Modular Integrated Sustainable Datacenter

Research into the Possible Variants of an Edge Cybersecurity Management Model based on the Highest Functionality and the Most Minimal Hardware Design

Deliverable D3.5

Deliverable: D3.5
Work-package: WP3 (Cybersecurity Edge)
Authors: Simon Kuhn (NBIP), Ramin Yazdani (NBIP)
Contributors:
Due Date: 31/03/2026
Submission Date: 31/03/2026
Pages: 25
Version: 1.0
Type: Public/Internal Report

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Table of Contents

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

1. Executive Summary

This deliverable outlines a cybersecurity management model for a federated edge data center infrastructure, to be developed in the scope of the Modular Integrated Sustainable Datacenters (MISD) project. We suggest addressing edge cybersecurity management by realizing a Zero Trust Architecture (ZTA): the principle that nothing is trusted by default, and every request must be verified regardless of where it comes from. Rather than treating this as an abstract framework, we focus on how ZTA can be put into practice through API gateways. Since nearly all interactions in a distributed infrastructure happen over APIs, the gateway is a natural place to enforce identity verification, access control, and monitoring consistently across all sites.

The report covers three areas. First, it walks through the core Zero Trust principles, including explicit identity verification, least-privilege access, micro-segmentation, and continuous monitoring, and explains why each of them matters specifically in an edge-cloud context. Second, it outlines how an API gateway-centric architecture can operationalize these principles: offloading authentication from individual services, enforcing fine-grained access policies, creating logical boundaries between workloads, and generating the telemetry needed to detect and respond to threats in real time. Third, it presents a reference implementation that can be built around widely adopted open-source tools, and reviews published performance benchmarks for several API gateway implementations.

The key takeaway of this report is that security across a heterogeneous edge-cloud ecosystem is achievable without requiring every service to implement its own authentication and policy logic. A well-integrated API gateway layer can handle that responsibility centrally, leaving services to focus on their actual function.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

2. Introduction

A federated/democratized edge-cloud continuum involves a wide range of stakeholders [1] (infrastructure providers, service providers, network/telecom providers, hardware vendors, regulatory bodies and law enforcement, and end users) who can deploy, manage, and consume services across shared and federated environments.

While this openness drives innovation and flexibility, it also introduces substantial security challenges. The ecosystem is inherently heterogeneous, spanning multiple administrative domains, trust boundaries, and operational models. Workloads move dynamically between edge and cloud layers, and devices may be intermittently connected, physically exposed, or managed by different entities. In such conditions, traditional perimeter-based security assumptions break down. There is no single, stable boundary that can be defended as “inside” versus “outside”.

This deliverable addresses these challenges by presenting a cybersecurity management model suitable for a democratized edge–cloud continuum. The approach is grounded in Zero Trust Architecture (ZTA) [4], which rejects implicit trust based on network location and instead requires continuous verification of every entity, request, and transaction. Rather than treating security as an external layer, the proposed model embeds it directly into the communication fabric of the ecosystem.

To operationalize Zero Trust principles, the architecture is centered around API gateways. In a distributed environment where interactions are predominantly API-driven, the gateway becomes a natural control point. It can enforce authentication and authorization policies, validate identities, apply rate limiting and threat detection, and provide consistent observability across domains. By positioning the API gateway as a policy enforcement layer, the model enables uniform security controls across edge and cloud components without imposing rigid centralization.

The key objectives of this report are:

O1: To elaborate on the key principles of a zero trust architecture and their relevance in a democratized edge-cloud ecosystem

O2: To Outline an API-gateway-centric implementation model, detailing the architectural components, control mechanisms, and governance considerations required to make the framework practical and scalable

O3: To Provide insights into the performance of various API gateway implementations

This deliverable builds on the outcomes of earlier work within WP3. Deliverable D3.1 identified the joint cybersecurity needs and acceptance criteria across the MISD stakeholder landscape. Deliverables D3.2 and D3.3 defined the technical and functional specifications for the cybersecurity module, including its API design and modular hardware requirements. The management model proposed in this report is informed by those specifications and is intended to guide the architecture of the Infrastructure Protection System (IPS) currently under development in D3.4.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

The remainder of this report is structured as follows. Section 3 presents the key principles of a zero trust architecture. Section 4 elaborates on realizing ZTA using an API gateway -centric approach. In Section 5, we propose a reference implementation architecture for an API security gateway. Section 6 presents performance benchmarks of various API gateway solutions. Finally, Section 7 concludes the report and provides recommendations and next steps to be considered.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

3. Zero-Trust Principles Applied to the Edge-Cloud Continuum

Zero Trust Architecture (ZTA) is often phrased as “never trust, always verify”. However, in practice it represents a broader shift in how trust is established and maintained in complex systems. In a democratized edge–cloud continuum, where devices, services, and users operate across organizational and technical boundaries, implicit trust assumptions can introduce significant security vulnerabilities. The following principles outline the core elements of Zero Trust and explain why they are particularly relevant in such heterogeneous environments [2].

