VULNERABILITY ASSESSMENT

Ravencoin

SENSITIVE INFORMATION – DISTRIBUTE WITH CAUTION

Executive Summary

Ravencoin engaged Independent Security Evaluators (ISE) to evaluate the security posture of the Ravencoin project, an open-source peer-to-peer blockchain used to enable transactions between users, with a specific focus on asset creation and transfer functionality. ISE performed an assessment to discover vulnerabilities within the system that could lead to the compromise of the Ravencoin asset layer, the integrity of user assets, or the availability of the service.

ISE considers attack surfaces, permissions, and application logic specific to Ravencoin that an advanced attacker may exploit, and manually tests against such exploits. ISE uses automated tools to gain an understanding of the system and identify common issues, but the focus of the assessment is discovering vulnerabilities that scanners will miss. ISE reviews all reported findings for accuracy and assigns severity based on exploit complexity, impact, and attack chaining.

ISE has assessed the following components:

- Asset Creation Functionality
- Asset Transfer Functionality
- Asset Related RPCs
- Source Code Review
- Review of KAWPOW
- Review of Distributable Binaries

While the focus of this assessment was on asset creation and transfer functionality, implementation of KAWPOW, and messaging, ISE did not uncover any security defects that would allow for unintended asset creation, asset transfer, or denial of service within the system. A summary of discovered issues in adjacent components is shown in the table below.

	Critical	High	Medium	Low	Info	Total
Discovered	0	0	0	3	2	5
Resolved	0	0	0	0	0	0
Closed	0	0	0	0	0	0
Remaining	**0**	**0**	**0**	**3**	**2**	**5**

A history of ISE's assessments is shown in the table below.

Revision	Date	Description
1	Nov – Dec 2020	Initial assessment of Ravencoin.

This report expires on December 31, 2021. Expiration facilitates ongoing communication and assessment efforts to address changes in technology, the product, and its supporting environment. ISE recommends that Ravencoin takes action to address current issues and continue security assessments to identify additional issues. A list of recommendations for further security review can be found in the Additional Recommendations section.

The following tables contain the vulnerabilities and strategic weaknesses ISE has identified in Ravencoin. Each issue is assigned a severity that is derived from the issue's impact to Ravencoin's assets and its exploitability. More information can be found in the Severity Ratings and Statuses sections.

Summary of Vulnerabilities

ISE discovered the following vulnerabilities, listed with their associated severity and status. For more information, visit the Vulnerabilities section.

RAVENCOIN CORE COMPONENTS

Vulnerability	Identifier	Severity	Status
Use of Out-of-Date Bitcoin Fork	ISE-RAVEN-2020-01	Low	Unresolved
Faulty Conditional Logic	ISE-RAVEN-2020-02	Info	Unresolved

Summary of Strategic Weaknesses

ISE discovered the following strategic weaknesses, listed with their associated severity and status. Strategic weaknesses may not be directly exploitable but could lead to additional security concerns over time. For more information, visit the Strategic Weaknesses section.

RAVENCOIN CORE COMPONENTS

Weakness	Identifier	Severity	Status
Use of Out-of-Date Qt Framework	ISE-RAVEN-2020-03	Low	Unresolved
Windows Signed Installer uses Untrusted Certificate	ISE-RAVEN-2020-04	Low	Unresolved
Lack of Signed Binaries	ISE-RAVEN-2020-05	Info	Unresolved

Table of Contents

Introduction

Ravencoin contracted Independent Security Evaluators (ISE) to evaluate the security of the Ravencoin platform. ISE performed an assessment to discover vulnerabilities within Ravencoin that could lead to unwanted results, such as compromise of assets or user information, disruption of service, or leveraging Ravencoin's systems or functionality for other attacks.

System Overview

Ravencoin is a peer-to-peer blockchain project with the focus on asset creation, transfer, and governance. The majority of the codebase stems from a fork from Bitcoin, consisting mostly of C and C++ code. Ravencoin offers approximately 169 RPC endpoints, and while many overlap with Bitcoin RPCs, the RPCs that largely differ or are unique to Ravencoin are listed below (Table 1). Most of the new RPC functionality revolves around asset issuance, transfer, tagging, and messaging.

