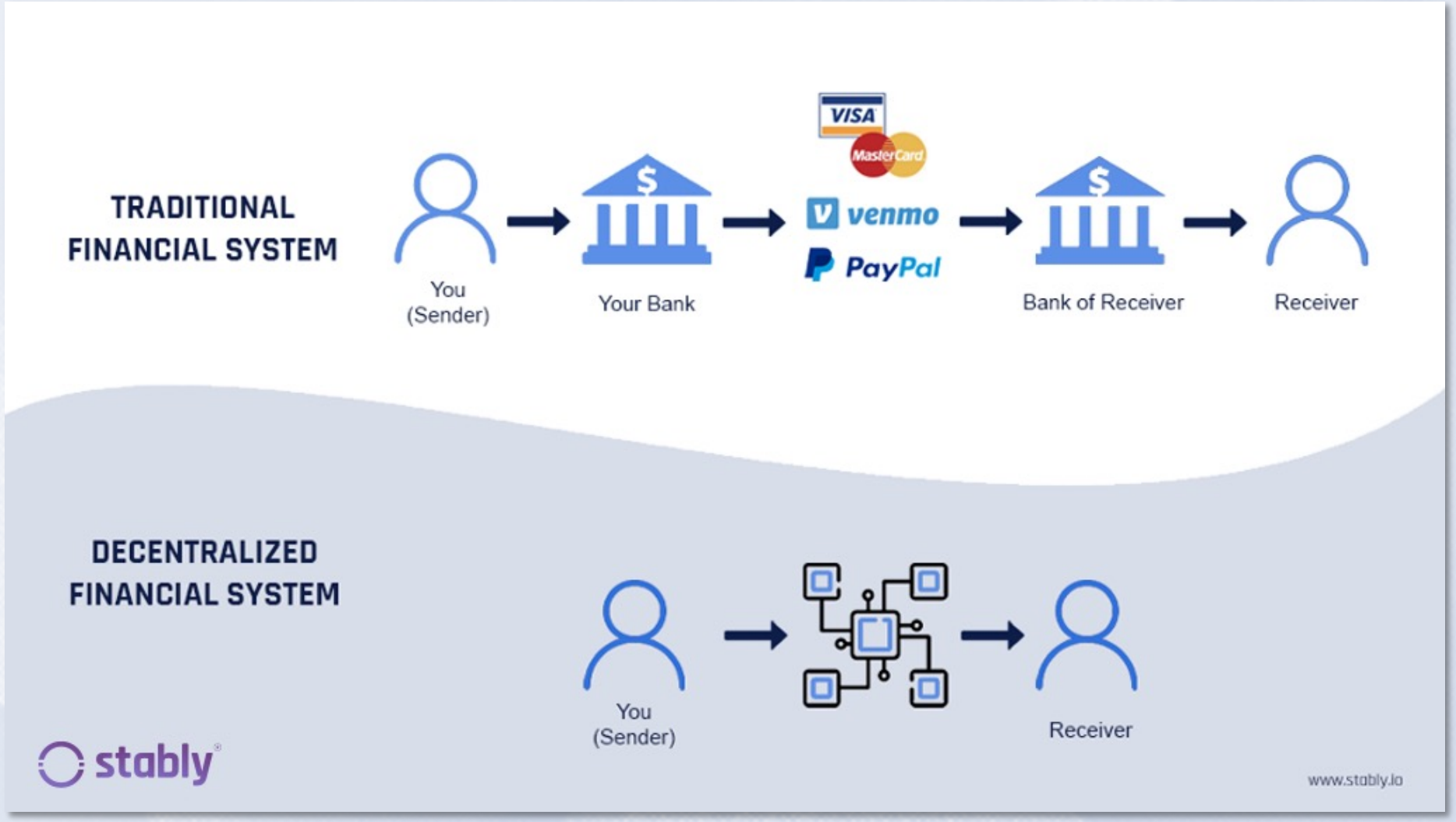
	Name	Brief (Marketing) Description	Website	Whitepaper/Docs
1	Bitcoin	First, and most secure PoW chain.	https://bitcoin.org/de/	https://bitcoin.org/bit.
2	Ethereum	Biggest smart contract platform	https://ethereum.org/	https://ethereum.org/
3	Near	Sharded PoS smart contract bloc...	https://nearprotocol.co...	https://near.org/pape.
4	Cosmos	Blockchain ecosystem of separat...	https://cosmos.network/	https://cosmos.netw..
5	Solana	High-throughput layer 1 blockch...	https://solana.com/	https://solana.com/s..
6	Polkadot	Blockchain ecosystem with share...	http://polkadot.network/	https://polkadot.netw
7	Avalanche	Smart contract blockchain	https://www.avalabs.org/	https://www.avalabs..
8	Terra	Protocol for stablecoins based o...	https://terra.money/pro...	https://terra.money/s.
9	Binance Smart...	Cheap Eth fork with solid adoptio...	https://www.binance.or...	https://www.binance..
10	Algorand	Smart asset blockchain, simple s...	https://www.algorand.c...	https://arxiv.org/abs/..
11	Waves	Smart asset blockchain with focu...	https://wavesplatform.c...	https://medium.com/..
12	Litecoin	Early, prominent Bitcoin fork	https://litecoin.org/	https://litecoin.info/in.
13	Tezos	Smart contract blockchain	https://tezos.com/	https://tezos.com/sta.
14	Celo	EVM-compatible platform for sta...	https://celo.org/	https://www.google.c.
15	Tron	Smart contract blockchain	https://tron.network/	https://tron.network/..
16	Fetch.ai	Artificial intelligence for blockch...	https://fetch.ai/	https://fetch.ai/wp-c..

