HACKEN

MINIMA BLOCKCHAIN SECURITY ANALYSIS

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Intro

This report may contain confidential information about IT systems and the intellectual property of the Customer, as well as information about potential vulnerabilities and methods of their exploitation.

The report can be disclosed publicly after prior consent by another Party. Any subsequent publication of this report shall be without mandatory consent.

Name	Minima
Website	https://www.minima.global/
Repository	https://github.com/minima-global/Minima
Commit	8619241839f1c2c2d0a48bd6fab0391551aee2a2
Platform	L1
Network	Minima
Languages	Java
Methods	Automated Code analysis, Manual review, Issues simulation, Fuzz testing
Auditors	Yarik Bratashchuk (y.bratashchuk@hacken.io), Bartosz Barwikowski (b.barwikowski@hacken.io)
Approver	Andriy Kashcheyev (a.kashcheyev@hacken.io), Luciano Ciattaglia (l.ciattaglia@hacken.io)
Timeline	03.11.2022 - 12.01.2023
Changelog	01.12.2023 (Preliminary Report)

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Table of contents

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Summary

Minima is a decentralized network for information and value transfer. Like Bitcoin, it uses UTxO model and has a similar Turing complete scripting language for writing "Smart Contracts" (scripts). Minima has only one kind of node - a full node, which is run by everyone accepting different devices, aiming for mobile phones the most. Every node mines its transactions before sending them to the network.

Hacken performed the security audit against commit hash: 8619241839f1c2c2d0a48bd6fab0391551aee2a2

The audit was performed based on the defined scope and covered all critical parts of the Minima protocol including Consensus, VM, Transactions, P2P, among other features detailed in the scope section. Overall, the project is consistent and the implementation of it is excellent. The Minima team fixed all the issues found during the audit period in time record following the recommendations given by the auditors.

The project needs to increase test coverage, as some critical parts are not properly tested at the moment of the audit.

Documentation quality

The total Documentation Quality score is **9.0** out of 10.

Based on technical documentation produced by the Minima development team, which covers all concepts in detail, such as TxPow, MMR database, Scripting etc. The score is also based on the node repository documentation and code comments. We recommend updating bits for running testnet locally as some of it is outdated. Finally, some terminal commands need more description. But the general documentation score is a 9 our of 10.

Code quality

The total Code Quality score is **6.5** out of 10.

Static code analysis run on Minimal projects indicated that the code is well written, with minimal flaws, most of which are false positives. What deserves to be highlighted in the score is that the code has poor test coverage at the moment of the audit, and some logic lack comments.

The given score considered that there had been an improvement in test coverage since the beginning of the audit.

Architecture quality

The architecture quality score is **8.5** out of 10.

Minimal Node is well designed, which is a standard for a respected software. All entities and components could be easily tested.

During the audit, it was found that terminal commands are hard to use, and there is room for improvement in the CLI experience. It would be a good idea to have a separate command that would not interfere with the main process log outputs.

Security score

The security score is **10.0** out of 10. Due to all the critical security issues being already fixed before the end of the audit.

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

General score

The general score is **9.4** out of 10. Due to all the scores detailed above.

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Scope of the audit

Protocol Audit

Cryptographic signatures and Key Management

- Cryptography Libraries
- Keys Generation
- Keystore storage
- Signing and Verification

State

- UTXO implementation review
- Security vectors analysis (data availbility, double spend)

Transactions

- Mempool review (defaults, timestamps)
- Attacks scenarios analysis (replay, maullability)

Consensus

- Consensus implementation review
- Attack scenarios analysis (liveness, finality, eclipse,split)

VM

- Execution Layer Review
- Attack scenarios abalysis (Gas, race, stack)

Storage

- Storage implementation review (database package)

P2P

- P2P analysis
- Standard attacks analysis (DoS, encryption, overflows, state machine)

MDS

- RPC (MDS) analisys
- Standard attacks analysis (defaults,DoS, overflows)

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Implementation

Code Quality

- Static Code Analysis
- Tests coverage

Protocol Tests

Node Tests

- Environment Setup
- E2E sync tests
- Crash tests
- E2E transaction tests

VM Tests

- Fuzz tests

Issues

Loop with unreachable exit condition using constant expressions

KISS VM allows to create infinite loop script.

ID	MINIMA-14
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Loop with Unreachable Exit Condition

Details

`TRUE` AND `1` are constant expressions and not counted as instructions, therefore we can create a script

```
WHILE TRUE EXEC 1
```

that will run and never reach max number of instructions (infinite loop).

Recommendation

Consider making `EXEC` a function so this construct is not possible.

Fix author comments

ConstantExpressions like TRUE now cause one instruction increment. Now all Expressions and all Statements cause 1 increment. Will run out of instructions.

