mollie

Mollie Internal PKI — Certification Practice Statement (CPS)

Version 1.2

Mollie B.V.

mollie

COLOPHON	
Title	Mollie Internal PKI — Certification Practice Statement (CPS)
Owner	Policy Management Authority (PMA), Mollie B.V.
Scope	Internal mTLS client-certificate PKI operated via the Mollie Certificate Manager (MCM)
Objective	Describes the practices and procedures used by the Mollie Internal PKI for issuing, managing, and revoking X.509 mTLS client certificates.
Version	1.2
Effective date	2026-07-31
Last updated	2026-08-03
Planned Review date	Reviewed at least annually
Classification	Public

mollie

Contents

mollie

1. Introduction

1.1 Overview

This Certification Practice Statement (CPS) describes the practices and procedures used by the Mollie Internal Public Key Infrastructure (PKI) for issuing, managing, and revoking X.509 certificates. The PKI is operated through the **Mollie Certificate Manager (MCM)** service, backed by **Google Cloud Certificate Authority Service (CAS)**.

This CPS covers:

- **mTLS client certificates** issued for service-to-service authentication with third-party services that explicitly trust the Mollie subordinate CA chain
- Certificates issued under the following domain namespaces:
  - `*.dev.ca.molops.net`, `*.stag.ca.molops.net`, `*.prod.ca.molops.net`, issued for client-authentication only
- The certificate lifecycle: issuance, approval, renewal, revocation, and expiry

This CPS does **not** cover:

- Public-facing TLS certificates (server certificates exposed on the public internet – managed via separate infrastructure)
- Certificates for `*.mollie.com`, `*.mollie.nl`, or any publicly-registrable Mollie domain (explicitly denied)
- Non-certificate authentication mechanisms
- Internal certificates (Mollie to Mollie authentication, managed by a separate infrastructure)

1.2 Document Name and Identification

Field	Value
Title	Mollie Internal PKI – Certification Practice Statement
Canonical URL	https://pki.mollie.com/cps.pdf
Version	1.2
OID	Not assigned (internal PKI – OID registration planned for future external interoperability)
Format	RFC 3647 nine-section framework

1.3 PKI Participants

1.3.1 Certificate Authorities

Participant	Role
Mollie B.V.	Certificate Authority (CA) – signs certificates, manages CRLs

1.3.2 Registration Authorities

Participant	Role
Mollie B.V.	Registration Authority (RA) – validates CSRs, enforces policy, mediates issuance. No RA function is delegated to third parties

1.3.3 Subscribers

Subscribers are internal Mollie services and teams that request and use mTLS client certificates for service-to-service authentication. Subscribers interact with MCM via:

- **Self-service portal** – Web UI for certificate operations
- **Command-line tooling** – Command-line tool for certificate operations
- **Direct HTTP** – `POST /certificates/requests` for programmatic HTTP API

1.3.4 Relying Parties

Relying parties are internal Mollie services and authorised third parties that authenticate incoming connections using mTLS client certificates issued by this PKI. Relying parties MUST validate the full certificate chain and check the CRL before granting access.

1.4 Certificate Usage

1.4.1 Permitted Uses

- `clientAuth` for mTLS client authentication FROM a Mollie service TO a third-party service that explicitly trusts the Mollie external subordinate CA chain

1.4.2 Prohibited Uses

- Public-facing server TLS (any listener reachable from the public internet)
- Code signing
- Email encryption or signing (S/MIME)
- Any purpose requiring public trust (these certificates chain to a Mollie-private root, not a public root)
- Issuance of `serverAuth` certificates – disallowed at both the application (MCM EKU policy) and infrastructure (GCP CAS `server_auth=false` baseline) layers
- Internal mTLS authentication

1.5 Policy Administration

1.5.1 Organization Administering the Document

Policy Management Authority (PMA), Mollie B.V. – `certificates@mollie.com`

1.5.2 Contact Person

For questions about this CPS, contact the Policy Management Authority (PMA) at `certificates@mollie.com`.

1.5.3 CPS Change Management

Changes to this CPS MUST be:

1. Submitted as a merge request to the CPS repository

2. Reviewed by at least one member of the Security Engineering team
3. Reviewed by at least one member of the Developer Enablement team
4. Approved before merging

1.5.4 CPS Review Cadence

This CPS SHOULD be reviewed and updated:

- At least once per year
- When significant changes are made to the PKI infrastructure or MCM application
- When new risks or mitigations are identified in the risk assessment

1.6 Definitions and Acronyms

Term	Definition
CA	Certificate Authority – Mollie B.V. entity that signs and issues certificates
CPS	Certification Practice Statement – this document
CRL	Certificate Revocation List – list of revoked certificates
CSR	Certificate Signing Request – request submitted by a subscriber for a certificate
EKU	Extended Key Usage – certificate extension specifying permitted uses
HSM	Hardware Security Module – tamper-resistant device for key storage
IAP	Identity-Aware Proxy – Google Cloud authentication gateway
MDP	Mollie Developer Portal - internal service portal for RBAC-authorised Mollie engineers
MCM	Mollie Certificate Manager – the RA application
mTLS	Mutual TLS – TLS with client certificate authentication
OCSP	Online Certificate Status Protocol – real-time certificate status checking
PMA	Policy Management Authority – body responsible for PKI governance
RA	Registration Authority – Mollie B.V. entity that validates certificate requests
RBAC	Role-Based Access Control – authorization model based on group membership
SA	Service Account – non-human identity used for programmatic access
SAN	Subject Alternative Name – certificate extension for additional identities
Agyeius	Internal privilege escalation tool with audit logs and mandatory business justification

2. Publication and Repository

2.1 Repositories

- **CRL (Certificate Revocation List):** Published automatically by GCP CAS to a GCS bucket. CRL distribution points are embedded in all issued certificates.
- **CA Issued Certificates:** Stored in GCP CAS. Not published externally.
- **CPS:** Published at the canonical URL (see §1.2).

