DEX-Solana

合約審計報告

VER 1.0

2022年7月21日

No. 2022072115030

項目總結

1. 項目介紹

Dex-Solana主要實現了交易聚合器兌換功能，接入適配了不同的DEX，通過後端詢價系統，尋找市場上最優的報價路徑，然後將最優路徑組裝成交易發送給合約執行。

2. 審計詳情

項目名稱	DEX-Solana	平台	N/A
通證名稱	N/A	通證代號	N/A
開始時間	2022年6月24日	語言	Rust
結束時間	2022年7月7日	官網	N/A
Github	https://github.com/okex/dex-solana/tree/v1.0.0	白皮書	N/A

3. 審計範圍

ID	文件	SHA-256 checksum
program-rust/src	/processor.rs	5c98dafeb9403bb993ee804184b555fcbbb1cc4c1c61f8d7bf988262f6619642
program-rust/src	/entrypoint.rs	633729b88a7a60a8c911526293088cec6bee5050ddb7b4b8f1c3fb5a3a837d7d
program-rust/src	/exchanger/aldrin/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/aldrin/instruction.rs	21f7d2a4db498e89aa41d77f3a4b8f63ff9e2547d7b899afb877e1138e254b54

2022年7月21日

ID	文件	SHA-256 checksum
program-rust/src	/exchanger/cropper/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/cropper/instruction.rs	c394708bff80868036d46a0a07f1ec36ede616b5d464035e06e8de2f449646dd
program-rust/src	/exchanger/serum_dex/order.rs	c71f30a5e3a1c16a6e91186c3600c22f3903bed9a8e8891dc672f95245aa5ea8
program-rust/src	/exchanger/serum_dex/mod.rs	16537ddb03da43e8ea71f2410f6575bb5bc2c82de50598334f787694cc8545a8
program-rust/src	/exchanger/serum_dex/state.rs	f6f957c3ca98ccad49cb3a776b4044bdf360a9c132a1a38b23b397fec71e9236
program-rust/src	/exchanger/serum_dex/instruction.rs	da02ad7bc642560f7ec17a989242fc926cbcdd2319bf7519fcd3af066a29e6d1
program-rust/src	/exchanger/serum_dex/matching.rs	8422e476da79230b4bae76b5153482626188cb3c330b287fdfe2a1a76358ae07
program-rust/src	/exchanger/stable_swap/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/stable_swap/instruction.rs	261b717c4b58777ed6f7e9399d4b638e7c1e88a31ea3c5129ad62b092ebed9dd
program-rust/src	/exchanger/mod.rs	2267faf100b3b304135d3d4065a8fbe079931eb84bb8350db2e9a48d6d37c9ff
program-rust/src	/exchanger/crema/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/crema/instruction.rs	bd5a57437f1b799f4539e1f52494aa16a51a0f2b14bb96c8227a886ff70585f4
program-rust/src	/exchanger/raydium/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/raydium/instruction.rs	8b369a15e661dfb619a0d08943491f6297241a10bd46f7af54102c91c7a05d83
program-rust/src	/exchanger/spl_token_swap/mod.rs	6bd82534b98aa1c36869aba31abab9f14827855c3b655166166028cc14ca81d3
program-rust/src	/exchanger/spl_token_swap/instruction.rs	36d6401b5bd0ef675453e1d7a63cc901a255a53e2e46596c7114d1e01645a89e
program-rust/src	/spl_token/error.rs	01ba4719c80b6fe911b091a7c05124b64eeece964e09c058ef8f9805daca546b
program-rust/src	/spl_token/mod.rs	615d6944d9e5cdf6a7c37b6829fb879f3dd73ab8517db34b81e0318b79e77544
program-rust/src	/spl_token/instruction.rs	f7d621328305c48e5ce5a96084f2228e57120de6a96a7d1a6ecfc82941c21467
program-rust/src	/error.rs	8e9fb29123019ee1809329b3ca1ece71ca31bff737e8e5cd6bb64e3c4308a10c
program-rust/src	/lib.rs	511024c0cfcb2198d1610bd8413367b91c44d6735f70347513f4c535fdad6a0c
program-rust/src	/parser/serum_dex.rs	0b521e4447c90ec55b1efc7420934800d15f316380741bc1de3287d216374319