Ravencoin Specific RPCs			
getaddressbalance	transferfromaddress	addtagtoaddress	removetagfromaddress
getaddressdeltas	transferfromaddresses	checkaddressrestriction	transferqualifier
getaddressmempool	clearmempool	checkaddresstag	unfreezeaddress
getaddresstxids	decodeblock	checkglobalrestriction	unfreezerestrictedasset
getaddressutxos	getblockhashes	freezeaddress	viewmyrestrictedaddresses
getassetdata	getspentinfo	freezerestrictedasset	viewmytaggedaddresses
getcacheinfo	setgenerate	getverifierstring	cancelsnapshotrequest
getsnapshot	clearmessages	issuequalifierasset	distributereward
issue	sendmessage	issuerestrictedasset	getdistributestatus
issueunique	subscribetochannel	isvalidverifierstring	getsnapshotrequest
listassets	unsubscribefromchannel	listaddressesfortag	listsnapshotrequests
listmyassets	viewallmessagechannels	listaddressrestrictions	requestsnapshot
purgesnapshot	viewallmessages	listglobalrestrictions	getmasterkeyinfo
reissue	getkawpowhash	listtagsforaddress	getmywords
transfer	pprpcsb	reissuerestrictedasset	sendfromaddress

Table 1. Ravencoin specific RPCs.

RAVEN DAEMON

The Raven daemon is the backend service responsible for providing JSON-based RPCs to interact with the Raven blockchain. The daemon is a fork of Bitcoin 0.15.0 released in mid-2017. The daemon is mostly written in C and C++ and supports Windows, Linux, and macOS.

RAVEN DESKTOP CLIENT

The Raven desktop client uses an open source-cross platform graphical user interface framework, Qt, version 5.7.1. User friendly GUI elements allow users to easily interact with the underlying Raven services to send Ravencoin, issue assets, transfer assets, and manage assets. The desktop client communicates with external services (api.github.com and api.binance.com) to get software version information and Ravencoin market rates.

Scope

ISE's evaluation covered the following components of Ravencoin:

- Ravend, Raven-qt, version 4.3.2.1
- Ravencoin specific RPCs shown in Table 1 above.

The following were not included in ISE's assessment and should be the focus of future assessments:

- iOS Wallet
- Android Wallet

Methodology

ISE specializes in hands-on assessments that consider assets, attack surfaces, permissions, and logic specific to the target system. The goal is to discover security issues that a variety of attackers may exploit, and manually test against such exploits using access to the platform, e.g., test accounts, server access, source code, documentation, etc. In general, ISE encourages sharing as much relevant access as possible because a deeper knowledge of the system facilitates more efficient testing and more valuable results.

ISE performed the assessment of Ravencoin with access to the following resources:

- Source code located at https://github.com/RavenProject/Ravencoin/tree/v4.3.2
- Documentation and build instructions

ISE used automated tools to gain an understanding of the system and identify common issues, but the focus of the assessment was discovering vulnerabilities that scanners will typically miss. ISE reviews all reported findings for accuracy and assigns severity based on exploit complexity, impact, and attack chaining.

Below is a list of core areas of testing, including but not limited to:

Ravencoin/Bitcoin Code Delta:

ISE conducted a review of code changes integrated since the Bitcoin 0.15.x code fork for security defects that may affect the usability, stability, or security guarantees provided by Ravencoin. This review was meant to familiarize ISE with Ravencoin implementation details as well as the identification of functionality that may be considered a security defect. This review consisted of a diff based analysis approach (to eliminate Bitcoin core code, which was out of scope for this engagement), along with static code review, and dynamic runtime testing. Results of these tests resulted in the identification of 'ISE-RAVEN-2020-02'. Additional tests were performed on frameworks or libraries used. Results of these tests resulted in the identification of 'ISE-RAVEN-2020-01' and 'ISE-RAVEN-2020-03'.