2.2 Publication of Certificate Information

CRLs are published automatically by GCP CAS in PEM format to a GCS bucket upon each revocation event. CRL distribution points are embedded in all issued certificates, enabling relying parties to check revocation status.

2.3 Time or Frequency of Publication

Document	Publication Frequency
CPS	Public – readable by any relying party at the canonical URL
CRL	Automatic by GCP CAS on each revocation event
CA Certificates	On CA creation or rotation

2.4 Access Controls on Repositories

Repository	Access Control
CPS	Internal repository access – available to all employees with repo access
CRL	GCS bucket with authenticated access via GCP IAM
CA Certificates	GCP CAS IAM policies restrict access to authorized principals

3. Identification and Authentication

3.1 Naming

3.1.1 Types of Names

All certificates use X.500 Distinguished Names with the following conventions:

Field	Requirement
Common Name (CN)	MUST be a free-form descriptive service identifier (e.g. `Mollie Banking System - PROD`); FQDNs are NOT permitted (to avoid confusion with host/server TLS); no wildcards; not a publicly-registrable name
Organization (O)	MUST be `Mollie B.V.`
Country (C)	MUST be `NL`

Field	Requirement
SAN DNS Names	Permitted but NOT validated against a namespace; deny-list, wildcard-reject and public-suffix-reject still apply (optional – the certificate is identified by its CN)
SAN IP Addresses	NOT permitted
SAN Email Addresses	NOT permitted

3.1.2 Domain Restrictions

Each CA pool enforces its own allowed pattern set. The deployed pools and their patterns are:

Pool	Allowed pattern
`external-nonprod`	Dev, Staging
`external-prod`	Production

Client-certificate names are **not** validated against a domain namespace: neither the Common Name nor the SAN DNS names are matched against an allow-list. A client certificate is identified by its free-form descriptive Common Name, and a third-party relying party may impose its own naming requirements. The following safety checks still apply to both the CN and any SAN DNS names, in order:

1. **Deny list** – any name matching a denied pattern (e.g. `*.mollie.com`, `*.mollie.nl`) is rejected.
2. **Wildcard reject** – any name containing `*` is rejected
3. **Public-suffix reject** – by default any CN or SAN that is itself a publicly-registrable name per the IANA Public Suffix List (e.g. `mollie.com`, `mollie.eu`, bare ccTLDs) is rejected.
4. **Allow list** – the name must match at least one allowed pattern for the selected pool.

Pool selection is automatic: the request's `environment` field combined with its `scope` (`external`) selects the correct pool.

3.2 Initial Identity Validation

3.2.1 Human Users

Human users are authenticated via **Google Cloud Identity-Aware Proxy (IAP)**:

1. IAP issues a signed JWT (`X-Goog-IAP-JWT-Assertion` header) containing the user's email claim.
2. MCM cryptographically validates the JWT signature against Google's published JWKS endpoint.
3. MCM compares the JWT `email` claim against the `X-Goog-Authenticated-User-Email` header for defense-in-depth.
4. The validated email becomes the identity for all certificate operations.

Human users cannot impersonate other users. The `requested_by` body field is ignored for non-service-account callers – identity is always locked to the cryptographically validated JWT.

3.2.2 Service Accounts

Service accounts (SAs) are technical users that are used by systems to access MCM programmatically (programmatic access):

1. SAs authenticate to IAP using their GCP credentials. IAP validates the SA identity cryptographically.
2. SAs MUST be in the `allowed_service_accounts` allowlist. Unauthorized SAs are rejected with `403 Forbidden`.
3. SAs MAY delegate identity via `requested_by` (POST) or `on_behalf_of` (GET) to attribute operations to the triggering human user.

Delegation controls:

Control	Effect
Format validation	Delegated identity MUST be a valid `@mollie.com` email address
Human-only	Delegated identity MUST NOT be a service account
Non-empty	Empty or whitespace-only values are rejected
Admin-user block	SAs MUST NOT delegate to admin users on visibility endpoints
Privileged-user visibility cap	SAs MUST NOT gain full visibility through delegation – delegation to users with `all Visible=true` (RBAC group members, god users) is silently rejected
Audit logging	All delegation events are audit-logged with both the SA and delegated identities

3.2.3 IAP Group Restrictions

IAP access to the RA is restricted to specific authorized teams. Team memberships are resolved via RBAC and managed under change-management controls.

