# VRust

## Security Assessment

O2Lab VRust Team

11/04/2022 16:44:04

# Contents

## Summary

This report has been prepared for O2Lab VRust Team to discover issues and vulnerabilities in the source code of the O2Lab VRust Team project as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis and Manual Review techniques. The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases;
- Provide more comments per each function for readability, especially contracts that are verified in public;
- Provide more transparency on privileged activities once the protocol is live.

# Overview

## Project Summary

| | |
|---|---|
| Project Name | O2Lab VRust Team |
| Platform | Ethereum |
| Language | Solana |
| Crate | spl_stake_pool |
| GitHub Location | https://github.com/parasol-aser/vrust |
| sha256 | Unknown |

## Audit Summary

| | |
|---|---|
| Delivery Date | 11/04/2022 |
| Audit Methodology | Static Analysis |
| Key Components | |

## Vulnerability Summary

| Vulnerability Level | Total |
|---|---|
| Critical | 15 |
| Major | 0 |
| Medium | 0 |
| Minor | 0 |
| Informational | 0 |
| Discussion | 0 |

## Findings

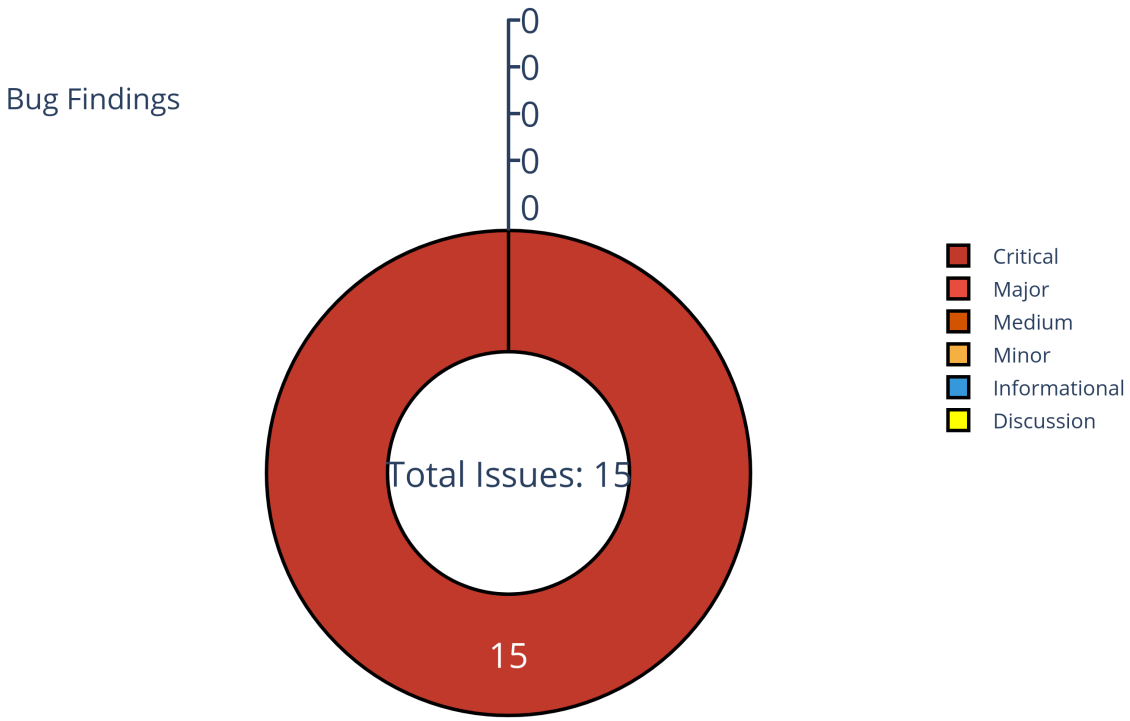


**Figure 1:** Findings

## Finding Statistic

| Category | Count |
|---|---|
| IntegerFlow | 4 |
| MissingKeyCheck | 3 |
| TypeConfusion | 8 |

| ID | Category | Severity | Status |
|---|---|---|---|
| 0 | IntegerFlow | Critical | UnResolved |
| 1 | IntegerFlow | Critical | UnResolved |
| 2 | IntegerFlow | Critical | UnResolved |
| 3 | IntegerFlow | Critical | UnResolved |
| 4 | MissingKeyCheck | Critical | UnResolved |
| 5 | MissingKeyCheck | Critical | UnResolved |
| 6 | MissingKeyCheck | Critical | UnResolved |
| 7 | TypeConfusion | Critical | GitHub Link to be added. |
| 8 | TypeConfusion | Critical | GitHub Link to be added. |
| 9 | TypeConfusion | Critical | GitHub Link to be added. |
| 10 | TypeConfusion | Critical | GitHub Link to be added. |
| 11 | TypeConfusion | Critical | GitHub Link to be added. |
| 12 | TypeConfusion | Critical | GitHub Link to be added. |
| 13 | TypeConfusion | Critical | GitHub Link to be added. |
| 14 | TypeConfusion | Critical | GitHub Link to be added. |

