Security Assessment Report

Sol-Token-Mill

November 18, 2024

Summary

The Sec3 team (formerly Soteria) was engaged to conduct a thorough security analysis of the Sol-Token-Mill smart contracts.

The artifact of the audit was the source code of the following programs, excluding tests, in a private repository.

The initial audit focused on the following versions and revealed 1 issues or questions.

program	type	commit
token-mill	Solana	`b3e6d6e83cfea8a111f50e4f5070fc6ba9f6ea24`

This report provides a detailed description of the findings and their respective resolutions.

After addressing the issues identified in this review, the codebase of the "`token-mill`" program was moved to a public repository at https://github.com/traderjoe-xyz/sol-token-mill, commit b0b9f84.

Table of Contents

Result Overview

Issue	Impact	Status
TOKEN-MILL		
[H-01] Missing token2022 extension checks for the quote token	High	Resolved

Findings in Detail

TOKEN-MILL

[H-01] Missing token2022 extension checks for the quote token

When creating the market, both "base_token_mint" and "quote_token_mint" can use token2022 mints.

```
/* programs/token-mill/src/instructions/create_market.rs */
024 | pub struct CreateMarket<'info> {
036 |     #[account(
037 |         init,
038 |         payer = creator,
039 |         mint::token_program = token_program,
040 |         mint::authority = market,
041 |         mint::decimals = MILL_TOKEN_DECIMALS,
042 |         extensions::metadata_pointer::authority = market,
043 |         extensions::metadata_pointer::metadata_address = base_token_mint,
044 |     )]
045 |     pub base_token_mint: Box<InterfaceAccount<'info, Mint>>,
066 |
067 |     pub quote_token_mint: Box<InterfaceAccount<'info, Mint>>,
075 | }
076 |
077 | pub fn handler(
078 |     ctx: Context<CreateMarket>,
079 |     name: String,
080 |     symbol: String,
081 |     uri: String,
082 |     total_supply: u64,
083 |     creator_fee_share: u16,
084 |     staking_fee_share: u16,
085 | ) -> Result<()> {
094 |     require!(
095 |         check_mint_extensions(&ctx.accounts.base_token_mint)?,
096 |         TokenMillError::UnsupportedTokenMint
097 |     );
```

Although "base_token_mint" is created in this instruction, the handler ensures it can include only the "MetadataPointer" and "TokenMetadata" extensions.

However, for "quote_token_mint", extensions that may introduce side effects are not rejected.

If the permanent delegate extension in the "quote_token_mint" is enabled, its authority gains unrestricted access to tokens and accounts. This authority can transfer or burn tokens in market

quote token vaults, such as "`market_quote_token_ata`".

If the transfer fee extension is enabled, fees are applied with each transfer. The accounting will become inconsistent with the actual quote tokens received. For example, the market may receive fewer tokens than the computed "`quote_amount`".

The "`base_token_mint`" in the extension check at line 94 should be replaced with "`quote_token_mint`".

Resolution

Resolved by commit "`b97059b`" in PR#45.

Appendix: Methodology and Scope of Work

Assisted by the Sec3 Scanner developed in-house, the manual audit particularly focused on the following work items:

- Check common security issues.
- Check program logic implementation against available design specifications.
- Check poor coding practices and unsafe behavior.
- The soundness of the economics design and algorithm is out of scope of this work

DISCLAIMER

The instance report ("Report") was prepared pursuant to an agreement between Coderrect Inc. d/b/a Sec3 (the "Company") and Joemart Ltd dba LFJ.gg (the "Client''). This Report solely includes the results of a technical assessment of a specific build and/or version of the Client's code specified in the Report ("Assessed Code") by the Company. The sole purpose of the Report is to provide the Client with the results of the technical assessment of the Assessed Code. The Report does not apply to any other version and/or build of the Assessed Code. Regardless of the contents of the Report, the Report does not (and should not be interpreted to) provide any warranty, representation or covenant that the Assessed Code: (i) is error and/or bug free, (ii) has no security vulnerabilities, and/or (iii) does not infringe any third-party rights. Moreover, the Report is not, and should not be considered, an endorsement by the Company of the Assessed Code and/or of the Client. Finally, the Report should not be considered investment advice or a recommendation to invest in the Assessed Code and/or the Client.

This Report is considered null and void if the Report (or any portion thereof) is altered in any manner.

ABOUT

The Sec3 audit team comprises a group of computer science professors, researchers, and industry veterans with extensive experience in smart contract security, program analysis, testing, and formal verification. We are also building automated security tools that incorporate static analysis, penetration testing, and formal verification.

At Sec3, we identify and eliminate security vulnerabilities through the most rigorous process and aided by the most advanced analysis tools.

For more information, check out our website and follow us on twitter.