3.2.4 RBAC – Privileged Operations

Sensitive operations require RBAC authorization:

Operation	Required Groups
Certificate revocation	RBAC-authorized operator role
Request approval	RBAC-authorized security role, RBAC-authorized operator role
Request denial	RBAC-authorized security role, RBAC-authorized operator role
List all requests	RBAC-authorized security role, RBAC-authorized operator role

Admin users (configured in the admin user configuration – under change management) bypass RBAC checks for operational purposes. Admin users are staff or senior staff infrastructure engineers (the most senior engineers in the company). Admin user access is audit-logged.

3.3 Identification and Authentication for Re-key Requests

No automatic re-key is supported. Re-key requests follow the same identification and authentication procedures as initial requests (Section 3.2).

3.4 Identification and Authentication for Revocation Requests

Revocation requests require RBAC authorization. The requesting user MUST belong to a group authorized for the `revoke` endpoint (Section 3.2.4). Identity is verified using the same IAP-based authentication as all other operations.

4. Certificate Lifecycle Operational Requirements

4.1 Certificate Application

Mollie engineers request certificates by submitting a PEM-encoded Certificate Signing Request (CSR) to MCM via:

- **Direct HTTP** – `POST /certificates/requests` (approval workflow)
- **Command-line tooling** – Command-line certificate operations
- **Self-service portal** – Self-service automation

The CSR MUST be signed with the corresponding private key (proof of possession). MCM validates the CSR signature before processing.

Any engineer in Mollie can request a new certificate, but the certificate must be validated by an authorized user (see Section 3.2.4)

4.2 Certificate Application Processing

All certificate issuance follows the approval workflow (four-eyes principle):

1. Subscriber submits CSR via `POST /certificates/requests`
2. MCM validates CSR against policy (Section 7.1)
3. Request is stored as `pending` (Redis-backed, 7-day TTL)
4. A different authorized user approves via `POST /requests/:id/approve`
5. **Four-eyes principle enforced:** The approver MUST be part of the approvers group (see Section 3.2.4). The approver MUST be a different user than the subscriber. Self-approval is rejected with `403 Forbidden` and a `policy.violation` audit event.
6. Upon approval, MCM submits the CSR to GCP CAS for signing with the chain specific to the environment and scope
7. Certificate is returned to the subscriber

4.3 Certificate Issuance

MCM submits validated CSRs to GCP CAS for signing. GCP CAS enforces its own baseline controls (Section 6.9) independently of MCM. This provides defense-in-depth: even if MCM is bypassed, GCP CAS enforces key type, EKU, `isCA`, and lifetime constraints.

Issued certificates include:

- The leaf certificate (PEM)
- The full certificate chain (intermediate + root CA certificates)
- CRL distribution point

4.4 Certificate Acceptance

The subscriber receives the certificate, chain, and serial number in the API response. It is the subscriber's responsibility to deploy the certificate and private key to their service.

4.5 Key Pair and Certificate Usage

4.5.1 Subscriber Obligations

Subscribers MUST:

- Protect the private key from unauthorized access and disclosure
- Deploy the certificate only to authorized services
- Revoke the certificate promptly upon private key compromise or suspected compromise
- Monitor certificate expiry and request replacement before expiration
- Use the certificate only for permitted purposes (Section 1.4)

4.5.2 Relying Party Obligations

Relying parties MUST:

- Validate the full certificate chain to the trusted root CA
- Check the CRL distribution point for revocation status before granting access
- Verify the Extended Key Usage includes `clientAuth`
- Reject certificates that fail chain validation, are revoked, or are expired

4.6 Certificate Renewal

Subscribers MAY renew an existing certificate via `POST /certificates/renew` without going through a second four-eyes approval, subject to the following self-service guards:

1. **Ownership match** – the renewal request's `team`, `application`, `domain`, and `environment` MUST match the labels on the existing certificate; mismatches are rejected with `403 Forbidden`.
2. **Eligibility window** – renewal is permitted only after the certificate has reached the configured fraction of its lifetime; early renewal attempts are rejected with `425 Too Early`. This limits GSM secret churn and prevents unlimited free re-issuance.
3. **Caller authorisation** – the renewing user MUST be part of the group owning the certificate.
4. **Key reuse prevention** – the renewal CSR MUST present a new key pair; reuse of an active certificate's public key is rejected with `409 Conflict` (same control as initial issuance, §6.3.2).
5. **CSR re-validation** – the renewal CSR is validated against the full pipeline in §7.1.3.

Renewal events emit `certificate.renewed` audit records and `certificate.renewal.denied` metrics tagged with the denial reason. Certificate expiry metrics are emitted to Datadog for monitoring (Section 8.7) so subscribers receive advance warning of pending expirations.

4.7 Certificate Re-key

Re-key – issuance of a certificate with the same subject but a new key pair – is the standard outcome of renewal (§4.6), because every renewal CSR MUST carry a fresh public key per the key-reuse-prevention control. Re-key without renewal (for example, to rotate keys mid-lifetime) follows the same renewal path.

4.8 Certificate Modification

Certificate modification is not supported. A new certificate MUST be requested to change any certificate attributes.