2022年7月21日

ID	文件	SHA-256 checksum
program-rust/src	/parser/cropper.rs	d58e60db2b5df7025679bedd522377e06ea294234d9523005658a83a01fbc39d
program-rust/src	/parser/raydium.rs	68acc1f332e98acd62a2aca86bd24cd78154e02d483f4e42541720c3afad49f4
program-rust/src	/parser/spl_token_swap.rs	bafc73679b1892553531517dc0e85a0fe1f7698e06b6b08aa3df0acd3b891f82
program-rust/src	/parser/base.rs	bb54e7aa7bb9e4e2b72245e5fa4aaae28c0e1961ea486bdb2424447ac703cd01
program-rust/src	/parser/mod.rs	fbfe3b42e101f874532095b8f86ccb52999700d77c5282aa797e315ed0469632
program-rust/src	/parser/crema.rs	57f98e552f3d96980227cc988128fef5c5d0a7fc127341eb6867cc33c2dfdb7d
program-rust/src	/parser/aldrin.rs	c6a5183e98994381609f3762a7eee6579d5e1b24cc6fc57713fd8e0b9e82eea5
program-rust/src	/parser/stable_swap.rs	067c9739e5f0b3143fe221cabb74753142e821724704a7c33d5e324ef104726e
program-rust/src	/state.rs	0f130fcdcc6cad5dfb79f98917efd7d8bfad8df72aa0a42e5fd5c44d3922622d
program-rust/src	/instruction.rs	5ca0c031cbb82107b5407465dd83b4ee0337be03ce550231a97ff284bf68a1a3

2022年7月21日

4. 代碼結構

```
├——program-rust/src
├—— entrypoint.rs                          #指令入口
├—— error.rs                               #業務邏輯錯誤定義
├—— exchanger
|   ├—— aldrin                             #aldrin 交易所轉接器
|   |   ├—— instruction.rs
|   |   └—— mod.rs
|   ├—— crema                              #crema 交易所轉接器
|   |   ├—— instruction.rs
|   |   └—— mod.rs
|   ├—— cropper                            #cropper 交易所轉接器
|   |   ├—— instruction.rs
|   |   └—— mod.rs
|   ├—— mod.rs
|   ├—— raydium                            #raydium 交易所轉接器
|   |   ├—— instruction.rs
|   |   └—— mod.rs
|   ├—— serum_dex                          #serum_dex 交易所轉接器
|   |   ├—— instruction.rs
|   |   ├—— matching.rs
|   |   ├—— mod.rs
|   |   ├—— order.rs
|   |   └—— state.rs
|   ├—— spl_token_swap                     #spl_token_swap 交易所轉接器
|   |   ├—— instruction.rs
|   |   └—— mod.rs
|   └—— stable_swap                        #stable_swap 交易所轉接器
|       ├—— instruction.rs
|       └—— mod.rs
├—— instruction.rs                         #指令定義
├—— lib.rs                                 #rust模塊定義
├—— parser
```

2022年7月21日

```
|   ├── aldrin.rs                    #aldrin 交易所解析模塊
|   ├── base.rs
|   ├── crema.rs                     #crema 交易所解析模塊
|   ├── cropper.rs                   #cropper 交易所解析模塊
|   ├── mod.rs
|   ├── raydium.rs                   #raydium 交易所解析模塊
|   ├── serum_dex.rs                 #serum_dex 交易所解析模塊
|   ├── spl_token_swap.rs            #spl_token_swap 交易所解析模塊
|   └── stable_swap.rs               #stable_swap 交易所解析模塊
├── processor.rs                     #指令處理程式碼
├── spl_token                        #spl_token合約庫
|   ├── error.rs
|   ├── instruction.rs
|   └── mod.rs
└── state.rs                         #程式狀態定義
```

2022年7月21日

審計報告匯總

1. 審計方式

通過清晰地理解該項目的設計目的、運行原理和實現管道，稽核團隊對合約程式碼進行了深入的研究和分析。 在分清各個合約及其函數的調用關係的基礎上，對合約可能存在的漏洞進行了定位及分析。 最終產生問題描述和給出相應的修改意見。