NULL Pointer Dereference when calling `to0xString()` on mTransactionID

ID	MINIMA-17
Scope	Transactions, Security
Severity	CRITICAL
Status	New
Vulnerability Type	NULL Pointer Dereference

Details

`calculateTransactionID()` is responsible for setting `Transaction.mTransactionID`. `mTransactionID` could be set to `null`, which leads to exception when calling `to0xString()` method on it.

Recommendation

Check if `mTransactionID` is not `null` before calling `to0xString()` on it.

NULL Pointer Dereference when calling method on StateVariable

ID	MINIMA-16
Scope	Transactions, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	NULL Pointer Dereference

Details

When transaction is read from the `DataInputStream` it sets `mType` (tx state variable type) value, that later in code is used to set `mData` (tx state variable data) value. It is possible to send such transaction that when deserialized, the state variable `mType` is set to the arbitrary value, which in turn will leave `mData` as `null`. Execution later crashes with `java.lang.NullPointerException` when calling `StateVariable.toString()`.

Recommendation

Handle arbitrary value for `mType` within `StateVariable.readDataStream` function.

Fix author comments

Have added code that check and throws error if undefined..

```
@Override
    public void readDataStream(DataInputStream zIn) throws IOException {
        mPort = MiniByte.ReadFromStream(zIn);
        mType = MiniByte.ReadFromStream(zIn);
        //What Data Type..
        if(mType.isEqual(STATETYPE_BOOL)) {
            MiniByte bool = MiniByte.ReadFromStream(zIn);
            if(bool.isTrue()) {
                mData = new MiniString("TRUE");
            }else {
                mData = new MiniString("FALSE");
            }
        }else if(mType.isEqual(STATETYPE_HEX)) {
            MiniData data = MiniData.ReadFromStream(zIn);
            mData = new MiniString(data.to0xString());
        }else if(mType.isEqual(STATETYPE_NUMBER)) {
            MiniNumber number = MiniNumber.ReadFromStream(zIn);
            mData = new MiniString(number.toString());
        }else if(mType.isEqual(STATETYPE_STRING)) {
            mData = MiniString.ReadFromStream(zIn);
        }else{
            throw new IOException("Invalid StateVariable type : "+mType);
        }
    }
```

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Stack-based buffer overflow in given function

KISS VM is not resilient to stack overflow attack when calling script functions.

ID	MINIMA-7
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Stack-based Buffer Overflow

Details

Contract parsing crashes with `java.lang.StackOverflowError` error if too many nested function calls present. Here is the script to reproduce it:

```
LET func = [ EXEC func ] EXEC func
```

Recommendation

Limit callstack size for a given function call.

Fix author comments

Maximum Callstack depth is set to 64

```
Minima @ 14/12/2022 12:11:33 [3.4 MB] : Script:LET func = [ EXEC func ] EXEC func
Minima @ 14/12/2022 12:11:33 [4.0 MB] : Clean:LET func=[ EXEC func ] EXEC func
Minima @ 14/12/2022 12:11:33 [4.0 MB] : INST[0] STACK[0] - Contract   : LET func=[ EXEC func ] EXEC func
Minima @ 14/12/2022 12:11:33 [4.0 MB] : INST[0] STACK[0] - Size       : 32
Minima @ 14/12/2022 12:11:33 [13.8 MB] : INST[0] STACK[0] - Transaction   : {"inputs":[],"outputs":[],"state":[],"linkha
Minima @ 14/12/2022 12:11:33 [13.8 MB] : INST[0] STACK[0] - Witness       : {"signatures":[],"mmrproofs":[],"scripts":[
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 0) Token : [COMMAND] LET
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 1) Token : [VARIABLE] func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 2) Token : [OPERATOR] =
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 3) Token : [VALUE] [ EXEC func ]
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 4) Token : [COMMAND] EXEC
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - 5) Token : [VARIABLE] func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - Script token parse OK.
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[0] STACK[0] - Start executing the contract
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[1] STACK[1] - LET func =  EXEC func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[2] STACK[1] - { func =  EXEC func , }
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[3] STACK[1] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[5] STACK[2] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[7] STACK[3] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[9] STACK[4] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[11] STACK[5] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[13] STACK[6] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.2 MB] : INST[15] STACK[7] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.4 MB] : INST[17] STACK[8] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.4 MB] : INST[19] STACK[9] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[21] STACK[10] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[23] STACK[11] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[25] STACK[12] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[27] STACK[13] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[29] STACK[14] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[31] STACK[15] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[33] STACK[16] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[35] STACK[17] - EXEC variable:func
```

