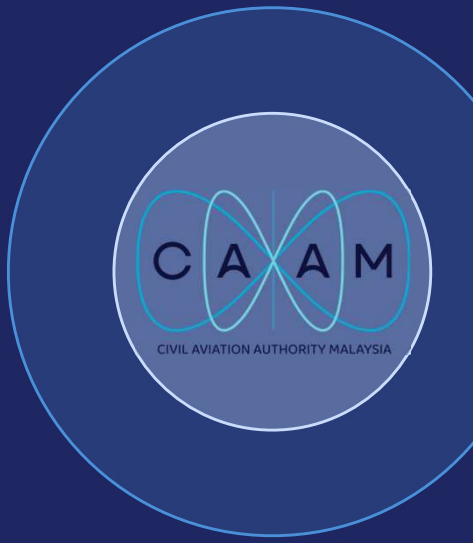
Task 6: SWIM Security Requirements

PKI Trust Infrastructure & Message-Level Security

Bangkok, Thailand | 1 June 2026
SWIM Implementation Pioneer Group



OVERVIEW

Agenda

01 **Background & ICAO Framework**

Task 6 objectives • ACCP Doc 10169 • MAIS Doc 10204

02 **Testbed Architecture & PKI**

Hierarchical CA model • Certificate lifecycle • Trust federation

03 **Security Testing Results**

S/MIME 4.0 signing & encryption • AMQP 1.0 • Performance metrics

04 **Standards Alignment**

ACCP / MAIS compliance • EUROCONTROL TI Yellow Profile

05 **Trust Framework & Way Forward**

ACCP TFI model • Bridge CA • APAC regional PKI roadmap

01 BACKGROUND

Task 6 Background & ICAO Framework

Task 6 — SWIM Security Requirements

- Establish security requirements for SWIM data exchange across APAC states
- Define a federated PKI trust model aligned with ICAO standards
- Test S/MIME 4.0 message-level security over AMQP 1.0 protocol
- Validate mutual TLS (mTLS) and SASL EXTERNAL transport security
- Develop a certificate lifecycle management framework for production
- Map requirements to ACCP (Doc 10169) and MAIS (Doc 10204)

ACCP — Doc 10169

Aviation Common Certificate Policy
First Edition 2025 | Baseline PKI requirements for civil aviation digital identity, TFI model, CA governance & certificate policy framework

MAIS — Doc 10204

Manual on Aviation Information Security
First Edition (Advance) 2024 | CIA objectives; Identity & Access Management; information security for ATM systems

EUROCONTROL Spec 170

SWIM Technical Infrastructure (TI) Yellow Profile
Mandates S/MIME 4.0 (RFC 8551) for message-level security in SWIM service exchanges

ICAO MAIS — Security Objectives for SWIM

ICAO Doc 10204 establishes Confidentiality, Integrity, and Availability (CIA) as the foundational pillars of aviation information security. SWIM exchanges must satisfy all three objectives.

C — Confidentiality

Only authorised entities may access SWIM payload content.

Implemented via: S/MIME 4.0 EnvelopedData encryption (AES-256-CBC), AMQPS channel encryption (TLS 1.2+), and SASL EXTERNAL authentication.

MAIS 1.1.2.1

I — Integrity & Non-Repudiation

SWIM messages must not be altered in transit; sender identity must be provable.

Implemented via: S/MIME 4.0 SignedData (RSA + SHA-256), signingTime authenticated attribute, X.509 certificate binding per ACCP.

MAIS 1.1.2.2

A — Availability

SWIM services must be accessible to authorised users when needed.

Implemented via: CRL and OCSP for timely revocation; automated trust store updates; broker resilience tested under load.

MAIS 1.1.2.3

02 Testbed Architecture & PKI

Hierarchical CA model • Certificate lifecycle • Trust federation

02 ARCHITECTURE

What Was Tested

Mutual TLS (mTLS)

End-to-end PKI-secured transport over AMQP 1.0. Both broker and client present X.509 certificates. SASL EXTERNAL used for broker authentication without passwords.