3.1 Explicit Identity and Device Verification

In a distributed edge data center model, identities are not limited to human administrators. The majority of interactions occur between workloads, orchestration systems, management services, and infrastructure components. Establishing reliable and verifiable identities for these entities is therefore essential.

Edge data centers may be deployed in remote or lightly staffed facilities. They may be operated by different business units or partner organizations. Network-level trust assumptions (e.g., considering all traffic within a site as trusted) create unnecessary exposure, especially when sites are interconnected over wide-area networks.

Zero Trust requires that every workload and service instance possess a unique, cryptographically verifiable identity. Authentication must occur for all control-plane and data-plane communications, including orchestration traffic, configuration updates, and service-to-service interactions. Identity lifecycle management is critical, particularly in environments where workloads and resources are dynamically deployed or migrated between cloud and edge sites.

3.1.1 Workload and Service Identity

Each workload instance, service, and infrastructure component should possess a unique cryptographic identity. Mutual authentication and certificate lifecycle management are two important aspects of this.

Mutual TLS is a practical way to ensure that services do not simply encrypt traffic, but also prove who they are to each other. With mTLS, both sides of a connection present verifiable X.509 certificates, and each validates the other before any meaningful data is exchanged.

In an edge–cloud data center continuum, this is especially important for control-plane communication between central cloud orchestrators and edge nodes, where configuration changes and workload deployments are initiated. It is equally relevant for replication and synchronization processes that move data between sites, as well as for internal infrastructure APIs that manage storage, networking, or monitoring functions.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Strong service identity depends not only on issuing certificates, but on managing them properly throughout their lifecycle. An internal public key infrastructure (PKI) or an automated certificate authority can handle certificate issuance, renewal, rotation, and revocation in a controlled and auditable manner.

Automation is particularly important in edge–cloud environments, where workloads are frequently instantiated, scaled, or relocated. Manual certificate handling does not scale and increases the likelihood of configuration errors.

Using short-lived certificates further limits risk. If a credential is exposed, its validity window is narrow, reducing the opportunity for misuse. This approach aligns with the Zero Trust objective of minimizing persistent trust relationships and ensuring that identities must be regularly revalidated rather than being assumed trustworthy indefinitely.

3.1.2 Token-Based Identity and Delegation

Where API-driven interactions are involved (particularly in management and orchestration contexts) token-based mechanisms complement certificate-based identity.

OAuth 2.0 enables delegated authorization. For example, a centralized automation system may request scoped access to an edge site's management API without sharing long-lived credentials. JSON Web Tokens (JWTs) allow identity claims and authorization context to be embedded in signed, verifiable tokens. Because JWTs are self-contained and cryptographically signed, they can be validated locally at edge sites without continuous reliance on centralized session stores.

Short-lived JWTs reduce exposure and ensure that access must be periodically re-validated. Claims within the token can include workload identity, role, site identifier, or environmental classification, supporting fine-grained authorization decisions.

3.2 Least-Privilege and Policy-Driven Access

Edge data centers are expected to host mixed-sensitivity workloads. They may process regulated or proprietary data, while providing generalized compute or caching functions. In this context, excessive permissions pose significant risk. An over-privileged orchestration component, if compromised, could propagate malicious configurations across multiple sites and workloads. Similarly, workloads granted broad access to storage or inter-site communication channels may unintentionally expose sensitive data.

Applying least privilege means constraining access at multiple levels:

- Administrative access to edge infrastructure must be tightly scoped and role-specific.
- Workloads should access only the data stores and services explicitly required for their function.
- Inter-site communication should be limited to defined service relationships.

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Policy-driven access control allows these constraints to adapt to operational context. Access decisions can account for workload classification, site location, operational state, and risk indicators.

3.2.1 Role-Based and Attribute-Based Models

In an edge–cloud data center continuum, access control must account for both operational clarity and environmental variability. Two complementary models are particularly relevant: Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC).

Role-Based Access Control (RBAC) structures permissions around clearly defined roles rather than individual identities. Instead of assigning privileges user by user or service by service, permissions are grouped and attached to roles such as edge site operator, cloud infrastructure administrator, workload deployment service, or monitoring agent.

However, in distributed environments where workloads move between sites and operational conditions change, RBAC alone can become too rigid. Attribute-Based Access Control (ABAC) addresses this limitation by incorporating contextual information into access decisions. Instead of relying solely on a predefined role, ABAC evaluates attributes associated with the request and the environment. These attributes may include workload classification (for example, regulated, internal, or public), site location, deployment environment (production versus staging), time-based restrictions, or a calculated risk score reflecting current security posture.

This additional context allows for more granular and adaptive decisions. For instance, access to sensitive data can be limited to specific edge sites that meet defined compliance requirements. Replication services can be constrained based on data classification levels. Administrative actions can be restricted outside designated maintenance windows.

In practice, RBAC and ABAC are often combined. Roles provide structural clarity, while attributes introduce the flexibility required for a dynamic edge–cloud ecosystem. Together, they support least-privilege enforcement without sacrificing operational practicality.