## Issue: 0: IntegerFlow

| Category | Severity | Status |
| --- | --- | --- |
| IntegerFlow | Critical | UnResolved |

- Location

stake-pool/program/src/state.rs:754:17: 754:43

```
754 self.numerator * old_denom
755
```

- Code Context

Vulnerability at Line: 754

```
749                 .map(|x|
    ↪  x.checked_mul(MAX_WITHDRAWAL_FEE_INCREASE.denominator as u128))
750                 .ok_or(StakePoolError::CalculationFailure)?
751         {
752             msg!(
753                 "Fee increase exceeds maximum allowed, proposed increase
                    ↪  factor ({} / {})",
754                 self.numerator * old_denom,
755                 old_num * self.denominator,
756             );
757             return Err(StakePoolError::FeeIncreaseTooHigh);
758         }
759
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
    ↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
2     fn processor::Processor::process(){//
        ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
3         fn processor::Processor::process_set_fee(){//
            ↪  stake-pool/program/src/processor.rs:2654:5: 2679:6 }
4             fn state::StakePool::update_fee(){//
                ↪  stake-pool/program/src/state.rs:462:5: 479:6 }
```

```
5                    fn state::Fee::check_withdrawal(){//
                     ↪  stake-pool/program/src/state.rs:728:5: 760:6 }
6
```

- description:
- link:
- alleviation:

## Issue: 1: IntegerFlow

| Category | Severity | Status |
| --- | --- | --- |
| IntegerFlow | Critical | UnResolved |

- Location

stake-pool/program/src/processor.rs:1334:33: 1334:47

```
1334 2 * stake_rent
1335
```

- Code Context

– Function Definition:

```
1221 fn process_increase_validator_stake(
1222         program_id: &Pubkey,
1223         accounts: &[AccountInfo],
1224         lamports: u64,
1225         transient_stake_seed: u64,
1226     ) -> ProgramResult
1227
```

Vulnerability at Line: 1334

```
1329             .saturating_sub(total_lamports)
1330             <= stake_rent
1331         {
1332             let max_split_amount = reserve_stake_account_info
1333                 .lamports()
1334                 .saturating_sub(2 * stake_rent);
1335             msg!(
1336                 "Reserve stake does not have enough lamports for increase,
                     ↪  must be less than {}, {} requested",
1337                 max_split_amount,
1338                 lamports
1339
```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
    fn processor::Processor::process(){//
    ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
        fn processor::Processor::process_increase_validator_stake(){//
        ↪  stake-pool/program/src/processor.rs:1221:5: 1377:6 }

```

- description:
- link:
- alleviation:

## Issue: 2: IntegerFlow

| Category | Severity | Status |
| --- | --- | --- |
| IntegerFlow | Critical | UnResolved |

- Location

stake-pool/program/src/processor.rs:855:33: 855:83

```
855 MINIMUM_ACTIVE_STAKE + rent.minimum_balance(space)
856
```

- Code Context

– Function Definition:

```
777 fn process_add_validator_to_pool(
778         program_id: &Pubkey,
779         accounts: &[AccountInfo],
780     ) -> ProgramResult
781
```

Vulnerability at Line: 855

```
850             &[bump_seed],
851         ];
852
853         // Fund the stake account with the minimum + rent-exempt balance
854         let space = std::mem::size_of::<stake::state::StakeState>();
855         let required_lamports = MINIMUM_ACTIVE_STAKE +
            ↪  rent.minimum_balance(space);
856
857         // Create new stake account
858         create_pda_account(
859             funder_info,
860
```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
    fn processor::Processor::process(){//
    ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
        fn processor::Processor::process_add_validator_to_pool(){//
        ↪  stake-pool/program/src/processor.rs:777:5: 906:6 }

```

- description:
- link:
- alleviation:

## Issue: 3: IntegerFlow

| Category | Severity | Status |
| --- | --- | --- |
| IntegerFlow | Critical | UnResolved |

- Location

stake-pool/program/src/big_vec.rs:114:44: 114:69

```
114 vec_len as usize * T::LEN
115
```