Payload Types Tested

JSON, XML, and plain text payloads simulating SWIM operational data: FIXM flight data, IWXXM meteorological messages, and AIXM aeronautical information.*

S/MIME 4.0 Message Security

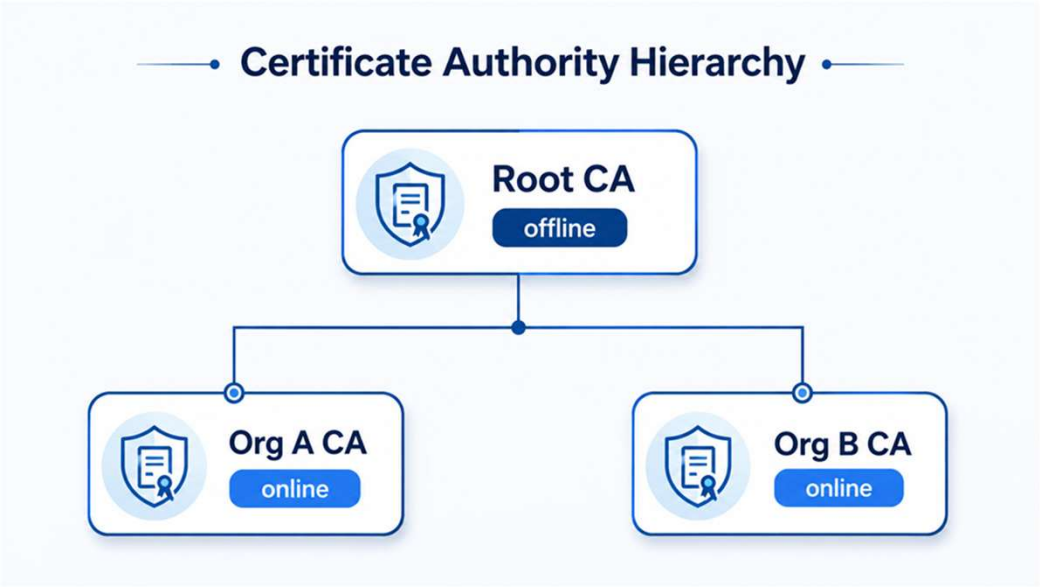
Applied S/MIME 4.0 (RFC 8551) to structured operational payloads. Signing (integrity + non-repudiation) and encryption (confidentiality) tested independently and combined.*

Transport Protocol

AMQP 1.0 plain and TLS variants. AMQPS with Mutual TLS. Mode signalling via AMQP ***content_type*** property - no out-of-band configuration required.

Testbed PKI Structure

The testbed modelled a federated PKI with two separate organisations — Malaysia (CAAM) and Singapore (CAAS) — each operating their own Certificate Authority, anchored to a common offline Root CA.



Cross-certification between CAAM CA and CAAS CA enables mutual trust without changes to the Root CA. Onboarding new states requires only a new Entity CA signed by the same Root.

PKI Roles & Responsibilities

Root CA (ACCP 6.2)

- Offline trust anchor — private key never used online
- Signs only intermediate CA certificates
- Stored in air-gapped or HSM-protected environment
- Root CA cert distributed to all participants as trust anchor
- Compromise invalidates entire trust chain — highest-value secret

Entity CA — Intermediate CA (ACCP 4)

- Online CA operated independently per organisation/State
- Issues and revokes end-entity certificates for SWIM nodes
- Maintains CRL and/or OCSP responder for revocation status
- Responsible for subscriber vetting and certificate policy compliance
- Manages its own key lifecycle per ACCP 6.1

End Entity / SWIM Node

- Holds a certificate issued by its Entity CA
- Uses certificate for S/MIME signing and mTLS
- Trusts Root CA and all cross-signed Entity CA certs
- Validates inbound certs against CRL/OCSP before accepting messages