3.2.2 Policy Enforcement

Policy engines can evaluate access requests based on identity claims (e.g., from JWTs), certificate metadata, and environmental attributes. Access to APIs, storage systems, configuration repositories, and orchestration endpoints should be granted only when policy conditions are satisfied. Effective enforcement typically follows several principles:

- Separation of control-plane and data-plane permissions.
- Just-in-time administrative access.
- Scoped OAuth tokens reflecting minimal required privileges.
- Immediate revocation capabilities when risk conditions change.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

3.3 Micro-Segmentation

As we discussed earlier, traditional data center security often relied on strong perimeter defenses combined with coarse internal segmentation. In an edge–cloud continuum, such a model is insufficient. Each edge data center is effectively a semi-autonomous extension of the broader infrastructure, and compromise at one site must not enable unrestricted movement across others.

Micro-segmentation enforces granular isolation both within individual edge sites and between sites. Within a single edge data center, workloads are segmented according to application domain, sensitivity level, or tenant. East–west traffic is permitted only when explicitly defined by policy.

Across the continuum, site-to-site communication is similarly constrained. Replication services, synchronization processes, and management channels operate over clearly defined and authenticated paths. This prevents a breach in one edge location from automatically exposing centralized cloud systems or peer edge nodes.

3.3.1 Segmentation within an edge data center

Within a single edge data center, segmentation should prevent implicit trust between internal components. Even though workloads share the same physical site, they should not automatically share unrestricted connectivity. Management plane traffic must be clearly separated from application workloads to reduce the risk that a compromised application can influence infrastructure control functions. Tenant workloads should be logically isolated, and where sensitivity demands it, cryptographic separation should be enforced. Service-to-service communication should require explicit authentication and authorization, commonly implemented through mechanisms such as mTLS. In addition, east–west traffic must be limited to predefined and policy-approved service relationships rather than open internal networks.

These controls can be implemented using software-defined networking (SDN), container- or VM-level network policies, identity-aware routing mechanisms, and service mesh architectures that enforce workload-level identity verification. The objective is to replace broad internal connectivity with narrowly defined verifiable communication paths.

3.3.2 Segmentation between edge data centers (and central cloud regions)

Segmentation is equally important between sites. Each edge data center should be treated as a distinct security domain, even when operated by the same organization. Replication and synchronization channels must be explicitly defined and authenticated. Broad, full-mesh trust relationships between sites should be avoided, as they increase the risk of lateral movement across the continuum. Instead, communication should be restricted to clearly identified services and endpoints. Identity claims embedded in JWTs or attributes contained in certificates can bind

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

communication to specific site identities. Cross-site interactions should require both verified identity and successful policy evaluation before access is granted.

By enforcing strict boundaries both within and between sites, the architecture limits the blast radius of compromise and maintains controlled trust relationships across the edge–cloud environment.

3.4 Continuous Assessment and Monitoring

Edge data centers operate under dynamic conditions. Workloads and resources scale in response to local demand, connectivity to central cloud regions may fluctuate, etc. These factors introduce ongoing security variability. Zero Trust therefore requires continuous assessment rather than static validation. Telemetry from workloads, infrastructure components, and inter-site connections must be collected and analyzed to detect anomalies.

Continuous monitoring also supports operational resilience. If an edge site exhibits signs of compromise, access can be restricted, workloads isolated, or trust relationships temporarily suspended without disrupting the entire ecosystem. Risk-based evaluation enables proportional responses instead of broad shutdown measures.

3.4.1 Telemetry and Logging

Comprehensive logging should be implemented to cover several categories of events. Authentication activities are a fundamental category, covering certificate validation outcomes, token issuance, and rejected authentication attempts. API access logs must be tied to verified identity claims so that actions can be traced back to specific workloads or administrative roles. Administrative operations on orchestration and management systems also require detailed logging, as these interfaces often represent high-impact control points. Finally, the lifecycle of cryptographic material (certificate issuance, renewal, and revocation) must be recorded to ensure accountability and traceability.

3.4.2 Adaptive and Risk-Based Controls

Continuous assessment is only meaningful if it enables proportionate responses. In a Zero Trust model, access decisions are not static; they can change as risk conditions evolve. When suspicious activity is detected, the system should be able to react in near real time. This may involve revoking OAuth tokens or JWTs that are associated with anomalous behavior, thereby preventing further use. If a certificate is suspected to be compromised, it should be invalidated promptly to prevent impersonation. Similarly, workloads that exhibit abnormal patterns can be isolated by updating network policies or segmentation rules, limiting their ability to communicate with other components until the situation is resolved. These adaptive measures ensure that trust remains conditional. Rather than allowing potentially compromised identities or services to operate unchecked until a manual review occurs, the system can reduce exposure immediately while further investigation takes place.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

4 API Gateway -Centric ZTA Realization

Several approaches to cybersecurity management in distributed edge environments were considered. Host-based intrusion detection and prevention offers direct workload protection but requires per-service deployment and does not scale easily across heterogeneous multi-site environments. Service mesh architectures provide strong service-to-service security but introduce additional resource overhead at each node, which conflicts with the minimal hardware objective. SDN-based segmentation can enforce network-level isolation but lacks visibility into application-layer identity and authorization. The API gateway-centric model was selected because it provides a balance between centralized policy enforcement, minimal per-service overhead, and practical deployability across sites with varying hardware capabilities.

