京麒
网络安全大会

2023
CTFCON

Solana Security 101

Q7

Security Researcher at Sec3

About Sec3

- The Sec3 audit team combines industry leading security professionals, CS professors, as well as exceptional whitehats
- Top CTF competitors — six-time DEF CON CTF finalist and auditors from elite teams around the world
- Team members frequently invited to present at prestigious industrial conferences such as Black Hat, DEF CON, and Pwn2Own
- www.sec3.dev

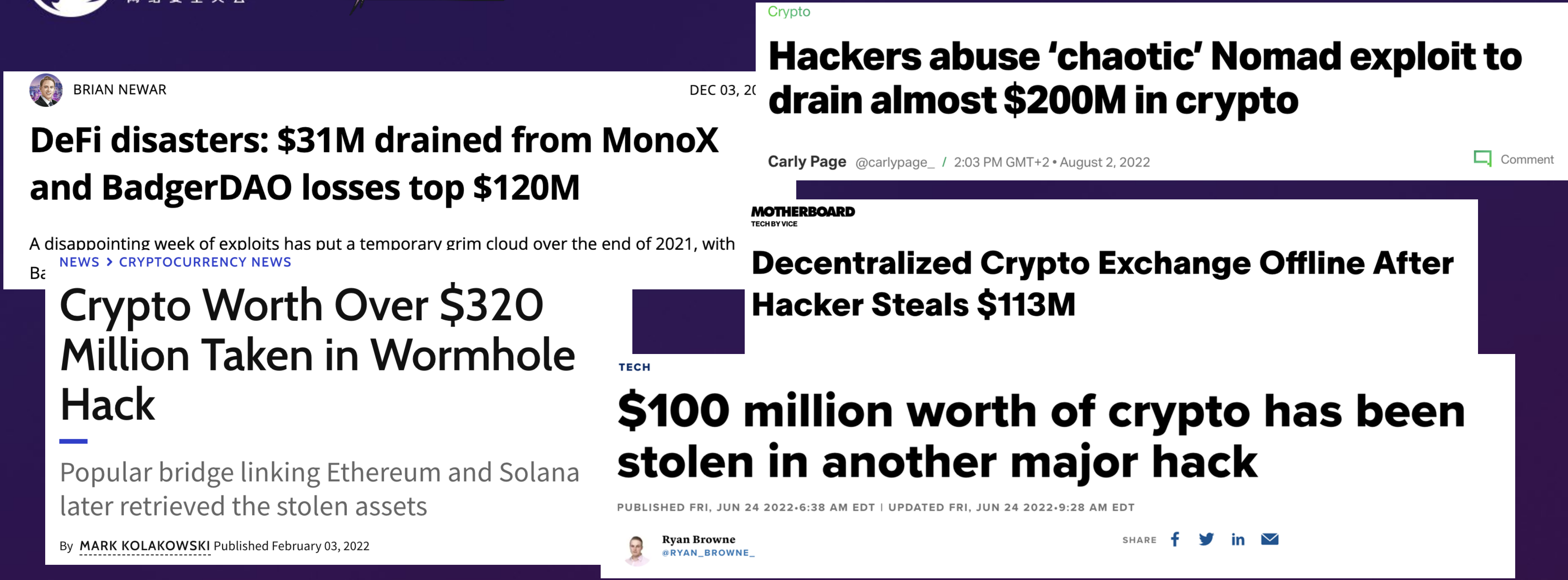
Content

- **Intro**
- Solana Basics
- Our CTF Challenges
- Web 2.5 Security

Why Security?

京麒 网络安全大会

CTFCON 2023



BRIAN NEWAR DEC 03, 20

DeFi disasters: $31M drained from MonoX and BadgerDAO losses top $120M

A disappointing week of exploits has put a temporary grim cloud over the end of 2021, with

Ba

Crypto

Hackers abuse 'chaotic' Nomad exploit to drain almost $200M in crypto

Carly Page @carlypage_ / 2:03 PM GMT+2 • August 2, 2022

Comment

NEWS > CRYPTOCURRENCY NEWS

Crypto Worth Over $320 Million Taken in Wormhole Hack

Popular bridge linking Ethereum and Solana later retrieved the stolen assets

By MARK KOLAKOWSKI Published February 03, 2022

MOTHERBOARD

TECH BY VICE

Decentralized Crypto Exchange Offline After Hacker Steals $113M

TECH

$100 million worth of crypto has been stolen in another major hack

PUBLISHED FRI, JUN 24 2022•6:38 AM EDT | UPDATED FRI, JUN 24 2022•9:28 AM EDT

Ryan Browne
@RYAN_BROWNE_

SHARE

According to DefiLlama, total value hacked this year is ~1.3 billion

Why I Prefer Solana?

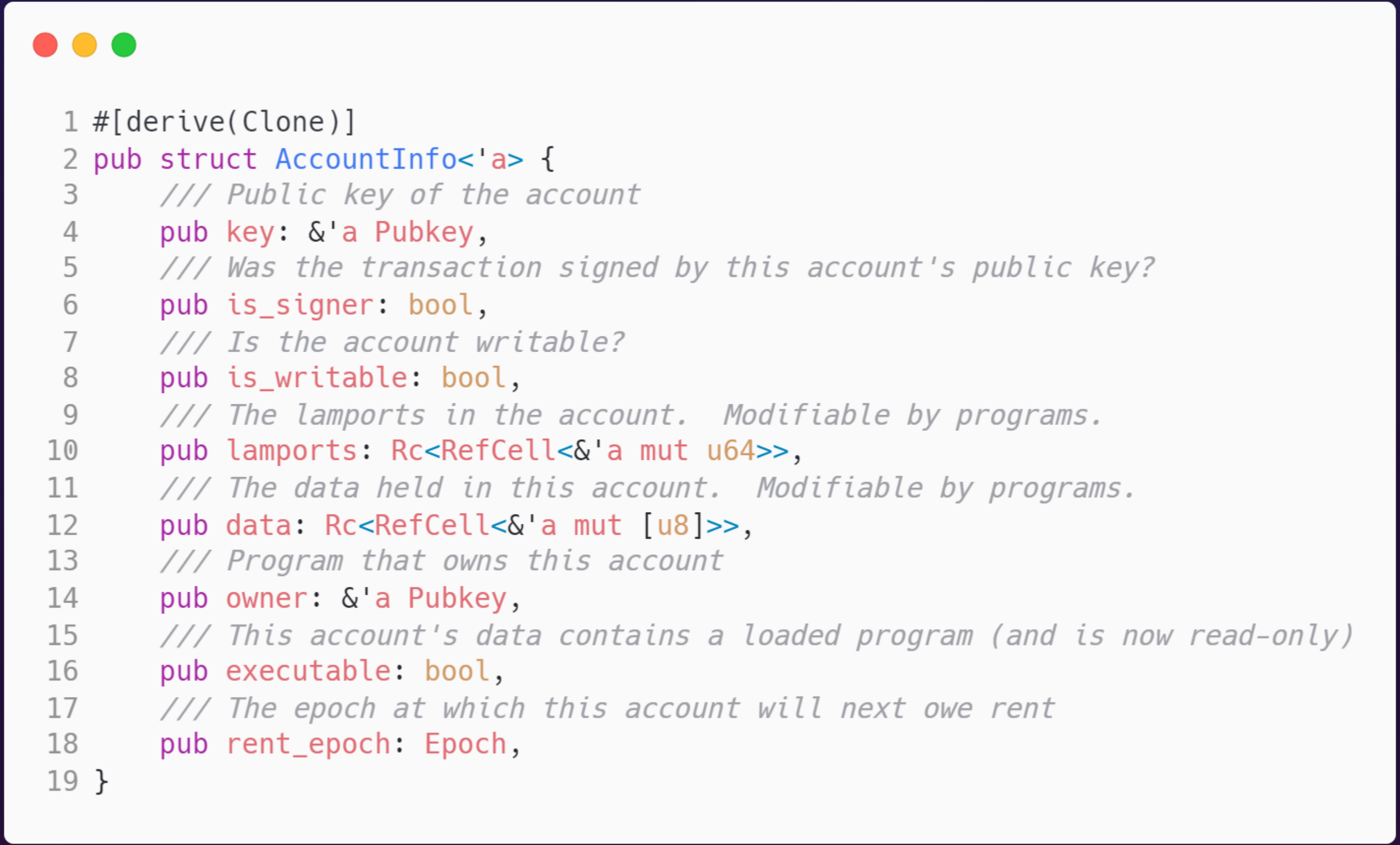
Why I Prefer Solana?