4.9 Certificate Revocation and Suspension

4.9.1 Circumstances for Revocation

Certificates SHOULD be revoked when:

- The private key is compromised or suspected of compromise
- The service is decommissioned
- The certificate subscriber's affiliation changes
- The certificate is superseded by a new one

4.9.2 Revocation Process

1. An authorized user submits a revocation request via `POST /certificates/revoke` with the certificate ID and reason
2. MCM checks RBAC authorization (user MUST belong to `revoke` endpoint group)
3. MCM calls GCP CAS to revoke the certificate
4. GCP CAS updates the CRL automatically
5. Revocation is audit-logged

4.9.3 Revocation Reasons

Standard X.509 revocation reasons are supported:

- `key_compromise`
- `ca_compromise`
- `affiliation_changed`
- `superseded`
- `cessation_of_operation`
- `certificate_hold`
- `privilege_withdrawn`
- `aa_compromise`

4.9.4 CRL Publication

CRLs are published automatically by GCP CAS in PEM format. CRL distribution points are embedded in all issued certificates. GCP CAS guarantees CRL update within minutes of revocation.

4.9.5 Certificate Suspension

Certificate suspension is not supported. Certificates may only be revoked (permanently).

4.10 Certificate Status Services

OCSP is not currently enabled. Certificate status is available via CRL distribution points embedded in certificates. OCSP implementation is under evaluation for future deployment.

4.11 End of Subscription

When a subscription ends, the certificate expires naturally or is revoked. There is no automatic cleanup mechanism. Subscribers are responsible for revoking certificates that are no longer in use.

4.12 Key Escrow and Recovery

Key escrow is not supported. Subscriber private keys are never transmitted to or stored by MCM. Key recovery is the subscriber's responsibility.

5. Facility, Management, and Operational Controls

5.1 Physical Controls

All CA infrastructure runs on Google Cloud Platform. Physical security is managed by Google per their SOC 2 Type II and ISO 27001 certifications. MCM runs on Google Kubernetes Engine (GKE).

5.2 Procedural Controls

- **Dual control:** The approval workflow enforces the four-eyes principle for certificate issuance.
- **Separation of duties:** Certificate request and approval require different users.
- **Access reviews:** IAP group memberships and RBAC configurations are managed via Terraform and subject to code review.

5.3 Personnel Controls

Access to MCM is restricted to employees in specific Google Groups (Section 3.2.3). Service account access is restricted via the SA allowlist. Admin-user access is limited to named individuals in the RBAC configuration.

5.4 Audit Logging Procedures

All certificate operations are audit-logged with structured fields:

Audit Action	Description
`certificate.issued`	Certificate issued (includes full CSR details)
`certificate.revoked`	Certificate revoked
`certificate.requested`	Certificate request submitted for approval
`certificate.approved`	Certificate request approved and issued

Audit Action	Description
`certificate.denied`	Certificate request denied with reason
`certificate.viewed`	Certificate details retrieved
`certificate.listed`	Certificates listed
`csr.validation_failed`	CSR validation failed (field, reason)
`policy.violation`	Policy violation attempt (e.g., self-approval)
`service_account.impersonation`	SA acting on behalf of a human
`rbac.denied`	RBAC check failed
`rbac.allowed`	RBAC check passed

Audit logs include: user identity, timestamp, operation, certificate ID, team, application, and environment.

Log forwarding: Production audit logs are forwarded to SIEM via Pub/Sub. The audit log forwarding configuration is kept in a file under change management (`mollie_certificate_manager.tf`).

GCP Cloud Audit Logs provide an independent infrastructure-level audit trail for all GCP CAS API calls.

5.5 Records Archival

Audit logs are retained per the SIEM retention policy. GCP Cloud Audit Logs are retained per GCP's default retention (400 days for Admin Activity, 30 days for Data Access).

5.6 Key Changeover

CA key changeover follows the STAGED/ENABLED/DISABLED lifecycle. A new CA is created in STAGED state, tested, then promoted to ENABLED. The old CA is moved to DISABLED after all certificates it issued have expired or been re-issued. See `docs/key-ceremony.md` for detailed procedures.

5.7 Compromise and Disaster Recovery

In the event of a CA key compromise:

1. Revoke all certificates issued by the compromised CA
2. Disable the compromised CA in GCP CAS (`desired_state = "DISABLED"`)
3. Activate a staged replacement CA (`desired_state = "ENABLED"`)
4. Re-issue certificates for affected services

CA rotation is supported via the STAGED/ENABLED/DISABLED lifecycle. Each CA pool MUST have at least one ENABLED or STAGED CA per region (enforced by Terraform validation).

5.8 CA or RA Termination

In the event of PKI termination:

1. Cease all new certificate issuance
2. Notify all subscribers with active certificates
3. Maintain CRL publication for the maximum certificate lifetime (1 year)
4. Archive all audit logs per retention policy
5. Destroy CA private keys via GCP CAS CA deletion (requires deletion protection to be removed by PMA approval)

5.9 Environment Separation

5.9.1 Infrastructure Separation

Separation is enforced by **environment** (prod vs. non-prod). The production MCM deployment manages two distinct CA chains (composed by Root CA and subordinate CA). The staging MCM deployment runs a CA chain used for non-production validation.

