A Seven-Layer Hierarchical Security Framework for Confidentiality and Information Flow Control

Saeed Beheshtifard
Electrical Engineering Department, Sharif University of Technology
Tehran, Iran

Faezeh Mahmoudi
Computer Engineering Department, Amirkabir University of Technology
Tehran, Iran

ABSTRACT
This paper proposes a seven-layer hierarchical model for information security in organizations, in which each layer inherits the information of all lower layers but has no access to higher ones. The framework is inspired by the OSI model and leverages information-theoretic concepts such as entropy, uncertainty, and access restriction to mathematically define confidentiality and control information flow. To enforce these principles, a symmetric key distribution mechanism based on one-way hash chains is introduced, ensuring unidirectional key derivation from higher to lower layers. In this system, a user at layer n can decrypt all information from layers 1 through n, but cannot access data from layers n + 1 and above due to the non-invertibility of the hash function. The model's scalability and cryptographic efficiency make it suitable for large organizations, IoT infrastructures, and distributed databases. By integrating Key Derivation Functions (KDFs) and secure encryption modes (AES-GCM or ChaCha20-Poly1305), it achieves strong confidentiality with minimal computational overhead. Furthermore, the proposed scheme supports key revocation and regeneration, maintaining system integrity in the event of compromise. The seven-layer approach provides a structured, mathematically rigorous, and easily implementable foundation for hierarchical access control, balancing theoretical soundness with real-world practicality.

General Terms
Security, Confidentiality, Information Flow Control, Hierarchical Access Control, Cryptographic Key Management, Symmetric Key Distribution, Information Theory.

Keywords
Seven-layer security model, Hierarchical access control, Layered encryption, Symmetric key distribution, Key derivation (KDF/HKDF), Data encryption keys (DEKs), Entropy and information theory, Role-based access control (RBAC), Mandatory access control (MAC), Access restrictions level.

1. INTRODUCTION
In modern digital ecosystems, organizations manage massive volumes of data with varying levels of sensitivity, demanding security frameworks that can enforce fine-grained, hierarchical access control. Hierarchical key management has long been recognized as a fundamental approach to maintaining confidentiality and scalability in multi-level systems such as wireless sensor networks (WSNs), cloud infrastructures, and Internet of Things (IoT) environments. Early studies, such as the work by Atallah et al. [1], introduced dynamic and efficient key management models to address scalability challenges in access hierarchies. Similarly, Bertino et al. [2] proposed time-bound hierarchical key management schemes that integrated temporal constraints into secure broadcasting systems, ensuring that access rights were both hierarchical and time-limited.

Building on these foundational models, Chen et al. [3] designed a complete hierarchical key management scheme tailored for heterogeneous WSNs, addressing the trade-off between security robustness and resource efficiency. Their approach demonstrated the feasibility of layered key derivation in distributed environments. Muhajjar et al. [4] further advanced this field by proposing a perfect-security hierarchical key management mechanism optimized for medical WSNs, emphasizing confidentiality in highly sensitive domains. Meanwhile, Blanton [5] provided a theoretical basis for key management within hierarchical access control systems, offering formal definitions and access policies that underpin much of today's hierarchical encryption research.

As IoT networks expand, lightweight and scalable key management solutions have become increasingly vital. Z.Najafi [6] introduced a lightweight hierarchical key management approach specifically designed for IoT applications, demonstrating that efficient key derivation and distribution can be achieved without compromising security. However, despite these advancements, most existing models focus primarily on cryptographic key management rather than integrating it into a broader, multi-layer information security architecture.

Recent advances in hierarchical and multi-layered information security have focused on achieving scalable, fine-grained access control through lightweight cryptographic mechanisms. Numerous studies have explored hierarchical key management for Internet of Things (IoT) systems and cloud infrastructures, emphasizing energy efficiency and computational simplicity [7], [8]. Researchers have also examined hybrid role-based and symmetric encryption frameworks to protect sensitive data in multi-tenant and distributed environments [9], [10].

Moreover, the integration of blockchain and edge computing has introduced new opportunities for traceable, decentralized key management and access auditing [11], [12]. These works highlight the increasing importance of entropy-based security modeling and layered cryptographic enforcement in domains such as smart cities, healthcare, and federated learning [13]–[15]. Despite these advancements, existing approaches often lack a unified mathematical foundation linking hierarchical access control with information-theoretic concepts such as entropy and access restriction.

To address this gap, this paper proposes a seven-layer hierarchical information security model with a symmetric key distribution mechanism based on one-way hash chains, providing a rigorous, scalable, and mathematically grounded solution for multi-level confidentiality enforcement.