- No “gas stress”
- Super fast
- Safer smart contracts (written in Rust)
- Code and data decoupling

Content

- Intro
- **Solana Basics**
- Our CTF Challenges
- Web 2.5 Security

Accounts

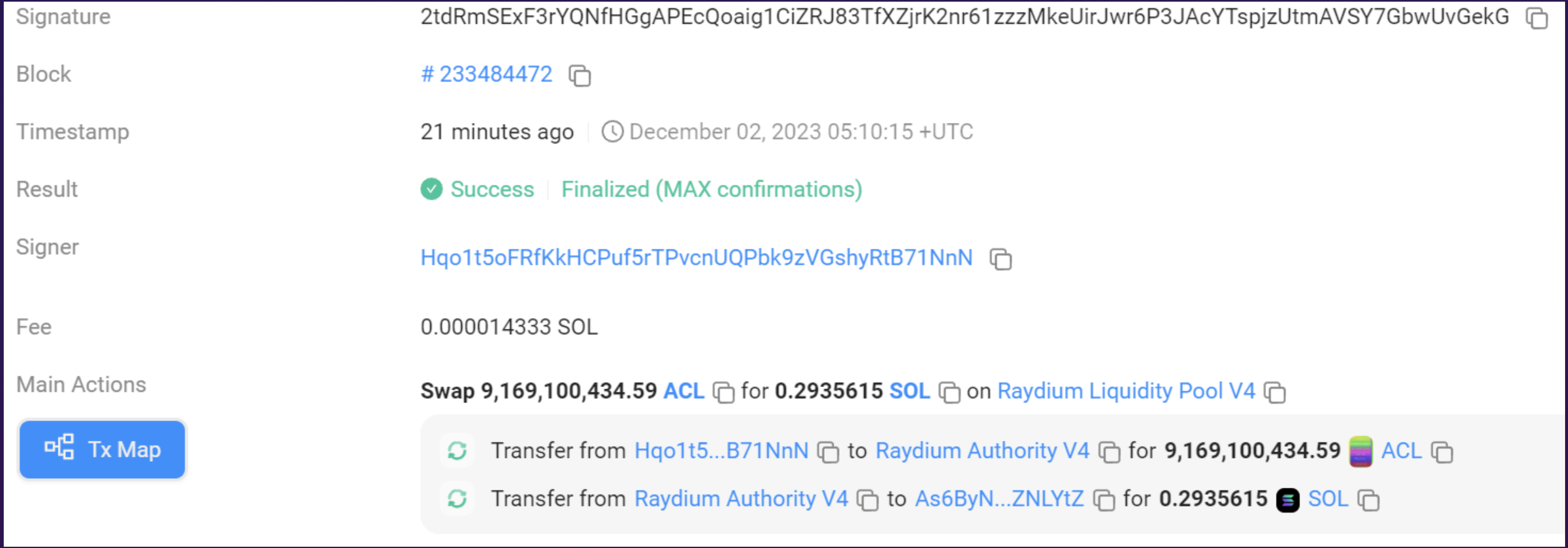


```rust
#[derive(Clone)]
pub struct AccountInfo<'a> {
    /// Public key of the account
    pub key: &'a Pubkey,
    /// Was the transaction signed by this account's public key?
    pub is_signer: bool,
    /// Is the account writable?
    pub is_writable: bool,
    /// The lamports in the account.  Modifiable by programs.
    pub lamports: Rc<RefCell<&'a mut u64>>,
    /// The data held in this account.  Modifiable by programs.
    pub data: Rc<RefCell<&'a mut [u8]>>,
    /// Program that owns this account
    pub owner: &'a Pubkey,
    /// This account's data contains a loaded program (and is now read-only)
    pub executable: bool,
    /// The epoch at which this account will next owe rent
    pub rent_epoch: Epoch,
}
```

Program Derived Addresses (PDAs)

- PDAs are 32-byte strings that look like public keys, but don't have corresponding private keys
- findProgramAddress will deterministically derive a PDA from a programId and seeds (collection of bytes)
- A bump (one byte) is used to push a potential PDA off the ed25519 elliptic curve
- Programs can sign for their PDAs by providing the seeds and bump to invoke_signed
- A PDA can only be signed by the program from which it was derived

