HALBORN

DeBridge - DLN-Solana Diff v0.1.3 - v0.2.0

Solana Program Security Audit

Prepared by: **Halborn**

Date of Engagement: **December 14th, 2022 - December 15th, 2022**

Visit: **Halborn.com**

DOCUMENT REVISION HISTORY

VERSION	MODIFICATION	DATE	AUTHOR
0.1	Document Creation	12/15/2022	Mateusz Garncarek
0.2	Final Draft	12/15/2022	Mateusz Garncarek
0.3	Draft Review	12/16/2022	Isabel Burruezo
0.4	Draft Review	12/16/2022	Piotr Cielas
0.5	Draft Review	12/16/2022	Gabi Urrutia
1.0	Remediation Plan	01/12/2023	Michael Smith
1.1	Remediation Plan Review	01/12/2023	Isabel Burruezo
1.2	Remediation Plan Review	01/12/2023	Piotr Cielas
1.3	Remediation Plan Review	01/12/2023	Gabi Urrutia

CONTACTS

CONTACT	COMPANY	EMAIL
Rob Behnke	Halborn	Rob.Behnke@halborn.com
Steven Walbroehl	Halborn	Steven.Walbroehl@halborn.com
Gabi Urrutia	Halborn	Gabi.Urrutia@halborn.com
Piotr Cielas	Halborn	Piotr.Cielas@halborn.com
Isabel Burruezo	Halborn	Isabel.Burruezo@halborn.com
Mateusz Garncarek	Halborn	Mateusz.Garncarek@halborn.com
Michael Smith	Halborn	Michael.Smith@halborn.com

EXECUTIVE OVERVIEW

1.1 INTRODUCTION

DeBridge engaged Halborn to conduct a security audit on their Solana programs beginning on December 14th, 2022 and ending on December 15th, 2022 . The security assessment was scoped to the diff between version v0.1.3 and v0.2.0 of deBridge dln-solana GitHub repository. Commit hashes and further details can be found in the Scope section of this report.

DLN-Solana Diff v0.1.3 - v0.2.0 is a protocol built on top of deBridge's infrastructure that allows cross-chain messaging between any smart contracts and programs on any chains. The protocol allows the creation of any cross-chain liquidity transfer order, settlement, and management of orders.

1.2 AUDIT SUMMARY

The team at Halborn was provided 2 days for the engagement and assigned 1 full-time security engineer to audit the security of the programs in scope. The security engineer is a blockchain and smart contract security expert with advanced penetration testing and smart contract hacking skills, and deep knowledge of multiple blockchain protocols.

The purpose of this audit is to:

- Identify potential security issues within the programs

In summary, Halborn did not identify any issues related to changes introduced in the commit in scope.

1.3 TEST APPROACH & METHODOLOGY

Halborn performed a combination of a manual review of the source code and automated security testing to balance efficiency, timeliness, practicality, and accuracy in regard to the scope of the program audit. While manual testing is recommended to uncover flaws in business logic, processes, and implementation; automated testing techniques help enhance coverage of programs and can quickly identify items that do not follow security best practices.

The following phases and associated tools were used throughout the term of the audit:

- Research into the architecture, purpose, and use of the platform.
- Manual program source code review to identify business logic issues.
- Mapping out possible attack vectors
- Thorough assessment of safety and usage of critical Rust variables and functions in scope that could lead to arithmetic vulnerabilities.
- Finding unsafe Rust code usage (cargo-geiger)
- Scanning dependencies for known vulnerabilities (cargo audit).
- Local runtime testing (solana-test-framework)

RISK METHODOLOGY:

Vulnerabilities or issues observed by Halborn are ranked based on the risk assessment methodology by measuring the **LIKELIHOOD** of a security incident and the **IMPACT** should an incident occur. This framework works for communicating the characteristics and impacts of technology vulnerabilities. The quantitative model ensures repeatable and accurate measurement while enabling users to see the underlying vulnerability characteristics that were used to generate the Risk scores. For every vulnerability, a risk

level will be calculated on a scale of 5 to 1 with 5 being the highest likelihood or impact.