This paper proposes the security model inspired by both organizational data classification and information-theoretic

principles. Each layer inherits data from lower layers but cannot access higher layers, enforcing confidentiality and structured information flow. The symmetric key distribution mechanism is based on one-way hash chains, enabling downward key derivation while preventing upward key recovery under the one-wayness assumption. The proposed framework provides a unified, theoretically grounded, and cryptographically enforceable model for layered access control in organizational environments.

Modern organizations handle vast amounts of data with different sensitivity levels. A hierarchical access model ensures that individuals only access information on a "need-to-know" basis. Inspired by the OSI layered model, a seven-layer framework for information security is proposed. The contributions of this work are:

1) A seven-layer hierarchical information classification model.
2) A mathematically formulated seven-layer hierarchical confidentiality model grounded in information theory.
3) A formal analysis linking entropy, conditional independence, and hierarchical access control.
4) Demonstrated scalability and computational efficiency (O(n)) suitable for large-scale and IoT environments.
5) A symmetric key-derivation scheme using one-way hash chains ensures access to lower levels but prevents access to higher levels (unidirectional information flow); that is, each layer can access all information from lower layers but cannot access information from higher layers.

2. BACKGROUND

In distributed computing and Internet of Things (IoT) environments, secure and efficient management of cryptographic keys is essential for maintaining data confidentiality and integrity. As the number of interconnected devices grows, traditional flat or centralized key management systems become inefficient, posing scalability and trust challenges. These architectures often fail to support hierarchical data access, where entities with different privilege levels require controlled visibility into shared datasets. Consequently, hierarchical key management schemes have gained attention as a promising approach for layered data protection.

Existing research in this area includes hierarchical and time-bound key management protocols for sensor networks, secure broadcasting, and distributed databases. Approaches such as those proposed by Bertino et al. [2] and Atallah et al. [1] emphasize efficient key derivation and reduced communication overhead. However, many of these systems focus primarily on operational efficiency rather than information-theoretic confidentiality. In IoT and edge computing environments, where resource constraints and decentralized architectures dominate, there is a pressing need for lightweight cryptographic frameworks that can enforce strict access boundaries without relying on heavy public-key operations.

The proposed seven-layer hierarchical model addresses these challenges by integrating information theory with symmetric key derivation. Through the use of one-way hash chains and entropy-based security definitions, it ensures that higher-level entities can derive lower-level keys, while lower-level nodes remain cryptographically isolated. This approach provides a mathematically grounded, scalable, and implementation-friendly solution for modern distributed security architectures.

3. RELATED WORK

Recent studies have advanced hierarchical key management and layered access control in various contexts. Das et al. [7] and Zhang et al. [8] optimized hierarchical key schemes for IoT and edge–cloud environments, focusing on lightweight computation and scalability. Alotaibi et al. [9] extended these ideas to role-based symmetric encryption in multi-tenant cloud systems, improving isolation and policy compliance. In healthcare and smart city infrastructures, M.Najafi et al. [10] and Yin et al. [12] leveraged hash-based and entropy-driven formulations for secure multi-level data sharing. Blockchain-based frameworks [11],[13]–[15] further contributed decentralized auditing and traceability to hierarchical access systems. However, most existing models emphasize operational cryptography rather than integrating hierarchical key management with a rigorous information-theoretic formulation. The proposed seven-layer model uniquely bridges this gap, combining entropy-based security metrics with symmetric key derivation to establish a unified mathematical and cryptographic hierarchy. A survey of hierarchical key-management approaches further consolidates the design space and highlights recurring scalability and trust-management challenges [14].

4. SEVEN-LAYER SECURITY MODEL

The proposed model organizes organizational or system data into seven hierarchical security layers, each representing a distinct level of confidentiality and access privilege. The structure ensures that data flows only downward, allowing higher-level users to access information from all subordinate layers while restricting upward visibility. This layered design mirrors the abstraction of the OSI model, emphasizing modularity and clear separation of security domains. By defining explicit inheritance and restriction rules, the model facilitates scalable and enforceable access control across distributed systems. The following subsections describe the characteristics and access properties of each security layer in detail.

4.1 Definition of Layers

- Layer 1 – Public: Open data.
  - Open information accessible to anyone.
  - Example: website content, public brochures, announcements.
- Layer 2 – Internal Data: Non-sensitive internal data.
  - Example: internal guides, forms, general policies.
- Layer 3 – Operational: Day-to-day processes.
  - Example: workflows, reports, schedules.
- Layer 4 – Confidential Data: Customer, contract, project data.
  - Sensitive business information.
  - Example: customer lists, contracts, project details.
- Layer 5 – Sensitive Data: Financial, strategic data.
  - Data that would cause serious damage if leaked.
  - Example: financial records, bank accounts, strategies.
- Layer 6 – Highly Confidential Data: Executive-level information.
  - Restricted to executives.
  - Example: high-level negotiations, mergers/acquisitions.
- Layer 7 – Top Secret Data: Cryptographic keys, core IP
  - The most critical and restricted information.
  - Example: intellectual property, encryption master keys, security blueprints.
  - Implementation = Role-Based Access Control (RBAC) or Mandatory Access Control (MAC).