Certificate Exchange: Testbed vs Production

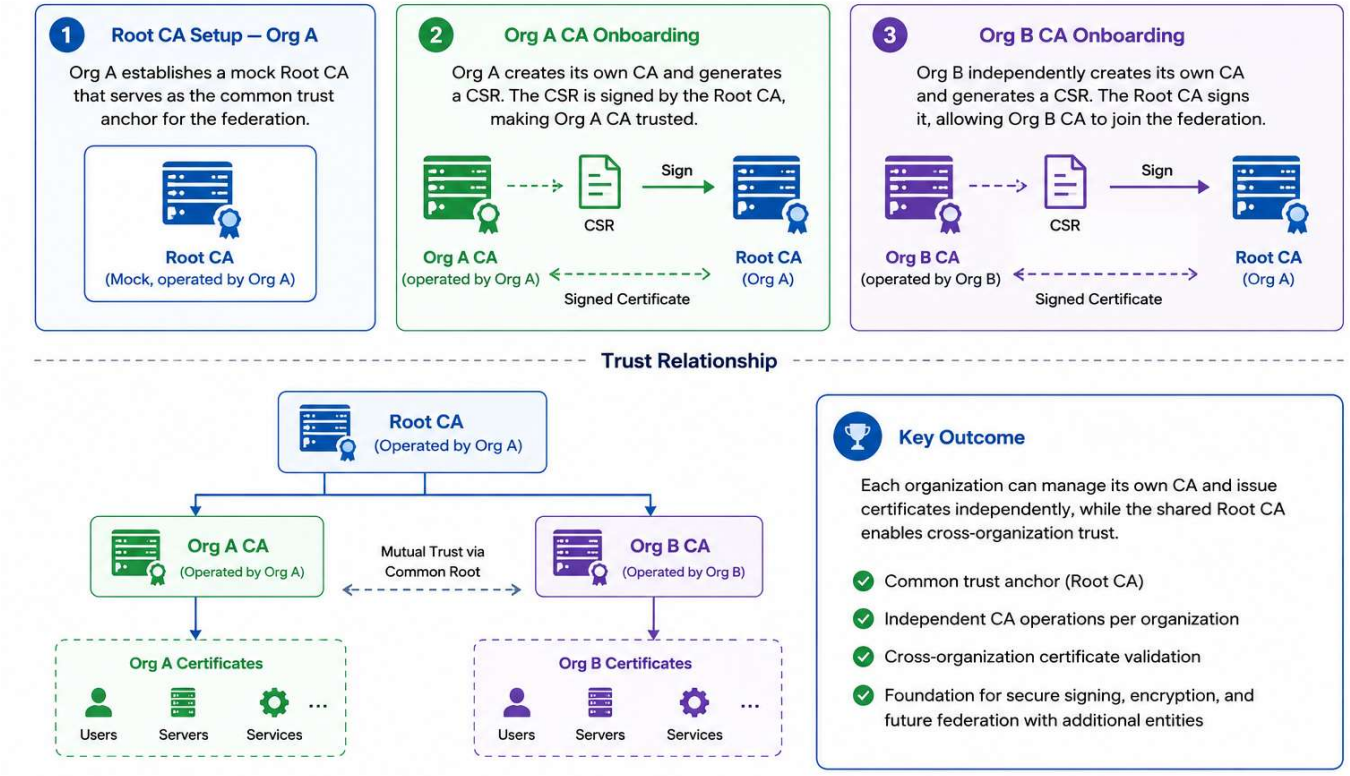
Aspect	Testbed Approach	Production Requirement
Certificate Exchange	Manual via email	LDAP directory or ACME/EST automated enrolment
Integrity Guarantee	None — cert could be substituted	Digital signature on CSR; verified by CA before signing
Discoverability	Out-of-band request required	LDAP/directory lookup or SDS (Service Discovery)
Revocation Visibility	No automated check	CRL Distribution Points (CDP) + OCSP embedded in cert (ACCP 4)
Scalability	Fails beyond ~5 parties	LDAP + BCA supports hundreds of participants

Recommendation: Each Entity CA publishes its CA certificate to an LDAP directory. CRL/OCSP URLs are embedded in all issued certificates per ACCP 4 to enable automated revocation checking.

