Fine penetration tests for fine websites

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Audit-Report ente Crypto Design & Code 02.-03.2023

Cure53, Dr.-Ing. M. Heiderich, Dr. N. Kobeissi

Index

Introduction

“ente is a cloud based mobile and desktop photo storage app with a focus on security and privacy.”

From https://ente.io/

This report details all phases of a comprehensive assessment initiated by Cure53 against the ente secure photo backup and sync service, including an *Executive Summary*, *Cryptography Review Methodology*, all identified findings, and conclusory observations. The test team placed particular scrutiny on the cryptographic implementation across multiple components, including the web and mobile applications, backend infrastructure, and overall system architecture.

The audit was requested by Ente Technologies, Inc. in November 2022 and carried out by two senior members of the Cure53 team across nine days in March 2023 (between CW10 and CW11).

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

For widespread coverage and trouble-free execution, preparations were completed in the week prior to the active review phase (CW09) and the work packages (WPs) were divided into three distinct groups, as follows:

- **WP1**: Crypto & design reviews of ente's general crypto architecture & design
- **WP2**: Code & crypto reviews of crypto implementation in ente photos-app
- **WP3**: Code & crypto reviews of crypto implementation in ente photos-web

The ente team provided sources, an overview of the applied design and architecture, detailed testing documentation and information, and any other pertinent items required for the testing team to achieve maximum audit efficacy. This particular engagement adhered to a white-box methodology.

Communications were facilitated via a dedicated and shared Discord server, which ensured effective discussions on the whole. All involved employees from the ente and Cure53 team were invited to join and partake. The review phase was not delayed by any notable blockers, attesting to the productive and transparent scope preparation. Live reporting was offered and implemented via the aforementioned Discord server. To accompany this, Cure53 relayed a number of progress updates to the ente team.

A total of four findings were observed and documented by the Cure53 team following strong coverage across all work packages. Notably, all four were deemed to exhibit exploitation potential and as such were assigned to the *Identified Vulnerabilities* section. Cure53 considers this yield to be relatively small in general. This, coupled with the complete lack of any *Critical* severity issues, reflects favorably on the resilience of the components in scope.

Nevertheless, the ente team should prioritize addressing the *High* impact finding at the earliest possible convenience, which facilitates weak password acceptance in the web application (see ENT-01-001). Incorporating the mitigation guidance offered in this ticket will help to restrict the potential attack surface. All in all, Cure53 is pleased to confirm that the ente team has made commendable progress in its pursuit of an excellent security foundation for all cryptography implementations. Nonetheless, the opportunities for security growth should be heeded to negate any potential threats and ensure the highest possible degree of user protection.

The report will now provide information regarding the scope, test setup, and available materials. This is followed by an *Executive Summary*, *Cryptography Review Methodology*, and a list of all issues encountered during this engagement.
These are offered in chronological order of detection and attach a technical description, PoC if necessary, and the ideal mitigation or fix procedure for each specific context.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

To finalize, Cure53 elaborates on the general impressions gained throughout this test in the *Conclusions* section. Here, a definitive overview of the ente web and mobile application's cryptography design and architecture, plus suggested follow-up actions, are offered.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Scope

- **Source code audits & cryptography reviews against ente cryptography design & code**
  - **WP1**: Crypto & design reviews of ente's general crypto architecture & design
    - **Design & architecture**
      - https://ente.io/architecture
    - **Relevant models:**
      - Key attributes available to server
      - Key attributes available on client
  - **WP2**: Code & crypto reviews of crypto implementation in ente photos-app
    - **In-scope application (crypto only):**
      - https://github.com/ente-io/photos-app
    - **Additional information:**
      - Key generation for a fresh user
      - Key derivation for a returning user
      - Key recovery
      - File encryption
      - Thumbnail / Metadata encryption
      - Collection (album) sharing
      - Verification ID generation
      - Password updation
  - **WP3**: Code & crypto reviews of crypto implementation in ente photos-web
    - **In-scope application (crypto only)**
      - https://github.com/ente-io/photos-web
    - **Primary focus areas / key privacy claims:**
      - "ente is a cloud storage provider that provides end-to-end encryption for your data. These keys are available only to you. Meaning only you can access your data elsewhere."
      - "Since only you know your password, only you can derive your *keyEncryptionKey*."
      - "Since only you can derive your *keyEncryptionKey*, only you have access to your *masterKey*."
  - **Test-supporting material was shared with Cure53**
  - **All relevant sources were shared with Cure53**

