The NFT Transition for Certificates of Mobile Network

Dowon Kim[1], Seongmin Park[1], Daeun Kim[1], Ilsun You[2]*

[1] Korea Internet and Security Agency, Republic of Korea
[2] Kookmin University, Republic of Korea
kimdw@kisa.or.kr, smpark@kisa.or.kr, whale53@kisa.or.kr, isyou@kookmin.ac.kr

Abstract

Fifth-generation (5G) systems favor elasticity, disaggregation, and zero-trust operations, yet certificate management for TLS/mTLS across Open RAN and the 5G Service-Based Architecture (SBA) still relies on CA-centric PKI with X.509, creating trust concentration and revocation latency. We present NFT-Cert-MN, an NFT-backed certificate framework that remains wire-compatible with TLS/mTLS while decentralizing issuance and lifecycle control via smart contracts. Certificates are represented as non-transferable, account-bound NFTs bound to an NF/operator identity, and lifecycle-control keys are separated from TLS authentication keys. A Trusted Authority (TA) bootstraps identity and signs certificate metadata; on-chain policies govern minting, renewal, suspension, revocation, and key rotation via burn-and-reissue. We articulate three deployment profiles (public mainnet/L2, permissioned consortium, and private operator/regulator) and show how site-local full-node gateways meet telecom latency and availability constraints. For TLS 1.3, we state required properties (entity authentication, channel security, and revocation freshness) and argue that standard TLS guarantees are preserved under standard cryptographic and ledger-finality assumptions when the peer's TLS public key and token status are validated from a finalized ledger view. Experiments show a prototype mint cost of 488,501 gas (~$0.0001–$4.21) and mTLS authentication workload reductions of 33–50% in signature verifications, with no online OCSP round trips when status is read locally.

Keywords: 5G Security, Open RAN, X.509, NFT-based Certificates, mTLS

1 Introduction

Mobile networks have evolved from vertically integrated stacks to cloud-native, disaggregated systems. Open Radio Access Network (Open RAN) introduces vendor-neutral interfaces and virtualization, and the 5G Service-Based Architecture (SBA) decomposes the core into microservice-style Network Functions (NFs) that communicate over RESTful APIs. These designs improve agility and scale but broaden the attack surface across RAN and core interfaces.

Current deployments secure NF-to-NF and RAN interfaces with Transport Layer Security (TLS) and Mutual TLS (mTLS), authenticating peers via X.509 certificates issued by Certificate Authorities (CAs). While cryptographically sound, CA-centric Public Key Infrastructure (PKI) struggles with (I) single points of failure and trust concentration, (II) slow and uneven revocation (Certificate Revocation List (CRL)/OCSP caching, "must-staple" gaps), and (III) operational friction in rapidly scaling environments (e.g., elastic NF onboarding). These pain points motivate Decentralized Public Key Infrastructure (DPKI) approaches that shift auditability and state publication to blockchains.

The remainder of this paper is structured as follows: Section 2 surveys Open RAN/SBA security and X.509 operations, clarifying revocation, auditability, and scale constraints. Section 3 details the NFT-Cert-MN management system and data structures (on-chain state vs. off-chain metadata), lifecycle FSM, and issuer/subject roles including key-semantics. Section 4 reports results and analysis, including candidate L1/L2 choices, security posture, measured enrollment time/cost, deployment-model assumptions, and a formal workload model for mTLS under X.509 vs. NFT-Cert-MN. Section 5 discusses operational considerations—key management, NFT standards, metadata integrity, smart-contract security, interoperability with X.509/CT/DANE, privacy/compliance, and availability/liveness—and situates NFT-Cert-MN relative to prior work. Section 6 concludes with insights into future research directions and potential improvements in NFT-based certificate management for next-generation mobile networks.

Contributions. This work introduces NFT-Cert-MN, an NFT-backed certificate framework that:

1. Defines certificate NFTs as non-transferable, account-bound credentials and specifies key roles (TA issuance keys, NFT lifecycle-control keys, and TLS authentication keys) with explicit recovery and rotation semantics [1].
2. Integrates with telecom stacks, mapping to Open RAN and 5G SBA flows that already require TLS/mTLS, enabling a drop-in migration path.
3. Automates lifecycle (issue, renew, suspend, revoke, expire) with event logs that support Certificate Transparency (CT)-like monitoring.
4. Optimizes cost and latency by storing only essential state on-chain and anchoring off-chain metadata with a cryptographic hash.

*Corresponding Author: Ilsun You; Email: isyou@kookmin.ac.kr
DOI: https://doi.org/10.70003/160792642026052703004

5. Articulates realistic deployment profiles (public mainnet/L2, permissioned consortium, and private operator/regulator) and maps validators, governance, SLA, and data-sovereignty considerations to each [2-4].
6. Provides an explicit security model for TLS 1.3 + NFT-Cert-MN and argues preservation of authenticated-channel guarantees under standard cryptographic and ledger-finality assumptions [5-6].

2 Background

2.1 Modern Mobile Network Architecture

Open Radio Access Network (Open RAN) decomposes the radio stack into the Open Radio Unit (O-RU), Open Distributed Unit (O-DU), Open Central Unit (O-CU), and RAN Intelligent Controller (RIC), improving modularity and multi-vendor interoperability [7], as illustrated in Figure 1. Table 1 summarizes the key Open RAN unit components.

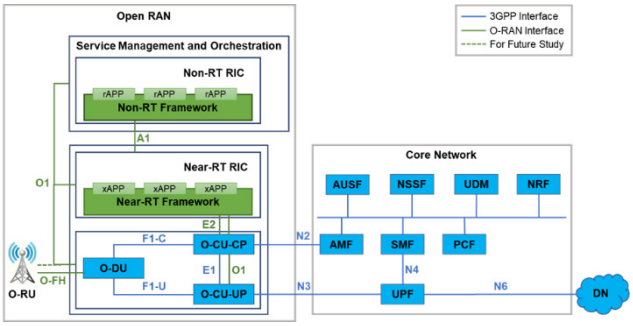

Figure 1. Open RAN and 5G core network architecture

Table 1. Open RAN unit components

Component	Description
Open Radio Unit (O-RU)	Responsible for transmitting and receiving radio signals. It connects directly to the antenna and handles physical signal processing.
Open Distributed Unit (O-DU)	Processes the lower layers of the radio protocol stack and handles signal processing for data received from the O-RU.
Open Central Unit (O-CU)	Manages the upper layers of the wireless network, handling user data and control information routing.
RAN Intelligent Controller (RIC)	Automates and optimizes RAN operations. It is categorized into non-real-time (non-RT RIC), which manages AI/ML based optimization and policy control, and near-real-time (near-RT RIC), which provides network adjustments on a millisecond-to-second scale.

In the core, the Service-Based Architecture (SBA) defines standalone NFs—e.g., Access and Mobility Management Function (AMF), Session Management Function (SMF), Policy Control Function (PCF), User Plane Function (UPF), Authentication Server Function (AUSF), Network Slice Selection Function (NSSF), Unified Data Management (UDM), and Network Repository Function (NRF). NFs discover each other via the NRF and interact through authenticated, authorized REST APIs [8]. Table 2 describes the primary 5G core network functions.

Table 2. 5G core network functions

Network function	Description
Access and Mobility Management Function (AMF)	Manages UE (User Equipment) connectivity and mobility. It handles initial access requests, forwards authentication requests, controls handovers, and supports network slice selection.
Session Management Function (SMF)	Manages data traffic paths, including PDU session establishment/ modification/release, IP address allocation, QoS policy enforcement, and interaction with the UPF.
Policy Control Function (PCF)	Defines and manages network policies and rules. It applies QoS and charging policies to optimize network resource management and service delivery.
User Plane Function (UPF)	Forwards user data, manages traffic, and supports local breakout. It integrates with Multi-access Edge Computing (MEC) to enable low-latency services.
Authentication Server Function (AUSF)	Processes UE authentication requests, ensures authentication security, and guarantees network integrity.
Network Slice Selection Function (NSSF)	Handles network slice selection, ensuring that the appropriate network slice is assigned based on UE requests.
Unified Data Management (UDM)	Manages subscriber data, provides authentication information, and supports policy management and session establishment.
Network Repository Function (NRF)	Manages the registration and discovery of network functions (NFs), facilitating service-based communication within the 5G Core.