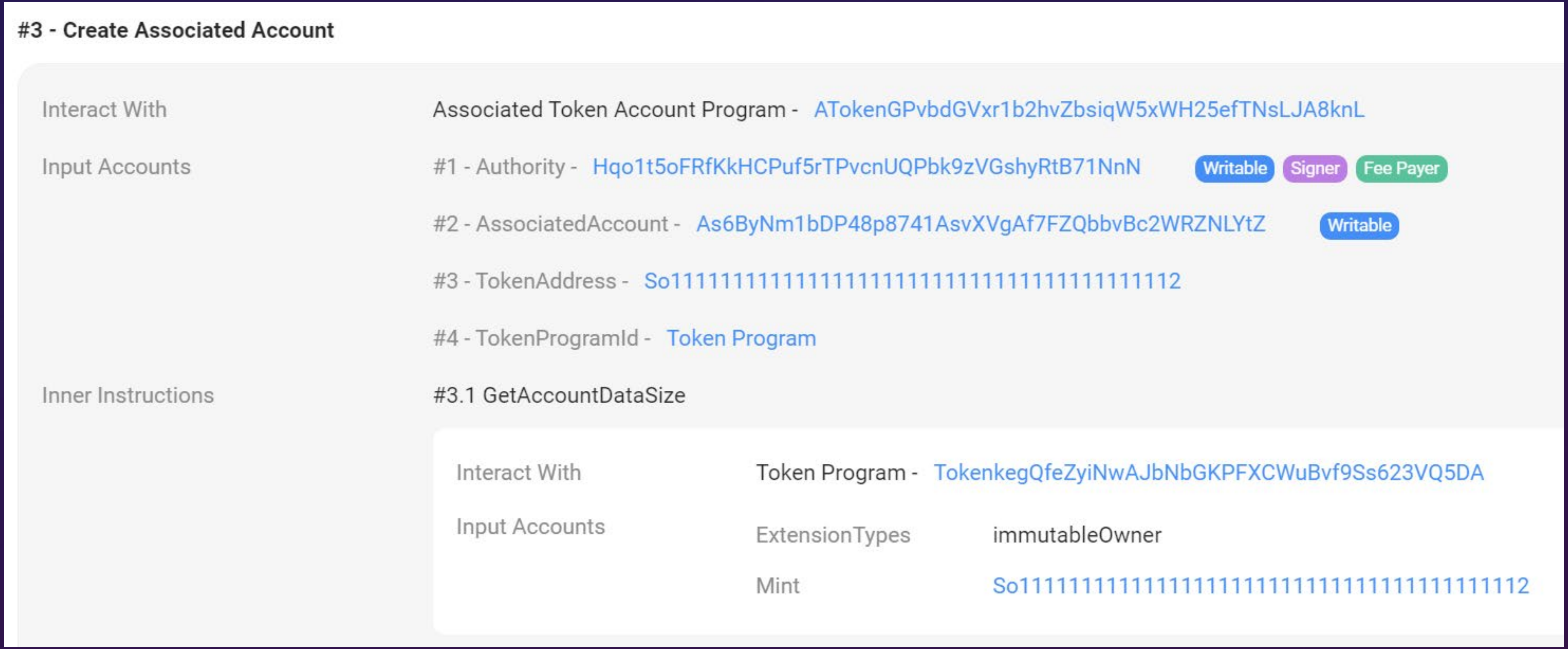
Transactions



Signature	2tdRmSExF3rYQNfHGgAPEcQoaig1CiZRJ83TfXZjrK2nr61zzzMkeUirJwr6P3JAcYTspjzUtmAVSY7GbwUvGekG
Block	# 233484472
Timestamp	21 minutes ago | December 02, 2023 05:10:15 +UTC
Result	✓ Success | Finalized (MAX confirmations)
Signer	Hqo1t5oFRfKkHCPuf5rTPvcnUQPbk9zVGshyRtB71NnN
Fee	0.000014333 SOL
Main Actions	**Swap 9,169,100,434.59** ACL for **0.2935615** SOL on Raydium Liquidity Pool V4

Transfer from Hqo1t5...B71NnN to Raydium Authority V4 for **9,169,100,434.59** ACL

Transfer from Raydium Authority V4 to As6ByN...ZNLYtZ for **0.2935615** SOL



#3 - Create Associated Account

Interact With — Associated Token Account Program - ATokenGPvbdGVxr1b2hvZbsiqW5xWH25efTNsLJA8knL

Input Accounts

- #1 - Authority - Hqo1t5oFRfKkHCPuf5rTPvcnUQPbk9zVGshyRtB71NnN — Writable Signer Fee Payer
- #2 - AssociatedAccount - As6ByNm1bDP48p8741AsvXVgAf7FZQbbvBc2WRZNLYtZ — Writable
- #3 - TokenAddress - So11111111111111111111111111111111111111112
- #4 - TokenProgramId - Token Program

Inner Instructions — #3.1 GetAccountDataSize

Interact With — Token Program - TokenkegQfeZyiNwAJbNbGKPFXCWuBvf9Ss623VQ5DA

Input Accounts

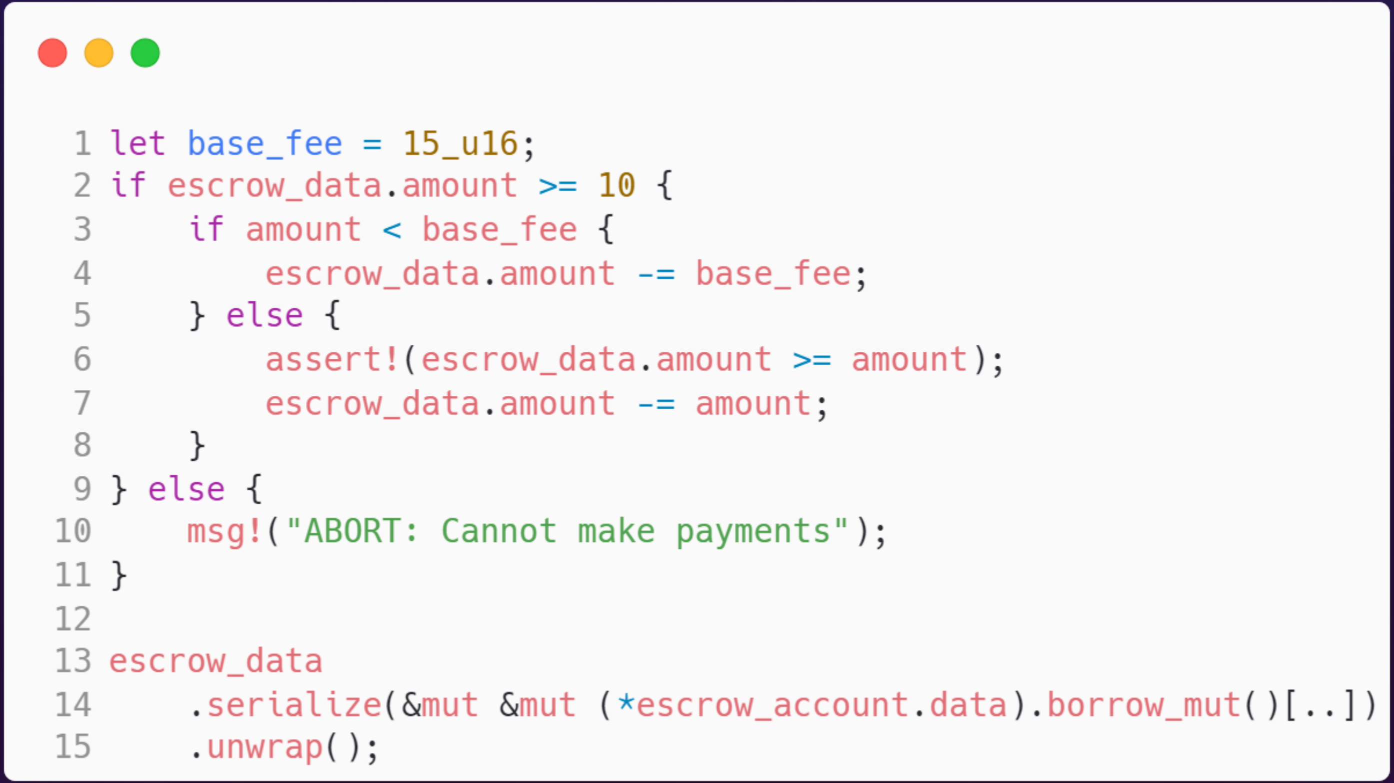
ExtensionTypes	immutableOwner
Mint	So11111111111111111111111111111111111111112