The Zero Trust principles discussed in the previous section define ***what*** must be achieved (explicit identity verification, least-privilege access, micro-segmentation, and continuous assessment). An API security gateway -centric deployment addresses ***how*** these principles can be enforced consistently across an edge-cloud data center continuum.

In modern distributed data center environments, nearly all control-plane and a growing portion of data-plane interactions are API-driven. Orchestration systems, storage systems, monitoring systems, configuration services, and replication mechanisms all rely on API endpoints. Positioning a security gateway at these interaction boundaries creates a logical enforcement layer through which trust decisions can be centralized in policy while remaining distributed in execution.

Deploying API security gateways within each edge data center creates a consistent enforcement model across the continuum through which:

- Policies can be defined centrally and propagated to distributed gateways.
- Identity validation logic remains uniform across sites.
- Cross-site interactions are governed by explicit, auditable rules.
- Backend services can remain simpler, focusing on business logic rather than security enforcement.

This approach does not eliminate the need for complementary controls such as network segmentation, host hardening, or infrastructure security. Rather, it provides a cohesive application-layer enforcement plane that aligns directly with Zero Trust principles. The API gateway becomes a policy enforcement point (PEP) that evaluates every request based on identity, context, and defined security rules before it reaches back-end services.

4.1 Enforcing Explicit Identity and Strong Authentication

A gateway-centric architecture ensures that no API request reaches internal services without undergoing identity verification. This directly operationalizes the principle of explicit identity and workload verification. Key enforcement mechanisms include:

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

- **mTLS termination and validation**

The gateway can require mutual TLS for inbound and inter-site API traffic. Client certificates are validated against trusted certificate authorities, ensuring that only authenticated workloads or management systems can initiate communication.

- **JWT validation and claim inspection**

When OAuth 2.0 is used for delegated access, the gateway validates JWT signatures, expiration times, and embedded claims. It can enforce constraints such as:

  - Required issuer and audience.
  - Allowed roles or scopes.
  - Site or workload identifiers embedded in claims.

- **Token verification and revocation checks**

For higher-sensitivity operations, the gateway can verify token validity against an authorization server to ensure that revoked or expired tokens are not accepted.

- **Certificate attribute enforcement**

Certificate fields (e.g., subject, SAN entries, organizational attributes) can be mapped to workload identity and evaluated against policy rules.

Offloading identity validation to the gateway relieves backend services from implementing complex authentication logic independently, contributing to a ***cybersecurity management model with minimal hardware design***. Additionally, this reduces inconsistency and eliminates gaps caused by uneven security implementation across edge data centers.

4.2 Enabling Least-Privilege and Policy-Driven Access Control

The API security gateway functions as a centralized enforcement layer for fine-grained authorization decisions, directly supporting least-privilege principles. For each API request, the gateway can evaluate identity claims (from a JWT or certificate metadata), role assignments (RBAC), and contextual attributes (ABAC). For example, a replication service may be authorized to read from a specific data store but not modify configurations, or an edge site management service may access only its own site's APIs, not those of peer locations. Because policy is externalized from application code, updates to access rules do not require redeploying workloads. This separation strengthens governance and reduces operational risk.

Least privilege extends beyond functional permissions - OAuth scopes, for instance, restrict token capabilities. The gateway can enforce rate limits per identity or workload, and sensitive APIs can require stronger authentication levels. This prevents excessive use of privileged endpoints and mitigates abuse scenarios such as credential misuse or automated exploitation attempts.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

4.3 Supporting Micro-Segmentation Through API-Level Isolation

While network-level segmentation remains important, an API security gateway provides logical micro-segmentation at the application layer.

Segmentation can be implemented inside an edge data center as follows, transforming service interaction into a set of defined contracts rather than unrestricted internal communication:

- Only traffic routed through the gateway is permitted to reach internal APIs.
- Direct east–west communication between workloads can be restricted or forced through controlled entry points.
- Access policies explicitly define which service identities may call which APIs.

Segmentation is also crucial across the continuum (inter-site). Rather than creating broad trust between sites, the gateway ensures that only explicitly authorized service relationships are allowed. This significantly reduces the risk of lateral movement across data center boundaries.

- Each edge site can expose only a minimal set of APIs externally.
- Inter-site calls must pass through authenticated gateway endpoints.
- Site identity (embedded in certificates or JWT claims) can be used to enforce strict cross-site boundaries.

4.4 Enabling Continuous Assessment and Adaptive Control

An API gateway sits directly in the path of control-plane and service-to-service communication, which gives it the practical advantage of seeing nearly every relevant request as it happens. This vantage point makes it a natural place to establish continuous observability and to support compliance requirements without introducing separate monitoring layers.

