Towards Minimal Certificates for Federated Space Public Key Infrastructure

Alin-Petru Roşu
Delft University of Technology, The Netherlands
a.rou@student.tudelft.nl
European Space Agency, The Netherlands
alin.rosu@ext.esa.int

Oana-Alexandra Graur
European Space Agency, The Netherlands
oana-alexandra.graur@esa.int

***Abstract*—Federated Space Public Key Infrastructure (PKI) can offer a scalable foundation for secure and interoperable communications in collaborative space missions. Yet, its deployment faces challenges stemming from resource-constrained assets, architectural complexity, and the transition to post-quantum (PQ) cryptography. Current CCSDS space guidelines rely on the Internet X.509 profile, whose extensive feature set, if left unrestricted, can increase implementation complexity, certificate size (especially under PQ algorithms), and the risk of interoperability issues. In parallel, the IETF C509 Certificates draft emerges as a streamlined subset of X.509 with a compact encoding specifically tailored for constrained environments. This paper provides an empirical comparison between X.509 and C509 to inform space mission designers about the associated advantages and costs of each, specifically when PQ cryptography is incorporated into space PKIs. To help pave the way for interoperability in federated space missions, a minimal certificate profile for space PKI is proposed.**

In addition, the work introduces the first open-source native C509 toolkit that supports PQ algorithms and evaluates open-source and proprietary certificate parsers. While the IETF C509 draft proposal reports a size reduction of over 50%, our evaluation confirms approximately 40% savings for traditional certificates generated according to our proposed minimal certificate profile. For PQ certificates, the savings plateau at around 200 bytes, rendering the size gains negligible. However, revocation lists consistently achieve a 60% reduction for 30,000 entries, independent of the cryptographic scheme (PQ or traditional). To quantify and compare the software implementation complexity of X.509 and C509, we conduct software complexity analysis using well-established heuristic metrics (e.g., cyclomatic complexity, Halstead metrics, logical lines of code). The findings further highlight the relative simplicity of the C509 parser implementation in software. Defining a standardised certificate profile for federated space would advance interoperability; however, adopting C509 requires carefully balancing modest PQ size savings against software simplification and the uncertainties associated with a draft standard.

***Index Terms*—federated public key infrastructure (PKI), space systems security, X.509, C509, certificate profile, constrained devices, post-quantum cryptography (PQC), CBOR encoding.**

I. Introduction

Recent large-scale space endeavours, such as the Artemis program [1], highlight a shift toward federated operations involving multiple space agencies and private actors. These collaborative missions demand secure, scalable communication across independently governed domains (e.g., LunaNet). While space standards today primarily rely on symmetric cryptography [2], such approaches do not scale across federated environments with multiple actors, each subject to different national regulatory constraints. These constraints can be cumbersome and difficult to satisfy simultaneously, especially when the sharing of secret key material is involved. On Earth, this challenge is addressed by Public Key Infrastructure (PKI), which enables trust establishment and key management at scale. However, deploying PKI in space introduces architectural and operational challenges [3], such as constrained bandwidth, hardware limitations, and the complexity of cross-organisation interoperability.

In this regard, the Consultative Committee for Space Data Systems (CCSDS) has initiated the specification of an Intergovernmental Certification Authority (IGCA) [4], an experimental federated PKI framework intended to foster trusted cooperation among entities. Still, CCSDS notes that PKI components from different vendors may be unable to communicate, and users may find they cannot process each other's certificates [3], raising interoperability concerns. To address this, IGCA builds on the CCSDS Certificate Profile [5], which, in turn, adopts the terrestrial X.509 Internet profile [6] without tailored adaptations.

X.509 was designed for general-purpose terrestrial use and lacks minimalism. It embeds redundant data and employs verbose encoding [7], thereby inflating the certificate size beyond its core cryptographic content and burdening bandwidth-limited links. Additionally, its implementation complexity broadens the attack surface: context-dependent parsing has been linked to memory errors [8], impersonation [9], and denial-of-service attacks [10]. These risks are amplified in embedded systems, where secure implementation is especially challenging. Thus, X.509 can raise performance and implementation concerns. Although widely deployed on Earth, the suitability of X.509 Internet profile for federated space PKI remains a matter of debate. In practice, many mission designers who recognise the benefits of using asymmetric cryptography in space might be incentivised to adopt the X.509 Internet profile as the default or "safe choice". However, once the implementation complexity on space-constrained hardware becomes evident, they may reduce it subsequently by pruning extensions, compressing chains, or possibly tunnelling revocation checks, all to preserve compatibility with the terrestrial X.509 standard. Nevertheless, such pursuits can often have a more apparent than real effect on achieving interoperability across federated space systems, especially when uncoordinated.

Furthermore, under IGCA, the current profile allows unrestricted use of extensions. In X.509, extensions are optional fields encoding attributes or constraints, often governing trust, key use, or policies. Allowing federation participants to define profiles without coordination risks security domain fragmentation and interoperability failures; for instance, unrecognised critical extensions may break validation. While X.509 (and thus the CCSDS Certificate Profile) recommends 17 standard extensions [6], it does not prohibit custom or vendor-specific ones [11]. In practice, a survey of 200 million certificates revealed nearly 200 distinct extensions [8], and an analysis of 11 million certificates found 21.5% syntactically incorrect, with 5.7–10.5% still accepted by major TLS libraries [9], highlighting inconsistencies among implementations claiming the same conformance [12]. Uncontrolled extension use, potentially deprecated ones [13], can yield inconsistent configurations, problematic in federated settings where certificates must be deterministic and uniformly interpreted.

The migration to post-quantum (PQ) cryptography presents additional challenges for X.509 in terms of interoperability and certificate size. While the IGCA acknowledges the need to address PQ algorithms, the CCSDS Authentication Credentials (Certificate Profile) does not currently include specific guidance. Of particular interest is guidance on the hybridisation of traditional (pre-quantum) and PQ algorithms within certificates. Hybridisation is presently part of the European Commission's recommendations [14] for transitioning to PQ. Moreover, PQ schemes generate much larger keys and signatures, often tens of kilobytes, straining bandwidth and embedded memory. These underscore the need to minimise certificate size and define additional requirements for PQ support.

To investigate alternatives to existing standards, this paper examines the emerging C509 profile, currently under standardisation by the Internet Engineering Task Force (IETF) [7]. Designed for constrained devices, C509 defines a restricted subset of X.509 features to reduce parsing complexity and certificate size, sometimes by over 50%, as reported in [7]. It employs a compact encoding [15], suitable for devices with about 10 KiB of RAM and 100 KiB of flash [16]. While promising, its relevance to federated space PKI remains to be evaluated.

This paper adopts a certificate-centric stance to explore the development of a minimal, interoperable profile for federated space PKI. Specifically, it: (i) analyses PQ formats and mitigations for interoperability challenges; (ii) proposes a preliminary minimal profile with fixed extension sets for federated use in space; and (iii) conducts an empirical comparison of X.509 and C509 in terms of certificate size and software implementation complexity. Additionally, it (iv) introduces the first open-source implementation of natively signed C509 certificates with PQ support, offering tooling for certificate generation, signing requests, and revocation lists.

By laying the groundwork for a dedicated and well-defined certificate profile, this work aims to support space standardisation bodies such as CCSDS in advancing an interoperable, certificate-based trust infrastructure for future federated space environments, particularly within the IGCA framework.

II. Preliminaries

A. Public Key Infrastructure and Space

Public Key Infrastructure (PKI) enables scalable trust establishment and key management in distributed systems through digital certificates—signed data structures binding public keys to vetted identities. Certificates are issued by trusted Certification Authorities (CAs), which also maintain the status of certificates, e.g., via Certificate Revocation Lists (CRLs). Hierarchical PKIs establish vertical trust through chains of subordinate CAs, anchored in a trusted root CA, requiring relying parties to validate certificates by constructing and verifying trust chains.

Federated PKIs extend trust horizontally across domains governed independently, using models such as cross-certification, bridge CAs, or mesh topologies. Bridge CAs serve as neutral intermediaries, mapping certificate policies and reducing the number of cross-domain trust relationships. A prominent terrestrial example is the U.S. Federal Bridge Certification Authority (FBCA) [17], which interconnects federal and commercial PKIs to enable cross-domain validation. While such architectures support scalable interoperability, they introduce complexity in governance and policy alignment [18], [19].

The Intergovernmental Certification Authority (IGCA) [4] defines a federated space PKI framework based on a bridge-CA model, enabling cross-domain trust among agencies. Participants may cross-certify their root CAs with the IGCA bridge or rely on the IGCA's issuing CA, fostering interoperability while preserving organisational autonomy. IGCA specifies the technical and operational requirements for issuing and managing certificates and currently relies on the CCSDS profile derived from X.509 Internet certificates [5].