- Code Context

Vulnerability at Line: 114

```
110 pub fn push<T: Pack>(&mut self, element: T) -> Result<(), ProgramError> {
111         let mut vec_len_ref = &mut self.data[0..VEC_SIZE_BYTES];
112         let mut vec_len = u32::try_from_slice(vec_len_ref)?;
113
114         let start_index = VEC_SIZE_BYTES + vec_len as usize * T::LEN;
115         let end_index = start_index + T::LEN;
116
117         vec_len += 1;
118         vec_len.serialize(&mut vec_len_ref)?;
119
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
2     fn processor::Processor::process(){//
      ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
3         fn processor::Processor::process_add_validator_to_pool(){//
          ↪  stake-pool/program/src/processor.rs:777:5: 906:6 }
4             fn big_vec::BigVec::<'data>::push(){//
              ↪  stake-pool/program/src/big_vec.rs:110:5: 126:6 }
5
```

- description:

- link:
- alleviation:

## Issue: 4: MissingKeyCheck

| Category | Severity | Status |
| --- | --- | --- |
| MissingKeyCheck | Critical | UnResolved |

- Location

stake-pool/program/src/processor.rs:2501:69: 2501:98

```
2501 stake_pool_info.data.borrow()
2502
```

- Code Context

– Function Definition:

```
2480 fn process_withdraw_sol(
2481         program_id: &Pubkey,
2482         accounts: &[AccountInfo],
2483         pool_tokens: u64,
2484     ) -> ProgramResult
2485
```

Vulnerability at Line: 2501

```
2496         let stake_program_info = next_account_info(account_info_iter)?;
2497         let token_program_info = next_account_info(account_info_iter)?;
2498         let sol_withdraw_authority_info =
              ↪  next_account_info(account_info_iter);
2499
2500         check_account_owner(stake_pool_info, program_id)?;
2501         let mut stake_pool =
              ↪  try_from_slice_unchecked::<StakePool>(&stake_pool_info.data.borrow())?;
2502         if !stake_pool.is_valid() {
2503             return Err(StakePoolError::InvalidState.into());
2504         }
2505
2506
```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
    fn processor::Processor::process(){//
    ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
        fn processor::Processor::process_withdraw_sol(){//
        ↪  stake-pool/program/src/processor.rs:2480:5: 2615:6 }

```

- description:
- link:
- alleviation:

## Issue: 5: MissingKeyCheck

| Category | Severity | Status |
| --- | --- | --- |
| MissingKeyCheck | Critical | UnResolved |

- Location

stake-pool/program/src/state.rs:292:46: 292:76

```
292 manager_fee_info.data.borrow()
293
```

- Code Context

Vulnerability at Line: 292

```
288 pub(crate) fn check_manager_fee_info(
289         &self,
290         manager_fee_info: &AccountInfo,
291     ) -> Result<(), ProgramError> {
292         let token_account =
        ↪  Account::unpack(&manager_fee_info.data.borrow())?;
293         if manager_fee_info.owner != &self.token_program_id
294             || token_account.state != AccountState::Initialized
295             || token_account.mint != self.pool_mint
296         {
297
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
2     fn processor::Processor::process(){//
      ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
3         fn processor::Processor::process_withdraw_sol(){//
          ↪  stake-pool/program/src/processor.rs:2480:5: 2615:6 }
4             fn state::StakePool::check_manager_fee_info(){//
              ↪  stake-pool/program/src/state.rs:288:5: 301:6 }
5
```

- description:
- link:
- alleviation:

## Issue: 6: MissingKeyCheck

| Category | Severity | Status |
| --- | --- | --- |
| MissingKeyCheck | Critical | UnResolved |

- Location

/home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/account_info.rs:66:11: 66:33

```
66 self.lamports.borrow()
67
```

- Code Context

Vulnerability at Line: 66

```
65 pub fn lamports(&self) -> u64 {
66         **self.lamports.borrow()
67     }
68
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  stake-pool/program/src/entrypoint.rs:12:1: 24:2 }
2     fn processor::Processor::process(){//
      ↪  stake-pool/program/src/processor.rs:2740:5: 2858:6 }
3         fn processor::Processor::process_withdraw_sol(){//
          ↪  stake-pool/program/src/processor.rs:2480:5: 2615:6 }
4             fn
              ↪  solana_program::account_info::AccountInfo::<'a>::lamports(){//
              ↪  /home/yifei/.cargo/registry/src/github.com-
              ↪  1ecc6299db9ec823/solana-program-
              ↪  1.9.9/src/account_info.rs:65:5: 67:6
              ↪  }
5
```

- description:

- link:
- alleviation:

## Issue: 7: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/instruction.rs:45:69: 45:80

```
BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/state.rs:485:1: 491:2
    pub struct ValidatorList {
    /// Data outside of the validator list, separated out for cheaper
     ↪  deserializations
    pub header: ValidatorListHeader,