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Executive Summary

This cryptographic report documents a comprehensive assessment of the ente secure photo backup and sync service, focusing on its cryptographic implementation across multiple components, including web and mobile applications, backend infrastructure, and system architecture. Sufficient safeguarding of sensitive user data - such as photos, videos, and personal information - represents the primary objective behind ente's cryptographic codebase composition. This stipulation is fulfilled by employing robust end-to-end encryption mechanisms, secure password handling, and other cryptographic best practices.

The audit evaluated various aspects of ente's cryptographic codebase, with targeted deep-dive examinations against the security of the encryption and decryption processes, key management, secure password handling, and sharing capabilities of the service. This involved an extensive review of the application's cryptographic library usage, such as *libsodium* and Argon2, to ensure optimal implementation and adherence to first-rate industry standards. Furthermore, the test team sought to analyze the application's overall key management strategy - including generation, storage, and rotation mechanisms - as well as the resilience of its password handling processes, which focused on password strength enforcement and recovery from compromise.

- **Encryption:** ente employs end-to-end encryption, primarily constructed upon the *libsodium* cryptographic framework and Argon2 password hashing primitive.
- **Key management:** ente implements a sophisticated key management system, utilizing asymmetric encryption with public and private keys to protect data sharing and access.
- **Authentication:** ente leverages a robust multi-factor authentication (MFA) mechanism to authenticate users, providing an additional layer of security against unauthorized access.
- **Secure data transmission:** ente ensures secure data transmission by employing Transport Layer Security (TLS) encryption for all data transfers between client devices and ente servers.
- **Code quality and security practices:** ente's source code was deemed soundly composed and documented, with stringent adherence to best practices.

This cryptographic evaluation serves to provide a comprehensive overview of the security posture exhibited by ente’s secure photo backup and sync service, with particular scrutiny against the application's cryptographic design and implementation. ente is committed to providing a secure platform for users to store, manage, and share their multimedia content. As such, end-to-end encryption and user privacy is deemed of utmost importance.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Cure53's assessment identified a number of areas whereby the current implementation would benefit from hardening to enhance the application's security and instill comprehensive user-data protection. In this report, the four primary issues discovered during the audit are outlined, including weaknesses concerning password handling, key management, recovery from compromise, and sharing functionality. Cure53 provides detailed technical descriptions of these behaviors and extrapolates the optimum mitigation guidance for each to ensure the ente platform's resilience and user trust model remains indisputable.

The evaluation also honed in on ente's secure sharing functionality, which allows users to share encrypted photo albums with third parties via web links whilst maintaining content confidentiality and privacy. The audit scrutinized the process of granting and revoking sharing permissions, specifically examining the implications of unrotated encryption keys against shared album protection.

To summarize, this rigorous analysis raised the presence of several issues of varying severity within ente's cryptographic implementation, which have been documented and addressed in this report. The ente team should review the advanced insights and recommendations offered throughout to help elevate the application's security posture to an exemplary standard. Integrating the guidance for future roll-outs will undoubtedly ensure impermeable protection of user data and maintain user trust in the platform's ability to securely store and synchronize highly personal data.

Cryptography Review Methodology

The following passages specify Cure53's examination of the security posture exhibited by the cryptographic components and processes within the ente web and mobile applications, based on shared architecture specifications. The methodology consists of five steps, as follows:

- **Identification:** The review was initiated by studying ente's cryptographic architecture documents for the web and mobile applications. This granted the test team a transparent understanding of the application's cryptographic design, key management, and encryption processes. The team then identified potential vulnerabilities and implementation issues within the applications, documenting each finding with a unique identifier and severity rank.
- **Analysis:** Following the enumeration of the findings, the team conducted a thorough investigation of the affected components, files, and code within the web and mobile applications. This involved examining the application's source code,

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

configuration files, and encryption algorithms to determine the root cause of each vulnerability and understand its implications on the application's security.

