# VRust

## Security Assessment

O2Lab VRust Team

11/04/2022 17:08:13

# 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 |
| 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 | 13 |
| Major | 0 |
| Medium | 0 |
| Minor | 0 |
| Informational | 0 |
| Discussion | 0 |

## Findings

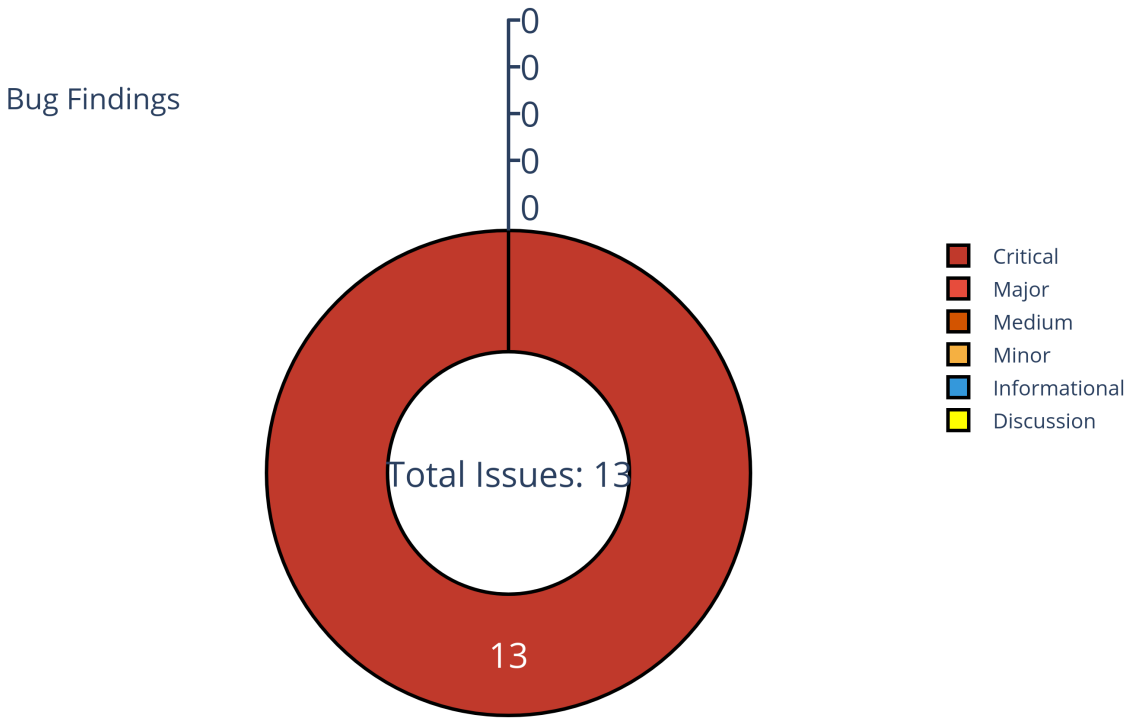


**Figure 1:** Findings

## Finding Statistic

| Category | Count |
|---|---|
| IntegerFlow | 1 |
| MissingKeyCheck | 12 |

| ID | Category | Severity | Status |
|---|---|---|---|
| 0 | IntegerFlow | Critical | UnResolved |
| 1 | MissingKeyCheck | Critical | UnResolved |
| 2 | MissingKeyCheck | Critical | UnResolved |
| 3 | MissingKeyCheck | 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 |

## Issue: 0: IntegerFlow

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

- Location

token/program/src/lib.rs:35:47: 35:59

```
35 decimals + 1
36
```

- Code Context

Vulnerability at Line: 34

```
31 pub fn amount_to_ui_amount_string(amount: u64, decimals: u8) -> String {
32     let decimals = decimals as usize;
33     if decimals > 0 {
34         // Left-pad zeros to decimals + 1, so we at least have an integer
       ↪  zero
35         let mut s = format!("{:01$}", amount, decimals + 1);
36         // Add the decimal point (Sorry, "," locales!)
37         s.insert(s.len() - decimals, '.');
38         s
39
```