2.2 Interface and Security

Open RAN defines mandatory security controls for A1/O1/O2 interfaces, emphasizing mTLS for authenticity and TLS for confidentiality/integrity. E2 uses IPsec. Authorization commonly relies on OAuth 2.0 [9]. In the 5G core SBA, NF-to-NF security similarly requires TLS/ mTLS plus token-based access control [10].

Table 3. Mandatory O-RAN interface security controls [9]

Security control	Non-fronthaul			
	A1	O1	O2	E2
Authenticity	mTLS	mTLS	mTLS	IPsec
Confidentiality	TLS	TLS	TLS	IPsec
Integrity	TLS	TLS	TLS	IPsec
Authorization	OAuth	NACM	OAuth	-
Data origination	mTLS	mTLS	mTLS	IPsec
Replay prevention	TLS	TLS	TLS	IPsec

2.3 X.509 Certificate Framework in Open RAN and 5G

X.509 specifies certificate syntax and path validation (RFC 5280 [11]) and, with OCSP (RFC 6960 [12]), underpins revocation. Open RAN and SBA deployments use X.509 to establish TLS/mTLS among components, with certificate enrollment workflows per O-RAN WG11 [13] and 3GPP TS 33.310 [8]/33.501 [10]. Figure 2 depicts the certificate enrollment process for Open RAN. While mature, these frameworks encounter revocation latency, cross-domain audit challenges, and operational bottlenecks during rapid scaling.

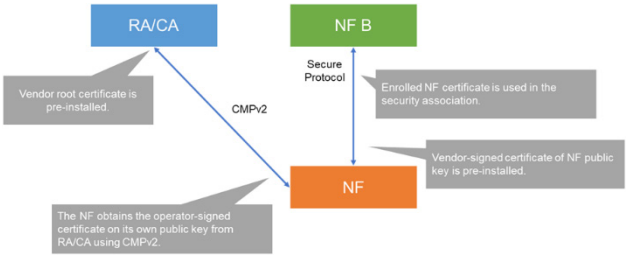

Figure 2. Certificate enrollment of open RAN

Figure 3 shows the corresponding NF certificate enrollment in the 5G Core.

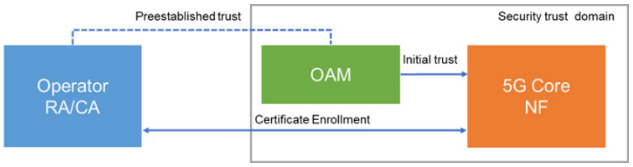

Figure 3. NF certificate enrollment of 5GC

2.4 Problem Statement and System Assumption

Problem: Provide certificate management that is verifiable across administrative domains, automatable at scale, and robust against single-operator failure—without breaking TLS/mTLS.

Approach: Replace CA-centric operations with Blockchain based credentials whose issuance, renewal, suspension, and revocation are governed by smart contracts [14]; bind public keys and attributes to tamper-evident on-chain records; and store rich metadata off-chain with on-chain hash anchoring.

Assumptions: Use a blockchain with economically secure consensus and configurable finality; TA-led identity proofing; authenticated off-chain storage (e.g., InterPlanetary File System (IPFS)) with content hashes on-chain; partially synchronous network conditions; adversaries spanning external, insider, and on-chain (e.g., Maximal Extractable Value (MEV)) classes.

Deployment assumption: NFT-Cert-MN can be instantiated on (a) a public mainnet/L2 for cross-domain public auditability, (b) a permissioned consortium ledger operated by carriers/vendors (optionally with regulators) for SLA-controlled operations, or (c) a private operator/regulator ledger. These options have different governance and failure modes and are treated explicitly in Section 4.8 [2-4]. In all cases, the ledger stores only non-identifying state and content hashes; identity attributes and full X.509 artifacts remain off-chain.

3 NFT-Cert-MN Management System and Structure

3.1 Operational Framework

Public blockchains leverage public key cryptography to ensure transaction integrity and transparency. However, due to the decentralized nature of public blockchains, verifying whether each node is a trusted vendor or operator remains challenging [15]. While factors such as past activity records, reputation systems, and transaction history can help assess trustworthiness, additional mechanisms are necessary to guarantee complete reliability [16]. In today's communication environment, legal responsibilities (e.g., data protection laws, ISO/IEC 27017) and security requirements (e.g., 3GPP TS 33.501 NF certification requirements) are becoming increasingly stringent [10]. As global connectivity expands, the demand for a reliable authentication framework grows. In this context, a robust trust mechanism is essential to verify whether Enrollment requests for Network Function (NF) certificates within Open RAN components and Core Networks originate from trusted nodes.

This paper proposes a blockchain-based framework that utilizes Non-Fungible Tokens (NFTs) to issue and manage certificates. Figure 4 presents the architectural blueprint of the proposed system, whose core components are delineated as follows. The NFT-based certification for mobile networks is referred to as NFT-Cert-MN.

- **Governance:** Defines security policy, selects and audits Trusted Authorities (TAs), and publishes TA registration as NFTs for transparency [17].
- **Trusted Authority (TA):** Performs initial identity proofing, operates the certificate contracts (issue/renew/suspend/revoke/status), and maintains ABI and deployment info for vendors/operators [18].
- **Vendors/Operators:** After TA approval, call contract methods for their NFs to enroll, renew, or revoke NFT-Cert-MN credentials; status checks are real-time on-chain reads [19].

NFT-Cert-MN separates (i) the blockchain lifecycle-control key bound to the subject address (used to invoke contract methods) from (ii) the TLS authentication private key stored on the NF (used only in TLS 1.3 CertificateVerify). To preserve identity binding and prevent

"identity takeover by token movement," certificate NFTs are issued as non-transferable credentials: ERC-721 transfer/approval functions are disabled by the certificate contract, and the contract may expose ERC-5192 feature detection ('locked(tokenId)=true') for interoperability [1].

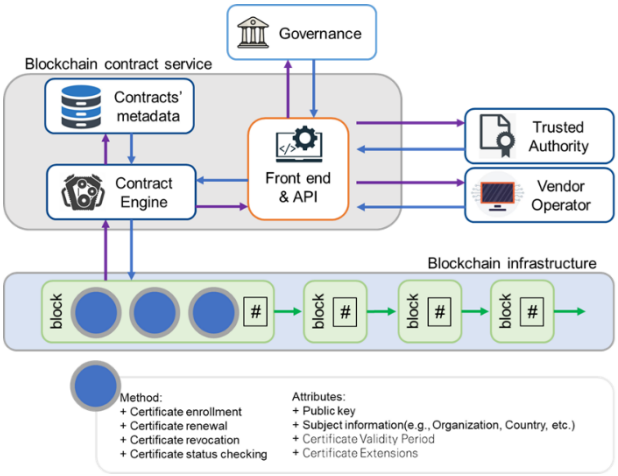

Figure 4. NFT-Cert-MN management system

Only the TA is authorized to deploy, upgrade, and administer the certificate smart contracts; vendors/operators have no deploy/upgrade privileges and interact solely via TA-exposed contract interfaces. Nodes with Enrollment Authority = false are restricted to read-only operations (e.g., status queries) and cannot invoke any issuance function.

3.2 Structure of NFT-Cert-MN

Each certificate is represented as a non-transferable certificate NFT bound to the requester's subject address. Unlike tradable ERC-721 assets, this token is an account-bound credential and MUST NOT be transferable between unrelated parties; the certificate contract therefore disables ERC-721 transfer/approval operations and may implement ERC-5192 ('locked(tokenId)') to signal permanent non-transferability [1]. Each certificate NFT uses an ERC-721 identifier tokenId (a uint256) [20]. Let $mhash=H(\text{meta})$ be the canonical 256-bit hash of the certificate metadata. The certificate owner generates a proof-of-possession signature $\sigma_{pop} = sign(sk_{tls}, mhash)$ (as in standard certificate request flows) [21]. We define the NFT identifier as $tokenId$: = uint256($H_{id}(\sigma_{pop})$), where H_{id} is a 256-bit hash (e.g., Keccak-256 on EVM chains). To avoid ECDSA malleability creating multiple valid identifiers for the same sk_{tls}, $mhash$, signatures are canonicalized (e.g., "low-s") before applying H_{id} [22]. Minimal state (version, serial number, subject address, subject public-key hash, validity, signature algorithm, enrollment authority flag, metadata hash, Token URI) is kept on-chain, while rich metadata (issuer/subject details, public key, contract/network references) resides off-chain.

Only TA-validated identities can issue subordinate certificates, enabling a hierarchical trust chain.

