FORM P1

REPUBLIC OF SOUTH AFRICA

PATENTS ACT, 1978

**APPLICATION FOR A PATENT AND ACKNOWLEDGEMENT OF RECEIPT**

**[Section 30 (1)-Regulation 22]**

The granting of a patent is hereby requested by the undermentioned applicant on the basis of the present application.

| Official Application No. | | | Applicant's or Agent's Reference |
|---|---|---|---|
| 21 | 01 | **2026/01434** | nfcbio |

| 71 | Full Name(s) of Applicant(s) |
|---|---|
| | Paul Armer<br>Parkada Farm, D176, Ballito, 4399, South Africa |

| 54 | Title of invention |
|---|---|
| | **A SYSTEM AND METHOD FOR BINDING BROWSER-BASED, HARDWARE-VERIFIED ZERO-KNOWLEDGE CRYPTOGRAPHIC PROCESSES FOR DECENTRALIZED DATA STORAGE** |

The applicant has claimed priority (if any) as set out in the abovementioned international application

~~This application is for a patent of addition to patent application No.~~

| 21 | 01 | |
|---|---|---|

~~This application is a fresh application in terms of section 37 and based on Application No.~~

| 21 | 01 | |
|---|---|---|

This application is accompanied by:

| | | |
|---|---|---|
| X | 1. | A single copy of a provisional specification of 8 pages. |
| | 2. | Drawings of _____ sheet(s). |
| | 3. | Publication particulars and Abstract(Form P8) |
| | 4. | A copy of Figure of the drawings (if any) for the abstract |
| | 5. | Assignment of invention |
| | 6. | Certified priority document(s) |
| | 7. | Translation(s) of the priority document(s) |
| | 8. | Assignment of priority rights |
| | 9. | A copy of the Form P.2 and the specification of S.A Patent Application (if applicable). |
| | 10. | A declaration and power of attorney on Form P3 |
| | 11. | Statement on the use of indigenous Biological Resource, Genetic Resource, Traditional Knowledge or Use on Form P26 |
| | 12. | Request(s) for ante-dating/delay of acceptance on Form P4 |
| | 13. | Request for classification on Form P4 |

| 74 | Address of Service: |
|---|---|
| | DEBRA RAY ARMER<br>Parkada Farm, D176, Ballito, 4399<br>SOUTH AFRICA |

Dated this 2nd day of February 2026

**Submitted online by : DEBRA RAY ARMER**

.....

Signature of Applicant(s)

This is returned to the applicant's
address for service as proof of lodging.

| RECEIVED |
|---|
| Official Date Stamp |
| .....<br>Registrar of Patents |

FORM P2

REPUBLIC OF SOUTH AFRICA **REGISTER OF PATENTS** PATENTS ACT, 1978

| Official application No. | Lodging date: Provisional | Acceptance date |
|---|---|---|
| 21 01 **2026/01434** | 22 2026/02/02 | 47 |
| International classification | Lodging date: Complete | Granted date |
| 51 | 23 | |

71 Full name(s) of applicant(s)/Patentee(s):

Paul Armer

| 71 Applicant substitued: | Date registrered |
|---|---|
| | |
| 71 Assignee(s): | Date registrered |
| | |

72 Full name(s) of inventor(s):

Paul Armer

| Priority claimed: | Country | Number | Date |
|---|---|---|---|
| | | | |

54 Title of invention

**A SYSTEM AND METHOD FOR BINDING BROWSER-BASED, HARDWARE-VERIFIED ZERO-KNOWLEDGE CRYPTOGRAPHIC PROCESSES FOR DECENTRALIZED DATA STORAGE**

Address of applicant(s)/patentee(s):

Parkada Farm, D176, Ballito, 4399
SOUTH AFRICA

74 Address for service

DEBRA RAY ARMER
Parkada Farm, D176, Ballito, 4399
SOUTH AFRICA
Reference No. nfcbio

| 61 Patent of addition No. | Date of any change |
|---|---|
| | |
| Fresh application based on. | Date of any change |
| | |

