Armertech White Paper: Executive Abstract

Title: Defeating the Data Breach: Elevating Enterprise Privacy through Armertech’s Zero-Knowledge Storage Architecture

Date: January 2026

Author: Paul Armer, Lead Architect, Armertech

Contact: paul@armertech.net

Subject: Zero-Knowledge Data Protection & Client-Side Encryption

Executive Summary

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 possesses the keys to the data they host.

The Armertech Solution

This white paper introduces the Armertech Zero-Knowledge Storage Architecture, a "privacy-by-design" framework built on standard end-to-end encryption (E2EE) protocols. Unlike conventional providers that "scramble" data but retain the master keys, Armertech implements a strict client-side encryption model.

Technical Core

Armertech’s technology stack leverages industry-standard AES-256 for data at rest and TLS 1.3 for data in transit, ensuring military-grade protection at every lifecycle stage. By generating and storing cryptographic keys exclusively on the user’s local device or on User owned and controlled physical media such as an NFC card, Armertech removes itself from the trust chain.

Key Benefits for 2026 Enterprises:

Immunity to Provider Breaches: Even if Armertech's servers are compromised, attackers gain access only to "digital gibberish"—unreadable ciphertext that cannot be decrypted without the user's local keys.

Zero-Knowledge Compliance: By ensuring the provider has no "knowledge" of the data content, businesses can meet the most stringent data sovereignty and privacy mandates without relying on third-party promises.

Reduced Insider Threat: Because administrative access does not grant decryption capability, Armertech eliminates the risk of unauthorized internal data access by service personnel.

Conclusion

Armertech transforms data security from a reactive measure into a mathematical guarantee. This paper details how organizations can leverage Armertech to reclaim their digital autonomy, ensuring that their most valuable intellectual property remains private, secure, and under their exclusive control.

Section 2: The Trustless Architecture – Breaking the Chain of Custody

2.1 The Philosophy of "Trustless" Storage

Traditional cloud models are built on a "Trust-Me" basis. Users trust that the provider won't look at their files, that their admins are honest, and that their encryption keys are stored securely. Armertech replaces "Trust-Me" with "Show-Me" (Mathematical Verification). Our architecture is "Trustless," meaning the user does not need to trust Armertech because, cryptographically, Armertech is incapable of accessing the data.

2.2 Client-Side Key Generation (The "Device-First" Rule)

In the Armertech ecosystem, the encryption process begins and ends on the user's local machine.

Key Derivation: We utilize PBKDF2 (Password-Based Key Derivation Function 2) with a minimum of 250,000 iterations and a unique salt. This transforms the user's PassCode into a high-entropy master key locally.

Zero-Knowledge Handshake: The user's actual PassCode never leaves their device. Instead, we use a secure identification hash to verify identity without ever seeing the secret used to generate it.

2.3 The "Digital Gibberish" Protocol (Data at Rest)

When a file is uploaded to Armertech:

Local Encryption: The file is encrypted via AES-256-GCM (Galois/Counter Mode) on the client device.

Encrypted Metadata: Unlike competitors who may leave file names or sizes visible, Armertech encrypts the metadata as well. To our servers, your "2026_Tax_Returns.pdf" looks like a randomized string: 8f3a1b9e....

Fragmented Storage: The encrypted ciphertext is stored in isolated blocks. Without the client-side key, these blocks are mathematically impossible to reassemble or decrypt, even with current supercomputing capabilities.

2.4 Immunity to Subpoena and Compromise

Because Armertech does not possess the keys, we are "subpoena-proof." If a government or third party requests access to a user's data, Armertech can only provide the encrypted ciphertext. Since we do not hold the keys, we cannot be legally or technically compelled to decrypt it. This transfers the legal "Last Line of Defense" back to the rightful owner of the data: The Client.

Section 3: Absolute Sovereignty – The 88-Character PassCode Standard

3.1 The End of the "Backdoor" Vulnerability

Most enterprise storage providers offer "Password Resets." Technically, a password reset is a systemic vulnerability; it implies the provider has a master key or a way to re-encrypt data. Armertech has abolished this vulnerability. By design, Armertech is a "Non-Custodial" storage provider. We do not hold your keys, and therefore, we cannot reset them.

3.2 The 88-Character Cryptographic Barrier

Armertech utilizes an 88-Character PassCode system. This PassCode is the sole source of entropy for the local encryption keys.

Entropy and Strength: An 88-character string provides a level of cryptographic strength that renders "brute-force" attacks statistically impossible within the lifetime of the universe.

User Ownership: This PassCode is generated on the client-side and is never transmitted to, or stored on, Armertech servers.

3.3 The "Responsible Adult" Protocol

Armertech is designed for entities that prioritize absolute security over administrative convenience.