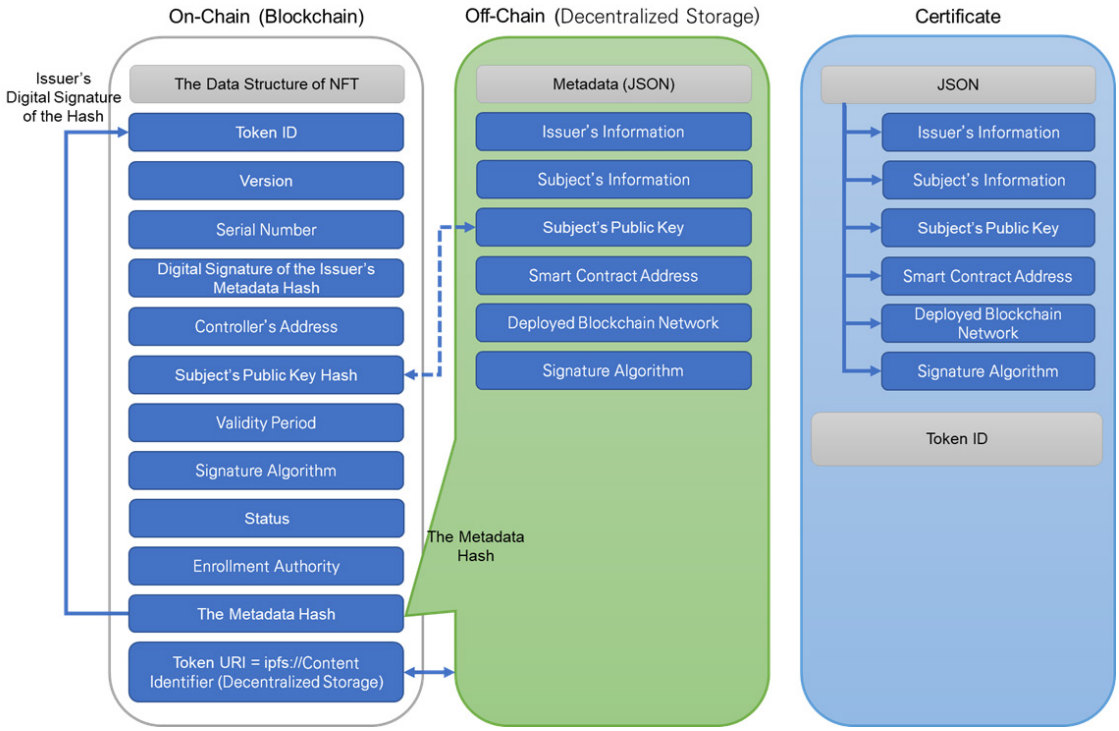

Figure 5. NFT-Cert-MN structure

Table 4. NFT-Cert-MN fields and storage allocation

Field name	Description	On-chain	Off-chain	Cert
Token ID	The value generated by signing the certificate metadata hash *mhash* with the certificate owner's private key $\sigma_{pop} = sign(sk_{tls}, mhash)$; *tokenId* is derived as uint256($H_{id}(\sigma_{pop})$) [20-22].	O	-	O
Version	Certificate version, used to verify Application Binary Interface (ABI) for operations.	O	-	-
Serial number	Unique serial number (per issuer) used for indexing, audit, and revocation tracking; not a signature.	O	-	-
Issuer's signature	Enables retrieval of issuer's public key for verification.	O	-	-
Subject's address	Blockchain address of the certificate requester used for lifecycle-control authorization (issue/renew/suspend/revoke requests); it is not the TLS authentication key.	O	-	-
Subject's public key	TLS authentication public key used in TLS 1.3 CertificateVerify. To minimize exposure, only a hash of the public key needs to be stored on-chain; the full key may remain in off-chain metadata.	O	O	O
Validity period	Includes start (Not Before) and expiration (Not After) dates.	O	-	-
Signature algorithm	The cryptographic algorithm used for signing the certificate.	O	O	O
Status	Details about the certificate status.	O		
Enrollment authority	Indicates whether the certificate allows issuing subordinate certificates.	O	-	-
Metadata hash	Hash value of the certificate metadata, including Issuer, Subject, and public key information.	O	-	-
Token URI	URI linking to off-chain metadata.	O	-	-
Issuer's information	Details about the certificate issuer.	-	O	O
Subject's information	Details about the certificate requester.	-	O	O
Smart contract address	Address of the smart contract managing certificate operations.	-	O	O
Deployed blockchain network	Blockchain network where the certificate operates.	-	O	O

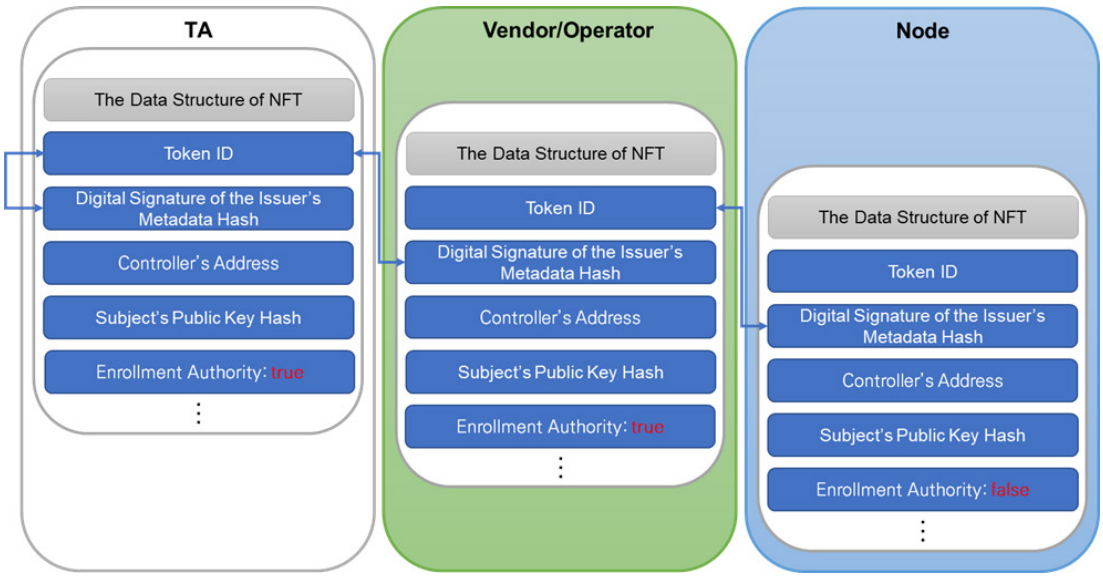

Figure 6. NFT-Cert-MN hierarchical structure

During mutual authentication, peers verify the issuer's signature over the signed metadata hash and cross-check public keys maintained by Open RAN components and core NFs [23]. Figure 7 illustrates an example of this mutual authentication process using NFT-Cert-MN.

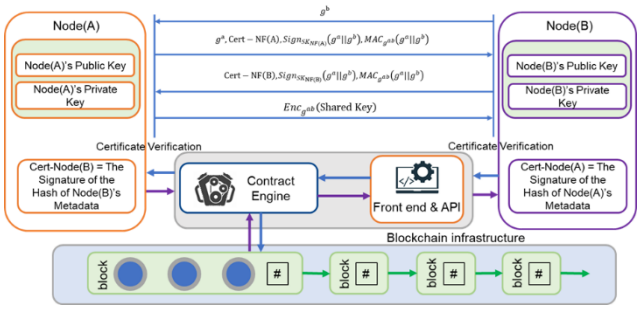

Figure 7. Example of using NFT-Cert-MN for mutual authentication

Hashing may use Secure Hash Algorithm 256 (SHA-256) or Keccak-256; signatures may use Elliptic Curve Digital Signature Algorithm (ECDSA) or Rivest–Shamir–Adleman (RSA). The chosen algorithms are indicated in the metadata's "Signature Algorithm" field.

3.3 Lifecycle and Revocation

NFT-Cert-MN is a finite-state machine: Requested → Active → {Suspended | Revoked | Expired} → Renewed.