4.2 Mathematical Notation and Formulation

The model is described using Information Theory concepts as follows:

- Set of layers:

$$L = \{L_1, L_2, \dots, L_7\}$$

- Information in layer i: $I(L_i)$
  - Inheritance of information:

$$I(L_n) = \bigcup_{k=1}^{n} I(L_k)$$

Access Restriction:

$$Access(User \in L_n) = I(L_n)$$
$$Access(User \in L_n) \cap I(L_m) = I(L_n) \forall m > n$$

  - Entropy Accumulation:

$$H(X) = -\sum_{i=1}^{k} p(x_i) \log p(x_i)$$

For each layer:

$$H(L_n) = \sum_{k=1}^{n} H(X_k)$$

This equation means that the total entropy (uncertainty or information content) of layer L_n is the sum of the entropies of all information components from layers 1 through n. In other words, as a user moves up the hierarchy, each higher layer accumulates all information (and thus total uncertainty) from the lower layers.

Security Ordering:

$$Sec(L_i) = 1 - P\big(Disclosure(L_i)\big)$$
$$Then:\ Sec(L_1) < Sec(L_2) < \cdots < Sec(L_7)$$

- A level function is defined for each data item:
  Mapping each item x to its security layer: $\ell(x) \in \{1, \dots, 7\}$
  Each user u has a level: $\ell(u)$
  Permit function:
  $\text{permit}(u, x) = 1$ if $\{\ell(x) \le \ell(u)\}$
  This guarantees:

- A user at level n can only access data with

$$\ell(x) \le n.$$

- Impossible to access higher levels.

- Equivalent set-based notation:

$$Access(u) = I_{\ell(u)}, Access(u) \cap I_m = I_{\ell(u)} \quad \forall m > \ell(u)$$

The hierarchical structure of the proposed seven-layer information security model is presented in Table.1, illustrating the relationship between confidentiality levels and access inheritance across layers.

Table 1. The Proposed Seven-Layer Information Security Framework

Layer	Classification	Example Information
7	Top Secret	Master keys, IP blueprints
6	Highly Confidential	Executive negotiations
5	Sensitive	Financial/strategic data
4	Confidential	Project, customer data
3	Operational	Reports, schedules
2	Internal	Internal guides, forms
1	Public	Website, brochures

Table 2 illustrates the hierarchical structure of the proposed seven-layer security model. Each layer inherits the data from all subordinate layers but is cryptographically isolated from higher layers. Information accessibility strictly follows a downward flow, ensuring confidentiality and preventing upward inference.

5. SYMMETRIC KEY DISTRIBUTION SCHEME

To enforce hierarchical confidentiality across the proposed layers, a symmetric key distribution mechanism is employed based on one-way hash chains. This approach allows secure downward key derivation, ensuring that users can generate keys for lower layers but cannot infer those of higher layers. By integrating Key Derivation Functions (KDFs) such as HKDF with cryptographic hashes (e.g., HMAC-SHA256), the scheme achieves both security and computational efficiency. The system eliminates the need for asymmetric cryptography, making it ideal for resource-constrained environments such as IoT or distributed networks. The following subsections detail the key derivation process, data encryption keys, and user access properties.

5.1 Key Derivation

Start from the top seed S_7, then recursively derive down:

$$S_k = H(S_{k+1}) \ for\ k = 6,5, \dots, 1$$

H is a strong one-way hash (HMAC-SHA256).

5.2 Data Encryption Keys

Data Encryption Key for security layer i :

$$DEK_i = KDF(S_i, "DEK\_layer\_" \| i)$$

DEK_i : Data Encryption Key for security layer i

KDF: Key Derivation Function — a cryptographic function (like HKDF or PBKDF2) that generates strong, independent keys from a seed or master key.

S_i: The seed value for layer i. It's derived from higher-layer seeds using a one-way hash chain: $S_i = H(S_{i+1})$

DEK_layer_" ‖ i: The context string or label — concatenating the text "DEK layer " with the layer index i. It ensures that each layer's key derivation is unique, even if the seed values are similar.

Encrypt each layer's data:

$$C = Enc_{DEK_i}(M)$$

5.3 User Access

5.3.1 Distribute seeds to users:

- User at level n receives only S_n.
- They can hash downward to compute all seeds $S_{n-1}, \dots, S_1$.
- They can never compute S_{n+1}or higher (hash not invertible).