4.4.1 Centralized Observability

Because all API calls pass through the gateway, it can produce a consolidated and structured record of activity across both edge and cloud sites. This typically includes:

- Successful and failed authentication attempts.
- Results of token validation, including expired or malformed tokens.
- Which API methods are invoked, and by which identities.
- Cross-site communication attempts between edge locations and central systems.
- Irregular traffic patterns, such as unexpected request volumes or deviations from established usage baselines.

When aggregated across the continuum, this telemetry provides a coherent view of system behavior. It becomes possible to identify patterns such as privileged roles being used outside their normal scope, as well as repeated attempts to access APIs that are not authorized for a given identity. This enables operators to gain a centralized and consistent perspective on API-level interactions, instead of piecing together logs from multiple subsystems.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

4.4.2 Dynamic Enforcement

Beyond passive monitoring, the gateway also acts as an active enforcement point. Because policy evaluation occurs at request time, responses to suspicious behavior can be immediate rather than retrospective. For example, the gateway can:

- Deny requests originating from identities that have been flagged as compromised.
- Reject or invalidate tokens based on updated authorization information.
- Tighten rate limits or temporarily block traffic from sources exhibiting anomalous behavior.
- Require renewed authentication for operations classified as high risk.

This capability aligns with the Zero Trust principle that trust is not permanent. Access is not granted indefinitely based on an earlier decision; it is reassessed with each interaction. Rather than depending solely on periodic audits or after-the-fact log analysis, enforcement is embedded directly in the execution path of every API call.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

5 Reference Implementation Architecture

This chapter aims at translating the previously discussed Zero Trust principles into a concrete reference architecture for an edge–cloud data center continuum. The goal is not to prescribe a single rigid implementation. Instead, it defines a consistent set of building blocks and integration patterns that can be adopted across sites and partners while maintaining a uniform security posture.

5.1 Core Components

At each edge site, a small set of logical components forms the foundation of the security architecture. These components correspond directly to the identity, policy, and observability concepts introduced earlier. We depict our proposed referenced architecture for edge cybersecurity management in Figure 1. This architecture is inspired by existing architectures around API gateways [2, 3].

