Date: 9th June-2026

MECHANISMS FOR ENSURING THE CONFIDENTIALITY, INTEGRITY, AND AUTHENTICITY OF USER DATA

Erkinov Shohjahon Sherali ogli
Axmatov Bekzod Nurali ogli
Kutlimurotov Abrorbek Hamid ogli
Ramazonova Marjona Ikrom kizi
Students of the Muhammad al-Khwarizm Technical University

Abstract: This study discusses mechanisms for ensuring the confidentiality, integrity, and authenticity of user data in modern information systems. A blockchain-based data protection model is proposed, integrating cryptographic techniques such as encryption, hash functions, digital signatures, and smart contracts. Sensitive data is encrypted using AES-256, while SHA-256 is used to ensure data integrity. Authentication and data authenticity are provided through RSA digital signatures. The proposed decentralized architecture ensures secure data storage, access control, and tamper resistance. The results show that the blockchain-based approach significantly improves data security compared to traditional centralized systems by preventing unauthorized access, modification, and forgery of information.

Keywords: Blockchain, information security, data protection, AES-256, SHA-256, RSA, smart contract, digital signature, confidentiality, integrity, authenticity, decentralization.

In modern information systems, ensuring the security of user information is one of the fundamental requirements of cybersecurity. User data often includes personal information, authentication credentials, financial records, healthcare information, and other sensitive digital assets. Unauthorized disclosure, modification, or forgery of such information may lead to serious security incidents and financial losses. Therefore, the proposed blockchain-based data protection system focuses on three primary security objectives: confidentiality, integrity, and authenticity.

Ensuring Data Confidentiality. Confidentiality refers to preventing unauthorized individuals from accessing sensitive information. In the proposed architecture, confidentiality is achieved through cryptographic encryption mechanisms and controlled access policies.

Before data is stored in the system, sensitive information is encrypted using the Advanced Encryption Standard (AES-256). Encrypted files are stored in off-chain storage systems such as IPFS, while only their cryptographic hash values and metadata are recorded on the blockchain ledger. This approach reduces storage overhead while preserving privacy and security.

The encryption process can be represented as follows:

[

Ciphertext = Encrypt(Plaintext, Key)]

where:

- Plaintext – original user data;
- Key – secret encryption key;
- Ciphertext – encrypted data.

An example of AES encryption implemented in Python is presented below.

```
from Crypto.Cipher import AES
from Crypto.Random import get_random_bytes
key = get_random_bytes(32)
cipher = AES.new(key, AES.MODE_EAX)
data = b"Confidential User Data"
ciphertext, tag = cipher.encrypt_and_digest(data)
print(ciphertext)
```

The encrypted data can only be accessed by authorized users possessing the corresponding decryption key. The result can be obtained as following Figure 2.2.

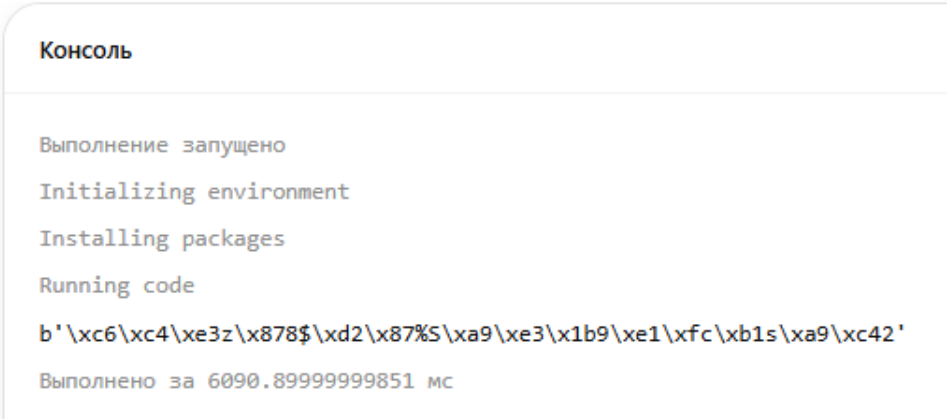

Figure 1.1. The encryption process

Ensuring Data Integrity. Data integrity ensures that information remains unchanged during storage and transmission. Blockchain technology provides integrity protection through cryptographic hash functions and immutable ledger structures. In the proposed system, every data object is processed using the SHA-256 hashing algorithm before being registered in the blockchain network. The hash generation process can be expressed as:

[

Hash = SHA256(Data)]

Even a minor modification of the original data produces a completely different hash value, making unauthorized alterations immediately detectable.