Content

- ➢ Intro
- ➢ Solana Basics
- ➢ **Our CTF Challenges**
- ➢ Web 2.5 Security

N1CTF 2022

Utility Payment Service



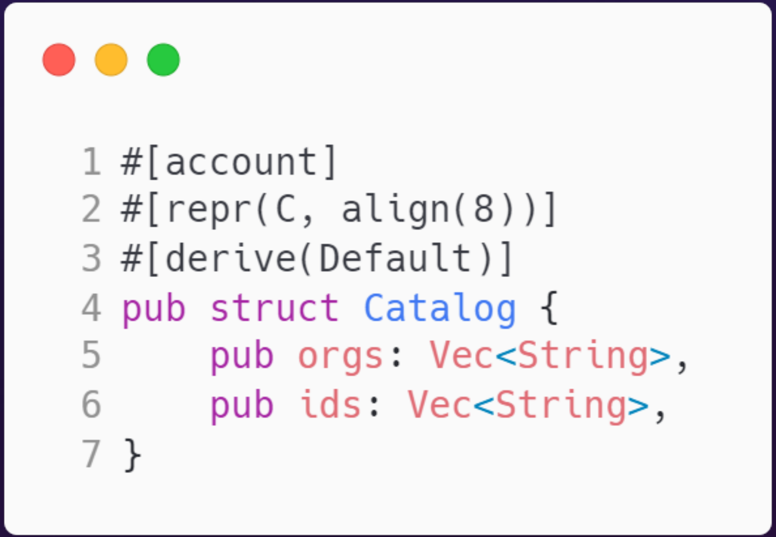
```
let base_fee = 15_u16;
if escrow_data.amount >= 10 {
    if amount < base_fee {
        escrow_data.amount -= base_fee;
    } else {
        assert!(escrow_data.amount >= amount);
        escrow_data.amount -= amount;
    }
} else {
    msg!("ABORT: Cannot make payments");
}

escrow_data
    .serialize(&mut &mut (*escrow_account.data).borrow_mut()[..])
    .unwrap();
```

N1CTF 2022

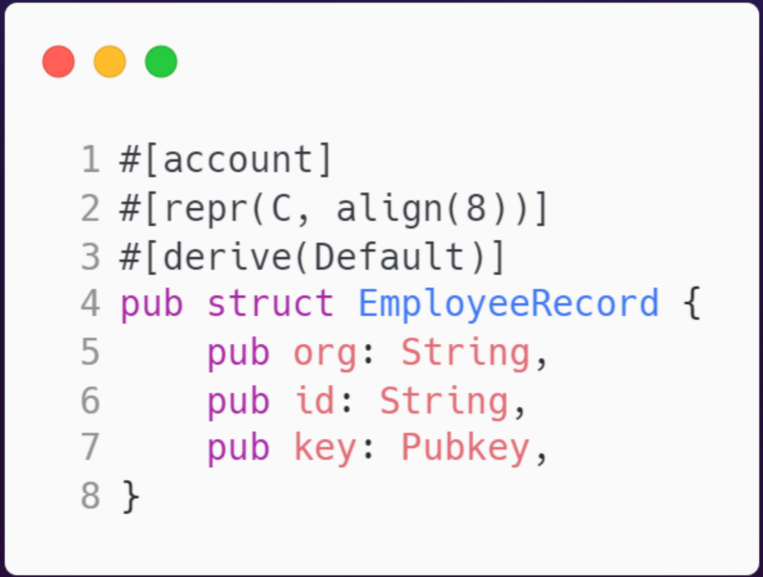
Simple Staking

Simple Staking

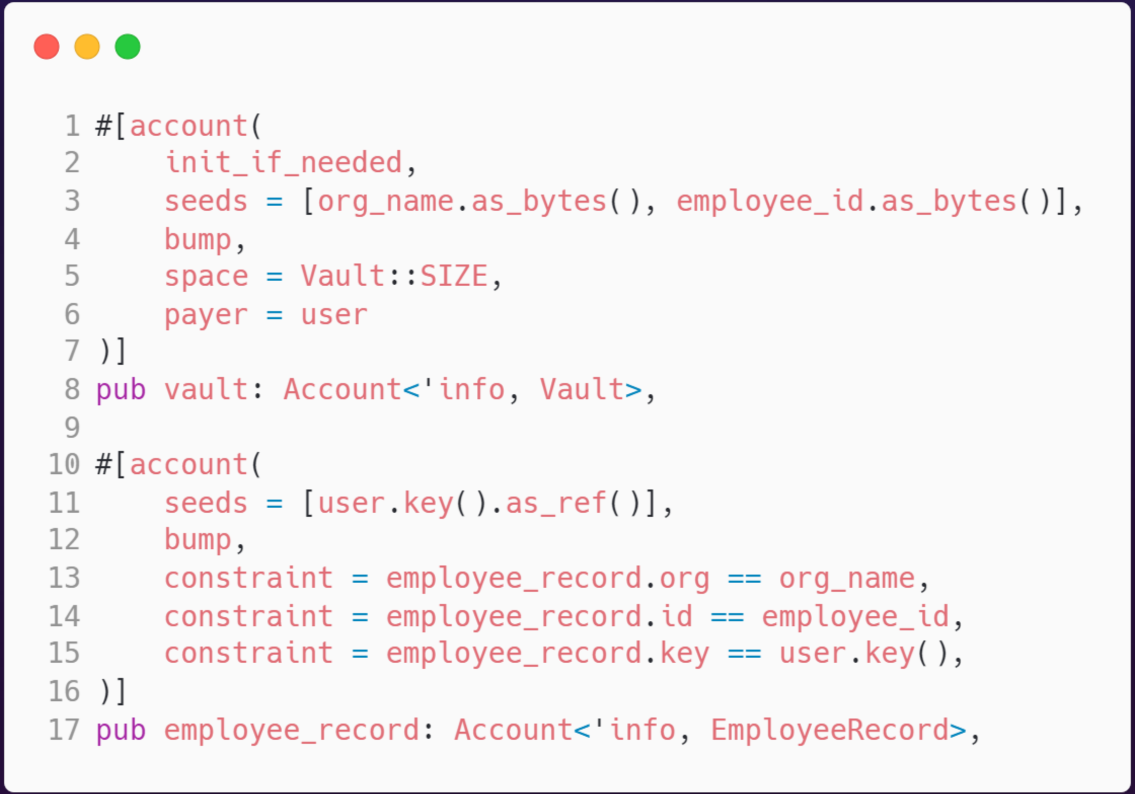
- Initialize
- Register (org_name, employee_id)
- Deposit (org_name, employee_id, amount)
- Withdraw (org_name, employee_id, amount)



```rust
#[account]
#[repr(C, align(8))]
#[derive(Default)]
pub struct Catalog {
    pub orgs: Vec<String>,
    pub ids: Vec<String>,
}
```



```rust
#[account]
#[repr(C, align(8))]
#[derive(Default)]
pub struct EmployeeRecord {
    pub org: String,
    pub id: String,
    pub key: Pubkey,
}
```



```
#[account(
    init_if_needed,
    seeds = [org_name.as_bytes(), employee_id.as_bytes()],
    bump,
    space = Vault::SIZE,
    payer = user
)]
pub vault: Account<'info, Vault>,

#[account(
    seeds = [user.key().as_ref()],
    bump,
    constraint = employee_record.org == org_name,
    constraint = employee_record.id == employee_id,
    constraint = employee_record.key == user.key(),
)]
pub employee_record: Account<'info, EmployeeRecord>,
```