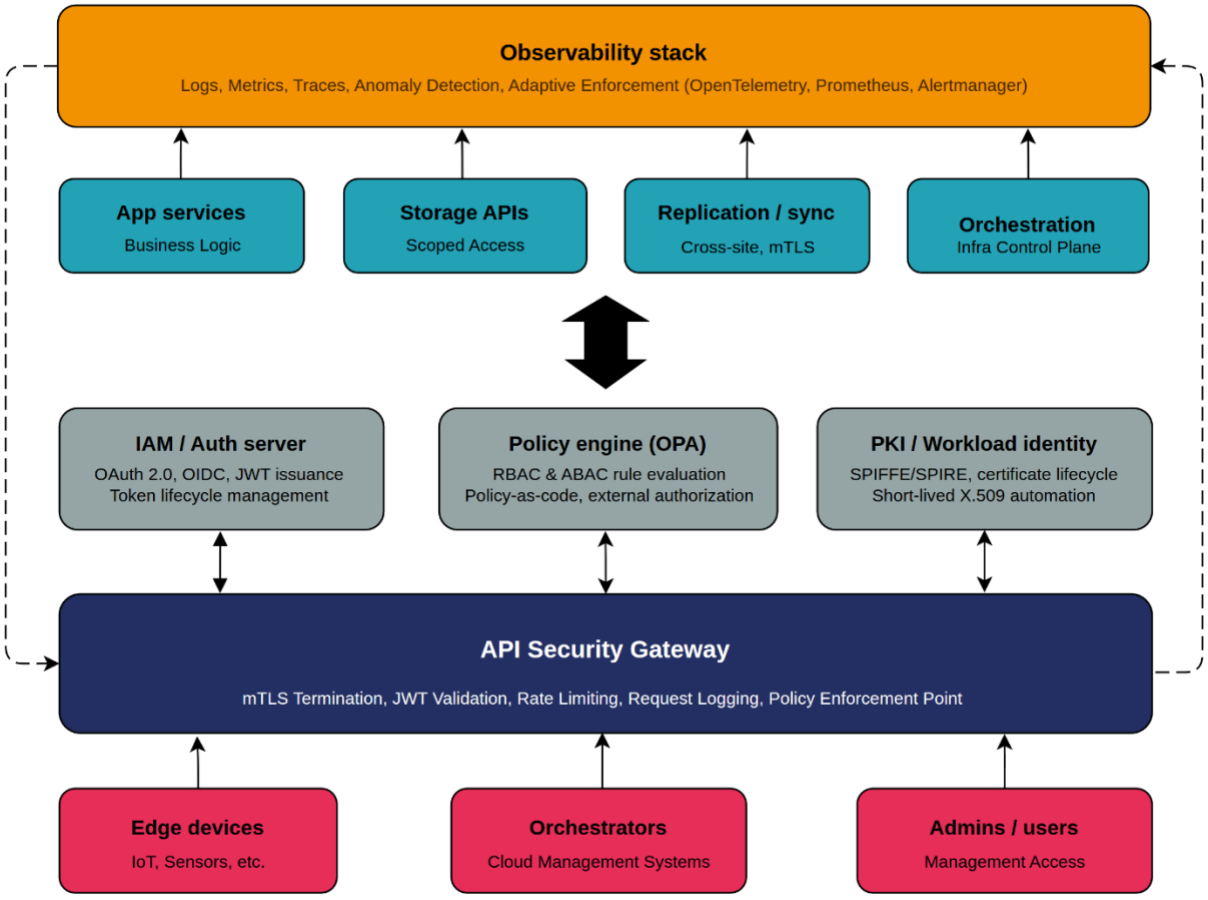

Figure 1 - The proposed reference architecture for edge cybersecurity management based on an API security gateway

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

5.1.1 Ingress / API Gateway Cluster

The API gateway layer acts as the primary entry point for all inbound and inter-site traffic. Every API request is routed through this layer, where it is authenticated, validated, and filtered before reaching internal services. In practice, the gateway handles TLS termination where appropriate, enforces mTLS for service-to-service communication, validates JWTs, and applies coarse-grained access controls. Only requests that meet these baseline requirements are forwarded further into the system. Technologies such as Envoy gateway, Kong gateway, and Apache APISIX all provide these capabilities, although they differ in how they structure policy and extensibility.

5.1.2 Authorization / Policy Decision Service

Behind the gateway sits the authorization or policy decision service. While the gateway enforces decisions, this component is responsible for making them. A policy engine such as Open Policy Agent (OPA) allows authorization logic to be defined externally as policy code, rather than embedded in applications. Integration with the gateway, commonly through external authorization interfaces, allows each request to be evaluated against fine-grained rules before it is accepted.

5.1.3 IAM / Authorization Server

Identity management is handled by an IAM or authorization server, which issues and manages access tokens. This component provides OAuth 2.0 and OpenID Connect functionality, including token issuance, key management, and claim definition. A commonly used reference implementation is Keycloak, which supports standard protocols and allows roles, scopes, and claims to be centrally defined. By relying on open standards, gateways and services can validate tokens locally without tight coupling to a specific vendor.

5.1.4 Workload Identity / PKI Service

The workload identity and PKI layer provides cryptographic identities to services and infrastructure components. In dynamic environments, where workloads are frequently created and destroyed, automated identity issuance is essential. Frameworks such as SPIFFE and SPIRE are particularly suited to this purpose, as they provide attested identities and support federation across trust domains. This aligns closely with the need to establish trust between distributed edge sites.

5.1.5 Certificate Lifecycle Automation

Closely related to the workload identity and PKI layer is certificate lifecycle automation, which ensures that certificates are issued, renewed, and revoked without manual intervention. In Kubernetes-based environments, cert-manager can request and renew certificates from private or public issuers. Vault PKI and step-ca both support automated issuance of short-lived X.509

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

certificates. Automating this process reduces operational overhead and supports the Zero Trust objective of minimizing long-lived trust.

5.1.6 Observability Stack

An observability stack provides visibility into system behavior. This includes logs, metrics, and traces collected from gateways and supporting services. OpenTelemetry offers a vendor-neutral approach to instrumentation. Besides, Prometheus and Alertmanager are widely used for metrics and alerting. Together, these components enable continuous monitoring and support both operational and security analysis.

5.2 Technology Choices and Justification

Different technologies can be used to implement each layer while preserving the conceptual consistency of the architecture. The following choices reflect a balance between flexibility, standardization, and alignment with Zero Trust principles. In the following, we elaborate on a couple of existing technologies that we briefly referred to in the previous section.

At the API gateway layer, an Envoy-based approach, such as Envoy gateway, is a strong reference choice. It provides native support for mTLS, JWT validation, and integration with external policy engines. Its design fits well with distributed, cloud-native environments and allows policy enforcement to be standardized across multiple sites. This makes it particularly suitable for an architecture that may be adopted by different partners. Alternatives such as Kong Gateway offer a more packaged API management experience, including lifecycle and governance features. This can be advantageous in environments where API management is a primary concern. Apache APISIX provides a flexible, plugin-driven model with strong performance characteristics, although it may require more effort to align with a unified policy framework.

For policy enforcement, Open Policy Agent is well suited as a dedicated decision point. It enables a policy-as-code approach and integrates cleanly with gateway layers. While some platforms allow authorization logic to be embedded directly in gateways or IAM systems, separating policy evaluation generally leads to better consistency and governance over time.

For identity and token services, Keycloak is a practical reference implementation. It supports OAuth 2.0 and OpenID Connect, provides token issuance and validation capabilities, and allows roles and claims to be centrally managed. In environments where an enterprise identity provider already exists, it can be used instead, provided it adheres to the same standards. A key design choice is to favor JWT-based access tokens that can be validated locally at the gateway, reducing reliance on centralized introspection.