Total Cryptocurrency Market Cap: $2,181,617,785,219

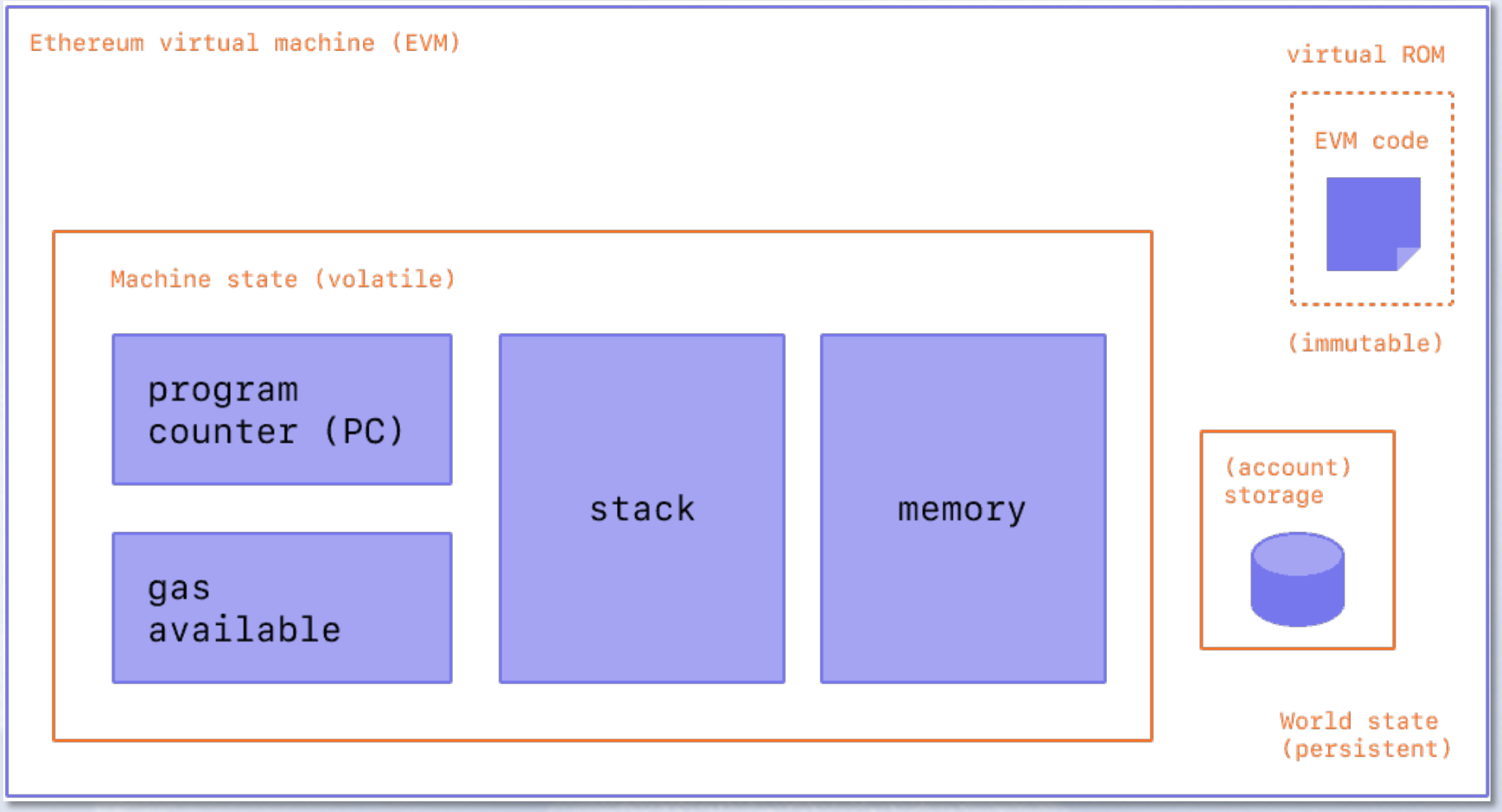
Rank	Name (Symbol)	Market Cap	Market Share	Price (USD)	24 Hr % Change
1	Bitcoin (BTC)	1,143,346,581,744	52.4082%	$58,059.3028660232	-3.61014622
2	Ethereum (ETH)	362,714,596,232	16.6259%	$2,971.6114725156	-0.50319152
3	Tether USDt (USDT)	110,429,669,698	5.0618%	$0.9984985103	-0.02877873
4	BNB (BNB)	82,301,058,695	3.7725%	$557.641758134	-3.45124171
5	Solana (SOL)	59,794,636,570	2.7408%	$133.7076389914	6.31103393
6	USDC (USDC)	33,024,535,065	1.5138%	$0.9998912841	-0.01967419
7	XRP (XRP)	28,448,124,632	1.304%	$0.515029498	3.6473889
8	Dogecoin (DOGE)	18,597,827,304	0.8525%	$0.1290480744	-2.45618034
9	Toncoin (TON)	16,902,193,348	0.7748%	$4.8668358369	-4.58249586
10	Cardano (ADA)	16,043,533,800	0.7354%	$0.4501158897	2.52304259
11	Shiba Inu (SHIB)	13,227,483,506	0.6063%	$0.0000224465	0.53544209
12	Avalanche (AVAX)	12,554,258,814	0.5755%	$33.0819376847	1.96334041
13	TRON (TRX)	10,490,758,990	0.4809%	$0.1198427426	0.82466309
14	Polkadot (DOT)	9,913,085,738	0.4544%	$6.8938851018	10.6987416
15	Bitcoin Cash (BCH)	8,286,754,722	0.3798%	$420.6514514599	-2.22802579
16	Chainlink (LINK)	7,779,686,788	0.3566%	$13.2510427173	1.6330653
17	Polygon (MATIC)	6,809,082,290	0.3121%	$0.6878540078	4.22123544