B. X.509 Certificates

The X.509 standard, defined by the International Telecommunication Union Telecommunication Standardisation Sector (ITU-T) [11], specifies the structure of public-key certificates using Abstract Syntax Notation One (ASN.1). Certificates are encoded using the Distinguished Encoding Rules (DER), a canonical Tag-Length-Value (TLV) format that ensures deterministic binary representations required for signing [20].

Version 3 of X.509 introduced an extension mechanism, allowing additional fields and semantics without breaking backwards compatibility. Each extension is identified by an object identifier (OID) and marked as critical or non-critical, indicating if the relying party must understand it to validate the certificate.

The IETF defines the X.509 Internet profile [6], which constrains the feature set to ensure consistency across implementations. It specifies mandatory fields, signature algorithms, name forms, and standard extensions, serving as the basis for Internet-wide interoperability. It also underpins application-specific profiles [21]–[24], which introduce tailored extension sets.

C. C509 Certificates

C509 is an emerging IETF profile for constrained devices where traditional X.509 certificates are too large or complex [7].

It preserves a subset of the X.509 information model but serialises it using deterministic Concise Binary Object Representation (CBOR) [15], reducing encoding overhead and implementation complexity. The draft also defines related CBOR-based structures, such as signing requests, private keys, and certificate chains; revocation protocols are under development[1].

To further reduce size, C509 maps verbose OIDs to small integers, omits fields such as the issuer in self-signed certificates, and removes redundant nesting. Compared to IoT-profiled X.509 certificates [23], C509 reports over 50% size savings and claims reduced parsing code and memory requirements.

C509 defines two compatibility models: 1) *Re-encoded certificates* retain the original DER signature and require DER reconstruction for validation, yielding bandwidth savings while maintaining compatibility with legacy PKI; and 2) *Natively signed certificates* compute the signature directly from the CBOR-encoded data, eliminating reconstruction but requiring C509-aware verifiers. Current open-source tooling supports only re-encoded certificates; no public implementation has been identified for natively signed C509 with PQ support.

III. Related Work

a) Federated Space Public Key Infrastructure: Koisser et al. propose TruSat, a decentralised PKI model in which multiple certificate authorities coordinate issuance and revocation via Byzantine consensus, reducing centralisation risks in federated space networks [25]. In another work, V'CER introduces a lightweight revocation framework based on sparse Merkle trees, achieving scalable and decentralised revocation dissemination with less than 3kB storage per device [26]. An SoK by Koisser et al. surveys PKI challenges in satellite networks, highlighting the unsuitability of conventional revocation models under orbital constraints [27]. Smailes et al. developed Keyspace, a simulator assessing PKI performance in interplanetary satellite networks, showing that terrestrial PKI mechanisms can be adapted for space with topology-specific optimisations [28].

b) Post-Quantum Certificates: Raavi et al. empirically assess the computational and storage costs of integrating PQ algorithms into X.509 [29]. Wang et al. qualitatively compare PQ certificate formats, outlining migration strategies and trade-offs between quantum-safe, hybrid, composite, and parallel chain approaches [30]. Ricchizzi et al. address tooling gaps by introducing an open-source CLI for generating hybrid and composite PQ X.509 certificates [31].

c) Certificate Profile Design and Implementation: Forsby et al. proposed XIOT, a CBOR-based certificate profile that hardwires algorithms and prunes X.509 fields for IoT use, achieving substantial size reductions but raising interoperability concerns [32]. Other works expose the complexities of securely implementing X.509: Barenghi et al. formalise its ambiguous grammar [9], Ebalard et al. develop a memory-safe parser with formal RTE guarantees [8], Tatschner et al. analyse in-the-wild parsing divergences [12], and Shi et al. uncover denial-of-service vectors from pathological certificate structures [10]. Formal verification efforts now extend to X.509 validation pipelines (ARMOR) [33] and verified parsers for both DER-encoded [34] and CBOR-based [35] structures.

Despite prior work, certificate profile design remains under-explored in the context of federated space PKI, PQ migration, and operational constraints, gaps that this paper aims to address.

IV. Post-Quantum Certificate Formats

To support the transition to PQ cryptography, several certificate formats have been proposed, including hybrid models that combine traditional and PQ algorithms [36], [37]. This section reviews formats relevant to federated space PKI.

a) Pure PQ Certificates: These contain only PQ keys and signatures, requiring no structural changes to X.509 aside from introducing new algorithm identifiers [38]–[40].

b) Hybrid Composite: Composite certificates combine traditional and PQ components into a single key and signature by concatenation, referenced by a unified OID [41], [42]. This approach enforces simultaneous use of traditional and PQ algorithms, enhancing cryptographic strength.

c) Hybrid with Extensions: The certificates embed PQ components as non-critical X.509 extensions, preserving legacy compatibility by keeping traditional algorithms in the primary fields. This format has been adopted by ITU-T X.509 [11].

d) Hybrid Chameleon and Hybrid Bound: Both use parallel certificate chains, with separate traditional and PQ certificates. Chameleon encodes the differences between the two (e.g., serial number, validity) as a delta descriptor in the base certificate, enabling reconstruction of the PQ certificate when needed [43]. Hybrid Bound links the certificates via a simple hash extension [44]. While these approaches offer flexibility, they significantly increase the complexity of synchronisation management. IETF no longer endorses Hybrid Chameleon.

A. Certificate Size

The size overhead of each format was evaluated using representative samples, summarised in Table I. The certificate samples were created using the tooling referenced in Table I and are available online[2]. For this experiment, the extensions were limited to the mandatory ones imposed by the specific format (if any). The results are consistent with the trends observed in Bouncy Castle's IETF interoperability tests[3]. Results show that the structural overhead, approximated as non-cryptographic content, is minor relative to the cryptographic payload.

Composite certificates incur no real overhead beyond a 6-byte increase from the change in OID. Earlier composite drafts concatenated per-component OIDs [45], which inflated the size. However, current specifications [41], [42] assign a single OID per dual combination, eliminating this redundancy and making prior claims of significant composite overhead [30] outdated.

The Hybrid with Extensions format incurs larger overheads compared to Pure PQ due to extra extension OIDs and nesting. Chameleon certificates add variable overhead depending on the encoded differences between paired certificates, such as

[1] https://github.com/cose-wg/CBOR-certificates

[2] https://doi.org/10.5281/zenodo.17496478

[3] https://github.com/IETF-Hackathon/pqc-certificates

TABLE I: Comparison of format sizes (bytes) for pure `ML-DSA:44` and hybrid variants with `ECDSA:secp256r1`. Body size (bytes) is the non-cryptographic content (total size minus key and signature). Relative Increase shows the size increase (bytes) over the pure PQ certificate. Footnotes indicate the tools used for generation. For Hybrid Bound, sizes are summed across component certificates.

Format	Cert. Size	Body Size	Rel. Increase
Pure [4]	3 894	152	–
Hybrid Composite [4]	4 045	158	6
Hybrid with Extensions [5]	4 112	229	77
Hybrid Chameleon [6]	4 193	310	158
Hybrid Bound (Approx.) [4]	4 247	363	211

serial numbers, validity periods, or subject and issuer names, alongside an additional OID for the delta descriptor. While they mitigate duplication by sharing common fields, this approach is more size-efficient than Hybrid Bound certificates, which retain duplication of both traditional and PQ chains.

Nevertheless, the overheads remain negligible relative to the cryptographic (PQ) payload, making certificate size a minor consideration in format selection for space systems.

B. Backwards Compatibility

Pure and Composite formats are not backwards compatible[7], as they introduce new OIDs unrecognised by legacy systems. The Hybrid with Extensions format embeds PQ components in non-critical fields, allowing legacy validators to ignore them and use traditional cryptography, while updated systems process both components. Parallel-chain schemes use separate traditional and PQ certificates, linked through extensions, allowing systems to negotiate or select the chains they support, thereby enabling flexible phased migration with operational continuity.

C. Certificate Lifecycle

Pure and Composite certificates require no changes to the standard certificate lifecycle. Pure certificates follow the classical model, while composite certificates, although combining two algorithms, are treated as a single unit under a unique OID. As such, if either component is compromised, the entire certificate is revoked, simplifying management by eliminating the need for per-component tracking or selective revocation. Traditional and PQ keys are concatenated within the composite certificates, similarly for signatures. Both traditional and PQ signatures are verified, and if the verification of either of them fails, the certificate is not considered successfully verified.