- **Issue:** TA/issuer verifies proof-of-possession $\sigma_{pop} = sign(sk_{tls}, mhash)$ and mints a new token with $tokenId$:= uint256($H_{id}(\sigma_{pop})$), while storing the issuer binding as σ_{iss} over $mhash$.
- **Suspend:** Temporarily disable use (e.g., suspected compromise or policy breach).
- **Revoke:** Permanently invalidate with X.509-compatible reason codes.
- **Expire/Renew:** Expiration per validity; renewal produces a new token linked to its parent.
- **Delegation control:** Only tokens with the Enrollment Authority attribute can issue subordinate credentials; TA alone can toggle this under RBAC + multi-sig.
- **Key rotation:** TLS key rotation is handled as "renewal by re-issuance." After proof of possession and policy checks, the TA mints a new token that binds the updated TLS public-key hash and links to the parent token; the prior token transitions to Expired or Revoked depending on the reason.
- **Account recovery (lost/compromised lifecycle-control key):** Because certificate NFTs are non-transferable, recovery is not modeled as token transfer. Instead, after re-proofing the NF/operator identity, the TA revokes the old token and re-issues a replacement token bound to a new subject address, preserving auditability while preventing unauthorized lifecycle actions.

Validation: Status API returns {Valid, Suspended, Revoked, Expired} with block-stamped evidence; clients wait for finality or K confirmations before trusting state transitions.

Algorithm 1: EnrollCertificate: Issue New Certificate NFT

Input: _version, _serialNumber, _csrSignedByReqNode, _signerSignature, _subjectAddress, _subjectPublicKey, _signatureAlgorithm, _metadataHash, _tokenURI
Output: Minted NFT tokenId and stored certificate record

```
tokenId ← BytesToUint256(_csrSignedByReqNode);
if OwnerOf(tokenId) ≠ address(0) then
    revert "Token ID already exists";
node ← GetHolderAddress(_signerSignature);
if node ≠ msg.sender then
    revert "Invalid Sender";
enrollOK ← GetEnrollmentAuthority(_signerSignature);
if not enrollOK then
    revert "EnrollmentAuthrity: false";
regDate ← BlockTimestamp();
expDate ← regDate + 60 · 60 · 24 · 365 ;                    // seconds in 365 days
Store certificateInfos[tokenId] ← { version : _version, serialNumber :
    _serialNumber, signerSignature : _signerSignature, subjectAddress :
    _subjectAddress, subjectPublicKey : _subjectPublicKey, startDate :
    regDate, expirationDate : expDate, signatureAlgorithm :
    _signatureAlgorithm, enrollmentAuthority :
    false, // persisted as false per contractmetadataHash :
    _metadataHash, tokenURI : _tokenURI };
Mint(_subjectAddress, tokenId);
Emit
    CertificateIssued(_subjectAddress, msg.sender, _subjectPublicKey, _csrSignedByReqNode);
```

Algorithm 2: GetCertificateInfo: Lookup Full Certificate Record

Input: _signature
Output: certificateInfo record

```
tokenId ← BytesToUint256(_signature);
if OwnerOf(tokenId) = address(0) then
    revert "Token ID does not exist";
return certificateInfos[tokenId];
```

Algorithm 3: DeleteCertificate: Certificate Revocation (Soft-Delete)

Input: _signature
Output: None (emits CertificateDelete event)

```
tokenId ← BytesToUint256(_signature);
if OwnerOf(tokenId) = address(0) then
    revert "Token ID does not exist";
holder ← GetHolderAddress(_signature);
if holder ≠ msg.sender then
    revert "Invalid holder";
certificateInfos[tokenId].subjectAddress ← address(0);
certificateInfos[tokenId].subjectPublicKey ← 0x00 ;     // zero/empty bytes
certificateInfos[tokenId].expirationDate ← BlockTimestamp() ;
  // invalidate now
Emit CertificateDelete(msg.sender, _signature);
```

4 Results and Analysis

4.1 Candidate Blockchains

Telecom operators can select Ethereum, Polygon, Binance Smart Chain (BSC), Solana, Avalanche, Optimism, or Arbitrum based on throughput (TPS), fees, and consensus. Because status queries are local reads in normal operation, the main determinants are enrollment latency, finality, and economic security.

Deployment type is not interchangeable: Table 5 compares popular public L1/L2 networks for cost and finality measurements, but NFT-Cert-MN can also be deployed on permissioned consortium ledgers or private operator/regulator ledgers. These choices have different governance, compliance, and failure modes; permissioned ledgers in particular are commonly positioned for industrial/governmental deployments where validator participation and SLA enforcement are required [2-3]. Section 4.8 provides an explicit operational mapping (validators, governance, SLA/latency, and data sovereignty) for Public, Consortium, and Private deployments.

Table 5. Comparison of different blockchain networks for NFT

Blockchain network	Number of transactions per second	Gas fee	Consensus algorithm	NFT standards
Ethereum	15-30	High	Proof of Stake (PoS)	ERC-721, ERC-1155
Polygon	2,000+	Low	PoS + Commit Chain	ERC-721, ERC-1155
Binance Smart Chain (BSC)	300	Low	PoS + Authority (PoSA)	BEP-721, BEP-1155
Solana	65,000+	Very Low	Proof of History (PoH) + PoS	SPL NFT (Metaplex)
Avalanche	4,500+	Medium	PoS + Avalanche Consensus	ERC-721, ERC-1155
Optimism	2,000+	Low	Optimistic Rollup + PoS	ERC-721, ERC-1155
Arbitrum	40,000+	Low	Optimistic Rollup + PoS	ERC-721, ERC-1155

4.2 Security Posture of Candidate Chains

Ethereum Proof of Stake (PoS) offers strong decentralization and mature tooling but higher fees. Polygon inherits Ethereum security but previously suffered validator-centered risks [24]. BSC [25] and Solana provide high TPS but have faced centralization and liveness incidents [26-27]. Avalanche, Optimism, and Arbitrum balance scalability with robust security models; Arbitrum's Optimistic Rollup model provides Ethereum-anchored security and high capacity [19], making Ethereum L1 and Arbitrum favorable for high-assurance issuance.

We emphasize that "security posture" depends not only on L1/L2 design but also on who controls validation and governance: permissionless chains prioritize open participation and public auditability, while consortium/private deployments prioritize contractual governance, predictable finality, and jurisdictional control, at the cost of a smaller validator set and different collusion assumptions [2, 4].

4.3 Enrollment Time

Assuming Organization Validation (OV)-like vetting for initial TA-issued identities and Domain Validation (DV)-like re-enrollment for operator-internal credentials, the dominant factor is identity proofing time rather than chain inclusion. Measured on-chain enrollment times span roughly 8–118 seconds depending on network and fees, as shown in Figure 8.

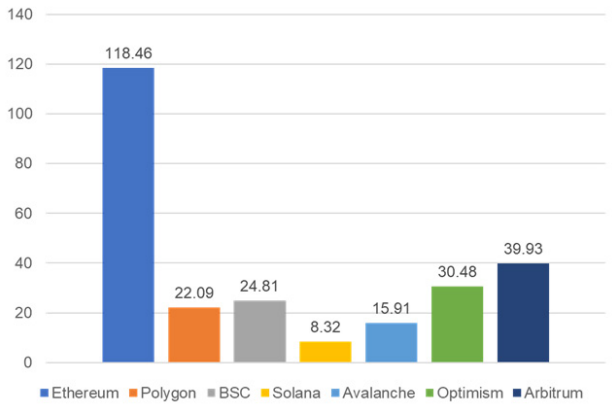

Figure 8. NFT-Cert-MN enrollment time (Seconds)

4.4 Enrollment Cost

A prototype enrollment consumes 488,501 gas; measured costs (Feb 24–Mar 2, 2025, peak-year gas) ranged from ≈$0.0001 on L2s to ≈$4.21 on Ethereum L1, as depicted in Figure 9. Off-chain storage (e.g., IPFS gateways) adds $0–$230/month depending on provider/tier [28-29]. For many deployments, the marginal cost undercuts commercial X.509 DV/OV pricing.