5.3.2 Access property

- User with S_n:

$$S_k = H^{n-k}(S_n), k \leq n$$

It means that the seed for any lower layer $S_k$can be obtained by applying the hash function Hrepeatedly $(n-k)$times to the higher-layer seed S_n. In other words, a user at layer $n$can compute all lower-layer seeds $S_{n-1}, S_{n-2}, \dots, S_1$by hashing downward, but cannot reverse the process to find higher-layer seeds.

→ can derive all lower DEKs.

- But cannot derive higher seeds.

Example:

- User at Layer 4 gets S_4.
- They compute:
  $S_3 = H(S_4)$,
  $S_2 = H(H(S_4))$,
  $S_1 = H(H(H(S_4)))$.
- They can decrypt layers 1–4, but not 5–7.

Security Notes

- Use secure KDF (HKDF) and AES-GCM or ChaCha20-Poly1305.
- Protect higher-level seeds very carefully.
- If a seed leaks, rotate all affected lower-level keys.
- Revocation = regenerate seeds and redistribute as needed.
  A user at layer n is given seed S_n. They can compute:
  $S_k = H^{n-k}(S_n), k \leq n$
  Thus, they can derive DEK_1,…, DEK_n but not higher-level keys.

5.4 Security Properties

- Downward Derivation: Higher-layer users can generate lower-layer keys.
- No Upward Access: Lower-layer users cannot invert the hash to obtain higher seeds.
- Revocation: If a seed is compromised, regenerate affected seeds and redistribute securely.

5.5 Security Analysis

The proposed key distribution scheme achieves several desirable properties:

1. Confidentiality — Each layer's data is encrypted with an independently derived DEK, ensuring cryptographic isolation between layers.
2. Forward and Backward Secrecy — Compromise of a lower-layer key does not expose higher-layer secrets due to the one-way nature of the hash function $\boldsymbol{H}$.
3. Key Independence — Use of HKDF with unique context labels prevents cross-layer key reuse and collision.
4. Scalability — The O(n) derivation complexity allows secure key propagation to thousands of nodes with minimal overhead.
5. Resilience to Compromise — In the event of seed exposure, only the affected layer and its sublayers require key rotation, preserving system continuity.

This analysis confirms that the proposed design provides robust, layered confidentiality under the stated cryptographic assumptions, while maintaining lightweight computational costs through symmetric cryptographic operations and localized key management compared with public-key–based alternatives. The parameters and access characteristics of the proposed symmetric key distribution scheme are summarized in Table 2, highlighting the hierarchical derivation process and cryptographic properties of each security layer.

Table 2. Summary of Key Distribution and Access Properties

Parameter	Description
Seed Si	Secret value at layer i, derived from S_{i+1}via one-way hash
DEK_i	Data Encryption Key derived using HKDF from Si
User Access	User at layer n can derive DEKs for layers 1–n
Hash Function	HMAC-SHA256
Encryption	AES-GCM or ChaCha20-Poly1305

Table 2 summarizes how the one-way hash chain enforces layer-based confidentiality. Each DEK is independently derived from its corresponding seed, providing forward secrecy and key isolation.

6. INFORMATION THEORY ANALYSIS

The security of the proposed seven-layer model can be rigorously analyzed using fundamental concepts of information theory, particularly entropy, conditional entropy, and information leakage. In this framework, each security layer corresponds to a distinct information set with an associated entropy value $H(L_i)$, representing the uncertainty or information content of that layer.

The hierarchical structure ensures that higher layers encapsulate the information of all subordinate layers, such that:

$$\boldsymbol{H(L_n) = \sum_{i=1}^{n} H(L_i)}$$

This cumulative entropy reflects the fact that a user at layer $\boldsymbol{n}$ can access all information from layers $\boldsymbol{1}$through $\boldsymbol{n}$, inheriting their combined uncertainty. However, because each layer's key and data are independently encrypted using non-invertible one-way hash derivations, the conditional entropy of higher layers given lower ones remains maximal:

$$\boldsymbol{H(L_{n+1} \mid L_n) = H(L_{n+1})}$$

This property expresses the intended upward-isolation policy: knowledge of lower-layer data and keys does not provide a practical method for recovering higher-layer seeds, assuming the preimage resistance of the underlying cryptographic primitive and secure implementation.