REPUBLIC OF SOUTH AFRICA
PATENTS ACT, 1978
**PROVISIONAL SPECIFICATION**
**[Section 30(1) – Regulation 28]**

| OFFICIAL APPLICATION NO. | | LODGING DATE | |
|---|---|---|---|
| 21 | 01 **2026/01434** | 22 | 2026/02/02 |

INTERNATIONAL CLASSIFICATION

| | |
|---|---|
| 51 | |

FULL NAME(S) OF APPLICANT(S)

| | |
|---|---|
| 71 | Paul Armer<br>Parkada Farm, D176, Ballito, 4399, South Africa |

FULL NAME(S) OF INVENTORS(S)

| | |
|---|---|
| 72 | 1. Paul Armer |

TITLE OF INVENTION

| | |
|---|---|
| 54 | **A SYSTEM AND METHOD FOR BINDING BROWSER-BASED, HARDWARE-VERIFIED ZERO-KNOWLEDGE CRYPTOGRAPHIC PROCESSES FOR DECENTRALIZED DATA STORAGE** |

Form P6

REPUBLIC OF SOUTH AFRICA

PATENTS ACT, 1978

# PROVISIONAL SPECIFICATION

(Section 30(l) - Regulation 27)

| Official Application No. | | |
|---|---|---|
| 21 | 01 | |

| Lodging Date |
|---|
| 22 |

| Full name(s) of applicant(s) |
|---|
| 71 |

| Full name(s) of inventors(s) |
|---|
| 72 |

| Title of invention |
|---|
| 54 |

(RP42E)

REPUBLIC OF SOUTH AFRICA

PATENTS ACT, 1978

PROVISIONAL SPECIFICATION (Form P6)

**TITLE OF THE INVENTION:**

A **System and Method** for **Binding** Browser-Based, Hardware-Verified Zero-Knowledge Cryptographic Processes for Decentralized Data Storage

**TECHNICAL FIELD:**

This invention relates to the field of secure digital data storage. More particularly, the invention pertains to a decentralized storage architecture employing client-side zero-knowledge encryption integrated with hardware-based identity verification.

**BACKGROUND OF THE INVENTION:**

- **The Problem:** In an era where the average cost of a data breach has surpassed $4 million, traditional cloud storage models—which rely on server-managed encryption keys—have become a critical liability for modern enterprises. As cyber threats and regulatory requirements like GDPR and HIPAA evolve, the only way to guarantee absolute data privacy is to ensure the service provider never sees or possesses the keys to the data they host. There remains a substantial reliance on software-only passwords. The era of password use in innovative solutions, is dead.
- **The Gap:** Current solutions (like standard AES-256 cloud storage) are insufficient for secure file storage as there remains reliance on the storage vendor to encrypt the data and files. He who owns and controls the encryption key, owns and controls the data and files. The User is purposely and permanently excluded from control of their own digital assets and own digital identity.

**SUMMARY OF THE INVENTION:**

The invention provides a system that ensures no identifying data or unencrypted file fragments are stored on server-side infrastructure.

- **Key Feature 1:** A method for generating and retaining a unique encryption key exclusively within the volatile memory of a client device during an active user session. The private key remains solely with the user.
- **Key Feature 2:** Integration of NFC SmartCards or biometric hardware as a mandatory factor for identity verification before data encryption or decryption.

- **Key Feature 3:** A decentralized distribution of encrypted data packets across multiple storage nodes.

**STATEMENT OF INVENTION**

**According to a first aspect of the invention**, there is provided a method for secure, decentralized data storage and retrieval, the method comprising the steps of:

- Interfacing a client-side browser with a hardware-verified identity medium to retrieve a primary cryptographic seed;
- Bifurcating said seed into a non-sensitive **Identity Component** and a sensitive **PassCode Component**;
- Transmitting the Identity Component to a remote server for user authentication while retaining the PassCode Component exclusively within the **volatile memory** of the client device; and
- Deriving a symmetric encryption key from said PassCode Component to perform client-side encryption or decryption of data packets prior to transmission or following receipt.

**According to a second aspect of the invention**, the hardware-verified identity medium is selected from a group comprising an **NFC-enabled RFID smartcard** storing a pre-generated 88-character string, and a **biometric sensor** utilizing a WebAuthn-compliant private key bound to the client hardware.