Ravencoin Asset Layer:

ISE conducted a review of Ravencoin Asset creation, reissue, messaging, and transfer workflows. The bulk of ISE resources was dedicated to this area. Review of the asset layer was done by performing a source code review, dynamic analysis/fuzzing of functionality, and evaluation of associated script code. Asset related RPCs and code paths were also reviewed to ensure implied security guarantees could not be violated. That is, asset workflows should not lead to the unauthorized generation of Ravencoin itself, nor should asset quantities be modified in an unintended way. The results of analysis in this area were negative. ISE did not find any functionality that could be used in unintended ways to bypass or exploit the creation, reissue, and transfer of assets. Nor could assets, or messaging be used to create an application layer of denial-of-service scenario through exploitation of application layer asset related code.

Ravencoin Consensus:

ISE performed a review of Ravencoin's implementation of a modified ProgPoW, created to help prevent centralization, adopted as KAWPOW. This entailed a code review of Ravencoin specific changes, and tuning parameters. ISE noted no implementation defects in KAWPOW.

Ravencoin Binaries:

ISE reviewed Ravencoin's installation workflows and binaries to identify items that would reduce the security of the distribution channel, or lack of mitigation protections against runtime exploits. Results of these tests resulted in 'ISE-RAVEN-2020-04' and 'ISE-RAVEN-2020-05'.

Timeline

This section includes a summary of the history of ISE's engagement with Ravencoin.

1: DECEMBER 2020 – INITIAL ASSESSMENT OF RAVENCOIN

ISE conducted an initial assessment of Ravencoin v4.3.2 focusing on Asset creation and transfer in order to protect the reputation of Ravencoin and its asset creators and stakeholders. This assessment discovered 2 implementation-level vulnerabilities, and 3 strategic weaknesses.

Components tested:

- Ravencoin, version 4.3.2

Threat Model

Any robust defensive model requires a thorough understanding of the system, its assets, and the threats targeting it. An adversary-focused threat model allows companies to manage risk by directing resources toward threats and vulnerabilities that pose the greatest risk.

Assets

Before accurately assessing a system's security posture, it is necessary to identify assets and their value. Assets include tangible elements such as information or equipment and extend to more abstract elements such as reputation. The impact of the loss of these assets should be quantified to the degree possible; however, this can be a difficult and subjective process and is outside the scope of ISE's insight into the client's operations.

FINANCIAL ASSETS

Ravencoin users' financial assets are the primary target of an attacker. The motivators for this attack could vary, but mainly the goal would be to take ownership of assets or use exploits to create unauthorized tokens or assets. This could include a competitor who only wishes to cause financial loss to Ravencoin or an attacker that wishes to obtain direct financial gain from Ravencoin users loss. All attacks and vulnerabilities will affect this asset either directly or indirectly, so ISE will only associate this asset with a vulnerability where it has a direct impact.

ACCESS AND AVAILABILITY

Because Ravencoin is integral to token and asset creation and transfer, its access and availability must be protected from attack. A vulnerability resulting in the denial of service (DoS) to Ravencoin's service could lead to delays in processing and negatively impact Ravencoin's reputation. Vulnerabilities that allow DoS can range from simply not rate limiting and authenticating high-cost requests to persistent crashes in a software application.

BUSINESS REPUTATION

The reputation of Ravencoin and its software is also a primary concern. Threats that would likely target reputation are typically competitors or politically motivated parties. The effect on reputation is inherent to any security risks in the system that may be exposed to Ravencoin's stakeholders.

Threats

ISE considers the following threats when assessing a system. Depending on the attack surfaces of a system and its reputation, certain threats may be of more concern than others. The capabilities and resources of each threat is also a factor when considering the complexity of a given vulnerability and its exploit.

NATION STATE INTELLIGENCE (ADVANCED PERSISTENT THREAT)

Nation states represent the most capable threat actors in the realm of computer network exploitation. Beyond having the largest budgets and payrolls, nation states have unique capabilities in terms of access to supply chains and human agents. They can rely upon all of a nation's resources, such as intelligence infrastructures, to gather as much data as possible against an enemy, engage in active information system damage, and to set specific objectives for their hacking program.