The following example demonstrates SHA-256 hash generation.

```
import hashlib
data = "User Personal Information"
hash_value = hashlib.sha256(data.encode()).hexdigest()
print(hash_value)
```

The generated hash value is stored in the blockchain ledger. Whenever a file is retrieved, its hash is recalculated and compared with the blockchain record. If the values differ, the system detects a potential integrity violation. The result can be seen as following Figure 1.2.

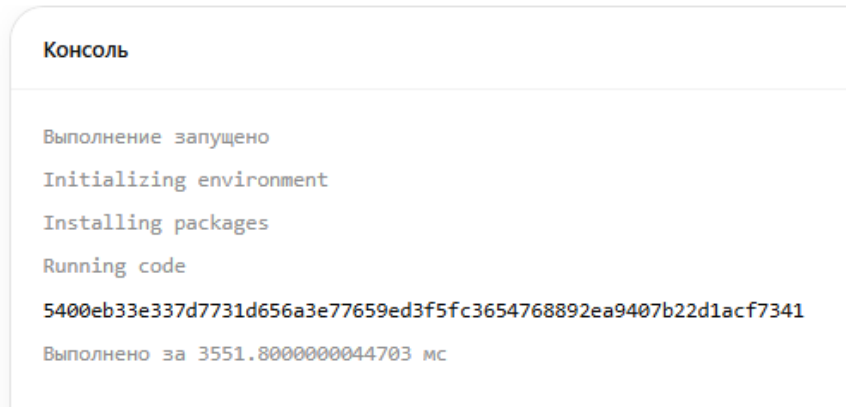

Figure 1.2. SHA-256 hash generation

Additionally, the immutable nature of blockchain prevents attackers from modifying previously confirmed records, thereby providing a reliable mechanism for data integrity verification.

Ensuring Data Authenticity. Authenticity guarantees that information originates from a legitimate source and has not been forged. In blockchain systems, authenticity is primarily achieved through digital signatures and decentralized identity verification mechanisms. Each transaction submitted to the blockchain network is digitally signed using the sender's private key. The signature can later be verified using the corresponding public key. The digital signature process is represented as:

[

Signature = Sign(Hash(Data), PrivateKey)]

The verification process is:

[

Verify(Signature, PublicKey)]

An example of digital signature generation using RSA cryptography is shown below.

```
from Crypto.PublicKey import RSA
from Crypto.Signature import pkcs1_15
from Crypto.Hash import SHA256
key = RSA.generate(2048)
message = b"Blockchain Transaction"
hash_obj = SHA256.new(message)
signature = pkcs1_15.new(key).sign(hash_obj)
print(signature)
```

The proposed architecture also incorporates blockchain-based digital identity management. During the authentication process, users are registered, verified, and assigned unique digital identities. Smart contracts automatically verify user permissions before granting access to protected resources. Figure 2.2 illustrates the blockchain-based digital identity authentication model used in the proposed system.