source: https://www.slickcharts.com/currency

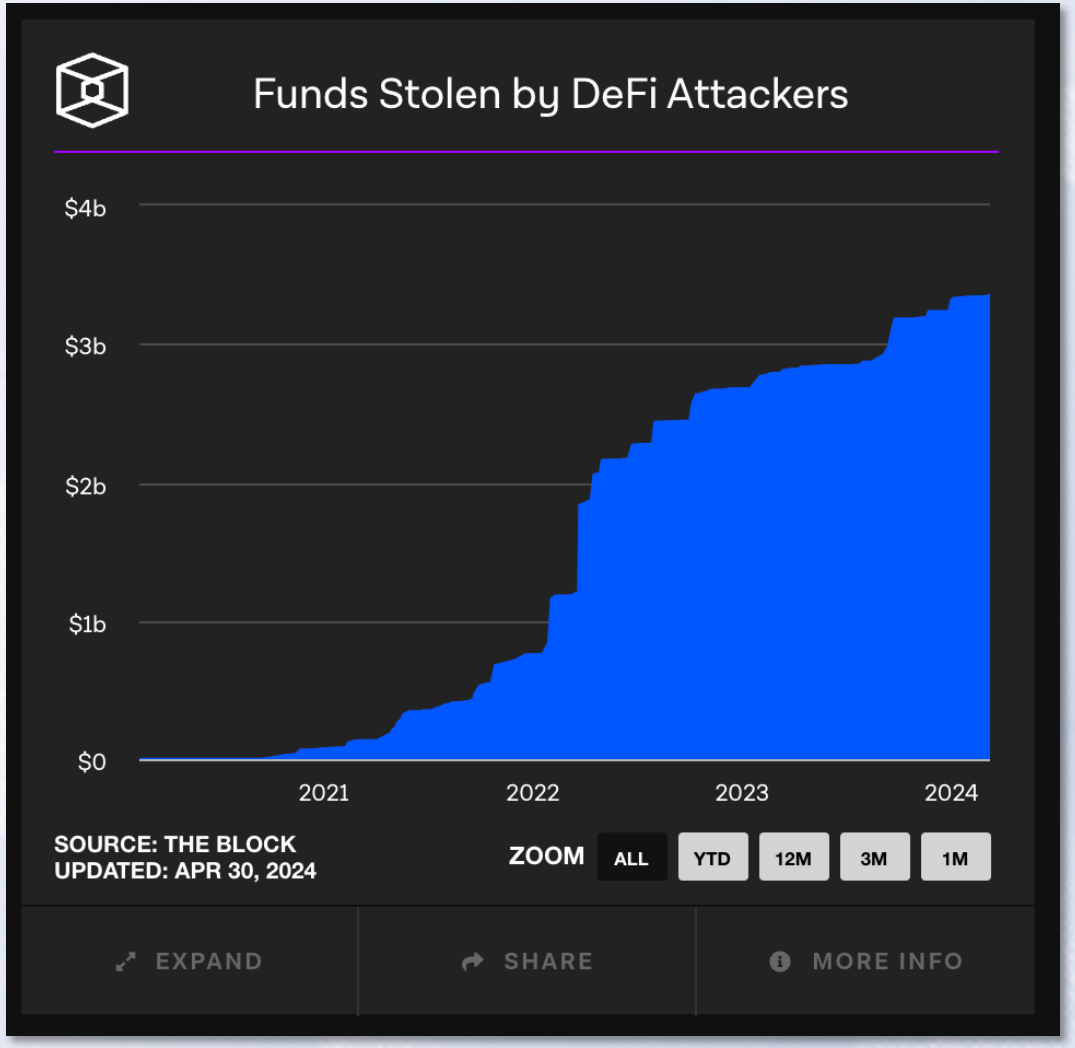
DeFi: Decentralized Finance