Other Use Case for Variable: decimals + 1

```
35         let mut s = format!("{:01$}", amount, decimals + 1);
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
       ↪  token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_amount_to_ui_amount(){//
           ↪  token/program/src/processor.rs:792:5: 807:6 }
```

```
4               fn amount_to_ui_amount_string_trimmed(){//
                ↪  token/program/src/lib.rs:46:1: 53:2 }
5                   fn amount_to_ui_amount_string(){//
                    ↪  token/program/src/lib.rs:31:1: 42:2 }
6
```

- description:
- link:
- alleviation:

## Issue: 1: MissingKeyCheck

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

- Location

token/program/src/processor.rs:819:34: 819:61

```
819 mint_info.data.borrow_mut()
820
```

- Code Context

Vulnerability at Line: 819

```
814     ) -> ProgramResult {
815         let account_info_iter = &mut accounts.iter();
816         let mint_info = next_account_info(account_info_iter)?;
817         Self::check_account_owner(program_id, mint_info)?;
818
819         let mint = Mint::unpack(&mint_info.data.borrow_mut())
820             .map_err(|_|
    ↪  Into::<ProgramError>::into(TokenError::InvalidMint))?;
821         let amount = try_ui_amount_into_amount(ui_amount.to_string(),
            ↪  mint.decimals)?;
822
823         set_return_data(&amount.to_le_bytes());
824
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
       ↪  token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_ui_amount_to_amount(){//
           ↪  token/program/src/processor.rs:810:5: 825:6 }
4
```

- description:
- link:
- alleviation:

## Issue: 2: MissingKeyCheck

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

- Location

token/program/src/processor.rs:783:50: 783:82

```
token_account_info.data.borrow()

```

- Code Context

Vulnerability at Line: 783

```
pub fn process_initialize_immutable_owner(accounts: &[AccountInfo]) ->
    ProgramResult {
        let account_info_iter = &mut accounts.iter();
        let token_account_info = next_account_info(account_info_iter)?;
        let account =
            Account::unpack_unchecked(&token_account_info.data.borrow())?;
        if account.is_initialized() {
            return Err(TokenError::AlreadyInUse.into());
        }
        msg!("Please upgrade to SPL Token 2022 for immutable owner
            support");

```

- Call Stack

```
fn entrypoint::process_instruction(){//
    token/program/src/entrypoint.rs:10:1: 21:2 }
    fn processor::Processor::process(){//
        token/program/src/processor.rs:828:5: 944:6 }
        fn processor::Processor::process_initialize_immutable_owner(){//
            token/program/src/processor.rs:780:5: 789:6 }

```

- description:

- link:
- alleviation:

## Issue: 3: MissingKeyCheck

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

- Location

token/program/src/processor.rs:44:48: 44:71

```
44 mint_info.data.borrow()
45
```

- Code Context

– Function Definition:

```
28 fn _process_initialize_mint(
29         accounts: &[AccountInfo],
30         decimals: u8,
31         mint_authority: Pubkey,
32         freeze_authority: COption<Pubkey>,
33         rent_sysvar_account: bool,
34     ) -> ProgramResult
35
```

Vulnerability at Line: 44

```
39             Rent::from_account_info(next_account_info(account_info_iter)?)?
40         } else {
41             Rent::get()?
42         };
43
44         let mut mint = Mint::unpack_unchecked(&mint_info.data.borrow())?;
45         if mint.is_initialized {
46             return Err(TokenError::AlreadyInUse.into());
47         }
48
49
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
       ↪  token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_initialize_mint2(){//
           ↪  token/program/src/processor.rs:74:5: 81:6 }
4             fn processor::Processor::_process_initialize_mint(){//
               ↪  token/program/src/processor.rs:28:5: 61:6 }
5
```

- description:
- link:
- alleviation:

## Issue: 4: 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(){//
  ↪  token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
       ↪  token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_initialize_mint2(){//
           ↪  token/program/src/processor.rs:74:5: 81:6 }
4             fn processor::Processor::_process_initialize_mint(){//
               ↪  token/program/src/processor.rs:28:5: 61:6 }
5                     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
                       ↪  }
6
```

- description:
- link:
- alleviation:

## Issue: 5: MissingKeyCheck

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

- Location

token/program/src/processor.rs:183:56: 183:83

