VRust

Security Assessment

O2Lab VRust Team

11/04/2022 16:38:58

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_token_lending
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	17
Major	0
Medium	0
Minor	0
Informational	0
Discussion	0

Findings

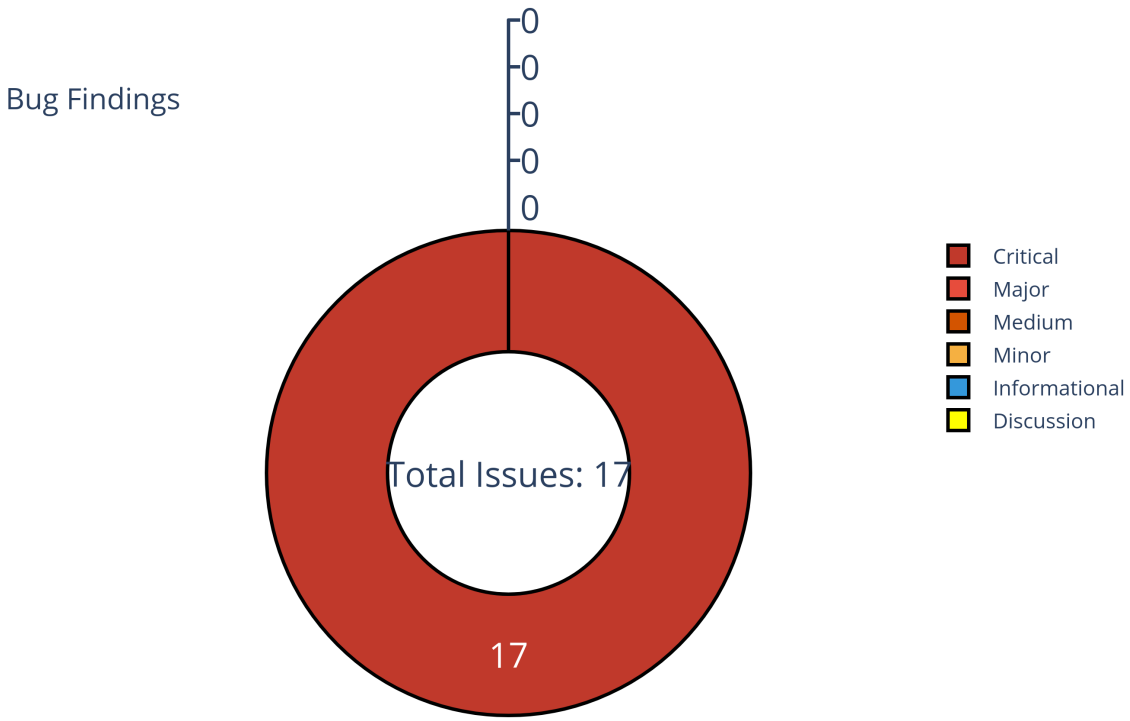

Figure 1: Findings

Finding Statistic

Category	Count
IntegerFlow	4
MissingKeyCheck	13

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	MissingKeyCheck	Critical	UnResolved
8	MissingKeyCheck	Critical	UnResolved
9	MissingKeyCheck	Critical	UnResolved
10	MissingKeyCheck	Critical	UnResolved
11	MissingKeyCheck	Critical	UnResolved
12	MissingKeyCheck	Critical	UnResolved
13	MissingKeyCheck	Critical	UnResolved
14	MissingKeyCheck	Critical	UnResolved
15	MissingKeyCheck	Critical	UnResolved
16	MissingKeyCheck	Critical	UnResolved

Issue: 0: IntegerFlow

Category	Severity	Status
IntegerFlow	Critical	UnResolved

- Location

token-lending/program/src/math/rate.rs:56:25: 56:56

```
56 percent as u64 * PERCENT_SCALER
57
```