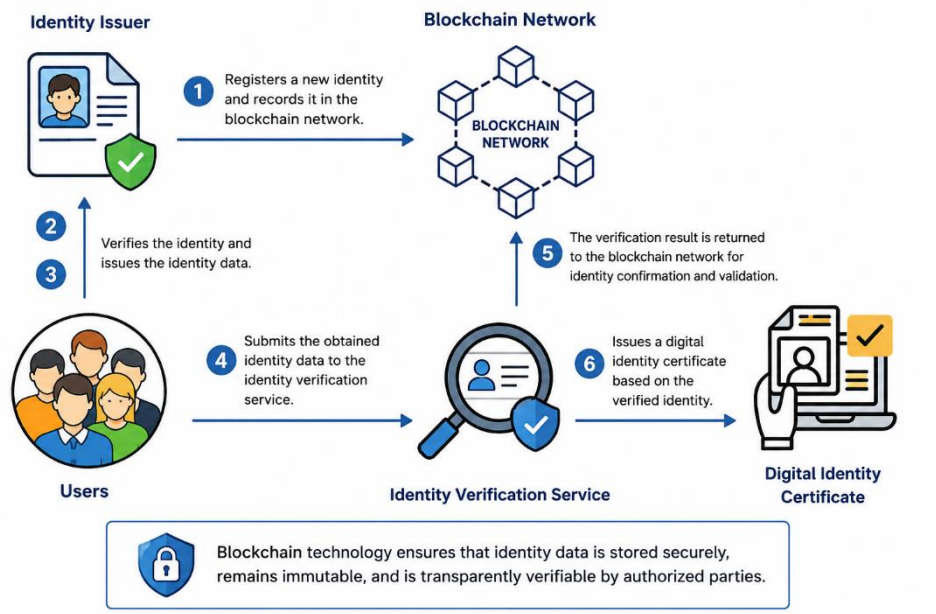

Figure 1.3. The blockchain-based digital identity authentication model

Integrated Security Mechanism. The combination of encryption, hashing, digital signatures, smart contracts, and distributed ledger technology creates a comprehensive security framework for user data protection.

Digital Signature for Data Authenticity. Authenticity is achieved through digital signatures generated using public-key cryptography. Example of RSA Digital Signature:

```
from Crypto.PublicKey import RSA
from Crypto.Signature import pkcs1_15
from Crypto.Hash import SHA256
key = RSA.generate(2048)
message = b"Blockchain Transaction"
hash_obj = SHA256.new(message)
signature = pkcs1_15.new(key).sign(hash_obj)
print(signature)
```

Digital signatures guarantee that data originates from a legitimate source and has not been forged.

```
Выполнение запущено
Initializing environment
Installing packages
Running code
b'exn\x18\xfe_)\x0e\xa0j\x97\xa3\x8f\x89]Z\x99H\x89\xf40<\x1c\x10\xb80Lg\xcb\xec\xc2\xda\xb8{\xa2J\xf7*\x8b\x1a1\xe8\x12\xf0\x05\x0f5\xf2\xff\xd0\xaa0pz\
x8a8\x84\x83\x077\xef1\xd6\x9e\x1e\xa8\xbbQ\x94\xaa\xe6\x17\nW\x1a\x17\x15Rm\xe3W\x8f\xc6\x90\xb1L\xe8\xb0n\xe3a\xa3\xc3\xd2\xda\x87g\xefK.\xd1b\xc8\xbb\
rf=>1\xa1U\xb1\xe5V\xb8\xb0\xc5F\xf4\x1f\x9fH)mn\x82\x8b\xe0\x8aO31\xfe\x0b\x83\xfd\xd7\xa4\xc6\xc9\xbf-
J%a\xe6\x15\x82\x0f\xc4\x9a\xe4\xf5\xcc\x01\xdc\x07\x7f\xf8\xe4\xc8\x04\x8a\xa9\xd3\x9b\xfa1\xf8?
\xbbM\r\x88\xe6`\xbd\xd4\xcd|\xe8I\xd2\r1@\xcf\x8e\x90W\xf2\n\x9eL\xb1\xd6G\x88\xf0\n\x9aCV\x07\xa9\xd9\xdd\xc4_3\xe1\xcf\xd6\xb2ul\xf3\x1d\xc0\x12\xcb]\
x05a\xcf\xfdwy@i\xe3\xe3\x1d\xa6\xd7\xeb\x0e7\x01}\xac\x1d\xdcI\xc0\x0f\rh\xe9\xa7a\xb3\xc1\x8b\x8c\x9e'
Выполнено за 5652.70000000298 мс
```

