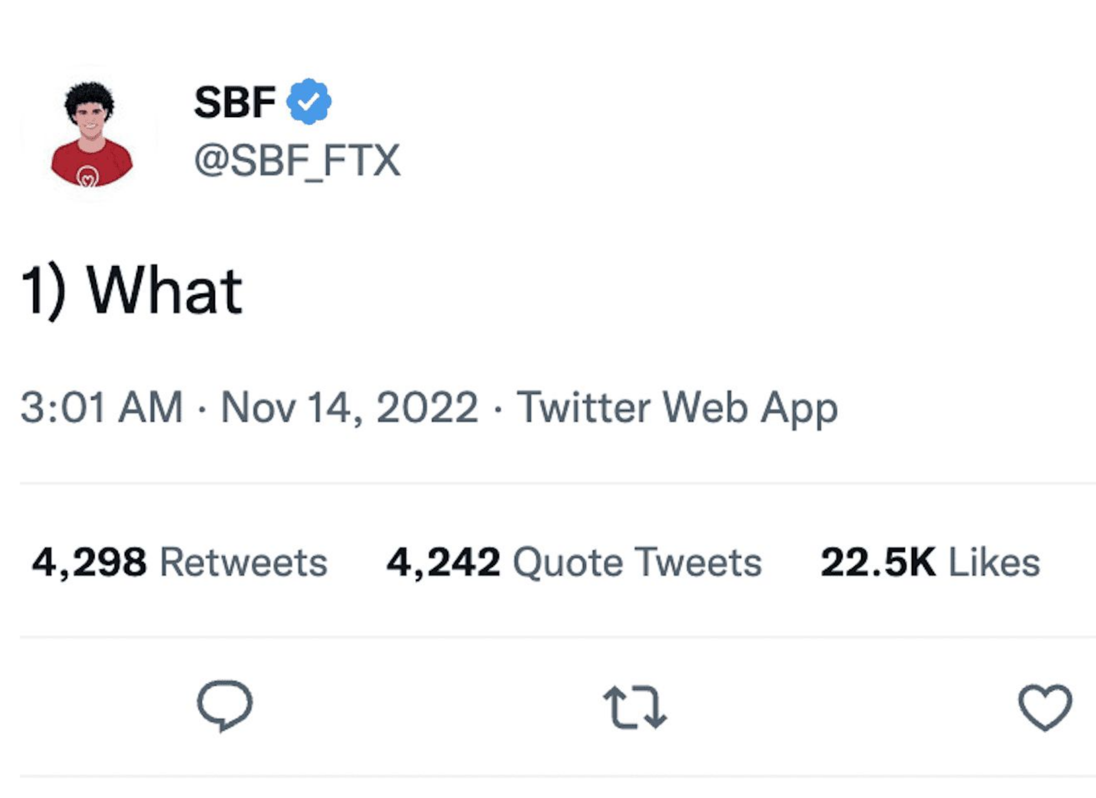
TORII

Adventure into Solana Security

About me

- @meltedblocks
- CEO and Co-Founder of Torii Security
- Web3 development and security
- Web developer and penetration tester
- Bots, infra, trading



SBF

@SBF_FTX

1) What

3:01 AM · Nov 14, 2022 · Twitter Web App

4,298 Retweets **4,242** Quote Tweets **22.5K** Likes

WHAT

- Interesting (subjective) Solana security traits
- Tips and tricks how to not get hacked
- Is this project secure?

WHO

- Developers
- Investors / Traders
- Security
- Curious minds

Adventure into Solana Security

WHY

- Know the threats - be prepared
- Code secure programs
- Avoid rug pulls

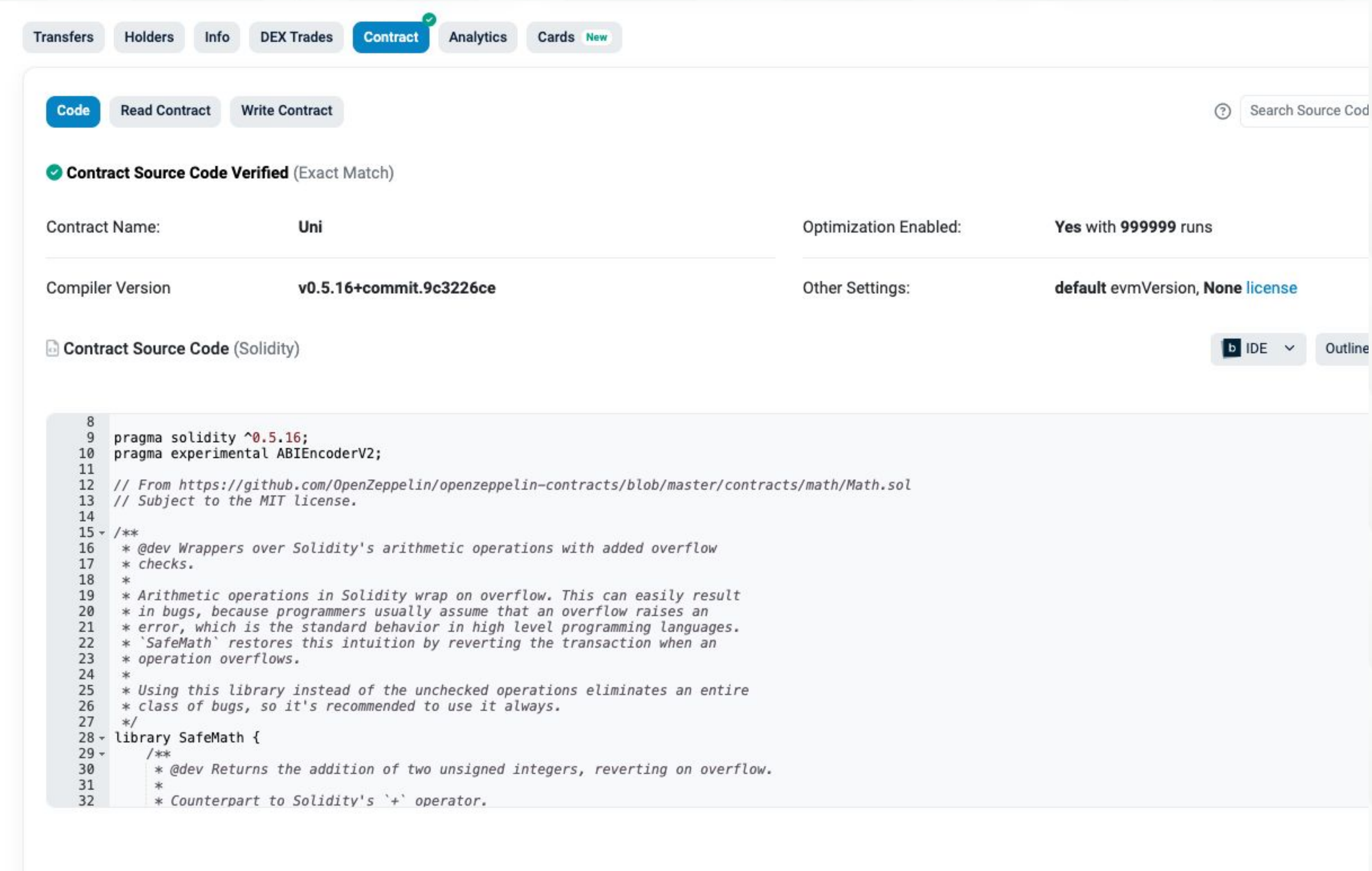
Adventure into Solana Security

Rust

- Steep learning curve
- Good compiler checks
- Systems programming language that typically compiles to **native machine code**

Scenario 1

1. Project is Open-Source
2. Protocol code is in GitHub
3. Developers claim that this code is deployed on chain
4. Can we trust them?



Contract Source Code Verified (Exact Match)

Contract Name: Uni

Optimization Enabled: Yes with 999999 runs

Compiler Version v0.5.16+commit.9c3226ce

Other Settings: default evmVersion, None license

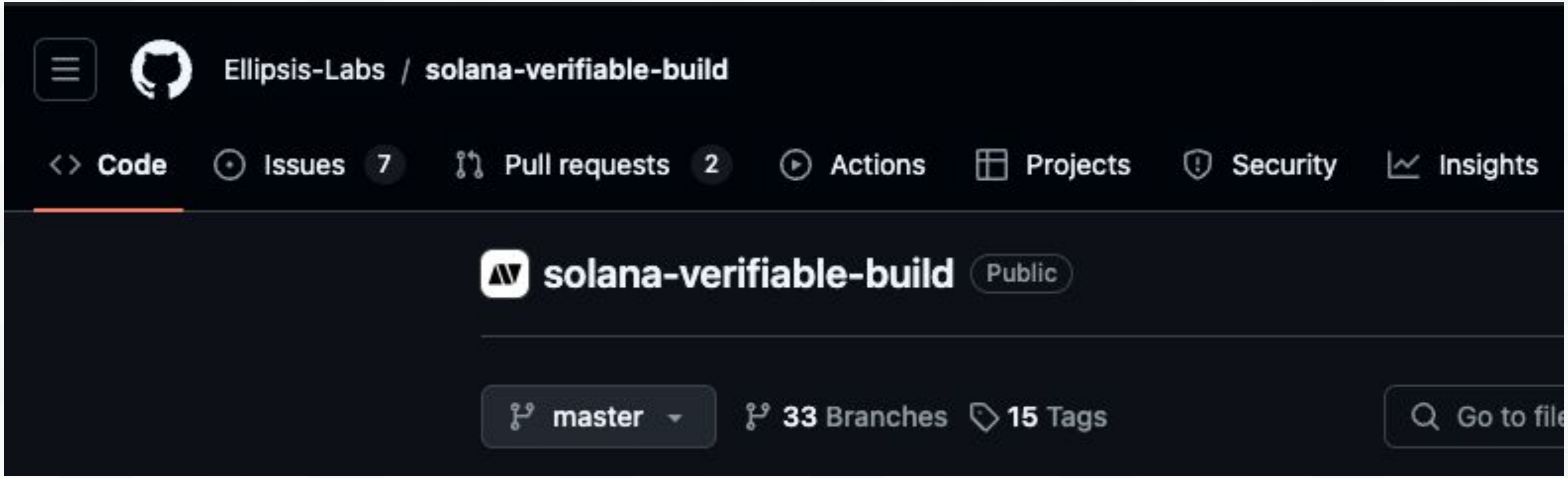
Contract Source Code (Solidity)

```

pragma solidity ^0.5.16;
pragma experimental ABIEncoderV2;

// From https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/math/Math.sol
// Subject to the MIT license.

/**
 * @dev Wrappers over Solidity's arithmetic operations with added overflow
 * checks.
 *
 * Arithmetic operations in Solidity wrap on overflow. This can easily result
 * in bugs, because programmers usually assume that an overflow raises an
 * error, which is the standard behavior in high level programming languages.
 * `SafeMath` restores this intuition by reverting the transaction when an
 * operation overflows.
 *
 * Using this library instead of the unchecked operations eliminates an entire
 * class of bugs, so it's recommended to use it always.
 */
library SafeMath {
    /**
     * @dev Returns the addition of two unsigned integers, reverting on overflow.
     *
     * Counterpart to Solidity's `+` operator.
```