- Code Context

Vulnerability at Line: 56

```
55 pub fn from_percent(percent: u8) -> Self {
56         Self(U128::from(percent as u64 * PERCENT_SCALER))
57     }
58
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪ 1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪ }
2     fn entrypoint::process_instruction(){//
      ↪ token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
          ↪ token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_flash_loan(){//
              ↪ token-lending/program/src/processor.rs:1521:1: 1697:2 }
5                     fn
                      ↪ state::reserve::ReserveFees::calculate_flash_loan_fees(){//
                      ↪ token-lending/program/src/state/reserve.rs:643:5:
                      ↪ 652:6 }
6                         fn state::reserve::ReserveFees::calculate_fees(){//
                          ↪ token-
                          ↪ lending/program/src/state/reserve.rs:654:5:
                          ↪ 701:6 }
```

```
7                                   fn math::rate::Rate::from_percent(){// token-
                                    ↪  lending/program/src/math/rate.rs:55:5: 57:6
                                    ↪  }
8
```

- description:
- link:
- alleviation:

Issue: 1: IntegerFlow

Category	Severity	Status
IntegerFlow	Critical	UnResolved

- Location

/home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/uint-0.9.1/src/uint.rs:547:9: 547:50

```
547 ((arr[1] as u128) << 64) + arr[0] as u128
548
```

- Code Context

– Function Definition:

```
540 fn try_from(u: $name) -> $crate::core_::result::Result<u128, &'static str>
541
```

Vulnerability at Line: 540

```
540 fn try_from(u: $name) -> $crate::core_::result::Result<u128, &'static str>
    ↪  {                let $name(arr) = u;             for i in 2..$n_words {
    ↪  if arr[i] != 0 {                         return Err("integer overflow
    ↪  when casting to u128");                    }               }
    ↪  Ok(((arr[1] as u128) << 64) + arr[0] as u128)           }
541
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪  }
2     fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_refresh_obligation(){//
               ↪  token-lending/program/src/processor.rs:680:1: 801:2 }
```

```
5                   fn
                    ↪ state::reserve::Reserve::collateral_exchange_rate(){//
                    ↪ token-lending/program/src/state/reserve.rs:122:5:
                    ↪ 125:6 }
6                       fn
                        ↪ state::reserve::ReserveCollateral::exchange_rate(){//
                        ↪ token-
                        ↪ lending/program/src/state/reserve.rs:526:5:
                        ↪ 538:6 }
7                           fn <math::rate::Rate as
                            ↪ std::convert::TryFrom<math::decimal::Decimal>>::try_from
                            ↪ token-
                            ↪ lending/program/src/math/rate.rs:107:5:
                            ↪ 109:6 }
8       fn math::decimal::Decimal::to_scaled_val(){//
        ↪ token-lending/program/src/math/decimal.rs:60:5: 62:6 }
9            fn math::decimal::<impl
             ↪ std::convert::TryFrom<math::decimal::U192> for
             ↪ u128>::try_from(){//
             ↪ /home/yifei/.cargo/registry/src/github.com-
             ↪ 1ecc6299db9ec823/uint-0.9.1/src/uint.rs:540:5: 548:6
             ↪ }
10
```

- description:
- link:
- alleviation:

Issue: 2: IntegerFlow

Category	Severity	Status
IntegerFlow	Critical	UnResolved

- Location

token-lending/program/src/math/rate.rs:80:13: 80:21

```
80 exp /= 2
81
```

- Code Context

Vulnerability at Line: 80

```
75         } else {
76             Rate(Self::wad())
77         };
78
79         while exp > 0 {
80             exp /= 2;
81             base = base.try_mul(base)?;
82
83             if exp % 2 != 0 {
84                 ret = ret.try_mul(base)?;
85
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪  }
2     fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_refresh_reserve(){//
               ↪  token-lending/program/src/processor.rs:409:1: 432:2 }
5                 fn state::reserve::Reserve::accrue_interest(){//
                   ↪  token-lending/program/src/state/reserve.rs:128:5:
                   ↪  136:6 }
```

```
6                       fn
                        ↪ state::reserve::ReserveLiquidity::compound_interest(){//
                        ↪ token-
                        ↪ lending/program/src/state/reserve.rs:451:5:
                        ↪ 467:6 }
7                       fn math::rate::Rate::try_pow(){// token-
                        ↪  lending/program/src/math/rate.rs:71:5: 89:6
                        ↪  }
8
```