    /// List of stake info for each validator in the pool
    pub validators: Vec<ValidatorStakeInfo>,
}
stake-pool/program/src/state.rs:703:1: 708:2
    pub struct Fee {
    /// denominator of the fee ratio
    pub denominator: u64,
    /// numerator of the fee ratio
    pub numerator: u64,
}
stake-pool/program/src/big_vec.rs:190:1: 196:2
    pub struct Iter<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec BigVec<'data>,
```

```
    phantom: PhantomData<T>,
}
stake-pool/program/src/big_vec.rs:217:1: 223:2
    pub struct IterMut<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec mut BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/state.rs:535:1: 564:2
    pub struct ValidatorStakeInfo {
    /// Amount of active stake delegated to this validator, minus the
     ↪ minimum
    /// required stake amount of rent-exemption +
     ↪ `crate::MINIMUM_ACTIVE_STAKE`
    /// (currently 0.001 SOL).
    ///
    /// Note that if `last_update_epoch` does not match the current epoch
     ↪ then
    /// this field may not be accurate
    pub active_stake_lamports: u64,

    /// Amount of transient stake delegated to this validator
    ///
    /// Note that if `last_update_epoch` does not match the current epoch
     ↪ then
    /// this field may not be accurate
    pub transient_stake_lamports: u64,

    /// Last epoch the active and transient stake lamports fields were
     ↪ updated
    pub last_update_epoch: u64,

    /// Start of the validator transient account seed suffixess
    pub transient_seed_suffix_start: u64,

    /// End of the validator transient account seed suffixes
    pub transient_seed_suffix_end: u64,

    /// Status of the validator stake account
    pub status: StakeStatus,
```

```
112
113     /// Validator vote account address
114     pub vote_account_address: Pubkey,
115 }
116
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 8: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/instruction.rs:45:69: 45:80

```
BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/state.rs:485:1: 491:2
    pub struct ValidatorList {
    /// Data outside of the validator list, separated out for cheaper
    ↪ deserializations
    pub header: ValidatorListHeader,

    /// List of stake info for each validator in the pool
    pub validators: Vec<ValidatorStakeInfo>,
}
stake-pool/program/src/state.rs:703:1: 708:2
    pub struct Fee {
    /// denominator of the fee ratio
    pub denominator: u64,
    /// numerator of the fee ratio
    pub numerator: u64,
}
stake-pool/program/src/big_vec.rs:190:1: 196:2
    pub struct Iter<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec BigVec<'data>,
    phantom: PhantomData<T>,
}
```

```
stake-pool/program/src/big_vec.rs:217:1: 223:2
    pub struct IterMut<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec mut BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/state.rs:535:1: 564:2
    pub struct ValidatorStakeInfo {
    /// Amount of active stake delegated to this validator, minus the
     ↪ minimum
    /// required stake amount of rent-exemption +
     ↪ `crate::MINIMUM_ACTIVE_STAKE`
    /// (currently 0.001 SOL).
    ///
    /// Note that if `last_update_epoch` does not match the current epoch
     ↪ then
    /// this field may not be accurate
    pub active_stake_lamports: u64,

    /// Amount of transient stake delegated to this validator
    ///
    /// Note that if `last_update_epoch` does not match the current epoch
     ↪ then
    /// this field may not be accurate
    pub transient_stake_lamports: u64,

    /// Last epoch the active and transient stake lamports fields were
     ↪ updated
    pub last_update_epoch: u64,

    /// Start of the validator transient account seed suffixess
    pub transient_seed_suffix_start: u64,

    /// End of the validator transient account seed suffixes
    pub transient_seed_suffix_end: u64,

    /// Status of the validator stake account
    pub status: StakeStatus,

    /// Validator vote account address
```

```
112     pub vote_account_address: Pubkey,
113 }
114
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 9: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/instruction.rs:45:69: 45:80

```
BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/state.rs:485:1: 491:2
    pub struct ValidatorList {
    /// Data outside of the validator list, separated out for cheaper
    ↪ deserializations
    pub header: ValidatorListHeader,

    /// List of stake info for each validator in the pool
    pub validators: Vec<ValidatorStakeInfo>,
}
stake-pool/program/src/state.rs:703:1: 708:2
    pub struct Fee {
    /// denominator of the fee ratio
    pub denominator: u64,
    /// numerator of the fee ratio
    pub numerator: u64,
}
stake-pool/program/src/big_vec.rs:190:1: 196:2
    pub struct Iter<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/big_vec.rs:217:1: 223:2
    pub struct IterMut<'data, 'vec, T> {
```