Simple Staking

- Rich victim:
  - org_name = “product”,
  - employee_id = “employ_A”

Simple Staking

- Rich victim:
  - org_name = “product”,
  - employee_id = “employ_A”

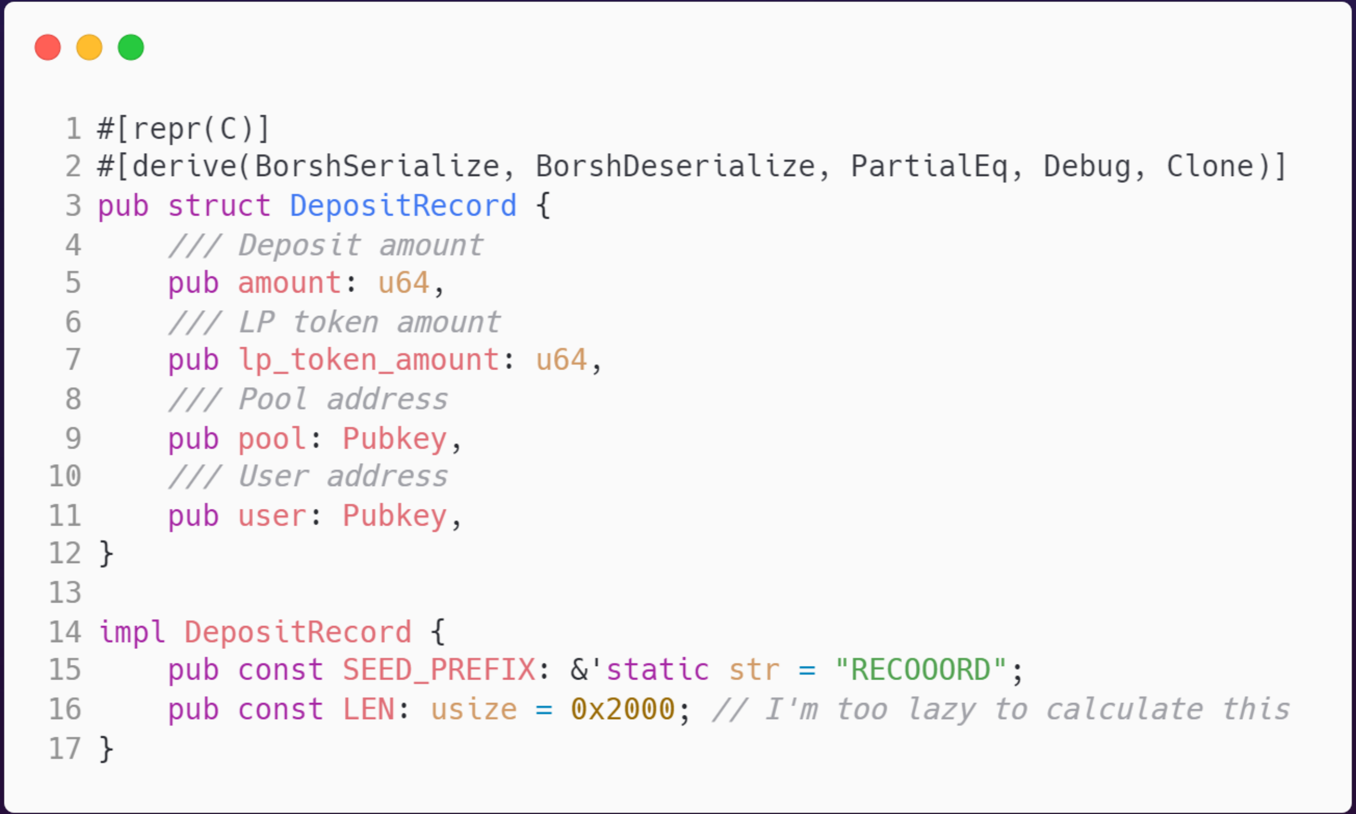
- Malicious user:
  - org_name = “producte”,
  - employee_id = “mploy_A”