RISK SCALE - LIKELIHOOD

5 - Almost certain an incident will occur.
4 - High probability of an incident occurring.
3 - Potential of a security incident in the long term.
2 - Low probability of an incident occurring.
1 - Very unlikely issue will cause an incident.

RISK SCALE - IMPACT

5 - May cause devastating and unrecoverable impact or loss.
4 - May cause a significant level of impact or loss.
3 - May cause a partial impact or loss to many.
2 - May cause temporary impact or loss.
1 - May cause minimal or un-noticeable impact.

The risk level is then calculated using a sum of these two values, creating a value of 10 to 1 with 10 being the highest level of security risk.

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL

10 - CRITICAL
9 - 8 - HIGH
7 - 6 - MEDIUM
5 - 4 - LOW
3 - 1 - VERY LOW AND INFORMATIONAL

1.4 SCOPE

Code repositories:

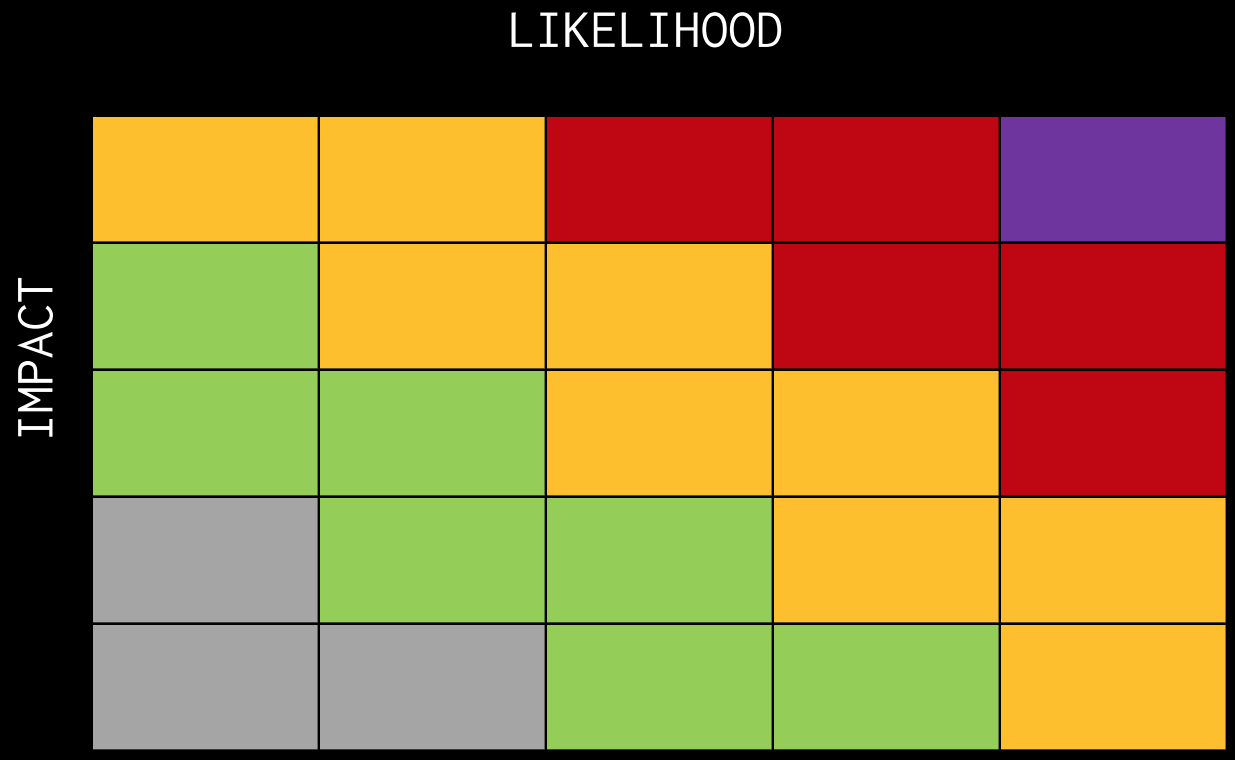
1. deBridge DLN

- Repository: deBridge dln-solana
- Diff in scope:
    1. v0.1.3. . . v0.2.0

Out-of-scope:
- third-party libraries and dependencies
- financial-related attacks

2. ASSESSMENT SUMMARY & FINDINGS OVERVIEW

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
0	0	0	0	0



SECURITY ANALYSIS	RISK LEVEL	REMEDIATION DATE

No vulnerabilities were identified.

MANUAL TESTING

In addition to changing the names of several symbols, the most significant update made to the code was to add an `else` statement to the control flow for the newly introduced role. Due to the minor scope of the changes introduced, in the manual testing phase of the audit the following scenario was identified as the most critical to the security of the program if exploited.

3.1 AUTHORIZATION ISSUES

Description:

The DLN Solana program was tested to ensure the new implementation of `OrderUnlockAuthority` role did not introduce any authorization issues with unlocking mechanisms which could result in loss of funds.