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

```
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[37] STACK[18] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[39] STACK[19] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[41] STACK[20] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[43] STACK[21] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[45] STACK[22] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[47] STACK[23] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [14.8 MB] : INST[49] STACK[24] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[51] STACK[25] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[53] STACK[26] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[55] STACK[27] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[57] STACK[28] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[59] STACK[29] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[61] STACK[30] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[63] STACK[31] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[65] STACK[32] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[67] STACK[33] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[69] STACK[34] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[71] STACK[35] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[73] STACK[36] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.2 MB] : INST[75] STACK[37] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[77] STACK[38] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[79] STACK[39] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[81] STACK[40] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[83] STACK[41] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[85] STACK[42] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[87] STACK[43] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[89] STACK[44] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[91] STACK[45] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[93] STACK[46] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[95] STACK[47] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[97] STACK[48] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[99] STACK[49] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [15.6 MB] : INST[101] STACK[50] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[103] STACK[51] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[105] STACK[52] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[107] STACK[53] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[109] STACK[54] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[111] STACK[55] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[113] STACK[56] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[115] STACK[57] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[117] STACK[58] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[119] STACK[59] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[121] STACK[60] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[123] STACK[61] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[125] STACK[62] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.0 MB] : INST[127] STACK[63] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.4 MB] : INST[129] STACK[64] - EXEC variable:func
Minima @ 14/12/2022 12:11:33 [16.4 MB] : org.minima.kissvm.exceptions.ExecutionException: org.minima.kissvm.exceptions.
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
```

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

```
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.4 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
```

```
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.commands.EXECstatement.execute(EXECstatement
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.Contract.run(Contract.java:345)
Minima @ 14/12/2022 12:11:33 [16.6 MB] :      org.minima.kissvm.Contract.main(Contract.java:718)
Minima @ 14/12/2022 12:11:33 [16.6 MB] : INST[130] STACK[64] - Execution Error - org.minima.kissvm.exceptions.Execution
Minima @ 14/12/2022 12:11:33 [16.6 MB] : INST[130] STACK[64] - Contract instructions : 130
Minima @ 14/12/2022 12:11:33 [16.6 MB] : INST[130] STACK[64] - Contract finished     : false
```

Stack-based buffer overflow within parseTokens function

KISS VM is not resilient to stack overflow attack when parsing contract tokens recursively.

ID	MINIMA-12
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Stack-based Buffer Overflow

Details

KISS VM crashes with `java.lang.StackOverflowError` when trying to parse a contract as below by recursively calling to `getElseOrElseIfOrEndIF` function.

```
IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN IF TRUE RETURN
```

Similar recursive functions are `getEndWHILE` and `parseToken` itself.

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Here is another example of recursive call to `getExpression` from within `getBaseUnit`:

```
IF((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((((
```

Recommendation

Limit callstack size when parsing script tokens.

Fix author comments

Call stack depth now has a limit of 64.

Uncontrolled memory allocation achieved with bitwise shift in hex values

KISS VM allows to do bitwise shifting operation in a hexadecimal value that causes `java.lang.OutOfMemory` crash later when the value is used.

ID	MINIMA-13
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Memory Allocation with Excessive Size Value

Details

Here is an example script that causes crash using bitwise shifting:

```
LET a = 0xff<<1000000000
```

`HexValue` is implemented using `MiniData` type which has maximum size of 256MB. There is no protection from exceeding max value using bitwise shifting operation.

Recommendation

Put limits on bitwise shifting operation.

Fix author comments

Fixed by setting 256bits shift limit

```
Minima @ 14/12/2022 12:09:48 [3.4 MB] : Script:LET a = 0xff<<1000000000
Minima @ 14/12/2022 12:09:48 [4.0 MB] : Clean:LET a=0xFF<<1000000000
Minima @ 14/12/2022 12:09:48 [4.0 MB] : INST[0] STACK[0] - Contract   : LET a=0xFF<<1000000000
Minima @ 14/12/2022 12:09:48 [4.0 MB] : INST[0] STACK[0] - Size       : 22
Minima @ 14/12/2022 12:09:48 [13.8 MB] : INST[0] STACK[0] - Transaction   : {"inputs":[],"outputs":[],"state":[],"linkha
Minima @ 14/12/2022 12:09:48 [13.8 MB] : INST[0] STACK[0] - Witness       : {"signatures":[],"mmrproofs":[],"scripts":[
Minima @ 14/12/2022 12:09:48 [13.8 MB] : INST[0] STACK[0] - 0) Token : [COMMAND] LET
Minima @ 14/12/2022 12:09:48 [13.8 MB] : INST[0] STACK[0] - 1) Token : [VARIABLE] a
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - 2) Token : [OPERATOR] =
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - 3) Token : [VALUE] 0xFF
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - 4) Token : [OPERATOR] <<
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - 5) Token : [VALUE] 1000000000
```

```
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - Script token parse OK.
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[0] STACK[0] - Start executing the contract
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[1] STACK[1] - LET a = ( 0xFF << 1000000000 )
Minima @ 14/12/2022 12:09:48 [14.2 MB] : org.minima.kissvm.exceptions.ExecutionException: Can only SHIFTLEFT 256 bits M
Minima @ 14/12/2022 12:09:48 [14.2 MB] :      org.minima.kissvm.expressions.OperatorExpression.getValue(OperatorExpress
Minima @ 14/12/2022 12:09:48 [14.2 MB] :      org.minima.kissvm.statements.commands.LETstatement.execute(LETstatement.j
Minima @ 14/12/2022 12:09:48 [14.2 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:09:48 [14.2 MB] :      org.minima.kissvm.Contract.run(Contract.java:345)
Minima @ 14/12/2022 12:09:48 [14.2 MB] :      org.minima.kissvm.Contract.main(Contract.java:718)
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[4] STACK[1] - Execution Error - org.minima.kissvm.exceptions.ExecutionExc
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[4] STACK[1] - Contract instructions : 4
Minima @ 14/12/2022 12:09:48 [14.2 MB] : INST[4] STACK[1] - Contract finished     : false
```