```
183 multisig_info.data.borrow()
184
```

- Code Context

– Function Definition:

```
169 fn _process_initialize_multisig(
170         accounts: &[AccountInfo],
171         m: u8,
172         rent_sysvar_account: bool,
173     ) -> ProgramResult
174
```

Vulnerability at Line: 183

```
178             Rent::from_account_info(next_account_info(account_info_iter)?)?
179         } else {
180             Rent::get()?
181         };
182
183         let mut multisig =
            ↪ Multisig::unpack_unchecked(&multisig_info.data.borrow())?;
184         if multisig.is_initialized {
185             return Err(TokenError::AlreadyInUse.into());
186         }
187
188
```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  token/program/src/entrypoint.rs:10:1: 21:2 }
    fn processor::Processor::process(){//
    ↪  token/program/src/processor.rs:828:5: 944:6 }
        fn processor::Processor::process_initialize_multisig2(){//
        ↪  token/program/src/processor.rs:217:5: 219:6 }
            fn processor::Processor::_process_initialize_multisig(){//
            ↪  token/program/src/processor.rs:169:5: 209:6 }

```

- description:
- link:
- alleviation:

## Issue: 6: MissingKeyCheck

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

- Location

token/program/src/processor.rs:104:54: 104:84

```
104 new_account_info.data.borrow()
105
```

- Code Context

– Function Definition:

```
83  fn _process_initialize_account(
84          program_id: &Pubkey,
85          accounts: &[AccountInfo],
86          owner: Option<&Pubkey>,
87          rent_sysvar_account: bool,
88      ) -> ProgramResult
89
```

Vulnerability at Line: 104

```
99              Rent::from_account_info(next_account_info(account_info_iter)?)?
100         } else {
101             Rent::get()?
102         };
103
104         let mut account =
            ↪ Account::unpack_unchecked(&new_account_info.data.borrow())?;
105         if account.is_initialized() {
106             return Err(TokenError::AlreadyInUse.into());
107         }
108
109
```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  token/program/src/entrypoint.rs:10:1: 21:2 }
    fn processor::Processor::process(){//
    ↪  token/program/src/processor.rs:828:5: 944:6 }
        fn processor::Processor::process_initialize_account3(){//
        ↪  token/program/src/processor.rs:161:5: 167:6 }
            fn processor::Processor::_process_initialize_account(){//
            ↪  token/program/src/processor.rs:83:5: 141:6 }

```

- description:
- link:
- alleviation:

## Issue: 7: MissingKeyCheck

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

- Location

token/program/src/processor.rs:586:51: 586:84

```
source_account_info.data.borrow()

```

- Code Context

Vulnerability at Line: 586

```

        let source_account_info = next_account_info(account_info_iter)?;
        let mint_info = next_account_info(account_info_iter)?;
        let authority_info = next_account_info(account_info_iter)?;

        let mut source_account =
        ↪  Account::unpack(&source_account_info.data.borrow())?;
        let mut mint = Mint::unpack(&mint_info.data.borrow())?;

        if source_account.is_frozen() {
            return Err(TokenError::AccountFrozen.into());

```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  token/program/src/entrypoint.rs:10:1: 21:2 }
    fn processor::Processor::process(){//
    ↪  token/program/src/processor.rs:828:5: 944:6 }
        fn processor::Processor::process_burn(){//
        ↪  token/program/src/processor.rs:574:5: 654:6 }

```

- description:

- link:
- alleviation:

## Issue: 8: MissingKeyCheck

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

- Location

token/program/src/processor.rs:523:49: 523:80

```
523 dest_account_info.data.borrow()
524
```

- Code Context

Vulnerability at Line: 523

```
518         let account_info_iter = &mut accounts.iter();
519         let mint_info = next_account_info(account_info_iter)?;
520         let dest_account_info = next_account_info(account_info_iter)?;
521         let owner_info = next_account_info(account_info_iter)?;
522
523         let mut dest_account =
            ↪ Account::unpack(&dest_account_info.data.borrow())?;
524         if dest_account.is_frozen() {
525             return Err(TokenError::AccountFrozen.into());
526         }
527
528
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪ token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
      ↪ token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_mint_to(){//
          ↪ token/program/src/processor.rs:512:5: 571:6 }
4
```

