Vexanium Chain Security Audit Report

Contents

1. Executive Summary

On June 01, 2021, the SlowMist security team received the vexanium team's security audit application for Vexanium chain, developed the audit plan according to the agreement of both parties and the characteristics of the project, and finally issued the security audit report.

The SlowMist security team adopts the strategy of "black, grey box lead, white box assists" to conduct a complete security test on the project in the way closest to the real attack.

SlowMist blockchain system test method:

Black box testing	Conduct security tests from an attacker's perspective externally.
Grey box testing	Conduct security testing on code module through the scripting tool, observing the internal running status, mining weaknesses.
White box testing	Based on the open source code, non-open source code, to detect wether there are vulnerabilities in programs suck as nodes, SDK, etc.

SlowMist blockchain risk level:

Critical vulnerabilities	Critical vulnerabilities will have a significant impact on the security of the blockchain, and it is strongly recommended to fix the critical vulnerabilities.
High-risk vulnerabilities	High-risk vulnerabilities will affect the normal operation of blockchain. It is strongly recommended to fix high-risk vulnerabilities.

Medium-risk vulnerablities	Medium vulnerability will affect the operation of blockchain. It is recommended to fix medium-risk vulnerabilities.
Low-risk vulnerabilities	Low-risk vulnerabilities may affect the operation of blockchain in certain scenarios. It is suggested that the project party should evaluate and consider whether these vulnerabilities need to be fixed.
Weaknesses	There are safety risks theoretically, but it is extremely difficult to reproduce in engineering.
Enhancement Suggestions	There are better practices for coding or architecture.

2. Project Background (Context)

2.1 Project Introduction

Project Website: https://vexanium.com

Coin Symbol: VEX

Project source code: https://github.com/vexanium/VexChain

Audit version: 102cb21f2e6548408d63be3749238defa45559c6

2.2 Scope of Audit

The main types of security audit include:

(other unknown security vulnerabilities are not included in the scope of responsibility of this audit)

No.	Audit Category	Audit Result
1	Code Compliance Audit	PASS
2	Random Number Generation Algorithm Audit	PASS
3	Keystore Audit	PASS
4	Cryptographic Component Call Audit	PASS
5	Encryption Strength Audit	PASS
6	Length Extension Attack Audit	PASS
7	Transaction Malleability Attack Audit	PASS
8	Replay Attack Audit	PASS
9	Top-up Program Audit	Some Risks
10	RPC Permission Audit	PASS

3. Code Overview

3.1 Infrastructure

VexChain is based on the open source EOSIO(https://github.com/eosio/eos) development.

3.2 Code Compliance Audit

Comparing with EOS, the main modification is the replacement of the brand logo, and change public key legacy prefix.

3.3 Random Number Generation Algorithm Audit

The private key is generated based on the openssl ecc library, which is a secure library.

- VexChain-main/libraries/fc/src/crypto/elliptic_common.cpp

```
private_key private_key::generate()
{
EC_KEY* k = EC_KEY_new_by_curve_name( NID_secp256k1 );
if( !k ) FC_THROW_EXCEPTION( exception, "Unable to generate EC key" );
if( !EC_KEY_generate_key( k ) )
{
FC_THROW_EXCEPTION( exception, "ecc key generation error" );

}

return private_key( k );
}
```

3.4 Keystore Audit

When cleos creates a wallet, the wallet password is automatically generated, and the encryption strength is extremely high. The random algorithm for generating the password is based on the private key generation algorithm, and the algorithm is secure.

3.5 Cryptographic Component Call Audit

Signature algorithm: Secp256k1

Hash algorithm: SHA256

No error calls were found.

3.6 Encryption Strength Audit

Weak hash functions such as md5 and sha1 are not used.

3.7 Length Extension Attack Audit

In cryptography and computer security, a length extension attack is a type of attack where an attacker can use Hash(message1) and the length of message1 to calculate Hash(message1 ‖ message2) for an attacker-controlled message2, without needing to know the content of message1. Algorithms like MD5, SHA-1, and SHA-2 that are based on the Merkle－Damgard construction are susceptible to this kind of attack. The SHA-3 algorithm is not susceptible.

No error calls were found.

3.8 Transaction Malleability Attack Audit

The ECDSA algorithm generates two large integers r and s combined as a signature, which can be used to verify transactions. And r and BN-s can also be used as signatures to verify transactions. In this way, the attacker gets a transaction, extracts the r and s of inputSig, uses r, BN-s to generate a new inputSig, and then forms a new transaction with the same input and output, but different TXID. Attacker Can successfully generate legal transactions at almost no cost without having the private key.