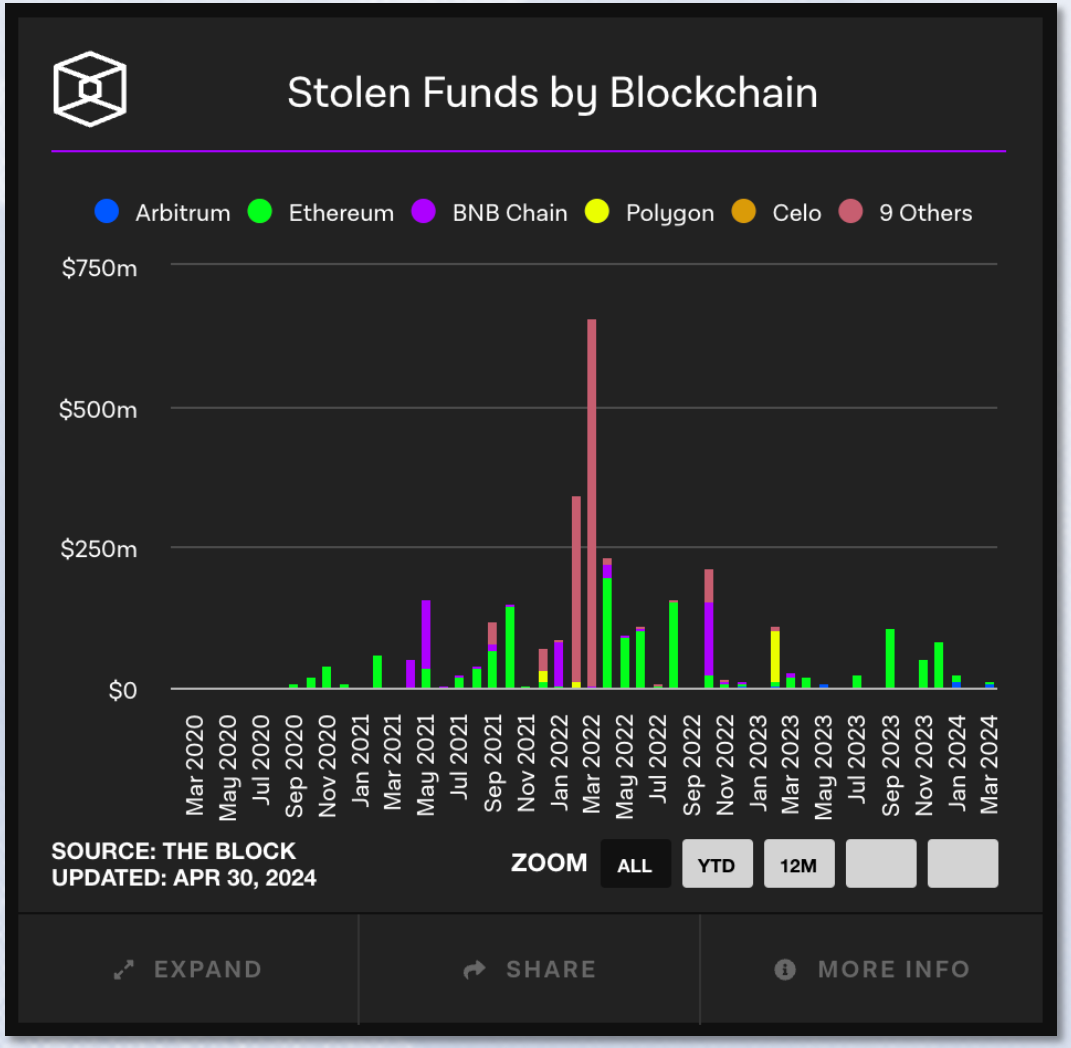


Solidity & Ethereum Virtual Machine (EVM)