Results:

Only accounts assigned the `OrderUnlockAuthority` or `Taker` roles can execute `unlock` successfully.

```
running 1 test
test authorization_check_user_without_proper_role ... FAILED

failures:

---- authorization_check_user_without_proper_role stdout ----
thread 'authorization_check_user_without_proper_role' panicked at 'called `Result::unwrap()` on an `Err` value: TransactionError(InstructionError(0, Custom(6000)))'
note: run with `RUST_BACKTRACE=1` environment variable to display a backtrace
```

AUTOMATED TESTING

4.1 AUTOMATED ANALYSIS

Description:

Halborn used automated security scanners to assist with the detection of well-known security issues and vulnerabilities. Among the tools used was cargo-audit, a security scanner for vulnerabilities reported to the RustSec Advisory Database. All vulnerabilities published in https://crates.io are stored in a repository named The RustSec Advisory Database. cargo audit is a human-readable version of the advisory database which performs a scanning on Cargo.lock. Security Detections are only in scope. All vulnerabilities shown here were already disclosed in the above report. However, to better assist the developers maintaining this code, the auditors are including the output with the dependencies tree, and this is included in the cargo audit output to better know the dependencies affected by unmaintained and vulnerable crates.

Results:

ID	package	short description
RUSTSEC-2020-0071	time	Potential segfault
RUSTSEC-2021-0139	ansiterm	unmaintained

4.2 UNSAFE RUST CODE DETECTION

Description:

Halborn used automated security scanners to assist with the detection of well-known security issues and vulnerabilities. Among the tools used was cargo-geiger, a security tool that lists statistics related to the usage of unsafe Rust code in a core Rust codebase and all its dependencies.

Results:

dln_dst

```
WARNING: Dependency file was never scanned: /root/.cargo/registry/src/github.com-1ecc6299db9ec823/tiny-bip39-0.8.2/src/langs/spanish.txt

Metric output format: x/y
    x = unsafe code used by the build
    y = total unsafe code found in the crate

Symbols:
    🔒  = No `unsafe` usage found, declares #![forbid(unsafe_code)]
    ?   = No `unsafe` usage found, missing #![forbid(unsafe_code)]
    ☢   = `unsafe` usage found

Functions  Expressions  Impls  Traits  Methods  Dependency

0/0        0/0          0/0    0/0     0/0      ?  dln-dst 0.2.0
0/0        0/0          0/0    0/0     0/0      ?  ├── anchor-spl 0.25.0
0/0        7/7          0/0    0/0     0/0      ☢  │   ├── borsh 0.9.3
0/0        0/0          0/0    0/0     0/0      ?  │   │   ├── borsh-derive 0.9.3
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── borsh-derive-internal 0.9.3
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   ├── proc-macro2 1.0.47
0/0        4/4          0/0    0/0     0/0      ☢  │   │   │   │   │   └── unicode-ident 1.0.5
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │   ├── quote 1.0.21
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   │   └── proc-macro2 1.0.47
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │   └── syn 1.0.103
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │       ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │       ├── quote 1.0.21
0/0        4/4          0/0    0/0     0/0      ☢  │   │   │   │       └── unicode-ident 1.0.5
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── borsh-schema-derive-internal 0.9.3
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │   ├── quote 1.0.21
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │   └── syn 1.0.103
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── proc-macro-crate 0.1.5
0/0        0/0          0/0    0/0     0/0      🔒 │   │   │   │   └── toml 0.5.9
0/0        5/5          0/0    0/0     0/0      ☢  │   │   │   │       └── serde 1.0.147
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │           └── serde_derive 1.0.147
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │               ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │               ├── quote 1.0.21
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │               └── syn 1.0.103
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   ├── proc-macro2 1.0.47
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   └── syn 1.0.103
2/2        1082/1198    19/22  1/1     51/58    ☢  │   │   └── hashbrown 0.11.2
0/0        26/30        0/0    0/0     0/0      ☢  │   │       ├── ahash 0.7.6
2/4        52/175       1/1    0/0     3/3      ☢  │   │       │   ├── getrandom 0.2.8
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   ├── cfg-if 1.0.0
0/21       12/368       0/2    0/0     2/40     ☢  │   │       │   │   └── libc 0.2.137
1/1        76/122       4/8    0/0     2/4      ☢  │   │       │   ├── once_cell 1.16.0
0/16       0/1323       0/0    0/0     0/56     ?  │   │       │   │   └── parking_lot_core 0.9.4
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │       ├── cfg-if 1.0.0
0/21       12/368       0/2    0/0     2/40     ☢  │   │       │   │       ├── libc 0.2.137
0/1        0/399        0/7    0/1     0/13     ?  │   │       │   │       └── smallvec 1.10.0
0/0        5/5          0/0    0/0     0/0      ☢  │   │       │   │           └── serde 1.0.147
0/0        5/5          0/0    0/0     0/0      ☢  │   │       │   └── serde 1.0.147
4/6        437/1158     4/10   1/1     13/26    ☢  │   │       ├── bumpalo 3.11.1
6/6        659/659      5/5    0/0     3/3      ☢  │   │       ├── rayon 1.6.0
```