Consequences of using Rust

- Hard to decompile
- Hard to verify bytecode (what was deployed on-chain)
- Issues with language, clean code

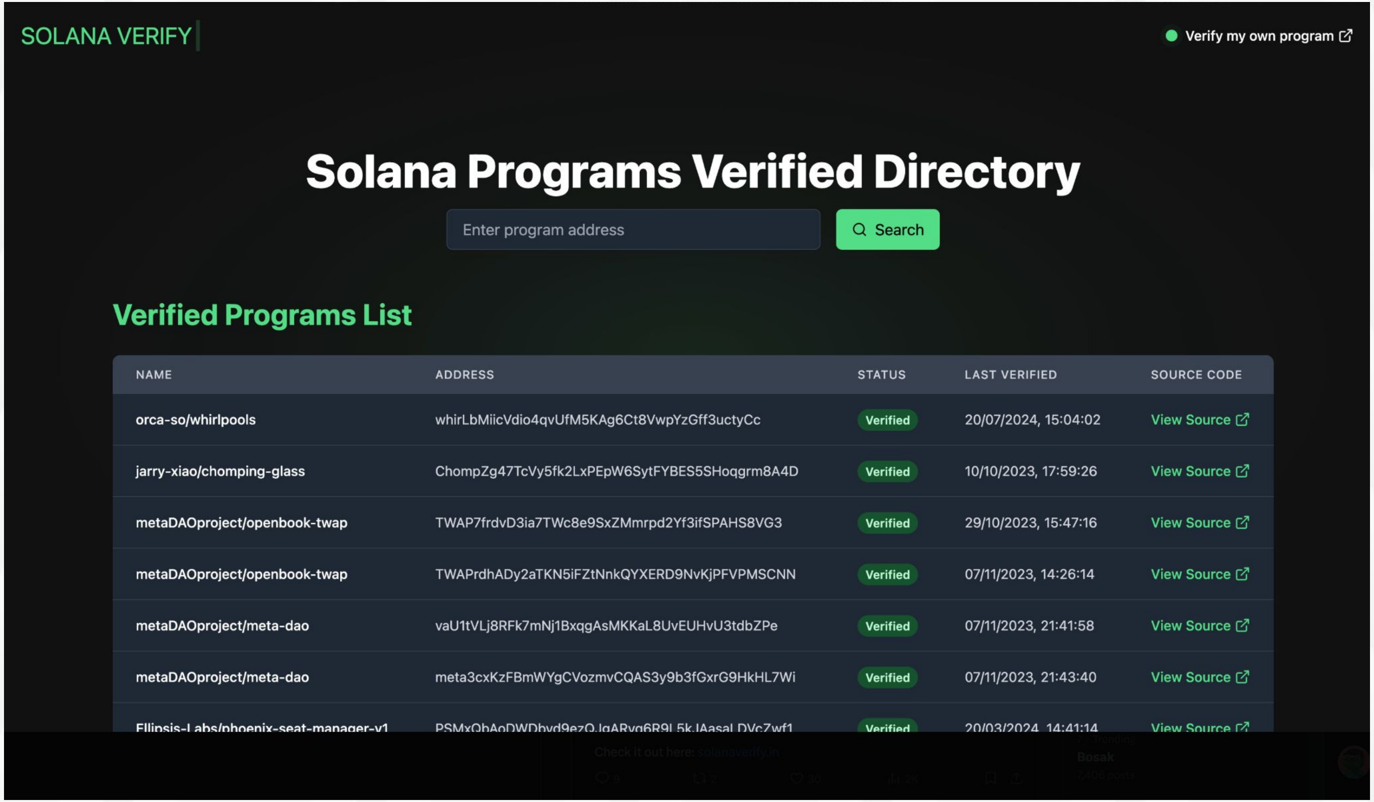
Solution



Ellipsis-Labs / solana-verifiable-build

solana-verifiable-build Public

Solution



SOLANA VERIFY

Solana Programs Verified Directory

Verified Programs List

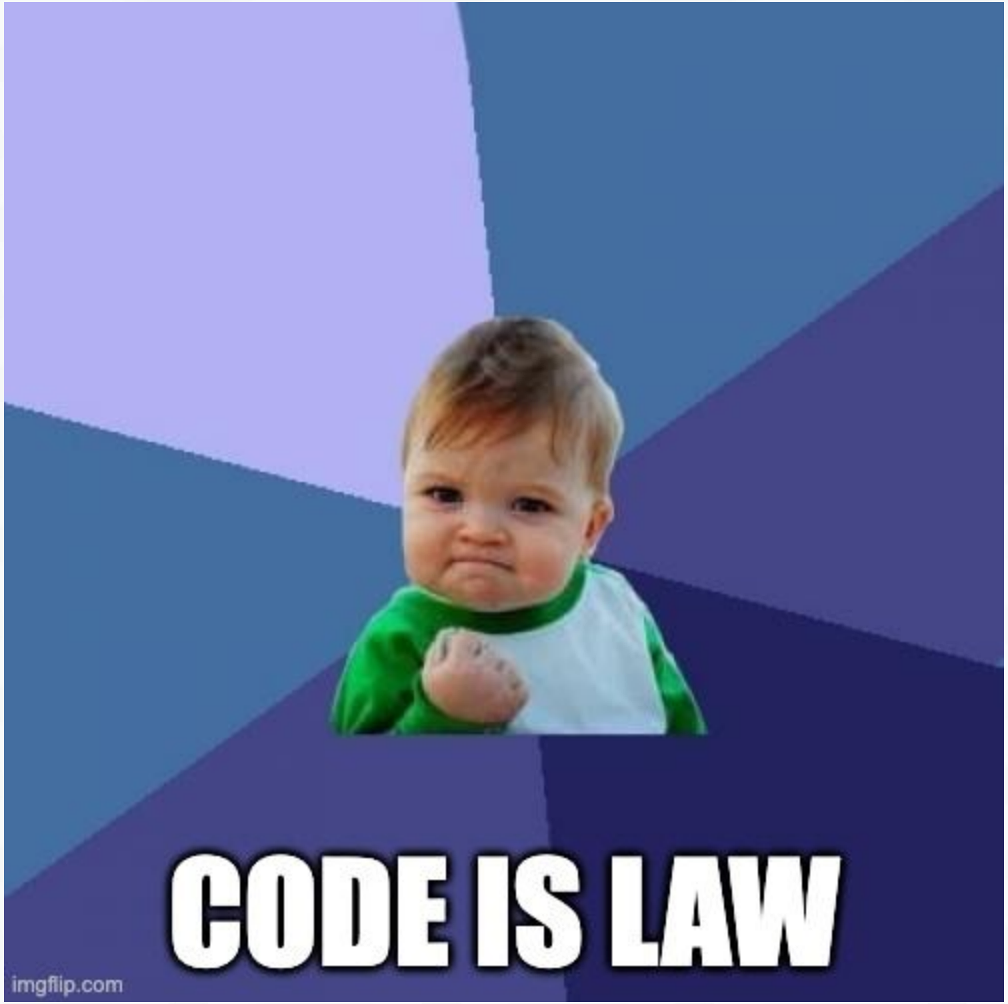
NAME	ADDRESS	STATUS	LAST VERIFIED	SOURCE CODE
orca-so/whirlpools	whirLbMiicVdio4qvUfM5KAg6Ct8VwpYzGff3uctyCc	Verified	20/07/2024, 15:04:02	View Source
jarry-xiao/chomping-glass	ChompZg47TcVy5fk2LxPEpW6SytFYBES5SHoqgrm8A4D	Verified	10/10/2023, 17:59:26	View Source
metaDAOproject/openbook-twap	TWAP7frdvD3ia7TWc8e9SxZMmrpd2Yf3ifSPAHS8VG3	Verified	29/10/2023, 15:47:16	View Source
metaDAOproject/openbook-twap	TWAPrdhADy2aTKN5iFZtNnkQYXERD9NvKjPFVPMSCNN	Verified	07/11/2023, 14:26:14	View Source
metaDAOproject/meta-dao	vaU1tVLj8RFk7mNj1BxqgAsMKKaL8UvEUHvU3tdbZPe	Verified	07/11/2023, 21:41:58	View Source
metaDAOproject/meta-dao	meta3cxKzFBmWYgCVozmvCQAS3y9b3fGxrG9HkHL7Wi	Verified	07/11/2023, 21:43:40	View Source
Ellipsis-Labs/phoenix-seat-manager-v1	PSMxQbAoDWDbvd9ezQJgARyq6R9L5kJAasaLDVcZwf1	Verified	20/03/2024, 14:41:14	View Source

