State of Work 20190118-00-01
Prepared for Monero Research Lab

By Markus Vervier / sales@x41-dsec.de

2019-05-10

X41 D-SEC GmbH
Dennewartstr. 25-27
D-52068 Aachen
Amtsgericht Aachen: HRB19989
VAT-ID: DE 303480751

Document History

Revision	*Date*	*Change*	*Editor*
0	2019-05-10	Finalized SoW	X41 & Secfault

Proposal Recipient

Derek Zimmer
Monero Research Lab

X41 D-Sec GmbH Contacts

Markus Vervier
CEO / Head of Research
Dennewartstr. 25-27
D-52068 Aachen
markus.vervier@x41-dsec.de
+49 241 980 94185

Eric Sesterhenn
Principal Security Consultant
Dennewartstr. 25-27
D-52068 Aachen
eric.sesterhenn@x41-dsec.de
+49 241 980 94186

1 Executive Summary

This is our proposal for an audit of the RandomX proof of work algorithm and implementation based on information given in the call with Howard Chu on 2019-04-30. It includes a breakdown of all proposed services and costs.

RandomX is a proof-of-work (PoW) algorithm that is aims to be hard to implement in hardware to prevent the development of a single-chip ASIC. The goal is to minimize the mining advantages of specialized hardware in comparison to a general-purpose CPU.

- Phase I - Design and Protocol Level Review
- Phase II - Creation of a Test Plan & Workshop
- Phase III - Execution of the Security Assessment and Documentation

Since the RandomX algorithm will be used in the context of the Monero (XMR) cryptocurrency, any flaws and insecurities can have a significant financial impact. Our goal is to uncover such flaws before they are exploited by real adversaries,

X41 D-Sec GmbH will perform a security review in cooperation with Secfault Security GmbH. X41 takes the leadership role and is the contracting party. The work will be performed by a team of experienced professionals in software security and cryptography.

X41 will create a technical report that includes a management summary, a detailed technical description of each finding, a rating using CVSS and CWE metrics as well as a solution advice.

Thank you for giving us the opportunity to propose our services to you. We hope the following information will help you in your decision about our future cooperation.

If you have any queries about the services, costs, or would like to discuss anything in more detail, please feel free to reach out any time to sales@x41-dsec.de.

2 Project Description

RandomX is a proof-of-work (PoW) algorithm that is supposedly hard to implement in hardware to prevent the development of a single-chip ASIC and minimize the mining advantages of specialized hardware in comparison to a general-purpose CPU.

Information was given about the project background in a conference call on 2019-04-30.

In the project X41 D-Sec GmbH will perform a review of the implementation and assumptions.

2.1 Scope

The documentation and implementation to be reviewed is hosted in the git repository at `https://github.com/tevador/RandomX`.

The general idea behind the scheme is to force miners to execute the mining algorithm on a general-purpose CPU, instead of on an ASIC or similar dedicated hardware. In order to achieve this goal, the RandomX implementation requires pseudo-randomly generated machine code to be executed. The generated machine code is supposed to be executed on a virtual machine, which resembles a general purpose CPU closely enough to force any hardware implementation to essentially make use of existing CPU layouts.

Furthermore, the implementation introduces requirements on the mining system's RAM, in order to mitigate possible web or botnet mining approaches.

The VM is specifically designed in order to contain as many common CPU building blocks as possible:

- Instruction fetch/decode
- Branch processing unit
- Caches
- An ALU
- Floating point instructions

- A memory controller and memory access instructions

The VM's CPU is a register machine. A detailed description of the design is provided at `https://github.com/tevador/RandomX/blob/master/doc/design.md`.

The PoW algorithm is driven by a custom PRNG, which generates programs to be executed on the VM from the block hash and the respective NONCE value. The specification demands to chain multiple (currently eight) program executions, in order prevent attackers from identifying possible shortcuts in the generated programs; the general heuristics here is that while an attacker might identify one program that is particularly easy to execute, the overall chance that all programs in the respective chain will be easily executable is sufficiently low. By utilizing hash algorithms, the design aims to prevent attackers from simply "reverse-executing" the overall PoW scheme.

