

A Novel Model of Comprehensive Data Encryption for the Website Using Blockchain

DOI: <https://doi.org/10.54654/isj.v1i27.1252>

Pham Van Huong*, Nguyen Duc Tuyen, Dang Hai Long,
Tran Quoc Toanh, Nguyen Ngoc Tuyen

Abstract—The paper proposes a comprehensive data security model for blockchain-based web applications. This model can be used as a general template for Web3 applications. The model consists of two parts: a blockchain core with integrated database encryption modules, replacing Fabric CA; and an application part that also integrates file encryption, database encryption, and digital signatures. The proposed model was tested on a VBCC management website using Hyperledger Fabric. File and database encryption uses AES, and digital signatures use ECDSA. To improve performance, we also replaced the GolevelDB database management system with LevelDB. Experimental results confirm the accuracy and good performance.

Tóm tắt—Bài báo này đề xuất một mô hình bảo mật dữ liệu toàn diện cho các ứng dụng Web sử dụng chuỗi khối. Mô hình này có thể sử dụng làm khuôn mẫu chung cho các ứng dụng Web3. Mô hình gồm 2 phần: phần lõi nền tảng chuỗi khối được tích hợp thêm các mô-đun mã hóa cơ sở dữ liệu, thay thế Fabric CA; phần ứng dụng cũng được tích hợp mã file, mã cơ sở dữ liệu và ký số. Mô hình đề xuất được thử nghiệm cho website quản lý VBCC sử dụng Hyperledger Fabric. Mã file và CSDL sử dụng AES, chữ ký số sử dụng ECDSA. Để cải tiến hiệu năng, chúng tôi cũng thay thế hệ quản trị CSDL GolevelDB bằng LevelDB. Kết quả thực nghiệm khẳng định về độ chính xác và hiệu năng tốt.

Keywords—Blockchain; file encryption; database encryption; hyperledger fabric; AES; Fabric-CA.

Từ khóa— Chuỗi khối; mã hóa file; mã hóa cơ sở dữ liệu; hyperledger fabric; AES; Fabric-CA.

This manuscript was received on January 5, 2026. It was reviewed on February 25, 2026, revised on March 10, 2026 and accepted on May 4, 2026.

* Corresponding author

I. INTRODUCTION

Nowadays, Blockchain and Web3 are ushering in a new generation of the internet, where data and digital assets are managed transparently, decentralized, and owned by the users themselves. Blockchain provides a transparent, and tamper-proof platform for storing information, while Web3 builds upon that foundation to create applications such as decentralized wallets, smart contracts, Decentralized Finance (DeFi), NFT Non-Fungible Token (NFT), and Decentralized Autonomous Organization (DAO). By eliminating intermediaries, transactions become faster, cheaper, and more reliable. Web3 model not only offers new ways to exchange value but also opens up opportunities for the development of entirely new business models, communities, and digital economies. Web3 model is a combination of the traditional Web2 model and Blockchain, along with the InterPlanetary File System (IPFS) [1-3].

In the Web3 model, data on the blockchain is very difficult to modify, but off-chain files and databases are vulnerable to attacks. Many applications store user files (images, documents, NFT metadata, etc.) on systems like IPFS or traditional servers, where data can be replaced, deleted, or injected with malicious code. Furthermore, applications still need off-chain databases like PostgreSQL, MongoDB, etc., to process user information, manage login sessions, or speed up queries. These systems can be vulnerable to SQL injection attacks, unauthorized access, or exploitation of poorly protected APIs.

Furthermore, in blockchain, the state database stores current information about

accounts, contracts, and balances. Although protected by consensus mechanisms, the state layer is still vulnerable to attacks through smart contract vulnerabilities, transaction execution errors, or attacks on the state processing node. Errors such as incorrect transaction order, logical errors in state updates, or manipulation of node synchronization can cause state to become skewed, resulting in asset loss or application malfunction. Therefore, protecting the state layer depends not only on the core blockchain but also on the security of the nodes, the execution environment, and the application logic.

Additionally, using blockchain only solves the problem of preventing alteration and ensuring transparency. Ensuring the integrity and non-repudiation of data and documents requires the use of digital signatures. To ensure legal validity, digital signatures must be issued through digital certificates and authenticated by a dedicated RootCA of the Government Cipher Committee or the RootCA of the Ministry of Science and Technology [18].

Therefore, off-chain data security, including file encryption, database encryption, and legal compliance, requires collaboration with the Government Cipher Committee's RootCA. To ensure transparent storage and prevent modification, blockchain technology is necessary. Furthermore, it's crucial to secure the private database and stateful database within the blockchain core. To address these challenges, this paper proposes a new, comprehensive model for holistic data security across the Web3 model, from the application layer to the blockchain layer. The proposed solution focuses on key points: file encryption using blockchain algorithms like AES and database encryption (using TDE), ECDSA-X509 digital signing with the Government Cipher Committee's RootCA at the application layer; and lightweight database encryption within the blockchain core [9, 18].

The remainder of the paper is organized as follows: *Section II* - Presents a review of related works. *Section III* - Covers the theoretical background. *Section IV* - Details the proposed method. *Section V* - Describes the experimental setup and evaluation of the approach. *Section VI*

- Concludes the paper and suggests future research directions.

II. RELATED WORKS

Many studies have focused on addressing data security and privacy issues in decentralized applications, particularly in the context of increasingly complex Web3 systems. Common approaches typically combine Blockchain with off-chain storage systems such as IPFS or cloud computing to address scalability challenges. However, research indicates that data stored on IPFS is public by default and lacks integrated access control mechanisms. To address this, some authors have proposed using Attribute-Based Encryption (ABE) or access control via Smart Contracts. While these methods enhance security, they often introduce significant transaction processing latency or fail to comprehensively address the challenge of securely and conveniently managing secret keys for end users [1 - 3, 24, 26].

Regarding database security, Transparent Data Encryption (TDE) techniques have been extensively researched and widely implemented in traditional database management systems such as SQL Server, Oracle, and MySQL to protect data-at-rest [11, 10]. However, in Hyperledger Fabric architecture, existing research primarily focuses on consensus mechanisms and network performance, with limited attention to the security of the state storage layer (State Database - typically LevelDB or CouchDB) [7, 12, 15]. The fact that state data is stored in plaintext at nodes creates risks of internal attacks or physical node compromise. Some proposals to encrypt data at the application layer before submitting to the blockchain ensure confidentiality but sacrifice the ability to perform complex queries or validate data by other peers in the network.