Uncontrolled memory allocation during POW function

KISS VM does not control how much memory has been used while executing `POW` function.

ID	MINIMA-15
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Memory Allocation with Excessive Size Value

Details

Here is the script that crashes KISS VM with `java.lang.OutOfMemory` exception:

```
LET a = POW(POW(9 9) 9)
```

Executing such script requires much more memory than `MiniNumber` allowed to consume.

Recommendation

Find a way to throw `java.lang.NumberFormatException: MiniNumber too large` before making a calculation that will lead to crash.

Fix author comments

POW is slow. Set max pow exponent to 1024

Uncontrolled memory allocation or resorce consumption when calling FUNCTION generic function

KISS VM is not controlling memory allocation when calling generic `FUNCTION` function.

ID	MINIMA-11
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)

Vulnerability Type	Memory Allocation with Excessive Size Value

Details

Generic `FUNCTION` function takes the contract as a first parameter and the rest of the `FUNCTION` parameters are used to pass parameters to the contract. KISS VM crashes when trying to replace (recursively) $n within a script with `FUNCTION` arguments for a given script:

```
LET a = [$1$1$1$1$1$1$1$1$1$1] // the script
LET b = [$2$2$2$2$2$2$2$2$2$2] // first script parameter
LET c = [$3$3$3$3$3$3$3$3$3$3]
LET d = [$4$4$4$4$4$4$4$4$4$4]
LET e = [$5$5$5$5$5$5$5$5$5$5]
LET f = [$6$6$6$6$6$6$6$6$6$6]
LET g = [$7$7$7$7$7$7$7$7$7$7]
LET h = [$8$8$8$8$8$8$8$8$8$8] // last script parameter
LET z = FUNCTION(a b c d e f g h)
```

If contract is given 6 parameters (instead of 7) - it creates CPU DoS instead of throwing `java.lang.OutOfMemory` exception, resulting in throttling (hanging) contract execution for unknown time.

Recommendation

Put memory limits for the $n parameter replacement logic, lso consider putting a time limits for the contract execution.

Fix author comments

Added a MAX number of function params(10) for all functions and a max number of replacements allowed in FUNCTION(64)

Uncontrolled memory allocation while calling CONCAT function

KISS VM is not controlling memory allocation when calling `CONCAT` function.

ID	MINIMA-8
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Memory Allocation with Excessive Size Value

Details

It is easy to crash KISS VM with `java.lang.OutOfMemoryError` using calls to `CONCAT` function. It will allocate all the memory available when the input grows exponentially:

```
LET a=0xFF LET g=0 WHILE g LT 100 DO LET a=CONCAT(a a) ENDWHILE
```

Recommendation

Put limits on memory allocation for `CONCAT` function within `runFunction`.

Fix author comments

Fixed by limiting hex value size

```
...
Minima @ 14/12/2022 12:13:31 [23.6 MB] : INST[170] STACK[1] - ( variable:g LT 100 ) returns:TRUE
Minima @ 14/12/2022 12:13:31 [39.6 MB] : INST[171] STACK[2] - LET a = function:CONCAT, params:[variable:a, variable:a]
Minima @ 14/12/2022 12:13:31 [55.6 MB] : org.minima.kissvm.exceptions.ExecutionException: MAX HEX value size reached : 
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.functions.hex.CONCAT.runFunction(CONCAT.java:46)
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.expressions.FunctionExpression.getValue(FunctionExpressi
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.statements.commands.LETstatement.execute(LETstatement.ja
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.statements.commands.WHILEstatement.execute(WHILEstatemen
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.Contract.run(Contract.java:345)
Minima @ 14/12/2022 12:13:31 [55.6 MB] :      org.minima.kissvm.Contract.main(Contract.java:718)
Minima @ 14/12/2022 12:13:31 [55.6 MB] : INST[176] STACK[3] - Execution Error - org.minima.kissvm.exceptions.ExecutionEx
Minima @ 14/12/2022 12:13:31 [71.6 MB] : INST[176] STACK[3] - Contract instructions : 176
Minima @ 14/12/2022 12:13:31 [87.6 MB] : INST[176] STACK[3] - Contract finished     : false
```

Uncontrolled memory allocation while calling REPLACE function

KISS VM is not controlling memory allocation when calling `REPLACE` function.