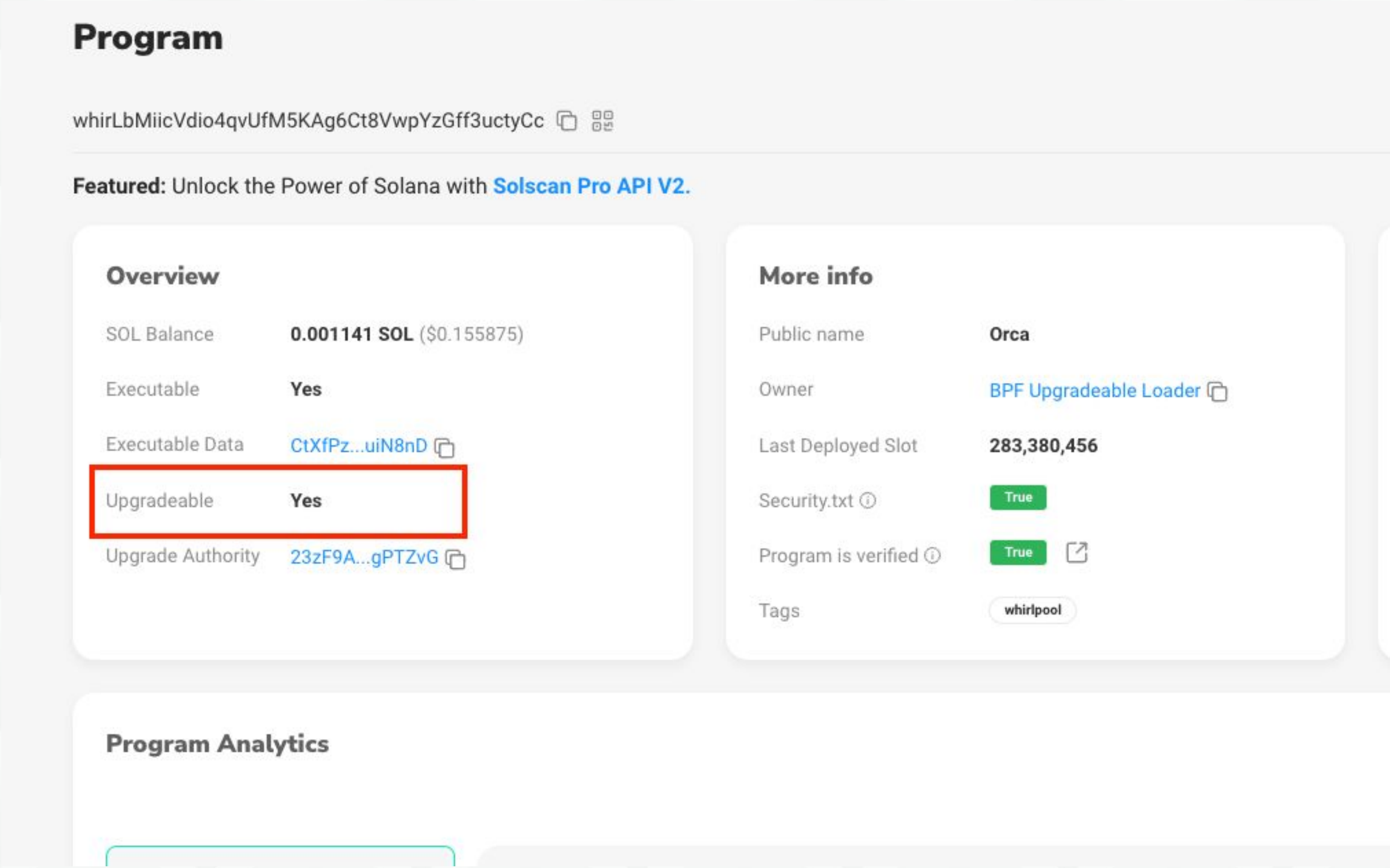
Check it out here: solanaverify.in

Adventure into Solana Security

Scenario 2

1. Program is secure
2. Bytecode is the same as verified
3. Can we trust such protocol?


CODE IS LAW
imgflip.com



Program

whirLbMiicVdio4qvUfM5KAg6Ct8VwpYzGff3uctyCc

Featured: Unlock the Power of Solana with **Solscan Pro API V2.**

Overview

SOL Balance	**0.001141 SOL** ($0.155875)
Executable	**Yes**
Executable Data	CtXfPz...uiN8nD
Upgradeable	**Yes**
Upgrade Authority	23zF9A...gPTZvG

More info

Public name	**Orca**
Owner	BPF Upgradeable Loader
Last Deployed Slot	**283,380,456**
Security.txt ⓘ	True
Program is verified ⓘ	True
Tags	whirlpool

Program Analytics

Upgradable programs

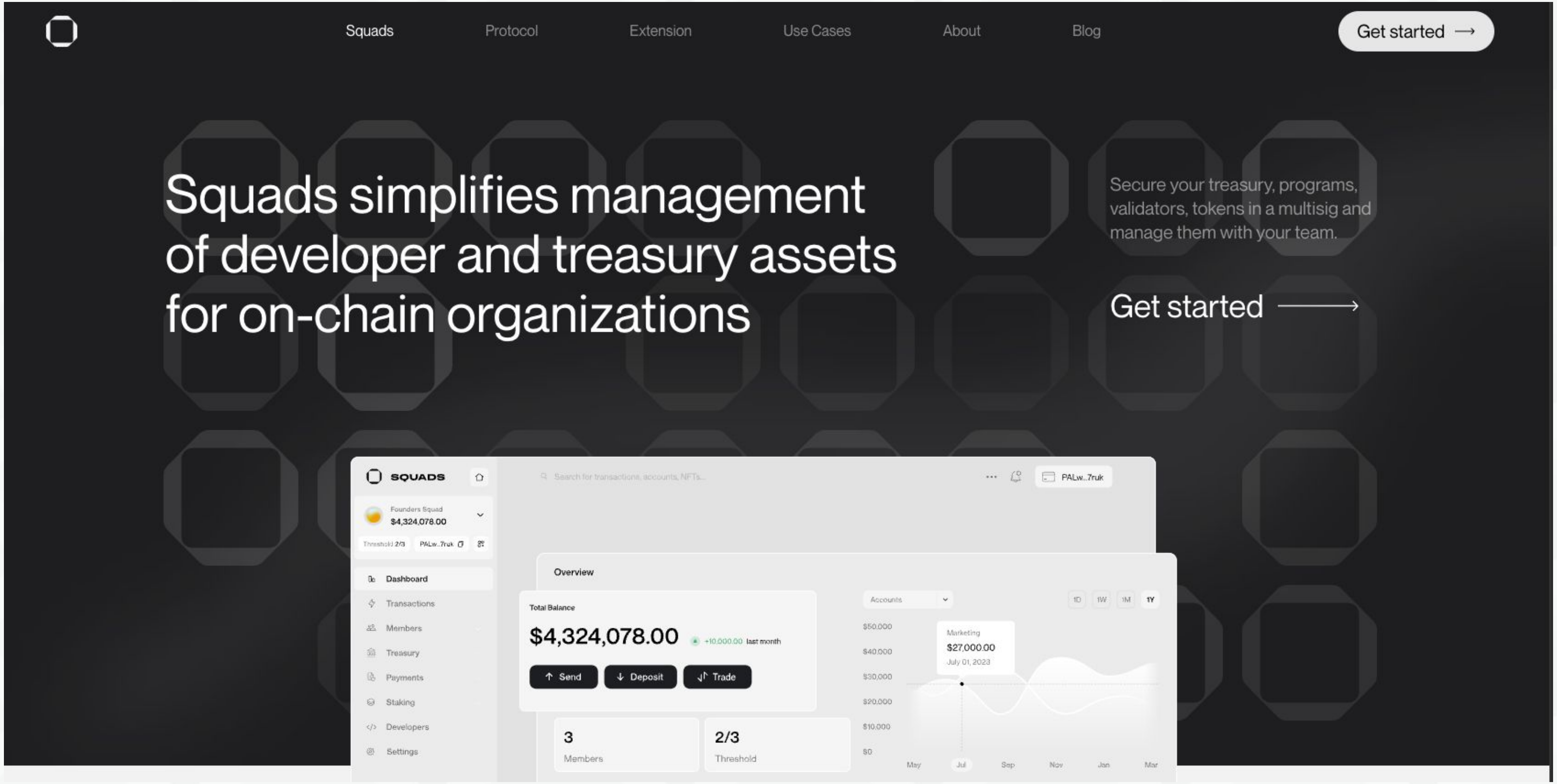
Pros

- bugs can be fixed
- avoid funds locks

Cons

- possible rekt / rug pull
- audits?