Additionally, the issue of identity and legal validity of transactions has received considerable attention. Although Hyperledger Fabric provides Fabric CA for identity management and X.509 digital certificates, this system typically operates independently and lacks integration with government-level or national Public Key Infrastructure [8]. Research on integrating digital signatures into Blockchain often stops at user

authentication without proposing a comprehensive model to ensure legally valid non-repudiation, particularly the integration with specialized RootCAs such as those of the Government Cipher Committee or the Ministry of Science and Technology in Vietnam [18 - 20].

III. BACKGROUND

A. Original Architecture

The original architecture is described in Figure 1. The basic Web3 application model on Hyperledger Fabric is typically designed with a layered architecture, comprising the Application Layer and the Blockchain Layer.

At the application layer, users interact through web/mobile interfaces to perform business operations. Due to cost and block size limitations on Blockchain, large data files are not stored directly on-chain. Instead, they are stored in off-chain storage systems such as IPFS, MinIO server, or cloud services (Google Drive, OneDrive) [14, 16]. Only the file hash and important metadata are sent down to the blockchain layer to ensure integrity.

Simultaneously, these systems maintain a traditional database (SQL or NoSQL) to manage sessions, user information, and serve high-speed queries [3, 13].

At the Blockchain layer, the Hyperledger Fabric platform manages the distributed ledger. Transaction data after being validated by peers is packaged into blocks and stored in the Ledger. The current system state (World State) is stored in a State Database (typically LevelDB or CouchDB) at each peer to serve rapid retrieval of the latest values of keys without needing to traverse the entire transaction history.

B. Security Risks

Although the above architecture leverages the advantages of transparency and decentralization of Blockchain technology, practical deployment reveals three groups of critical information security risks that default mechanisms have not thoroughly addressed [27, 28].

First are vulnerabilities at off-chain storage and processing points. In this model, user data

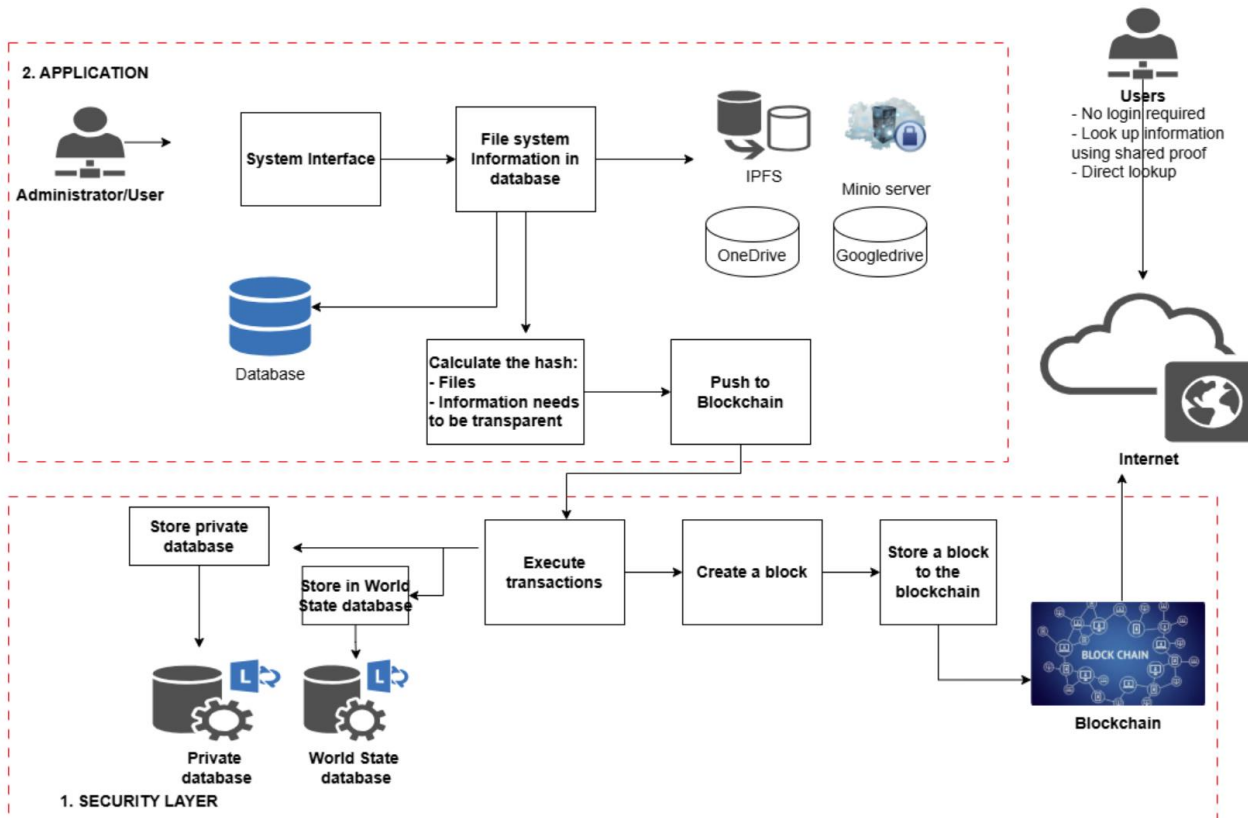


Figure 1. Original Web3 application architecture without encryption or digital signatures

stored on IPFS or traditional servers is typically in plaintext form to serve rapid access [15]. Without an intermediate encryption layer, this data is completely public and faces risks of being replaced, deleted, or injected with malicious code if the storage system is compromised or physically attacked. Simultaneously, business databases used for session management also have common attack surfaces such as SQL Injection or API exploitation, allowing attackers to extract sensitive data before it can be hashed and uploaded to the chain.

Second is the risk of direct attacks on the State Database. A technical characteristic of Hyperledger Fabric is maintaining the state database at each peer in the form of unencrypted key-value pairs to optimize read/write performance. This creates a serious vulnerability: if attackers infiltrate the node's physical infrastructure or exploit vulnerabilities in execution logic, they can directly read or manipulate asset states without interfering with the historical blockchain. Technical errors such as incorrect transaction ordering or node synchronization errors can also cause state deviation from the original ledger, causing errors throughout the application.

Third is the limitation regarding legality and non-repudiation capability. The original architecture uses an identity mechanism through Fabric CA to issue digital certificates to members. However, these certificates only have technical authentication value within the network and lack connection with national-level Public Key Infrastructure (PKI). Therefore, transactions, although technically transparent, lack the necessary legal binding to replace paper documents in administrative procedures or legal disputes. The absence of an integration mechanism with government-specialized RootCAs prevents the system from ensuring complete authentication according to current legal regulations.