審計方法	Static analysis, Manual Review

2. 審計流程

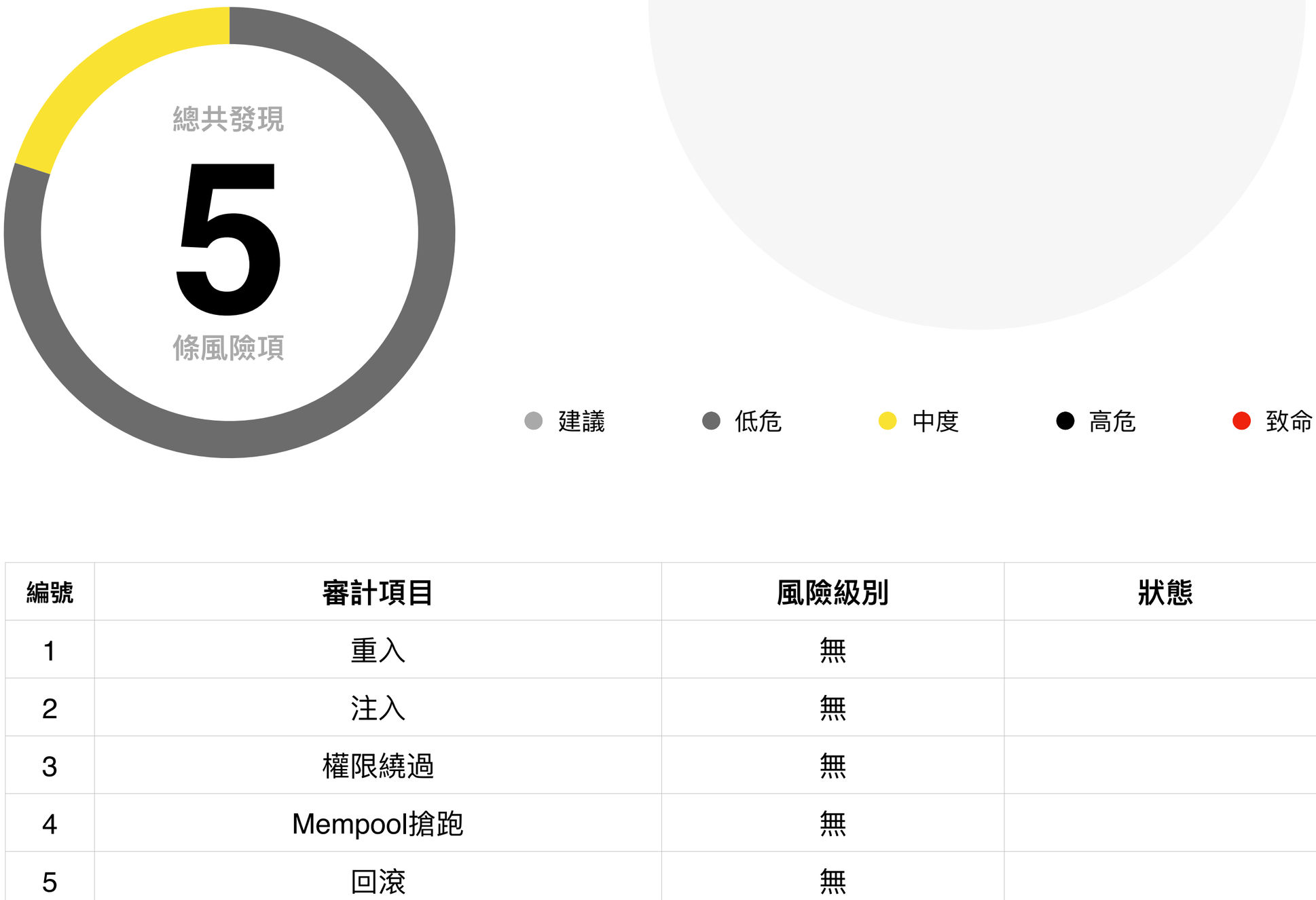
步驟	操作	詳細內容
1	背景研究	閱讀項目介紹、白皮書、合約源碼等項目方團隊提供的相關信息，確保正確理解項目功能
2	自動化檢測	主要用自動化工具掃描源碼，找到常見的潛在漏洞
3	人工審閱	工程師逐行閱讀代碼，找到潛在漏洞
4	邏輯校對	工程師將對代碼的理解和項目方提供的信息比較，檢查代碼實現是否符合項目白皮書信息
5	測試用例檢測	包括測試用例設計，測試範圍分析、符號執行等
6	優化審查	根據應用場景、調用方式及最新的研究成果從可維護性、安全性及可操作性等方面審查項目