- description:
- link:
- alleviation:

Issue: 3: IntegerFlow

Category	Severity	Status
IntegerFlow	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1734:27: 1734:41

```
1734 start + length
1735
```

- Code Context

– Function Definition:

```
1724 fn get_pyth_product_quote_currency(pyth_product: &pyth::Product) ->
     ↪  Result<[u8; 32], ProgramError>
1725
```

Vulnerability at Line: 1734

```
1729     while start < pyth::PROD_ATTR_SIZE {
1730         let mut length = pyth_product.attr[start] as usize;
1731         start += 1;
1732
1733         if length == LEN {
1734             let mut end = start + length;
1735             if end > pyth::PROD_ATTR_SIZE {
1736                 msg!("Pyth product attribute key length too long");
1737                 return Err(LendingError::InvalidOracleConfig.into());
1738             }
1739
```

Other Use Case for Variable: start + length

```
1746                end = start + length;
```

- Call Stack

```
1  fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
   ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
   ↪  }
2      fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3          fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4              fn processor::process_init_reserve(){//
               ↪  token-lending/program/src/processor.rs:166:1: 407:2 }
5                  fn processor::get_pyth_product_quote_currency(){//
                   ↪  token-lending/program/src/processor.rs:1724:1:
                   ↪  1764:2 }
6
```

- description:
- link:
- alleviation:

Issue: 4: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1547:49: 1547:82

```
1547 lending_market_info.data.borrow()
1548
```

- Code Context

– Function Definition:

```
1521 fn process_flash_loan(
1522     program_id: &Pubkey,
1523     liquidity_amount: u64,
1524     accounts: &[AccountInfo],
1525 ) -> ProgramResult
1526
```

Vulnerability at Line: 1547

```
1542     if program_id == flash_loan_receiver_program_id.key {
1543         msg!("Lending program cannot be used as the flash loan receiver
             ↪  program provided");
1544         return Err(LendingError::InvalidFlashLoanReceiverProgram.into());
1545     }
1546
1547     let lending_market =
         ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
1548     if lending_market_info.owner != program_id {
1549         return Err(LendingError::InvalidAccountOwner.into());
1550     }
1551     if &lending_market.token_program_id != token_program_id.key {
1552
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_flash_loan(){// token-lending/program/src/processor.rs:1521:1: 1697:2 }

```

- description:
- link:
- alleviation:

Issue: 5: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1353:49: 1353:82

```
1353 lending_market_info.data.borrow()
1354
```

- Code Context

– Function Definition:

```
1329 fn process_liquidate_obligation(
1330     program_id: &Pubkey,
1331     liquidity_amount: u64,
1332     accounts: &[AccountInfo],
1333 ) -> ProgramResult
1334
```

Vulnerability at Line: 1353

```
1348     let lending_market_authority_info =
         ↪  next_account_info(account_info_iter)?;
1349     let user_transfer_authority_info =
         ↪  next_account_info(account_info_iter)?;
1350     let clock =
         ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
1351     let token_program_id = next_account_info(account_info_iter)?;
1352
1353     let lending_market =
         ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
1354     if lending_market_info.owner != program_id {
1355         msg!("Lending market provided is not owned by the lending
             ↪  program");
1356         return Err(LendingError::InvalidAccountOwner.into());
```

```
1357     }
1358
```

- Call Stack

```
1  fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
   ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
   ↪  }
2      fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3          fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4              fn processor::process_liquidate_obligation(){//
               ↪  token-lending/program/src/processor.rs:1329:1: 1518:2 }
5
```

- description:
- link:
- alleviation:

Issue: 6: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1245:49: 1245:82

```
1245 lending_market_info.data.borrow()
1246
```