IV. PROPOSED METHOD

A. The proposed solution model

Based on the initial architectural model of systems using the Web3 model and the security risks associated with it, we propose a secure Web3 system model as described in Figure 2. According to this solution model, data is comprehensively secured from the application level to the blockchain core. At the application layer, data stored in files and databases is encrypted. Files are digitally

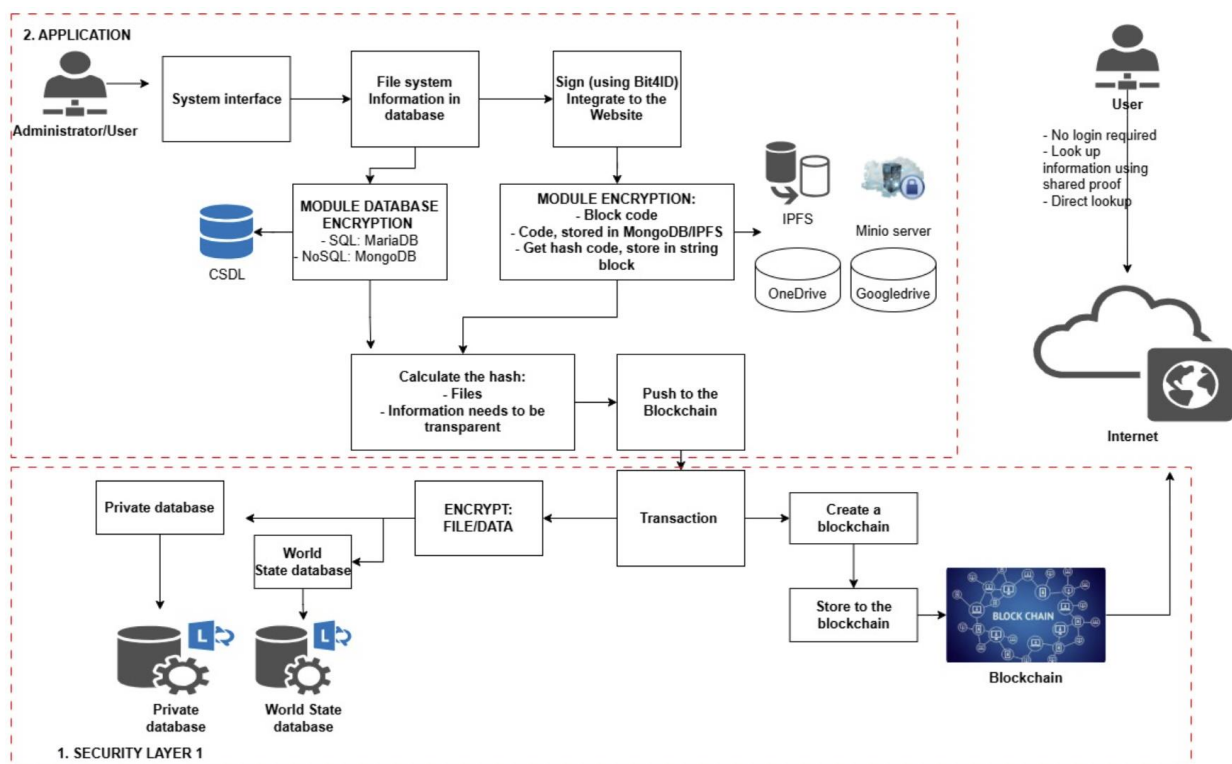


Figure 2. The proposed solution model

signed using the RootCA of the Government Cipher Committee to ensure legality, integrity, authenticity, and non-repudiation. The digitally signed files are encrypted and stored distributed across cloud servers using the S3 or IPFS protocol. Encrypted files are also hashed to generate h1 hash codes. Data stored in databases is also encrypted using TDE techniques. Several open-source database management systems such as MariaDB, MongoDB, etc., have integrated TDE modules [10, 11] with the internationally standardized AES algorithm [30]. These database management systems can be customized to use the Government Cipher Committee's proprietary encryption algorithms. The identifier field and other data fields requiring transparency will be hashed to create an h2 hash. The hash (h1, h2) will then be incorporated into the transaction. A block containing the transaction will be created and added to the blockchain. In layer 1, transaction state information and confidential information are stored in the transaction state database and a separate database [25]. These databases are not secure. Therefore, in our solution model, we also implement encryption for these databases. Furthermore, the LevelDB database encryption solution does not affect the consensus algorithm in the system because encryption is only performed when writing data to the file and decryption is performed when reading data from the file. However, it reduces performance due to the time spent on encryption/decryption during data queries [21 - 23].

B. Digital signature using USB token

Blockchain only ensures transparency and tamper-proofing. To ensure legal compliance, the proposed solution combines the use of blockchain with a digital signature authentication system. Digital signing is performed via an X509 certificate in a USB token device; the certificate is authenticated by the RootCA of the Government Cipher Committee or the RootCA of the Ministry of Science and Technology [17]. PDF files can be digitally signed and the signature embedded within the file; other file formats can be hashed and signed using a hash code. In the proposed solution, we use a USB token and a corresponding background digital

signing service to sign files and messages [11, 19, 20].

C. File encryption

File encryption algorithm

Input:

- Digitally signed file
- File containing password

Output:

- Cipher file
 - Cipher of (K_1, IV_1)
1. Generate random K_2, IV_2
 2. Encrypt file: cipher file = AES (plain file, IV_2, K_2).
 3. Generate random K_1, IV_1
 4. Encrypt (K_2, IV_2) using AES algorithm with key K_1 and IV_1
 5. Create a new cipher file = cipher file (in Step 2) + Cipher of (K_2, IV_2) . Save the cipher file to MinIO/IPFS
 6. Derive key: $K_0 = \text{PBKDF2}(\text{Password})$
 7. Encrypt (K_1, IV_1) : Cipher of $(K_1, IV_1) = \text{AES}(K_1, IV_1)$, using key K_0 and $IV = K_0/2$
 8. Save the cipher of (K_1, IV_1) to the database.

Key Management (Password Change)

1. $K_0 = \text{PBKDF2}(\text{old password})$
Use K_0 to decrypt to (K_1, IV_1)
2. $K_0' = \text{PBKDF2}(\text{new password})$
Use K_0' encrypt (K_1, IV_1) and save the cipher of (K_1, IV_1) to database.