- **Evaluation:** In the evaluation stage, Cure53 assessed the overall impact of the identified vulnerabilities on the confidentiality, integrity, and availability of ente's web and mobile applications. By evaluating the potential consequences of each, the team prioritized the issues based on the perceived severity impact, with due consideration given to those deemed most critical.
- **Reporting:** The tickets from the identification, analysis, and evaluation stages were compiled into a comprehensive report. This offers an exhaustive outline of the weaknesses persisted within ente's web and mobile applications, sharing information regarding the catalyst for each finding and any potential consequences. Subsequently, the report was shared with the ente development team to facilitate the remediation process.
- **Recommendation:** The final stage of the *Cryptography Review Methodology* serves to impart all recommended guidance for mitigating the identified vulnerabilities, with a view to upgrading the overall security posture of ente's web and mobile applications. Specifically, the key areas of weakness pertain to implementing robust password policies, securely updating encryption keys upon password alteration, and rotating encryption keys when sharing permissions are amended or revoked.

The cryptography review team’s strict adherence to this methodology guaranteed that ente's web and mobile applications received meticulous audit scrutiny. This, in turn, raised a number of opportunities for security growth. Cure53 strongly advises that the ente team allocates ample time and resources for appropriate follow-up actions. Resolving all findings documented in this report will help to strengthen application security, safeguard user data and privacy, and minimize the risk of unauthorized access or exploitation.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Identified Vulnerabilities

This section discusses all vulnerabilities and implementation issues detected by Cure53 during this engagement. Findings are cited in order of identification rather than by degree of severity and are linearly structured with a unique identifier for reference purposes (e.g., *ENT-01-001*), followed by the title heading, and ending with the severity marker. The latter is offered in brackets, e.g. *(High)*.

ENT-01-001 WP3: Null passwords accepted in web application *(High)*

***Note**: Whilst this issue was discovered on the ente web application's production deployment during the review, the ente team had already implemented appropriate codebase amendments prior to documentation in this report.*[1]

Testing confirmed that the ente web application does not implement any security checks for password strength. Passwords are used to derive the root key material for all ente symmetric encryption keys, and thus, their strength is integral to the confidentiality and integrity guarantees provided by the application's end-to-end encryption design.

Permitting users to set weak passwords, such as *1*, *password*, or *iloveyou*, significantly undermines the safeguard measures applied for the ente web application. Weak passwords are generally susceptible to guessing, cracking, and enumeration via various techniques such as brute forcing, dictionary attacks, or social engineering. Consequently, this exposes user accounts to unauthorized access, potentially compromising the privacy and integrity of stored photos, videos, and personal information. Furthermore, users are often found to reuse passwords across multiple services. As such, weak password adoption in one application may facilitate a detrimental cascading effect against other platforms in addition.

Affected file:
src/components/SignUp.tsx

Affected code:
Cure53 is unable to cite the affected code, since the vulnerability originates from a lack of code rather than insecurely-implemented existing code.

To mitigate this issue, Cure53 advises implementing a robust password policy that enforces the use of sufficiently strong passwords, which can be achieved by employing the *zxcvbn* password strength estimation library[2], for instance.

[1] https://github.com/ente-io/photos-web/pull/960
[2] https://github.com/dropbox/zxcvbn

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Additionally, the ente team could provide users with password strength indicators during the account creation or password alteration processes to encourage adoption of adequate password complexity.

ENT-01-002 WP1: Cryptographic recovery from compromise impossible *(Medium)*

Note: *Similarly to ENT-01-004, the ente team recognized this issue as a known limitation in ente's current design and plans to address it in a future roadmap update.*

The observation was made that the user's *masterKey* and *collectionKeys* do not rotate when a user updates their ente password.

Retention of the same encryption keys - even after a password change in the ente application - represents a significant security vulnerability. In the event of a password compromise, a malicious actor may gain unauthorized access to the user's sensitive data, including multimedia content, personal information, or even confidential documents. Ideally, when a password is altered, the corresponding encryption keys should also be updated to ensure data integrity and confidentiality. However, if the keys remain the same, the user will not be able to recover from a password compromise, rendering their data susceptible to unauthorized access and potential misuse.

This flawed approach undermines the core principles of data management and user trust, since the inability to recover from a password compromise raises a false sense of security. The persistence of unchanged encryption keys may inadvertently expose users to persistent security risks, even after they have taken corrective measures to update their passwords. As a result, the ente team should prioritize resolving this vulnerability by implementing a robust key management system that includes securely updating encryption keys upon password alteration. This will not only improve the overall security posture of the application but also bolster user confidence in the platform's ability to protect their digital memories and sensitive information.