N1CTF 2023

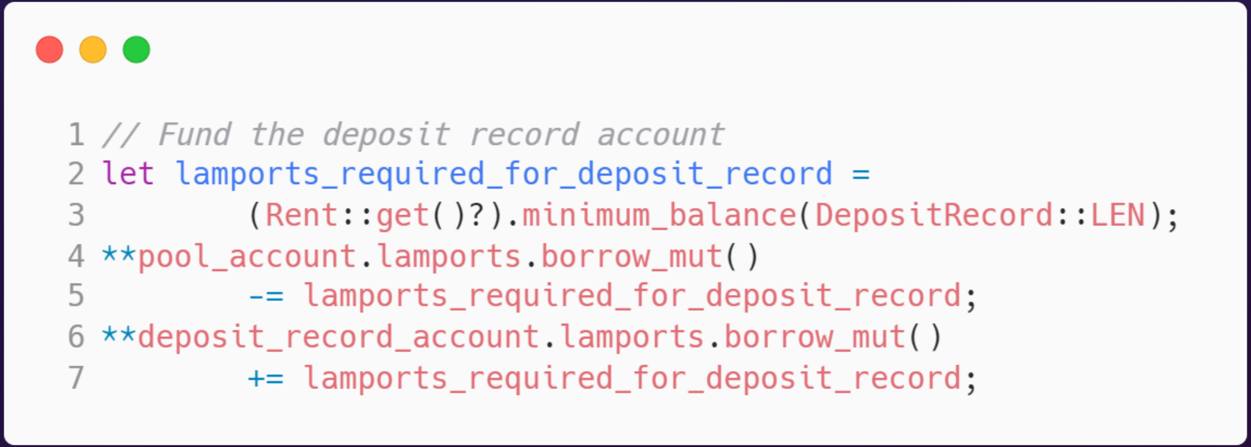
Pool

Pool

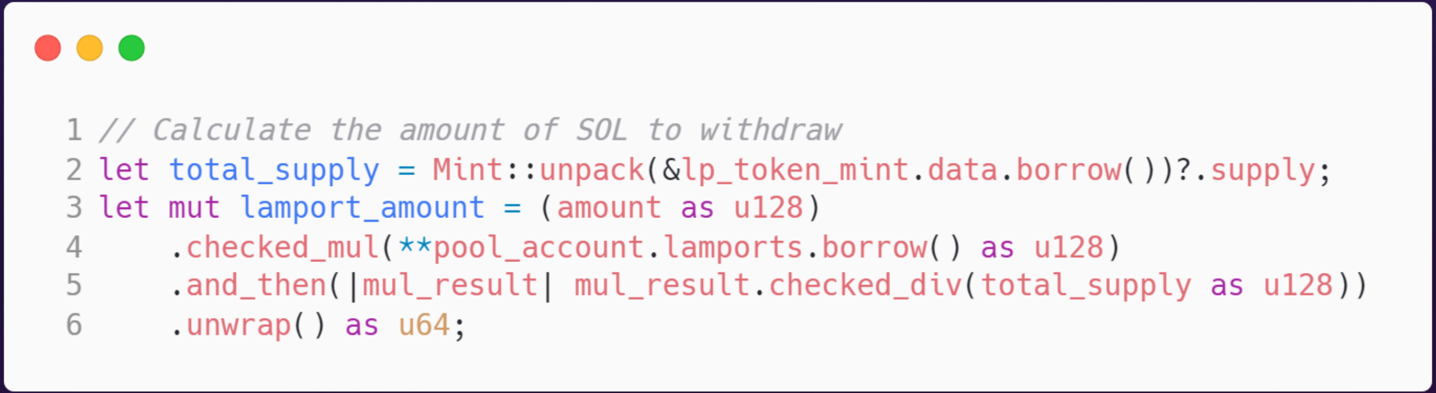
- InitPool (args)
- Deposit (amount, account_name)
- Withdraw (amount, account_name)



```rust
#[repr(C)]
#[derive(BorshSerialize, BorshDeserialize, PartialEq, Debug, Clone)]
pub struct DepositRecord {
    /// Deposit amount
    pub amount: u64,
    /// LP token amount
    pub lp_token_amount: u64,
    /// Pool address
    pub pool: Pubkey,
    /// User address
    pub user: Pubkey,
}

impl DepositRecord {
    pub const SEED_PREFIX: &'static str = "RECOOORD";
    pub const LEN: usize = 0x2000; // I'm too lazy to calculate this
}
```



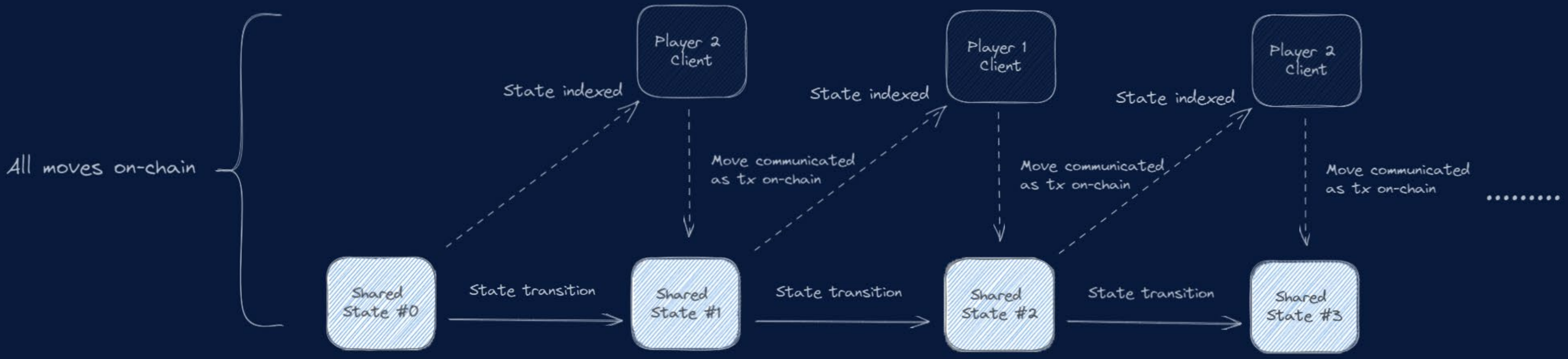
```rust
// Fund the deposit record account
let lamports_required_for_deposit_record =
        (Rent::get()?).minimum_balance(DepositRecord::LEN);
**pool_account.lamports.borrow_mut()
        -= lamports_required_for_deposit_record;
**deposit_record_account.lamports.borrow_mut()
        += lamports_required_for_deposit_record;
```



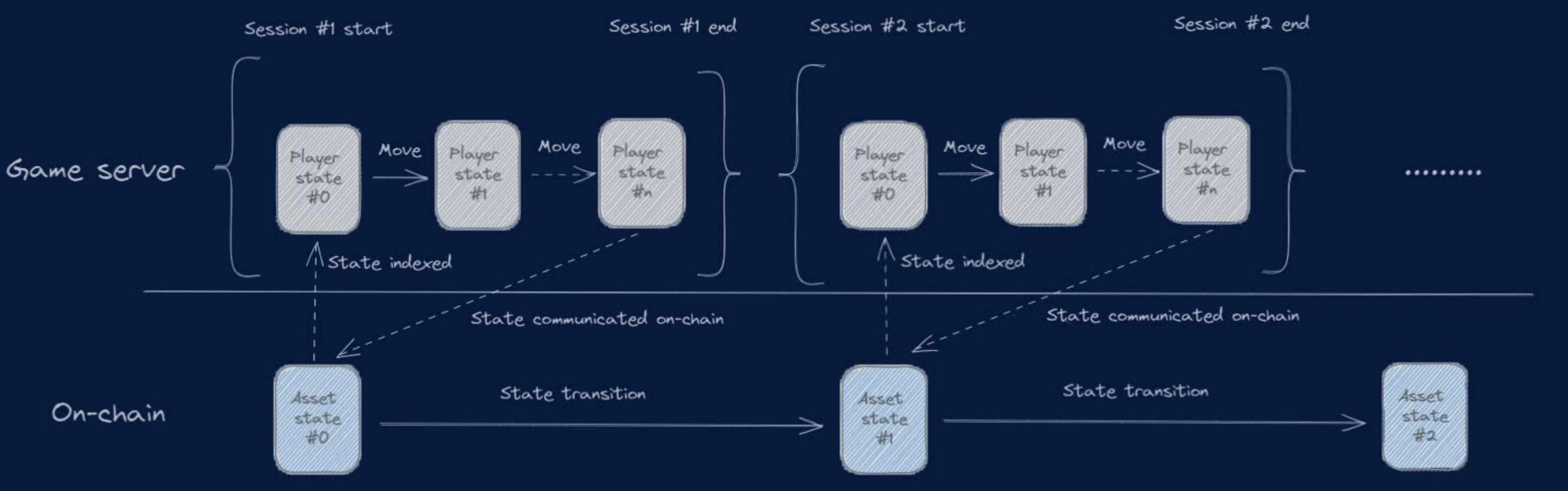
```rust
// Calculate the amount of SOL to withdraw
let total_supply = Mint::unpack(&lp_token_mint.data.borrow())?.supply;
let mut lamport_amount = (amount as u128)
    .checked_mul(**pool_account.lamports.borrow() as u128)
    .and_then(|mul_result| mul_result.checked_div(total_supply as u128))
    .unwrap() as u64;
```