Workload identity is best addressed using SPIFFE and SPIRE, which provide a standardized way to assign and verify identities across dynamic environments. Alternative PKI-based approaches can be used, but they typically lack the same level of workload attestation and federation support.

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

For certificate lifecycle management, tools such as cert-manager, HashiCorp Vault PKI, and step-ca provide reliable automation. The key requirement is not the specific tool, but the ability to issue and rotate short-lived certificates in a controlled and auditable way.

Observability should be standardized around OpenTelemetry for data collection, even if different storage or analysis platforms are used downstream. This ensures that telemetry remains consistent across sites and partners.

A key consideration in the technology selection is alignment with the minimal hardware design objective of the MISD project. By centralizing security enforcement at the API gateway layer, individual workloads and services are relieved from implementing their own authentication and policy logic. This reduces the computational and memory footprint required per service, which is particularly relevant for edge deployments operating on constrained hardware. The selected components (Envoy, OPA, Keycloak, SPIRE) are designed for containerized environments and can operate within resource budgets typical of edge nodes.

5.3 Integration with IAM, PKI, Policy Engines, and Observability Tools

The effectiveness of the architecture depends less on individual components and more on how they are integrated. The IAM layer acts as the authoritative source for identities, roles, scopes, and token issuance. Gateways are configured to trust the IAM issuer and retrieve metadata, such as signing keys, through standard discovery endpoints. In most cases, access tokens should be validated locally at the gateway using JWT signatures. Only in more sensitive scenarios token introspection or revocation checks might be required. A consistent token structure (e.g., by using RFC 9068 [10]), including claims such as issuer, audience, subject, and role information, ensures interoperability across components.

The PKI layer provides identities for gateways, workloads, and inter-site endpoints. Certificate issuance must be automated and tied to policy, ensuring that only authorized entities receive credentials. Gateways should trust only approved certificate chains, and certificate attributes should map clearly to workload or site identity. Where SPIFFE/SPIRE is adopted, workload identities can be issued as SPIFFE Verifiable Identity Documents and used for mTLS between workloads and gateways.

The policy engine operates as a dedicated authorization decision point that handles RBAC checks, ABAC checks, cross-site policy, service relation allowlists, and risk- or context-aware controls. It evaluates requests based on inputs such as JWT claims, certificate attributes, request metadata, and environmental context. This creates a clear separation of responsibilities: the gateway enforces decisions, the policy engine defines them, and IAM and PKI provide the underlying identity data. This separation reduces duplication and helps maintain consistency across services.

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Finally, the observability layer ties the system together by providing visibility into all interactions. Gateways should emit logs for authentication and authorization events, token validation failures, certificate issues, and rate-limiting actions. These logs, along with metrics and traces, are collected locally and can be aggregated centrally. A minimal set of security-relevant telemetry should include failed mTLS handshakes, invalid certificates, rejected tokens, denied policy decisions, and unusual request patterns. This data supports both real-time response and longer-term analysis.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD·EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

6 Existing Benchmark Results

Stronger security doesn't come free. The authentication and policy enforcement mechanisms that ZTA requires add processing overhead to every request, and understanding how much is important before committing to an implementation. In this section, we briefly summarize the performance analysis of various API security gateways reported in the literature to better understand the performance trade-offs involved in using an API gateway to harden the security of edge data centers. We invite readers to refer to the discussed publications for extensive comparisons of API gateway implementations.

Liu et al. [3] briefly compare multiple open source API gateways and analyze the performance of their Kong API gateway instance that is configured with various security authentication plugins (Basic, Key, JWT, OAuth 2.0, LDAP, and IP restriction), load balancing, and a reverse proxy. They test the gateway on an edge computing platform with 100 back-end micro-services. They measure the RTT for requests using the API without any authentication and ones using different authentication plugins. They show that the average RTT is always lower than 0.023s (see Table 1) which is below the average visual reaction time of users. Authors also report on the gateway's data processing performance, arguing that it satisfies load balancing requirements.

Table 1: Round trip time test results (source: [3])

Test Item	Average time
Round trip time for using the microservices API without any authentication	0.004s
Round trip time for using the microservices API with Basic authentication	0.0078s
Round trip time for using the microservices API with Key authentication	0.0065s
Round trip time for using the microservices API with JWT authentication	0.0189s
Round trip time for using the microservices API with OAuth 2.0	0.0205s
Round trip time for using the microservices API with LDAP authentication	0.0223s
Round trip time for using the microservices API with IP restriction	0.0111s

Moreira et al. [5] introduce AGE, a service to automatically deploy and configure API gateways starting from their documentation, load tests based on user-defined workloads and monitor the performance of the gateway instance. Their tool finally returns a simple metric to help users decide which gateway implementation better fits their use case. AGE supports three API gateway implementations: Kong, KrakenD, and Tyk with the possibility to be extended with new options. Authors further conduct a proof-of-concept to demonstrate the usefulness of their tool in comparing API gateway implementations. Under the specific workloads in that study, KrakenD outperformed Kong and Tyk as we can see in Table 2. Note that this outcome should not be assumed to hold across all workload types.