AUTOMATED TESTING

dln_src

```
WARNING: Dependency file was never scanned: /root/.config/debridge/settings.pubkey
WARNING: Dependency file was never scanned: /root/.cargo/registry/src/github.com-1ecc6299db9ec823/tiny-bip39-0.8.2/src/langs/french.txt

Metric output format: x/y
    x = unsafe code used by the build
    y = total unsafe code found in the crate

Symbols:
    🔒  = No `unsafe` usage found, declares #![forbid(unsafe_code)]
    ?   = No `unsafe` usage found, missing #![forbid(unsafe_code)]
    ☢   = `unsafe` usage found

Functions  Expressions  Impls  Traits  Methods  Dependency

0/0        0/0          0/0    0/0     0/0      ?  dln-src 0.2.0
0/0        0/0          0/0    0/0     0/0      ?  ├── anchor-spl 0.25.0
0/0        7/7          0/0    0/0     0/0      ☢  │   ├── borsh 0.9.3
0/0        0/0          0/0    0/0     0/0      ?  │   │   ├── borsh-derive 0.9.3
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── borsh-derive-internal 0.9.3
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   ├── proc-macro2 1.0.47
0/0        4/4          0/0    0/0     0/0      ☢  │   │   │   │   │   └── unicode-ident 1.0.5
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │   ├── quote 1.0.21
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   │   └── proc-macro2 1.0.47
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │   └── syn 1.0.103
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │       ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │       ├── quote 1.0.21
0/0        4/4          0/0    0/0     0/0      ☢  │   │   │   │       └── unicode-ident 1.0.5
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── borsh-schema-derive-internal 0.9.3
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │   ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │   ├── quote 1.0.21
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │   └── syn 1.0.103
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   ├── proc-macro-crate 0.1.5
0/0        0/0          0/0    0/0     0/0      🔒 │   │   │   │   └── toml 0.5.9
0/0        5/5          0/0    0/0     0/0      ☢  │   │   │   │       └── serde 1.0.147
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │           └── serde_derive 1.0.147
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   │               ├── proc-macro2 1.0.47
0/0        0/0          0/0    0/0     0/0      ?  │   │   │   │               ├── quote 1.0.21
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   │               └── syn 1.0.103
0/0        15/15        0/0    0/0     3/3      ☢  │   │   │   ├── proc-macro2 1.0.47
0/0        69/69        3/3    0/0     2/2      ☢  │   │   │   └── syn 1.0.103
2/2        1082/1198    19/22  1/1     51/58    ☢  │   │   └── hashbrown 0.11.2
0/0        26/30        0/0    0/0     0/0      ☢  │   │       ├── ahash 0.7.6
2/4        52/175       1/1    0/0     3/3      ☢  │   │       │   ├── getrandom 0.2.8
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   ├── cfg-if 1.0.0
0/21       12/368       0/2    0/0     2/40     ☢  │   │       │   │   └── libc 0.2.137
1/1        76/122       4/8    0/0     2/4      ☢  │   │       │   ├── once_cell 1.16.0
0/16       0/1323       0/0    0/0     0/56     ?  │   │       │   │   └── parking_lot_core 0.9.4
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │       ├── cfg-if 1.0.0
0/21       12/368       0/2    0/0     2/40     ☢  │   │       │   │       ├── libc 0.2.137
0/1        0/399        0/7    0/1     0/13     ?  │   │       │   │       └── smallvec 1.10.0
0/0        5/5          0/0    0/0     0/0      ☢  │   │       │   │           └── serde 1.0.147
0/0        5/5          0/0    0/0     0/0      ☢  │   │       │   └── serde 1.0.147
4/6        437/1158     4/10   1/1     13/26    ☢  │   │       ├── bumpalo 3.11.1
6/6        659/659      5/5    0/0     3/3      ☢  │   │       ├── rayon 1.6.0
0/0        453/453      6/6    0/0     6/6      ☢  │   │       │   ├── crossbeam-deque 0.8.2
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   ├── cfg-if 1.0.0
3/3        448/460      11/11  0/0     29/29    ☢  │   │       │   │   ├── crossbeam-epoch 0.9.13
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   │   ├── cfg-if 1.0.0
4/4        94/94        16/16  0/0     3/3      ☢  │   │       │   │   │   ├── crossbeam-utils 0.8.14
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   │   │   └── cfg-if 1.0.0
0/0        0/0          0/0    0/0     0/0      ?  │   │       │   │   │   ├── memoffset 0.7.1
0/0        18/18        1/1    0/0     0/0      ☢  │   │       │   │   │   └── scopeguard 1.1.0
4/4        94/94        16/16  0/0     3/3      ☢  │   │       │   │   └── crossbeam-utils 0.8.14
0/0        14/14        0/0    0/0     0/0      ☢  │   │       │   ├── either 1.8.0
0/0        5/5          0/0    0/0     0/0      ☢  │   │       │   │   └── serde 1.0.147
7/7        524/527      4/4    0/0     23/23    ☢  │   │       │   └── rayon-core 1.10.1
```

THANK YOU FOR CHOOSING

// HALBORN