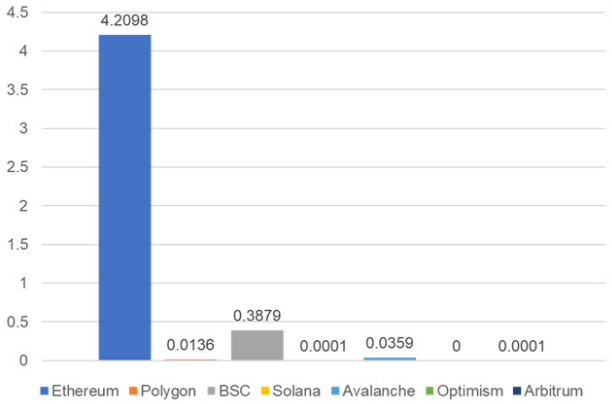

Figure 9. NFT-Cert-MN enrollment costs (USD)

4.5 Threat Model and Goals

- **Adversaries:** (I) External attackers (control/data plane), (II) insiders with stolen operator keys, (III) on-chain adversaries (MEV/front-running, censorship, reorgs).
- **STRIDE:** Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege.
- **Goals:** Authenticity, integrity, availability, auditability, and prompt revocation.
- **Mitigations:** Contract-level Role-Based Access Control (RBAC), multi-sig for admin actions, time-locked upgrades; commit-reveal for sensitive updates; acceptance after K confirmations; fast Suspend()/Revoke() with key rotation per NIST Special Publication (SP) 800-57/800-63; hash-anchored off-chain metadata with minimization and access control.

4.6 Methodology

We report enrollment gas usage as the median of M mints per chain and convert cost using the daily median gas price and native-token/USD rate r. For each run, total cost is

$$Cost_{\$} = (\text{gas used}) \times (\text{median gas price}) \times r \quad \textbf{(1)}$$

We publish median, interquartile range (IQR), and 95% CI across days to account for volatility. To isolate on-chain costs from storage, we report off-chain metadata fees separately by provider and durability tier. In our prototype, a single enrollment consumed 488,501 gas; the addendum introduces distributional statistics and measurement scripts to ensure reproducibility. Time-to-finality is measured as (block inclusion + K confirmations) per chain policy.

4.7 Authentication Workload: X.509/PKI vs. NFT-Cert-MN

This section formalizes the Transport Layer Security (TLS) 1.3 Mutual TLS (mTLS) authentication workload for Network Function (NF)–to–NF communication in Open Radio Access Network (Open RAN) and the 5G Service-Based Architecture (SBA) under two credentialing regimes: (I) traditional Public Key Infrastructure (PKI) with X.509 certificates and (II) the proposed Non-Fungible Token (NFT)-backed certificate model (NFT-Cert-MN). The analysis is consistent with the architecture, token structure, and lifecycle defined in Figure 4, Figure 5, Figure 6 and Table 4, and with the interface protection requirements summarized in Table 3.

Security properties required from "TLS 1.3 + NFT-Cert-MN": (P1) NF entity authentication (the peer is bound to the intended NF/operator identity), (P2) channel security (confidentiality/integrity and forward secrecy as in TLS 1.3), and (P3) revocation freshness (a peer is accepted only if its credential is not Suspended/Revoked/Expired in a finalized ledger view).

Each peer presents its TLS authentication public key (via the standard TLS 1.3 Certificate/CertificateVerify flow) and a reference to its certificate token (e.g., tokenId, contract address, chainId, and metadata hash/URI). The verifier checks (i) the TA/issuer signature binding over the canonical metadata hash, (ii) equality of the presented TLS public-key hash to the on-chain subject public-key hash, and (iii) token status at a finalized block B. We denote this check by:

$$\begin{aligned}
&\text{IssuerSigV}(id) \\
&:= \text{VerifySig}\left(pk_{iss}(id), mhash(id), \sigma_{iss}(id)\right) \\
&= 1 \\
&\text{VerifyNFTCert}(id, B, pk_{tls}) \\
&:= \text{IssuerSigV}(id) \wedge Status(id, B) \\
&= \text{Valid} \wedge H(pk_{tls}) \\
&= pkh_{\text{onchain}}(id) \wedge \text{SubjectID}(id) \\
&= \text{ExpectedSubjectID}
\end{aligned} \quad \textbf{(2)}$$

Here, $pk_{iss}(id)$ denotes the issuer's verification key registered on-chain (TA for root issuance, or a delegated Enrollment Authority for subordinate issuance), and $\sigma_{iss}(id)$ is the issuer signature over the canonical metadata hash $mhash(id)$. ExpectedSubjectID is obtained from the operator's inventory/NRF registration and must match the subject identifier contained in the issuer-signed metadata.

Composition argument: TLS 1.3 uses signature-based authentication (CertificateVerify) to bind the handshake transcript to the peer's authentication key [5]. Formal analyses show that, under standard cryptographic assumptions, the TLS 1.3 handshake establishes session keys with the intended authentication and channel-security properties when the peer authentication key is correctly bound to the peer identity [6]. NFT-Cert-MN replaces "X.509 chain + OCSP/CRL" with "ledger-backed identity binding + finalized status," so the authenticated-channel guarantees of TLS 1.3 carry over as long as VerifyNFTCert enforces the same binding and freshness. Revocation freshness is reduced from OCSP/CRL staleness to ledger finality: the verifier accepts status transitions only after K confirmations/finality and rejects stale views beyond a freshness window W (Section 5.7).

We model the per-session authentication cost as the sum of cryptographic verification and freshness/status work:

$$W = C_{TLS} + V_{path}(A) + V_{path}(B) + R(A) + R(B) \quad \textbf{(3)}$$

where:

- C_{TLS} is the TLS 1.3 core cost (one ECDHE per party; one CertificateVerify signature [5].
- $V_{path}(X)$ is the certificate-path verification cost for party X. For an X.509 chain with I_X intermediates $V_{path}^{PKI}(X) = I_X + 1$ (intermediates + end-entity). In our framework, after initial delegation from TA to a Vendor/Operator, subsequent certificate issuance by that Vendor/Operator occurs at a consistent single issuance depth, so we set, $V_{path}^{NFT}(X) = 1$ for cost analysis; cf. Sec. 3 and Figure 5, Figure 6)
- $R(X)$ is the revocation/status workload for X's credential. Under PKI this is CRL/OCSP I/O + signature verification [11, 33]; under NFT-Cert-MN this is a local on-chain state read (Valid/Suspended/Revoked/Expired) plus an optional hash check of off-chain meta-data (Sec. 3.3)

We model the per-session authentication cost as the sum of cryptographic verification and freshness/status work: (I) N_{sig}, the number of signature verifications attributable to authentication (excluding ECDHE scalars, which are identical), and (II) N_{net}, the number of Internet round trips triggered during status checks.

X.509/PKI path: With server and client chains of lengths (I_A + 1) and (I_B + 1) (intermediates *I*. plus the leaf), the signature-verification workload is

$$N_{sig}^{PKI} = (I_A + 1) + (I_B + 1) + 2 \quad \textbf{(4)}$$

where the final +2 accounts for verifying each peer's CertificateVerify message [11]. In common deployments $I_A = I_B = 1$ (one intermediate), thus $N_{sig}^{PKI} = 6$; $I_A = I_B = 2$, $N_{sig}^{PKI} = 8$. Revocation/status checking contributes

$$R^{PKI} = \sum_{X \in \{A,B\}} \begin{bmatrix} OCSP / \\ CRL\ verification\ on\ checked\ certs \\ +network\ RTTs \end{bmatrix} \quad (5)$$

so if N_{ocsp} online checks are required, then $N_{net}^{PKI} = N_{ocsp}$ and

$$latency_{status}^{PKI} \approx N_{ocsp} \cdot RTT_{ocsp} + local\ verify, \quad (6)$$

noting that OCSP stapling typically covers only the server's leaf; client-certs and intermediates often still require online checks or CRL fetches [12, 33].

NFT-Cert-MN path: Under NFT-Cert-MN, the verifier checks the issuer signature σ_{iss} over the canonical metadata hash $mhash(id)$ and queries the on-chain token state (Valid/Suspended/Revoked/Expired) keyed by (*Contract address*, *tokenId*), optionally validating off-chain metadata by comparing to $mhash(id)$. This yields

$$N_{sig}^{NFT} = 1 + 1 + 2 = 4 \quad (7)$$

i.e., one issuer-signature verification for each party plus two CertificateVerify checks. Status lookups are local smart-contract reads against a site-local full node or gateway, hence

$$N_{net}^{NFT} = 0 \text{ (no Internet RTT during handshake)}$$

Comparison and quantitative deltas: For symmetric chains ($I_A = I_B = I$),

$$\frac{N_{sig}^{NFT}}{N_{sig}^{PKI}} = \frac{4}{2(I+1)+2} = \frac{2}{I+2} \quad (8)$$

Therefore, with one intermediate per side (I=1) the workload shows 33% fewer signature verifications; with two intermediates (I=2), 50% fewer. Figure 10 plots the total signature verifications per handshake as a function of chain depth. For freshness, $N_{net}^{PKI} \in \{0, 1, \ldots\}$ depending on OCSP/CRL policy, versus $N_{net}^{NFT} = 0$ [12, 33]. A simple cache model yields $E[N_{net}^{PKI}] = N_{ocsp}(1-h)$ where h is the cache-hit ratio; absent near-perfect coverage (including client certs and intermediates), at least one RTT often remains on the critical path. Figure 11 compares the expected number of Internet round trips under varying OCSP/CRL cache-hit ratios for PKI versus NFT-Cert-MN.