2022年7月21日

3. 風險分級

風險級別	風險描述
致命	存在致命風險及隱患，需要立即解決
高危	存在高危風險及隱患，將引發相同問題，必須解決
中度	存在中度風險及隱患，可能導致潛在風險，最終仍然需要解決
低危	存在低風險及隱患，指各類處理不當或會引發警告信息的細節，這類問題可暫時擱置
建議	存在可優化的部分，這類問題可以擱置，但建議最終解決

4. 審計結果



編號	審計項目	風險級別	狀態
1	重入	無	
2	注入	無	
3	權限繞過	無	
4	Mempool搶跑	無	
5	回滾	無	
6	條件競爭	無	
7	循環耗盡gas	無	
8	閃電貸高影響	無	
9	經濟模型不合理	無	
10	可預見的隨機數	無	
11	投票權管理混亂	無	

2022年7月21日

編號	審計項目	風險級別	狀態
12	數據隱私洩露	無	
13	鏈上時間使用不當	無	
14	Fallback函數編碼不當	無	
15	鑒權不當	無	
16	Opcode使用不當	無	
17	內聯匯編使用不當	無	
18	構造函數不規範	無	
19	返回值不規範	無	
20	Event不規範	無	
21	關鍵字使用不規範	無	
22	未遵循ERC標準	無	
23	條件判斷不規範	無	
24	流動性枯竭風險	無	
25	中心化風險	無	
26	邏輯變更風險	無	
27	整數溢出	無	
28	函數可見性不當	無	
29	變量初始化不當	無	
30	合約間調用不當	無	
31	變量不規範	無	
32	重放	無	
33	隨機存儲位置寫入	無	
34	蜜罐邏輯	無	
35	哈希碰撞	無	
36	領獎邏輯不當	無	
37	使用不推薦的方法	無	
38	未遵循基本編碼原則	低	已修改
39	帳戶缺少簽名者檢查	中	已修改
40	帳戶缺少所有者檢查	無	
41	解析帳戶數據前未驗證帳戶	無	
42	缺少相同帳戶校驗	無	

2022年7月21日

編號	審計項目	風險級別	狀態
43	帳戶無法安全關閉	無	
44	帳戶資料類型混淆	無	
45	缺少帳戶可寫檢查	低	已修改
46	過時的外部依賴	無	
47	程式邏輯缺陷	低	已修改

上述表格中，狀態欄內容若為「已告知」，則表示審計團隊已告知項目方項目存在的漏洞，但項目方未對漏洞進行修改，或未告知審計團隊漏洞的修改進度。若狀態欄中填寫「已修改」則表示項目方已進行對漏洞的修改，並通過審計團隊確認。

2022年7月21日

5. 風險項與修改方案

以下部分為審計後得知的風險項相關詳細信息，其中內容包括風險類型、風險級別、問題位置、問題描述、修改建議及項目方反饋。

1. 帳戶缺少簽名者檢查

位置	文件	風險状态	風險級別
Line 361	processor.rs	已修改	中風險

① 風險描述

在關閉`Swap_ Info`帳戶函數`process_ close_ swap_ info`中，缺少對`owner_ account`帳戶進行簽名驗證，存在`Swap_ Info`帳戶被攻擊者關閉並盜取租金的風險。