No error calls were found.

Vulnerability reference:

https://en.bitcoinwiki.org/wiki/Transaction_Malleability

3.9 Transaction Replay Attack Audit

Each signed transaction contains ref-block information, and the chain_id is added when hashing the transaction, which can prevent replay attacks between similar chains (EOS).

Including expiration transaction expiration time, to ensure that failed transactions will not be used for replay.

- VexChain-main/libraries/chain/include/eosio/chain/transaction.hpp

```
struct transaction_header {
time_point_sec expiration; ///< the time at which a transaction expires
uint16_t ref_block_num = 0U; ///< specifies a block num in the last 2^16 blocks.
uint32_t ref_block_prefix = 0UL; ///< specifies the lower 32 bits of the blockid at get_ref_blocknum
fc::unsigned_int max_net_usage_words = 0UL; /// upper limit on total network bandwidth (in 8 byte words) billed for this transaction
uint8_t max_cpu_usage_ms = 0; /// upper limit on the total CPU time billed for this transaction
fc::unsigned_int delay_sec = 0UL; /// number of seconds to delay this transaction for during which it may be canceled.
```

3.10 Top-up Program Audit

The features on the VexChain may lead to "false top-up" attacks. There are multiple execution states of transactions. The corresponding categories and corresponding descriptions are as follows:

executed: The transaction was executed correctly without triggering the error handler

soft_fail: The transaction failed objectively (not executed), but the error handler was triggered correctly

hard_fail: The transaction failed objectively, but the error handler was not triggered

delayed: The transaction is delayed/deferred/waiting to be executed in the queue

```
expired: transaction expired
```

The attacker can initiate a "false top-up" by setting the construction transaction parameters, and the exchange needs to check the deposit conditions when processing the deposit.

3.11 RPC Permission Audit

Node RPC is started by nodeos and has no sensitive information interface; Wallet RPC is started by keosd and is responsible for managing private keys and signature information. By default, ports are not opened to WAN and cannot be cross-domain by default. If the RPC port of the wallet is opened to WAN, the attacker can sign the transfer remotely .

Vulnerability reference: https://mp.weixin.qq.com/s/Kk2IsoQ1679Gda56Ec-zJg

4. Findings

Vulnerability distribution:

Critical vulnerabilities	0	
High-risk vulnerabilities	0	
Medium-risk vulnerablities	0	
Low-risk vulnerabilities	0	
Weaknesses	1	■
Enhancement Suggestions	0	
Total	1	

■Code Compliance ■Random Number Generation Algorithm ■Keystore ■Cryptographic Component Call ■Encryption Strength ■Length Extension Attack ■Transaction Malleability Attack ■Replay Attack ■ Top-up Program ■RPC Permission

4.1 False Top-up Attack[weakness]

The attacker can initiate a "false top-up" by setting the construction transaction parameters, and the exchange needs to check the deposit conditions when processing the deposit. This is a feature of Vexanium chain, not a vulnerability, but it may cause the loss of exchange funds.

5. Conclusion

Audit result: PASS

Audit No. : BCA002106080001

Audit date: June 08, 2021

Audit team: SlowMist security team

Summary conclusion: After correction, all problems found have been fixed and the above risks have been eliminated by Vexanium chain. Comprehensive assessed, Vexanium chain no risks above already.

6. Statement

SlowMist issues this report with reference to the facts that have occurred or existed before the issuance of this report, and only assumes corresponding responsibility base on these.

For the facts that occurred or existed after the issuance, SlowMist is not able to judge the security status of this project, and is not responsible for them. The security audit analysis and other contents of this report are based on the documents and materials provided to SlowMist by the information provider till the date of the insurance this report (referred to as "provided information"). SlowMist assumes: The information provided is not missing, tampered with, deleted or concealed. If the information provided is missing, tampered with, deleted, concealed, or inconsistent with the actual situation, the SlowMist shall not be liable for any loss or adverse effect resulting therefrom. SlowMist only conducts the agreed security audit on the security situation of the project and issues this report. SlowMist is not responsible for the background and other conditions of the project.

Official Website
www.slowmist.com

E-mail
team@slowmist.com

Twitter
@SlowMist_Team

Github
https://github.com/slowmist