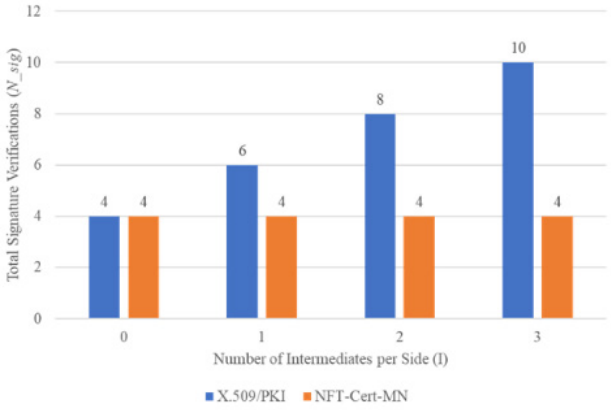

Figure 10. Signature verifications per TLS 1.3 mTLS handshake vs. Chain depth (I)

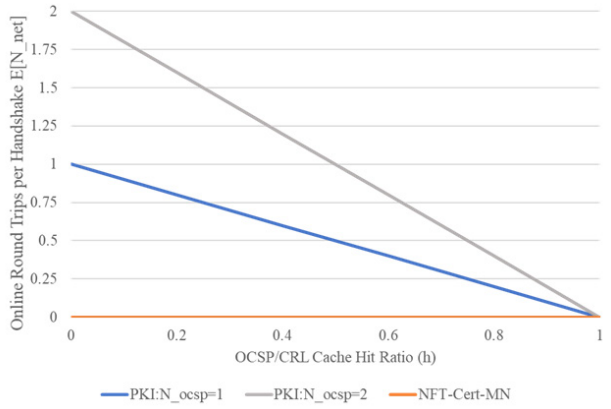

Figure 11. Internet round trips for Status checks vs. Hit ratio (h)

4.8 Deployment Models and Operational Assumptions

To allow telecom operators to assess feasibility, we distinguish three deployment profiles with different trust, governance, and operational consequences.

- **Public (permissionless) mainnet/L2:** Validators are open and governance is external to mobile operators. This profile maximizes public auditability and reduces reliance on any single telecom entity, but fees and inclusion latency are market-driven. Operationally, NFs SHOULD not depend on Internet RPC during handshake; each site runs a local full node or hardened gateway for reads, and lifecycle writes (Issue/Renew/Suspend/Revoke) occur out of band from the data-plane handshake.
- **Permissioned consortium ledger (recommended for inter-operator/Open RAN):** Validators are operated by participating carriers (and optionally major vendors/regulators) under contractual governance. ETSI's work on permissioned distributed ledgers highlights that permissioned ledgers are often better suited to industrial/governmental use cases, including SLA verification

and regulatory enforcement via smart contracts [2], and telecom-specific PDL service provisioning requires explicit interaction with existing telecom functions [3]. This profile enables predictable fees, controllable finality, and jurisdictional validator placement to satisfy data-sovereignty constraints [4].

- **Private operator/regulator ledger:** A single operator or regulator controls validation. This offers low latency and operational simplicity but reintroduces concentrated trust similar to internal PKI; it should be viewed primarily as an automation and audit-log layer rather than decentralization of trust.

Table 6 summarizes the operational mapping.

Table 6. Deployment profiles for NFT-Cert-MN

Feature	Public (Open)	Consortium (Federated)	Private (Closed)
Validators	Open (Anyone can participate)	Restricted (Carriers, vendors, regulator)	Single Entity (Centralized control)
Governance	External (Decentralized community)	Contractual (Agreements among members)	Single Entity (Internal policy)
SLA	Best-effort	Negotiated (Guaranteed service levels)	Internal (Strictly controlled)
Main risk	Congestion, Censorship, Fee Volatility	Validator Collusion, Partition within Consortium	Single Point of Failure (SPoF), Insider Abuse

A hybrid approach is also possible: a consortium ledger periodically anchors checkpoints (e.g., block hashes) to a public chain to obtain public auditability without exposing operational writes to public fee markets.

5 Discussion

5.1 Key Management and Ownership

NFT-Cert-MN involves multiple keys with different semantics: (i) the TLS authentication private key used in TLS 1.3 CertificateVerify, (ii) the lifecycle-control key bound to the subject address that authorizes on-chain lifecycle requests, and (iii) the TA issuance signing key that binds identities to metadata hashes. Therefore, "ownership follows private keys" must be interpreted per key role: compromising the lifecycle-control key alone MUST NOT be sufficient to impersonate an NF in TLS, and token transfer is disabled to prevent identity takeover by moving a token [1]. Table 7 summarizes the compromise scenarios and corresponding recovery strategies.

Operationally, private keys for TA and high-value NFs SHOULD be generated and stored in certified cryptographic modules (e.g., HSMs) and protected with multi-signature approvals and escrow/backup procedures consistent with NIST key-management guidance [30-31]. If the subject address is a smart-contract wallet, signature validation for lifecycle requests follows ERC-1271, enabling signer rotation without changing the on-chain subject address [32].

Table 7. Compromise scenarios and recovery strategies for NFT-Cert-MN

Compromise scenario	Impact & Risk analysis	Mitigation & Recovery strategy
TLS Key Compromise (sk_{tls} leaked)	- **High Risk** - Attacker may impersonate the entity until the credential is suspended or revoked.	- Immediate Action: Trigger on-chain Suspend or Revoke. - Recovery: Re-issuance (Renewal) with a new TLS public-key hash (pkh').
Lifecycle Key Compromise (sk_{lc}, leaked, without sk_{tls})	- **Medium Risk** - Attacker may attempt unauthorized lifecycle actions (e.g., status change). - Note: Cannot complete TLS 1.3 handshake (cannot impersonate traffic).	- Restriction: Update logic restricted to TA-approved operations. - Recovery: Revoke current NFT and Re-issue to a new subject address after re-proofing.
Loss of Lifecycle Key (Access lost)	- **Operational Risk** - Subject loses management capability of the certificate.	- TA Mediation: Issue a new token to a new address via TA interaction. - Cleanup: Revoke the old token to block future unauthorized requests.
TA Key Compromise (Root of Trust leaked)	- **Catastrophic** - Complete collapse of system trust foundation.	- Prevention: Use HSM-grade controls and multi-party governance. - Recovery: Execute emergency key rotation procedures immediately.

5.2 NFT Standards and Algorithm

Ethereum Virtual Machine (EVM) chains (Ethereum, Polygon, BSC, Optimism, Arbitrum) commonly use ERC-721/1155; Solana adopts Metaplex SPL; Avalanche supports ERCs and Avalanche-native assets. Differences in account models, fee markets, and finality influence minting/batching, metadata handling, and security assumptions. Choose chain-specific optimizations (e.g., batched mints on high-fee L1s) [34] and enforce contract-level permissions and anti-reentrancy guards [35].

5.3 Metadata Integrity

To prevent off-chain tampering or loss, hash the canonical metadata (e.g., SHA-256) and store the digest on-chain [35]. Maintain recovery paths by embedding essential certificate data at the NF and validating by hash re-anchor when gateways are restored.

5.4 Smart-Contract Security

Common pitfalls include reentrancy, arithmetic bugs, missing access control [35], and unsafe upgrades [36]. Apply formal review and external audits; enforce RBAC plus multi-sig for issuance/revocation; gate upgrades behind governance with time-locks and staged rollouts.

5.5 Interoperability with X.509, CT, and DANE

Embed a DER-encoded X.509 off-chain and anchor its hash on-chain to aid coexistence. Emit CertificateIssued, Renewed, Revoked events for CT-style transparency. Publish optional Transport Layer Security Authentication (TLSA) records for Domain Name System Security Extensions (DNSSEC)-anchored DANE validation [37-38].