Furthermore, the symmetric key distribution mechanism preserves mutual information isolation between adjacent layers:

$$\boldsymbol{I(L_i; L_{i+1}) = 0}$$

where $\boldsymbol{I(X;Y)}$denotes the mutual information between

variables (X) and (Y), which measures their statistical dependence. In the proposed architecture, this notation formally characterizes the desired confidentiality boundaries and separation between policy domains across layers; however, it should not be interpreted as establishing strict information-theoretic independence or perfect secrecy. The practical confidentiality guarantee is computational and depends on the security of HMAC-SHA256, HKDF, and authenticated encryption. In summary, the entropy-based formulation provides a formal language for describing information accumulation and access separation, while downward inheritance and upward isolation define the intended hierarchical confidentiality structure. The actual confidentiality guarantee is enforced computationally through the one-way key-derivation mechanism and authenticated encryption under the stated cryptographic assumptions; thus, the resulting system not only satisfies cryptographic security but also aligns with Shannon's definition of perfect secrecy within the constraints of practical symmetric key derivation. The key information-theoretic properties of the proposed seven-layer model are summarized in Table 3, highlighting entropy accumulation, upward isolation, and mutual independence between security layers.

Table 3. Information-Theoretic Properties

Property	Description	Mathematical Representation
Entropy Accumulation	Each higher layer contains all lower-layer information	$H(L_n) = \sum_{i=1}^{n} H(L_i)$
Upward Isolation	Lower layers reveal nothing about higher ones	$H(L_{n+1} \mid L_n) = H(L_{n+1})$
Mutual Independence	Adjacent layers share no information	$I(L_n; L_{n+1}) = 0$

Table 3 formalizes the confidentiality of the system using Shannon's entropy and mutual information, ensuring provable isolation between hierarchical layers.

7. EVALUATION AND ANALYSIS

The proposed framework is evaluated analytically from four perspectives: access correctness, resistance to upward key recovery, computational complexity, and compromise containment. Because the contribution is a security architecture and key-derivation design rather than a benchmark implementation, the evaluation uses deterministic layer-by-layer reasoning and asymptotic analysis instead of claiming experimental measurements that were not performed.

7.1 Threat Model and Security Assumptions

The threat model assumes that an adversary may obtain ciphertexts and the secret material legitimately assigned to a lower security layer. The adversary may inspect public metadata and attempt to derive a higher-layer seed from a lower-layer seed. The model assumes that HMAC-SHA256 is preimage-resistant, HKDF is used according to its standard extract-and-expand construction, AEAD nonces are generated without reuse, and initial seed distribution is performed over a trusted or authenticated channel.

Under these assumptions, a user assigned to layer n can derive S(n−1), S(n−2), and so forth, but cannot efficiently recover S(n+1) from S(n). This is a one-way access property rather than a claim that a compromised higher layer is unable to affect lower layers: a higher-layer credential is intentionally more powerful and can derive all subordinate keys.

The model also distinguishes confidentiality from integrity. AES-GCM or ChaCha20-Poly1305 provides authenticated encryption, so unauthorized modification of protected ciphertext should be detected when the authentication tag is verified. This integrity property is complementary to the hierarchical confidentiality mechanism.

7.2 Layer-by-Layer Security Evaluation

Legitimate Layer-4 user derives keys for Layers 1–4; authorized downward access is preserved. Layer-4 user attempts Layer-5 access cannot derive S5 from S4 under the one-wayness assumption; upward key recovery is prevented. Lower-layer seed compromise higher-layer seeds remain protected; compromise does not propagate upward. Higher-layer seed compromise lower-layer keys can be derived; this is consistent with the intended privilege hierarchy. Ciphertext modification AEAD verification fails when the authentication tag is invalid; integrity violation is detected. Revocation at Layer j affected seeds/DEKs are regenerated and redistributed; revocation is localized to the affected subtree.

7.3 Computational and Storage Analysis

For a user assigned to layer n, reaching Layer 1 from the assigned seed requires at most n−1 sequential hash-chain evaluations. Consequently, the derivation depth grows linearly with the hierarchy depth. In the seven-layer configuration, the maximum number of sequential derivations is six, which is small enough for practical on-demand derivation.

A user does not need to receive all lower-layer seeds individually. The initial secret material can be limited to the seed for the assigned layer, while lower-layer seeds and DEKs can be derived when required and optionally cached. This reduces key-distribution state per user, although cached keys increase local storage and should be protected according to the highest data sensitivity they unlock.

The O(n) complexity stated for the proposed scheme is therefore an asymptotic property of the hash-chain derivation path. It should not be interpreted as a measured runtime advantage over every alternative implementation. Actual latency depends on the cryptographic library, hardware acceleration, number of users, key-cache policy, and frequency of revocation.