CORPORATE SPONSORED ESPIONAGE

Corporations, particularly certain overseas corporate environments, will at times utilize espionage techniques to gain advantage over competitors or save on research and development. Corporations have significant budgets and can hire professional teams to conduct computer network exploitation of their competitors' network. Targeted assets are likely business plans, financial information, proprietary software, etc., and corporations may benefit from harming a competitor's reputation and availability of services.

HACKER GROUP

This adversary includes hacker organizations such as the group "Anonymous." They are motivated by socio-political activities, pride, fun, and notoriety. They choose high profile, opportunistic targets over high value targets, and typically disclose stolen information rather than use it for identity theft or profit. Common activities include web-site defacement, "doxing", releasing embarrassing internal communications, and denial of service or other acts of cyber-vandalism.

Other noteworthy groups include those such as the "Russian Business Network" (RBN). These groups are a for-profit organized crime syndicate focusing primarily on cyber-crime activities, such as identity theft, for hire targeted denial-of-service attacks, black market exchange of botnets, exploits and other information, and illegal hosting of copyrighted materials.

INDIVIDUAL HACKER

Individual hacker capabilities vary widely from those only able to apply existing tools, all the way to a highly professional individual who is capable of custom exploit development. Generally, these threat actors have limited budgets and time; however, if properly motivated by a perceived slight or challenge, they may be persistent.

INSIDER THREAT

The insider threat encompasses any employee, contractor, or other individual who has some level of authorized access to the system. Often, these are the most dangerous threats, as they have already bypassed the outer layers of defense and have a foothold on the system. Insider threats may be malicious, but more often they are employees who are uninformed in security practices and make mistakes.

Secure Design Principles

ISE considers the following secure design principles in an assessment. Most issues that ISE encounters are a direct result of violating at least one of these principles.

TRUST

A trust model clearly defines the assumptions made by the system and may highlight improper assumptions on the part of the designers. Unauthenticated API calls, unverified signatures and certificate chains, and lack of input validation and sanitization are all examples of misplaced trust.

SECURE-BY-DEFAULT

A system is secure-by-default if the out-of-the-box settings put the system in a secure state. A corollary is that the secure state must be the easiest state to obtain and maintain—as users will typically choose convenience over security. Frequently, applications integrate with third-party or cloud services and the secure configuration of such services is also a consideration for this principle.

DEFENSE-IN-DEPTH

Defense in depth seeks to array layers of defensive measures such that the failure of any one component will not lead to total system compromise. The classic physical world example is the concentric walls of a fortification. Regarding software development and networking, examples of defense in depth are deploying firewalls, opening only the minimum set of ports needed for the system to function, and following any existing company recommended hardening practices.

FAIL-SECURE

Fail-secure refers to the tendency for a failure of the system to leave it in a secure state as opposed to an insecure state. For example, if an electronic lock were to failsafe under a power loss, it would remain locked rather than unlocked. From a software perspective, incorrect error handling is a common example, e.g., an application may disclose sensitive information upon receiving malformed input.

AUDIT

Audit is a critical part of the system that assists in recovery, investigation, and deterrence. Successful auditing will include a combination of logging and intrusion detection. Logging allows the organization to record information that assists in improving software and retracing the steps of a breach. Intrusion detection may come in the form of a system firewall, web application firewall, or IDS, and should provide information about who is accessing the system at all times in order to detect and potentially block attacks.

IDENTITY

Identification is the process of distinguishing users. It is closely related to authentication (verifying that identity—the two generally abbreviated as I&A), and authorization (granting permissions to users). In terms of identity there is one golden rule: do not share user identities. Users should be accountable for their actions, and the sharing of identities undermines this accountability. This principle is also a requirement for assigning the least privilege to a user and their processes.