No Recovery Mechanism: There is no "Forgot Password" link. There is no hidden recovery question.

The Zero-Knowledge Pledge: Our inability to recover a lost PassCode is the technical proof that we cannot access your data. If we could "help" a user who lost their code, it would mean we could also "help" a hacker or an unauthorized government agency.

Client Responsibility: Armertech clients are strictly responsible for the offline backup and physical security of their 88-character PassCode. This "Manual Backup" requirement is the final bridge in the zero-knowledge chain, ensuring that the user—and only the user—is the master of their digital estate.

Section 4: Industry Applications & Liability Shielding

4.1 The Cost of Compliance in 2026

With data privacy regulations now enforcing "Strict Liability" for data holders, the mere possession of readable client data is a financial risk. Armertech’s Zero-Knowledge architecture transforms this liability into an asset by ensuring the business—though it uses the cloud—never truly "hands over" the data to a third party.

4.2 Law Firms: Protecting the Attorney-Client Privilege

In the legal sector, a compromised cloud provider can lead to a breach of Attorney-Client privilege, resulting in disbarment or massive malpractice lawsuits.

The Armertech Advantage: Discovery documents, strategy memos, and witness statements are encrypted locally via the 88-character PassCode.

The "Subpoena Wall": If a law firm's storage provider is subpoenaed, Armertech can only provide encrypted data. The firm remains the sole gatekeeper of the keys, maintaining total control over legal privilege.

4.3 Healthcare: Beyond HIPAA/POPIA Requirements

Healthcare providers handle Sensitive Personal Information (SPI) that requires "reasonable and appropriate" safeguards.

The Armertech Advantage: Standard encryption is "reasonable," but Zero-Knowledge is "impenetrable."

Breach Notification Immunity: In many jurisdictions, if data is breached but is proven to be "unreadable and unlinked to a key," the legal requirement to notify every single patient (a process costing millions) can be waived. Armertech provides the cryptographic proof that the data remains unreadable.

4.4 Private Equity & Intellectual Property (IP)
For firms holding trade secrets, blueprints, or M&A (Merger & Acquisition) details, the risk is not just a fine—it is the loss of competitive advantage.

The Armertech Advantage: Armertech prevents "Insider Threats" at the provider level. Even a rogue engineer at the data center cannot "scrape" the drives for sensitive IP, because the decryption logic and the key never leaves the client's office.

Conclusion: The Future of Digital Sovereignty
Armertech is not a utility; it is a fortress. By removing the "human element" of password resets and provider-side keys, we offer the only storage solution that respects the maturity and responsibility of the modern enterprise. With an 88-character PassCode and our Zero-Knowledge protocol, your data is no longer a liability—it is a locked vault to which only you hold the key.

A Lost Passcode (encryption key) will result in this response: "I can't recover your PassCode, and that's exactly why you should pay me. It means no one else can get your data either."

Appendix A: Technical Specifications & Cryptographic Primitives

A.1 Encryption Standards

Armertech employs the Advanced Encryption Standard (AES) in Galois/Counter Mode (GCM) with a 256-bit key length.

Why GCM?: Unlike older modes (like CBC), GCM provides both confidentiality and authenticity. It includes an Authentication Tag that ensures the data has not been tampered with while stored. If a single bit of the encrypted file is altered, the decryption will fail, alerting the user to a potential "bit-rot" or malicious injection.

A.2 Key Derivation Function (KDF)

To transform the user’s 88-character PassCode into a cryptographic key, Armertech utilizes PBKDF2-HMAC-SHA512.

Iterations: 600,000+ (Configurable to higher limits for Enterprise clients).
Salt: A unique, cryptographically secure 32-byte salt is generated for every user to prevent "Rainbow Table" or pre-computed credential attacks.
SHA-512: By using a 512-bit hash function, we ensure that the internal state of the KDF is significantly wider than the resulting 256-bit key, eliminating collision risks.

A.3 Implementation Libraries (Verified 2026)

Armertech utilizes only peer-reviewed, open-source cryptographic libraries to ensure no proprietary "backdoors" exist in the source code:

Web Environment: Web Crypto API (SubtleCrypto). This is the gold standard for browser-based encryption, as it runs in a hardware-isolated environment within the browser, protecting keys from most side-channel JavaScript attacks.

Mobile/Desktop Core: libsodium or OpenSSL 3.x. These are the most scrutinized cryptographic libraries in history.

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).

A.4 Zero-Knowledge Verification

Identity Proof: Armertech uses a Salted Hashing mechanism for login. The server stores a hash of the user’s identifier, but not the hash used for file encryption.

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.

Why Armertech Uses the PassCode Over Biometrics for Encryption:
While Armertech supports WebAuthn (Fingerprint) for Account Access (Login), we use the 88-Character PassCode, stored on NFC or USB, for the actual Data Encryption.