To mitigate this issue, Cure53 recommends offering users the ability to rotate high-level keys, such as the *masterKey* and *collectionKeys*, after a password change. Notably, this may involve re-encryption, which can be expensive for large photo libraries. As such, an effective solution could be to adopt the new keys only for future collections, images, and files that are uploaded after the password amendment.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

ENT-01-003 WP1: Encrypted *masterKey* obtainable via email compromise *(Low)*

The testing team noted that the ente service allows access to a user's encrypted *masterKey* via a user's email address. Whilst the protection of the user's encrypted *masterKey* through mandated email-based authentication was considered a reasonable mechanism, it is possible to tie divulgation of the user's encrypted *masterKey* to the user's password, such that:

- A compromise of the user's email does not entail a compromise of their encrypted *masterKey*,
- A user may use their password to authenticate themselves and obtain a copy of their encrypted *masterKey* from the ente service, without revealing information regarding their password.

The proposed improvement would retain email-based authentication whilst integrating supplementary defense-in-depth, enforcing that the ente server can only send the authentication email after knowledge of the user password is proved by the user.

Currently, the ente service utilizes the user's password to generate the *masterKey's* encryption key, *keyEncryptionKey*, via the following (simplified) mechanism:

```
password -> Argon2 -> keyEncryptionKey
```

Cure53 suggests two potential improvements, in addition to email-based authentication prior to revealing the encrypted *masterKey*:

- An easier to implement, but less secure, HKDF-based method.
- A difficult to implement, though much more secure and backwards compatible method relying on a Password-Based Key Exchange, such as SRP or OPAQUE.

HMAC-based Key Derivation Function (HKDF)[3] is a widely-adopted key derivation function (KDF) based on the Hash-based Message Authentication Code (HMAC) construction. HKDF is designed to extract and expand cryptographically secure keying material from a given input keying material (IKM) and is often used for key derivation in various cryptographic protocols.

In light of this, HKDF could be adopted for the purpose of deriving a *loginKey*. This *loginKey* could be provided to the ente server as a user authentication token without granting the ente server knowledge of the *keyEncryptionKey* or the password:

[3] https://www.rfc-editor.org/rfc/rfc5869

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

```
password -> Argon2 -> HKDF -> keyEncryptionKey
                        |
                        └---> loginKey
```

As such, access to the encrypted *masterKey* would depend on the *loginKey* and subsequently strict password knowledge, rather than access to the user's email. The user would provide the *loginKey* to the ente server via TLS and the ente server would match it with a hash of the *loginKey* stored server-side, before revealing to the user the encrypted *masterKey*.

HKDF initiates several input parameters to perform the necessary extraction and expansion processes. These input parameters include:

- **Input Keying Material (IKM)**, which constitutes the initial secret or raw key material that may originate from a shared secret, password, or key exchange. In this example context, this would represent the output of the Argon2 password hashing function.
- **A salt,** which constitutes an optional non-secret random value that can be used to enhance the security of the extraction process, particularly in cases whereby the input keying material offers low entropy or may be reused across multiple instances. In an example context, this could represent either a string unique to each ente user, or the ente user's unique user ID/username.
- **The desired output length**, which specifies the number of bytes to be generated by the HKDF function. In this example, this would constitute 512 bytes, or suffice for one 256-bit encryption key and one 256-bit login token (*keyEncryptionKey*, and *loginKey*).
- **An optional *Info* parameter**, which represents a context-specific and application-specific value that helps to bind the derived keys to a specific context, preventing cross-protocol or cross-application attacks. In Cure53's example, this would constitute a string specifying the keys' purpose, such as *EnteUserLogin*.

The aforementioned approach may, however, incur the risk of replay attacks. A replay attack is a threat vector in which an attacker intercepts and retransmits a legitimate message or data transmission to gain unauthorized access, impersonate a user, or disrupt a communication system. This attack exploits either authentication-specific weaknesses or a lack of adequate timestamping and nonce mechanisms, thereby allowing the attacker to reuse previously-captured messages and deceive the target system or user. Since ente utilizes TLS 1.3, replay attacks are prevented on the transport layer.