AUTHENTICATION (RBAC AND MFA)

Authentication mechanisms fall into three general classes: something one knows (knowledge factor, e.g. passwords), something one possesses (possession factor, e.g. RFID key fobs), and something one is (biometrics factor, e.g. fingerprints). Standards and regulations might govern how to accomplish

authentication tasks and to tailor policy; a business decision may be necessary to use a certain required standard, but in general, ISE recommends following NIST authentication standards when possible.

Passwords are the most commonly used mechanism for authenticating an entity to a system. However, they are also notorious for being used and implemented incorrectly. Authentication becomes stronger when multi-factor authentication (MFA) is required. MFA may be required by a third-party customer's internal policies, but also may cause limitations for usability that affect the user experience. Based on its feasibility, MFA can be implemented to further protect systems holding sensitive data.

Another aspect of authentication is session control, i.e. how a system maintains a user's session. Some examples of session management are expiration of sessions after a threshold of inactivity and proper invalidation of a session upon logout.

AUTHORIZATION (LEAST PRIVILEGE)

User authorization is concerned with the privileges that a user and the processes that work on behalf of that user can do on the system. The principle of least privilege refers to the principle that a task should be accomplished with the lowest level of privilege required.

CRYPTOGRAPHY

When a system needs to implement cryptography, it should use industry standard, vetted cryptographic techniques and libraries specific for a task. This is often required by regulation or industry standards. Misuse of cryptography occurs frequently and is often due to using a system that is not secure-by-default. Verifying cryptographic algorithm suites, the generation of random numbers, and the management of cryptographic keys is crucial to security when using cryptography.

PATCH MANAGEMENT

Many attacks, especially from opportunistic actors, are done by exploiting vulnerabilities in software that has already been fixed, but not up to date in a targeted system. More sophisticated attackers may still target these easily exploitable attack surfaces of the system. ISE suggests applying consistent patches to all software that is being used, whether it be an application, operating system, or library. A company procedure should be in place to test patches before deploying them into production.

Severity Ratings

The severity ratings given in this report include critical, high, medium, low, informational, and unknown, with critical indicating the most severe, low the least, and unknown expressing that insufficient information was available to make a proper assessment. In determining severity, ISE takes into consideration public vulnerability ranking systems; however, in practice, the severity of vulnerabilities varies widely with actual deployment, configuration, and implementation of systems, as well as the value of assets protected and the perceived and anticipated threats to those assets. Thus, the severity ratings chosen here are custom to the system or infrastructure evaluated, and not copied from these sources verbatim.

The two metrics that most affect severity are exposure and impact of a successful attack. Exposure is a combination of elements including how accessible a vulnerable system is and the ease in which an attack can be performed. Impact is determined by factors such as asset value and damage to those assets. The following chart illustrates how severity is assigned:

CRITICAL	These are issues that are either readily exploitable or of substantial exposure paired with excessive damage should an attack be successful. In some cases, the risk of discovery may be lower, but the significance of the damage warrants immediate attention.
HIGH	These issues expose the system or infrastructure heavily but are either not readily exploitable or require additional attack material to exploit successfully.
MEDIUM	These issues alone do not present significant risk to the system or infrastructure but could lead to a successful attack when leveraged with other medium or high severity issues.
LOW	Low severity issues do not pose an immediate threat to the most valuable assets or represent partial exposure.
UNKNOWN	Issues of unknown severity typically could not be fully assessed due to scope or a lack of resources such as source code. They are a security concern that must be addressed through additional investigation to either assign a severity or eliminate the issue.
INFO	Informational issues are unlikely to be a threat to the system but provide important information that stakeholders should be aware of, especially if they could be affected by future changes to the system.

Statuses

The following are descriptions of the various statuses with which ISE marks reported issues. Unresolved issues are unmarked, while other statuses reflect potential changes in a reported issue throughout the remediation process.

RESOLVED Resolved issues have been remediated. Occasionally, the implemented mitigations may no longer be effective, or ISE identifies additional instances of the issue. In these cases, the issue may become unresolved again.