② 修改建議

添加`owner_account`签名验证

③ 項目方迴響

已修改

2022年7月21日

④ 關聯程式碼

```rust
/**
* pub fn process_close_swap_info(program_id: &Pubkey, accounts:
&[AccountInfo]) -> ProgramResult {
    ...
        let swap_info =
SwapInfo::unpack(&swap_info_account.data.borrow())?;

        //@OKLink Audit Description: 僅檢查swap_ info帳戶owner是否與owner_
account相等，未驗證owner_ account帳戶是否已簽名
        //@OKLink Audit Solution:添加owner_ account簽名驗證

        if !Self::cmp_pubkeys(&swap_info.owner, owner_account.key) {
            return Err(ProtocolError::InvalidOwner.into());
        }
        let dest_starting_lamports = destination_account.lamports();
        **destination_account.lamports.borrow_mut() =
    dest_starting_lamports
            .checked_add(swap_info_account.lamports())
            .ok_or(ProtocolError::Overflow)?;
        **swap_info_account.lamports.borrow_mut() = 0;
        sol_memset(*swap_info_account.data.borrow_mut(), 0,
    SwapInfo::LEN);
        Ok(())
    }
```

2022年7月21日

2. 缺少帳戶可寫檢查

位置	文件	風險状态	風險級別
Line 547	processor.rs	已修改	低風險

① 風險描述

在構建兌換傳入條件函數`process_ single_ step_ swap_ in`中，缺少對`swap_ info_ args. swap_ info_ acc`帳戶可寫檢查，傳入不可寫的`swap_ info_ args. swap_ info_ acc`帳戶會導致交易失敗。

② 修改建議

添加對`swap_info_args.swap_info_acc`帳戶可寫檢查

③ 項目方迴響

已修改

2022年7月21日

④ 關聯程式碼

```
RUST/**
 *pub fn process_single_step_swap_in(
    program_id: &Pubkey,
    data: &SwapInInstruction,
    accounts: &[AccountInfo],
    exchanger: ExchangerType,
) -> ProgramResult {
...
    user_args
    .token_source_account
    .check_owner(user_args.source_account_owner.key, false)?;
```

```
    //@OKLink Audit Description:缺少對swap_ info_ args. swap_ info_ acc帳戶可寫檢查
```

```
    //@OKLink Audit Solution:添加對swap_ info_ args. swap_ info_ acc帳戶可寫檢查
```

```
    match swap_info_args.swap_info.token_account {
        COption::Some(k) => {
            if k != *user_args.token_destination_account.pubkey() {
                return Err(ProtocolError::InvalidTokenAccount.into());
            }
        }
        COption::None => {
            return Err(ProtocolError::InvalidTokenAccount.into());
        }
    };
...
}
```

2022年7月21日

3. 未遵循基本編碼原則

位置	文件	風險状态	風險級別
Line 668	processor.rs	已修改	低風險

① 風險描述

在構建單次兌換傳入條件函數`process_ single_ step_ swap_ in`中，缺少對`to_ amount`是否為零的檢測。 當`to_ amount`值為零時，代表目標交易失敗，未做檢測則沒有輸出交易失敗資訊。

② 修改建議

添加對`to_ amount`是否為零的檢查

③ 項目方迴響

已修改

2022年7月21日

④ 關聯程式碼

```
RUST/**
pub fn process_single_step_swap_in(
    program_id: &Pubkey,
    data: &SwapInInstruction,
    accounts: &[AccountInfo],
    exchanger: ExchangerType,
) -> ProgramResult {
...
    let from_amount_changed =
from_amount_before.checked_sub(from_amount_after).unwrap();
    let to_amount = to_amount_after.checked_sub(to_amount_before).unwrap();
    msg!("from_amount changed: {}", from_amount_changed);
    msg!("to_amount: {}", to_amount);
```

```
    //@OKLink Audit Description:缺少對to_ amount是否為零的檢查
    //@OKLink Audit Solution: 添加對to_ amount是否為零的檢查
```

```
    let mut swap_info = swap_info_args.swap_info;
    swap_info.token_latest_amount = to_amount;
    SwapInfo::pack(
        swap_info,
        &mut swap_info_args.swap_info_acc.data.borrow_mut(),
    )?;
...
}
```