Solution: Processes + Multisig

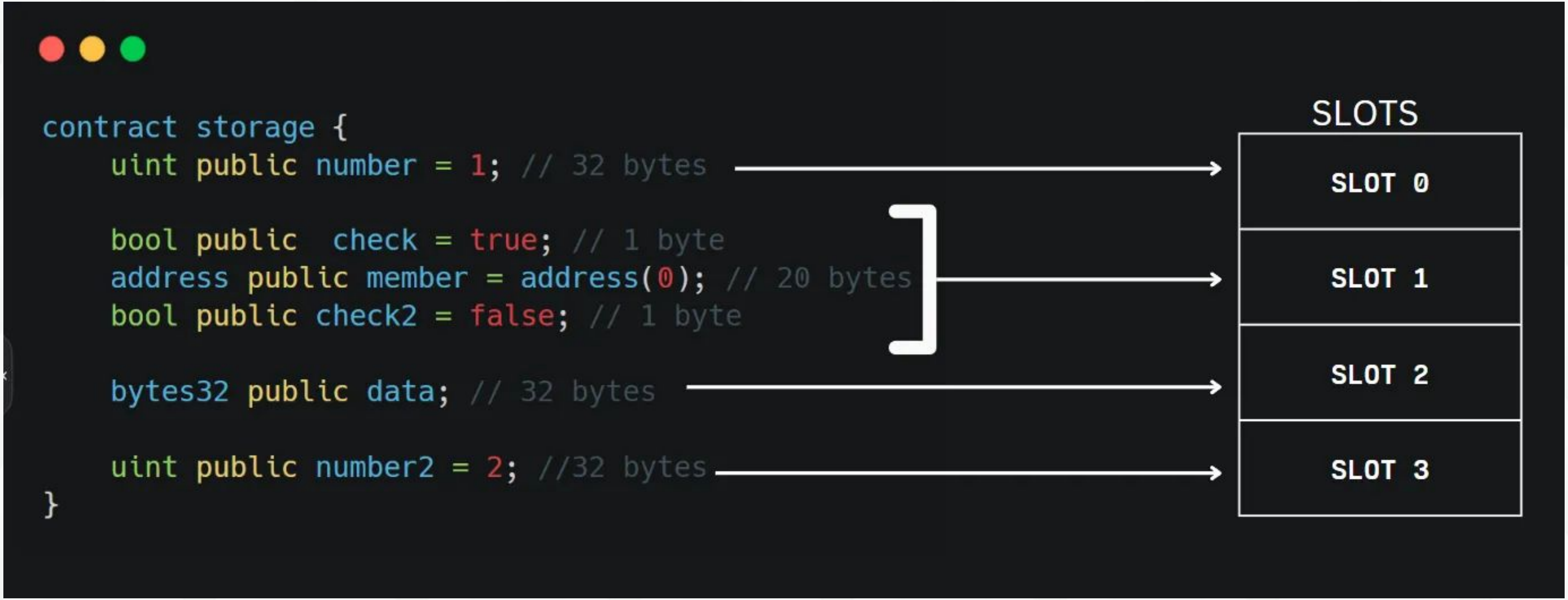


Squads simplifies management of developer and treasury assets for on-chain organizations

Secure your treasury, programs, validators, tokens in a multisig and manage them with your team.

Storage and Access Control

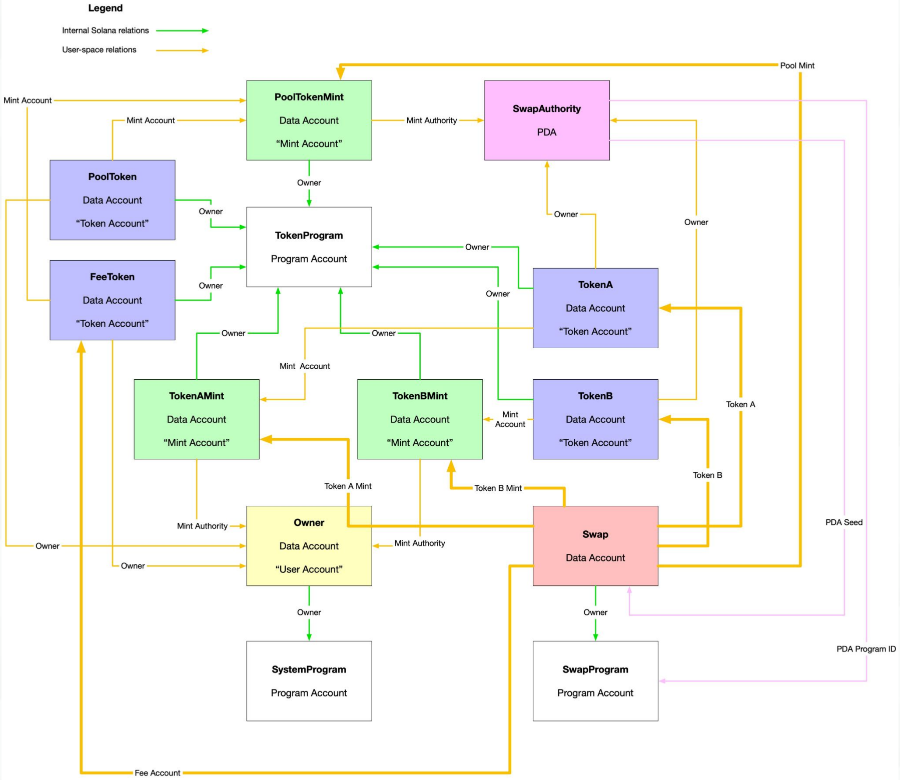
Storage - EVM



```
contract storage {
    uint public number = 1; // 32 bytes

    bool public  check = true; // 1 byte
    address public member = address(0); // 20 bytes
    bool public check2 = false; // 1 byte

    bytes32 public data; // 32 bytes

    uint public number2 = 2; //32 bytes
}
```

Adventure into Solana Security

Storage - Solana

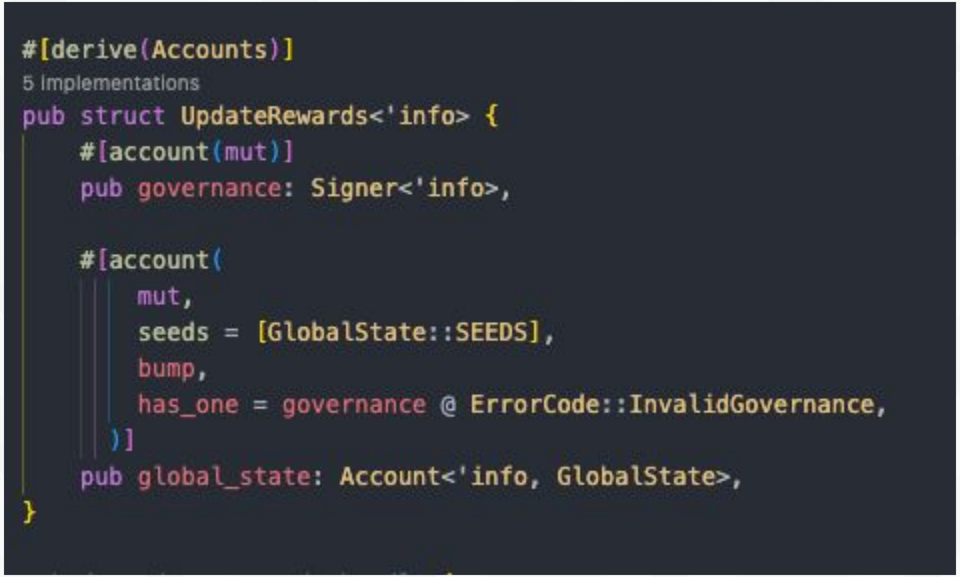


AN ACCOUNT
EVERYTHING IS.
imgflip.com

Solana Account

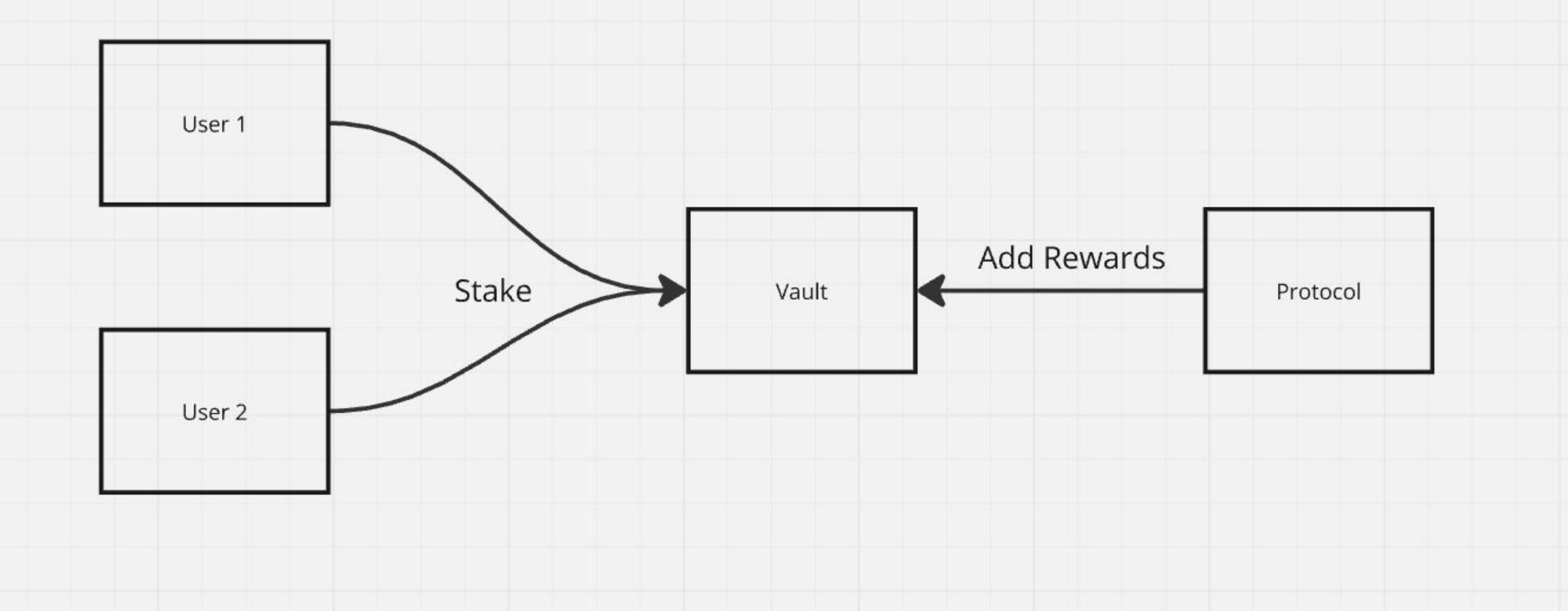
- Designing and documenting programs
- Composability
- Harder to integrate and track data flows
- Access control and ownership