PARTIAL Partially resolved issues have been mitigated to an extent, but not fully resolved. The meaning may depend on the context of the issue. For example, an issue with several different instances may be partially resolved if only a portion of the instances are resolved.

DEFERRED Deferred issues are unresolved; however, the client acknowledges the issue and a remediation plan is in place. This status is used to reflect the client's intention to fix the issue in the near future.

CLOSED Closed issues are unresolved or partially resolved, but the client is aware of their impact and accepts them as a risk. ISE's policy is to only close issues that do not have a direct impact to the security of a system and their remediation costs outweigh the security benefits.

Assessment Results

ISE discovered the following issues within Ravencoin. Issues are categorized as vulnerabilities and strategic weaknesses. ISE provides recommendations for addressing identified issues in the form of resolutions and mitigations. Resolutions fully remediate the issue, while mitigations reduce the risk of, but may not completely remediate, the issue.

IMPORTANT NOTE: ISE's goal is to discover as many issues as possible within the boundaries of an assessment's budget and scope. However, undiscovered issues or additional instances of reported issues may always exist within a system. Ravencoin should work with ISE to implement remediations for issues in this report and conduct ongoing assessments to address code and infrastructure changes.

Vulnerabilities

This section addresses security vulnerabilities discovered within the system. These issues are directly exploitable in some manner, and ISE's analysis considers the complexity of performing a given exploit, along with the impact if an attack were successful.

Ravencoin

Use of Out-of-Date Bitcoin Fork

ISE-RAVEN-2020-01

LOW

Attack Requirements	Exploitation of discovered vulnerabilities in shared Bitcoin/Raven code.
Affected Assets	All.
Impact	Impact depends on vulnerabilities identified in out-of-date code.

Ravencoin is a mid-2017 fork of Bitcoin 0.15.x. Bitcoin is currently on version 0.20.1 (August 01, 2020). The table below lists possible known weaknesses or unintended functionality that may be present and adversely impact Ravencoin users (Table 2)

CVE	Status
CVE-2017-18350	Open
CVE-2018-17144	Open
CVE-2018-17145	Open
CVE-2018-20586	Open
CVE-2018-20587	Open
CVE-2019-15947	Open
CVE-2019-16761	Open
CVE-2019-16762	Open

Table 2. CVEs discovered since Ravencoin's fork.

Resolution: Merge Upstream Bitcoin Core Changes

ISE recommends that Ravencoin evaluate and merge any potential security related code changes from the upstream codebase. Many changes are routinely incorporated into Bitcoin core and features that may enhance the security and stability of core functionality should be evaluated and merged in.

Resolution Status

As of revision 1, this is a newly discovered issue.

History

- Rev. 1: issue added.

Faulty Conditional Logic

ISE-RAVEN-2020-02

INFO

Attack Requirements	Insider accidental re-use of faulty code.
Affected Assets	Raven assets.
Impact	May lead to undesirable behavior when processing Raven asset transactions.

On line 4520 of 'src/assets/assets.cpp#L4520[1]' there is a check to ensure that the value of the 'flag' variable must be either 0 or 1. However, this expression is faulty and will always evaluate to true, indicating that the 'flag' value is not 0 or 1 even in cases where it is. The affected code is shown below:

```
if (flag != 0 || flag != 1) {
    strError = _("Flag must be 1 or 0");
    return false;
}
```

This fault is likely due to the use of the || operator and not the intended && operator. Correct usage of this expression is present on 'src/assets/assets.cpp#L4939[2]' and shown below:

```
if (flag != 0 && flag != 1) {
    strError = "bad-txns-null-data-flag-must-be-0-or-1";
    return false;
}
```

Resolution: Modify Conditional Expression

Currently this does not adversely impact any workflows, but should be corrected incase the code gets reused in other more meaningful parts of the project with the assumption that it works as intended.

Resolution Status

As of revision 1, this is a newly discovered issue.

[1] https://github.com/RavenProject/Ravencoin/blob/master/src/assets/assets.cpp#L4520

[2] https://github.com/RavenProject/Ravencoin/blob/master/src/assets/assets.cpp#L4939

History

- Rev. 1: issue added.