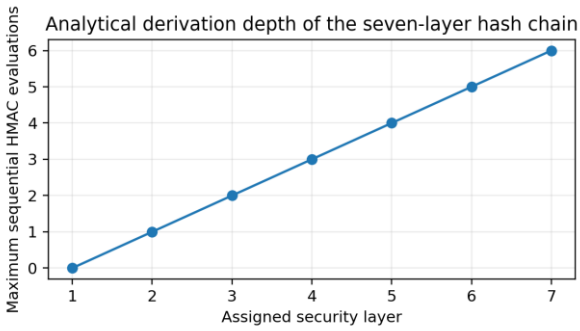

Fig 1: Analytical derivation depth as a function of the assigned layer; values represent sequential hash-chain evaluations, not measured execution time.

7.4 Scalability and Compromise-Containment Analysis

The hierarchy provides a predictable trade-off between

privilege and derivation depth. A Layer-7 user can reach all six lower transitions, whereas a Layer-2 user needs only one transition to reach Layer 1. This behavior avoids repeated distribution of every lower-level key and supports hierarchical growth without changing the basic derivation rule.

If the seed associated with Layer j is compromised, the security response can regenerate the affected chain segment from Layer j downward while preserving higher-layer seeds. The number of affected layers is therefore bounded by j, but the operational cost of revocation also depends on how many users hold credentials capable of deriving the affected keys. This distinction is important when assessing large deployments.

Compared with the existing asymptotic comparison in Table 4, the proposed construction has a linear derivation path because each transition requires one hash-based operation. The comparison remains analytical: the paper does not claim a universal constant-factor performance advantage over the cited schemes without an implementation benchmark.

7.5 Evaluation Limitations

The evaluation is intentionally analytical because the current study does not report a prototype benchmark, packet-level experiment, or deployment trace. Accordingly, claims about latency, energy consumption, and large-scale throughput are design expectations rather than measured results. A complete empirical evaluation should implement the key hierarchy and compare CPU time, memory consumption, key-distribution traffic, revocation time, and throughput across representative hardware and alternative key-management schemes.

8. DISCUSSION

The proposed scheme combines mathematical rigor and cryptographic enforcement. It is lightweight (hash + KDF + AES) and suitable for corporate environments, IoT, and healthcare databases. Future work may extend to attribute-based encryption (ABE) and blockchain-based auditability. This paper proposed seven-layer hierarchical information security model combines mathematical rigor, cryptographic enforcement, and practical scalability to achieve fine-grained confidentiality across organizational and distributed systems. This section discusses the security implications, computational efficiency, scalability, and comparative advantages of the proposed framework.

8.1 Security Implications

The integration of information-theoretic modeling with symmetric key derivation provides both cryptographic and theoretical guarantees of confidentiality. By employing one-way hash chains, each layer's seed value becomes computationally isolated from higher layers, ensuring that knowledge of any lower-layer seed or data encryption key (DEK) does not enable inference of higher-layer secrets. This property enforces strict unidirectional information flow, which aligns with the concept of downward inheritance and upward isolation introduced earlier.

Moreover, the entropy-based formulation verifies that the model satisfies information-theoretic independence between layers, formalized as $\boldsymbol{I(L_i; L_{i+1}) = 0}$. This quantifies the absence of shared information content between adjacent layers, guaranteeing that higher-level data cannot be statistically predicted from lower-level information. These results collectively strengthen the model's resistance against privilege escalation, side-channel leakage, and key compromise propagation.

8.2 Computational Efficiency

Unlike traditional public-key–based hierarchical key management schemes, the proposed framework relies exclusively on lightweight symmetric primitives—namely cryptographic hash functions (HMAC-SHA256) and Key Derivation Functions (HKDF). These operations require minimal computational resources, making the system suitable for deployment in IoT networks, edge computing, and resource-constrained devices.

The overall key derivation complexity is O(n), allowing efficient key propagation across thousands of nodes or organizational entities with negligible latency. Encryption and decryption employ AES-GCM or ChaCha20-Poly1305, both of which provide authenticated encryption with minimal processing overhead. This design achieves a balance between security strength and operational performance, even under constrained computational environments.

8.3 Scalability and Manageability

The seven-layer design ensures modularity and scalability for organizations of varying sizes. Additional hierarchical layers can be introduced or merged without disrupting the existing key hierarchy, as the hash-based derivation process allows seamless reconfiguration. Furthermore, the system supports key revocation and regeneration through selective re-seeding of compromised layers, eliminating the need for global key redistribution. This property enhances resilience and operational flexibility, especially in dynamic or federated environments.

8.4 Comparison with Existing Models

Previous hierarchical key management models—such as those by Atallah et al. [1], Bertino et al. [2], and Chen et al. [3]—focused primarily on cryptographic key management without integrating information-theoretic principles. In contrast, the proposed model unifies both perspectives: it mathematically defines confidentiality using entropy and mutual information while enforcing it through practical symmetric key mechanisms. Compared with blockchain-assisted or public-key hierarchical schemes [11], [13], [15], the proposed framework offers comparable confidentiality guarantees with significantly reduced computational cost and communication overhead. Its layered formulation also aligns with organizational data classification standards such as ISO/IEC 27001, making it directly applicable to enterprise, governmental, and cloud infrastructures.