Signer checks

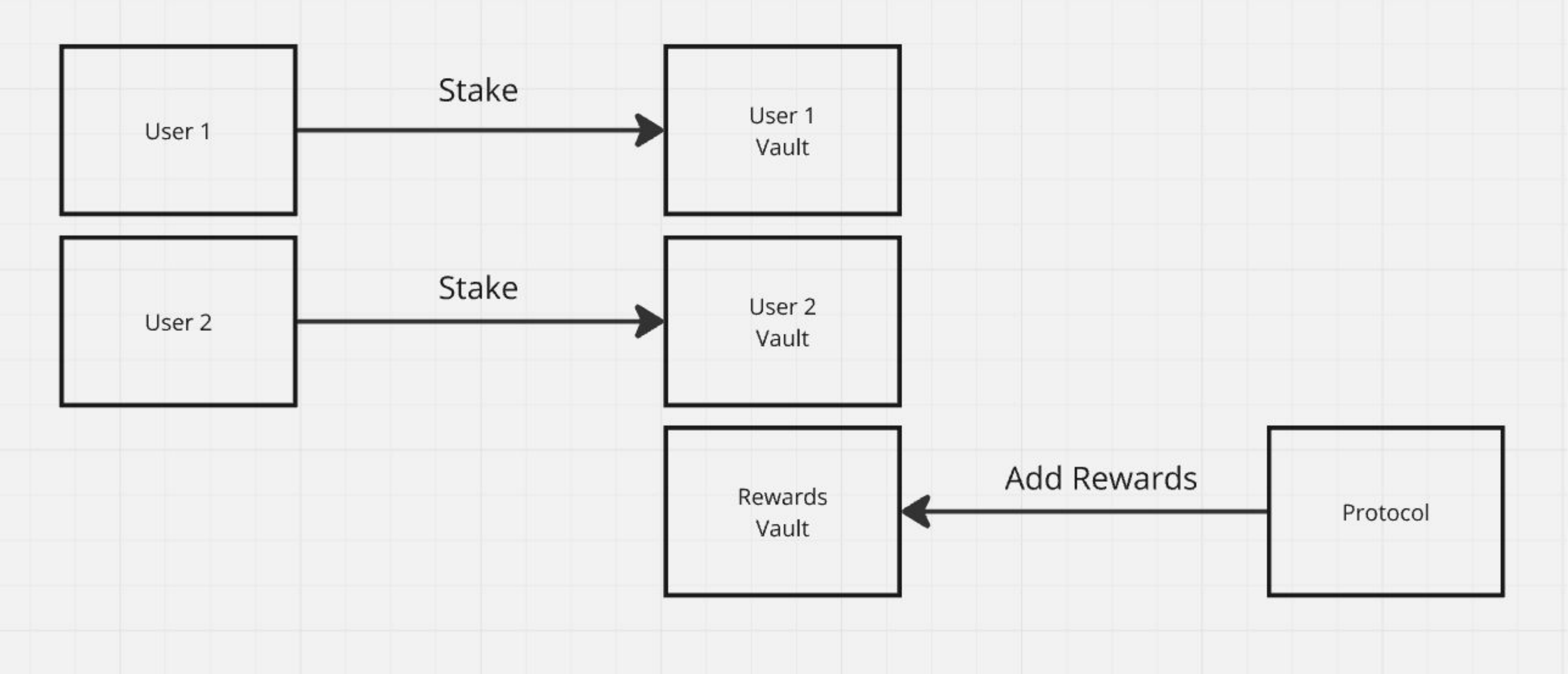


```
#[derive(Accounts)]
5 implementations
pub struct UpdateRewards<'info> {
    #[account(mut)]
    pub governance: Signer<'info>,

    #[account(
        mut,
        seeds = [GlobalState::SEEDS],
        bump,
        has_one = governance @ ErrorCode::InvalidGovernance,
    )]
    pub global_state: Account<'info, GlobalState>,
}
```

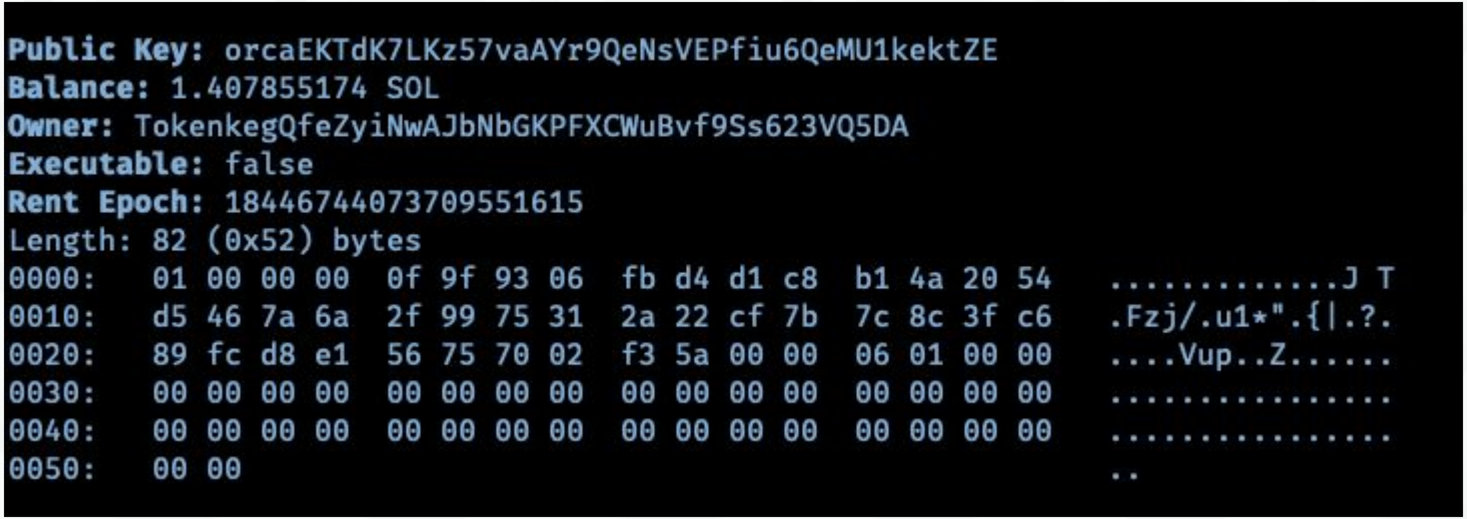
Composability - Staking



Composability - Better Staking



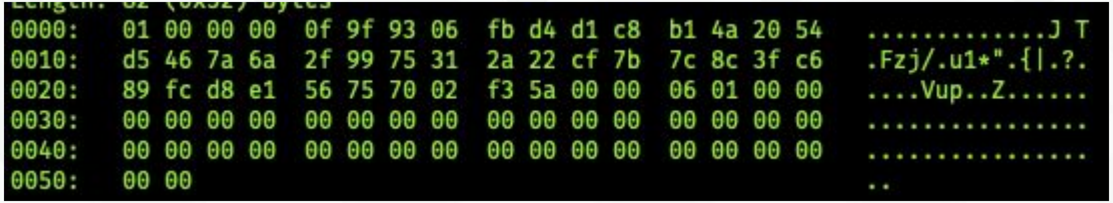
Account Data



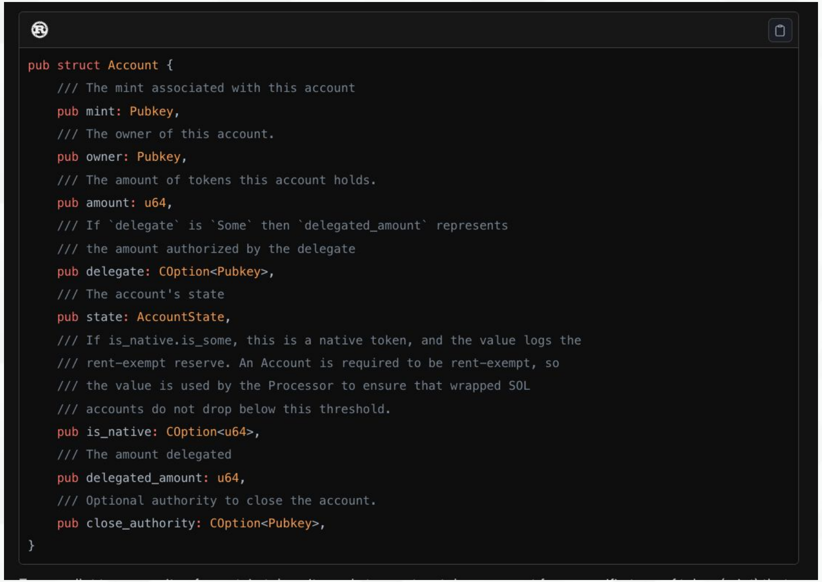
```
Public Key: orcaEKTdK7LKz57vaAYr9QeNsVEPfiu6QeMU1kektZE
Balance: 1.407855174 SOL
Owner: TokenkegQfeZyiNwAJbNbGKPFXCWuBvf9Ss623VQ5DA
Executable: false
Rent Epoch: 18446744073709551615
Length: 82 (0x52) bytes
0000:   01 00 00 00  0f 9f 93 06  fb d4 d1 c8  b1 4a 20 54   .............J T
0010:   d5 46 7a 6a  2f 99 75 31  2a 22 cf 7b  7c 8c 3f c6   .Fzj/.u1*".{|.?.
0020:   89 fc d8 e1  56 75 70 02  f3 5a 00 00  06 01 00 00   ....Vup..Z......
0030:   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00   ................
0040:   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00   ................
0050:   00 00                                                ..
```