For executing the PoW scheme, there are three main approaches:

- Interpreted execution of the VM opcodes
- A JIT version of the VM
- An AOT compiler

This generally implies that the algorithm will likely be executable on all common CPU architectures, while still maintaining a low parallelization, memory hardness (by using the Argon2d hash function) and a "binding" to actual CPUs instead of ASICs or other dedicated chips.

The implementation built in C and C++ and contains about 7000 lines of code (excluding comments).

2.2 Objectives

The main objective of this test is to identify vulnerabilities present in the design or implementation that allow the efficient implementation of RandomX in hardware or otherwise affect the security of its users.

Possible vulnerabilities might include (but are not limited to):

- Shortcuts in the algorithm, allowing for the PoW scheme to be executed more efficiently
- Cryptographic weaknesses (e.g., in the hashing or PRNG functions, which might result in biases in the generated instructions)
- Possible means to build dedicated hardware for significantly speeding up the PoW computation
- Violations of the one-way property of the PoW scheme (i.e., ways to "reverse-execute" parts of the scheme in order to speed up the computation)
- Implementation vulnerabilities in the compilers or JIT implementation, resulting in DoS vulnerabilities or the execution or unwanted code

The following subsection describes how the audit team will approach the project in order to identify vulnerabilities related to the above-mentioned possible issue types.

2.3 Activities

A team of security specialists will perform an audit of the design and source code of RandomX in 27 to 30 person-days. Additionally, we will identify, which parts can be implemented in hardware efficiently. If less days than anticipated are used, billing will be done according to the actual efforts.

- Phase I - Design and Protocol Level Review
- Phase II - Creation of a Test Plan & Workshop
- Phase III - Execution of the Security Assessment and Documentation

2.3.1 Phase I - Design and Protocol Level Review

The assessment will be performed following a white-box methodology. This means that the design documents and source code will be available to X41 D-Sec GmbH. Details about the target code will be given to the consultants beforehand. This methodology generally yields high-quality results, as it significantly reduces the amount of uncertainty and guesswork about the target implementation.

In a first step, X41 D-Sec GmbH will review the design documents and specification for architectural security flaws. This will uncover any logical issues inherent in the design and potential grey areas, where no specific behavior is defined. Furthermore, this will help to identify potential diversions from the specification during the source code audit.

The provided software will be subject to a cursory inspection in order to identify general programming practices, cryptographic parts and potential areas of interest from a security point of view. Please be aware that this is step is not an in-depth code review, but rather serves the purpose of generally understanding the code structure and the overall design of the solution.

2.3.2 Phase II - Test Plan and Conference Call Workshop

During the second project phase, the project team will build a light threat model, aiming to identify and rate the most important risks for the target platform. Although the threat modeling has not been performed yet, it is still possible to describe the most likely threat classes and areas that will be encountered:

- Design and Logic Issues
- Cryptographic Problems
- Implementation Bugs

Following the creation of the light threat model, the project team will coordinate further testing and analysis steps with RandomX in a workshop (e.g., by means of a conference call or web meeting).

2.3.2.1 Design and Logic Issues A solid consensus algorithm is one of the core components of blockchain-based solutions. The proposed PoW algorithm should therefore be carefully analyzed for potential logic issues in its design or implementation. Such issues could include means to significantly improve the computation speed of the algorithm in order to gain an unfair advantage over other miners, but also issues that could be leveraged to mount DoS attacks against he honest participants in the network.

Possible issues and focus areas related to these aspects will be identified and described during Phase II of the project. This will later on guide the execution of Phase III, in which an in-depth analysis of the identified areas will be concluded.

2.3.2.2 Cryptographic Problems The proposed PoW scheme relies on cryptographic algorithms such as a PRNG and hash functions. In contrast to "classical" implementations, these however do not serve the purpose of providing well-understood goals such as confidentiality or integrity, but are rather intended to make the PoW algorithm hard to compute. Possible issues in this area could include biases in the generated random numbers (which could result in an overall computation speedup of the randomly-generated code), or means to reduce the memory hardness of the algorithm in order to build specialized hardware implementations.

Possibly interesting focus areas related to such attacks will therefore be analyzed during phase II of the project, so that during the execution of phase III the audit team can directly focus on the respective parts of the solution.

