OFFSIDE LABS

Dynamic AMM 0.5.2

Smart Contract Security Assessment

November 2024

Prepared for:

Meteora

Prepared by:

Offside Labs

Sirius Xie

Ronny Xing

Contents

1 About Offside Labs **2**

2 Executive Summary **3**

3 Summary of Findings **4**

4 Key Findings and Recommendations **5**

4.1 Invalid Liquidity Removal Check 5

4.2 Missing QUOTE_MINTS Validation 6

4.3 Informational and Undetermined Issues 6

5 Disclaimer **8**

1 About Offside Labs

Offside Labs is a leading security research team, composed of top talented hackers from both academia and industry.

We possess a wide range of expertise in modern software systems, including, but not limited to, *browsers*, *operating systems*, *IoT devices*, and *hypervisors*. We are also at the forefront of innovative areas like *cryptocurrencies* and *blockchain technologies*. Among our notable accomplishments are remote jailbreaks of devices such as the **iPhone** and **PlayStation 4**, and addressing critical vulnerabilities in the **Tron Network**.

Our team actively engages with and contributes to the security community. Having won and also co-organized *DEFCON CTF*, the most famous CTF competition in the Web2 era, we also triumphed in the **Paradigm CTF 2023** within the Web3 space. In addition, our efforts in responsibly disclosing numerous vulnerabilities to leading tech companies, such as *Apple*, *Google*, and *Microsoft*, have protected digital assets valued at over **$300 million**.

In the transition towards Web3, Offside Labs has achieved remarkable success. We have earned over **$9 million** in bug bounties, and **three** of our innovative techniques were recognized among the **top 10 blockchain hacking techniques of 2022** by the Web3 security community.

https://offside.io/

https://github.com/offsidelabs

https://twitter.com/offside_labs

2 Executive Summary

Introduction

Offside Labs completed a security audit of *Dynamic AMM* smart contracts, starting on Nov 12, 2024, and concluding on Nov 15, 2024.

Project Overview

This update primarily expands the flexibility in creating and configuring *Alpha Vault* related vaults and pools in *Dynamic AMM*.

Dynamic AMM introduces customizable permissionless pool creation and partner fee distribution features, that allows pool creator (a.k.a partner) to claim profit shared from the protocol fees.

Audit Scope

The assessment scope mainly contains amm program for the *Dynamic AMM* project.

The audit is based on the following specific branches and commit hashes of the codebase repositories:

- Dynamic AMM
  - Codebase: https://github.com/MeteoraAg/meteora-dynamic-amm
  - Commit Hash: 392c864ec74295717b366dd21bd92a7d65cdee86
  - Branch: staging

We listed the files we have audited below:

- Dynamic AMM PR-167

Findings

The security audit revealed:

- 0 critical issue
- 0 high issues
- 1 medium issues
- 1 low issues
- 1 informational issues

Further details, including the nature of these issues and recommendations for their remediation, are detailed in the subsequent sections of this report.

3 Summary of Findings

ID	Title	Severity	Status
01	Invalid Liquidity Removal Check	Medium	Fixed
02	Missing QUOTE_MINTS Validation	Low	Fixed
03	Recommend Sufficient Buffer for Settings Regarding Activation Point	Informational	Fixed

4 Key Findings and Recommendations

4.1 Invalid Liquidity Removal Check

Severity: Medium	Status: Fixed
Target: Smart Contract	Category: Logic

Description

The `ActivationHandler.validate_remove_balanced_liquidity` method returns `Ok(())` in all branches. This is not applicable for customizable permissionless pools.

```
70      pub fn validate_add_balanced_liquidity(&self) -> Result<()> {
71          if self.curr_point >= self.activation_point {
72              return Ok(());
73          }
74          Ok(())
75      }
76
77      pub fn validate_remove_balanced_liquidity(&self) -> Result<()> {
78          self.validate_add_balanced_liquidity()
79      }
```

programs/amm/src/activation_handler.rs#L70-L79

Impact

This may allow pool creators to manipulate price curves by withdrawing liquidity before alpha vault swap begins.

Recommendation

The validation logic should be different between `validate_remove_balanced_liquidity` and `validate_add_balanced_liquidity`.

The method `validate_remove_balanced_liquidity` should return an Error in the default branch.

Mitigation Review Log

Fixed in the commit **0e79042e0d2e894460efd82f4ea4c147c9549216**.

4.2 Missing QUOTE_MINTS Validation

Severity: Low	Status: Fixed
Target: Smart Contract	Category: Logic

Description

The `initialize_permissionless_constant_product_pool_with_config` IX doesn't check if the `token_b_mint` is one of the `QUOTE_MINTS` when `config.vault_config_key` is not `Pubkey::default()` .

Impact

Since the Pool address in `InitializePermissionlessConstantProductPoolWithConfig` is uniquely determined by `token_a_mint` , `token_b_mint` , and `config` , if users want to create a pool with an alpha vault, but reversing the order of the quote token and the base token may result in the pool and vault becoming unusable.