```
75      len: usize,
76      current: usize,
77      current_index: usize,
78      inner: &'vec mut BigVec<'data>,
79      phantom: PhantomData<T>,
80  }
81  stake-pool/program/src/state.rs:535:1: 564:2
82      pub struct ValidatorStakeInfo {
83      /// Amount of active stake delegated to this validator, minus the
        ↪  minimum
84      /// required stake amount of rent-exemption +
        ↪  `crate::MINIMUM_ACTIVE_STAKE`
85      /// (currently 0.001 SOL).
86      ///
87      /// Note that if `last_update_epoch` does not match the current epoch
        ↪  then
88      /// this field may not be accurate
89      pub active_stake_lamports: u64,
90
91      /// Amount of transient stake delegated to this validator
92      ///
93      /// Note that if `last_update_epoch` does not match the current epoch
        ↪  then
94      /// this field may not be accurate
95      pub transient_stake_lamports: u64,
96
97      /// Last epoch the active and transient stake lamports fields were
        ↪  updated
98      pub last_update_epoch: u64,
99
100     /// Start of the validator transient account seed suffixess
101     pub transient_seed_suffix_start: u64,
102
103     /// End of the validator transient account seed suffixes
104     pub transient_seed_suffix_end: u64,
105
106     /// Status of the validator stake account
107     pub status: StakeStatus,
108
109     /// Validator vote account address
110     pub vote_account_address: Pubkey,
111 }
```

```
112
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 10: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/state.rs:774:69: 774:80

```
774 BorshSchema
775 stake-pool/program/src/state.rs:774:69: 774:80
776     BorshSchema
777 stake-pool/program/src/state.rs:496:1: 502:2
778     pub struct ValidatorListHeader {
779     /// Account type, must be ValidatorList currently
780     pub account_type: AccountType,
781 
782     /// Maximum allowable number of validators
783     pub max_validators: u32,
784 }
785 stake-pool/program/src/state.rs:46:1: 156:2
786     pub struct StakePool {
787     /// Account type, must be StakePool currently
788     pub account_type: AccountType,
789 
790     /// Manager authority, allows for updating the staker, manager, and fee
        ↪ account
791     pub manager: Pubkey,
792 
793     /// Staker authority, allows for adding and removing validators, and
        ↪ managing stake
794     /// distribution
795     pub staker: Pubkey,
796 
797     /// Stake deposit authority
798     ///
799     /// If a depositor pubkey is specified on initialization, then deposits
        ↪ must be
800     /// signed by this authority. If no deposit authority is specified,
801     /// then the stake pool will default to the result of:
```

```
802     /// `Pubkey::find_program_address(
803     ///     &[&stake_pool_address.to_bytes()[..32], b"deposit"],
804     ///     program_id,
805     /// )`
806     pub stake_deposit_authority: Pubkey,
807 
808     /// Stake withdrawal authority bump seed
809     /// for `create_program_address(&[state::StakePool account,
        ↪ "withdrawal"])`
810     pub stake_withdraw_bump_seed: u8,
811 
812     /// Validator stake list storage account
813     pub validator_list: Pubkey,
814 
815     /// Reserve stake account, holds deactivated stake
816     pub reserve_stake: Pubkey,
817 
818     /// Pool Mint
819     pub pool_mint: Pubkey,
820 
821     /// Manager fee account
822     pub manager_fee_account: Pubkey,
823 
824     /// Pool token program id
825     pub token_program_id: Pubkey,
826 
827     /// Total stake under management.
828     /// Note that if `last_update_epoch` does not match the current epoch
        ↪ then
829     /// this field may not be accurate
830     pub total_lamports: u64,
831 
832     /// Total supply of pool tokens (should always match the supply in the
        ↪ Pool Mint)
833     pub pool_token_supply: u64,
834 
835     /// Last epoch the `total_lamports` field was updated
836     pub last_update_epoch: u64,
837 
838     /// Lockup that all stakes in the pool must have
839     pub lockup: Lockup,
840 
```

```
841     /// Fee taken as a proportion of rewards each epoch
842     pub epoch_fee: Fee,
843 
844     /// Fee for next epoch
845     pub next_epoch_fee: Option<Fee>,
846 
847     /// Preferred deposit validator vote account pubkey
848     pub preferred_deposit_validator_vote_address: Option<Pubkey>,
849 
850     /// Preferred withdraw validator vote account pubkey
851     pub preferred_withdraw_validator_vote_address: Option<Pubkey>,
852 
853     /// Fee assessed on stake deposits
854     pub stake_deposit_fee: Fee,
855 
856     /// Fee assessed on withdrawals
857     pub stake_withdrawal_fee: Fee,
858 
859     /// Future stake withdrawal fee, to be set for the following epoch
860     pub next_stake_withdrawal_fee: Option<Fee>,
861 
862     /// Fees paid out to referrers on referred stake deposits.
863     /// Expressed as a percentage (0 - 100) of deposit fees.
864     /// i.e. `stake_deposit_fee`% of stake deposited is collected as deposit
      ↪  fees for every deposit
865     /// and `stake_referral_fee`% of the collected stake deposit fees is
      ↪  paid out to the referrer
866     pub stake_referral_fee: u8,
867 
868     /// Toggles whether the `DepositSol` instruction requires a signature
      ↪  from
869     /// this `sol_deposit_authority`
870     pub sol_deposit_authority: Option<Pubkey>,
871 
872     /// Fee assessed on SOL deposits
873     pub sol_deposit_fee: Fee,
874 
875     /// Fees paid out to referrers on referred SOL deposits.
876     /// Expressed as a percentage (0 - 100) of SOL deposit fees.
877     /// i.e. `sol_deposit_fee`% of SOL deposited is collected as deposit
      ↪  fees for every deposit
878     /// and `sol_referral_fee`% of the collected SOL deposit fees is paid
      ↪  out to the referrer
```

```
879     pub sol_referral_fee: u8,
880
881     /// Toggles whether the `WithdrawSol` instruction requires a signature
        ↪  from
882     /// the `deposit_authority`
883     pub sol_withdraw_authority: Option<Pubkey>,
884
885     /// Fee assessed on SOL withdrawals
886     pub sol_withdrawal_fee: Fee,
887
888     /// Future SOL withdrawal fee, to be set for the following epoch
889     pub next_sol_withdrawal_fee: Option<Fee>,
890
891     /// Last epoch's total pool tokens, used only for APR estimation
892     pub last_epoch_pool_token_supply: u64,
893
894     /// Last epoch's total lamports, used only for APR estimation
895     pub last_epoch_total_lamports: u64,
896 }
897
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 11: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/state.rs:774:69: 774:80