2.3.2.3 Implementation Bugs A third type of issues that could arise in RandomX are implementation issues. The system makes use of a custom VM implementation, along with an interpreter, a JIT and an AOT compiler. It should be noted that all participants of the PoW scheme (including non-mining nodes that only verify the PoW results) will be required to use one of the VM implementations. Therefore, during phase II of the project, the audit team will aim to identify possible threats related to the handling of the VM code. Such threats could for instance include memory safety issues in the VM implementation or possible DoS conditions. The results of phase II will be used during phase III, where an in-depth review of the identified code areas will be performed.

2.3.3 Phase III - Execution of the Security Assessment and Documentation

Source code auditing will be employed to identify the relevant vulnerabilities and bugs the implementation. Results from the previous two phases are used to quickly identify threats.

2.3.3.1 Theoretical Review A theoretical review of the concepts and their aimed hardness properties will be conducted. The testers will try to reduce the hardness properties of RandomX to hardness of the individual building blocks such as the hash algorithms used. This also includes the identification of possible means to "short-cut" the PoW algorithm, e.g., by leveraging properties emerging from the design of the cryptographic components or from the instruction set. Such issues could for instance include biases in the distribution of the pseudo-randomly generated instructions. Furthermore, the project team will aim to identify possible ways for "reverse-executing" parts of the PoW algorithm, which might result in possible meet-in-the-middle attacks.

Following this, practical attacks will be used to identify possible flaws. An example could be the application of SMT-Solvers to find faster and easier solutions for problem instances. While the problem of solving SMT-instances is an NP-complete problem in general, solutions might be found efficiently for specific sub-classes of problem instances.

This strategy will yield sound assumptions about the theoretical and practical limits of RandomX.

2.3.3.2 Analysis of Possible Hardware Implementations Additionally, it will be attempted to identify which parts can be implemented in hardware to perform execution quicker than on a general purpose CPU. This part of the assessment would look into possible theoretical custom hardware (RTL targeting FPGAs or ASICs) to implement a more cost-effective approach to solving RandomX than existing CPUs. One approach to be evaluated is a full hardware implementation of a pipelined, superscalar and possibly out-of-order CPU design targeting the RandomX ISA and a subset of its virtual peripherals. An estimate will be made of the RandomX CPU MIPS or PoW HPS that such an implementation could attain. Such an analysis would provide clues on whether custom 'mining' hardware is economically viable to design, produce and run on high-end FPGA hardware and/or ASICs targeting various fabrication processes.

2.3.3.3 Analysis for Implementation Vulnerabilities Besides the above-described application-level issues, X41 D-Sec GmbH will also focus on reviewing the implementation of the VM itself, along with the interpreter and compiler. Potential threats in this area could include the involuntary generation of malicious machine code or general C/C++-related implementation flaws such as memory issues (e.g., Buffer Overflows), data races or arithmetic problems. While X41 D-Sec GmbH understands that attacker can likely not freely provide target programs for being executed on the VM, it still might be possible to enumerate potentially harmful instruction sequences by performing a brute-force search.

It should be noted that – due to the nature of the PoW scheme and its envisioned use – the identification of DoS vulnerabilities could also pose a significant threat to the overall system. Therefore, the audit process will generally also include possible ways for mounting DoS attacks against honest PoW verifiers in the system.

2.4 Deliverables

All results are documented in a technical report. This report includes a management summary as well as technical details of the identified vulnerabilities. Additionally a general description of the system, the scope and general recommendations are given.

In conclusion X41 D-Sec GmbH will conduct a source code audit on the implementation as well as a security audit on the design and architecture to identify vulnerabilities and weaknesses. The test is conducted remote in a white-box fashion using an 27 to 30 person-days of security auditing and documentation.

2.5 Project Investment

Security Code Audit				
1	Audit	Design and source code audit	42,000.00 € x 1FP	42,000.00 €
SUBTOTAL				**42,000.00 €**
VAT 19%				0.00 €
TOTAL				42,000.00 €

3 Team

The following team of senior security experts will be working on this project.

3.1 Markus Vervier

Markus Vervier is Head of Research and Managing Director at X41 D-Sec GmbH. Software security is his main focus of work. During the last 15 years of professional experience in offensive IT security he worked as a penetration tester and security consultant and was doing active security research.