Figure 1.4. RSA Digital Signature result

Smart contracts automatically manage user permissions within the blockchain environment. Example Smart Contract (Solidity):

```
pragma solidity ^0.8.0;
contract AccessControl {
```

```
address public admin;
mapping(address => bool) public authorizedUsers;
constructor() {
admin = msg.sender;
}
function authorizeUser(address user) public {
require(msg.sender == admin);
authorizedUsers[user] = true;
}
function checkAccess(address user)
public
view
returns(bool)
{
return authorizedUsers[user];
}
}
```

This smart contract grants and verifies access permissions for blockchain participants. Table 1.1 summarizes the primary security mechanisms implemented in the proposed system.

Table 1.1.
Security Mechanisms Used in the Proposed System

Security Requirement	**Mechanism**	**Technology**
Confidentiality	Encryption	AES-256
Integrity	Hash Function	SHA-256
Authenticity	Digital Signature	RSA
Access Control	Smart Contract	Solidity
Authentication	Identity Verification	Blockchain DID
Auditability	Immutable Ledger	Hyperledger Fabric

The proposed blockchain-based architecture provides an integrated approach for protecting user data confidentiality, integrity, and authenticity. Encryption mechanisms

ensure confidentiality, cryptographic hash functions guarantee integrity, and digital signatures provide authenticity. Combined with smart contracts and decentralized identity management, these mechanisms significantly improve the overall security of information systems and create a trustworthy environment for managing sensitive user data.

Blockchain-Based Model for Protecting User Personal Data. To enhance the protection and security of user personal information, a conceptual blockchain-based security model is proposed. The experimental design and simulation results confirmed the effectiveness of the proposed model in ensuring secure management of sensitive information. The primary objective of the model is to provide reliable mechanisms for storing, managing, and sharing sensitive user data while preserving data confidentiality, integrity, authenticity, and immutability throughout the entire information lifecycle.