The Hybrid with Extensions format breaks the traditional model where one certificate binds to one key by embedding an alternative key and signature. It requires a two-step signing process, where validators verify either the primary or alternative component, depending on the extension support, but not both. If the alternative extension can be processed, its verification takes precedence over the primary scheme. Revocation applies to the entire certificate, as individual components cannot be revoked. Once adopted for the root certificate, consistent extension support across the hierarchy is recommended.

Parallel chains introduce the highest lifecycle complexity. Separate issuance, tracking, and validation are needed for each chain, with synchronised signing requests, policies, validity periods, and renewals. Verification becomes fragmented between the two components, requiring linked-certificate discovery [46] and consistent cross-referencing, while the volume of revocation information doubles, increasing operational complexity.

D. Security Considerations

Pure certificates rely solely on PQ algorithms, eliminating downgrade risks but depending entirely on still-maturing PQ schemes. Composite is the only format explicitly designed for cryptographic security during migration, enforcing dual use: both traditional and PQ components must validate. This condition offers resilience against both conventional and quantum adversaries but increases object size and reduces backwards compatibility. Once cryptographically relevant quantum computers (CRQCs) become available, the traditional part becomes obsolete, and composites are replaced by pure PQ certificates.

By contrast, extension-based and parallel-chain hybrids prioritise backwards compatibility over cryptographic strength, embedding PQ components as auxiliary while keeping the primary fields legacy-compatible. The Hybrid Bound standard explicitly states it is not suitable for composite use [44], meaning it does not enforce dual validation. Without strict downgrade protections, relying parties may validate only the traditional algorithm, leaving security tied to the weakest component and exposing the system to downgrade attacks. These formats, therefore, rely on careful validation policy and phased enforcement to ensure PQ protections are adequate during migration.

E. Federation Interoperability in Space

Maintaining backwards compatibility with traditional PKIs poses significant challenges. Pure and composite formats avoid bandwidth waste by design. In contrast, Hybrid with Extensions and Chameleon always transmit PQ components, even to legacy systems that may ignore them, causing unnecessary network overhead. Hybrid Bound may improve efficiency by selectively transmitting either the traditional or PQ certificate, depending on the recipient's capabilities. However, parallel chains double the revocation load, which can be operationally unmanageable.

Nonetheless, backwards compatibility should be secondary to security and interoperability within the federation. Although enabling "existing implementations that will not yet have been updated to support the PQ algorithms" [4] is desirable, legacy PKI infrastructure in space is minimal, if any. Moreover, backwards compatibility conflicts with bandwidth, security, and interoperability requirements. The limited migration timeframe [47] and the current lack of asymmetric cryptography deployment in space motivate prioritising hybrid PQ solutions from the outset.

[4] https://github.com/open-quantum-safe/liboqs

[5] https://github.com/pqcli

[6] https://github.com/CBonnell/chameleon-certs

[7] Backwards compatibility here denotes ensuring operational continuity for legacy systems that have not yet been updated to recognise PQ algorithms.

TABLE II: Proposal for a Minimal Set of Space-Link Certificate Profiles with Extension Configurations.
M – Mandatory, **O** – Optional, *Empty or not present* - Disallowed.

Extension	Self-Signed	Self-Issued	Cross	Intermediate	Signature	Key Exchange
Authority Key Identifier		M	M	M	M	M
Subject Key Identifier	M	M	M	M	M	M
Key Usage	M (critical)	M (critical)	M (critical)	M (critical)	M (critical)	M (critical)
Certificate Policies		M	M	M	M	M
Policy Mappings			M			
Subject Alternative Name	O	O	O	O	O	O
Basic Constraints	M (critical)	M (critical)	M (critical)	M (critical)		
Name Constraints				O (critical)		
Policy Constraints			M (critical)	O (critical)		
Extended Key Usage					O	O
CRL Distribution Points		M	M	M	M	M
Inhibit anyPolicy			M (critical)	O (critical)		
Authority Information Access		M	M	M	M	M
Subject Information Access	M	M	M	M		

Divergent security guidelines pose a challenge to federation-level interoperability. While some agencies, such as BSI [48] and ANSSI [49], [50], recommend hybrid deployments, especially for ML-KEM and ML-DSA, to ensure fallback resilience, others, like the U.S. NSA [51], recommend pure PQ deployments, expressing confidence in standalone security; however, they do not forbid hybridised deployments. This divergence creates a trade-off between interoperability, due to potentially unrecognised OIDs, and local policy autonomy. Moreover, the wide range of possible hybrid combinations amplifies the need for a well-defined specification, as seen in OpenPGP [52], to enable interoperable PQ profiles across the federated space PKI.

Mitigation: Interoperability can be maintained by enforcing a single certificate profile, pure or composite, across the federation. This measure ensures all parties recognise the same OIDs and parameters, eliminating cross-domain mismatches. Standardising the accepted algorithms trades local flexibility for consistent cryptographic behaviour and reduced interoperability risk. However, although both formats are trivially compatible with the X.509 syntax, we argue that for space links (ground-to-space, space-to-space), the entire X.509 Internet profile currently recommended by CCSDS (unbounded in extensions) is not optimal. There is a clear need to standardise a tailored minimal profile specifically for space links.

V. Minimal Federal Profiles and Extension Configurations

As previously argued, the unregulated use of X.509 extensions poses risks to implementation and interoperability in federated PKI. To mitigate this, structured extension usage can be employed. The U.S. Federal Bridge Certification Authority (FBCA) exemplifies this by explicitly categorising extensions as mandatory, optional, or prohibited [53]. Operating under a bridge-architected model (similar to IGCA), FBCA's established profiles ensure consistent extension configurations across the federation. In turn, affiliated PKIs, such as the Foundation for Trusted Identity (FTI), adopt FBCA's profiles verbatim [54], maintaining local autonomy only when compatible with the federation. By disallowing all other extensions, FBCA and its affiliates ensure cross-domain certificate interoperability.

Building on IGCA requirements and the CCSDS Certificate Profile, and drawing from FBCA and FTI, we propose a minimal set of space-link tailored profiles for secure communication in a bridge-based PKI. The extension configurations for each profile are provided in Table II, comprising:

1) *Self-Signed CA Certificate*: root certificate establishing trust anchors, typically distributed out-of-band.
2) *Self-Issued Certificate*: key rollover (link) certificate used for CA key transitions without changing the issuer identity.
3) *Cross Certificate*: certificate issued by one CA to another CA in a different PKI domain to establish cross-domain trust and enable interoperability through policy mapping.
4) *Intermediate Certificate*: CA certificate issued to a subordinate CA, forming a certification chain.
5) *Signature Certificate*: end-entity certificate used for signature verification on data or communications.
6) *Key Exchange Certificate*: end-entity certificate used for key exchange or key agreement protocols.

Despite the benefits of structured extension use, limitations remain. The current set does not cover specialised applications such as code signing or time-stamping, which can be integrated as needed. Additionally, the configuration relies on URI-based extensions, which might not be directly suitable for space. In some missions, the certificate and revocation information can be distributed to spacecrafts via telecommand by ground control, rather than being automatically fetched by the spacecraft from a predefined location. Other missions, with more restrictive contact planning, may decide to deploy dedicated spacecraft elements that can support certificate and revocation information distribution to other spacecrafts/users. Thus, it serves as a preliminary structured foundation for IGCA profile development, recognising that standardisation requires broader alignment.

Generally, constrained space hardware cannot be assumed to be able to support the code, memory, and processing of full path validation [6] or the extensive extensions used in federated profiles [55]. Instead, these endpoints operate within mission-local trust domains, using rigid profiles with fixed algorithms, simplified names (e.g., single common name), no unique identifiers, and minimal extensions, if any. Heavyweight PKI tasks, such as policy mapping, name constraints, and revocation, are delegated to edge gateways or ground proxies [55], [56], which validate, re-encode, or re-sign certificates before interfacing with the broader federation, ensuring end-to-end trust without burdening constrained devices.

VI. c509-native: A Tool for CBOR-Encoded Certificates

The C509 draft defines a compact CBOR-encoded certificate profile aligned with the X.509 information model. Currently, no public tools support natively signed C509 certificates with PQ algorithms. To address this gap, we developed `c509-native`[8], which provides an open-source implementation for generating, parsing, and managing certificates, certificate signing requests (CSRs), and certificate revocation lists (CRLs). The tool intends to serve as a proof-of-concept, not to be deployed as operational code (potentially subject to certification) in future missions.

A. Requirements

Written in C++, the tool is designed to: (i) support both pure PQ (`ML-DSA`, `ML-KEM`) and ECC-PQ hybrid certificates. (ii) ensure deterministic CBOR encoding to achieve reproducible "to-be-signed" payloads. (iii) provide a lightweight implementation that avoids dynamic memory, limits standard-library reliance, and omits heavyweight features such as inheritance, virtual dispatch, and exceptions. (iv) enable an intuitive command-line interface.