Per-environment infrastructure:

Component	Production	Staging / Test
GCP Project	Production GCP project hosting MCM	Staging GCP project hosting MCM
Root CA	Separate root CA hierarchy	Separate root CA hierarchy
MCM deployment	Separate Kubernetes deployment	Separate Kubernetes deployment

Per-pool subordinate CAs and domain namespaces (production):

Pool	Scope	Subordinate CA hierarchy
`external-nonprod`	external	External non-prod subordinate CA
`external-prod`	external	External prod subordinate CA

5.9.2 Application Enforcement

MCM enforces environment matching: the production instance (`current_environment = prod`) only accepts requests with `environment=prod`. The staging instance only accepts `environment=stag`. Cross-environment requests are rejected with `400 Bad Request`.

5.9.3 Trust Isolation

Certificates issued by the production CA hierarchy are not trusted by the staging environment and vice versa. The root CAs are completely independent – there is no cross-signing.

6. Technical Security Controls

6.1 Key Pair Generation and Installation

- **CA keys:** Generated inside GCP Cloud HSM. Keys NEVER leave the HSM boundary.

- **Subscriber keys:** Generated by the subscriber outside MCM. MCM receives only the CSR (public key + signature). Private keys are never transmitted to or stored by MCM.

6.2 Private Key Protection and Cryptographic Module Engineering Controls

CA	Algorithm	Key Size	HSM-backed	Validity
Root CA (both prod and staging)	ECDSA	P-384 (SHA-384)	Yes	5 years
Subordinate CA (both prod and staging)	ECDSA	P-256 (SHA-256)	Yes	3 years

CA private keys are:

- Generated and stored exclusively in GCP Cloud HSM
- Never exportable
- Never accessible by users (both humans and service accounts)
- Only accessible to CAS internally
- Deletion-protected when in ENABLED state (`deletion_protection = true`)

6.3 Other Aspects of Key Pair Management

6.3.1 Subscriber Key Requirements

Key Type	Minimum Size	Status
ECDSA P-256	256-bit	Recommended
ECDSA P-384	384-bit	Allowed
RSA	2048-bit minimum	Discouraged, but allowed for backward compatibility

RSA keys below 2048 bits are rejected by both MCM and GCP CAS.

6.3.2 Key Reuse Prevention

MCM computes a SHA-256 fingerprint of the CSR's public key and checks it against active certificates. If a matching fingerprint is found, the request is rejected with `409 Conflict`. This prevents reuse of potentially compromised private keys.

6.4 Activation Data

CA private keys are activated via GCP IAM authorization. No additional activation data (PINs, passwords) is required beyond GCP authentication.

6.5 Computer Security Controls

MCM runs on GKE with workload identity, network policies, and IAP protection. Access to the MCM application requires IAP authentication (Section 3.2). The application uses GCP Workload Identity for service account binding, eliminating the need for static credentials.

6.6 Lifecycle Technical Controls

6.6.1 Rate Limiting

MCM enforces rate limits to protect against DoS attacks and CA quota exhaustion:

Limit	Value
Global requests/second	100
Global burst	200
Per-user requests/second	10
Per-user burst	20

6.7 Network Security Controls

IAP serves as the network perimeter for MCM. There is no direct internet access to the MCM application. All traffic is authenticated and authorized by IAP before reaching MCM.

6.8 Time-stamping

All timestamps use UTC. Certificate validity periods use GCP CAS system time synchronized via Google's internal NTP infrastructure.

6.9 GCP CA Pool Baseline Controls (Defense-in-Depth)

Each subordinate CA pool enforces baseline controls at the GCP infrastructure level. These controls are applied by GCP CAS **independently of MCM** and cannot be bypassed even with direct API access using the MCM service account credentials:

Controls common to all pools:

Control	Setting	Effect
Allowed key types	ECDSA P-256, P-384; RSA min 2048 bit	Only strong algorithms accepted
Issuance mode	CSR-based only	No config-based issuance
Subject passthrough	Allowed	MCM handles subject validation
`is_ca` baseline	`false` (all subordinate pools)	Cannot issue CA certificates
`client_auth` EKU	`true` (all subordinate pools)	clientAuth always permitted
CRL publishing	Enabled (PEM format)	Revocation information available
Maximum lifetime	1 year (prod) / 90 days (staging)	Enforce creation of a new key pair every expiration

Defense-in-depth: Every policy enforced by MCM at the application layer is also enforced (or further restricted) by GCP CAS at the infrastructure layer. An attacker who

bypasses MCM entirely and calls GCP CAS directly with the MCM SA credentials would still be constrained by these baseline controls – including the critical guarantees that no CA certificate or `serverAuth` certificates can be issued from a subordinate pool.

6.10 Threat Model

Reference Appendix A for mitigations listed.