5.6 Privacy and Compliance

Minimize on-chain data to hashes, timestamps, and non-identifying attributes. Keep Personally Identifiable Information (PII) off-chain with access controls and audit logging. Align identity proofing with NIST SP 800-63 [39] and key lifetimes with NIST SP 800-57 [30]. Conduct Data Protection Impact Assessment (DPIA) [40] where vendor/operator PII is processed.

5.7 Availability and Liveness

Deploy local full-node gateways with read-through caches; apply grace periods around expiry; checkpoint for fast resync; accept state transitions only after configured finality. In extended partitions, fall back to bounded Time To Live (TTL) on last-known-good state.

We define the freshness window W as the maximum tolerated staleness of the local ledger view; if the node's last finalized block timestamp exceeds W, the verifier fails closed (or falls back to an operator policy such as a short-lived cached status).

5.8 Comparison with Prior Work

We contrast NFT-Cert-MN with blockchain-based credential systems across decentralization [34, 41], lifecycle automation [42-43], scalability/cost [43-44], metadata integrity [43], smart-contract security [42], and telecom fit [41]. Prior DPKI work decentralizes issuance but often lacks automated lifecycle and telecom integration; several NFT-based schemes store documents but omit strong off-chain binding or RBAC on contracts. NFT-Cert-MN combines hierarchical delegation, evented lifecycle, gas-optimized anchoring, audited RBAC, and explicit Open RAN/5G SBA alignment. Table 8 presents a comparative evaluation of NFT-Cert-MN against prior blockchain-based certificate management systems.

Table 8. Comparative evaluation of blockchain-based certificate management systems

Criterion	NFT-Cert-MN	Papageorgiou et al. [34]	Wijethilaka et al. [41]	Zaman et al. [42]	Kumawat & Naik [43]	Turuta et al. [44]
Decentralization of trust	✓ Full (TA-anchored, NFT hierarchy)	✓ Full (CA eliminated)	✓ Partial (limited decentralization)	× Central authority remains	✓ Partial (admin-issued NFTs)	× Central authority remains
Certificate lifecycle automation	✓ Full via smart contracts	× Limited support	× Manual control	✓ Partial automation	× No automation	× Manual issuance
Scalability & Cost efficiency	✓ Off-chain hash storage, gas-optimized	× No optimization	× Not analyzed	× High overhead	× Not considered	✓ Layer 2 (Polygon) used
Metadata integrity & Off-chain support	✓ Advanced binding (On-chain hash + IPFS)	× Not addressed	× Not addressed	✓ Basic binding	× Hash mention only	× Off-chain not addressed
Smart contract security	✓ RBAC, upgradability, verified logic	× Not specified	× Not specified	✓ Basic RSA verification	× Minimal logic	× No contract security
Mobile network integration	✓ 5G/6G, Open RAN, SBA compatible	× IoT focused	✓ Network slicing	× Academic domain only	× Generic credential focus	× Retail product focus

6 Conclusion

This work introduced **NFT-Cert-MN**, an NFT-backed certificate framework that preserves TLS/mTLS interoperability while decentralizing trust and lifecycle control. By anchoring governance in a TA yet enforcing issuance and status via smart contracts with RBAC and multi-sig, NFT-Cert-MN provides transparent, auditable state transitions without relying on CA-centric online status infrastructure.

Empirically, the framework achieves concrete gains: 33–50% fewer signature verifications during mTLS authentication for typical chain depths; elimination of online OCSP round trips when status is read from a finalized local chain; and predictable enrollment costs (≈$0.0001 on L2 to ≈$4.21 on Ethereum L1 for a 488,501-gas mint), with measured on-chain inclusion in the 8–118 s band. These results indicate that decentralizing certificate state is feasible for carrier-grade environments and compatible with existing NF control/data-plane protection.

Operationally, the design's hybrid storage (on-chain minimal state; off-chain rich metadata with on-chain hash) limits blockchain footprint while retaining verifiability, and the evented lifecycle integrates cleanly with CT and DANE practices and prevailing 3GPP/O-RAN controls. The chain analysis suggests pragmatic deployment profiles—e.g., Ethereum L1 or Ethereum-anchored L2s (such as Arbitrum) for high-assurance issuance—while allowing operators to trade off cost, finality, and throughput.

Looking ahead to **6G** and more dynamic, multi-domain orchestration, NFT-Cert-MN offers a forward-compatible foundation for secure function onboarding, mutual authentication, and responsive revocation. Future research includes deeper formal analysis and verification of the contracts and lifecycle policies, broader interoperability testing across vendors/operators, and expanded integration with transparency mechanisms and DNSSEC-anchored records to strengthen cross-domain auditability.

Acknowledgements

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00444170, Research and international collaboration on trust model-based intelligent incident response technologies in 6G open network environment).

References

[1] T. Daubenschütz, A. T. Anders, *ERC-5192: Minimal Soulbound NFTs*, Ethereum Improvement Proposals, No. 5192, July, 2022. [Online]. Available: https://eips.ethereum.org/EIPS/eip-5192

[2] ETSI, *Permissioned Distributed Ledgers (PDL)*, ETSI ISG PDL. [Online]. Available: https://www.etsi.org/technologies/permissioned-distributed-ledgers [Accessed: Mar. 1, 2025].

[3] ETSI, *ETSI GS PDL 024 V1.1.1: Architecture Enhancements for PDL Service Provisioning in Telecom Networks*, November, 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/PDL/001_099/024/01.01.01_60/gs_PDL024v010101p.pdf

[4] ETSI, *ETSI GR PDL 021 V1.1.1: 3GPP Use Cases*, October, 2023. [Online]. Available: https://www.etsi.org/deliver/etsi_gr/PDL/001_099/021/01.01.01_60/gr_PDL021v010101p.pdf

[5] E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.3*, RFC 8446, August, 2018. https://doi.org/10.17487/RFC8446

[6] B. Dowling, M. Fischlin, F. Günther, D. Stebila, A Cryptographic Analysis of the TLS 1.3 Handshake Protocol, *Journal of Cryptology*, Vol. 34, No. 4, Article No. 37, October, 2021. https://doi.org/10.1007/s00145-021-09384-1

[7] S. Marinova, A. Leon-Garcia, Intelligent O-RAN Beyond 5G: Architecture, Use Cases, Challenges, and Opportunities, *IEEE Access*, Vol. 12, pp. 27088–27114, February, 2024. https://doi.org/10.1109/ACCESS.2024.3367289

[8] 3GPP, *Network Domain Security (NDS); Authentication Framework (AF)*, 3GPP TS 33.310, Release 19, 2025.

[9] O-RAN ALLIANCE, *WG11 Security Requirements and Controls*, O-RAN Specification, 2024.

[10] 3GPP, *Security Architecture and Procedures for 5G System*, 3GPP TS 33.501, Release 19, 2025.

[11] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, RFC 5280, May, 2008. https://doi.org/10.17487/RFC5280.

[12] S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams, *Online Certificate Status Protocol (OCSP)*, RFC 6960, June, 2013. https://doi.org/10.17487/RFC6960

[13] O-RAN ALLIANCE, *WG11 Study on Certificate Management Framework*, O-RAN Technical Report, 2024.

[14] E. Bellini, E. Damiani, S. Marrone, Blockchain-Based Trustworthy O2O Interaction in the Next 6G Ecosystem, *Proc. 2023 International Conference on Cyber Security and Resilience (CSR)*, Venice, Italy, 2023, pp. 92–98. https://doi.org/10.1109/CSR57506.2023.10224977

[15] I. C. Lin, T. C. Liao, A Survey of Blockchain Security Issues and Challenges, *International Journal of Network Security*, Vol. 19, No. 5, pp. 653–659, September, 2017. https://doi.org/10.6633/IJNS.201709.19(5).01

[16] T. Maksymyuk, J. Gazda, M. Vološin, G. Bugar, D. Horvath, M. Klymash, M. Dohler, Blockchain-Empowered Framework for Decentralized Network Management in 6G, *IEEE Communications Magazine*, Vol. 58, No. 9, pp. 86–92, September, 2020. https://doi.org/10.1109/MCOM.001.2000175

[17] V. Buterin, Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform, White Paper, 2013. [Online]. Available: https://ethereum.org/en/whitepaper/

[18] Z. Zheng, S. Xie, H. N. Dai, X. Chen, H. Wang, An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends, *Proc. 2017 IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, USA, 2017, pp. 557–564. https://doi.org/10.1109/BigDataCongress.2017.85

[19] D. Yaga, P. Mell, N. Roby, K. Scarfone, Blockchain Technology Overview, NISTIR 8202, National Institute of Standards and Technology, October, 2018.

https://doi.org/10.6028/NIST.IR.8202

[20] W. Entriken, D. Shirley, J. Evans, N. Sachs, E*RC-721: Non-Fungible Token Standard*, Ethereum Improvement Proposals, No. 721, January, 2018. [Online]. Available: https://eips.ethereum.org/EIPS/eip-721

[21] M. Nystrom, B. Kaliski, *PKCS #10: Certification Request Syntax Specification Version 1.7*, RFC 2986, November, 2000. https://doi.org/10.17487/RFC2986.