B. Design

The tool's modules are illustrated in Figure 1, comprising:

- *CLI*: User interface implemented with `argparse`, mapping commands to core functions; `brotli` compression [57] is integrated for experimental purposes.
- *Core*: Coordinates workflows for key, certificate, CSR and CRL generation and management.
- *Crypto*: Loads `oqs-provider` and interfaces with OpenSSL for traditional and PQ cryptographic operations.
- *Codec*: Performs CBOR serialisation/deserialisation using `zcbor`; a codec (coder-decoder) transforms data between internal and encoded forms.
- *Structure*: Contains models of the C509 schema [7].

C. Command-Line Features

`c509-native` offers four primary commands:

- `genpkey`: Generate PQ, or hybrid key pairs.
- `req`: Generate or process CSRs; issue self-signed or CA-signed certificates.
- `crl`: Manage CRL generation and revocation operations.
- `parse`: Print human-readable details of objects.

The CLI intentionally mirrors OpenSSL to provide a familiar set of arguments and workflows, including options for subjects, validity periods and extensions, limited to the C509 profile.

D. Prototype Status and Availability

Our prototype comprises under 3,000 logical lines of code, with around 60% line and 53% branch test coverage on low-level codec components. While sufficient for this study's evaluations, it lacks advanced CA functions, such as certificate status information tracking, that are present in mature PKI systems. Still, it marks an initial step toward addressing the lack of C509 tooling. Future work will expand testing, algorithm support, and features. The tool is available under the MIT License[9].

[8]https://doi.org/10.5281/zenodo.17496586

[9]https://github.com/rosualinpetru/c509-native

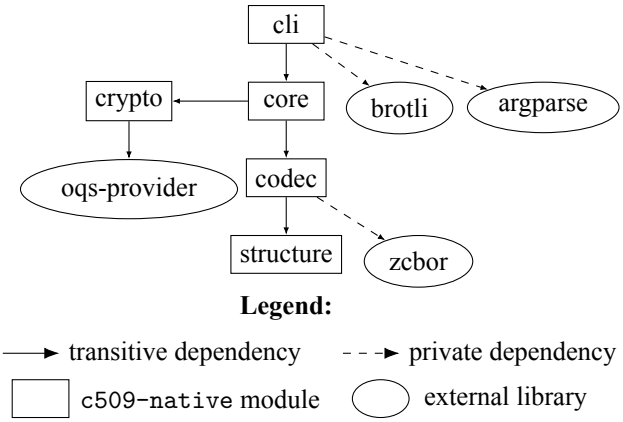

Fig. 1: The Design of `c509-native`.

TABLE III: Cert. sizes (bytes) and Brotli compression rates (%) for X.509 and C509 traditional (ECDSA/ECDH:secp256r1) and PQ (ML-DSA:44), according to section V.

Certificate	Size (bytes)			Compression (%)	
	X.509	*C509*	*Red. (%)*	*X.509*	*C509*
ECDSA/ECDH:secp256r1					
Self-Signed	424	232	45.3	16.7	-1.7
Self-Issued	578	323	44.1	20.9	10.8
Cross	631	358	43.2	16.0	12.3
Intermediate	581	334	42.5	24.6	13.2
Signature	497	290	41.6	14.3	4.5
Key Exchange	491	284	42.1	14.7	1.1
ML-DSA:44					
Self-Signed	4 019	3 855	4.1	1.3	0.0
Self-Issued	4 174	3 946	5.5	3.3	1.1
Cross	4 227	3 981	5.8	3.1	1.3
Intermediate	4 177	3 957	5.3	3.1	1.2
Signature	4 094	3 913	4.4	1.4	0.3
Key Exchange	4 076	3 945	3.2	1.2	0.3

VII. Results

C509 offers reduced certificate size and parser footprint compared to X.509. This section presents an empirical trade-off analysis between the two profiles in these aspects.

A. Object Size

For a direct comparison of size efficiency, X.509 and C509 certificates and revocation lists were generated with identical content, following the profiles defined in section V. The former were created with `openssl`, while the latter were generated with `c509-native`. Both traditional and PQ algorithms were used.

C509 achieves substantial size reductions for traditional certificates, summarised in Table III, with observed savings of 40–45% compared to X.509 (e.g., a self-signed ECDSA certificate drops from 424 bytes to 232 bytes, a 45.3% reduction). While the IETF C509 draft reports reductions of over 50% in some cases, the differences here stem from variations in reference profiles. These gains arise mainly from C509's CBOR encoding, OID removal, and structural optimisations, which eliminate redundancies inherent to DER-encoded X.509.

TABLE IV: Structural encoding size comparison (in bytes) between X.509-DER and C509-CBOR encodings for a cross certificate using ECDSA:secp256r1. Each internal node is mapped to the number of bytes required to represent that element. Leaves of the tree are mapped to the total number of bytes contained by that element. Leaves contain identical concrete values.

X.509	C509	DER	CBOR
Certificate	Certificate	4	1
├── tbsCertificate	│	4	0
│ ├── version	├── version	5	1
│ ├── serialNumber	├── serialNumber	3	2
│ ├── signature	├── signatureAlgorithm	12	1
│ ├── issuer (commonName)	├── issuer (commonName)	28	16
│ ├── validity	│	2	0
│ │ ├── notBefore	├── notBefore	17	9
│ │ └── notAfter	├── notAfter	17	9
│ ├── subject (commonName)	├── subject (commonName)	29	17
│ ├── subjectPublicKeyInfo	│	2	0
│ │ ├── algorithm	├── publicKeyAlgorithm	21	1
│ │ └── subjectPublicKey	├── publicKeyValue	68	67
│ └── extensions	├── extensions	8	1
│ ├── keyUsage	│ ├── keyUsage	16	3
│ ├── basicConstraints	│ ├── basicConstraints	14	2
│ ├── subjectKeyIdentifier	│ ├── subjectKeyIdentifier	31	22
│ ├── authorityKeyIdentifier	│ ├── authorityKeyIdentifier	33	22
│ ├── crlDistributionPoints	│ ├── crlDistributionPoints	42	26
│ ├── authorityInformationAccess	│ ├── authorityInformationAccess	58	33
│ ├── subjectInformationAccess	│ ├── subjectInformationAccess	58	34
│ ├── certificatePolicies	│ ├── certificatePolicies	18	6
│ ├── policyMappings	│ ├── policyMappings	23	11
│ ├── policyConstraints	│ ├── policyConstraints	17	5
│ └── inhibitAnyPolicy	│ └── inhibitAnyPolicy	15	3
├── signatureAlgorithm	│	12	0
└── signatureValue	└── signatureValue	74	66

To demonstrate the in-depth optimisation performed by C509, a field-level breakdown for cross-certificates is provided in Table IV. The analysis reveals consistent per-field reductions: `version` and `serialNumber` are simplified through representation and CBOR encoding; `issuer` and `subject` are nearly halved by eliminating redundant DER wrappers and replacing verbose OIDs with compact integers. Extensions achieve notable savings via flattened encoding and integrated criticality markers, while validity timestamps are halved by using epoch-based integers instead of `GeneralizedTime` strings. Some fields are omitted or deduplicated entirely. Although cryptographic payloads yield limited savings, algorithm-specific optimisations such as point compression provide marginal improvements. Overall, the cross-certificate size is reduced by 43.2%, underscoring C509's effectiveness in eliminating structural and syntactic overhead while preserving semantic content.

Inspired by the C509 draft, Brotli compression (quality 11, window size 22) [57] was applied to both profiles to quantify syntactic redundancy. This approach leverages the principle that compressing equivalent semantic content under identical settings reveals structural redundancy via residual gains. As shown in Table III, X.509 certificates compress by 14–25%, while C509 achieves only 0–13%, reflecting proximity to its entropy floor. For example, DER's repetitive OID patterns (e.g., 2.5.29.) are replaced by compact integers in C509. Notably, Brotli's fixed overhead disproportionately impacts small, low-redundancy C509 objects, yielding negative rates. In practice, compression can help minimise transmission overhead.

TABLE V: Absolute sizes and relative reductions (bytes) for C509 and X.509 pure PQ/hybrid composite end-entity certificates according to section V, across ML-DSA ± ECDSA:secp256r1 and ML-KEM ± ECDH:secp256r1.