```
774 BorshSchema
775 stake-pool/program/src/state.rs:496:1: 502:2
776     pub struct ValidatorListHeader {
777     /// Account type, must be ValidatorList currently
778     pub account_type: AccountType,
779 
780     /// Maximum allowable number of validators
781     pub max_validators: u32,
782 }
783 stake-pool/program/src/state.rs:46:1: 156:2
784     pub struct StakePool {
785     /// Account type, must be StakePool currently
786     pub account_type: AccountType,
787 
788     /// Manager authority, allows for updating the staker, manager, and fee
      ↪  account
789     pub manager: Pubkey,
790 
791     /// Staker authority, allows for adding and removing validators, and
      ↪  managing stake
792     /// distribution
793     pub staker: Pubkey,
794 
795     /// Stake deposit authority
796     ///
797     /// If a depositor pubkey is specified on initialization, then deposits
      ↪  must be
798     /// signed by this authority. If no deposit authority is specified,
799     /// then the stake pool will default to the result of:
800     /// `Pubkey::find_program_address(
801     ///     &[&stake_pool_address.to_bytes()[..32], b"deposit"],
```

```
802     ///     program_id,
803     /// )`
804     pub stake_deposit_authority: Pubkey,
805
806     /// Stake withdrawal authority bump seed
807     /// for `create_program_address(&[state::StakePool account,
        ↪ "withdrawal"])`
808     pub stake_withdraw_bump_seed: u8,
809
810     /// Validator stake list storage account
811     pub validator_list: Pubkey,
812
813     /// Reserve stake account, holds deactivated stake
814     pub reserve_stake: Pubkey,
815
816     /// Pool Mint
817     pub pool_mint: Pubkey,
818
819     /// Manager fee account
820     pub manager_fee_account: Pubkey,
821
822     /// Pool token program id
823     pub token_program_id: Pubkey,
824
825     /// Total stake under management.
826     /// Note that if `last_update_epoch` does not match the current epoch
        ↪ then
827     /// this field may not be accurate
828     pub total_lamports: u64,
829
830     /// Total supply of pool tokens (should always match the supply in the
        ↪ Pool Mint)
831     pub pool_token_supply: u64,
832
833     /// Last epoch the `total_lamports` field was updated
834     pub last_update_epoch: u64,
835
836     /// Lockup that all stakes in the pool must have
837     pub lockup: Lockup,
838
839     /// Fee taken as a proportion of rewards each epoch
840     pub epoch_fee: Fee,
```

```rust

    /// Fee for next epoch
    pub next_epoch_fee: Option<Fee>,

    /// Preferred deposit validator vote account pubkey
    pub preferred_deposit_validator_vote_address: Option<Pubkey>,

    /// Preferred withdraw validator vote account pubkey
    pub preferred_withdraw_validator_vote_address: Option<Pubkey>,

    /// Fee assessed on stake deposits
    pub stake_deposit_fee: Fee,

    /// Fee assessed on withdrawals
    pub stake_withdrawal_fee: Fee,

    /// Future stake withdrawal fee, to be set for the following epoch
    pub next_stake_withdrawal_fee: Option<Fee>,