Recommendation

Add `QUOTE_MINTS` check for `token_b_mint` when `config.vault_config_key` is not `Pubkey::default()` .

Please note, it's not this branch programs/amm/src/instructions/initialize_pool/initialize_permissionless_pool_with_config.rs#L220-L220 .

Mitigation Review Log

Fixed in the commit **0e79042e0d2e894460efd82f4ea4c147c9549216.**

4.3 Informational and Undetermined Issues

Recommend Sufficient Buffer for Settings Regarding Activation Point

Severity: Informational	Status: Fixed
Target: Smart Contract	Category: Data Validation

Since adjusting `LbPair.pre_activation_duration` affects pre_activation_start_point, it is recommended to include buffer checks in the PermissionLbPairActionAccess.validate_set_pre_activation_duration method to maintain consistency with other validations. For instance, in the PermissionLbPairActionAccess.validate_update_new_activation_point method, `new_pre_activation_start_point` should be greater than `current_point + time_buffer (1 hour)` .

Alternatively, a minimum 5-minute buffer should be maintained to ensure the alpha vault's last join point has not been reached.

When creating `CustomizablePermissionless` pairs, it's also recommended to standardize buffer validation checks instead of only checking `vault_last_join_point` :

DLMM/../programs/lb_clmm/src/instructions/initialize_pool/initialize_customizable_permissionless_lb_pair.rs#L127.

Similar validation also exists in Dynamic AMM:

meteora-dynamic-amm/../programs/amm/src/instructions/initialize_pool/initialize_customizable_permissionless_pool.rs#L281-L281

5 Disclaimer

This audit report is provided for informational purposes only and is not intended to be used as investment advice. While we strive to thoroughly review and analyze the smart contracts in question, we must clarify that our services do not encompass an exhaustive security examination. Our audit aims to identify potential security vulnerabilities to the best of our ability, but it does not serve as a guarantee that the smart contracts are completely free from security risks.

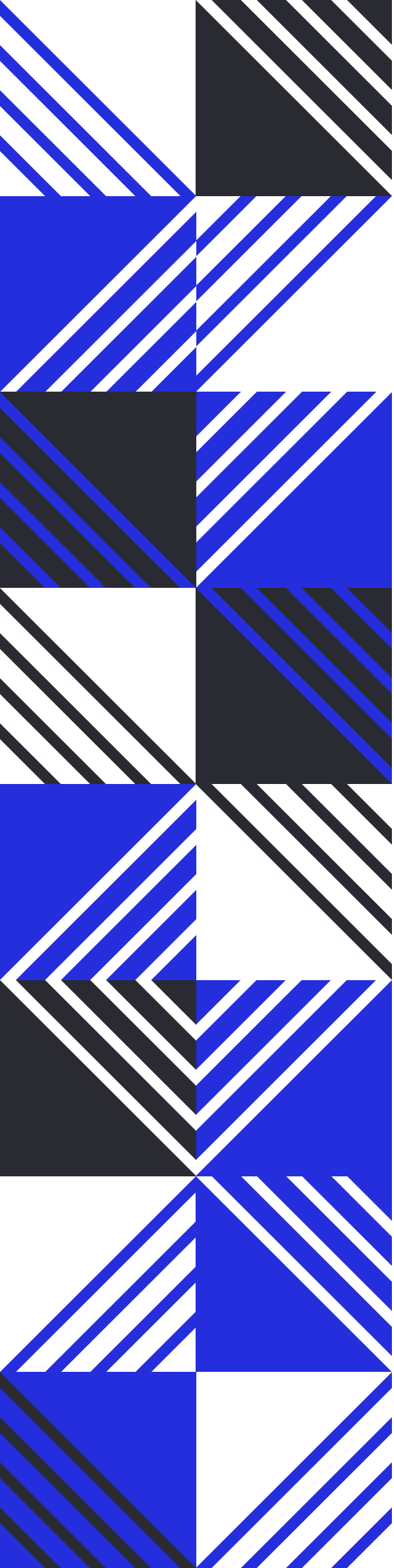
We expressly disclaim any liability for any losses or damages arising from the use of this report or from any security breaches that may occur in the future. We also recommend that our clients engage in multiple independent audits and establish a public bug bounty program as additional measures to bolster the security of their smart contracts.

It is important to note that the scope of our audit is limited to the areas outlined within our engagement and does not include every possible risk or vulnerability. Continuous security practices, including regular audits and monitoring, are essential for maintaining the security of smart contracts over time.

Please note: we are not liable for any security issues stemming from developer errors or misconfigurations at the time of contract deployment; we do not assume responsibility for any centralized governance risks within the project; we are not accountable for any impact on the project's security or availability due to significant damage to the underlying blockchain infrastructure.

By using this report, the client acknowledges the inherent limitations of the audit process and agrees that our firm shall not be held liable for any incidents that may occur subsequent to our engagement.

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



OFFSIDE LABS

https://offside.io/

https://github.com/offsidelabs

https://twitter.com/offside_labs