Adventure into Solana Security

Account Deserialization



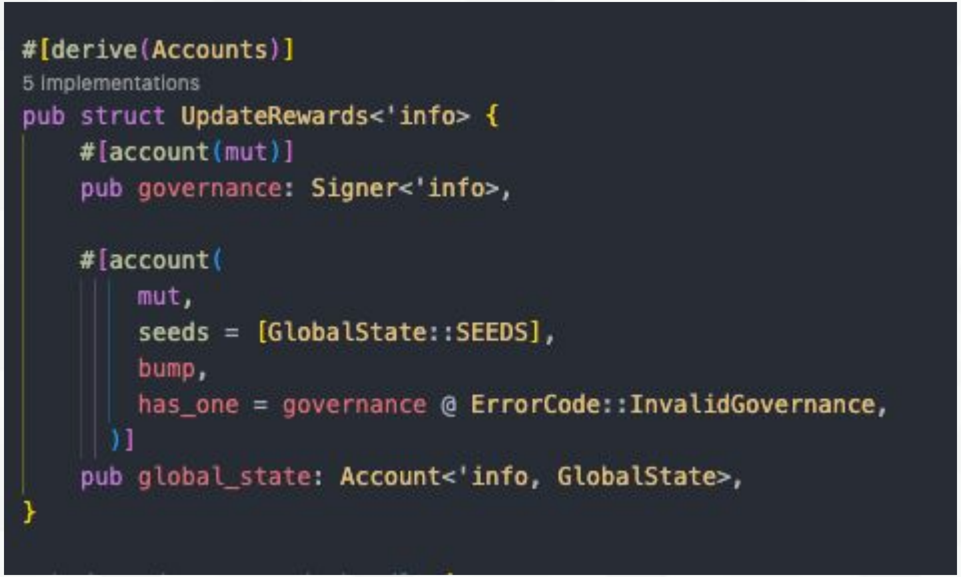
```
Length: 82 (0x52) bytes
0000:   01 00 00 00  0f 9f 93 06  fb d4 d1 c8  b1 4a 20 54   ..............J T
0010:   d5 46 7a 6a  2f 99 75 31  2a 22 cf 7b  7c 8c 3f c6   .Fzj/.u1*".{|.?.
0020:   89 fc d8 e1  56 75 70 02  f3 5a 00 00  06 01 00 00   ....Vup..Z......
0030:   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00   ................
0040:   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00   ................
0050:   00 00                                                ..
```



```rust
pub struct Account {
    /// The mint associated with this account
    pub mint: Pubkey,
    /// The owner of this account.
    pub owner: Pubkey,
    /// The amount of tokens this account holds.
    pub amount: u64,
    /// If `delegate` is `Some` then `delegated_amount` represents
    /// the amount authorized by the delegate
    pub delegate: COption<Pubkey>,
    /// The account's state
    pub state: AccountState,
    /// If is_native.is_some, this is a native token, and the value logs the
    /// rent-exempt reserve. An Account is required to be rent-exempt, so
    /// the value is used by the Processor to ensure that wrapped SOL
    /// accounts do not drop below this threshold.
    pub is_native: COption<u64>,
    /// The amount delegated
    pub delegated_amount: u64,
    /// Optional authority to close the account.
    pub close_authority: COption<Pubkey>,
}
```

Account Deserialization

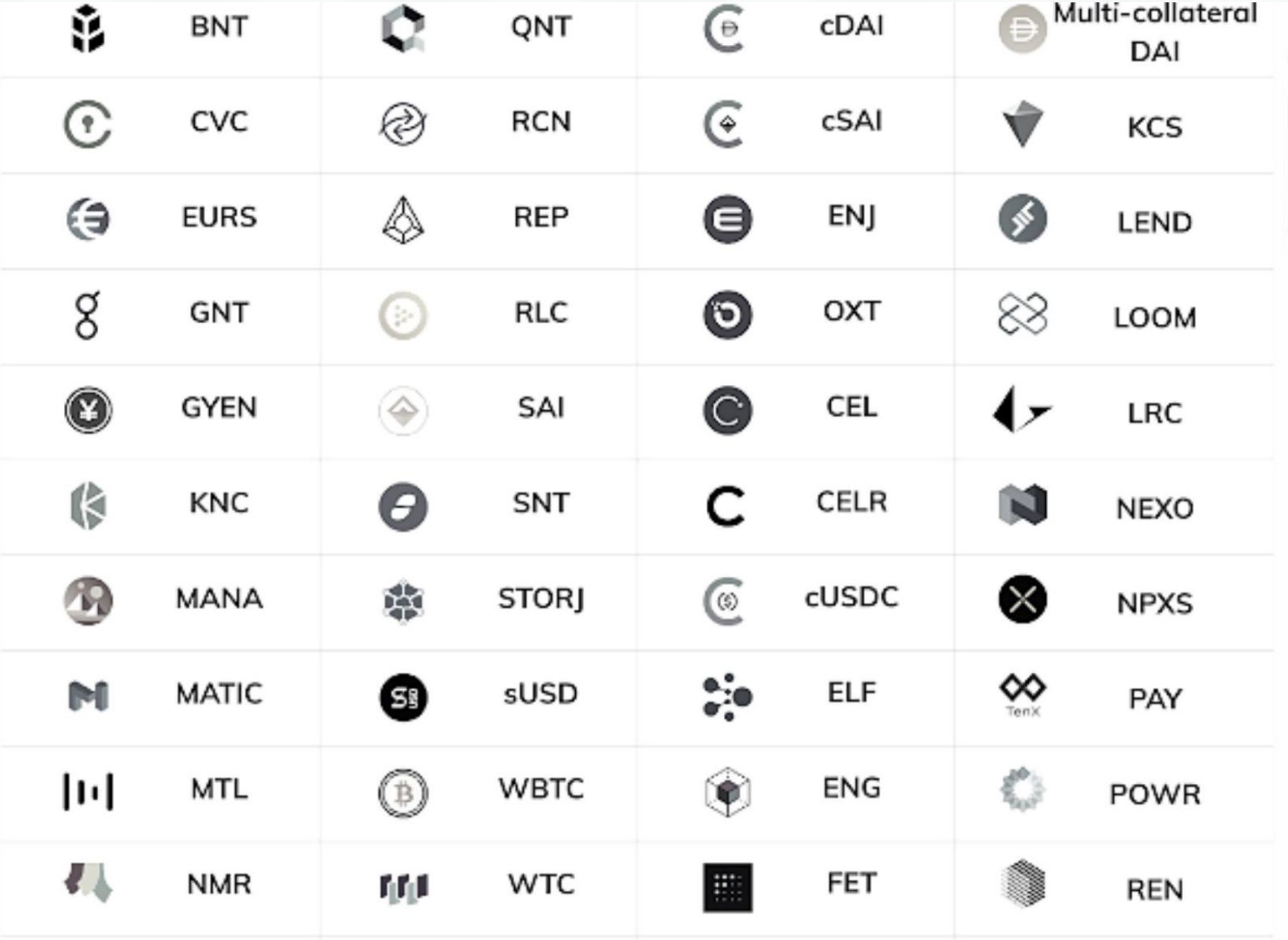
- Check data type
- Check ownership
- or use Anchor



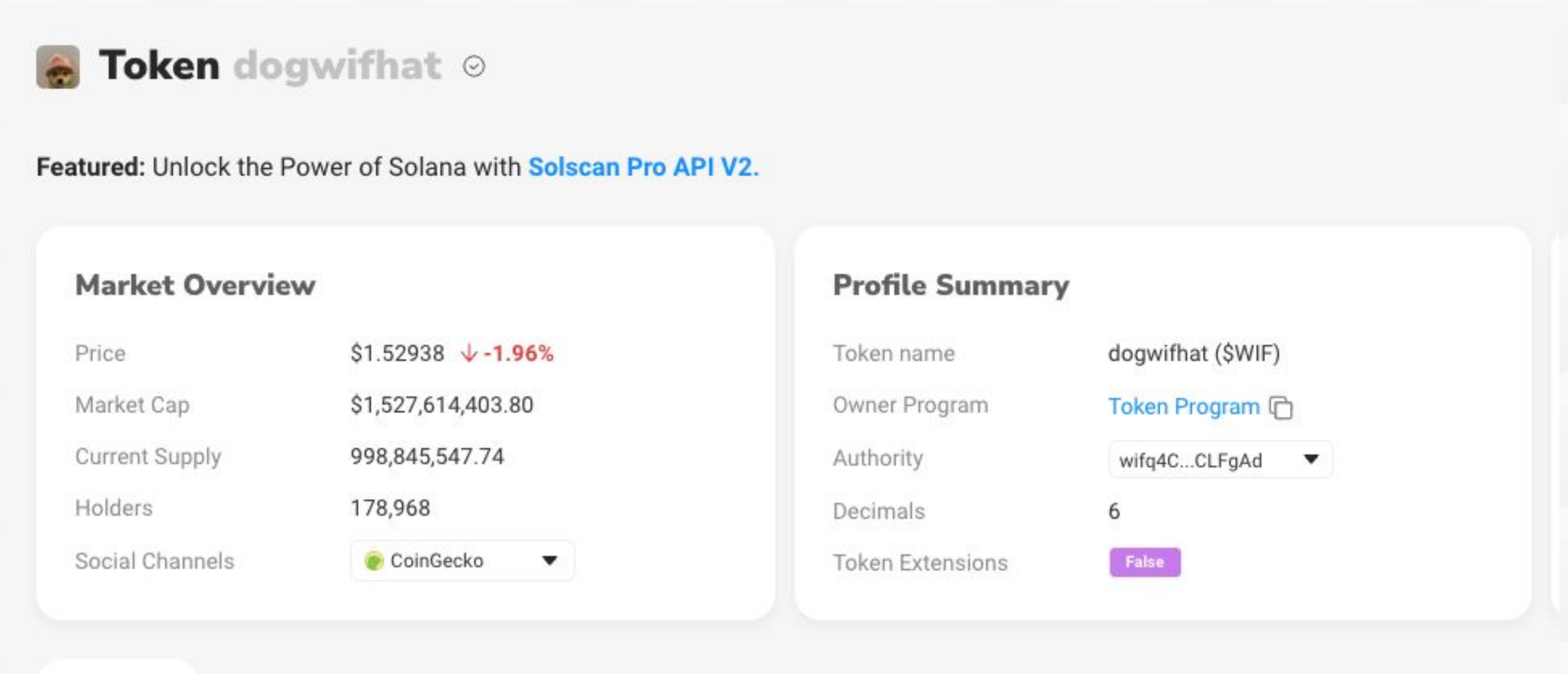
```
#[derive(Accounts)]
5 implementations
pub struct UpdateRewards<'info> {
    #[account(mut)]
    pub governance: Signer<'info>,

    #[account(
        mut,
        seeds = [GlobalState::SEEDS],
        bump,
        has_one = governance @ ErrorCode::InvalidGovernance,
    )]
    pub global_state: Account<'info, GlobalState>,
}
```