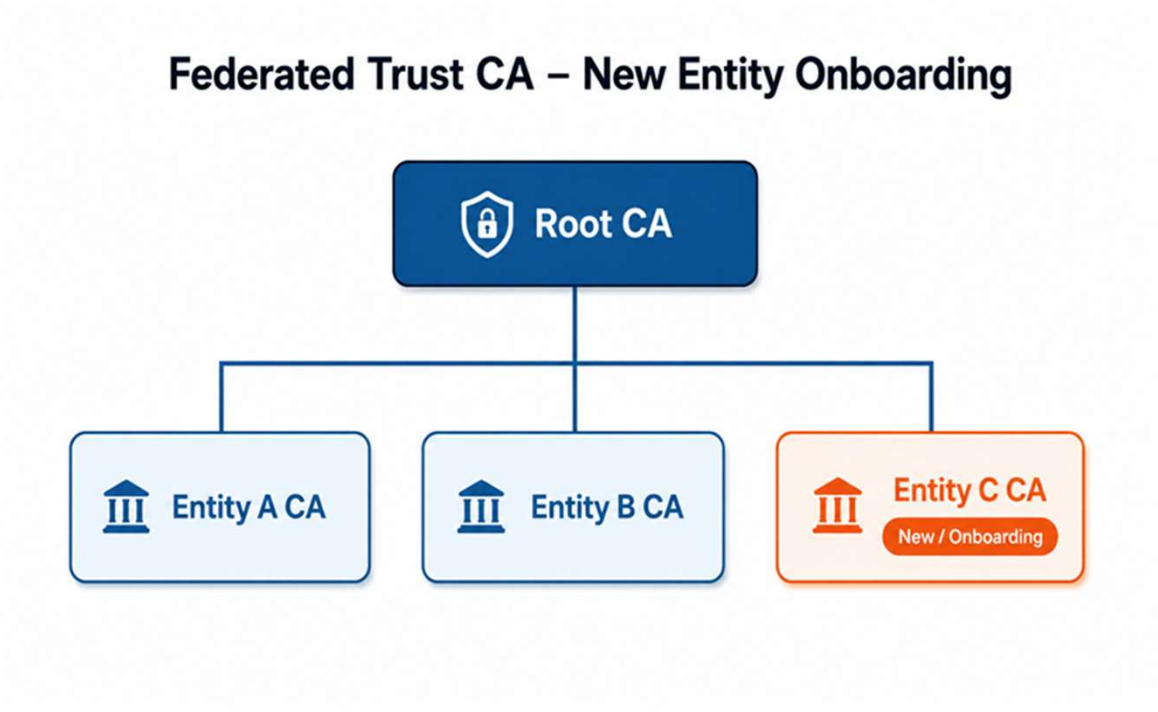
03 Security Testing Results

S/MIME 4.0 signing & encryption • AMQP 1.0 • Performance metrics

Federated PKI Establishment



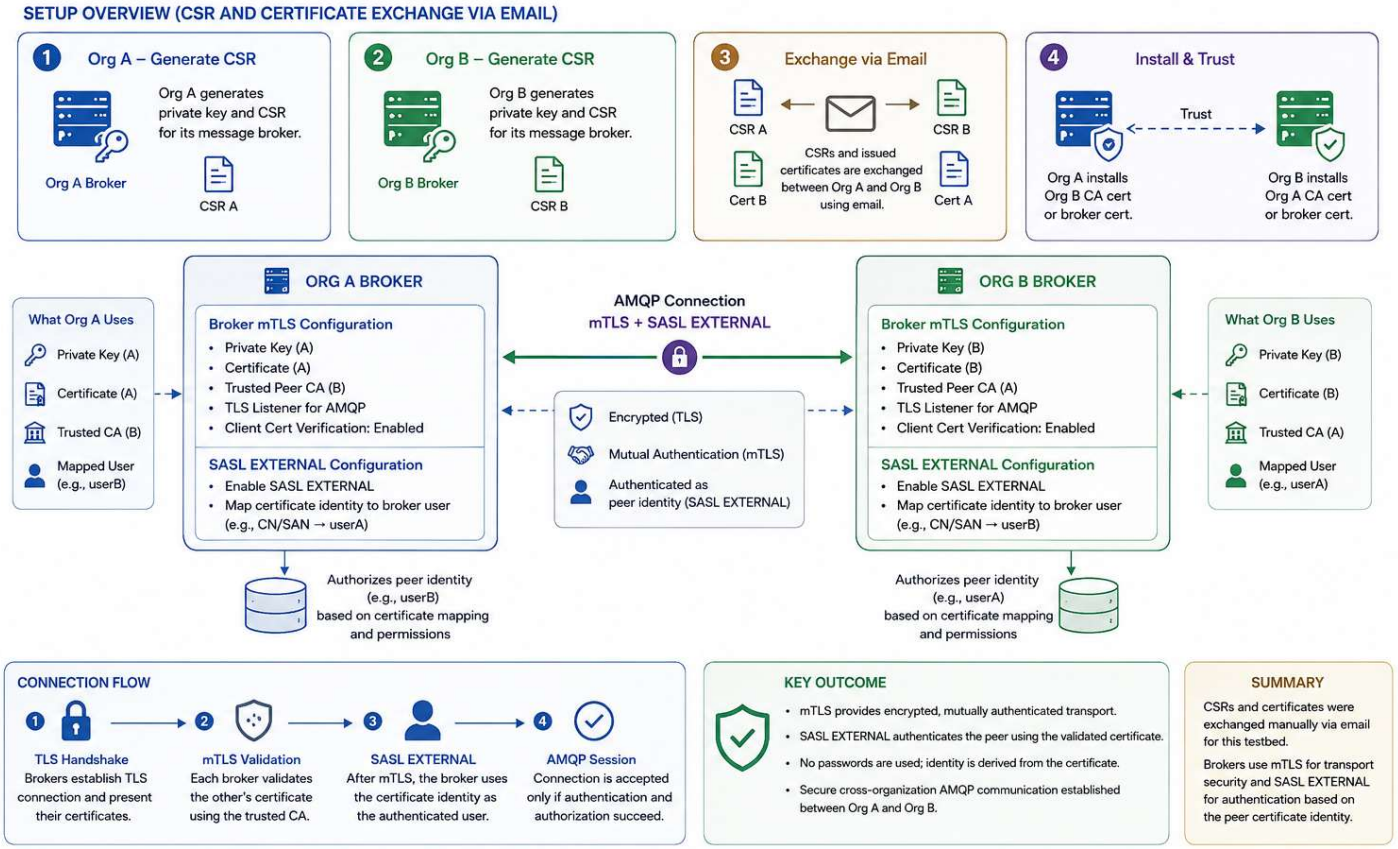
Trust Federation – Adding a New Participant