**According to a third aspect of the invention**, the method further comprises the step of transforming the PassCode Component into a high-entropy 256-bit key utilizing a **PBKDF2-HMAC-SHA512** derivation process, characterized in that the derivation parameters and the resulting key are purged from the client device memory upon termination of the browser session.

**The invention further provides** for a system architecture for hardware-verified zero-knowledge storage, the system comprising:

- A physical layer consisting of a **hardware-bound identity token** (NFC card or biometric sensor);
- An ephemeral layer resident in the **client-side RAM** configured to handle key derivation and encryption; and
- A storage layer consisting of **decentralized nodes** configured to host encrypted data packets without possession of the corresponding decryption keys.

**According to a fourth aspect of the invention,** there is provided a method for multi-factor authentication characterized by a **separation of concerns,** wherein **biometric data** (e.g., a fingerprint or facial scan) is utilized to prove the **identity** of the user, while a **physically-held token** (e.g., an NFC SmartCard or 44-character PassCode) is utilized to prove the **ownership** of specific digital assets (2).

**DETAILED DESCRIPTION:**

1. **System Architecture:** Browser based standard Javascript functions and APIs are used. No Persistence: Cryptographic keys exist only in the volatile memory (RAM) of the client's device during an active session. They are never written to the client's local disk in plain text and are wiped immediately upon logout or session timeout.

Separation of Concerns: We use the Fingerprint to prove who you are, but we use the PassCode to prove what you own.

On Client step 1 – Use `window.crypto.getRandomValues()` to generate a cryptographically strong random 88 character string called a "PassCode" and hold in temporary storage in the browser window.
On Client step 2 - Use **Web NFC API** to write the string from step 1 to a industry standard RFID card. When the user is ready to login to the storage portal, the RFID card is presented and read by the NFC API.

Login option 1:**A method step comprising** extracting a unique identifier component from a physically-stored 88-character string, wherein said identifier component is utilized for server-side user authentication **without** the transmission of the corresponding encryption key component. No email, no password. No Registration.

Login Option 2: Use Webauthn API to register on the storage portal and to register an encryption key saved to the local device secured by fingerprint. This allows users to authenticate with biometrics (fingerprint, face scan) or hardware keys (like FIDO security keys) instead of passwords.

Once the User is authenticated access is given to the portal to save or retrieve files. User selects file to upload to storage and prior to transmission the file is encrypted with a key derived from the second 44 characters using the Web Crypto API.

User selects file to download to local device. On receipt the file is decrypted with the key derived from the second 44 characters of a RFID card using the Web Crypto API **OR** from the hardware bound key generated by webauthn API previously saved to the smart device.

Transport Layer: TLS 1.3. All data in transit is wrapped in TLS 1.3, which removes legacy, vulnerable ciphers and ensures "Perfect Forward Secrecy" (PFS).

**2. Technical Separation of Concerns: Identity vs. Ownership**

The system operates on a dual-track verification methodology that decouples the user's physical person from their digital assets to ensure a zero-knowledge state.

**2.1 Proof of Identity (Who You Are):**

Identity verification is performed via the **WebAuthn API** or server-side lookup of the 44-character **Unique Identifier**. This layer confirms that the individual attempting access is the authorized account holder. However, a technical distinction is made in that **identity verification does not grant the server access to the data**. Successful identification merely permits the client-side environment to initiate the subsequent "Ownership" layer.

**2.2 Proof of Ownership (What You Own):**

Ownership is verified through the possession of the physical **NFC SmartCard** or the hardware-bound **PassCode**. This component contains the cryptographic material (the second 44 characters) required to derive the decryption keys.

- **Technical Effect:** Unlike traditional systems where identity and encryption keys are bundled (allowing a server administrator to "impersonate" a user and read their data), the **ArmerTech** framework ensures that even if the **Identity** layer is compromised or bypassed, the **Ownership** layer (the physical key) remains an absolute requirement for data legibility.

**2.3 The Resulting Security Posture:**

By binding the **Identity** (Biometric/UID) to the **Ownership** (Physical PassCode), the system achieves a state of "Privilege Separation." The server manages the **Identity** database but is mathematically excluded from the **Ownership** layer. Consequently, the user is the only entity that holds both the "who" and the "what," creating a cohesive, hardware-verified zero-knowledge environment where the storage provider is reduced to a "blind host."