ID	MINIMA-9
Scope	KISS VM, Security
Severity	CRITICAL
Status	Fixed (12892f729)
Vulnerability Type	Memory Allocation with Excessive Size Value

Details

It is easy to crash KISS VM with `java.lang.OutOfMemoryError` using calls to `REPLACE` function. It will allocate all the memory available when the input grows exponentially:

```
LET a=[AA] LET g=0 WHILE g LT 100 DO LET a=REPLACE(a [A] a) ENDWHILE
```

Recommendation

Put limits on memory allocation for `REPLACE` function.

Fix author comments

Max String size is set to 32k.

Uncontrolled memory allocation while calling SUBSET function

KISS VM is not controlling memory allocation when calling `SUBSET` function.

ID	MINIMA-9
Scope	KISS VM, Security
Severity	CRITICAL

Status	Fixed (12892f729)
Vulnerability Type	Memory Allocation with Excessive Size Value

Details

It is easy to crash KISS VM by consuming all the heap space during a call to `SUBSET` function. Internally, `SUBSET` allocates bytes of memory like this

```
// len here is b - a = 2147483638 (interger overflow), resulting in allocating ~2GB of memory
byte[] subs = new byte[len];
```

and after it calls to `System.arraycopy` that allocates even more memory. Depending on the system it will throw `java.lang.OutOfMemory` during one of these allocations. The crash can be reproduced with this contract:

```
LET a = 10 LET b = -2147483648 LET c = HEX(a - b) RETURN SUBSET(a b c)
```

Recommendation

Put limits on memory allocation for `SUBSET` function and check for integer overflow within corresponding `runFunction` implementation.

Fix author comments

Fixed to throw error when size is too large.

```
Minima @ 14/12/2022 12:54:31 [3.4 MB] : Script:LET a = 10 LET b = -2147483648 LET c = HEX(a - b) RETURN SUBSET(a b c)
Minima @ 14/12/2022 12:54:31 [4.0 MB] : Clean:LET a=10 LET b=-2147483648 LET c=HEX(a-b) RETURN SUBSET(a b c)
Minima @ 14/12/2022 12:54:31 [4.5 MB] : INST[0] STACK[0] - Contract   : LET a=10 LET b=-2147483648 LET c=HEX(a-b) RETURN
Minima @ 14/12/2022 12:54:31 [4.5 MB] : INST[0] STACK[0] - Size       : 62
Minima @ 14/12/2022 12:54:31 [14.2 MB] : INST[0] STACK[0] - Transaction   : {"inputs":[],"outputs":[],"state":[],"linkha
Minima @ 14/12/2022 12:54:31 [14.2 MB] : INST[0] STACK[0] - Witness       : {"signatures":[],"mmrproofs":[],"scripts":[
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 0) Token : [COMMAND] LET
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 1) Token : [VARIABLE] a
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 2) Token : [OPERATOR] =
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 3) Token : [VALUE] 10
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 4) Token : [COMMAND] LET
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 5) Token : [VARIABLE] b
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 6) Token : [OPERATOR] =
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 7) Token : [OPERATOR] -
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 8) Token : [VALUE] 2147483648
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 9) Token : [COMMAND] LET
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 10) Token : [VARIABLE] c
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 11) Token : [OPERATOR] =
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 12) Token : [FUNCTION] HEX
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 13) Token : [OPENBRACKET] (
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 14) Token : [VARIABLE] a
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 15) Token : [OPERATOR] -
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 16) Token : [VARIABLE] b
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 17) Token : [CLOSEBRACKET] )
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 18) Token : [COMMAND] RETURN
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 19) Token : [FUNCTION] SUBSET
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 20) Token : [OPENBRACKET] (
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 21) Token : [VARIABLE] a
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 22) Token : [VARIABLE] b
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 23) Token : [VARIABLE] c
Minima @ 14/12/2022 12:54:31 [14.4 MB] : INST[0] STACK[0] - 24) Token : [CLOSEBRACKET] )
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[0] STACK[0] - Script token parse OK.
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[0] STACK[0] - Start executing the contract
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[1] STACK[1] - LET a = 10
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[2] STACK[1] - { a = 10, }
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[3] STACK[1] - LET b = -2147483648
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[4] STACK[1] - { b = -2147483648, a = 10, }
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[5] STACK[1] - LET c = function:HEX, params:[( variable:a - variable:b )]
```

```
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[9] STACK[2] - ( variable:a - variable:b ) returns:2147483658
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[9] STACK[1] - function:HEX, params:[( variable:a - variable:b )] returns:(
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[9] STACK[1] - { b = -2147483648, a = 10, c = 0x8000000A, }
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[10] STACK[1] - RETURN function:SUBSET, params:[variable:a, variable:b, var
Minima @ 14/12/2022 12:54:31 [14.8 MB] : org.minima.kissvm.exceptions.ExecutionException: SUBSET size too large 2147483(
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.functions.hex.SUBSET.runFunction(SUBSET.java:33)
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.expressions.FunctionExpression.getValue(FunctionExpressi
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.statements.commands.RETURNstatement.execute(RETURNstater
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.statements.StatementBlock.run(StatementBlock.java:56)
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.Contract.run(Contract.java:350)
Minima @ 14/12/2022 12:54:31 [14.8 MB] :      org.minima.kissvm.Contract.main(Contract.java:733)
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[13] STACK[2] - Execution Error - org.minima.kissvm.exceptions.ExecutionExc
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[13] STACK[2] - Contract instructions : 13
Minima @ 14/12/2022 12:54:31 [14.8 MB] : INST[13] STACK[2] - Contract finished     : false
```