Steps to onboard Entity C

- Entity C generates its own CA key pair
- Entity C submits a Certificate Signing Request (CSR) to the Root CA operator
- Root CA signs Entity C's CA certificate
- Entity C can now issue end-entity certificates trusted by A and B - with no changes to A or B's configuration

Message Broker Authentication

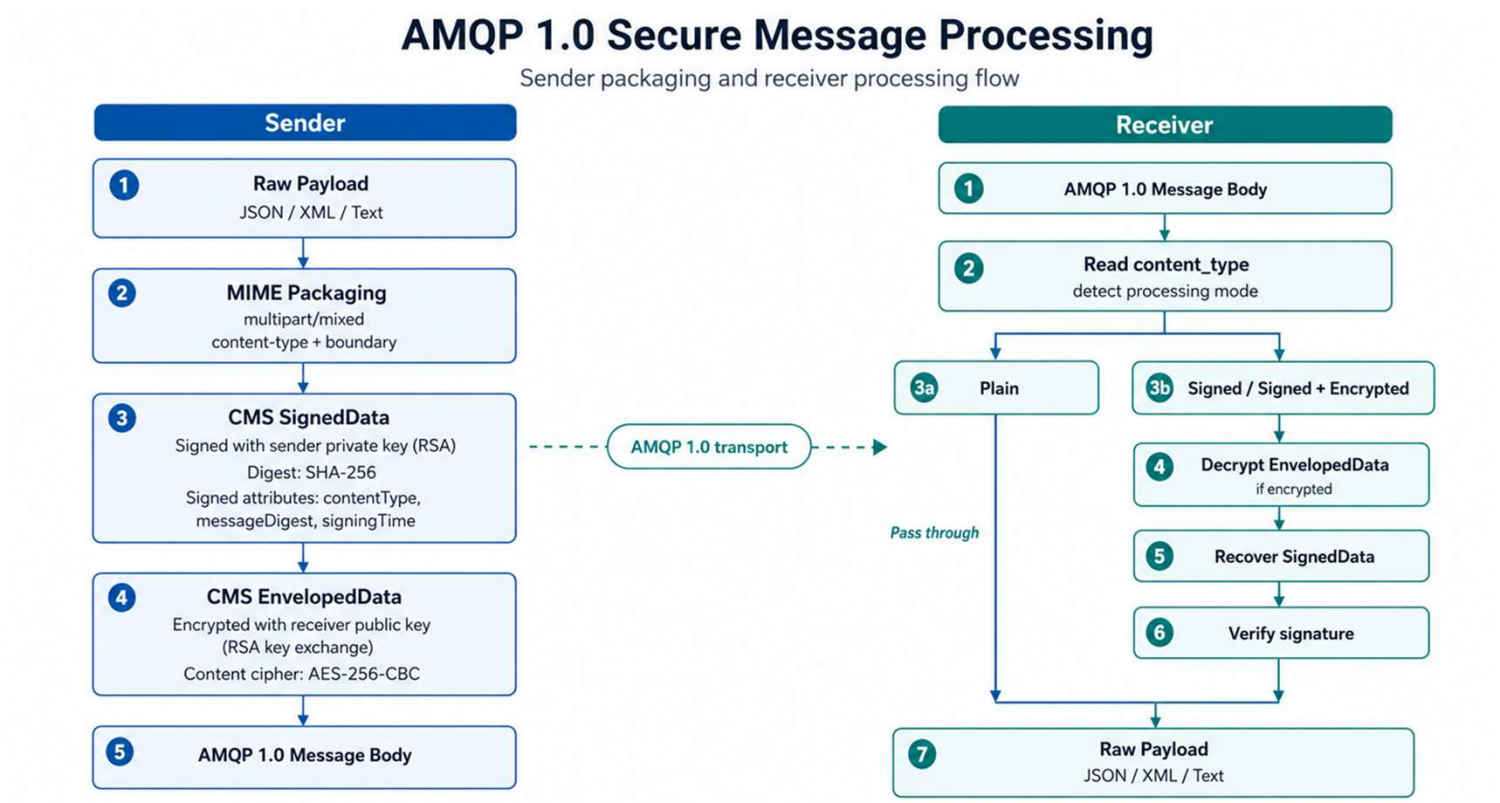


Message Exchange: Security Modes Tested

plain	signed	signed+encrypted
No signing or encryption. AMQP content_type: application/octet-stream. Optional SHA-256 checksum in AMQP application_properties for basic corruption detection only.	S/MIME 4.0 PKCS#7 SignedData. RSA private key + SHA-256 digest. signingTime attribute embedded. Sender certificate included in the signature block.	Sign-then-encrypt (recommended S/MIME order). SignedData wrapped in PKCS#9 EnvelopedData. AES-256-CBC content cipher, RSA key transport. Signature hidden inside encryption envelope.
Use when: Internal low-sensitivity data. Not suitable for inter-state SWIM exchange.	**Use when:** Default baseline for all SWIM operational exchanges. Provides integrity, authenticity, and non-repudiation.	**Use when:** Required for sensitive, classified, or PII data. Recommended for cross-organisation exchange.