Statistics #1: DeFi Hacks by Year and Chain



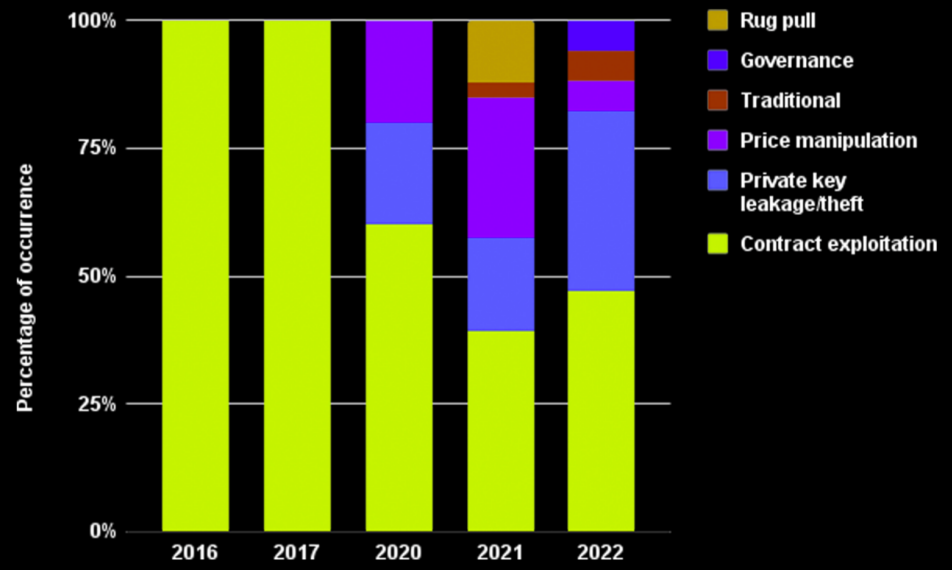


source: https://www.theblock.co/data/decentralized-finance/exploits

Statistics #2: DeFi Hacks by Type

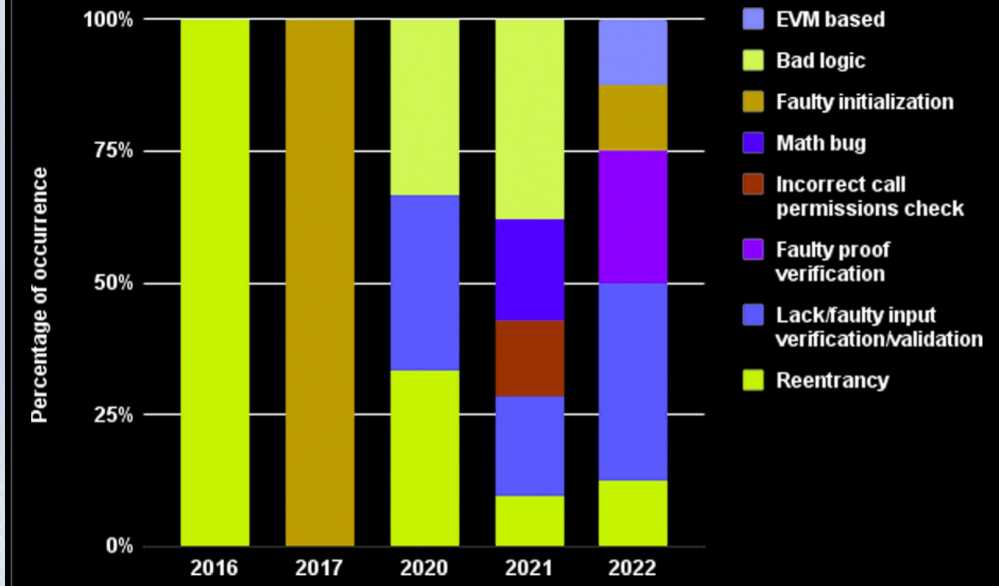
TYPES OF ATTACKS PER YEAR[1]

1. Smart contract vulnerabilities account for 47% of the top 50 attacks.
2. The second most common attack is private key leakage or theft, which represents 22% of all hacks.
3. Lastly, price manipulation accounts for 19% of attacks.



A BREAKDOWN OF SMART CONTRACT VULNERABILITY TYPES[1]

1. The most common smart contract vulnerability type is a logic bug, accounting for 26% of all smart contract hacks.
2. Failed input validation is the second most common vulnerability type, making up 23% of attacks.



source: https://www.halborn.com/reports/top-50-defi-hacks

Statistics #3: Top Hacks

Name	Date	Classification	Technique	Amount lost
Ronin	23 Mar, 2022, 00:00	Infrastructure	Private Key Compromised (Social Engin...	$624m
Poly Network	10 Aug, 2021, 00:00	Protocol Logic	Access Control Exploit	$611m
Binance Bridge	6 Oct, 2022, 00:00	Protocol Logic	Proof Verifier Bug	$570m
FTX	12 Nov, 2022, 00:00	Infrastructure	Private Key Compromised (Unknown M...	$450m
Wormhole	2 Feb, 2022, 00:00	Protocol Logic	Signature Exploit	$326m
Gate.io	21 Apr, 2018, 00:00	Infrastructure	Private Key Compromised (Unknown M...	$235m
Mixin Network	23 Sep, 2023, 00:00	Infrastructure	Database Attack	$200m
Euler Finance	13 Mar, 2023, 00:00	Protocol Logic	Flashloan Donate Function Logic Exploit	$197m
Bitmart	4 Dec, 2021, 00:00	Infrastructure	Private Key Compromised (Unknown M...	$196m
Nomad	1 Aug, 2022, 00:00	Protocol Logic	Trusted Root Exploit	$190m
Beanstalk	17 Apr, 2022, 00:00	Ecosystem	Flashloan Governance Attack	$181m
Wintermute	20 Sep, 2022, 00:00	Infrastructure	Private Key Compromised (Brute Force)	$160m

source: https://defillama.com/hacks

Arithmetic Overflow/Underflow

TRY DEMO

Only for Solidity < 0.8

Integers in Solidity overflow / underflow without any errors.