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

Privacy from Big Tech: Biometrics are often tied to Huawei (HMS) or Google (GMS) system keys. By using an independent 88-character string, Armertech ensures that even if a government forces Google or Huawei to hand over their master system keys, your Armertech vault remains locked because the encryption was never tied to their system biometrics.

Future-Proofing: Phones break, and NFC cards de-magnetize. A 100% mathematical string (the PassCode) is the only "key" that will still work in 20 years, regardless of what hardware you are using.

Appendix B: Comparative Methodology Analysis

Security vs. Accessibility: Choosing the Right Authentication Path

Methodology	**Pros**	**Cons**
NFC Security Card	**Physical Air-Gap**: The key is "something you have." If the device is stolen, the key is not on it. **Speed**: Rapid "Tap-and-Go" access.	**Physical Vulnerability**: If the card is lost or stolen, access is gone. **Clone Risk**: Basic NFC tags can be skimmed or cloned by sophisticated actors. Mitigated by faraday cage envelope. **Compatibility**: Requires specific hardware (a NFC chip) which is standard on modern smartphones or

		as a USB reader for PC.
WebAuthn Fingerprint	**Non-Transferable**: Biometrics cannot be "lost" or "stolen" like a card. **Convenience**: High user-adoption rate; no need to remember strings. **Anti-Phishing**: Hardware-bound to the specific domain (eg, armertech.net).	**Hardware Dependency**: Relies on "System Integrity." If the OS (HMS/GMS) is compromised or outdated, biometrics fail. **Not True Zero-Knowledge**: The "key" is often managed by the OS (Google/Huawei), not the user. **Device Bound**: If the phone breaks (e.g., a "smashed P40"), the biometric credential is lost forever unless synced to a cloud account. So User Autonomy, ownership and control is LOST.
Armertech 88-Char PassCode	**Absolute Sovereignty**: The only method that is truly "Zero-Knowledge." The key is independent of hardware and OS providers. **Universal Recovery**: Can be used on *any* device, anywhere, provided the user has their code. **Brute-Force Immunity**: 88 characters exceed the "Heat Death of the Universe" computational limit for cracking.	**Human Responsibility**: Requires the user to act as a "Responsible Adult." Loss of the code equals permanent data loss. **Input Friction**: Requires a secure manager or physical copy; it cannot be typed from memory like a 4-digit PIN.

Appendix C: Biometric Login - Anonymity and Autonomy Lost

When a device used for WebAuthn/FIDO registration is lost, what happens next depends on the type of passkey used and the backup measures you previously established. Because biometric data and private keys never leave the device, they cannot be recovered from the lost hardware itself

Recovery typically follows one of these paths:

1. Synced Passkeys (Cloud Recovery)

If you use "synced passkeys" (provided by ecosystems like **Apple iCloud Keychain**, **Google Password Manager**, or **Microsoft**), the private key is securely backed up in the cloud.

- **Action:** When you sign into your cloud account on a new device, your passkeys are automatically restored.
- **Security:** This eliminates the need for manual backups while maintaining phishing resistance.

2. Device-Bound Credentials (Hardware Recovery)

If you use a physical security key (like a **YubiKey**) or a "device-bound" passkey that does not sync, the private key exists *only* on that specific hardware.

- **Multiple Registered Keys:** The primary FIDO recommendation is to register at least **two** authenticators for every account (e.g., your phone and a backup security key kept in a safe place). If one is lost, you use the second to log in and immediately **revoke** the lost device.
- **Recovery Codes:** Many services (like GitHub or Google) provide a set of one-time-use **recovery codes** during setup. You can use these to bypass the biometric/FIDO check, regain access, and register a new device.

3. Relying Party (Website) Fallbacks

If no backup FIDO device or recovery code is available, you must rely on the specific website's manual recovery process.

- **Alternative MFA:** The site may allow you to verify your identity via email, SMS, or a different 2FA method you previously linked.
- **Customer Support:** In extreme cases, you may need to contact the service provider's support team to prove your identity through other means (e.g., verifying recent transactions or personal details).

Summary of Risks

- **Locked Out:** If you have only one device-bound authenticator and no recovery codes or alternative MFA, you may be **permanently locked out** of the account.
- **Revocation is Critical:** If a device is stolen, you must log in using a backup method and **remove** the lost device from your account settings to prevent unauthorized access.

Conclusion

Biometric Passkeyless login is not straight forward and faces more complex challenges vs the use of a NFC SmartCard. There should be a backup of the 88 character PassCode either on another SmartCard or on USB drive. There is no need to sync to cloud, no need to reregister on the website, no need to give up personal information such as email or mobile number, no need for anything else except your PassCode value.