D. Create blocks and save them to the blockchain

Algorithm of creating and saving

Input:

- File signed
- File containing password
- Fields need to be transparent in the database

Output:

- Block is created and saved to the blockchain
1. Read the files encrypted
 2. Hash the cipher file F
 $h_1 = \text{SHA256}(F)$
 3. Read the fields needed to be transparent in the database
 4. Create the hash h_2 of the fields needed to be transparent
 5. Create transaction

$T = \langle ID, h_1, h_2, \text{etc.} \rangle$

6. Package the transaction into a block and upload the block to the chain

E. Database encryption

Initialization

1. Generate a random 32-byte K_1
2. Generate a random 32-byte IV_1
3. Generate K_0 from the password (read from password.txt) using PBKDF2
4. Encrypt (K_1, IV_1) using K_0

Use after initialization

1. Load the encrypted K_1 from the file
2. Decrypt K_1 using the password
3. Use K_1 to encrypt/decrypt data

F. Information Verification

Information Verification Algorithm

Input: Information to be verified

Output: Verification status

1. Search the database for the ID corresponding to the information to be verified

2. Retrieve the code file corresponding to the ID and hash it

$$h_1' = \text{SHA256}(\text{file})$$

3. Retrieve the data fields to be transparent corresponding to the ID and hash it

$$h_2' = \text{SHAs56}(\text{fields})$$

4. Query the block containing the ID from the blockchain

5. Retrieve h_1, h_2 from the block and match them with h_1', h_2' . If they match, the data is intact.

V. SYSTEM IMPLEMENTATION

To realize the proposed model and validate research hypotheses, we proceeded to develop and pilot the "Digital Diploma and Certificate Management System" applied at the Academy of Cryptography Techniques [4-6]. The system, shown in Figure 3, was designed to ensure integrity and transparency.

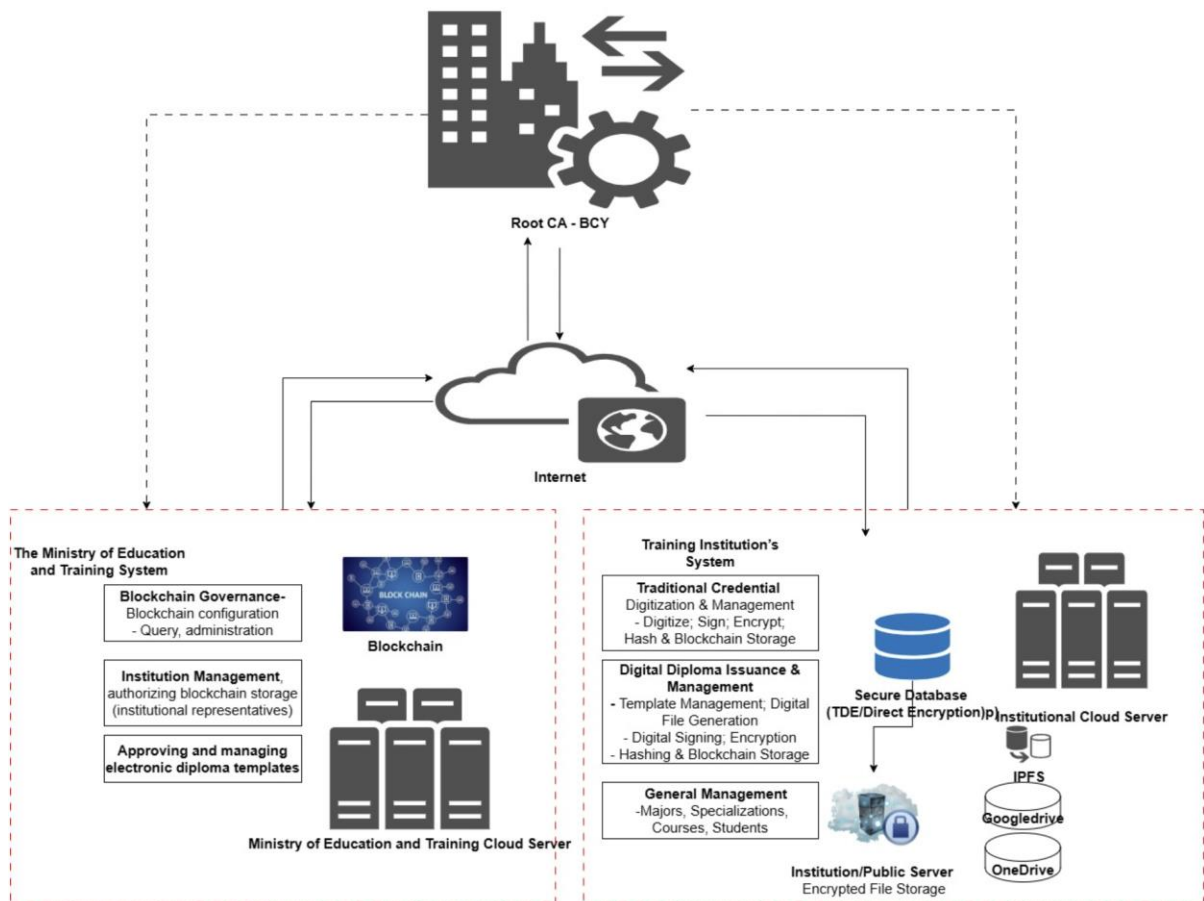


Figure 3. System Management Architecture

The system was designed and developed based on a layered architectural model, with clear separation between user interface, business logic processing, and the data storage layer combined with blockchain. The main actors interacting with the system include Administrative Staff and Students, performing operations such as querying and issuing digital diplomas and certificates [11, 21 - 23].

The Application Layer serves as the central hub for business processing and operates as an API Gateway for the entire system. This layer receives requests from the interface through RESTful API standards in JSON format. The main function of the application layer is to distribute and coordinate requests to corresponding storage subsystems: reading/writing metadata or student and diploma information to the database; uploading/downloading signed PDF files; or sending Invoke/Query Chaincode commands to the Blockchain network through the Fabric Gateway SDK. Administrative staff will use the system to issue certificates, and students will use QR codes to query information.

The Data Layer is designed following a hybrid storage model to optimize performance and costs. MongoDB database is responsible for storing metadata and user information, allowing the API to quickly retrieve diploma-related information. MinIO Server serves as Object Storage containing original signed diploma PDF files. Before any data is stored, the encryption module encrypts the content of diploma files (PDF) to ensure confidentiality and prevent unauthorized access even if the storage system is compromised.