Threat	Mitigation	Residual Risk
Client cert impersonating an internal server	EKU restriction enforced per-pool at both MCM and GCP CAS; baseline values pin `server_auth=false` at the infrastructure layer	Very low
CA certificate issuance via CSR	`validateLeafOnly` (rejects `is CA=true`) + `pathLenConstraint` reject on leaves + GCP `is_ca=false` infrastructure baseline – three independent layers	Low
Unauthorized certificate issuance	IAP + RBAC + SA allowlist + four-eyes approval	Very low
Cross-environment certificate use	Environment enforcement + separate CA hierarchies per environment	Very low
SA privilege escalation via delegation	Delegation validation + Admin-user block + full-visibility cap	Low
Certificate for public Mollie domains	`server_auth=false` blocks the creation of server certificates	Low
Wildcard certificate covering an unbounded host set	`server_auth=false` blocks the creation of server certificates	Low
CSR carrying unexpected extensions (smuggled trust hints)	X.509 extension allowlist: only standard leaf extensions accepted; any other OID is rejected	Low
Key compromise propagation	Key reuse detection on issuance and renewal (§6.3.2) + revocation capability + short lifetimes (90d non-prod, 1y prod)	Very Low

Threat	Mitigation	Residual Risk
Subscriber private-key exposure	Subscriber key-handling documentation; public-key fingerprint label for reuse detection; short certificate lifetimes (90 days non-prod, 1 year prod); compromise response runbook	Medium

6.11 Known Limitations

1. **Subscriber-generated keys:** Subscribers generate the leaf key pair themselves and submit a CSR; MCM never sees the private key. Responsibility for secure key handling rests with the subscriber. Compensating controls: documented key-handling procedure, public-key-fingerprint reuse prevention (rejects re-issuance from the same key), short certificate lifetimes (90 days non-prod, 1 year prod), revocation runbook.
2. **CN value originates from the subscriber CSR.** MCM validates them against per-pool policy, but does not derive them. The subscriber is responsible for constructing a CSR whose CN accurately describes the intended use. This is further mitigated by `server_auth=false`: clientAuth certificates' CN is not relevant for the clientAuth flow.
3. **SA delegation trust model:** MCM trusts allowed service accounts to honestly report the human user's identity. The delegated identity is validated for format but not cryptographically verified. The SA allowlist is the trust boundary.
4. **No automatic revocation on personnel or service lifecycle events:** There is no automated mechanism to revoke certificates when a user leaves the organization or a service is decommissioned. This requires manual intervention or SIEM-triggered automation.
5. **CRL-based revocation:** Revocation relies on CRL distribution. Relying parties MUST check the CRL to detect revoked certificates. OCSP is not enabled.

7. Certificate, CRL, and OCSP Profiles

7.1 Certificate Profile

7.1.1 Leaf Certificate Profile

All certificates issued by MCM are leaf (end-entity) certificates. The following fields are common to every pool:

Field	Value	Enforced By
Type	Leaf only – `isCA=true` and any explicit `pathLenConstraint` are rejected	MCM (`validateLeafOnly`) + GCP (`is_ca = false` baseline)
Subject Organization	`Mollie B.V.` (required)	MCM (organization policy)

Field	Value	Enforced By
Subject Country	`NL` (required)	MCM (organization policy)
Common Name	MUST be a free-form descriptive service identifier (e.g. `Mollie Banking System - PROD`); FQDNs are NOT permitted (to avoid confusion with host/server TLS); no wildcards; not a publicly-registrable name	MCM (deny-list + wildcard reject + public-suffix reject)
SAN DNS Names	Not permitted	MCM
SAN IP Addresses	Not permitted	MCM (`allow_ip_addresses = false`)
SAN Email Addresses	Not permitted	MCM (`allow_email_addresses = false`)
Max SAN Count	10	MCM (`max_san_count = 10`)
Key Usage	`digitalSignature`, `keyEncipherment`	GCP (baseline)
Basic Constraints	`isCA = false` (no `pathLenConstraint` permitted on leaves)	MCM + GCP (defense-in-depth)
X.509 extensions in CSR	Only the standard leaf set is accepted: `BasicConstraints`, `KeyUsage`, `ExtKeyUsage`, `SubjectAlternativeName`, `AuthorityKeyIdentifier`, `SubjectKeyIdentifier`. Any other OID is rejected. Per-pool allowlist extensions MAY be configured for explicit additions.	MCM (extension allowlist)
Permitted EKUs	`clientAuth` only	MCM and GCP

7.1.2 Certificate Lifetime

Environment	Maximum Lifetime	GCP Baseline Max
Production	1 year (8760h)	1 year
Staging	90 days (2160h)	90 days

7.1.3 CSR Validation Pipeline

MCM validates every CSR through the following pipeline before submission to GCP CAS. Each step is a guard clause: the first failure aborts the request with a structured error and an audit log record naming the failed field.

1. **PEM parsing** – Valid PEM block with `CERTIFICATE REQUEST` type.
2. **Signature verification** – CSR signature proves private-key possession.
3. **Key type / size** – Key type in pool's allowed list, EC curve in pool's allowed list, RSA minimum 2048 bits.
4. **Subject** – Common Name non-empty and ≤64 characters.
5. **Organization** – `O` matches pool's required organization (`Mollie B.V.`), `C` matches required country (`NL`), `OU` in allowlist if configured.
6. **Naming policy** – Deny-list checked against the CN and any SAN DNS names; no namespace allow-list is applied to client certificates (domain-rejection is applied in steps 9–10). IP/email SAN gates enforced. SAN count within pool bounds.
7. **EKU policy** – Only `clientAuth` EKU is permitted. Any other EKU is rejected.
8. **Leaf-only** – `basicConstraints.isCA=true` rejected; any explicit `pathLenConstraint` on a leaf rejected.
9. **Wildcard reject** – CN and SAN containing `*` or any domain rejected by default.
10. **Extension allowlist** – Any X.509 extension OID outside the standard leaf set (`BasicConstraints`, `KeyUsage`, `ExtKeyUsage`, `SubjectAlternativeName`, `Authority KeyIdentifier`, `SubjectKeyIdentifier`) is rejected.
11. **Key reuse** – Public-key SHA-256 fingerprint checked against active certificates issued by MCM; reuse rejected with `409 Conflict`.
12. **Environment** – Request `environment` field MUST match the MCM instance's `current_environment`.