Content

- Intro
- Solana Basics
- Our CTF Challenges
- **Web 2.5 Security**



Reference: https://jumpcrypto.com/writing/defining-on-chain-gaming/



Reference: https://jumpcrypto.com/writing/defining-on-chain-gaming/

京麒CTF 2023

CTFDAO

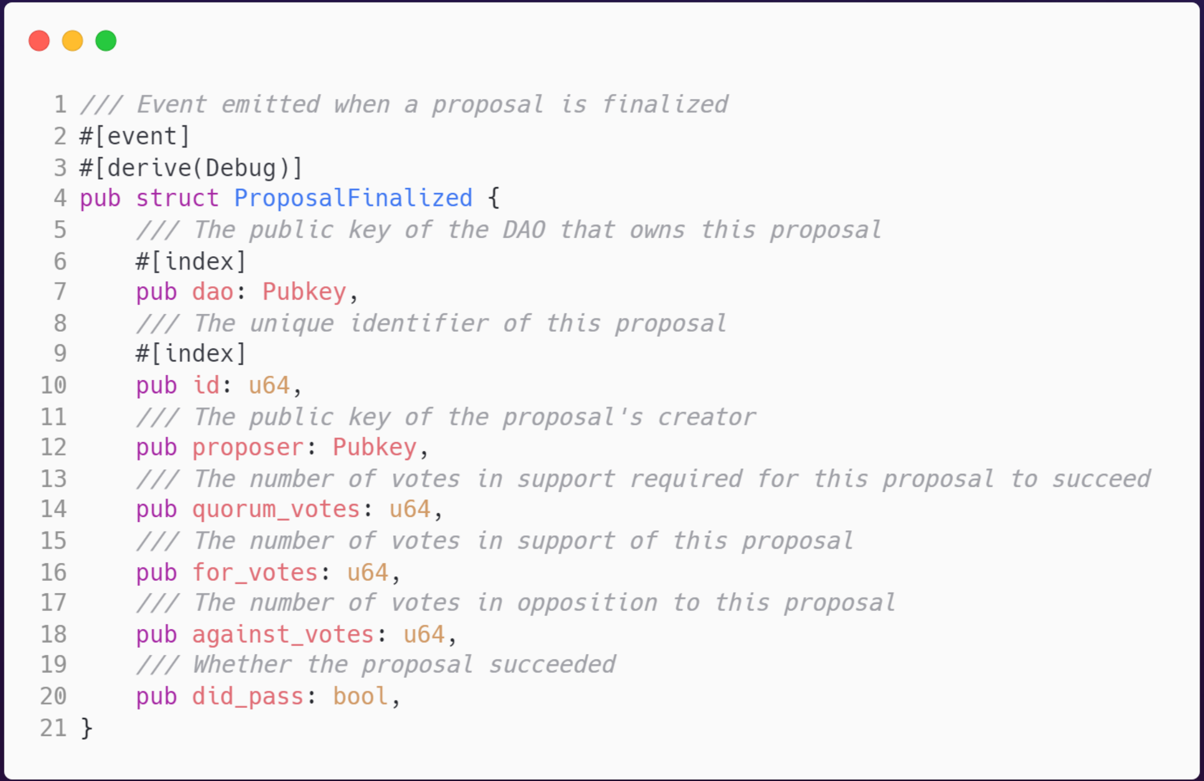
CTFDAO

- CreateDao (quorum_votes)

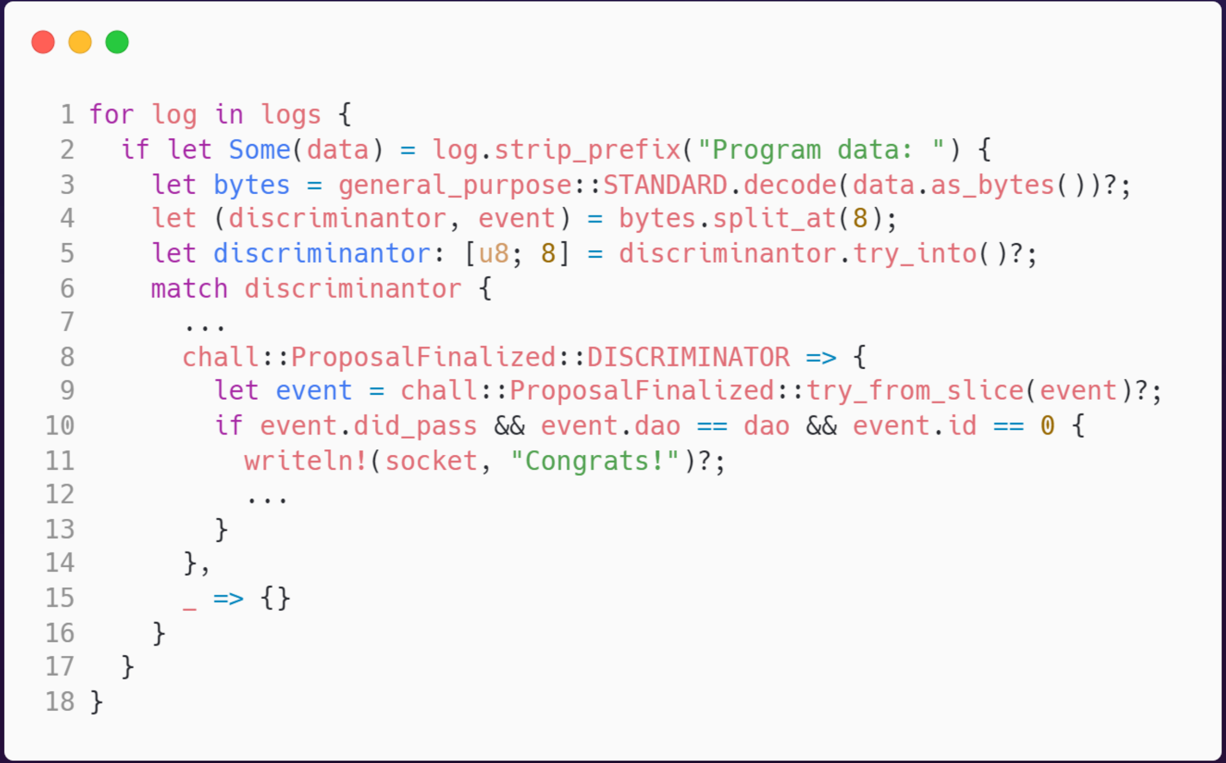
- CreateProposal (description)