2022年7月21日

4. 程式邏輯缺陷

位置	文件	風險状态	風險級別
Line 703	processor.rs	已修改	低風險

① 風險描述

在`process_ single_ step_ swap_ middle`方法中，缺少對`token_ destination_ account`目標帳戶的`owner`進行檢查，在特殊情況通過多筆交易實現多跳過程中，可能會因為前後輸入不一致導致交易失敗。

② 修改建議

添加對`token_ destination_ account`帳戶`owner`的檢查

③ 項目方迴響

已修改

2022年7月21日

④ 關聯程式碼

```RUST
/**
pub fn process_single_step_swap_middle(
    program_id: &Pubkey,
    accounts: &[AccountInfo],
    exchanger: ExchangerType,
) -> ProgramResult {
    ...

    if !user_args.source_account_owner.is_signer {
        return Err(ProtocolError::InvalidSignerAccount.into());
    }
    user_args
        .token_source_account
        .check_owner(user_args.source_account_owner.key, false)?;

    //@OKLink Audit Description:缺少對token_ destination_ account帳戶owner的檢查
    //@OKLink Audit Solution:添加對token_ destination_ account帳戶owner的檢查

    if !swap_info_args.swap_info_acc.is_writable {
        return Err(ProtocolError::ReadonlyAccount.into());
    }

    ...
}
```

2022年7月21日

5. 未遵循基本編碼原則

位置	文件	風險状态	風險級別
Line 896、900、1001	processor.rs	已修改	低風險

① 風險描述

在`process_ single_ step_ swap_ out`方法中， `expect_ amount_ out`參數沒有參與程式運算。 與`process_ single_ step_ swap_ out_ slim`方法沒有實際區別。

② 修改建議

檢查關聯程式碼中變數使用場景，是否為冗餘變數

③ 項目方迴響

已修改

2022年7月21日

④ 關聯程式碼

RUST/**

```rust
pub fn process_single_step_swap_out(
    program_id: &Pubkey,
    data: &SwapOutInstruction,
    accounts: &[AccountInfo],
    exchanger: ExchangerType,
) -> ProgramResult {
    ...
```

---

*//@OKLink Audit Description: expect_ amount_ out*變數沒有被真正使用

*//@OKLink Audit Solution:*檢查該變數使用場景，是否為冗餘變數

```rust
    msg!(
        "from_amount_before: {}, to_amount_before: {}, amount_in: {},
expect_amount_out: {}, minimum_amount_out: {}",
        from_amount_before,
        to_amount_before,
        amount_in,
        data.expect_amount_out,
        data.minimum_amount_out,
    );

    ...

    msg!(
        "to_amount: {}, expect: {}, minimum: {}",
        to_amount,
        data.expect_amount_out,
        data.minimum_amount_out,
    );

    ...
}
```

2022年7月21日

免責聲明

i. 本審計報告僅針對最終出具報告中載明的審計類型進行審計，其他未知安全漏洞不在本次審計責任範圍之內，我方無需為此承擔責任。

ii. 我方僅應根據審計報告發布之前存在或發生的攻擊或漏洞發布審計報告。對於將來存在或發生的新攻擊或漏洞，我方無法確定對其項目安全狀態的可能影響，對此概不負責。

iii. 我方發布的審計報告中的安全審計分析及其他內容應僅基於項目方在發布審計報告之前向我方提供的文件和材料（包括但不限於合約代碼），並且上述文件和資料不應該存在缺乏信息、被篡改、刪除或隱藏的情況，如果項目方提供的文件和資料存在不真實、不準確、缺乏信息、被篡改、刪除或隱藏的情況，或者對上述文件和資料的改動是在發布審計報告之後作出的，我方不承擔因反映情況與實際情況不一致引起的損失和不利影響 。

iv. 項目方知曉我方出具的審計報告系根據項目方提供的文件和資料、依靠我方現掌握的技術而作出的。但由於任何機構均存在技術的局限性，我方作出的審計報告仍存在無法完整檢測出全部風險的可能性。我方審計團隊鼓勵項目的開發團隊以及任何相關利益方對項目進行後續的測試及審計。

v. 項目方保證其委托我方提供審計或測試服務的項目合法、合規，且不違反適用法律。審計報告僅用於項目方參考，審計報告的內容、獲取方式、使用以及任何其所涉及的服務或資源都不能作為任何形式的投資、稅務、法律、監管及建議等的依據，我方不因此承擔相關責任。在未經我方書面同意之前，項目方不得將審計報告的全部或部分內容以任何形式提及、引用、展示或發送給任何第三方，否則由此產生的任何損失和責任由項目方自行承擔。我方對任何人依賴審計報告或將之用於任何目的概不承擔責任 。