    /// Fees paid out to referrers on referred stake deposits.
    /// Expressed as a percentage (0 - 100) of deposit fees.
    /// i.e. `stake_deposit_fee`% of stake deposited is collected as deposit
    ↪ fees for every deposit
    /// and `stake_referral_fee`% of the collected stake deposit fees is
    ↪ paid out to the referrer
    pub stake_referral_fee: u8,

    /// Toggles whether the `DepositSol` instruction requires a signature
    ↪ from
    /// this `sol_deposit_authority`
    pub sol_deposit_authority: Option<Pubkey>,

    /// Fee assessed on SOL deposits
    pub sol_deposit_fee: Fee,

    /// Fees paid out to referrers on referred SOL deposits.
    /// Expressed as a percentage (0 - 100) of SOL deposit fees.
    /// i.e. `sol_deposit_fee`% of SOL deposited is collected as deposit
    ↪ fees for every deposit
    /// and `sol_referral_fee`% of the collected SOL deposit fees is paid
    ↪ out to the referrer
    pub sol_referral_fee: u8,
```

```
878
879     /// Toggles whether the `WithdrawSol` instruction requires a signature
        ↪  from
880     /// the `deposit_authority`
881     pub sol_withdraw_authority: Option<Pubkey>,
882
883     /// Fee assessed on SOL withdrawals
884     pub sol_withdrawal_fee: Fee,
885
886     /// Future SOL withdrawal fee, to be set for the following epoch
887     pub next_sol_withdrawal_fee: Option<Fee>,
888
889     /// Last epoch's total pool tokens, used only for APR estimation
890     pub last_epoch_pool_token_supply: u64,
891
892     /// Last epoch's total lamports, used only for APR estimation
893     pub last_epoch_total_lamports: u64,
894 }
895
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 12: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/instruction.rs:45:69: 45:80

```
BorshSchema
stake-pool/program/src/instruction.rs:45:69: 45:80
    BorshSchema
stake-pool/program/src/state.rs:703:1: 708:2
    pub struct Fee {
    /// denominator of the fee ratio
    pub denominator: u64,
    /// numerator of the fee ratio
    pub numerator: u64,
}
stake-pool/program/src/big_vec.rs:190:1: 196:2
    pub struct Iter<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/big_vec.rs:217:1: 223:2
    pub struct IterMut<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec mut BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/state.rs:535:1: 564:2
    pub struct ValidatorStakeInfo {
    /// Amount of active stake delegated to this validator, minus the
     ↪ minimum
    /// required stake amount of rent-exemption +
     ↪ `crate::MINIMUM_ACTIVE_STAKE`
```

```
75      /// (currently 0.001 SOL).
76      ///
77      /// Note that if `last_update_epoch` does not match the current epoch
        ↪  then
78      /// this field may not be accurate
79      pub active_stake_lamports: u64,
80
81      /// Amount of transient stake delegated to this validator
82      ///
83      /// Note that if `last_update_epoch` does not match the current epoch
        ↪  then
84      /// this field may not be accurate
85      pub transient_stake_lamports: u64,
86
87      /// Last epoch the active and transient stake lamports fields were
        ↪  updated
88      pub last_update_epoch: u64,
89
90      /// Start of the validator transient account seed suffixess
91      pub transient_seed_suffix_start: u64,
92
93      /// End of the validator transient account seed suffixes
94      pub transient_seed_suffix_end: u64,
95
96      /// Status of the validator stake account
97      pub status: StakeStatus,
98
99      /// Validator vote account address
100     pub vote_account_address: Pubkey,
101 }
102
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

## Issue: 13: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/instruction.rs:45:69: 45:80

```
BorshSchema
stake-pool/program/src/state.rs:703:1: 708:2
    pub struct Fee {
    /// denominator of the fee ratio
    pub denominator: u64,
    /// numerator of the fee ratio
    pub numerator: u64,
}
stake-pool/program/src/big_vec.rs:190:1: 196:2
    pub struct Iter<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/big_vec.rs:217:1: 223:2
    pub struct IterMut<'data, 'vec, T> {
    len: usize,
    current: usize,
    current_index: usize,
    inner: &'vec mut BigVec<'data>,
    phantom: PhantomData<T>,
}
stake-pool/program/src/state.rs:535:1: 564:2
    pub struct ValidatorStakeInfo {
    /// Amount of active stake delegated to this validator, minus the
     ↪  minimum
    /// required stake amount of rent-exemption +
     ↪  `crate::MINIMUM_ACTIVE_STAKE`
    /// (currently 0.001 SOL).
```