Integrations

ERC-20



BNT	QNT	cDAI	Multi-collateral DAI
CVC	RCN	cSAI	KCS
EURS	REP	ENJ	LEND
GNT	RLC	OXT	LOOM
GYEN	SAI	CEL	LRC
KNC	SNT	CELR	NEXO
MANA	STORJ	cUSDC	NPXS
MATIC	sUSD	ELF	PAY
MTL	WBTC	ENG	POWR
NMR	WTC	FET	REN



Token dogwifhat

Featured: Unlock the Power of Solana with Solscan Pro API V2.

Market Overview

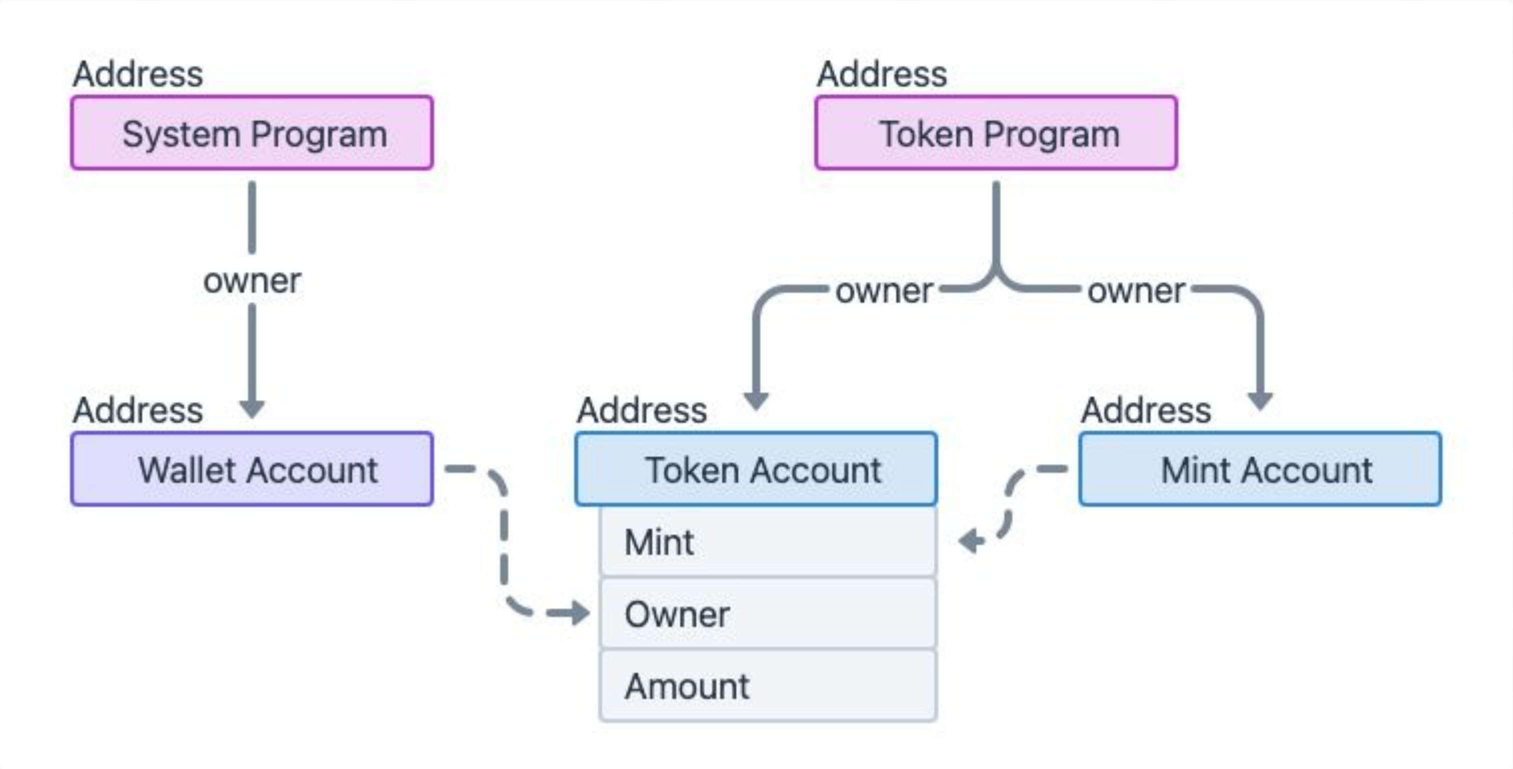
Price	$1.52938 ↓ -1.96%
Market Cap	$1,527,614,403.80
Current Supply	998,845,547.74
Holders	178,968
Social Channels	CoinGecko ▼

Profile Summary

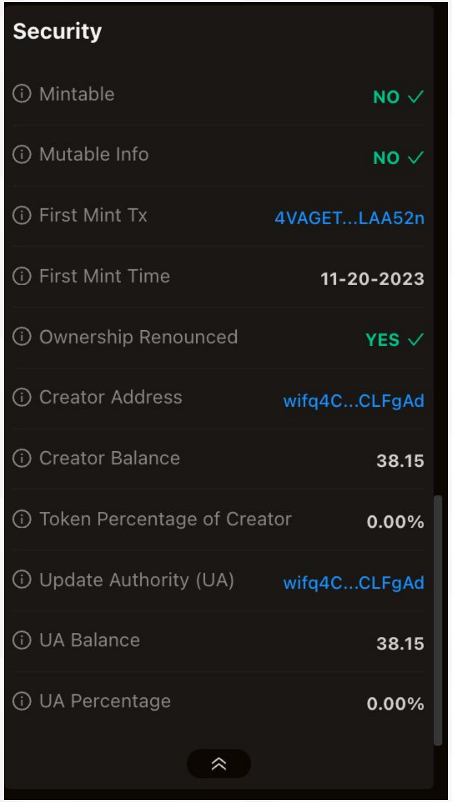
Token name	dogwifhat ($WIF)
Owner Program	Token Program
Authority	wifq4C...CLFgAd ▼
Decimals	6
Token Extensions	False

Adventure into Solana Security

Token Program



Is Token OK?

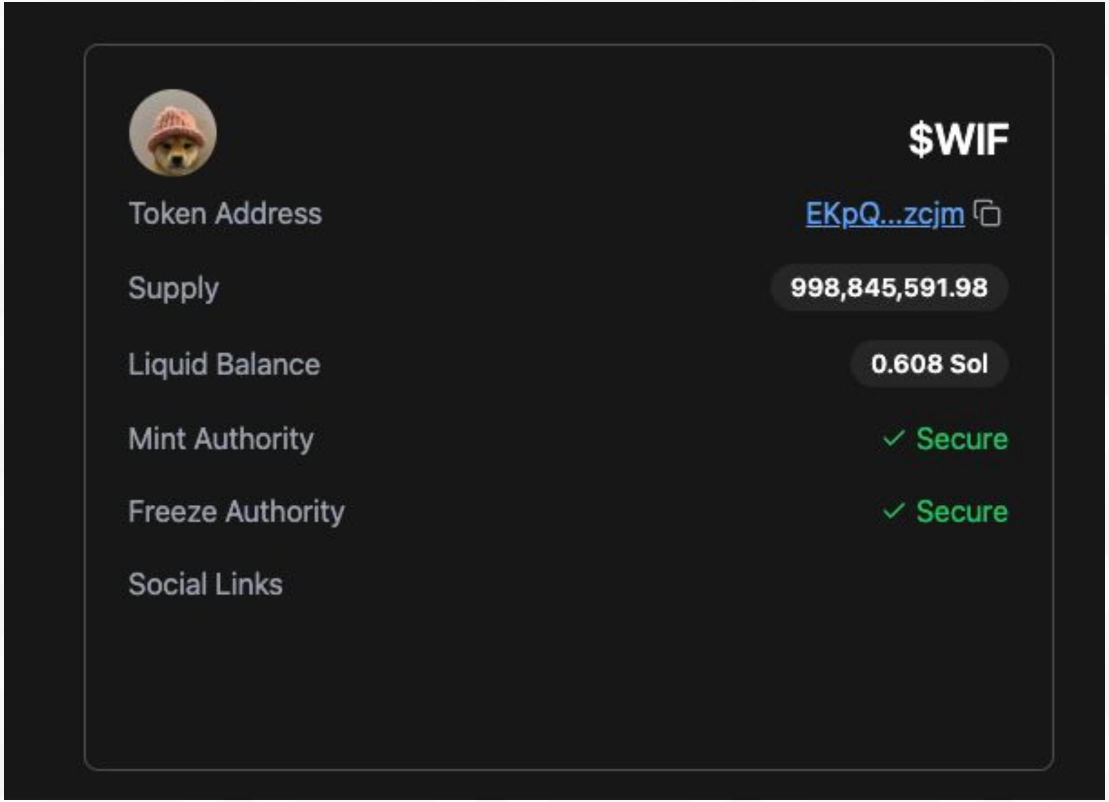


Security

Mintable	NO ✓
Mutable Info	NO ✓
First Mint Tx	4VAGET...LAA52n
First Mint Time	11-20-2023
Ownership Renounced	YES ✓
Creator Address	wifq4C...CLFgAd
Creator Balance	38.15
Token Percentage of Creator	0.00%
Update Authority (UA)	wifq4C...CLFgAd
UA Balance	38.15
UA Percentage	0.00%