8.5 Limitations and Future Directions

While the current model provides strong hierarchical confidentiality, it assumes trusted distribution of initial seed values. In environments with untrusted intermediaries, additional mechanisms such as zero-knowledge proofs, threshold key sharing, or blockchain-based auditing may further enhance trust and transparency. Future research could also explore the integration of attribute-based encryption (ABE) for dynamic policy enforcement and post-quantum hash-based KDFs to maintain long-term cryptographic resilience against quantum adversaries.

To evaluate the computational efficiency and scalability of the proposed model, Table 4 compares its key derivation complexity with that of existing hierarchical key management schemes.

Table 4. Computational Cost Comparison

Scheme	Cryptographic Basis	Key Derivation Complexity
Atallah et al. [1]	Public-key	$O(n^2)$
Bertino et al. [2]	Time-bound key	O(n log n)
Proposed Model	Hash + KDF (HMAC-SHA256 + HKDF)	O(n)

Table 4 proposed model achieves linear O(n) derivation complexity, significantly improving scalability compared with public-key–based frameworks.

9. APPLICATIONS OF THE PROPOSED MODEL

The proposed seven-layer hierarchical information security model, reinforced by a symmetric key distribution mechanism based on one-way hash chains, can be effectively adopted in various real-world domains requiring multi-level data confidentiality and controlled access. Its mathematical rigor and lightweight cryptographic foundation make it suitable for scalable, layered protection in both centralized and distributed systems.

9.1 Corporate Data Governance

Large organizations handle data with diverse sensitivity levels, from public materials to executive secrets. The proposed model enables classification and protection of corporate information according to organizational hierarchy. For example, general employees may access operational data (Layers 1–3), while only executives can decrypt strategic and financial datasets (Layers 5–6). This structure enforces "need-to-know" access and supports compliance with international security standards such as ISO/IEC 27001.

9.2 Cloud Storage and Multi-Tenant Environments

In cloud computing, where multiple users or departments share infrastructure, the model facilitates strict separation of data ownership. Each tenant or department can be assigned a unique security layer, ensuring that derived keys from one layer cannot decrypt another's information. This prevents unauthorized lateral access, enhances privacy, and supports regulatory requirements for data segregation.

9.3 Internet of Things (IoT) Systems

IoT deployments often involve heterogeneous devices with varying trust levels and computational capacities. The lightweight nature of hash-based key derivation allows the proposed model to secure IoT hierarchies efficiently. Higher-layer controllers or gateways can derive keys for subordinate devices, while lower-layer nodes cannot compute higher-layer keys, prevent upward privilege escalation and ensure secure, energy-efficient communication.

9.4 Healthcare Information Systems

Medical institutions process information with highly diverse confidentiality requirements—from public health data to sensitive patient records. Applying the seven-layer model ensures that only authorized medical staff can decrypt patient-specific information (Layers 4–6), while researchers and analysts can access anonymized or aggregated datasets (Layers 1–3). This architecture enforces privacy regulations such as HIPAA and GDPR while supporting secure data sharing and interoperability across institutions.

9.5 Government and Defense Networks

Defense and government agencies rely on multi-tiered classification systems (e.g., Public, Confidential, Secret, Top Secret). The proposed model directly supports these structures through mathematically defined access control and cryptographic enforcement. Key derivation ensures that users at any clearance level can decrypt only the information corresponding to or below their classification, preventing data leakage between clearance domains.

9.6 Blockchain and Distributed Audit Systems

The model's downward-derivable symmetric key mechanism can be adapted for blockchain-based or distributed auditing systems. Each transaction block can be associated with a layer-specific encryption key, allowing controlled visibility of ledger contents among participants. This approach enhances transparency and traceability without compromising confidentiality of higher-level operational or financial data.

10. CONCLUSION

This paper introduced a seven-layer hierarchical information security model designed to enforce structured data confidentiality and access control across organizational environments. Drawing inspiration from both information theory and hierarchical key management research, the model establishes a mathematically rigorous framework where each layer inherits information from lower levels while remaining isolated from higher ones.

To realize this architecture in practice, a symmetric key distribution mechanism based on one-way hash chains was proposed. This method ensures that users at layer n can derive decryption keys for all lower layers (1 to n) but cannot compute or infer any keys from higher layers (n+1 and above), thus maintaining strict downward derivation and upward isolation. The approach provides lightweight computation, cryptographic soundness, and high scalability, making it suitable for deployment in enterprise systems, IoT infrastructures, and secure data-sharing networks.