```
74      ///
75      /// Note that if `last_update_epoch` does not match the current epoch
         ↪  then
76      /// this field may not be accurate
77      pub active_stake_lamports: u64,
78
79      /// Amount of transient stake delegated to this validator
80      ///
81      /// Note that if `last_update_epoch` does not match the current epoch
         ↪  then
82      /// this field may not be accurate
83      pub transient_stake_lamports: u64,
84
85      /// Last epoch the active and transient stake lamports fields were
         ↪  updated
86      pub last_update_epoch: u64,
87
88      /// Start of the validator transient account seed suffixess
89      pub transient_seed_suffix_start: u64,
90
91      /// End of the validator transient account seed suffixes
92      pub transient_seed_suffix_end: u64,
93
94      /// Status of the validator stake account
95      pub status: StakeStatus,
96
97      /// Validator vote account address
98      pub vote_account_address: Pubkey,
99  }
100
```

- Call Stack

```
1  UnResolved
```

- description:
- link:
- alleviation:

## Issue: 14: TypeConfusion

| Category | Severity | Status |
| --- | --- | --- |
| TypeConfusion | Critical | GitHub Link to be added. |

- Location

stake-pool/program/src/state.rs:703:1: 708:2

```
703 pub struct Fee {
704     /// denominator of the fee ratio
705     pub denominator: u64,
706     /// numerator of the fee ratio
707     pub numerator: u64,
708 }
709 stake-pool/program/src/big_vec.rs:190:1: 196:2
710     pub struct Iter<'data, 'vec, T> {
711     len: usize,
712     current: usize,
713     current_index: usize,
714     inner: &'vec BigVec<'data>,
715     phantom: PhantomData<T>,
716 }
717 stake-pool/program/src/big_vec.rs:217:1: 223:2
718     pub struct IterMut<'data, 'vec, T> {
719     len: usize,
720     current: usize,
721     current_index: usize,
722     inner: &'vec mut BigVec<'data>,
723     phantom: PhantomData<T>,
724 }
725 stake-pool/program/src/state.rs:535:1: 564:2
726     pub struct ValidatorStakeInfo {
727     /// Amount of active stake delegated to this validator, minus the
      ↪  minimum
728     /// required stake amount of rent-exemption +
      ↪  `crate::MINIMUM_ACTIVE_STAKE`
729     /// (currently 0.001 SOL).
730     ///
731     /// Note that if `last_update_epoch` does not match the current epoch
      ↪  then
```

```
732     /// this field may not be accurate
733     pub active_stake_lamports: u64,
734 
735     /// Amount of transient stake delegated to this validator
736     ///
737     /// Note that if `last_update_epoch` does not match the current epoch
        ↪  then
738     /// this field may not be accurate
739     pub transient_stake_lamports: u64,
740 
741     /// Last epoch the active and transient stake lamports fields were
        ↪  updated
742     pub last_update_epoch: u64,
743 
744     /// Start of the validator transient account seed suffixess
745     pub transient_seed_suffix_start: u64,
746 
747     /// End of the validator transient account seed suffixes
748     pub transient_seed_suffix_end: u64,
749 
750     /// Status of the validator stake account
751     pub status: StakeStatus,
752 
753     /// Validator vote account address
754     pub vote_account_address: Pubkey,
755 }
756 
```

- Call Stack

```
1 UnResolved
```

- description:
- link:
- alleviation:

# Appendix

Copied from https://leaderboard.certik.io/projects/aave

## Finding Categories

### Gas Optimization

Gas Optimization findings do not affect the functionality of the code but generate different, more optimal EVM opcodes resulting in a reduction on the total gas cost of a transaction.

### Mathematical Operations

Mathematical Operation findings relate to mishandling of math formulas, such as overflows, incorrect operations etc.

### Logical Issue

Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how block.timestamp works.

### Language Specific

Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of private or delete.

### Coding Style

Coding Style findings usually do not affect the generated byte-code but rather comment on how to make the codebase more legible and, as a result, easily maintainable.

## Checksum Calculation Method

The “Checksum” field in the “Audit Scope” section is calculated as the SHA-256 (Secure Hash Algorithm 2 with digest size of 256 bits) digest of the content of each file hosted in the listed source repository under the specified commit.

The result is hexadecimal encoded and is the same as the output of the Linux “sha256sum” command against the target file.

## Disclaimer

Copied from https://leaderboard.certik.io/projects/aave

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you (“Customer” or the “Company”) in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without CertiK’s prior written consent in each instance.

This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts CertiK to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology. Blockchain technology and cryptographic assets present a high level of ongoing risk. CertiK’s position is that each company and individual are responsible for their own due diligence and continuous security. CertiK’s goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by CertiK is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.