Solidity < 0.8

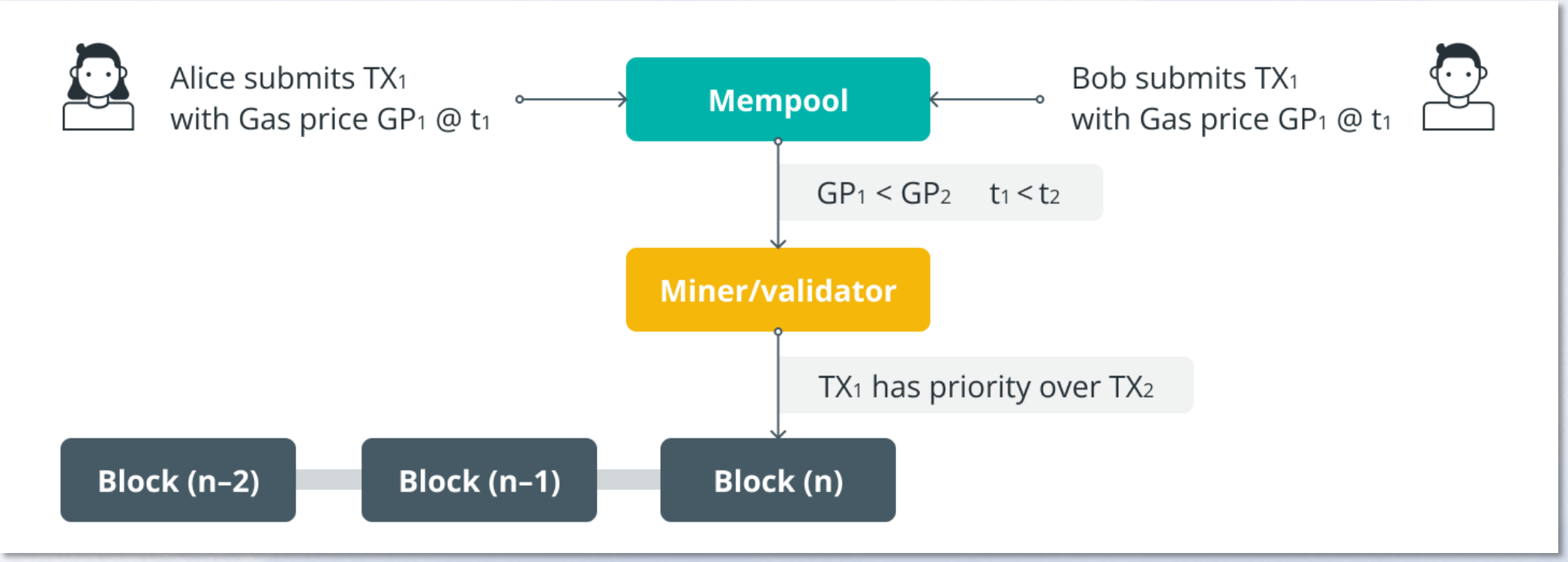
Integers in Solidity overflow / underflow with exception thrown.

Solidity >= 0.8


```
1  type(uint256).max + 1
```

How to overflow

Front Running



Transactions take some time before they are mined. An attacker can watch the transaction pool and send a transaction, have it included in a block **before** the original transaction. This mechanism can be abused to re-order transactions to the attacker's advantage.

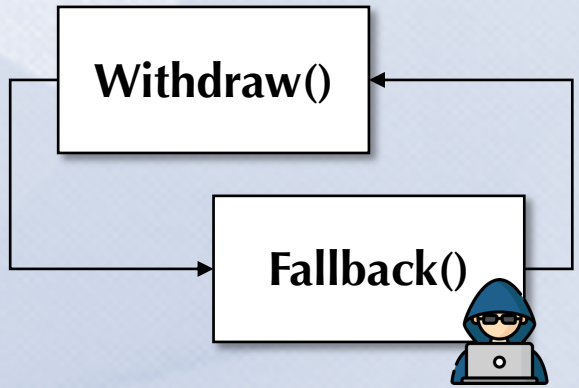
Reentrancy

TRY DEMO

Let's say that contract A calls contract B. Reentracy exploit allows B to call back into A *before A finishes execution*.

```
contract Example {

    mapping(address => uint) public balances;

    function deposit() public payable {
        balances[msg.sender] += msg.value;
    }

    function withdraw() public {
        uint bal = balances[msg.sender];
        require(bal > 0);

        (bool sent, ) = msg.sender.call{value: bal}("");
        require(sent, "Failed to send Ether");

        balances[msg.sender] = 0;
    }

}
```

Victim




```
fallback() external payable {
    if (address(example).balance >= 1 ether) {
        example.withdraw();
    }
}
```

Attacker

Withdraw multiple times!!