7.2 CRL Profile

CRLs are published by GCP CAS in PEM format to a GCS bucket. CRL distribution points are embedded in all issued certificates. CRLs are updated automatically by GCP CAS upon each revocation event. GCP CAS guarantees CRL update within minutes of revocation. CRL format follows the X.509 v2 CRL profile.

7.3 OCSP Profile

OCSP is not currently deployed. Certificate status is provided exclusively via CRL. OCSP deployment is under evaluation.

8. Compliance Audit

8.1 Frequency or Circumstances of Assessment

The PKI is assessed annually as part of the organization's information security audit program. Ad-hoc assessments are triggered by:

- Significant PKI infrastructure changes
- Security incidents involving certificates
- Changes to regulatory requirements

8.2 Identity/Qualifications of Assessor

Assessments are conducted by the Platform department or qualified external auditors with PKI/X.509 expertise. External auditors MUST be independent of the Security Engineering team.

8.3 Assessor's Relationship to Assessed Entity

Internal assessments are conducted by the Platform department, which is organizationally separate from PKI operations team. External auditors are contractually independent.

8.4 Topics Covered by Assessment

Assessments cover the following topics:

- Identity verification controls
- Certificate lifecycle procedures
- Key management practices
- Certificate request review adequacy
- Audit logging completeness
- Audit event response
- RBAC enforcement
- Environment separation
- Incident response readiness
- CPS compliance

8.5 Actions Taken as Result of Deficiency

Findings are classified by severity (Critical, High, Medium, Low). Critical and High findings require remediation within 30 days. Medium findings require remediation within 90 days. Remediation is tracked by the PMA and verified by the reporter of the finding. See `docs/control-testing.md` for evidence requirements.

8.6 Communication of Results

Assessment results are communicated to the PMA and relevant stakeholders. Critical findings are escalated to the CISO within 24 hours.

8.7 Continuous Monitoring

8.7.1 Monitoring Infrastructure

MCM is monitored through two complementary pipelines:

- **Datadog (operational monitoring):** 30+ monitors deployed via Terraform, with a comprehensive dashboard covering certificate lifecycle, security events, RBAC, performance, and collector health. Alerts route to Slack and PagerDuty.
- **Chronicle SIEM (security monitoring):** Audit logs are forwarded via Pub/Sub log sink to Chronicle SIEM for security correlation and forensic analysis.

8.7.2 Key Metrics

MCM emits the following security-relevant metrics to Datadog:

Metric	Purpose
`certificate.expiry.days`	Expiry monitoring per certificate
`ca.expiry.days`	CA certificate expiry monitoring
`certificate.key.reuse`	Key reuse detection
`csr.validation.failed`	CSR policy violations by reason
`rbac.denied` / `rbac.allowed`	Authorization decisions
`rbac.god_user_access`	Admin-user access tracking
`certificate.inventory`	Certificate inventory by status/team
`request.submitted` / `approved` / `denied` / `self_approval`	Approval workflow tracking
`ratelimit.exceeded`	Rate limiting events

8.7.3 Active Alerts

Alert	Threshold	Severity	Status
Certificate expiry warning	< 30 days	Warning	Active
Certificate expiry critical	< 14 / 7 days	Critical	Active
CA certificate expiry	< 90 / 30 days	Warning / Critical	Active
RBAC denial spike	> 10 in 15m	Warning	Active
Key reuse detected	> 0	Critical	Active
Self-approval attempt	> 0	Critical	Active
Admin-user access	> 0	Warning	Active
High issuance error rate	> 10%	Warning	Active
Revocation failure	> 0	Critical	Active
GCP CAS errors	> 5 in 15m	Critical	Active
Collector not running	No data in 15m	Critical	Active

9. Other Business and Legal Matters

9.1 Fees

No fees are charged for certificate issuance, revocation, or other PKI services. The PKI is operated as an internal infrastructure service.

9.2 Financial Responsibility

Not applicable. Internal PKI only.

9.3 Confidentiality of Business Information

Mollie classifies data on a four-level scale according to the impact of unauthorised disclosure: C1 – Public (no impact), C2 – Medium (internal use; low impact), C3 – High (confidential; medium impact), and C4 – Critical (restricted; high impact). Under this scale:

- Certificate information (subject, serial number, validity) – C2 (Medium).
- Audit logs, which record the requesting and approving user's email address – C2 (Medium).
- Private keys and other cryptographic key material – C4 (Critical). Subscriber private keys are generated by the subscriber and are never transmitted to or stored by the RA.