[22] V. Buterin, *Homestead Hard-Fork Changes*, Ethereum Improvement Proposals, No. 2, 2016. [Online]. Available: https://eips.ethereum.org/EIPS/eip-2

[23] E. Beckwith, G. Thamilarasu, BA-TLS: Blockchain Authentication for Transport Layer Security in Internet of Things, *2020 7th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, Paris, France, 2020, pp. 1–8. https://doi.org/10.1109/IOTSMS52051.2020.9340204

[24] H. Maishera, *Polygon (MATIC) Reveals It Was Hacked Earlier This Month*, Yahoo Finance, Dec. 2021. [Online]. Available: https://finance.yahoo.com/news/polygon-matic-reveals-hacked-earlier-103532665.html

[25] E. Howcroft, *Binance-Linked Blockchain Hit by $570 Million Crypto Hack*, Reuters, October, 2022. [Online]. Available: https://www.reuters.com/technology/hackers-steal-around-100-million-cryptocurrency-binance-linked-blockchain-2022-10-07

[26] Solana Foundation, *Network Outage Report*, 2022. [Online]. Available: https://status.solana.com/ [Accessed: Mar. 1, 2025].

[27] Solana, *Solana (Blockchain Platform)*, Wikipedia, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Solana_(blockchain_platform) [Accessed: Mar. 1, 2025].

[28] Pinata, *Storage Pricing*, 2025. [Online]. Available: https://pinata.cloud/pricing [Accessed: Mar. 1, 2025].

[29] Infura, *Storage Pricing*, 2025. [Online]. Available: https://www.infura.io/pricing [Accessed: Mar. 1, 2025].

[30] E. Barker, *Recommendation for Key Management: Part 1 – General*, NIST SP 800-57 Part 1, Rev. 5, National Institute of Standards and Technology, May, 2020. https://doi.org/10.6028/NIST.SP.800-57pt1r5.

[31] National Institute of Standards and Technology (NIST), *FIPS 140-3: Security Requirements for Cryptographic Modules*, March, 2019. https://doi.org/10.6028/NIST.FIPS.140-3.

[32] F. Giordano, M. Condon, P. Castonguay, A. Bandeali, J. Izquierdo, B. Masius, *ERC-1271: Standard Signature Validation Method for Contracts*, Ethereum Improvement Proposals, No. 1271, July, 2018. [Online]. Available: https://eips.ethereum.org/EIPS/eip-1271

[33] P. Hallam-Baker, R. Stradling, *X.509v3 Transport Layer Security (TLS) Feature Extension*, RFC 7633, October, 2015. https://doi.org/10.17487/RFC7633.

[34] A. Papageorgiou, A. Mygiakis, K. Loupos, T. Krousarlis, DPKI: A Blockchain-Based Decentralized Public Key Infrastructure System, *Proc. Global Internet of Things Summit (GIoTS)*, Dublin, Ireland, 2020, pp. 1–5. https://doi.org/10.1109/GIOTS49054.2020.9119673

[35] P. Mell, D. Yaga, *Non-Fungible Token Security*, NISTIR 8472, National Institute of Standards and Technology, March, 2024. https://doi.org/ 10.6028/NIST.IR.8472.

[36] S. Yang, J. Chen, Z. Zheng, Definition and Detection of Defects in NFT Smart Contracts, *ISSTA 2023: Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, Seattle, WA, USA, 2023, pp. 373–384. https://doi.org/10.1145/3597926.3598063

[37] B. Laurie, A. Langley, E. Kasper, *Certificate Transparency*, RFC 6962, June, 2013. https://doi.org/10.17487/RFC6962.

[38] P. Hoffman, J. Schlyter, *The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA*, RFC 6698, August, 2012. https://doi.org/10.17487/RFC6698.

[39] D. Temoshok, D. Proud-Madruga, Y. Choong, R. Galluzzo, S. Gupta, C. LaSalle, N. Lefkovitz, A. Regenscheid, *Digital Identity Guidelines*, NIST SP 800-63-4, National Institute of Standards and Technology, July, 2025. https://doi.org/10.6028/NIST.SP.800-63-4

[40] European Parliament and Council of the European Union, *Regulation (EU) 2016/679 — General Data Protection Regulation (GDPR)*, Official Journal of the European Union, Article No. 35, April, 2016. [Online]. Available: https://gdpr-info.eu/art-35-gdpr/

[41] S. Wijethilaka, A. K. Yadav, A. Braeken, M. Liyanage, Blockchain-Based Secure Authentication and Authorization Framework for Robust 5G Network Slicing, *IEEE Transactions on Network and Service Management*, Vol. 21, No. 4, pp. 3988–4005, August, 2024. https://doi.org/10.1109/TNSM.2024.3416418

[42] N. Zaman, I. K. Aksakalli, N. Baygın, Digital Certificate Security: A Blockchain-Based Approach for Fraud Prevention and Verification, *Bitlis Eren University Journal of Science*, Vol. 12, No. 4, pp. 1128–1138, December, 2023. https://doi.org/10.17798/bitlisfen.1343747

[43] N. Kumawat, D. Naik, Utilizing NFTs to Revolutionize Document Verification and Authentication through Blockchain: Redefining Trust in the Digital Era, *Proc. 15th International Conference on Computing Communication and Networking Technologies (ICCCNT 2024)*, IIT Mandi, Himachal Pradesh, India, 2024, pp. 835–840. https://doi.org/10.1109/ICCCNT61001.2024.10724719

[44] O. Turuta, M. Kozulia, V. Nikitin, Certification of Digital Content Based on NFT Technologies, *Proc. 8th International Conference on Computational Linguistics and Intelligent Systems. Volume IV: Computational Linguistics Workshop*, Lviv, Ukraine, 2024, pp. 343–355. https://doi.org/10.31110/COLINS/2024-4/019

Biographies

Dowon Kim received an M.S. degree in Information and Communication Engineering from Korea University, Seoul, South Korea, in 2010. Since 2005, he has been with the Korea Internet & Security Agency (KISA), Naju, South Korea, where he served as a team leader responsible for security vulnerability information analysis and management, as well as cybersecurity R&D. He is currently a Guest Researcher at the National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, engaged in research on Open RAN security. His research interests include enhancing

security in mobile networks, with a particular focus on anomaly prevention and detection.

Seongmin Park received the B.S. degree in Physics and Electronic engineering from Sogang University, Seoul, South Korea, in 2009. Also, He received the M.S. degree in Management of Technology from the same university in 2015, respectively, and the Ph.D. degree in Information Security Engineering at Kookmin University, Seoul, South Korea. From 2009 to 2013, he worked as a Researcher with Core Network Development Lab, LGUplus co., Seoul, South Korea. Since 2013, he has worked as a General Researcher with the Korea Internet Security Center, Korea Internet & Security Agency, Naju, South Korea. His research interest includes Mobile security, Network security, Convergence security and AI security analysis.

Daeun Kim received the B.S. degree in Computer Engineering from Chonnam National University, Gwangju, South Korea, in 2015, and the M.S. degree in Information Security from the same university in 2017. Since 2017, she has worked as a Researcher with the Korea Internet & Security Agency (KISA), Naju, South Korea. Her research interests include digital forensics, wireless communication security, and maritime cybersecurity.

Ilsun You (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science from Dankook University, Seoul, South Korea, in 1997 and 2002, respectively, and the Ph.D. degree from Kyushu University, Japan, in 2012. He is currently working as a Full Professor with the Department of Financial Information Security, Kookmin University, South Korea. His research interests include internet security, authentication, access control, and formal security analysis. He is a fellow of the IET. He is on the Editorial Board of Information Sciences, Journal of Network and Computer Applications, International Journal of Ad Hoc and Ubiquitous Computing, Computing and Informatics, and Intelligent Automation and Soft Computing.