Mode detection: AMQP content_type property signals the security mode — a standard AMQP 1.0 property. No out-of-band configuration required.

Message Exchange: Message Pipeline — Signed + Encrypted



Message Exchange: AMQP 1.0 Message Structure

```
AMQP 1.0 Message

Properties
  content_type : "multipart/mixed"                    ← plain
                 "application/pkcs7-mime;             ← signed
                  smime-type=signed-data"
                 "application/pkcs7-mime;             ← signed+
                  smime-type=enveloped-data"            encrypted

Application Properties
  MIME-Version         : "1.0"                (always present)
  X-Checksum-SHA256    : "<sha256hex>"        (plain + --checksum)
  APAC_SOURCE          : "WM_CAAM"            (from APAC HEADERS)
  APAC_RECIPIENT_LIST  : "WS_CAAS"            (from APAC HEADERS)
  APAC_CATEGORY        : "WM_CAAM"            (from APAC HEADERS)
  APAC_CATEGORY_VERSION : "FIXM_4_3_FF_ICE" (from APAC HEADERS)
  APAC_MESSAGE_TYPE    : "FILED_FLIGHT_PLAN"
  FFICE_PHASE          : "FILED"
  ... (any other header)

Body  (UTF-8 bytes)

── plain ──────────────────────────────────────────────────
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="-----=_Part_<hex>"

-------=_Part_<hex>
Content-Type: application/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit

<?xml ...>  ← raw payload
-------=_Part_<hex>--

── signed ─────────────────────────────────────────────────
MIME-Version: 1.0
Content-Type: application/pkcs7-mime; smime-type=signed-data;
  name="smime.p7m"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7m"

-----BEGIN PKCS7-----
<CMS SignedData — contains:
  • signer certificate
  • RSASSA-PKCS1-v1_5/SHA-256 signature
  • encapsulated multipart/mixed MIME (plain body above)>
-----END PKCS7-----

── signed+encrypted ───────────────────────────────────────
MIME-Version: 1.0
Content-Type: application/pkcs7-mime; smime-type=enveloped-data;
  name="smime.p7m"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7m"

-----BEGIN PKCS7-----
<CMS EnvelopedData — contains:
  • RecipientInfo: encrypted AES-256 session key (RSA-OAEP/SHA-1)
  • AES-256-CBC ciphertext of the CMS SignedData above>
-----END PKCS7-----
```

- Properties
  - *content_type: In AMQP 1.0, content-type is an optional field in the Properties section of a message used to describe the RFC-2046 MIME type of the application data. It is primarily intended for cases where the message body is sent as opaque binary data (the Data section) rather than native AMQP-encoded types*
- Application Properties (Additional)
  - *MIME-Version*: *EUROCONTROL Specification for SWIM Technical Infrastructure (TI) Yellow Profile*
  - X-Checksum-SHA256: *(Optional) For checking against accidental message corruption.*
- Body
  - Content-Type and boundary tag (Note: S/MIME 4.0)

03 TESTING

Performance Metrics — Overhead & Latency

Payload Size Overhead

Mode	Original	After Processing	Overhead
Plain	256 B	256 B	0%
Signed	256 B	~2.8 KB	~1000%
Signed+Encrypted	256 B	~3.4 KB	~1230%
Signed (10 KB payload)	10 KB	~12.8 KB	~28%

Key finding: Overhead is largely fixed (cert embedding + ASN.1 + base64). For payloads > 10 KB the relative overhead drops to < 30% and becomes operationally negligible.