- Vote (amount, support)

- CloseProposal



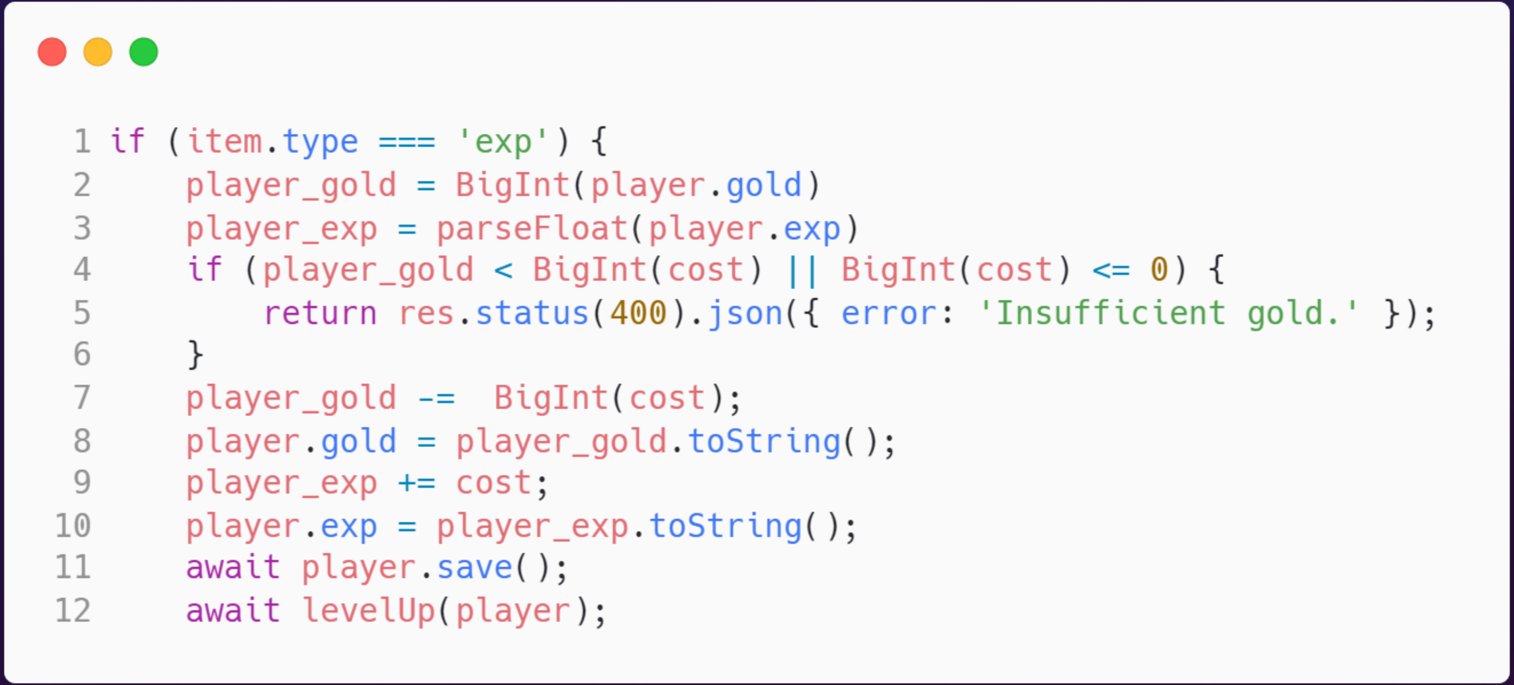
```rust
/// Event emitted when a proposal is finalized
#[event]
#[derive(Debug)]
pub struct ProposalFinalized {
    /// The public key of the DAO that owns this proposal
    #[index]
    pub dao: Pubkey,
    /// The unique identifier of this proposal
    #[index]
    pub id: u64,
    /// The public key of the proposal's creator
    pub proposer: Pubkey,
    /// The number of votes in support required for this proposal to succeed
    pub quorum_votes: u64,
    /// The number of votes in support of this proposal
    pub for_votes: u64,
    /// The number of votes in opposition to this proposal
    pub against_votes: u64,
    /// Whether the proposal succeeded
    pub did_pass: bool,
}
```



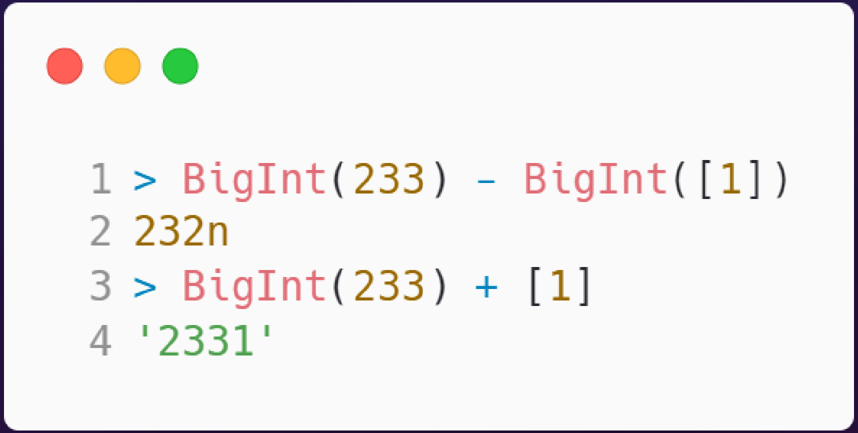
```rust
for log in logs {
  if let Some(data) = log.strip_prefix("Program data: ") {
    let bytes = general_purpose::STANDARD.decode(data.as_bytes())?;
    let (discriminantor, event) = bytes.split_at(8);
    let discriminantor: [u8; 8] = discriminantor.try_into()?;
    match discriminantor {
      ...
      chall::ProposalFinalized::DISCRIMINATOR => {
        let event = chall::ProposalFinalized::try_from_slice(event)?;
        if event.did_pass && event.dao == dao && event.id == 0 {
          writeln!(socket, "Congrats!")?;
          ...
        }
      },
      _ => {}
    }
  }
}
```

京麒CTF 2023

闪耀！优俊CTFer (by wupco)



```
if (item.type === 'exp') {
    player_gold = BigInt(player.gold)
    player_exp = parseFloat(player.exp)
    if (player_gold < BigInt(cost) || BigInt(cost) <= 0) {
        return res.status(400).json({ error: 'Insufficient gold.' });
    }
    player_gold -=  BigInt(cost);
    player.gold = player_gold.toString();
    player_exp += cost;
    player.exp = player_exp.toString();
    await player.save();
    await levelUp(player);
```



```
> BigInt(233) - BigInt([1])
232n
> BigInt(233) + [1]
'2331'
```

JS is weird (.com)

京麒
网络安全大会

2023
CTFCON

THANKS