- Code Context

– Function Definition:

```
1225 fn process_repay_obligation_liquidity(
1226     program_id: &Pubkey,
1227     liquidity_amount: u64,
1228     accounts: &[AccountInfo],
1229 ) -> ProgramResult
1230
```

Vulnerability at Line: 1245

```
1240     let lending_market_info = next_account_info(account_info_iter)?;
1241     let user_transfer_authority_info =
         ↪  next_account_info(account_info_iter)?;
1242     let clock =
         ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
1243     let token_program_id = next_account_info(account_info_iter)?;
1244
1245     let lending_market =
         ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
1246     if lending_market_info.owner != program_id {
1247         msg!("Lending market provided is not owned by the lending
             ↪  program");
1248         return Err(LendingError::InvalidAccountOwner.into());
1249     }
1250
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_repay_obligation_liquidity(){// token-lending/program/src/processor.rs:1225:1: 1326:2 }

```

- description:
- link:
- alleviation:

Issue: 7: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1075:49: 1075:82

```
1075 lending_market_info.data.borrow()
1076
```

- Code Context

– Function Definition:

```
1053 fn process_borrow_obligation_liquidity(
1054     program_id: &Pubkey,
1055     liquidity_amount: u64,
1056     accounts: &[AccountInfo],
1057 ) -> ProgramResult
1058
```

Vulnerability at Line: 1075

```
1070     let lending_market_authority_info =
         ↪  next_account_info(account_info_iter)?;
1071     let obligation_owner_info = next_account_info(account_info_iter)?;
1072     let clock =
         ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
1073     let token_program_id = next_account_info(account_info_iter)?;
1074
1075     let lending_market =
         ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
1076     if lending_market_info.owner != program_id {
1077         msg!("Lending market provided is not owned by the lending
             ↪  program");
1078         return Err(LendingError::InvalidAccountOwner.into());
1079     }
1080
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_borrow_obligation_liquidity(){// token-lending/program/src/processor.rs:1053:1: 1222:2 }

```

- description:
- link:
- alleviation:

Issue: 8: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:921:49: 921:82

```
921 lending_market_info.data.borrow()
922
```

- Code Context

– Function Definition:

```
900 fn process_withdraw_obligation_collateral(
901     program_id: &Pubkey,
902     collateral_amount: u64,
903     accounts: &[AccountInfo],
904 ) -> ProgramResult
905
```

Vulnerability at Line: 921

```
916     let lending_market_authority_info =
        ↪  next_account_info(account_info_iter)?;
917     let obligation_owner_info = next_account_info(account_info_iter)?;
918     let clock =
        ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
919     let token_program_id = next_account_info(account_info_iter)?;
920
921     let lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
922     if lending_market_info.owner != program_id {
923         msg!("Lending market provided is not owned by the lending
            ↪  program");
924         return Err(LendingError::InvalidAccountOwner.into());
925     }
926
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_withdraw_obligation_collateral(){// token-lending/program/src/processor.rs:900:1: 1050:2 }

```

- description:
- link:
- alleviation:

Issue: 9: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:825:49: 825:82

```
825 lending_market_info.data.borrow()
826
```

- Code Context

– Function Definition:

```
804 fn process_deposit_obligation_collateral(
805     program_id: &Pubkey,
806     collateral_amount: u64,
807     accounts: &[AccountInfo],
808 ) -> ProgramResult
809
```

Vulnerability at Line: 825

```
820     let obligation_owner_info = next_account_info(account_info_iter)?;
821     let user_transfer_authority_info =
        ↪  next_account_info(account_info_iter)?;
822     let clock =
        ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
823     let token_program_id = next_account_info(account_info_iter)?;
824
825     let lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
826     if lending_market_info.owner != program_id {
827         msg!("Lending market provided is not owned by the lending
            ↪  program");
828         return Err(LendingError::InvalidAccountOwner.into());
829     }
830
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_deposit_obligation_collateral(){// token-lending/program/src/processor.rs:804:1: 897:2 }

```

- description:
- link:
- alleviation:

Issue: 10: 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::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪ 1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪ }
2     fn entrypoint::process_instruction(){//
      ↪ token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
          ↪ token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_init_obligation(){//
              ↪ token-lending/program/src/processor.rs:637:1: 678:2 }
5                 fn processor::assert_rent_exempt(){//
                  ↪ token-lending/program/src/processor.rs:1699:1:
                  ↪ 1706:2 }
6                     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
                      ↪ }
```

7

- description:
- link:
- alleviation:

Issue: 11: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:1711:43: 1711:69

```
1711 account_info.data.borrow()
1712
```

- Code Context

– Function Definition:

```
1708 fn assert_uninitialized<T: Pack + IsInitialized>(
1709     account_info: &AccountInfo,
1710 ) -> Result<T, ProgramError>
1711
```

Vulnerability at Line: 1711

```
1708 fn assert_uninitialized<T: Pack + IsInitialized>(
1709     account_info: &AccountInfo,
1710 ) -> Result<T, ProgramError> {
1711     let account: T = T::unpack_unchecked(&account_info.data.borrow())?;
1712     if account.is_initialized() {
1713         Err(LendingError::AlreadyInitialized.into())
1714     } else {
1715         Ok(account)
1716
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪  }
2     fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
```

```
        fn processor::process_instruction(){//
        ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_init_obligation(){//
            ↪  token-lending/program/src/processor.rs:637:1: 678:2 }
                    fn processor::assert_uninitialized(){//
                    ↪  token-lending/program/src/processor.rs:1708:1:
                    ↪  1717:2 }

```

- description:
- link:
- alleviation:

Issue: 12: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:653:49: 653:82

```
653 lending_market_info.data.borrow()
654
```

- Code Context

– Function Definition:

```
637 fn process_init_obligation(program_id: &Pubkey, accounts: &[AccountInfo])
    ↪  -> ProgramResult
638
```

Vulnerability at Line: 653

```
648     if obligation_info.owner != program_id {
649         msg!("Obligation provided is not owned by the lending program");
650         return Err(LendingError::InvalidAccountOwner.into());
651     }
652
653     let lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
654     if lending_market_info.owner != program_id {
655         msg!("Lending market provided is not owned by the lending
            ↪  program");
656         return Err(LendingError::InvalidAccountOwner.into());
657     }
658
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪ 1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪ }
2     fn entrypoint::process_instruction(){//
      ↪ token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
          ↪ token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_init_obligation(){//
              ↪ token-lending/program/src/processor.rs:637:1: 678:2 }
5
```

- description:
- link:
- alleviation:

Issue: 13: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:558:49: 558:82

```
558 lending_market_info.data.borrow()
559
```

- Code Context

– Function Definition:

```
536 fn process_redeem_reserve_collateral(
537     program_id: &Pubkey,
538     collateral_amount: u64,
539     accounts: &[AccountInfo],
540 ) -> ProgramResult
541
```

Vulnerability at Line: 558

```
553     let lending_market_authority_info =
        ↪  next_account_info(account_info_iter)?;
554     let user_transfer_authority_info =
        ↪  next_account_info(account_info_iter)?;
555     let clock =
        ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
556     let token_program_id = next_account_info(account_info_iter)?;
557
558     let lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
559     if lending_market_info.owner != program_id {
560         msg!("Lending market provided is not owned by the lending
            ↪  program");
561         return Err(LendingError::InvalidAccountOwner.into());
```

```
562     }
563
```

- Call Stack

```
1  fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
   ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
   ↪  }
2      fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3          fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4              fn processor::process_redeem_reserve_collateral(){//
               ↪  token-lending/program/src/processor.rs:536:1: 634:2 }
5
```

- description:
- link:
- alleviation:

Issue: 14: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:456:49: 456:82

```
456 lending_market_info.data.borrow()
457
```

- Code Context

– Function Definition:

```
434 fn process_deposit_reserve_liquidity(
435     program_id: &Pubkey,
436     liquidity_amount: u64,
437     accounts: &[AccountInfo],
438 ) -> ProgramResult
439
```

Vulnerability at Line: 456