9.4 Privacy of Personal Information

Certificates do not contain personal data in the certificate fields.

However, MCM **audit logs** record the identity of the requesting and approving user (email address from IAP). Processing of this personal data in audit logs is governed by Mollie's internal privacy policies and GDPR obligations. Data subjects may request information about certificate operations attributed to their identity.

9.5 Intellectual Property Rights

Mollie B.V. retains all intellectual property rights in the PKI infrastructure, including the MCM application, CPS, and CP documents.

9.6 Representations and Warranties

9.6.1 CA Warranties

The CA warrants that it SHALL:

- Issue certificates in accordance with this CPS
- Maintain CRL publication for revocation status
- Revoke certificates upon valid request from authorized parties

9.6.2 RA Warranties

The RA (MCM) warrants that it SHALL:

- Verify the identity of certificate requesters per this CPS
- Enforce certificate policy on all CSRs
- Maintain audit logs of all certificate operations

9.6.3 Subscriber Warranties

Subscribers warrant that they SHALL:

- Provide accurate information in CSRs
- Protect private keys from unauthorized access
- Notify the PKI operations team promptly upon key compromise or suspected compromise
- Use certificates only for permitted purposes (Section 1.4)

9.6.4 Relying Party Warranties

Relying parties warrant that they SHALL:

- Validate the full certificate chain before relying on a certificate
- Check the CRL for revocation status
- Use certificates only for permitted purposes

9.7 Disclaimers of Warranties

Mollie warrants conformance with this CPS for the lifetime of the trust relationship, subject to the change-management procedure in §1.5.3 and the notice mechanism in §9.11. The third-party relying party MUST validate the full certificate chain to the trusted root, check the CRL distribution point on each authentication attempt, and constrain acceptance to the agreed `clientAuth` use case.

The PKI is not designed for, and MUST NOT be used for, public-facing server TLS, code signing, document signing, or any purpose requiring public-trust anchor inclusion (e.g. browser root stores).

9.8 Limitations of Liability

Mollie is not liable for damages resulting from: (a) subscriber failure to protect private keys, (b) relying party failure to check CRL, (c) use of certificates for prohibited purposes.

9.9 Indemnities

Not applicable for internal PKI.

9.10 Term and Termination

This CPS is effective from the date of publication and remains in effect until superseded. The PMA may terminate the PKI with 30 days notice to all subscribers.

9.11 Individual Notices and Communications

Notices regarding PKI changes are communicated via: (a) Slack (#certificate-management channel), (b) email to affected team leads and third parties, (c) MDP announcements for user-facing changes.

9.12 Amendments

Amendments to this CPS follow the change management process (Section 1.5). Substantive changes require PMA approval. Editorial changes may be made by the PKI operations team. All amendments are versioned and tracked in the repository.

9.13 Dispute Resolution

Disputes regarding certificate issuance, revocation, or policy interpretation are resolved by the PMA. Unresolved disputes are escalated to the CISO.

9.14 Governing Law

This CPS is governed by the laws of the Netherlands.

9.15 Compliance with Applicable Law

The PKI operates in compliance with applicable Dutch and EU regulations, including GDPR for personal data in audit logs (see Section 9.4).

9.16 Miscellaneous Provisions

If any provision of this CPS is found unenforceable, the remaining provisions remain in effect.

9.17 Other Provisions

9.17.1 PCI DSS Alignment

The following controls align with PCI DSS requirements:

- **Requirement 4.1:** Strong cryptography (ECDSA P-256/P-384, RSA 2048+ minimum)
- **Requirement 8.2:** Unique identification (IAP-authenticated identity per user)
- **Requirement 10.1:** Audit trails (all certificate operations logged)
- **Requirement 10.5:** Audit log integrity (SIEM forwarding, GCP Cloud Audit Logs)

Appendix A: Mitigation Reference

ID	Mitigation	Status
M-025	EKU restrictions at GCP CA pool level	In Place
M-026	Certificate policy validation in MCM	In Place
M-027	Identity delegation validation and audit logging	In Place
M-028	Domain pattern enforcement (allow/deny lists)	In Place
M-029	Environment separation enforcement	In Place
M-030	Privileged user delegation visibility cap	In Place
M-036	Separate root CAs for production and non-production purposes	In Place
M-037	Separate root CAs for internal and external private PKI	In Place
M-038	External private PKI restricted to client certificates only	In Place
M-040	Environment separation enforcement in MDP / MCM	In Place
M-041	Four-eyes principle: SecEng review before certificate issuance	In Place

ID	Mitigation	Status
M-043	Authentication enforced on the secret vault (GCP IAM on GSM/KMS)	In Place
M-044	Certificate inventory queries on key metrics (algorithm, size, expiry)	In Place
M-045	Reject CSRs containing wildcard CN/SAN (per-pool opt-in to relax)	In Place
M-046	Reject CSRs whose CN/SAN is a publicly-registrable name (IANA Public Suffix List)	In Place
M-047	Reject leaf CSRs carrying an explicit `pathLenConstraint` (defense-in-depth alongside `isCA=true` rejection)	In Place
M-048	Strict X.509 extension allowlist on CSRs: deny any OID outside the standard leaf set; per-pool extensions OPT-IN	In Place