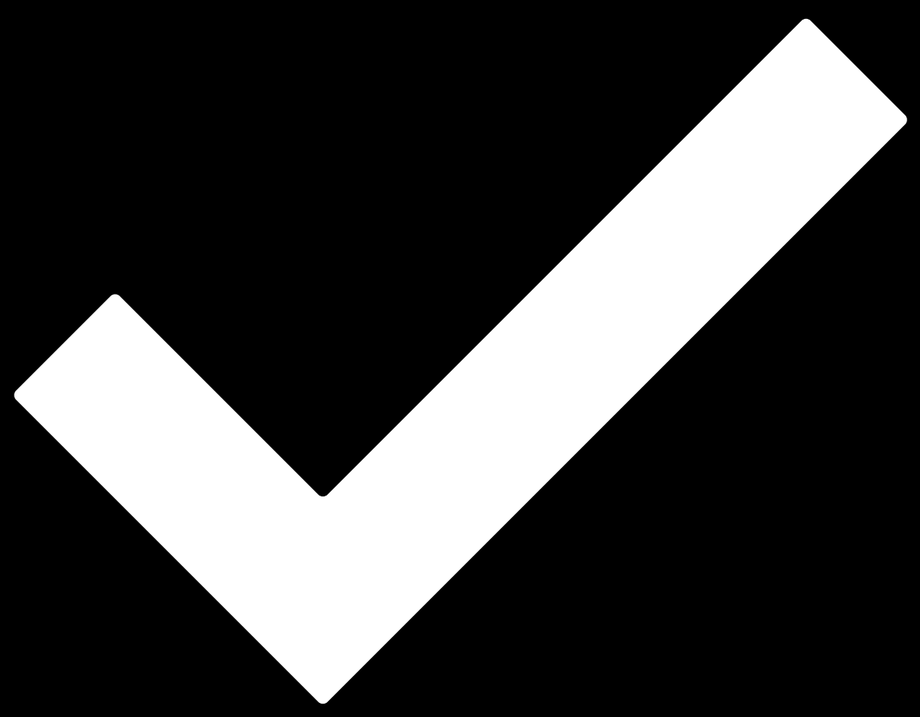
vi. 本審計報告不涉及合約的編譯器及任何超出智能合約編程語言的領域，所審計的智能合約因引用鏈下信息或資源所導致的風險及責任，由項目方自行承擔 。

vii. 不可抗力。不可抗力是指雙方在訂立合同時不能預見、對其發生和後果不能避免且不能克服的事件，包括但不限於戰爭、臺風、水災、火災、地震、潮汐、雷電、天災、罷工、核爆炸、流行病等自然災害和法律、法規和政策變更及政府行為等其它不可預見，對其發生和後果不能防止或避免的事件，且該事件妨礙、影響或延誤任何一方根據合同履行其全部或部分義務。

viii. 如果有一方認為不可抗力發生影響履行本協議義務，應迅速通知另一方，按事件對履約影響的程度，由雙方協商決定是否終止合同或部分免除履約的責任，或者延期履約。

ix. 當不可抗力發生時，任何一方都不能被視作違約或不履行本協議義務。在事件前存在的經濟上的責任，不應受到影響，項目方應對我方已完成工作做出支付。

2022年7月21日



審計通過.

日期 2022年7月21日

審計 歐科雲鏈

本次稽核的目的是為了審閱自研Dex-Solana項目基於Rust語言編寫的DEX交易聚合器功能，研究其設計、架構，發現潛在的安全隱患，並試圖找到可能存在的漏洞。