```
451     let lending_market_authority_info =
        ↪  next_account_info(account_info_iter)?;
452     let user_transfer_authority_info =
        ↪  next_account_info(account_info_iter)?;
453     let clock =
        ↪  &Clock::from_account_info(next_account_info(account_info_iter)?)?;
454     let token_program_id = next_account_info(account_info_iter)?;
455
456     let lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
457     if lending_market_info.owner != program_id {
458         msg!("Lending market provided is not owned by the lending
            ↪  program");
459         return Err(LendingError::InvalidAccountOwner.into());
```

```
460     }
461
```

- Call Stack

```
1  fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
   ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
   ↪  }
2      fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3          fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4              fn processor::process_deposit_reserve_liquidity(){//
               ↪  token-lending/program/src/processor.rs:434:1: 534:2 }
5
```

- description:
- link:
- alleviation:

Issue: 15: 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:102:9: 103:26

```
102 self.data
103             .try_borrow()
104
```

- Code Context

Vulnerability at Line: 102

```
101 pub fn try_borrow_data(&self) -> Result<Ref<&mut [u8]>, ProgramError> {
102         self.data
103             .try_borrow()
104             .map_err(|_| ProgramError::AccountBorrowFailed)
105     }
106
```

- Call Stack

```
1 fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-
  ↪  1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10
  ↪  }
2     fn entrypoint::process_instruction(){//
       ↪  token-lending/program/src/entrypoint.rs:12:1: 23:2 }
3         fn processor::process_instruction(){//
           ↪  token-lending/program/src/processor.rs:33:1: 103:2 }
4             fn processor::process_refresh_reserve(){//
               ↪  token-lending/program/src/processor.rs:409:1: 432:2 }
5                     fn processor::get_pyth_price(){//
                       ↪  token-lending/program/src/processor.rs:1766:1:
                       ↪  1820:2 }
```

```
6                       fn
                        ↪ solana_program::account_info::AccountInfo::<'a>::try_borrow_
                        ↪ /home/yifei/.cargo/registry/src/github.com-
                        ↪ 1ecc6299db9ec823/solana-program-
                        ↪ 1.9.9/src/account_info.rs:101:5: 105:6
                        ↪ }
7
```

- description:
- link:
- alleviation:

Issue: 16: MissingKeyCheck

Category	Severity	Status
MissingKeyCheck	Critical	UnResolved

- Location

token-lending/program/src/processor.rs:146:53: 146:86

```
146 lending_market_info.data.borrow()
147
```

- Code Context

– Function Definition:

```
137 fn process_set_lending_market_owner(
138     program_id: &Pubkey,
139     new_owner: Pubkey,
140     accounts: &[AccountInfo],
141 ) -> ProgramResult
142
```

Vulnerability at Line: 146

```
141 ) -> ProgramResult {
142     let account_info_iter = &mut accounts.iter();
143     let lending_market_info = next_account_info(account_info_iter)?;
144     let lending_market_owner_info = next_account_info(account_info_iter)?;
145
146     let mut lending_market =
        ↪  LendingMarket::unpack(&lending_market_info.data.borrow())?;
147     if lending_market_info.owner != program_id {
148         msg!("Lending market provided is not owned by the lending
            ↪  program");
149         return Err(LendingError::InvalidAccountOwner.into());
150     }
151
```

- Call Stack

```
fn entrypoint::entrypoint(){// /home/yifei/.cargo/registry/src/github.com-1ecc6299db9ec823/solana-program-1.9.9/src/entrypoint.rs:120:9: 127:10 }
    fn entrypoint::process_instruction(){// token-lending/program/src/entrypoint.rs:12:1: 23:2 }
        fn processor::process_instruction(){// token-lending/program/src/processor.rs:33:1: 103:2 }
            fn processor::process_set_lending_market_owner(){// token-lending/program/src/processor.rs:137:1: 164:2 }

```

- 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.