### 3. The Encryption Process: Local Key Derivation and Application

The system utilizes a **Client-Side Key Derivation Function (CKDF)** to ensure that the cryptographic master key never traverses the network. The process is characterized by the following technical specifications:

#### 3.1. Key Derivation (PBKDF2-HMAC-SHA512):

The second component of the 88-character string (the 44-character **PassCode**) is subjected to a Password-Based Key Derivation Function 2 (PBKDF2).

- **Pseudo-Random Function (PRF):** The system utilizes **HMAC-SHA512**. By employing a 512-bit hash, the internal state of the KDF remains significantly wider than the final key, mitigating the risk of bit-bias or collision.
- **Work Factor (Iterations):** A minimum of **600,000 iterations** is applied. This technical "cost" is designed to thwart "brute-force" and "offline dictionary" attacks by making the computational overhead for a single key derivation prohibitively high for unauthorized entities.
- **Salt Management:** A unique, cryptographically secure **32-byte salt** is generated per user/file. This ensures that identical PassCodes on different accounts result in entirely different encryption keys, effectively neutralizing "Rainbow Table" attacks.

#### 3.2. Symmetric Encryption (AES-256 GCM):

The resulting 256-bit key is applied to the data utilizing **Advanced Encryption Standard (AES) in Galois/Counter Mode (GCM)**.

- **Authenticated Encryption:** The choice of **GCM** is critical to the methodology as it provides both **confidentiality** and **integrity (authenticity)**. Any unauthorized alteration of the encrypted data packet during storage or transit will result in a failed decryption attempt on the client side, alerting the user to a "Data Integrity Breach."
- **Non-Persistence:** All intermediate values—including the derived key, the initialisation vector (IV), and the raw PassCode—are held exclusively in the **volatile RAM** of the browser and are programmatically overwritten or cleared upon session termination.

#### 3.3 Identity Verification Protocol: Blind Authentication Methodology

The system employs a "Blind Authentication" protocol to verify a user's right to access the storage portal without exposing the underlying cryptographic keys. This is achieved through two functionally equivalent hardware-verified pathways.

### 3.3.1 Pathway 1: Physical Medium (NFC/RFID) Identifier

- **Unique Identifier (UID) Extraction:** Upon the physical presentation of the NFC SmartCard, the system extracts only the first 44 characters of the 88-character PassCode.
- **Server-Side Comparison:** This 44-character UID is transmitted to the server as a non-sensitive "public" identifier. The server compares this value against a database of registered UIDs.
- **Security Effect:** Because the probability of a collision (two identical 44-character strings) is astronomically low ( $½^{256}$ combinations), the UID serves as a collision-resistant proxy for a traditional username/password, without the inherent vulnerabilities of user-generated credentials.

### 3.3.2 Pathway 2: Hardware-Bound Biometric Verification (WebAuthn)

- **Asymmetric Handshake:** The system utilizes the **WebAuthn API** to interface with the device's biometric sensor. During registration, a unique public/private key pair is generated; the private key is locked within the device's **Hardware Security Module (HSM)** or **Trusted Execution Environment (TEE)**.
- **Challenge-Response:** For authentication, the server issues a unique "challenge." The client signs this challenge using the hardware-bound private key. The server verifies the signature using the stored public key.

### 3.4 Technical Effect of the Verification Protocol

Regardless of the pathway chosen, the authentication process is characterized by a **complete isolation of the encryption keys**.

- **The Zero-Knowledge Handshake:** Successful authentication via either the UID (NFC) or the signed challenge (Biometric) grants the user access to the storage portal UI, but it **does not** provide the server with the decryption materials.
- **Client-Side Handover:** Only after the server validates the identity is the second 44-character component (NFC) or the hardware-stored encryption key (Biometric) retrieved from the local hardware and moved into the client’s **volatile RAM** for the encryption/decryption process.

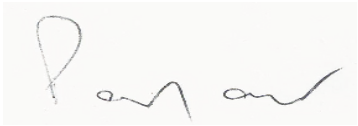
**BRIEF DESCRIPTION OF DRAWINGS (If any):**

*Nil*

**DATED at Ballito this 18 day of January, 2026.**

---

**SIGNATURE OF APPLICANT**



Paul Armer