- description:

- link:
- alleviation:

## Issue: 9: MissingKeyCheck

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

- Location

token/program/src/processor.rs:353:51: 353:84

```
source_account_info.data.borrow()

```

- Code Context

Vulnerability at Line: 353

```
            None
        };
        let delegate_info = next_account_info(account_info_iter)?;
        let owner_info = next_account_info(account_info_iter)?;

        let mut source_account =
        ↪  Account::unpack(&source_account_info.data.borrow())?;

        if source_account.is_frozen() {
            return Err(TokenError::AccountFrozen.into());
        }

```

- Call Stack

```
fn entrypoint::process_instruction(){//
↪  token/program/src/entrypoint.rs:10:1: 21:2 }
    fn processor::Processor::process(){//
    ↪  token/program/src/processor.rs:828:5: 944:6 }
        fn processor::Processor::process_approve(){//
        ↪  token/program/src/processor.rs:335:5: 383:6 }

```

- description:

- link:
- alleviation:

## Issue: 10: MissingKeyCheck

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

- Location

token/program/src/processor.rs:241:51: 241:84

```
241 source_account_info.data.borrow()
242
```

- Code Context

Vulnerability at Line: 241

```
236         };
237
238         let dest_account_info = next_account_info(account_info_iter)?;
239         let authority_info = next_account_info(account_info_iter)?;
240
241         let mut source_account =
            ↪ Account::unpack(&source_account_info.data.borrow())?;
242         let mut dest_account =
            ↪ Account::unpack(&dest_account_info.data.borrow())?;
243
244         if source_account.is_frozen() || dest_account.is_frozen() {
245             return Err(TokenError::AccountFrozen.into());
246
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪ token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
      ↪ token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_transfer(){//
          ↪ token/program/src/processor.rs:222:5: 332:6 }
4
```

- description:
- link:
- alleviation:

## Issue: 11: MissingKeyCheck

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

- Location

token/program/src/processor.rs:706:51: 706:84

```
706 source_account_info.data.borrow()
707
```

- Code Context

Vulnerability at Line: 706

```
701         let account_info_iter = &mut accounts.iter();
702         let source_account_info = next_account_info(account_info_iter)?;
703         let mint_info = next_account_info(account_info_iter)?;
704         let authority_info = next_account_info(account_info_iter)?;
705
706         let mut source_account =
            ↪ Account::unpack(&source_account_info.data.borrow())?;
707         if freeze && source_account.is_frozen() || !freeze &&
            ↪ !source_account.is_frozen() {
708             return Err(TokenError::InvalidState.into());
709         }
710         if source_account.is_native() {
711
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪ token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
      ↪ token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_toggle_freeze_account(){//
          ↪ token/program/src/processor.rs:696:5: 737:6 }
4
```

- description:
- link:
- alleviation:

## Issue: 12: MissingKeyCheck

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

- Location

token/program/src/processor.rs:390:51: 390:84

```
390 source_account_info.data.borrow()
391
```

- Code Context

Vulnerability at Line: 390

```
386 pub fn process_revoke(program_id: &Pubkey, accounts: &[AccountInfo]) ->
    ↪  ProgramResult {
387         let account_info_iter = &mut accounts.iter();
388         let source_account_info = next_account_info(account_info_iter)?;
389
390         let mut source_account =
            ↪  Account::unpack(&source_account_info.data.borrow())?;
391
392         let owner_info = next_account_info(account_info_iter)?;
393
394         if source_account.is_frozen() {
395
```

- Call Stack

```
1 fn entrypoint::process_instruction(){//
  ↪  token/program/src/entrypoint.rs:10:1: 21:2 }
2     fn processor::Processor::process(){//
       ↪  token/program/src/processor.rs:828:5: 944:6 }
3         fn processor::Processor::process_revoke(){//
           ↪  token/program/src/processor.rs:386:5: 411:6 }
4
```

- 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.