The most critical component is the Hyperledger Fabric network, comprising core components: Fabric Gateway connecting the API (written in Go) with the network; Endorsing Peers receiving commands, processing and endorsing transactions; Ordering Service performing sequencing and distributing new blocks; and finally the Ledger together with World State database storing hashes of diplomas and files, helping authenticate the integrity and origin of documents. Notably, the research team

implemented improvements by *completely replacing the default Certificate Authority (CA) system* (Fabric-CA) with a custom CA solution KMAS-CA (Custom-CA). Instead of using available automatic key generation tools, the system integrates the KMAS-CA module using OpenSSL combined with Go modules to establish a separate Root CA, issuing X.509 digital certificates and managing Membership Service Provider (MSP) for each organization participating in the network. This solution ensures the system meets stringent requirements for legality and information security in government-level transactions.

To address the risk of direct attacks on the State Database, which stores data in plaintext, the system has integrated a *Core Database Encryption Module*. Unlike the default mechanism of Hyperledger Fabric, this module intervenes at the Peer's storage layer, encrypting all Key-Value pairs using AES algorithm before they are written to LevelDB or CouchDB. This mechanism ensures state data is always in ciphertext form when at-rest, rendering physical intrusion attempts into nodes or exploiting vulnerabilities to directly read data from hard drives ineffective. The combination of on-chain and off-chain data encryption with custom identity mechanisms creates a comprehensive defense-in-depth security architecture for the system.

However, applying encryption typically leads to processing speed degradation due to decryption computational costs. To solve this trade-off between security and performance, the research team optimized the storage architecture by *replacing the default GolevelDB database management system with LevelDB*. Switching to LevelDB significantly improves Input/Output (I/O) access speed, thereby compensating for latency caused by the AES encryption process. As a result, the system both ensures data security at the highest level (encryption at rest) while maintaining stable transaction processing performance, adequately meeting real-time requirements of the national diploma management problem.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

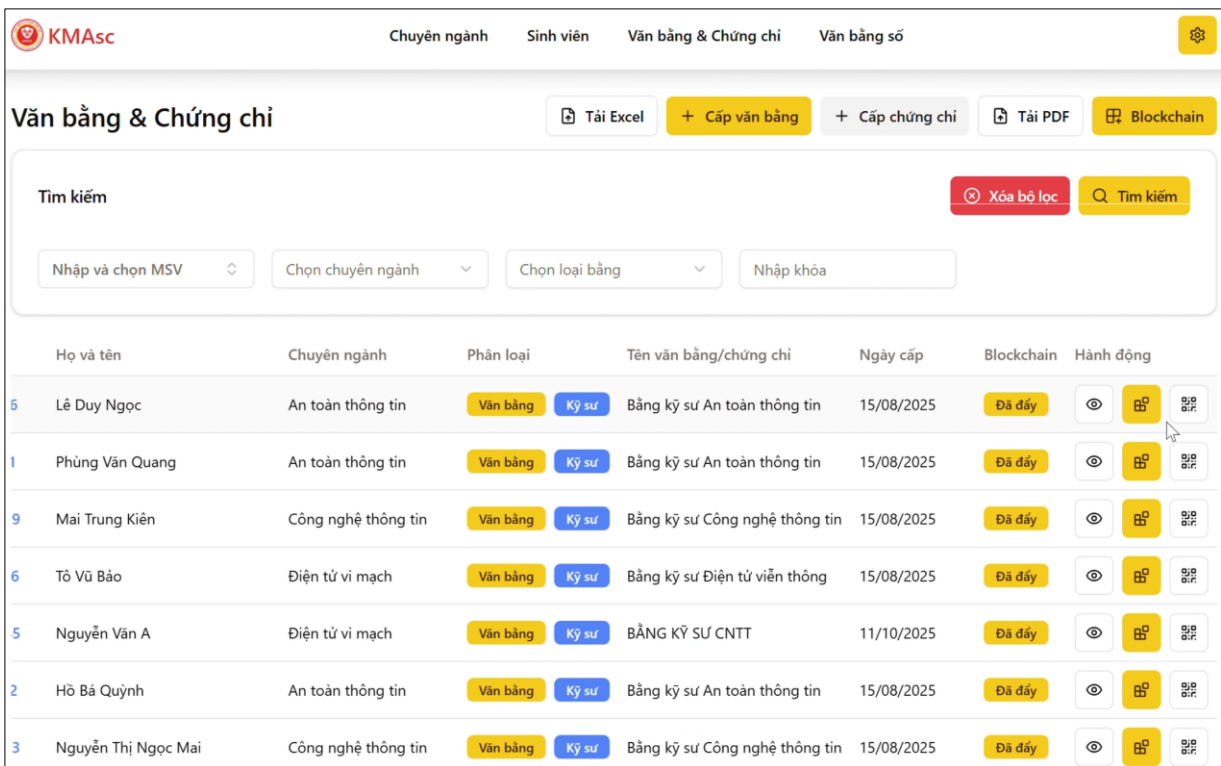
A. Experimental Results

In the functional testing scenario, with permission from the Academy of Cryptography Techniques, the research team executed the issuance and authentication process on a sample dataset of 200 diplomas as shown in Figure 4 and Figure 5. Experimental results demonstrate that the encryption mechanism operates accurately and efficiently. All diploma files after being uploaded to the off-chain storage system were automatically converted to ciphertext form, ensuring content cannot be accessed illegally using common tools without valid decryption keys. Simultaneously, the distributed Ledger successfully and immutably recorded transactions containing file and metadata hashes, establishing a reliable reference basis for the authentication process.

The system demonstrated the capability to detect fraudulent activities. When the research team performed simulated attack scenarios such as changing file content on the storage server or interfering with information fields in the

management database, the system's validation module immediately detected discrepancies between the recalculated hash and the hash stored on the blockchain. The system refused authentication and issued accurate warnings for modified diplomas, thereby confirming that data integrity is absolutely guaranteed.