Signature	Public Key	X.509	C509	Difference
Security Level 1/2				
mldsa44	mldsa44	4 094	3 913	181
mldsa44	mlkem512	3 576	3 395	181
mldsa44_p256	mldsa44_p256	4 259	4 064	195
mldsa44_p256	mlkem512	3 664	3 479	185
Security Level 3				
mldsa65	mldsa65	5 623	5 442	181
mldsa65	mlkem768	4 849	4 668	181
mldsa65_p256	mldsa65_p256	5 790	5 593	197
mldsa65_p256	mlkem768	4 936	4 753	183
Security Level 5				
mldsa87	mldsa87	7 581	7 400	181
mldsa87	mlkem1024	6 551	6 370	181
mldsa87_p384	mldsa87_p384	7 811	7 616	195
mldsa87_p384	mlkem1024	6 670	6 487	183

For PQ certificates, C509's size savings show a profile-specific upper bound (Table V). Across all tested cases, including pure and hybrid composites with ML-DSA, ML-KEM, and ECDSA, the absolute reduction ranges from 180 to 197 bytes for end-entity certificates. However, this becomes marginal relative to the large PQ keys and signatures that dominate overall size.

TABLE VI: CRL sizes (bytes) and Brotli compression rates (%) signed with traditional (ECDSA with secp256r1) and PQ (ML-DSA:44) algorithms, based on the IETF preliminary schema[10].

Revocations	Size (bytes)			Compression (%)	
	DER	*CBOR*	*Red. (%)*	*DER*	*CBOR*
ECDSA/ECDH with secp256r1					
1	183	107	41.5	0.5	-3.7
10	413	197	52.3	26.2	-2.0
100	2 664	1 098	58.8	54.8	6.0
1 000	25 159	10 100	59.9	60.8	10.7
10 000	250 118	100 099	60.0	62.1	12.2
20 000	500 066	200 099	60.0	62.4	12.3
30 000	750 035	300 100	60.0	62.5	12.3
ML-DSA:44					
1	2 538	2 466	2.8	0.6	-0.2
10	2 766	2 556	7.6	3.0	0.0
100	5 017	3 457	31.1	28.7	1.0
1 000	27 512	12 458	54.7	55.6	8.6
10 000	252 471	102 458	59.4	61.5	11.8
20 000	502 419	202 458	59.7	62.1	12.1
30 000	752 388	302 458	59.8	62.3	12.2

Thus, while C509 offers substantial savings for traditional certificates, its impact on PQ certificate size is inherently limited by the cryptographic primitives, which are not compressible.

CBOR-encoded CRLs achieve substantial size reductions over DER-encoded counterparts, regardless of the used cryptography, demonstrated in Table VI. For PQ signatures, relative reductions grow sharply (e.g., from approximately 3% with a single revoked certificate to nearly 60% at 30,000 entries). These gains stem primarily from CBOR's efficient time encoding and flattened per-entry structure. Notably, while PQ signatures introduce significant absolute size, they reside only in the final signature block, making their contribution negligible relative to the cumulative size of large CRLs. Still, even with these reductions, large CRLs remain sizeable in absolute terms.

C509 reduces PKI message overhead, providing a compact, consistent encoding across the stack. While size gains are modest for PQ certificates, CBOR significantly reduces revocation list sizes, regardless of the cryptography used.

B. Software Complexity

C509 represents a purposefully restricted subset of X.509, retaining only essential features. Coupled with its simplified encoding, it demands a more straightforward implementation. CBOR defines eight major self-describing types, comparable to JSON [15], whereas ASN.1 DER involves four tag classes, over 25 universal types, and schema-dependent tagging and parsing [20]. From the outset, C509 is expected to be simpler to implement [7]. This quantitative analysis does not aim to prove that claim, but rather to measure the extent to which such simplification is actually reflected in practice.

Corpus: This evaluation considered representative X.509 and C509 implementations with equivalent functionality for certificate and revocation list parsing and serialisation. For X.509: (i) `x509-parser`[11], an open-source, custom-built C parser by ANSSI, formally verified with Frama-C and ACSL-instrumented, providing parser-only functionality including DER decoding; and (ii) `ASN1C (gen)`[12], an industrial-grade C parser and serialiser generated from ASN.1 schema using the `asn1c` compiler, delivered with a pre-compiled BER/DER library. For C509: (i) `c509-native`, the proposed custom-built (C-like) C++ implementation optimised for embedded systems, using `zcbor` for CBOR encoding; and (ii) `zcbor (gen)`[13], generated low-footprint C encoders/decoders from C509 data schemas using the open-source `zcbor` tool.

Experimental Setup: Three settings were covered:

- *Setting 1*: custom-built, parser-only including the DER/CBOR layer: `x509-parser`, `c509-native`;
- *Setting 2*: generated, parser-only excluding the DER/CBOR layer: `ASN1C`, `zcbor`;
- *Setting 3*: generated, parser and serialiser excluding the DER/CBOR layer: `ASN1C`, `zcbor`.

These experimental setups enabled consistent comparisons across custom and automatically generated code while isolating the impact of encoding layers.

Heuristic Metrics: Well-established static-analysis heuristics serve as practical indicators of software complexity. Such metrics are routinely employed in industry [58], [59] to approximate the effort required to understand, maintain, and test code, with some formally recommended in the European Cooperation for Space Standardisation (ECSS) guidelines for space systems implementations of all criticality levels [60]. The selected metrics include logical lines of code (LLOC) for codebase size, cyclomatic complexity (CCN) for control-flow complexity, Halstead volume for token-level complexity, Halstead difficulty for comprehension effort, and function count for contextualising modularity [61], [62]. As the metrics are computed per function, aggregation is needed to characterise implementations holistically. Values are summarised using the sum, mean, or median, based on the metric's nature.

Tooling: Two independent static analysers were used to extract code complexity metrics across the selected corpus: (i) `Jarod42/ccccc`[14], a tool specialised for C/C++ codebases; and (ii) `mozilla/rust-code-analysis`[15], a multi-language static analyser [63]. Both enable the mentioned metrics. While absolute values may differ due to tool implementation, observed trend consistency reinforces the reliability of the analysis.

Results for the considered experimental setups are summarised in Table VII.

Setting 1: Regarding LLOC, `c509-native` implementation remains significantly smaller than `x509-parser`, with *Tool 1* showing average reduction of 78.9% across the two tools. For CCN, `c509-native` consistently reports lower values, with total CCN dropping by factors of $\approx 3\times$ in both tools. The

[10] https://github.com/cose-wg/CBOR-certificates/blob/master/draft-cose-cbor-revocation-management.md

[11] https://github.com/ANSSI-FR/x509-parser

[12] https://www.obj-sys.com/products/asn1c/index.php

[13] https://github.com/NordicSemiconductor/zcbor

[14] https://github.com/Jarod42/ccccc

[15] https://github.com/mozilla/rust-code-analysis

TABLE VII: Comparison of certificate and CRL parser(-serialiser) implementations across the mentioned experimental settings.

Implementation	Total LLOC	Mean CCN	Total CCN	Total Volume	Median Diff	Q-95 Diff	Func Count
Setting 1: Parser-only including binary encoding layer (Tool 1: ccccc)							
`x509-parser`	8 019	7.80	1 622	243 775.46	28.67	62.28	208
`c509-native`	1 939	2.99	535	66 309.72	7.88	25.14	179
Setting 1: Parser-only including binary encoding layer (Tool 2: rust-code-analysis)							
`x509-parser`	7 432	7.69	1 630	214 955.33	26.39	56.57	212
`c509-native`	1 346	3.34	565	55 192.19	7.50	25.23	169
Setting 2: Parser-only excluding binary encoding layer (Tool 1: ccccc)							
`ASN1C (gen)`	4 611	6.73	1 090	181 210.87	12.23	52.73	162
`zcbor (gen)`	538	9.02	379	47 710.39	12.29	16.36	42
Setting 2: Parser-only excluding binary encoding layer (Tool 2: rust-code-analysis)							
`ASN1C (gen)`	4 421	6.73	1 090	161 474.18	10.80	47.62	162
`zcbor (gen)`	243	10.21	429	35 456.66	10.47	14.31	42
Setting 3: Parser and serialiser excluding binary encoding layer (Tool 1: ccccc)							
`ASN1C (gen)`	6 991	7.09	1 752	277 721.04	19.72	48.92	247
`zcbor (gen)`	1 041	7.70	647	89 113.70	11.84	16.36	84
Setting 3: Parser and serialiser excluding binary encoding layer (Tool 2: rust-code-analysis)							
`ASN1C (gen)`	6 389	7.09	1 752	247 644.75	17.60	45.33	247
`zcbor (gen)`	441	8.94	751	65 836.81	10.45	14.80	84

reductions in LLOC and CCN suggest that C509 parsers offer smaller codebases with greatly simplified control flow, enhancing verifiability and testability. Regarding Halstead metrics, the notable decreases in total volume and aggregated difficulty values indicate a reduction in token-level complexity for C509. These results underscore that C509 parsers not only shrink code size but can also improve conceptual clarity and maintainability.