Strategic Weaknesses

This section addresses high-level weaknesses in the design and development of Ravencoin. These weaknesses may not be directly exploitable but are the catalysts that lead to security problems over time when maintaining a complex code base over many years. Weaknesses are more often costlier to address, but at the strategic level they can allow vulnerabilities to go undetected or become easily detected by outside adversaries.

Ravencoin

Use of Out-of-Date Qt Framework

ISE-RAVEN-2020-03

LOW

Attack Requirements	Exploitation of discovered vulnerabilities in Qt framework.
Affected Assets	All.
Impact	Vulnerabilities could be leveraged for a wide range of attacks.

Ravencoin uses Qt version 5.7.1 which was released on June 16, 2016 and was supported until June 16, 2017. Bug fixes and support is available for versions that fall outside of the support window with Qt extended support.

Resolution: Update Qt

ISE recommends that Qt be updated to a version still within the support window. Newer versions of Qt also include updates to the Chromium web engine, of which 5.7.1 uses Chromium 49 whereas the most recent version of Qt (5.15) uses Chromium 80.

Resolution Status

As of revision 1, this is a newly discovered issue.

History

- Rev. 1: issue added.

Windows Signed Installer uses Untrusted Certificate

ISE-RAVEN-2020-04

LOW

Attack Requirements	Exploitation of distribution channels/Social engineering.
Affected Assets	All.
Impact	Malicious binaries may be more effectively distributed.

The signed Windows setup binary, the current version as of this report being ‘raven-4.3.2.1-win64-setup-signed.exe’ is digitally signed, however it appears to be signed by a certificate meant for Apple devices as the certificate issuer is “*Apple Worldwide Developer Relations Certification Authority*”. The required CA is not in the trusted root CA of Windows and therefore the signature is not trusted, as illustrated below when attempting to run the ‘raven-4.3.2.1-win64-setup-signed.exe’ setup executable:

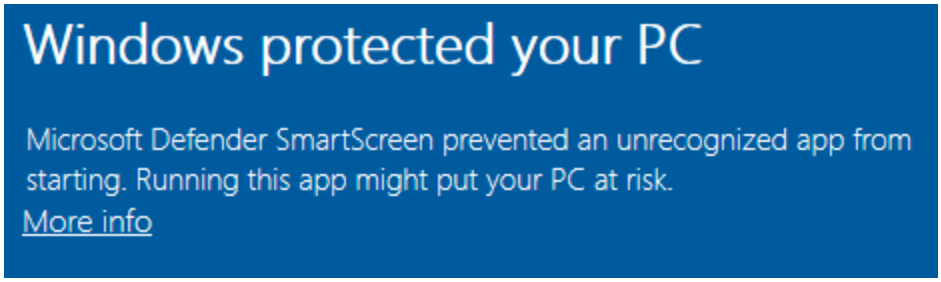

Figure 1. Microsoft Defender blocking the execution of a signed raven install package due to digital signing misconfiguration.

This forces the user to clickthrough dialogs that are meant to deter users from installing 'untrusted' apps, which may train users to clickthrough legitimate warnings and make them more apt to execute malicious binaries.

Resolution: Sign Windows Binaries using Authenticode Code Signing Certificates

ISE recommends that Authenticode code signing certificates be used to sign any Windows binaries meant for distribution. More information on Authenticode can be found in the below link:

- https://docs.microsoft.com/en-us/windows-hardware/drivers/install/authenticode

Resolution Status

As of revision 1, this is a newly discovered issue.

History

- Rev. 1: issue added.

Lack of Signed Binaries

ISE-RAVEN-2020-05

INFO

Attack Requirements	Exploitation of distribution channels/Social engineering.
Affected Assets	All.
Impact	Malicious binaries may be more effectively distributed. SmartScreen compatibility issues.