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

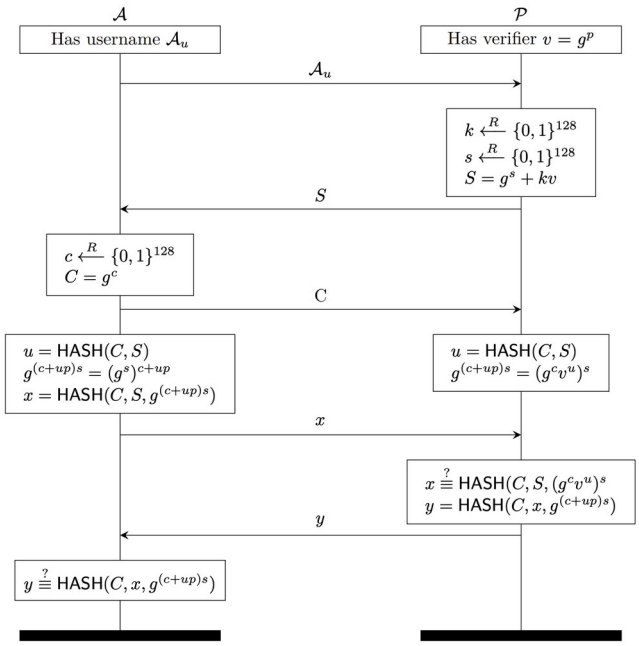

Fig.: High-level overview of SRP.

Alternatively, a design based on SRP (the Secure Remote Password protocol) may offer greater benefits here. SRP is a cryptographic authentication protocol that allows two parties to establish a shared secret key over an insecure network. The SRP protocol is designed to protect against a multitude of compromise scenarios, including replay attacks, Man-in-the-Middle (MitM) attacks, and dictionary attacks. The protocol uses a combination of cryptographic hash functions, modular arithmetic, and symmetric key cryptography to securely authenticate the parties involved. The SRP protocol is particularly useful for web applications because it provides a method to authenticate users without transmitting passwords over the network.

To integrate SRP into a web application, the server and client must implement the protocol. The client sends its username to the server, then the server responds with a random value, entitled a *salt*. The client then calculates a value (the *verifier*) using its password and the salt. The client sends the verifier to the server and the server uses it to calculate a value, namely the *server session key*. The server sends the server session key to the client, which uses it to calculate its own session key. If the two session keys match, the client is authenticated. This process is performed without transmitting the password over the network, thus providing zero-knowledge authentication.

Newer alternatives to SRP, such as OPAQUE, are also available. SRP and OPAQUE are both cryptographic authentication protocols that provide zero-knowledge password

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

verification. However, Cure53 recommends implementing OPAQUE over SRP - and indeed over all other aforementioned guidance - because it offers greater efficiency, stronger security proofing, and easier implementation in comparison[4].

ENT-01-004 WP1: Share revocation still permits third-party decryption *(Medium)*

Note: *The ente team is already aware of this issue, as documented in public-facing discussions on the ente roadmap website.*[5]

ente offers a feature for sharing end-to-end encrypted photo albums with third parties via a web link. Albums are encrypted with a *collectionKey*, which is also disclosed during the sharing process. However, Cure53 noted that the *collectionKey* for said album remains the same and is not rotated after sharing permissions are revoked, rendering all previously shared content accessible to anyone in possession of the original *collectionKey*.

The absence of an encryption key rotation mechanism for shared photo albums in the ente application can lead to potential security risks. When sharing permissions are revoked, the album's *collectionKey* remains the same, which means that anyone who held access to the shared album prior to the revocation may still retain the ability to decrypt and access the album's content. This lack of key rotation undermines the application's ability to maintain the privacy and confidentiality of user data, since it fails to fully restrict access to the album once sharing permissions have been revoked. Furthermore, this behavior may cause users to inadvertently expose their content to unintended recipients, which evidently may incur grave implications, particularly when dealing with sensitive information or private memories.

To mitigate this issue, ente should consider implementing a key rotation mechanism that updates the *collectionKey* when sharing permissions are changed or revoked. By generating a new encryption key and re-encrypting the shared album with the updated key, ente can ensure that only authorized users retain access to the shared content. Additionally, employing a robust key management system that tracks and manages the distribution of updated keys to authorized users will help prevent unauthorized access to the shared albums. This approach will not only bolster the application’s security offering in general but also enhance user trust by providing superior control over the privacy and access of their shared content. Alternatively, in order to prevent re-encryption - which may prove costly for large albums - ente could enforce the new *collectionKey* only for photos uploaded into the album after share access has been revoked, thereby assuming the optimal middle-ground between security and performance.