Inadequate Encryption Strength

ID	MINIMA-5
Scope	Cryptography, Security
Severity	HIGH
Status	Fixed (false positive)
Vulnerability Type	Inadequate Encryption Strength

Details

Minima uses key length of 1024 bits for for RSA cipher algorithm.

src/org/minima/utils/encrypt/GenerateKey.java:

```
SecureRandom random     = new SecureRandom();
KeyPairGenerator keyGen = KeyPairGenerator.getInstance(ASYMETRIC_ALGORITHM_GEN);
keyGen.initialize(1024, random);
```

Recommendation

Use a key length of at least 2048 bits in order for cryptographic keys to be robust against brute-force attacks.

Fix author comments

This RSA encryption is ONLY used for Maxima - not for Minima

The name of the user is their public key so a 2048 bit key is twice as long and can be problematic to share

Inadequate Encryption Strength

The software stores or transmits sensitive data using an encryption scheme that is theoretically sound, but is not strong enough for the level of protection required.

ID	MINIMA-3
Scope	Cryptography, Security
Severity	HIGH

Status	Fixed (12892f729)
Vulnerability Type	Inadequate Encryption Strength

Details

Weak SSL/TLS protocols should not be used. This rule raises an issue when an insecure TLS protocol version (i.e. a protocol different from "TLSv1.2", "TLSv1.3", "DTLSv1.2", or "DTLSv1.3") is used or allowed. It appears here in the code.

Recommendation

Use TLS 1.2 as the minimum protocol version.

Fix author comments

Upgraded to TLS1.2

Inefficient Regular Expression Complexity

ID	MINIMA-17
Scope	KISS VM, Security
Severity	**HIGH**
Status	Fixed (7164cb87)
Vulnerability Type	Inefficient Regular Expression Complexity

Details

Regular expression patterns that contain unescaped untrusted input can consume arbitrary amounts of CPU time. This security issue was introduced here while fixing existing audit issues.

Recommendation

To properly escape the input, wrap it with `Pattern.quote(...)`

Missing Encryption of Wallet DB

ID	MINIMA-2
Scope	Cryptography, Security
Severity	**HIGH**
Status	Fixed (12892f729)
Vulnerability Type	Missing Encryption of Sensitive Data

Details

Minima Keys are stored unencrypted in WalletDB. This could lead to leaking keys if gained user access to the machine.

Recommendation

Store private keys encrypted.

Fix author comments

Wallet DB can now be AES encrypted with password specified on command line as ENV variable

-dbpassword

Weak Password Requirements

ID	MINIMA-4
Scope	Cryptography, Security
Severity	**HIGH**
Status	Fixed (12892f729)
Vulnerability Type	Weak Password Requirements

Details

A secure password should be used when connecting to a database.

src/org/minima/utils/SqlDB.java:

```
//Create the connection
mSQLConnection = DriverManager.getConnection(h2db, "SA", "");
```

Recommendation

Use secure password when connecting to a database.

Fix author comments

Fixed. you can now specify a -dbpassword on startup which AES encrypts the Wallet SQL db and private keys

Incomplete Cleanup

ID	MINIMA-6
Scope	Code Quality
Severity	**MEDIUM**
Status	New
Vulnerability Type	Incomplete Cleanup

Details

Some resources within Minima codebase are closed in a `try` block. For example:

```
try {
  //...
  //Open the socket..
  Socket sock = new Socket();
  //... Closing in/out streams later here
} catch(Exception exc){
  //...
}
```

This issue appears at:

- src/org/minima/system/network/maxima/MaxMsgHandler.java#L138,L239
- src/org/minima/system/commands/backup/archive.java#L365,L401
- src/org/minima/system/commands/backup/backup.java#142,L164
- src/org/minima/system/mds/MDSManager.java#L561
- src/org/minima/utils/ZipExtractor.java#L35

Recommendation

Use try-with-resources or close the resource in a "finally" clause.