Setting 2: This experiment isolates the complexity introduced purely by the feature sets of each certificate format. The parsers are schema-generated, ensuring no human optimisation influences the results. By excluding the binary-encoding layers, the codebase reflects only feature-level parsing logic. Under these conditions, `zcbor (gen)` shows order-of-magnitude reductions across total-value metrics compared to `ASN1C (gen)`, highlighting the structural simplicity of C509. Given the inherent complexity of DER relative to CBOR, the actual implementation gap might be even larger.

Setting 3: Extending the feature set to include serialisation preserves the ranking of implementations while exposing differences in scalability. All projects show increases in code size and complexity, with growth more pronounced in `ASN1C (gen)`, widening the absolute gap in LLOC and total CCN. While `zcbor (gen)` implementation nearly doubles their LLOC, they remain substantially smaller than `ASN1C (gen)`. Halstead Volume grows across all, disproportionately: `ASN1C (gen)` increases by about 50%, while `zcbor (gen)` variant grows by 85–95%. Despite this, C509 parsers and serialisers maintain a 3–5$\times$ smaller token footprint than their X.509 counterparts.

VIII. Discussion

Assuming C509 adoption, mission designers must choose between natively signed and re-encoded certificates. Natively signed certificates apply signatures directly over CBOR-encoded data, eliminating ASN.1/DER dependencies. This approach suits closed ecosystems but lacks practicality in federated environments that rely on interoperability with standard X.509-based PKIs. In contrast, re-encoded certificates enable transformations to achieve compatibility with traditional X.509-based PKIs. Rather than signing the CBOR-encoded certificate directly, the CBOR format serves as an intermediate representation of the same data structure originally signed in DER.

The C509 draft does not prescribe where or by whom re-encoding must occur, allowing system designers to align placement with operational, trust, and resource constraints. A common architecture offloads re-encoding to a border gateway, such as an element in the ground segment, which acts as an intermediary between constrained environments (space elements) and the broader ground network. This approach enables the constrained device to pay only the light penalty of parsing compact CBOR representations, instead of the costly DER parsing. The gateway handles conversion between X.509 and C509, and vice versa, and consequently pays the heavier cost of DER parsing. In other words, the heavy cost of heavy parsing is shifted from a resource-constrained spacecraft to a ground gateway.

Under such deployments, beyond reducing transmission size, re-encoded C509 can eliminate the need for X.509/DER parsing. As shown in Figure 2, the gateway follows the steps indicated by the blue arrows. First, it parses the X.509 certificate, typically the most complex and resource-intensive step, and extracts the Abstract Data Type (ADT). The ADT is then serialised into CBOR and sent to the constrained client.

The constrained device, in turn, follows the steps illustrated by the yellow arrows. Upon reception of the CBOR-encoded certificate, it first performs lightweight CBOR parsing. While signature verification still requires DER serialisation, this operation is more straightforward than DER parsing, as it can leverage

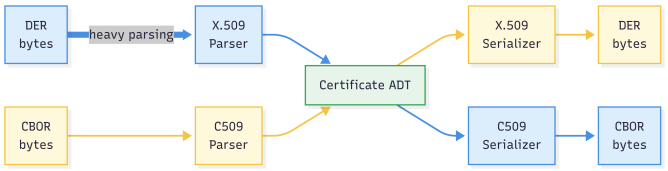

Fig. 2: X.509 vs. C509 Parsing and Serialising.

the ADT's structured context. Depending on peer requirements, the device can serialise certificates in CBOR or DER as needed.

It is worth noting that not all X.509 certificates are convertible to C509, as C509 represents a constrained subset of X.509. However, this can be addressed by enforcing a restricted X.509 profile to ensure round-trip re-encoding remains possible.

While C509 offers clear benefits in bandwidth and implementation simplicity, its adoption is limited by ecosystem immaturity. Full integration requires a CBOR-based infrastructure, which is currently underdeveloped. As C509 is still an IETF draft, and CBOR-encoded PQ certificates and revocation formats (e.g., CRLs, OCSP) are not yet standardised, the lack of normative completeness hinders industry uptake and limits suitability for space-standardisation efforts.

IX. Future Work

While this study focuses on defining a minimal and interoperable certificate profile, the greater challenge is ensuring its efficient validation across heterogeneous nodes, from constrained satellites to ground systems. Limited computation, intermittent links, and unreliable time references render onboard path construction, revocation checks, and policy enforcement disproportionately demanding, making certificate validation a significant bottleneck for operational trust.

A promising mitigation is the Server-based Certificate Validation Protocol (SCVP) [64], which enables delegated validation (DPV) or path discovery (DPD) to a trusted server [65]. SCVP offers particular advantages: it offloads computational burdens, enables signature verification across differing cryptographic stacks, and centralises policy enforcement. Validation policies can mandate anchors, revocation sources, and required extensions, promoting cross-domain consistency.

SCVP meets key security requirements: messages can be signed or MAC-protected, include nonces to prevent replay, and allow client-specified time references, relevant for delay-tolerant networks. SCVP supports relayed requests, such as those from lunar relays to terrestrial authorities, thereby reducing the need for clients to process CRLs or OCSP responses directly. It is already supported by commercial solutions [66]–[68] and has seen practical use in mobile networks [69].

Nevertheless, further research is needed to assess SCVP's performance and trust implications in space. Future work should involve prototyping a bridge validation authority within an IGCA-aligned federation, evaluating it under simulated space conditions, and developing tooling to manage and apply SCVP policies. SCVP appears to be a promising enabler for scalable, policy-driven validation in quantum-ready federated space PKI.

X. Conclusion

Future federated space missions must balance constrained onboard resources, pressure from PQ migration, and cross-domain interoperability. First, this paper evaluated PQ certificate formats and their trade-offs in a federated space setting, outlined a minimal, structured extension profile for space links, drawing from terrestrial federated PKIs. Then, the work performed a quantitative comparative analysis between X.509 and C509 in terms of message size and software complexity. Together, the findings aim to inform and support standardisation bodies, such as CCSDS Security Working Groups, in tailoring a certificate profile for federated PKI, with a focus on space links.

Our results confirm that X.509's verbosity and implementation complexity hinder its use in constrained systems. C509 mitigates these issues through compact CBOR encoding and reduced structural overhead, yielding 40–45% size reductions for traditional certificates and up to 60% for revocation lists, regardless of the employed cryptography. For PQ and hybrid certificates, the gains remain negligible.

Using well-established heuristics and practical experiments, this work quantifies the substantial difference in software complexity between X.509 and C509 implementations. The results demonstrate an approximately 80% reduced codebase footprint when using C509, as well as a 2-3$\times$ reduced cyclomatic complexity and a decrease of over 60% for total Halstead volume (lower cognitive complexity metric). The reduction in software complexity is perhaps more relevant than the size gains, especially considering the high demands of space software qualification requirements and security certification requirements.

Among C509 deployment options, re-encoded certificates with gateway-based translation offer a practical compromise. They preserve X.509 compatibility while offloading DER parsing from constrained clients. However, broader adoption remains limited by the draft status of C509 and the absence of standardised CBOR-native revocation protocols.

Furthermore, this paper reviewed PQ certificate formats and highlighted that regulatory divergence, e.g., between composite and pure approaches, poses interoperability risks. This work advocates for federation-wide support of composite certificates to ensure dual-algorithm trust during PQ migration.

Finally, certificate validation, not encoding, is the next bottleneck. Delegated models such as SCVP can offload path discovery and policy enforcement to trusted intermediaries. Standardising a minimal, IGCA-aligned profile and validation architecture is essential to enable secure, scalable, and quantum-ready trust infrastructures for future space systems.

Acknowledgements

The authors gratefully acknowledge Objective Systems Inc. for providing a temporary licence to their toolset and the support of the Cybersecurity Research Group at Delft University of Technology as an essential enabler of this research.

References

[1] N. O. of Inspector General, "NASA's Management of the Artemis Missions," National Aeronautics and Space Administration (NASA), Tech. Rep. IG-22-003, November 2021.

[2] Consultative Committee for Space Data Systems (CCSDS), *Symmetric Key Management*, ser. CCSDS Recommended Practice. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), December 2023, no. 354.0-M-1.

[3] ——, *Space Missions Key Management Concept*, ser. CCSDS Informational Report. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), November 2011, no. 350.6-G-1.

[4] ——, *Intergovernmental Certification Authority*, ser. CCSDS Experimental Specification. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), December 2024, no. 357.1-O-1.

[5] ——, *Authentication Credentials*, ser. CCSDS Recommended Standard. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), July 2019, no. 357.0-B-1.