Finally, regarding operational performance, the integration of the LevelDB database optimization solution initially showed positive effectiveness [21 - 23]. Although the system must perform additional complex encryption and decryption tasks to ensure data security, practical experience on the test website shows that response speeds for issuance and query transactions remained stable, with no network congestion or excessive latency affecting users. Digital signature functions, pushing data to the blockchain, and QR code authentication operated smoothly, adequately meeting the requirements of an online diploma management system. The system administration interface and diploma verification interface through QR code are depicted in Figures 4 and 5. Cryptographic



The screenshot displays the 'Văn bằng & Chứng chỉ' (Diploma & Certificate) management interface. At the top, there are navigation tabs: 'Chuyên ngành' (Specialty), 'Sinh viên' (Students), 'Văn bằng & Chứng chỉ' (Diploma & Certificate), and 'Văn bằng số' (Digital Diploma). The main header includes a search bar with 'Tìm kiếm' (Search) and a 'Xóa bộ lọc' (Clear filter) button. Below the search bar, there are filters for 'Nhập và chọn MSV' (Enter and select MSV), 'Chọn chuyên ngành' (Select specialty), 'Chọn loại bằng' (Select type of diploma), and 'Nhập khóa' (Enter key). The main content area is a table with columns: 'Họ và tên' (Full name), 'Chuyên ngành' (Specialty), 'Phân loại' (Classification), 'Tên văn bằng/chứng chỉ' (Diploma/Certificate name), 'Ngày cấp' (Issuance date), 'Blockchain' (Blockchain status), and 'Hành động' (Action). The table lists several diplomas, each with a unique ID, name, specialty, classification, issuance date, and blockchain status. The 'Hành động' column contains icons for viewing, deleting, and other actions.

	Họ và tên	Chuyên ngành	Phân loại	Tên văn bằng/chứng chỉ	Ngày cấp	Blockchain	Hành động
6	Lê Duy Ngọc	An toàn thông tin	Văn bằng Kỹ sư	Bằng kỹ sư An toàn thông tin	15/08/2025	Đã đầy	  
1	Phùng Văn Quang	An toàn thông tin	Văn bằng Kỹ sư	Bằng kỹ sư An toàn thông tin	15/08/2025	Đã đầy	  
9	Mai Trung Kiên	Công nghệ thông tin	Văn bằng Kỹ sư	Bằng kỹ sư Công nghệ thông tin	15/08/2025	Đã đầy	  
6	Tô Vũ Bảo	Điện tử vi mạch	Văn bằng Kỹ sư	Bằng kỹ sư Điện tử viễn thông	15/08/2025	Đã đầy	  
5	Nguyễn Văn A	Điện tử vi mạch	Văn bằng Kỹ sư	BẰNG KỸ SƯ CNTT	11/10/2025	Đã đầy	  
2	Hồ Bá Quỳnh	An toàn thông tin	Văn bằng Kỹ sư	Bằng kỹ sư An toàn thông tin	15/08/2025	Đã đầy	  
3	Nguyễn Thị Ngọc Mai	Công nghệ thông tin	Văn bằng Kỹ sư	Bằng kỹ sư Công nghệ thông tin	15/08/2025	Đã đầy	  

Figure 4. System Administration Interface

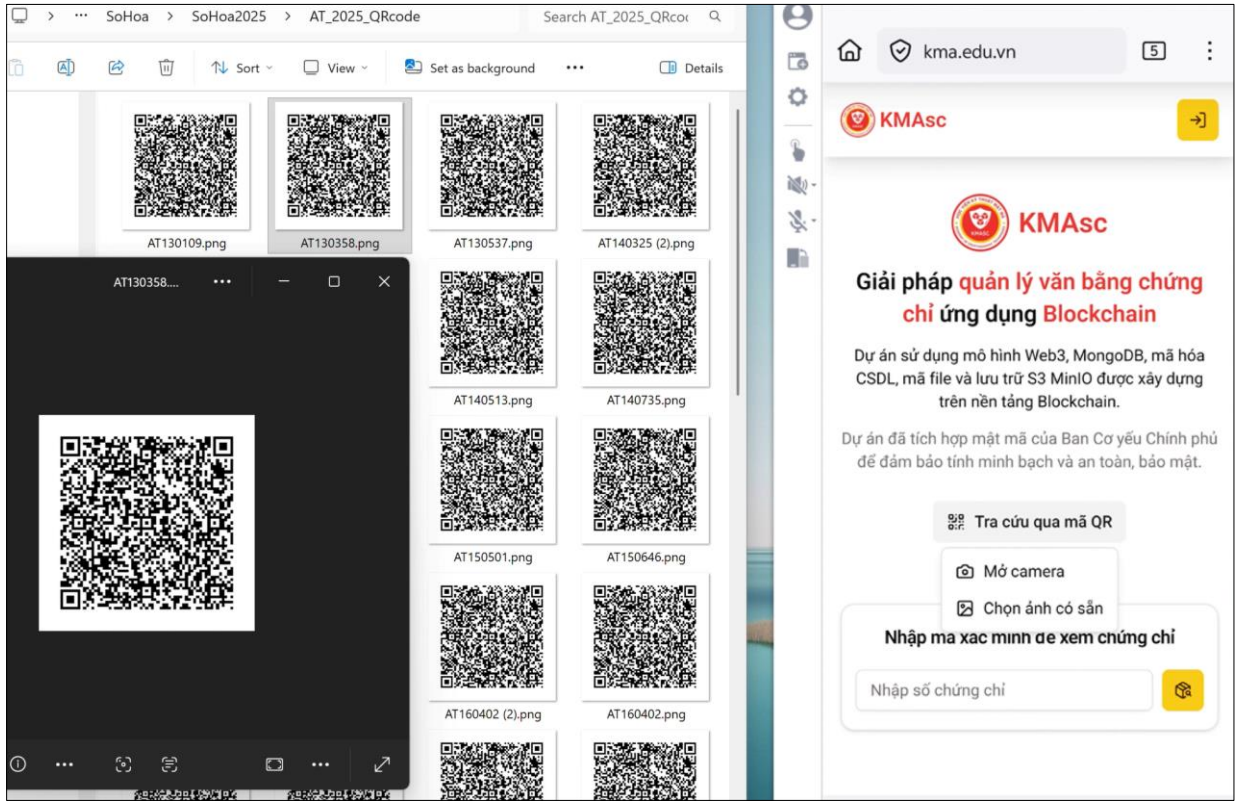


Figure 5. Diploma verification based on QR code

features integrated into the system are summarized in Table 1.

TABLE 1. SYSTEM SECURITY FEATURES

No.	Feature	Specification	Layer
1	File Encryption	AES-CBC 256 bit	Application
2	Digital Signature	RSA 4096 bit Using Bit4ID	Application
3	DB Encryption	AES-CBC 256	Layer 1
4	CA Replacement	Replace Fabric-CA with OpenSSL-CA ECDSA	Layer 1

B. Discussion

Evaluation of Integrity, Authenticity, and Non-Repudiation

Assume that:

- Message: m
- Hash function: H
- Private key: d
- Public key: $Q = d.G$

- ECDSA signature:

$$\sigma = (r, s) = \text{Sign}(H(m))$$

Integrity

Because hash codes are unique, to prove their integrity we have:

$$H(m) = H(m') \Leftrightarrow m = m'$$

$$\text{Verify}(H(m'), \sigma) = \text{false if } m \neq m'.$$

Authenticity

$\text{Verify}(H(m), \sigma) = \text{true} \Leftrightarrow m$ is signed with private key d

Non-Repudiation

Because of $Q = d.G \wedge \sigma = \text{Sign}(H(m))$, only the owner with d can create σ .