Compiled binaries for the Windows operating system come as a signed installer or zip package. The binaries installed using either method result in the unpacking/installation of 'raven-cli.exe', 'ravend.exe', 'raven-qt.exe', and 'libravenconsensus-0.dll' binaries which are not digitally signed. As an example, performing the 'Get-AuthenticodeSignature' Windows PowerShell command on 'raven-qt.exe' is shown below:

```
PS C:\Raven\network\bin> Get-AuthenticodeSignature .\raven-qt.exe

    Directory: C:\Raven\network\bin

SignerCertificate                         Status                                 Path
-----------------                         ------                                 ----
                                          NotSigned                              raven-qt.exe
```

Figure 2. Getting the digital signature details on raven-qt.exe.

This forces the user to clickthrough dialogs that are meant to deter users from installing 'untrusted' apps. However, the resulting installed binaries lack digital signatures as shown in the below table:

Windows Binaries Lacking Authenticode Signatures	Status
raven-cli.exe	Not Signed
ravend.exe	Not Signed
raven-qt.exe	Not Signed
libravenconsensus-0.dll	Not Signed

Table 3. Raven binaries lacking Authenticode signatures.

Resolution: Sign Binaries (OS Specific)

Ravencoin should use operating system specific methods of digitally signing both the wallet binaries themselves and the installer files. On Windows, binaries are signed using code signing certificates issued by the existing certificate authority infrastructure using the Windows SDK. On MacOS, developer certificates are issued by Apple directly. On Linux and other Unix-like operating systems, there is no first-party code signing functionality.

Windows Specific Note: Lack of code signing on Windows platforms may lead to Windows Defender/SmartScreen compatibility issues such as having binaries flagged as malware or malicious cryptocurrency miners. SmartScreen works by building reputation of a binary based on a number of factors, such as the number of active installations of a particular binary. Signing Windows binaries may increase reputation of binaries released and signed by a specific entity. To establish reputation with SmartScreen more quickly it is possible to get an Extended Validation code signing certificate. More information about EV certificates and SmartScreen can be found in the below link:

- https://docs.microsoft.com/en-us/archive/blogs/ie/microsoft-smartscreen-extended-validation-ev-code-signing-certificates

Resolution: Sign Binaries (PGP)

In addition to using OS-specific code signing, Ravencoin should also distribute GPG signature files for each binary package. This allows users to manually verify binaries even when their OS does not support signature verification and protects against potential compromised or malicious certificate authorities. GPG keys should be generated and stored securely, possibly using a hardware security device such as a smart card or Yubikey. To guard against malicious developers, Ravencoin may wish to implement a multi-key protocol that protects the master signing key; such a workflow is beyond the scope of this report.

Resolution Status

As of revision 1, this is a newly discovered issue.

History

- Rev. 1: issue added.

Additional Recommendations

ISE focused on assessing the security of Ravencoin components that deal with the token itself and the asset layer, and recommends the following to further improve Ravencoin's security posture:

- Updated status of Ravencoin through recurring reassessments to further enhance its security and verify mitigations put in place by Ravencoin.
- Source code review at a regular cadence.
- Review of mobile applications.
- Review of development workflows.
- Continued review of third-party libraries in use.

About ISE

ISE is an independent security firm headquartered in Baltimore, Maryland. We are dedicated to providing clients proven scientific strategies for defense. On every assessment our team of analysts and developers use adversary-centric approaches to protect digital assets, harden existing technologies, secure infrastructures, and work with development teams to improve our clients' overall security.

Assessing the client through the eyes of potential attackers allows us to understand possible threats and how to protect against those attacks. Our security analysts are experienced in information technology and product development which allows them to understand the security problems the client faces at every level. In addition, we conduct independent security research which allows us to stay at the forefront of the ever-changing world of information security.

Attacks on information systems cannot be stopped, however with robust security services provided by an experienced team, the effects of these attacks can often be mitigated or even prevented. We appreciate the confidence placed in us as a trusted security advisor. Please don't hesitate to get in touch for additional assistance with your security needs.

Independent Security Evaluators, LLC

4901 Springarden Drive
Suite 200
Baltimore, MD 21209
(443) 270-2296

contact@ise.io

https://www.ise.io/