[6] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," Internet Engineering Task Force (IETF), RFC 5280, May 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5280

[7] J. P. Mattsson, G. Selander, S. Raza, J. Höglund, and M. Furuhed, "CBOR Encoded X.509 Certificates (C509 Certificates)," Internet Engineering Task Force (IETF), Internet-Draft, March 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-cose-cbor-encoded-cert/

[8] A. Ebalard, P. Mouy, and R. Benadjila, "Journey to a RTE-free X.509 parser," in *Proceedings of the Symposium on Information and Communications Technology Security (SSTIC)*, Rennes, France, 2019, pp. 1–20.

[9] A. Barenghi, N. Mainardi, and G. Pelosi, "Systematic Parsing of X.509: Eradicating Security Issues with a Parse Tree," *Journal of Computer Security*, vol. 26, no. 6, pp. 817–849, 2018.

[10] B. Shi, W. Li, Y. Wang, X. Bai, and L. Xing, "X.509DoS: Exploiting and Detecting Denial-of-Service Vulnerabilities in Cryptographic Libraries using Crafted X.509 Certificates," in *Proceedings of the 34th USENIX Security Symposium*. Seattle, WA: USENIX Association, 2025, pp. –.

[11] International Telecommunication Union, "Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks," (ITU-T), Tech. Rep. X.509, October 2019.

[12] S. Tatschner, S. N. Peters, M. P. Heinl, T. Specht, and T. Newe, "ParsEval: Evaluation of Parsing Behavior using Real-world Out-in-the-wild X.509 Certificates," in *Proceedings of the 19th International Conference on Availability, Reliability and Security*, ser. ARES '24. Vienna, Austria: Association for Computing Machinery, 2024, pp. 143:1–143:9.

[13] OpenSSL Project, "Deprecated Extensions in `x509v3_config`," OpenSSL Documentation, Version 3.5, 2025, Accessed: June 2025. [Online]. Available: https://docs.openssl.org/3.5/man5/x509v3_config/#deprecated-extensions

[14] European Commission, "A coordinated implementation roadmap for the transition to post-quantum cryptography," European Commission, Tech. Rep. Part 1, Version 1.1, 6 2025, Accessed: June 2025. [Online]. Available: https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography

[15] C. Bormann and P. Hoffman, "Concise Binary Object Representation (CBOR)," Internet Engineering Task Force (IETF), RFC 8949, December 2020. [Online]. Available: https://www.rfc-editor.org/info/rfc8949

[16] C. Bormann, M. Ersue, and A. Keränen, "Terminology for Constrained-Node Networks," Internet Engineering Task Force (IETF), RFC 7228, May 2014. [Online]. Available: https://www.rfc-editor.org/info/rfc7228

[17] Protiviti Government Services, "Federal Public Key Infrastructure (FPKI) Concept of Operations," U.S. General Services Administration, Washington, D.C., Tech. Rep. Version 2.0.0, October 2016.

[18] R. Prodanović, I. Vulić, and I. Tot, "A Survey of PKI Architecture," in *ERAZ 2019 – Selected Papers*, Belgrade, Serbia, 2019, pp. 169–175.

[19] C. Connolly, P. van Dijk, F. Vierboom, and S. Wilson, "PKI Interoperability Models," Galexia, Technical Report, February 2005, Accessed: June 2025. [Online]. Available: http://www.galexia.com/public/research/articles/research_articles-art32.html

[20] International Telecommunication Union (ITU-T), "ITU-T X.690: Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)," International Telecommunication Union (ITU), ITU-T Recommendation X.690, February 2021, accessed: 2025-05-01. [Online]. Available: https://www.itu.int/rec/T-REC-X.690

[21] S. Santesson, M. Nystrom, and T. Polk, "Internet X.509 Public Key Infrastructure: Qualified Certificates Profile," Internet Engineering Task Force (IETF), RFC 3739, March 2004. [Online]. Available: https://www.rfc-editor.org/info/rfc3739

[22] S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson, "Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile," Internet Engineering Task Force (IETF), RFC 3820, June 2004. [Online]. Available: https://www.rfc-editor.org/info/rfc3820

[23] H. Tschofenig and T. Fossati, "Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things," Internet Engineering Task Force (IETF), RFC 7925, July 2016. [Online]. Available: https://www.rfc-editor.org/info/rfc7925

[24] J. Peterson and S. Turner, "Secure Telephone Identity Credentials: Certificates," Internet Engineering Task Force (IETF), RFC 8226, February 2018. [Online]. Available: https://www.rfc-editor.org/info/rfc8226

[25] D. Koisser, D. Fischer, M. Wallum, and A.-R. Sadeghi, "TruSat: Building Cyber Trust in Collaborative Spacecraft Networks," in *2022 IEEE Aerospace Conference (AERO)*. Big Sky, MT: IEEE, 2022, pp. 1–12.

[26] D. Koisser, P. Jauernig, G. Tsudik, and A.-R. Sadeghi, "V'CER: Efficient Certificate Validation in Constrained Networks," in *Proceedings of the 31st USENIX Security Symposium*. Boston, MA: USENIX Association, 2022, pp. 4491–4508.

[27] D. Koisser, R. Mitev, N. Yadav, F. Vollmer, and A.-R. Sadeghi, "Orbital Trust and Privacy: SoK on PKI and Location Privacy Challenges in Space Networks," in *Proceedings of the 33rd USENIX Security Symposium*. Philadelphia, PA: USENIX Association, 2024, pp. 6093–6111.

[28] J. Smailes, S. Köhler, S. Birnbach, M. Strohmeier, and I. Martinovic, "KeySpace: Public Key Infrastructure Considerations in Interplanetary Networks," 2024, Accessed: June 2025. [Online]. Available: https://arxiv.org/abs/2408.10963

[29] M. Raavi, P. Chandramouli, S. Wuthier, X. Zhou, and S.-Y. Chang, "Performance Characterization of Post-Quantum Digital Certificates," in *2021 International Conference on Computer Communications and Networks (ICCCN)*, Athens, Greece, 2021, pp. 1–9.

[30] C. Wang, W. Xue, and J. Wang, "Integration of Quantum-Safe Algorithms into X.509v3 Certificates," in *2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI)*, Changchun, China, 2023, pp. 384–388. [Online]. Available: https://doi.org/10.1109/ICETCI57876.2023.10176713

[31] G. Ricchizzi, V. Temin, S. Spini, D. Kugler, M. Meier, and S. Gürgens, "Industrial Post-Quantum Identity Management with Hybrid and Composite Certificates," 2025, accessed: June 2025. [Online]. Available: https://arxiv.org/abs/2505.04333

[32] F. Forsby, M. Furuhed, P. Papadimitratos, and S. Raza, "Lightweight X.509 Digital Certificates for the Internet of Things," in *Interoperability, Safety and Security in IoT*. Cham: Springer International Publishing, 2018, pp. 123–133.

[33] J. Debnath, C. Jenkins, Y. Sun, S. Y. Chau, and O. Chowdhury, "ARMOR: A Formally Verified Implementation of X.509 Certificate Chain Validation," in *IEEE Symposium on Security and Privacy, SP 2024, San Francisco, CA, USA, May 19-23, 2024*. IEEE, 2024, pp. 1462–1480.

[34] T. Ramananandro, A. Delignat-Lavaud, C. Fournet, N. Swamy, T. Chajed, N. Kobeissi, and J. Protzenko, "EverParse: Verified Secure Zero-Copy Parsers for Authenticated Message Formats," in *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*. USENIX Association, 2019, pp. 1465–1482.