Security Evaluation

According to the file encryption scheme in the previous section, each file is encrypted with a randomly generated 256-bit key K_2 ; the file key is then encrypted with a randomly generated 256-bit key K_1 , and key K_1 is then encrypted with key K_0 , also 256 bits. Therefore, the brute-force key

space is $2^{256} \times 2^{256} \times 2^{256}$ keys. Assuming a key trial time of 1 second, the brute-force time to find the key would be approximately 2^{223} years.

Performance Evaluation

TABLE II. UNIT TEST FOR THE PUT AND GET FUNCTIONS

Order	put function		get function	
	Using GoLevelDB and No-encryption	Using LevelDB-C and encryption	Using GoLevelDB and No-encryption	Using LevelDB-C and encryption
1	0.004266	0.001819	0.001335	0.00022
2	0.002383	0.001600	0.000913	0.00021
3	0.002246	0.001768	0.001058	0.00021
4	0.002285	0.001675	0.000812	0.00019
5	0.002186	0.001619	0.000852	0.00021
6	0.002642	0.001695	0.001580	0.00021
7	0.002425	0.001781	0.000935	0.00021
8	0.002365	0.001648	0.000913	0.00020
9	0.002350	0.001664	0.001127	0.00022
10	0.002432	0.001705	0.000882	0.00021
Average	0.002459	0.001675	0.001141	0.00021

TABLE III. SUBMISSION AND QUERY TRANSACTION TIME

Order	Number of trans	Submit transaction (s)		Query transaction (s)	
		Using GoLevelDB and No-encryption	Using LevelDB-C and encryption	Using GoLevelDB and No-encryption	Using LevelDB-C and encryption
1	10	0.0022	0.0134	0.0008	0.0432
2	10	0.0021	0.0192	0.0009	0.0521
3	10	0.0023	0.0251	0.0011	0.0394
4	10	0.0020	0.0113	0.0007	0.0513
5	10	0.0022	0.0176	0.0008	0.0467
6	10	0.0023	0.0235	0.0010	0.0583
7	10	0.0020	0.0152	0.0009	0.0416
8	10	0.0021	0.0218	0.0008	0.0472
9	10	0.0022	0.0189	0.0009	0.0499
10	10	0.0022	0.0204	0.0009	0.0560
Average		0.00216	0.0186	0.00088	0.0486

Legal Validity Assessment

The digital signature authentication system using the Government Cipher Committee's RootCA has full legal validity according to Vietnamese law regulations [17]. The Government Cipher Committee's RootCA is a national root certification organization, ensuring the highest level of trust for the digital signature authentication chain. Digital signatures issued and authenticated in this system have legal validity equivalent to handwritten signatures and

seals when conducting electronic transactions. The system meets requirements for authentication, integrity, non-repudiation, and key management according to the Electronic Transactions Law and related decrees. Therefore, deploying a digital signature authentication system based on the Government Cipher Committee's RootCA ensures legality, security, and widespread applicability in government agencies and organizations.

VII. CONCLUSION

Blockchain is one of the key national strategic technologies. The Web3 model, utilizing blockchain and the IPFS distributed file system, serves as a framework for digital transformation applications. The main contribution of this paper is the proposal of a comprehensive data security model from the application layer to the blockchain core, combined with a digital signature authentication system to ensure legal validity. The proposed model has been analyzed, its security proven, and key management schemes evaluated. The proposed model has been validated through practical application deployment and performance evaluation. Experimentally, we have developed a digital diploma management system based on the proposed model, and the system is currently being piloted at the Academy of Cryptography Techniques.

Besides the theoretical and experimental achievements, the paper still has some limitations: it has not replaced Vietnam's own cryptography in the MongoDB database code using the TDE model; the password file is stored in plain text and is not protected; and there is a risk of cryptographic systems being attacked with the advent of quantum computing. To overcome this limitation, in subsequent studies, we will replace the AES algorithm [30] with Vietnamese-developed algorithms in the TDE model; integrate a software HSM solution to store passwords; and research and integrate post-quantum cryptography into the system [29].

ACKNOWLEDGEMENT

The article was completed with support from the Academy of Cryptography Techniques.