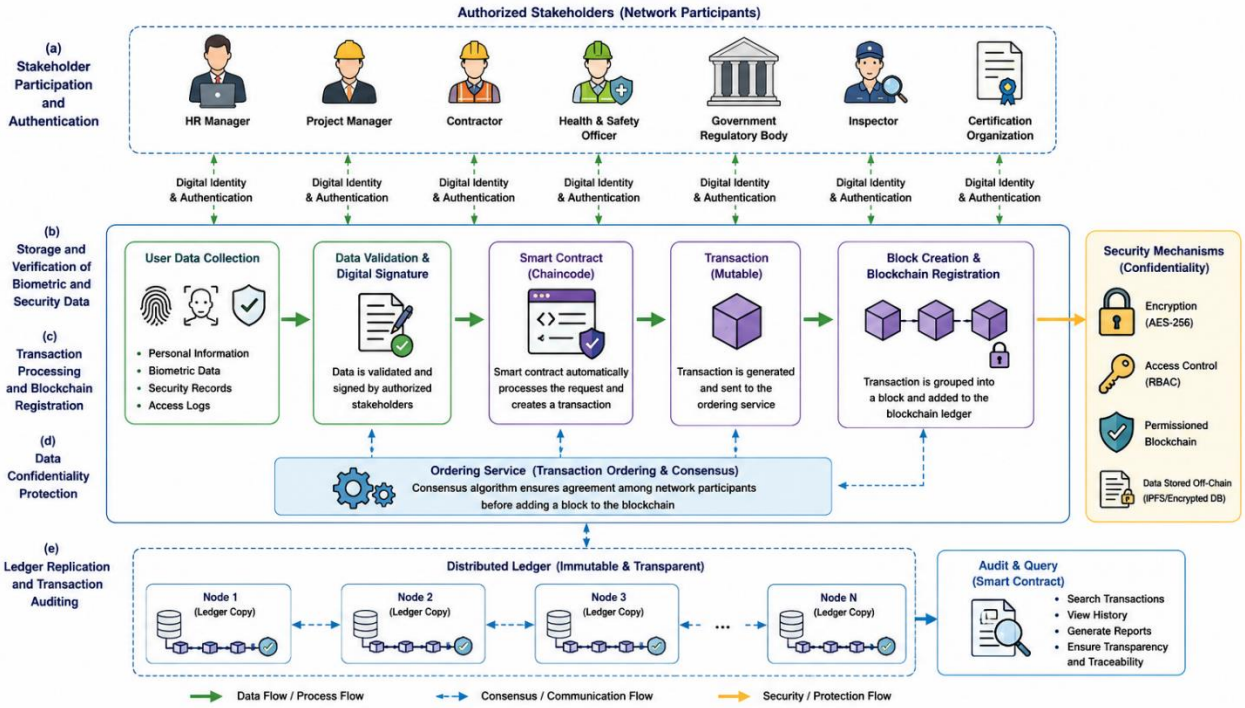

Figure 1.4. Conceptual Blockchain-Based Model for Protecting User Personal Data and Ensuring Information Security

The operational principles of the proposed model can be described as follows:

(a) Stakeholder Participation and Authentication. Various authorized participants are registered as members of the blockchain network, including system administrators, security officers, auditors, service providers, regulatory authorities, inspectors, and certification organizations. Each participant possesses a unique digital identity and is authorized to digitally sign and validate user-related information according to predefined access control policies.

(b) Storage and Verification of Biometric and Security Information. User biometric information, authentication records, access logs, and security-related data are securely stored within the blockchain ecosystem. Only authorized stakeholders are permitted to validate and approve such information through digital signatures. The validity of records is confirmed through the consensus mechanism implemented within the blockchain network, ensuring trust among all participating entities.

(c) Transaction Processing and Blockchain Registration. Whenever an authorized stakeholder requests access to user information or performs a security-related operation, a smart contract (chaincode) automatically converts the request into a blockchain transaction. The transaction is subsequently validated, grouped into a block, and appended to the blockchain ledger through the ordering service. Cryptographic protection mechanisms guarantee that recorded information cannot be modified after confirmation.

(d) Protection of Data Confidentiality. The proposed model utilizes consensus algorithms, cryptographic encryption, and access control policies to protect sensitive information from unauthorized disclosure. Only verified participants possessing appropriate permissions can access protected records. This mechanism significantly enhances data confidentiality and reduces the risk of insider threats and unauthorized access.

(e) Distributed Ledger Replication and Transaction Auditing. Each blockchain participant maintains a synchronized copy of the distributed ledger. Through smart contracts and blockchain query mechanisms, authorized entities can retrieve, verify, and analyze historical transactions when necessary. This capability improves transparency, traceability, accountability, and auditability across the entire system.

The proposed blockchain-based model provides a secure framework for managing sensitive user information by combining cryptographic protection, decentralized storage, consensus-based validation, digital signatures, and smart contract automation. As a result, the model significantly improves the confidentiality, integrity, authenticity, and reliability of user data compared to traditional centralized information systems.

Conclusion

In this research, a blockchain-based model for protecting user data was developed. The proposed system ensures confidentiality through AES-256 encryption, integrity through SHA-256 hashing, and authenticity through RSA digital signatures. Smart contracts are used to automatically manage access control and verify user permissions within the network. All transactions are recorded in an immutable distributed ledger, ensuring transparency and traceability. The findings indicate that the proposed approach is more secure, reliable, and efficient compared to traditional centralized systems. The model can be effectively applied in areas such as e-government, healthcare, and financial systems to enhance data security.

REFERENCES:

1. Blockchain Basics – A comprehensive introduction to blockchain concepts and technology.
2. Mastering Blockchain – Detailed explanation of blockchain architecture and applications.
3. Blockchain Revolution – Insights into how blockchain transforms business and society.
4. The Basics of Bitcoins and Blockchains – Overview of cryptocurrencies and blockchain systems.