[35] T. Ramananandro, G. Ebner, G. Martínez, and N. Swamy, "Secure Parsing and Serializing with Separation Logic Applied to CBOR, CDDL, and COSE," 2025, Accessed: June 2025. [Online]. Available: https://arxiv.org/abs/2505.17335

[36] W. Newhouse, M. Souppaya, W. Barker, C. Brown, P. Kampanakis, M. Manzano, D. McGrew, A. Dames, V. Soukharev, P. Lafrance, A. Hu, D. Hook, R. Garcia, E. Gervis, E. Kim, and C. Lee, "Migration to Post-Quantum Cryptography: Preparation for Considering the Implementation and Adoption of Quantum Safe Cryptography," National Institute of Standards and Technology (NIST), NIST Special Publication 1800-38, December 2023, Accessed: June 2025. [Online]. Available: https://csrc.nist.gov/pubs/sp/1800/38/iprd-(1)

[37] F. Driscoll, M. Parsons, and B. Hale, "Terminology for Post-Quantum Traditional Hybrid Schemes," Internet Engineering Task Force (IETF), Internet-Draft, January 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-pquip-pqt-hybrid-terminology

[38] S. Turner, P. Kampanakis, J. Massimo, and B. Westerbaan, "Internet X.509 Public Key Infrastructure: Algorithm Identifiers for ML-KEM," Internet Engineering Task Force (IETF), Internet-Draft, April 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-lamps-kyber-certificates

[39] J. Massimo, P. Kampanakis, S. Turner, and B. E. Westerbaan, "Internet X.509 Public Key Infrastructure: Algorithm Identifiers for ML-DSA," Internet Engineering Task Force (IETF), Internet-Draft, April 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-lamps-dilithium-certificates

[40] K. Bashiri, S. Fluhrer, S. Gazdag, D. Van Geest, and S. Kousidis, "Internet X.509 Public Key Infrastructure: Algorithm Identifiers for SLH-DSA," Internet Engineering Task Force (IETF), Internet-Draft, May 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-lamps-x509-slhdsa

[41] M. Ounsworth, J. Gray, M. Pala, J. Klaußner, and S. Fluhrer, "Composite ML-DSA for use in X.509 Public Key Infrastructure and CMS," Internet Engineering Task Force (IETF), Internet-Draft, March 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-lamps-pq-composite-sigs

[42] ——, "Composite ML-KEM for use in X.509 Public Key Infrastructure and CMS," Internet Engineering Task Force (IETF), Internet-Draft, March 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-lamps-pq-composite-kem

[43] C. Bonnell, J. Gray, D. Hook, T. Okubo, and M. Ounsworth, "A Mechanism for Encoding Differences in Paired Certificates," Internet Engineering Task Force (IETF), Internet-Draft, April 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-bonnell-lamps-chameleon-certs

[44] A. Becker, R. Guthrie, and M. J. Jenkins, "Related Certificates for Use in Multiple Authentications within a Protocol," Internet Engineering Task Force (IETF), RFC 9763, June 2025. [Online]. Available: https://www.rfc-editor.org/info/rfc9763

[45] M. Ounsworth, J. Gray, M. Pala, and J. Klaußner, "Composite Public and Private Keys For Use In Internet PKI," Internet Engineering Task Force (IETF), Internet-Draft draft-ounsworth-pq-composite-keys, July 2021, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ounsworth-pq-composite-keys

[46] T. Okubo, C. Bonnell, J. Gray, M. Ounsworth, and J. Mandel, "A Mechanism for X.509 Certificate Discovery," Internet Engineering Task Force (IETF), Internet-Draft draft-ietf-lamps-certdiscovery, 4 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-lamps-certdiscovery

[47] W. Burr, N. Lefkovitz, K. Scarfone, and D. Cooper, "NIST Migration to Post-Quantum Cryptography," National Institute of Standards and Technology (NIST), NIST Interagency Report IR 8547 (Initial Public Draft), 3 2024, Accessed: June 2025. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8547.ipd.pdf

[48] F. O. for Information Security (BSI), "Cryptographic Mechanisms: Recommendations and Key Lengths," Federal Office for Information Security (BSI), Technical Guideline TR-02102-1, 3 2025, Accessed: June 2025. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TG02102/BSI-TR-02102-1.pdf

[49] M. Rossi, "PQC Transition in France: ANSSI Views," Agence nationale de la sécurité des systèmes d'information (ANSSI), Technical Report, 3 2023, Accessed: June 2025. [Online]. Available: https://cyber.gouv.fr/sites/default/files/document/pqc-transition-in-france.pdf

[50] M. Rossi, A. Dupont, J. Lefèvre, and S. Moreau, "ANSSI views on the Post-Quantum Cryptography transition (2023 follow up)," Agence nationale de la sécurité des systèmes d'information (ANSSI), Technical Report, 12 2023, Accessed: June 2025. [Online]. Available: https://cyber.gouv.fr/sites/default/files/document/follow_up_position_paper_on_post_quantum_cryptography.pdf

[51] National Security Agency, "CNSA Suite 2.0 and Quantum Computing FAQ," National Security Agency, Fort Meade, MD, Cybersecurity Information Sheet U/OO/194427-22 | PP-24-4014, 12 2024, Accessed: June 2025. [Online]. Available: https://media.defense.gov/2022/Sep/07/2003071836/-1/-1/0/CSI_CNSA_2.0_FAQ_.PDF

[52] S. Kousidis, J. Roth, F. Strenzke, and A. Wussler, "Post-Quantum Cryptography in OpenPGP," Internet Engineering Task Force (IETF), Internet-Draft draft-ietf-openpgp-pqc, 4 2025, Accessed: June 2025. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-openpgp-pqc

[53] Federal Public Key Infrastructure Policy Authority, "Federal Bridge Certification Authority (FBCA) X.509 Certificate and CRL Extensions Profile," U.S. Federal PKI Policy Authority, Tech. Rep., 10 2022, Accessed: June 2025. [Online]. Available: https://www.idmanagement.gov/docs/fpki-x509-cert-profiles-fbca.pdf

[54] Booz Allen Hamilton Inc. and National Institute of Standards and Technology, "Federal public key infrastructure (pki) x.509 certificate and crl extensions profile," Federal PKI Policy Authority, Tech. Rep., 10 2005, Accessed: June 2025. [Online]. Available: https://www.foundationfortrustedidentity.org/federally-certified/uploads/fti-ca/Federal%20Public%20Key%20Infrastructure%20(PKI)%20X.509%20Certificate%20and%20CRL%20Extensions%20Profile.pdf

[55] Consultative Committee for Space Data Systems (CCSDS), *Rationale, Scenarios, and Requirements for DTN in Space*, ser. CCSDS Informational Report. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), 8 2010, no. 734.0-G-1.

[56] ——, *The Application of Security to CCSDS Protocols*, ser. CCSDS Informational Report. Washington, DC, USA: CCSDS Secretariat, National Aeronautics and Space Administration (NASA), 3 2019, no. 350.0-G-3.

[57] J. Alakuijala and Z. Szabadka, "Brotli Compressed Data Format," Internet Engineering Task Force (IETF), RFC 7932, 7 2016. [Online]. Available: https://www.rfc-editor.org/info/rfc7932

[58] SonarSource. (2024) Understanding measures and metrics. SonarQube Server 10.8 documentation, accessed 3 July 2025. [Online]. Available: https://docs.sonarsource.com/sonarqube-server/10.8/user-guide/code-metrics/metrics-definition/

[59] Scientific Toolworks, Inc. (2021) Metrics overview. Accessed: June 2025. [Online]. Available: https://support.scitools.com/support/solutions/articles/70000582289-metrics-overview

[60] *ECSS-Q-HB-80-04A: Space Product Assurance — Software Metrication Programme Definition and Implementation*, European Cooperation for Space Standardization (ECSS), Noordwijk, The Netherlands, 3 2011, Accessed: June 2025. [Online]. Available: https://ecss.nl/wp-content/uploads/handbooks/ecss-q-hb/ECSS-Q-HB-80-04A30March2011.pdf

[61] *IEEE Standard Dictionary of Measures to Produce Reliable Software*, IEEE Computer Society Std. IEEE Std 982.1-1988, 1989.

[62] *IEEE Guide for the Use of IEEE Standard Dictionary of Measures to Produce Reliable Software*, IEEE Computer Society Std. IEEE Std 982.2-1988, 1989.

[63] L. Ardito, L. Barbato, M. Castelluccio, R. Coppola, C. Denizet, S. Ledru, and M. Valsesia, "rust-code-analysis: A rust library to analyze and extract maintainability information from source codes," *SoftwareX*, vol. 12, p. 100635, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352711020303484

[64] T. Freeman, R. Housley, A. Malpani, D. Cooper, and W. Polk, "Server-Based Certificate Validation Protocol (SCVP)," Internet Engineering Task Force (IETF), RFC 5055, 12 2007. [Online]. Available: https://www.rfc-editor.org/info/rfc5055

[65] D. Pinkas and R. Housley, "Delegated Path Validation and Delegated Path Discovery Protocol Requirements," Internet Engineering Task Force (IETF), RFC 3379, 9 2002. [Online]. Available: https://www.rfc-editor.org/info/rfc3379

[66] HID Global. Actividd validation authority: Scvp support. Accessed: June 2025. [Online]. Available: https://docs.hidglobal.com/actividd-validation-authority-v7.4/docs/overview/scvp-validation.htm

[67] Ascertia. Adss scvp server. Accessed: June 2025. [Online]. Available: https://www.ascertia.com/products/adss-scvp-server/

[68] Axway. Validation authority suite. Accessed: June 2025. [Online]. Available: https://www.axway.com/en/products/axway-va-suite

[69] Motorola Inc., "Utilizing a stapling technique with a server-based certificate validation protocol," US Patent Application US20 130 159 703A1, 2013, Accessed: June 2025. [Online]. Available: https://patents.google.com/patent/US20130159703A1/en