Operation on a Resource after Expiration or Release

The software uses, accesses, or otherwise operates on a resource after that resource has been expired, released, or revoked.

ID	MINIMA-1
Scope	Cryptography, Security
Severity	**LOW**
Status	New
Vulnerability Type	Operation on a Resource after Expiration or Release

Details

Bouncy Castle Cryptography APIs of version 1.69 is used throughout Minima codebase. Zero vulnerabilities have been discovered since that version.

Recommendation

Use the latest release of Bouncy Castle package (1.70).

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Analysis

Tests coverage of all Minima classes

22.8% of lines are covered with unit tests

Coverage Breakdown:

Package	Class, %	Method, %	Line, %
org.minima	0% (0/3)	0% (0/10)	0% (0/92)
org.minima.database	0% (0/1)	0% (0/31)	0% (0/130)
org.minima.database.archive	0% (0/4)	0% (0/34)	0% (0/301)
org.minima.database.cascade	0% (0/2)	0% (0/28)	0% (0/124)
org.minima.database.maxima	0% (0/3)	0% (0/53)	0% (0/270)
org.minima.database.minidapps	0% (0/2)	0% (0/18)	0% (0/78)
org.minima.database.txpowdb	0% (0/1)	0% (0/21)	0% (0/45)
org.minima.database.txpowdb.sql	0% (0/1)	0% (0/9)	0% (0/90)
org.minima.database.userprefs	0% (0/1)	0% (0/27)	0% (0/37)
org.minima.database.userprefs.txndb	0% (0/2)	0% (0/25)	0% (0/90)
org.minima.database.wallet	0% (0/3)	0% (0/46)	0% (0/289)
org.minima.system.brains	0% (0/8)	0% (0/67)	0% (0/1015)
org.minima.system.commands	0% (0/3)	0% (0/34)	0% (0/297)
org.minima.system.commands.backup	0% (0/3)	0% (0/13)	0% (0/366)
org.minima.system.commands.base	0% (0/41)	0% (0/124)	0% (0/1444)
org.minima.system.commands.maxima	0% (0/2)	0% (0/8)	0% (0/244)
org.minima.system.commands.mds	0% (0/1)	0% (0/3)	0% (0/172)
org.minima.system.commands.network	0% (0/8)	0% (0/25)	0% (0/230)
org.minima.system.commands.persistent	0% (0/2)	0% (0/6)	0% (0/55)
org.minima.system.commands.scripts	0% (0/3)	0% (0/9)	0% (0/120)
org.minima.system.commands.search	0% (0/4)	0% (0/12)	0% (0/122)
org.minima.system.commands.signatures	0% (0/2)	0% (0/6)	0% (0/27)
org.minima.system.commands.txn	0% (0/16)	0% (0/51)	0% (0/601)
org.minima.system.genesis	0% (0/3)	0% (0/4)	0% (0/42)
org.minima.system.mds	0% (0/5)	0% (0/32)	0% (0/386)

org.minima.system.mds.handler	0% (0/7)	0% (0/15)	0% (0/325)
org.minima.system.mds.hub	0% (0/4)	0% (0/8)	0% (0/36)
org.minima.system.mds.pending	0% (0/1)	0% (0/5)	0% (0/12)
org.minima.system.mds.polling	0% (0/2)	0% (0/10)	0% (0/33)
org.minima.system.mds.runnable	0% (0/5)	0% (0/33)	0% (0/103)
org.minima.system.mds.runnable.shutter	0% (0/3)	0% (0/6)	0% (0/10)
org.minima.system.mds.sql	0% (0/1)	0% (0/3)	0% (0/44)
org.minima.system.network	0% (0/3)	0% (0/14)	0% (0/85)
org.minima.system.network.maxima	0% (0/4)	0% (0/39)	0% (0/613)
org.minima.system.network.maxima.message	0% (0/5)	0% (0/35)	0% (0/128)
org.minima.system.network.maxima.mls	0% (0/4)	0% (0/38)	0% (0/131)
org.minima.system.network.rpc	0% (0/3)	0% (0/14)	0% (0/147)
org.minima.system.network.webhooks	0% (0/1)	0% (0/8)	0% (0/40)
org.minima.system.sendpoll	0% (0/2)	0% (0/12)	0% (0/49)
org.minima.utils.archivenode	0% (0/2)	0% (0/6)	0% (0/126)
org.minima.utils.encrypt	0% (0/4)	0% (0/30)	0% (0/153)
org.minima.utils.encrypt.javajs	0% (0/1)	0% (0/10)	0% (0/42)
org.minima.utils.ssl	0% (0/2)	0% (0/10)	0% (0/71)
org.minima.system.network.minima	11.1% (1/9)	3.8% (4/106)	1.1% (11/1001)
org.minima.utils.messages	40% (2/5)	12.2% (6/49)	6% (11/183)
org.minima.utils	41.2% (7/17)	25.9% (36/139)	15.9% (159/1000)
org.minima.system.network.p2p.params	50% (1/2)	20% (1/5)	34.8% (16/46)
org.minima.objects	72.2% (13/18)	41.1% (120/292)	40% (672/1678)
org.minima.system.network.p2p.messages	75% (3/4)	35.5% (22/62)	47.2% (108/229)
org.minima.utils.json	75% (3/4)	22.7% (15/66)	24% (93/387)
org.minima.utils.json.parser	75% (3/4)	50% (25/50)	42.4% (225/531)
org.minima.system.network.p2p	77.8% (7/9)	31.2% (34/109)	14.1% (106/753)
org.minima.database.txpowtree	85.7% (6/7)	53.1% (34/64)	44.7% (157/351)
org.minima.database.mmr	100% (6/6)	61.3% (73/119)	43.7% (231/529)
org.minima.database.txpowdb.ram	100% (2/2)	33.3% (7/21)	29.9% (26/87)
org.minima.kissvm	100% (1/1)	86% (37/43)	84.8% (228/269)
org.minima.kissvm.exceptions	100% (3/3)	100% (3/3)	100% (6/6)