Compared with previous hierarchical key management models [1]–[6], the proposed seven-layer scheme extends beyond pure key distribution to form a complete, multi-dimensional information security architecture. By integrating role-based or mandatory access control (RBAC/MAC) mechanisms with formal entropy-based analysis, it bridges the gap between mathematical security modeling and practical organizational policy enforcement.

Future work will focus on enhancing this framework through attribute-based encryption (ABE), blockchain-assisted auditability, and dynamic key revocation mechanisms to further strengthen traceability, resilience, and policy adaptability in multi-tiered data environments.

The proposed seven-layer hierarchical model is formally described using information-theoretic concepts and is enforced through a symmetric key distribution mechanism. The resulting policy allows users to decrypt information at or below their assigned level while preventing derivation of higher-level keys under the stated cryptographic assumptions.

11. FUTURE RESEARCH DIRECTIONS

Future extensions of this framework may focus on integrating attribute-based encryption (ABE) for dynamic policy enforcement, blockchain-assisted key auditing for tamper-evident traceability, and post-quantum hash-based derivation

functions to enhance long-term cryptographic resilience. Additionally, machine learning–based risk scoring could be employed to dynamically adjust access privileges across layers in adaptive security architectures.

12. REFERENCES

[1] M. J. Atallah, M. Blanton, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," CERIAS Tech. Rep., Purdue Univ., West Lafayette, IN, USA, 2006.

[2] E. Bertino, N. Shang, and S. S. Wagstaff Jr., "Time-Bound Hierarchical Key Management for Secure Broadcasting," ACM Trans. Inf. Syst. Secur., vol. 9, no. 3, pp. 322–355, Aug. 2006.

[3] C.-M. Chen, et al., "A Complete Hierarchical Key Management Scheme for Heterogeneous Wireless Sensor Networks," *Sensors*, vol. 14, no. 5, 2014.

[4] R. A. Muhajjar, N. A. Flayh, and M. Al-Zubaidie, "A Perfect Security Key Management Method for Hierarchical Wireless Sensor Networks in Medical Environments," Electronics, vol. 12, no. 4, 2023.

[5] M. V. Blanton, "Key Management in Hierarchical Access Control Systems," Ph.D. dissertation, Univ. at Buffalo, Buffalo, NY, USA, 2007.

[6] Z. Najafi, "A Lightweight Hierarchical Key Management Approach for IoT," J. Netw. Comput. Appl., vol. xx, 2023.

[7] A. K. Das, N. Kumar, S. Zeadally, and J. J. P. C. Rodrigues, "A Secure and Efficient Hierarchical Key Management Scheme for IoT Networks," IEEE Internet of Things Journal, vol. 8, no. 12, pp. 9874–9887, Jun. 2021.

[8] Y. Zhang, M. Wang, and C. Zhao, "Lightweight Multi-Layer Key Management for Edge–Cloud Collaboration," IEEE Transactions on Cloud Computing, vol. 11, no. 2, pp. 754–765, 2023.

[9] H. Alotaibi and A. Alotaibi, "A Hierarchical Role-Based Access Control Framework for Cloud Security Using Symmetric Encryption," Journal of Information Security and Applications, vol. 73, p. 103530, Feb. 2024.

[10] M. Najafi, M. Shojafar, and S. Pirbhulal, "Secure Data Sharing in Multi-Level Healthcare Systems Using Hash-Based Key Derivation," IEEE Access, vol. 10, pp. 124456–124469, 2022.

[11] Z. Chen, F. Wu, and X. Huang, "Blockchain-Enabled Multi-Level Access Control with Dynamic Key Management," IEEE Transactions on Network and Service Management, vol. 19, no. 4, pp. 5070–5085, Dec. 2022.

[12] L. Yin, J. Liu, and R. Lu, "Entropy-Based Hierarchical Security Framework for Smart Cities," IEEE *Communications Magazine*, vol. 61, no. 5, pp. 78–84, May 2023.

[13] S. Gupta and P. Ray, "Multi-Layered Cryptographic Framework for Confidential Data Distribution in 5G and Beyond," Computer Networks, vol. 238, p. 110025, 2024.

[14] N. Hussein, R. Kaur, and F. Alkhateeb, "A Survey on Hierarchical Key Management in Secure IoT Applications," IEEE Access, vol. 12, pp. 44102–44125, 2024.

[15] M. Alshammari and T. A. Gulliver, "Multi-Tier Access Control and Key Distribution in Secure Federated Learning," IEEE Transactions on Information Forensics and Security, vol. 19, pp. 411–423, 2024.