Notable works include:

- Extensive experience in the field of code-review, reverse engineering, and vulnerability analysis of applications on various platforms and architectures;
- reverse engineering and security analysis of embedded firmware for mobile devices (Android device baseband firmware);
- discovery of the first vulnerabilities in the *Signal Private Messenger*[1];
- speaker at Infiltrate, HITBSECCONF, and Troopers security conferences about offensive security topics such as baseband reverse engineering and application security;
- memory corruption vulnerability in *libOTR*[2].

3.2 Gregor Kopf

Gregor Kopf is an IT security expert with several years of applied security experience. He graduated in computer science (Dipl. Inf.), with a thesis about a novel PKI system (awarded as "outstanding" with the faculty price). From 2005 on, he performed penetration testing and security consulting engagements. In 2016, he co-founded Secfault

[1] https://pwnaccelerator.github.io/2016/signal-part1.html

[2] https://x41-dsec.de/lab/advisories/x41-2016-001-libotr/

Security GmbH together with Dirk Breiden.

Notable works include:

- extensive experience in the field of code-review, cryptography and reverse engineering on various and architectures;
- identification of various critical vulnerabilities in high-profile targets (including Apple iOS CVE-2011-0228);
- speaker at various security conferences (e.g., DEFCON, CONFidence, SIGINT) about various topics in offensive security;
- various publications (e.g., "Non-Obvious Bugs by Example" [3] or "Secure Function Evaluation vs. Deniability"[4] [5]);

3.3 Serge Bazanski

Serge Bazanski is an expert in hardware and embedded security. Previous work includes:

- Reverse engineering of hardware solutions;
- extracting secrets from hardware and software;
- reverse engineering and security analysis of protocols.

Serge is actively playing CTF contests and publishing writeups of advanced CTF challenges.

3.4 Eric Sesterhenn

Eric Sesterhenn is working as an IT Security consultant for more than 15 years, working mostly in the areas of penetration testing and source code auditing.

[3] http://gregorkopf.de/blog/security/nonObvious.html

[4] http://gregorkopf.de/blog/security/sfe.html

[5] http://www.phrack.org/issues.html?issue=68&id=14

Notable works include:

- Identified vulnerabilities in various software projects including the Linux kernel and X.org[6];
- analysis of complex software applications and infrastructures and extensive experience in code reviewing, penetration testing, and vulnerability analysis;
- speaker at DEF CON, beVX 2018 and 35c3 about smartcard driver security[7];
- speaker at Nullcon 2018 about security issues in IoT OS[8];
- speaker at zeronights 2018 about security issues in fax software[9];
- part of the winning team of the Deutsche Post Security Cup 2013.

3.5 Luis Merino

Luis Merino is a Security Engineer with over five years of experience in designing, implementing and reviewing security-sensitive systems. He has presented his work on Secure Enclaves at Black Hat and other security conferences.

Previous work includes:

- Security analysis of Intel SGX (talk at Blackhat 2016);
- design of crypto systems;
- vulnerability analysis of embedded devices and fax machines.

[6] https://www.x41-dsec.de/lab/advisories/x41-2017-001-xorg/

[7] https://www.x41-dsec.de/lab/blog/smartcards/

[8] https://nullcon.net/website/goa-2018/speakers/eric-sesterhenn.php

[9] https://2018.zeronights.ru/en/reports/zero-fax-given/

4 References

The following references show a relevant part of the experience that X41 D-Sec GmbH has in the field.

4.1 Customer References

X41 D-Sec GmbH regards confidentiality and protection of customers as essential. Information about customers is only given with their explicit permission. The following customers published all results after X41 D-Sec GmbH successfully conducted the project:

4.1.1 The Quantum Resistent Ledger

The project has been executed in a joint effort with Secfault Security GmbH to combine the best expertise in all required fields.

Reference Contact:

Juan Leni
Senior Consultant
E-Mail: juan@theqrl.org

Provided services:

- Design-level review of the solution
- Analysis of the used cryptographic primitices and protocols
- Source code audit of the solution

4.1.2 Wire - Cryptography and Application Level Audit (with Kudelski)

Reference Contact:

Alan Duric
CTO & Wire Swiss GmbH
E-Mail: alan@wire.com

Provided Services:

- Cryptography audit of the secure messaging protocol
- Review of protocol implementation to identify security flaws
- Review of the Wire Web-, Android, and iOS-Client

4.1.3 Browser Security White Paper

Reference Contact:

Andrew Fife
Program Manager Chrome Enterprise& Google LLC
E-Mail: afife@google.com

Provided Services:

- Independent security research
- Review of modern web-browser design and implementation
- Comparison of security mitigations in effectiveness
- `https://browser-security.x41-dsec.de/X41-Browser-Security-White-Paper.pdf`

4.1.4 Mozilla Firefox

Reference Contact:

Julien Vehent
Security Engineering Manager at Mozilla, DevSecOps Leader E-Mail: jvehent@mozilla.com

Provided Services:

- Application security (source code review)
- Penetration testing

5 Insurance

X41 D-Sec GmbH is insured by *Hiscox Europe Underwriting Limited* during research projects with a professional liability insurance of up to 1,000,000 €. Necessary documents can be provided on request.

6 About X41 D-Sec GmbH

X41 D-Sec GmbH is an expert provider for application security and penetration testing services. Having extensive industry experience and expertise in the area of information security, a strong core security team of world-class security experts enables X41 D-Sec GmbH to perform premium security services.

X41 D-Sec GmbH has the following references that show their experience in the field:

- Review of the Mozilla Firefox updater[10]
- X41 Browser Security White Paper[11]
- Review of Cryptographic Protocols (Wire)[12]
- Identification of flaws in Fax Machines[13][14]
- SmartCard Stack Fuzzing[15]

The testers at X41 D-Sec GmbH have extensive experience with penetration testing and red teaming exercises in complex environments. This includes enterprise environments with thousands of users and vendor infrastructures such as the Mozilla Firefox Updater (Balrog).

Fields of expertise in the area of application security encompass security-centered code reviews, binary reverse-engineering and vulnerability-discovery. Custom research and IT security consulting, as well as support services, are the core competencies of X41 D-Sec GmbH. The team has a strong technical background and performs security reviews of complex and high-profile applications such as Google Chrome and Microsoft Edge web browsers.

X41 D-Sec GmbH can be reached via `https://x41-dsec.de` or `mailto:info@x41-dsec.de`.

[10] `https://blog.mozilla.org/security/2018/10/09/trusting-the-delivery-of-firefox-updates/`
[11] `https://browser-security.x41-dsec.de/X41-Browser-Security-White-Paper.pdf`
[12] `https://www.x41-dsec.de/reports/Kudelski-X41-Wire-Report-phase1-20170208.pdf`
[13] `https://www.x41-dsec.de/lab/blog/fax/`
[14] `https://2018.zeronights.ru/en/reports/zero-fax-given/`
[15] `https://www.x41-dsec.de/lab/blog/smartcards/`

7 Terms and Conditions

This offer is subject to the following terms and conditions:

1. Location of provided services: remote or on-site if requested.
2. Travel expenses will be billed according to the actual expenditures.
3. Any additional services need a separate commercial agreement.
4. A separate penetration testing agreement will be signed before the test in order to define and verify the legal scope of all tests.
5. All prices quoted are net, excluding VAT. All prices are quoted in €.
6. Reverse Charge - According to Article 194, 196 of Council Directive 2006/112/EEC, VAT liability rests with the service recipient.
7. One person-day (PD) equates eight hours.
8. Latest end of project: 2019-12-31
9. This quotation is a non-binding offer until your order is confirmed by X41 D-Sec GmbH and an order confirmation is sent to you. In case of sending an order confirmation by X41 D-Sec GmbH the quotation gets a binding offer.
10. This proposal is valid until 2019-08-01.
11. Invoices are due on receipt payment should be made within 14 days.
12. Payment terms: 30% at project start, 100% after delivery of the final report.
13. This offer and its contents are intellectual property of X41 D-Sec GmbH. Any changes need to be approved in writing by X41 D-Sec GmbH.

8 Acceptance

8.1 Client

I hereby confirm the acceptance and accept all terms and conditions as defined in this proposal:

Full Name

Title

Date

Signature (Monero Research Lab)

Full Name

Title

Date

Signature (Monero Research Lab)

Please submit two signed copies by mail to X41 D-Sec GmbH. A preliminary order may be sent via email or fax in order to allocate resources and speed up the beginning of the project.

8.2 X41 D-Sec GmbH

We hereby confirm the order:

Full Name

Title

Date

Signature (X41 D-Sec GmbH)

Confirmed Date of Project Begin