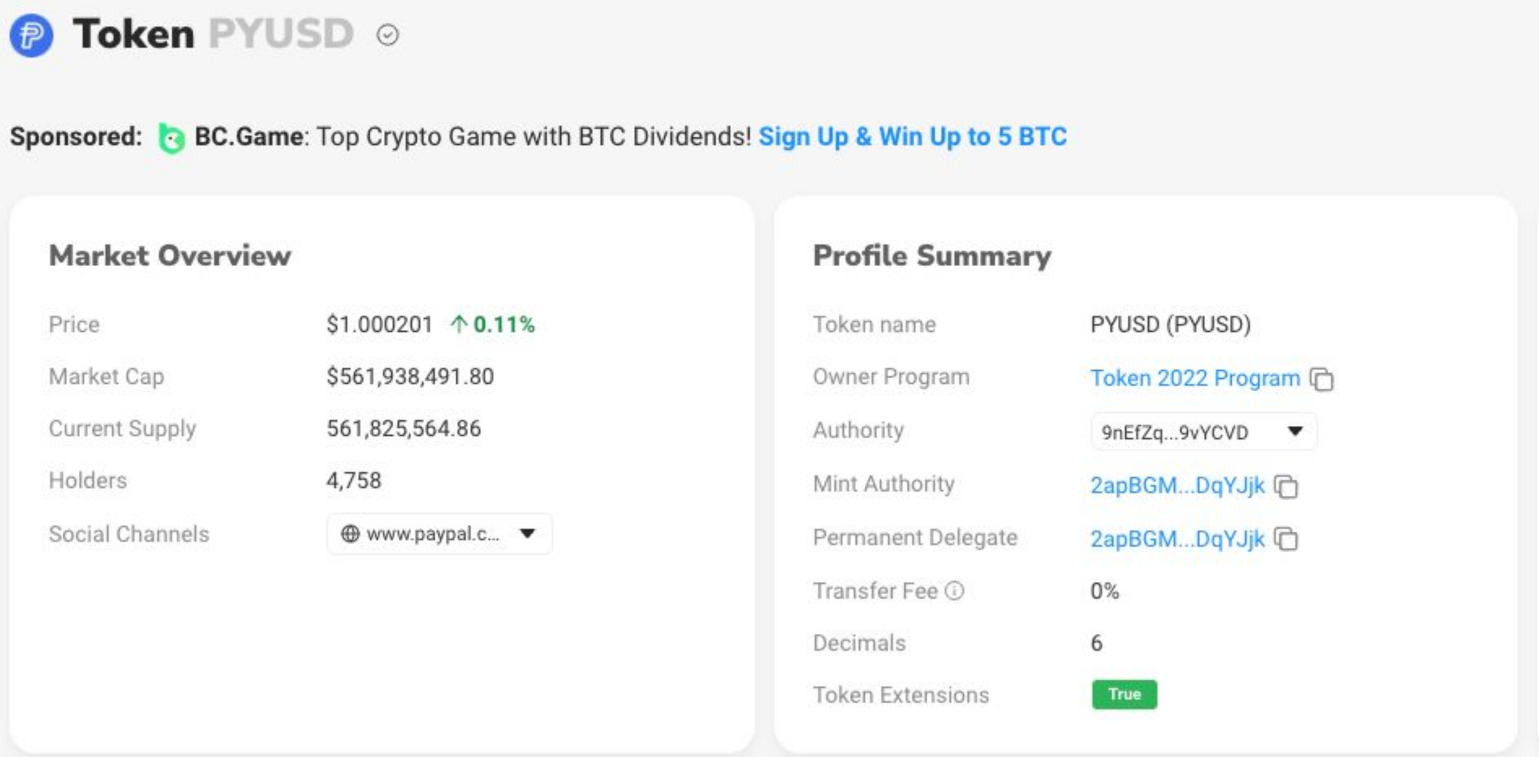
- can new tokens be minted?
- can token change its name?
- when it was minted?
- does token have extensions?

Freeze Authority



	$WIF
Token Address	EKpQ...zcjm
Supply	998,845,591.98
Liquid Balance	0.608 Sol
Mint Authority	✓ Secure
Freeze Authority	✓ Secure
Social Links	

Extensions



Token PYUSD

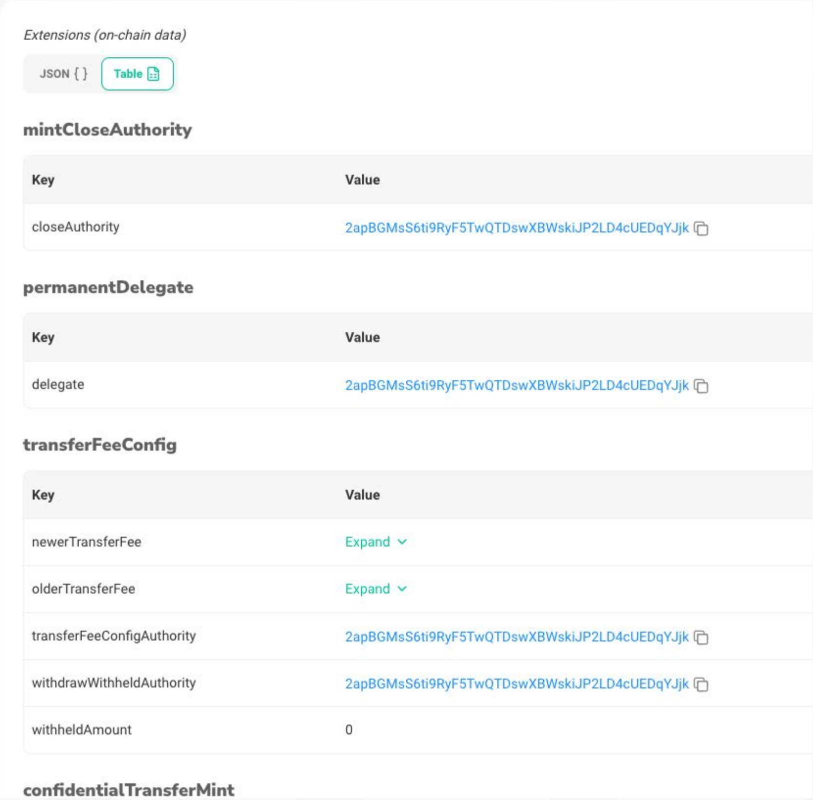
Market Overview

Price	$1.000201 ↑0.11%
Market Cap	$561,938,491.80
Current Supply	561,825,564.86
Holders	4,758
Social Channels	www.paypal.c... ▼

Profile Summary

Token name	PYUSD (PYUSD)
Owner Program	Token 2022 Program
Authority	9nEfZq...9vYCVD ▼
Mint Authority	2apBGM...DqYJjk
Permanent Delegate	2apBGM...DqYJjk
Transfer Fee ⓘ	0%
Decimals	6
Token Extensions	True

Extensions



Extensions (on-chain data)

mintCloseAuthority

Key	Value
closeAuthority	2apBGMsS6ti9RyF5TwQTDswXBWskiJP2LD4cUEDqYJjk

permanentDelegate

Key	Value
delegate	2apBGMsS6ti9RyF5TwQTDswXBWskiJP2LD4cUEDqYJjk

transferFeeConfig

Key	Value
newerTransferFee	Expand ⌄
olderTransferFee	Expand ⌄
transferFeeConfigAuthority	2apBGMsS6ti9RyF5TwQTDswXBWskiJP2LD4cUEDqYJjk
withdrawWithheldAuthority	2apBGMsS6ti9RyF5TwQTDswXBWskiJP2LD4cUEDqYJjk
withheldAmount	0

confidentialTransferMint

Integrations

- Token Program
- Metaplex
- Oracles
- Randomness
- Other protocols

Non-blockchain Threats

Adventure into Solana Security

Non-blockchain Threats

- Website hacked
- Social media hacked
- Key leaked
- Insider threat

Adventure into Solana Security

Non-blockchain Threats

- Website hacked - **pentest, aware devs**
- Social media hacked - **2fa**
- Key leaked - **multisig**
- Insider threat - **least privilege**

Summary

Code Validation

Verify if contract was audited and that the correct version is deployed

Team Validation

Ensure that team has proper procedures - program upgrades, management

Access Control

Understand access control in Solana (account), design it well. Check for signers and ownership.

Integrations

Master protocol that you are integrating with, either if it is token, oracle or DEX.

Contact us

torii.team

@toriisecurity

TORII