Summary-Report Standard Notes Auth, Backend, API & Server 08.-09.2022

Cure53, Dr.-Ing. M. Heiderich, Dipl.-Ing. D. Gstir, R. Weinberger, Dr. A. Pirker

Index

Scope

- **Penetration tests and security assessments against Standard Notes server and backend**
    - **WP1**: White-box pentests and assessments against Standard Notes API gateway
        - **Sources:**
            - Sources were shared with the testing team.
        - **Environment:**
            - Access to a testing environment and three user accounts were provided for dynamic tests of this working package
    - **WP2**: White-box pentests and assessments against Standard Notes authentication
        - **Sources:**
            - Sources were shared with the testing team.
        - **Environment:**
            - Access to a testing environment and three user accounts were provided for dynamic tests of this working package
    - **WP3**: White-box pentests and assessments against Standard Notes files service
        - **Sources:**
            - Sources were shared with the testing team.
        - **Environment:**
            - Access to a testing environment and three user accounts were provided for dynamic tests of this working package
    - **WP4**: White-box pentests and assessments against Standard Notes syncing server
        - **Sources:**
            - Sources were shared with the testing team.
        - **Environment:**
            - Access to a testing environment and three user accounts were provided for dynamic tests of this working package
    - **Source code:**
        - https://github.com/standardnotes/server/
            - Branch: main
            - Commit ID: b7f7c3f164a1472d2356d9680c76718259dc85b7
        - https://github.com/standardnotes/app/
            - Branch: main
            - Commit ID: 26fe530deb6ddf5e5a12ea60ea7a190881bc411c
    - **Test environment:**
        - **URL:**
            - https://app.standardnotes.com/
    - **Test accounts were provided and created by Cure53**
    - **Test-supporting material was shared with Cure53**
    - **All relevant sources were shared with Cure53**

Identified Vulnerabilities

The following section lists all vulnerabilities and implementation issues identified throughout the testing period. Please note that findings are listed in chronological order rather than by their degree of severity and impact. The aforementioned severity rank is given in brackets following the title heading for each vulnerability. Furthermore, each vulnerability is given a unique identifier (e.g., *SN-02-001*) to facilitate any future follow-up correspondence.

SN-02-002 WP2: Lack of password-alteration session invalidation *(Low)*

***Note**: Client notes that the ability to revoke other sessions is present as a separate user action. While there is no intention to alter the behavior of password changes to invalidate all sessions, client will bring both actions closer together in the UI to allow user to decide if total session revocation is their intention.*

SN-02-003 WP2: Username enumeration upon registration *(Low)*

***Note**: Client notes registration is a mutating action that notifies the end user, and is thus not an ideal form of account enumeration. In addition, response obscurity is the only possible solution to this, but if the responses are published in the open-source codebase, there is not much to be gained.*

SN-02-006 WP3: Directory traversal in FS file service *(**Critical**)*

***Note**: Client notes that the FS file service is an internal feature (primarily built for the automated testing suite) not available to production users.*

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github.*

SN-02-007 WP3: DoS for FS file service *(Medium)*

***Note**: Client notes that the FS file service is an internal feature (primarily built for the automated testing suite) not available to production users.*

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github.*

SN-02-008 WP3: S3 bucket flooding for S3 file service *(Medium)*

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github, as well as the described changes to the related infrastructure.*

SN-02-010 WP4: Lack of *sync*-server upper bound may facilitate DoS (*High*)

***Note**: Client notes that payload persistence limits are dictated on the database level and infrastructure level, and not just the app level, so the impact of this issue is highly mitigated by preceding layers.*

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github.*

Miscellaneous Issues

This section covers any and all noteworthy findings that did not lead to an exploit but might assist an attacker in successfully achieving malicious objectives in the future. Most of these results are vulnerable code snippets that did not provide an easy way to be called. Conclusively, while a vulnerability is present, an exploit might not always be possible.

SN-02-001 WP1-4: Auth token leakage via URL (*Low*)

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diffs on Github.*

SN-02-004 WP2: *bcrypt* usage for password hashing (*Info*)

***Note**: This issue is considered a false positive, since the client application already applies Argon2 (or PBKDF2, depending on the version) to user passwords for server-relief[1] purposes such as reducing server-side memory and computational requirements.*

SN-02-005 WP2: Lack of email verification upon account creation (*Info*)

***Note**: Client notes that this is behavior is intended, and serves as a privacy-preserving mechanism as users can register using non-email strings, such as a unique username.*

SN-02-009 WP2: Lack of *Valet-Token* operation check in file service (*Info*)

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github.*

SN-02-011 WP2: Lack of *mute* endpoint authentication (*Info*)

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diffs on Github.*

[1] https://libsodium.gitbook.io/doc/password_hashing#server-relief

SN-02-012 WP2: Unbounded subscription sharing by cascading invites *(Low)*

***Note**: This issue is currently not seen as a security flaw. Both Cure53 and the maintainer agree that the severity is minimal and the issues can hence be ignored.*

SN-02-013 WP1: Subscription token leakage via URL *(Low)*

***Note**: Client notes that these tokens are single-use and ephemeral, and any potential impact by server-side logging is mitigated by their auto-expiring nature.*

SN-02-014 WP3: Lack of antivirus scan for uploaded files *(Info)*

***Note**: This issue is currently not seen as a security flaw and there are no plans for fixes on the current release schedule. Both Cure53 and the maintainer agree that the severity is minimal and the issues can hence be ignored.*

SN-02-015 WP2: Continued support for outdated authentication methods *(Low)*

***Note**: This issue was addressed by the Standard Notes team and the fix was verified by Cure53 by inspecting the respecting PR/Diff on Github.*

SN-02-016 WP1: Insufficient HTTP security header configuration *(Low)*

***Note**: Client notes that the current security header configuration for CSP and X-Frame-Options are currently at their maximal optimized point, particularly relating to the fact that unsafe-inline and unsafe-eval in script-src are required in order for usage of WebAssembly based cryptographic libraries in the web application.*