[4] https://datatracker.ietf.org/doc/html/draft-krawczyk-cfrg-opaque-00
[5] https://roadmap.ente.io/option-to-download-re-encrypt-and-re-upload-an-unshared-album-p-2652/

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

Conclusions

In this cryptographic audit, the ente secure photo backup and sync service was subjected to an in-depth review process, with due scrutiny against the provided architecture documents, web application codebase, and mobile application codebase.

This comprehensive assessment aimed to evaluate the effectiveness and defense integration of the cryptographic implementation in all key areas of the platform, such as password handling, key management, recovery from compromise, and sharing functionality. By scrutinizing the design and implementation of cryptographic primitives and protocols across a host of ente characteristics, Cure53 was able to locate a selection of potential security vulnerabilities and provide connected guidance to enhance platform security.

During the cryptographic audit, the team observed that ente's usage of *libsodium* and Argon2 was correctly implemented in the web and mobile application codebases. *Libsodium* is a modern, easy-to-use software library for encryption, decryption, signatures, password hashing, and other functionality. By leveraging the first-rate cryptographic primitives provided by *libsodium*, ente ensures that its encryption and decryption processes are secure and adhere to best practices in the field of cryptography. Similarly, the valid incorporation of the Argon2 password hashing algorithm provides an auxiliary protective measure by securely storing and verifying user passwords.

The astute utilization of *libsodium* and Argon2 in ente's cryptographic implementation offers several key benefits. First and foremost, user data is safeguarded by employing well-vetted, industry-standard cryptographic algorithms that have undergone extensive analysis and scrutiny by the cryptographic community. This helps protect user data against a vast range of potential threats and attacks. Secondly, leveraging these established libraries and algorithms minimizes the likelihood of introducing vulnerabilities as a result of custom or improperly implemented cryptographic solutions. This, in turn, allows the ente team to focus on other aspects, such as user experience and functionality, while maintaining platform resilience. Overall, sound usage of *libsodium* and Argon2 contributes significantly to the reliability of ente's cryptographic design and implementation.

The focus on end-to-end encryption, user privacy, and access control mechanisms during the audit alleviated four primary issues, as outlined in this report. By analyzing the password policies, key derivation processes, authentication methods, and sharing features, the team was able to identify impactful flaws that could potentially undermine the confidentiality, integrity, and privacy of user data stored within ente. The

Dr.-Ing. Mario Heiderich, Cure53
Bielefelder Str. 14
D 10709 Berlin
cure53.de · mario@cure53.de

recommended advice purports to resolve any risk directly incurred, as well enhance the platform's threat defense, ensuring that ente continues to provide a durable and trustworthy environment for users to store, manage, and share their multimedia content.

In conclusion, this cryptographic audit of the ente secure photo backup and sync service has pinpointed several key areas for hardening improvement. By addressing these vulnerabilities, the ente team can enact due diligence concerning its commitment to providing a secure environment for users to store, manage, and share their multimedia content. The four primary issues discovered during the audit are summarized as follows:

- ENT-01-001: Weak password policies undermine the security of the application by rendering user account compromise easier to achieve.
- ENT-01-002: Retention of the same encryption keys after a password alteration increases user data susceptibility to unauthorized access in the event of a password breach.
- ENT-01-003: The current implementation of email-based authentication can be improved to prevent unauthorized access to a user's encrypted *masterKey*.
- ENT-01-004: The lack of key rotation when revoking sharing permissions may provoke unauthorized access to shared photo albums.

To finalize this report, Cure53 believes that ente can bolster its overall security posture, optimally protect user data, and maintain the privacy and confidentiality of user content by adhering to all guidance stipulated. Generally speaking, end-to-end encryption and user privacy should remain the key focal points for the ente team moving forward, since these are considered integral facts of any secure photo backup and sync service.

Cure53 would like to thank Vishnu Mohandas, Abhinav Kumar, and Neeraj Gupta from the Ente Technologies, Inc. team for their excellent project coordination, support, and assistance, both before and during this assignment.