REFERENCES

- [1] E. Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in *Proc. EuroSys*, pp. 1–15, 2018.
- [2] Hyperledger Foundation, "Hyperledger Fabric Documentation" (2025). Access time: 06/15/2026, <https://hyperledgerfabric.readthedocs.io/>.
- [3] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System," 2014. Access time: 06/15/2026, <https://arxiv.org/abs/1407.3561>.
- [4] A. Grech and A. F. Camilleri, "Blockchain in Education," *JRC Science for Policy Report*, vol. 10, pp. 10–13, 2017.
- [5] T. Alam, "Blockchain-based Framework for Educational Certificates Verification," *Int. J. Comput. Appl.*, vol. 975, pp. 8887, 2020.
- [6] C. Cheng et al., "Blockchain-based Digital Diploma System Validation," in *MATEC Web of Conferences*, vol. 200, 2018.
- [7] X. Luo, Y. Li, and J. Wang, "A Security Problem Caused by the State Database in Hyperledger Fabric," in *Proc. IEEE TrustCom*, pp. 1021–1028, 2021.
- [8] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, and A. De Caro, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in *Proc. 13th EuroSys Conference*, 2018, pp. 1–15. DOI: 10.1145/3190508.3190538.
- [9] D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *Int. J. Inf. Secur.*, vol. 1, no. 1, pp. 36–63, 2001.
- [10] P. Gupta and A. Singh, "Transparent Data Encryption - Solution for Security of Database Contents," *Int. J. Comput. Appl.*, vol. 65, no. 14, 2013.
- [11] Oracle, "Transparent Data Encryption (TDE) in Oracle Database," *White Paper, Oracle Corporation*, 2022.
- [12] Google, "LevelDB: A Fast Key-Value Storage Library", 2025. Access time: 06/15/2026, <https://google.github.io/leveldb/>.
- [13] Q. Wang et al., "Hyper-IPFS: A Secure Off-chain Storage Scheme for Blockchain," in *Proc. IEEE SCC*, 2022.
- [14] S. Steichen, B. Fiz, and R. Norvill, "Blockchain-Based, Decentralized Access Control for IPFS," in *Proc. IEEE iThings*, pp. 1499–1506, 2018.
- [15] T. T. Nguyen, "Storing and Securing Data on IPFS in Blockchain Applications," *Journal of Science and Technology on Information Security*, vol. 12, no. 3, pp. 150–165, 2023.
- [16] MinIO, "MinIO Object Storage for Linux", 2025, Access time: 06/15/2026, <https://docs.min.io/aistor/>.
- [17] Quốc hội Việt Nam, "Luật Giao dịch điện tử số 20/2023/QH15", Hà Nội, 2023.
- [18] Ban Cơ yếu Chính phủ, "Thông tư 185/2019/TT-BQP về cung cấp, quản lý, sử dụng dịch vụ chứng thực chữ ký số chuyên dùng Chính phủ", 2019.
- [19] B. H. Nguyen, "A Novel Architecture for E-Diploma Management System using Hyperledger Fabric," *Vietnam J. Comput. Sci.*, vol. 8, no. 2, 2022.
- [20] T. D. Nguyen and T. H. Le, "Building a Blockchain-based Student Record Management System in Vietnam," in *Proc. RIVF*, 2021.
- [21] M. Valenta et al., "A Performance Analysis of Hyperledger Fabric 2.0," in *Proc. IEEE Blockchain*, 2021.
- [22] H. Sukhwani et al., "Performance Modeling of Hyperledger Fabric (Permissioned Blockchain Network)," in *Proc. IEEE NCA*, 2017.
- [23] Y. D. Lin et al., "Performance Benchmarking of Hyperledger Fabric 2.x," in *IEEE Access*, vol. 9, pp. 12345–12356, 2021.
- [24] Z. Zheng et al., "Blockchain Challenges and Opportunities: A Survey," *Int. J. Web Grid Serv.*, vol. 14, no. 4, pp. 352–375, 2018.
- [25] S. Xu et al., "A Taxonomy of Blockchain-Based Systems for Architecture Design," in *Proc. IEEE ICSA*, 2017.
- [26] D. Yaga et al., "Blockchain Technology Overview," *NIST Interagency Report 8202*, 2018.
- [27] D. Ongaro and J. Ousterhout, "In Search of an Understandable Consensus Algorithm," in *Proc. USENIX ATC*, 2014.
- [28] S. Bigi, A. Bracciali, G. Mezzandro, and M. Tuosto, "Validation of Smart Contracts in Hyperledger Fabric," in *Proc. IEEE Blockchain*, 2020.
- [29] S. Ben E.H. Soulam, Y. Connan, S. Guilley, S. Takarabt, "Secure Implementation of Post-Quantum Cryptography", *Journal of Science and Technology on Information Security*, vol. 1, no. 24, 2025. DOI: 10.54654/isj.v1i24.1079.
- [30] T.S. Nam, L.T. Dung, N.V. Long, "A High Throughput, Low Latency 105 Gbps Four-Pipeline Stage AES", *Journal of Science and Technology on Information Security*, vol. 1, no. 21, 2024. DOI: 10.54654/isj.v1i21.1029.

ABOUT THE AUTHOR



Pham Van Huong

Workplace: Academy of Cryptography Techniques, Vietnam.

Email: huongpv@actvn.edu.vn

Education: PhD in Information Technology.

Recent research direction: AI Application, Blockchain, Mobile and IoT.

Tên tác giả: **Phạm Văn Hương**

Cơ quan công tác: Học viện Kỹ thuật mật mã, Việt Nam

Email: huongpv@actvn.edu.vn

Quá trình đào tạo: Nhận bằng tiến sĩ ngành công nghệ thông tin năm 2015

Hướng nghiên cứu hiện nay: Công nghệ chuỗi khối, AI cho an toàn thông tin và AI trong phát triển phần mềm.



Nguyen Duc Tuyen

Workplace: Academy of Cryptography Techniques, Vietnam.

Email: tuyenngduc12@gmail.com

Education: Student, majoring in Information Technology

Recent research direction: Web Development and Blockchain Technology, Vietnam

Tên tác giả: **Nguyễn Đức Tuyền**

Cơ quan công tác: Học viện Kỹ thuật mật mã, Việt Nam

Email: huongpv@actvn.edu.vn

Quá trình đào tạo: Hiện là sinh viên năm cuối tại Học viện Kỹ thuật mật mã

Hướng nghiên cứu hiện nay: Phát triển ứng dụng Web, công nghệ chuỗi khối.



Dang Hai Long

Workplace: Academy of Cryptography Techniques, Vietnam

Email: hailong27102000@gmail.com

Education: Second-year master, majoring in Cybersecurity

Recent research direction: Information Technology, Cryptography and Information Security.

Tên tác giả: **Đặng Hải Long**

Cơ quan công tác: Học viện Kỹ thuật mật mã, Việt Nam

Email: huongpv@actvn.edu.vn

Quá trình đào tạo: Học viên cao học năm 2 tại Học viện Kỹ thuật mật mã

Hướng nghiên cứu hiện nay: An toàn và bảo mật thông tin.



Tran Quoc Toanh

Workplace: Academy of Public Administration and Governance, Vietnam.

Email: toanh.tranquoc@gmail.com

Education: Master of Cryptographic Engineering.

Recent research directions:

Applications of Artificial Intelligence (AI) and Digital Transformation in Public Administration.

Tên tác giả: **Trần Quốc Toanh**

Cơ quan công tác: Học viện Hành chính và Quản trị công, Việt Nam

Email: toanh.tranquoc@gmail.com

Quá trình đào tạo: Thạc sĩ Kỹ thuật mật mã

Hướng nghiên cứu hiện nay: Ứng dụng trí tuệ nhân tạo và Chuyển đổi số trong Quản lý hành chính công.



Nguyen Ngoc Tuyen

Workplace: Academy of Cryptography Techniques, Vietnam

Email: tuyenboy1234@gmail.com

Education: Five-year student, major in Information Technology

Recent research direction: Web application development, mobile application development, and backend systems.

Tên tác giả: **Nguyễn Ngọc Tuyền**

Cơ quan công tác: Học viện Kỹ thuật mật mã, Việt Nam

Email: huongpv@actvn.edu.vn

Quá trình đào tạo: Hiện là sinh viên năm cuối tại Học viện Kỹ thuật mật mã

Hướng nghiên cứu hiện nay: Phát triển ứng dụng Web, di động và hệ thống phụ trợ.