Denial of Service

TRY DEMO

Attack a smart contract and make it unusable. There are many ways to do that.

Suspicious Code Snippets to Look at

```
require(msg.value > balance, "Need to pay more to become the king");
```

```
(bool sent,) = king.call{value: balance}("");
```

```
if (address(asset) != _token)
    revert UnsupportedCurrency();
```

```
assembly { // better safe than sorry
    if eq(sload(0), 2) {
        mstore(0x00, 0xed3ba6a6)
        revert(0x1c, 0x04)
    }
}
```

… and many more

Self Destruct

selfdestruct-0

Contracts can be deleted from the blockchain by calling selfdestruct.

Deprecated after the Cancun hard fork

selfdestruct sends all remaining Ether stored in the contract to a designated address.

Still valid!!

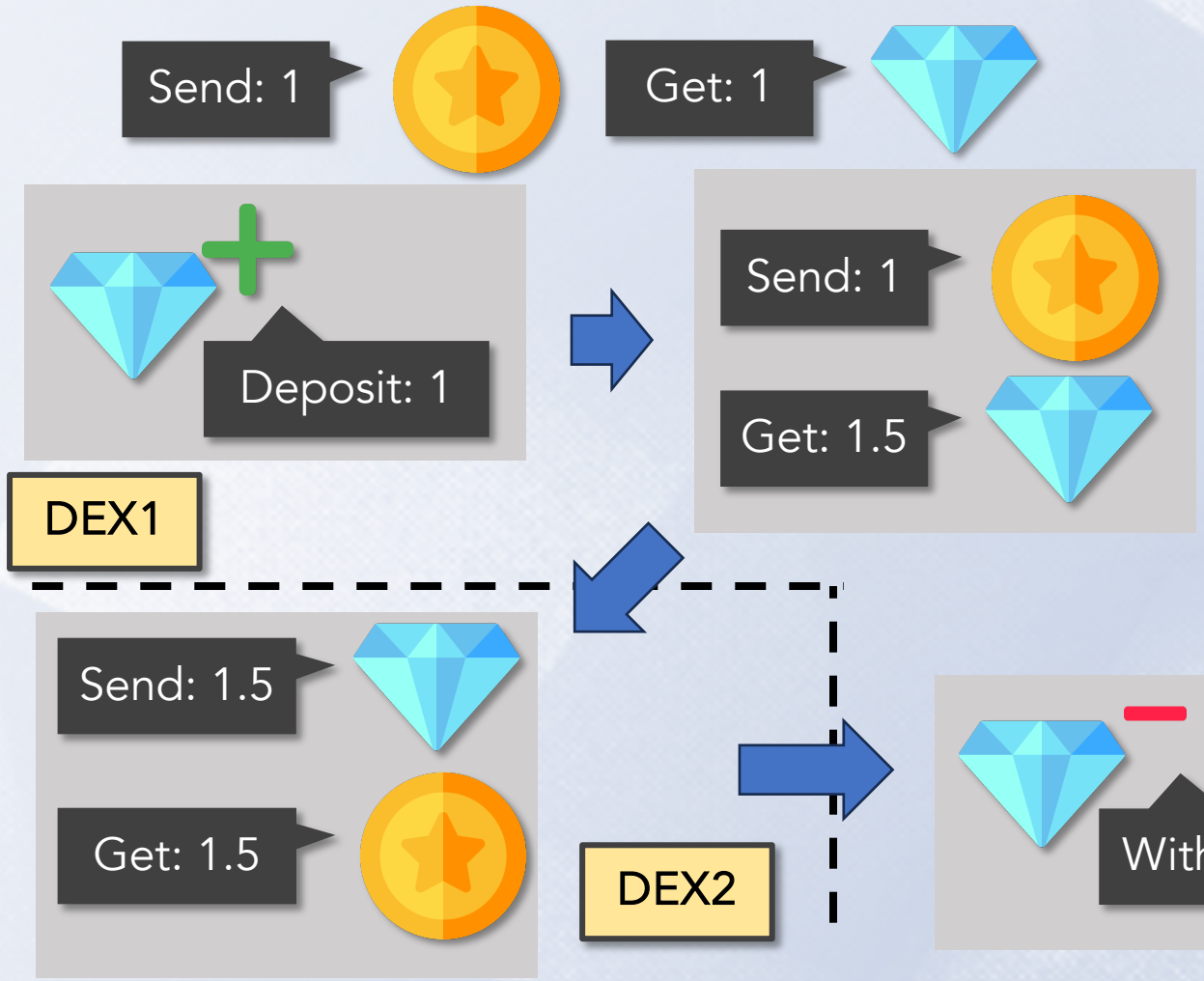
```
1  selfdestruct(beneficiary);
```

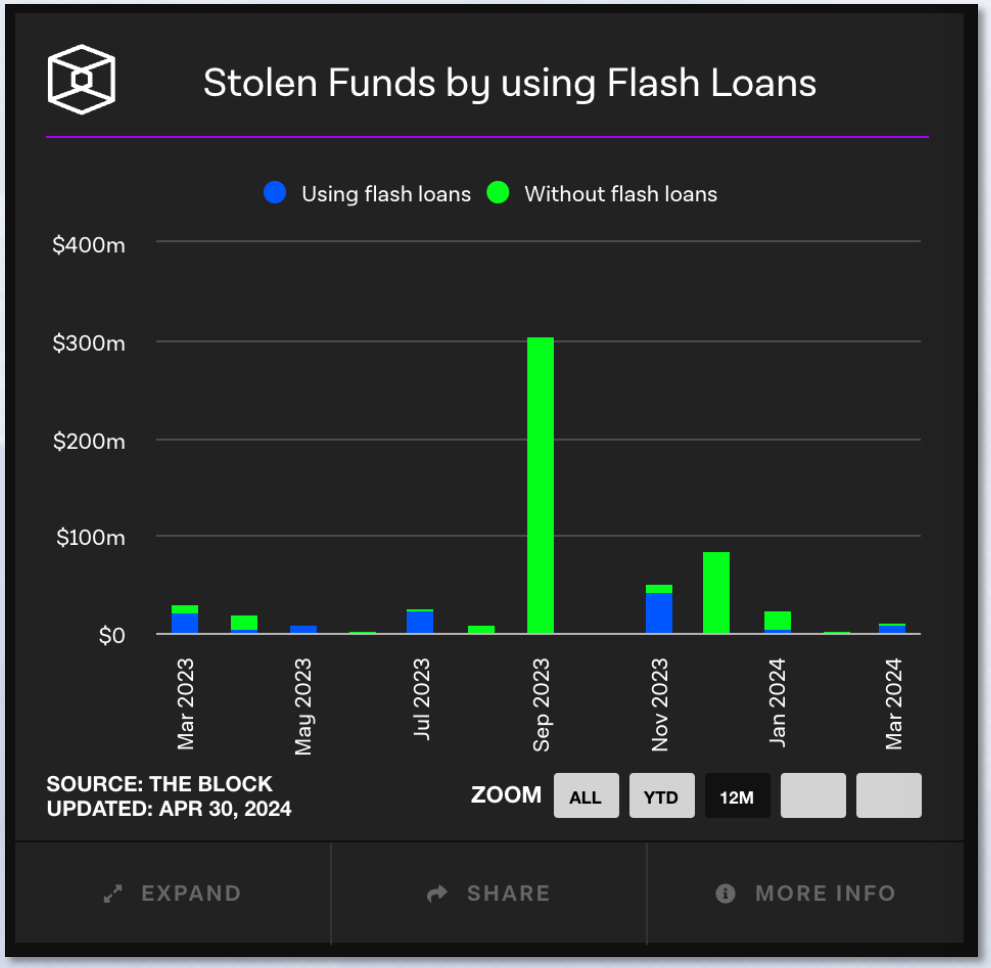
How to exploit

It could usually cause denial of service, but also needs to combine with other exploits.

Price Manipulation

usually paired with flashloan attacks





A Real-World Price Manipulation Bug

```
1  function DeFiLender.flashloan(...) public {}
2  function DeFiLender.payback(...) public {}
3
4  function Uniswap.swap(...) public {}
5
6  // transfer and swapBack of victim
7  function transfer(to, amt) public {
8      ...
9      if (...) { swapBack(); }
10     ...
11 }
12
13 function swapBack() internal {
14     ...
15
16     Uniswap.swap(victim, wBNB, amt_1, 0);
17     ...
18 }
```

```
1  function exploit() {
2      // flashloan wBNB
3      DeFiLender.flashloan(wBNB, 6.3e16);
4
5      // trigger the bug by transfer
6      Victim.transfer(address(0x0), 0);
7
8      // swap: wBNB -> victim -> bUSD -> wBNB
9
10     Uniswap.swap(wBNB, victim, 6.3e16, 1.2e12);
11
12     Uniswap.swap(victim, bUSD, 1.2e12, 16.5e18);
13
14     Uniswap.swap(bUSD, wBNB, 16.5e18, 6.5e16);
15
16     // payback wBNB
17     DeFiLender.payback(wBNB, 6.3e16);
18 }
```

credit to Hongbo Wen @ PLSELab