Cryptographic Latency (per message)

< 5 ms — **Signing latency**
Modern hardware, small payload

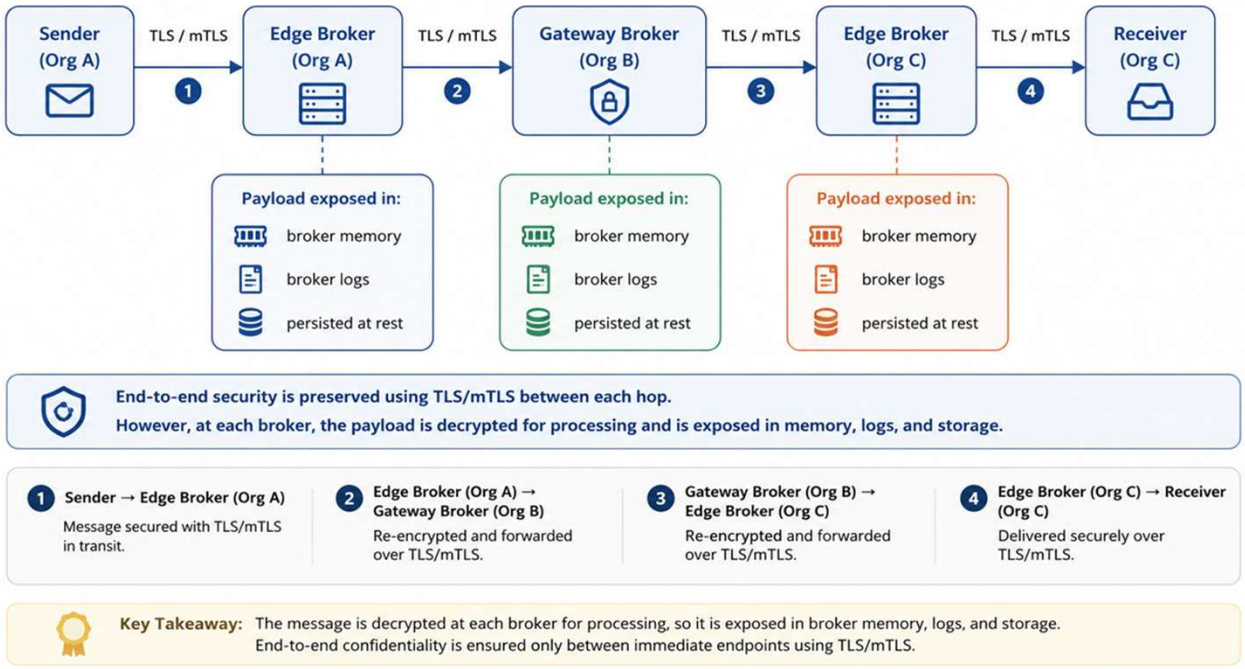
< 5 ms — **Verification latency**
RSA public-key operation

< 10 ms — **Encrypt + decrypt**
AES-256-CBC + RSA key wrap

For high-frequency scenarios, consider HSM acceleration or ECDSA/ECDH migration.

Why Message-Level Security Matters — Broker Hops

In a multi-organisation SWIM deployment, messages traverse multiple brokers. TLS secures only each individual link. S/MIME protects the message itself — regardless of how many brokers it passes through.



	TLS Only	TLS + S/MIME (Recommended)
What broker sees	Full plaintext — in memory, logs, and at rest	Encrypted ciphertext — broker cannot read
After TLS terminates	Payload fully exposed at broker	Still protected by S/MIME envelope
Signing survives broker hop?	N/A — no signature	Yes — original S/MIME signature intact end-to-end

03 TESTING

Transport Security — AMQPS + SASL EXTERNAL

In addition to message-level S/MIME security, the transport layer was tested with AMQPS and mutual TLS. These are complementary layers — both must be deployed in production.

AMQPS (AMQP over TLS)

Encrypts the communication channel between client and broker. Prevents passive eavesdropping. TLS 1.2 minimum; TLS 1.3 recommended for new deployments.

SASL EXTERNAL

Broker authenticates the client using the TLS client certificate — no separate password required. Simplifies credential management across multiple SWIM nodes.

Mutual TLS (mTLS)

Both broker and client present X.509 certificates. Ensures only authorised SWIM nodes can connect. Eliminates password-based broker authentication entirely.

Layered Defence

TLS protects the channel. S/MIME protects the message. Both layers must be deployed together. A compromised broker still cannot read S/MIME-encrypted payloads.

04 Standards Alignment

ACCP / MAIS compliance • S/MIME 4.0 • EUROCONTROL TI Yellow Profile

Why S/MIME 4.0 — Standards Rationale

S/MIME Version Comparison

Feature	S/MIME 3.x	S/MIME 4.0 (RFC 8551)
Minimum RSA Key size	1024-bit	2048-bit
Hash algorithm	SHA-1 (deprecated)	SHA-256 minimum
Symmetric cipher	3DES, RC2 (weak)	AES-128/256-CBC
Key exchange	RSA only	RSA + ECDH supported
EUROCONTROL Spec 170 (Yellow Profile)	**Not compliant**	**Required**

Why not a custom format? Rolling your own crypto envelope creates long-term interoperability and audit risk. S/MIME 4.0 is a well-audited, vendor-neutral envelope that any conformant SWIM system can process.