org.minima.kissvm.expressions	100% (7/7)	96.7% (29/30)	99.8% (402/403)
org.minima.kissvm.functions	100% (1/1)	100% (14/14)	93.3% (42/45)
org.minima.kissvm.functions.cast	100% (6/6)	75% (18/24)	83.8% (83/99)
org.minima.kissvm.functions.general	100% (4/4)	40% (8/20)	32.4% (22/68)
org.minima.kissvm.functions.hex	100% (8/8)	88.6% (31/35)	85.8% (109/127)
org.minima.kissvm.functions.number	100% (9/9)	94.7% (36/38)	97.9% (95/97)
org.minima.kissvm.functions.sha	100% (4/4)	81.2% (13/16)	73% (54/74)
org.minima.kissvm.functions.sigs	100% (3/3)	92.3% (12/13)	97.8% (45/46)
org.minima.kissvm.functions.state	100% (3/3)	100% (12/12)	100% (30/30)
org.minima.kissvm.functions.string	100% (3/3)	25% (3/12)	16.2% (6/37)
org.minima.kissvm.functions.txn.input	100% (6/6)	87.5% (21/24)	84.2% (80/95)
org.minima.kissvm.functions.txn.output	100% (6/6)	75% (18/24)	74% (74/100)
org.minima.kissvm.statements	100% (2/2)	87.5% (7/8)	98.4% (181/184)
org.minima.kissvm.statements.commands	100% (7/7)	100% (24/24)	94.6% (122/129)
org.minima.kissvm.tokens	100% (3/3)	95.5% (21/22)	94.1% (144/153)
org.minima.kissvm.values	100% (5/5)	95.7% (45/47)	98.1% (106/108)
org.minima.objects.base	100% (4/4)	94% (94/100)	86.6% (252/291)
org.minima.objects.keys	100% (5/5)	74.5% (35/47)	71.6% (146/204)
org.minima.system	100% (1/1)	7.7% (2/26)	1.3% (3/237)
org.minima.system.params	100% (7/7)	60% (39/65)	74.9% (215/287)

22.8% coverage is considered below high code quality, especially for blockchain projects which run in a hostile environment and have a number of attack vectors.

We recommend targeting at least 80% test coverage overall for the project and 100% for critical subsystems (KISSVM, MDS, P2P)

Hacken OÜ
Parda 4, Kesklinn, Tallinn
10151 Harju Maakond, Eesti
Kesklinna, Estonia
support@hacken.io

Disclaimers

Hacken disclaimer

The code base provided for audit has been analyzed according to the latest industry code quality, software processes and cybersecurity practices at the date of this report, with discovered security vulnerabilities and issues the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functional specifications). The report contains no statements or warranties on the identification of all vulnerabilities and security of the code. The report covers the code (branch/tag/commit hash) submitted to and reviewed, so it may not be relevant to any other branch. Do not consider this report as a final and sufficient assessment regarding the utility and safety of the code, bug-free status, or any other contract statements. While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only — we recommend proceeding with several independent audits, public bug bounty program and CI/CD process to ensure security and code quality. English is the original language of the report. The Consultant is not responsible for the correctness of the translated versions.

Technical disclaimer

Protocol Level Systems are deployed and executed on hardware and software underlying platforms and platform dependencies (Operating System, System Libraries, Runtime Virtual Machines, linked libraries, etc.). The platform, programming languages, and other software related to the Protocol Level System may have vulnerabilities that can lead to security issues and exploits. Thus, Consultant cannot guarantee the explicit security of the Protocol system in full execution environment stack (hardware, OS, libraries, etc.)