EUROCONTROL Spec 170 SWIM TI Yellow Profile

- Mandates S/MIME 4.0 (RFC 8551) for SWIM message-level security
- Formally retires SHA-1, RC2, 3DES from SWIM deployments
- Requires AES-128 or AES-256-CBC for symmetric encryption
- Testbed implementation is fully compliant with Spec 170
- Aligns with ICAO ACCP 7 cryptographic algorithm requirements

Compliance Mapping — ACCP & MAIS

ACCP / MAIS Requirement	Reference	Testbed Implementation	Status
Hierarchical CA with offline Root	ACCP 1.3, 6.2	Root CA (offline) + CAAM/CAAS Entity CAs via EJBCA	✓ Met
Certificate revocation via CRL/OCSP	ACCP 4	CRL and OCSP endpoints configured; URLs embedded in certs	✓ Met
SHA-256 + AES-256 cryptography	ACCP 7, MAIS 4	RSA + SHA-256 signing; AES-256-CBC encryption; S/MIME 4.0	✓ Met
Message integrity & non-repudiation	MAIS 1.1.2.2	S/MIME SignedData with signingTime attribute	✓ Met
Payload confidentiality	MAIS 1.1.2.1	S/MIME EnvelopedData (AES-256-CBC) for sensitive payloads	✓ Met
Trust Framework Instance (TFI)	ACCP 1.3.1	Testbed models TFI; full Bridge CA under design for Phase 2	⚠ Partial
HSM for CA private key protection	ACCP 6.2.1	Software key store used in testbed; HSM required for production	⚠ Deferred

05 Trust Framework & Way Forward

ACCP TFI model

ACCP Trust Framework Instance (TFI) Model

ICAO ACCP Doc 10169 promotes a peer-to-peer Trust Framework Instance (TFI) where APAC states form a trust ecosystem with a shared Trust Validation Anchor (TVA) — either a Bridge CA (BCA) or a Trust List (TL).

★ Recommended — Bridge Certificate Authority (BCA)

Advantages

- Handles policy differences between CAs
- Scalable for many participants (proven: US Federal Bridge CA)
- Zero reconfiguration when adding new APAC states

Limitations

- Complex infrastructure (dedicated CA required)
- Higher governance overhead; requires formal TFI agreements

Trust List (TL)

Advantages

- Simple to set up — publish and maintain a list
- No complex infrastructure; works well for small/medium groups
- Easier governance for initial deployment

Limitations

- Inconsistent validation at scale; no CA policy translation
- Managing updates across many participants is error-prone

05 WAY FORWARD

Key Findings

01 S/MIME 4.0 over AMQP 1.0 is technically feasible and operationally viable for SWIM inter-state exchanges. Overhead is fixed and manageable for typical ATM operational payload sizes.

02 AMQP content_type is the correct mechanism for security mode signalling — a standard AMQP 1.0 property requiring no out-of-band configuration.

03 Sign-then-encrypt is the correct S/MIME order. The signature is protected inside the encryption envelope, preventing metadata leakage from the signature wrapper.

04 Transport TLS and message-level S/MIME serve different threat models. Both must be deployed together in production.

05 The hierarchical PKI model (Root CA → Entity CA) is aligned with ACCP Doc 10169. Trust federation between APAC states is achieved without changes to the Root CA.

06 Certificate lifecycle management (issuance, renewal, revocation via CRL/OCSP) was successfully validated. Production requires HSM-protected CA keys and automated revocation checking.

05 WAY FORWARD

Recommendations

HIGH	Adopt signed mode as the mandatory baseline for all operational SWIM inter-state message exchanges.
HIGH	Deploy AMQPS with Mutual TLS for all broker connections. Use SASL EXTERNAL to eliminate password-based authentication.
HIGH	Adopt S/MIME 4.0 (RFC 8551) exclusively. Prohibit SHA-1 and 3DES in all new SWIM deployments — ACCP 7 + EUROCONTROL Spec 170.
MED	Use signed+encrypted mode when payloads contain sensitive, classified, or privacy-constrained operational data.
MED	Maintain a dedicated Entity CA per trust domain / State. Do not share CA keys across organisations or test/production environments.
MED	Embed CRL Distribution Points (CDP) and OCSP URLs in all issued certificates to enable automated revocation per ACCP 4.
LOW	Migrate to ECDSA/ECDH for high-frequency message scenarios to reduce CPU overhead. Deploy HSM for CA key protection in production (ACCP 6.2.1).

Thank You

SIPG Task 6 — SWIM Security Requirements

Questions & Discussion