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Table 2: Overall AGE score (source: [5])

API Gateway	Kong	KrakenD	Tyk
Global score	80	94	90

Çakıcı [6] presents a performance evaluation of four standalone API gateway implementations (APISIX, Kong, KrakenD, and Tyk) as well as four tools that offer some API gateway -like features (Express Gateway, YARP, Ocelot, and Nginx). The benchmark comprised of 5 identical Go applications, and each gateway was configured to **round-robin** between those five services. The results of their benchmark are given in Table 3, suggesting that each implementation may outperform others in certain aspects at the expense of a worse performance in others. For example, while APISIX shows the highest handled requests per second, it also consumes the highest amount of memory. The author suggests that the selection of an API gateway is not just about the performance, but also the supported features, user-friendliness, and how well its architecture fits the entire ecosystem.

Table 3: API gateways performance benchmark (source: [6])

Gateway	RPS	CPU (%)	Memory (MB)	P50 Latency (ms)	P99 Latency (ms)	Mean Response Time (ms)	Image Size (MB)	Language
APISIX	31584.47	203.7	252.5	2.47	528.6	60.44	382.9	Lua
Kong	29972.47	203.3	66.3	2.76	528.0	59.91	405.6	Lua
YARP	26877.64	204.7	114.7	2.77	538.1	61.93	142.1	.NET
Nginx	26213.51	170.8	16.3	3.34	532.8	60.11	197.7	C
Tyk	25961.50	268.9	153.8	3.40	512.4	59.67	208.8	Go
Ocelot	24188.34	239.9	152.6	3.69	511.6	58.59	257.0	.NET
KrakenD	23688.75	220.9	110.1	4.43	480.9	55.93	124.0	Go
Express Gateway	6350.93	105.9	132.2	31.18	43.1	31.65	132.8	Node.js

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Finally, individual API gateway developers also publish benchmark results [7,8,9], typically comparing their tool to one or more of other popular gateway implementations. We exclude these benchmarks in this report to avoid potential comparison biases.

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

7 Conclusions

Cybersecurity management in a democratized edge-cloud continuum is not a problem that can be solved with a single tool or a one-time configuration. The environments in question are inherently dynamic: workloads move, sites come online and go offline, and the organizations operating them may change over time. A security model that works under stable, well-bounded conditions will not hold up here.

Zero Trust Architecture provides the right starting point precisely because it makes no assumptions about stability. Every entity must prove who it is, every request is evaluated on its own merits, and access is granted only to the extent actually needed. These are not ideals to aspire to; they are operational requirements in an environment where a breach at one edge site could otherwise propagate freely across the rest.

The API gateway-centric approach proposed in this report offers a practical way to enforce these principles without demanding that every service in the ecosystem take on the full complexity of identity management and policy enforcement. By positioning the gateway as a consistent enforcement layer, the architecture achieves uniform security controls across sites that may be operated by different teams, running different infrastructure stacks. Policy can be updated centrally and applied immediately, without redeploying workloads. Observability comes naturally from the gateway's position in the request path, rather than requiring a separate instrumentation effort for each service.

The performance benchmarks reviewed in this report suggest that the overhead introduced by gateway-based security is acceptable for production use. Authentication mechanisms such as JWT and OAuth 2.0 add latency in the range of tens of milliseconds, which is within normal bounds for API-driven interactions. However, the benchmarks reviewed in this report are drawn from existing literature and controlled test environments. A dedicated performance evaluation conducted under conditions representative of the target deployment, including realistic edge hardware, variable connectivity, and mixed workload types, would provide stronger empirical grounding for implementation decisions.

The cybersecurity management model proposed in this report will serve as the architectural foundation for the Infrastructure Protection System (IPS) being developed within WP3. The reference implementation outlined in Section 5 provides a concrete starting point for the software development activities in D3.4, which are already underway. Subsequent deliverables will address the graphical user interface (D3.6), lab deployment and testing (D3.7), and end-user service validation (D3.8).

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

References

[1] R. Yazdani, and S. Kuhn, “Joint Cybersecurity Needs and Acceptance Criteria Research Aimed at End-users and All Other Key Stakeholders”, MISD D3.1, June 2025.

[2] S. Anasuri, "Zero-Trust Architectures for Multi-Cloud Environments", *International Journal of Emerging Trends in Computer Science and Information Technology* 3.4, 64-76, 2022.

[3] S. Liu, Z.J. Zhang, Y. Cui, and Y. Zhang, "An Industrial-Grade API Secure Access Gateway in the Cloud-Edge Integration Scenario", *International Conference on Smart Grid and Internet of Things,* Springer International Publishing, 2020.

[4] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero Trust Architecture”, Tech. Rep. Special Publication 800-207, National Institute of Standards and Technology, Gaithersburg, MD, August 2020.

[5] P. Moreira, A. Ribeiro, and J.M. Silva, "Age: automatic performance evaluation of API gateways", *IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2023.

[6] Zahid Çakıcı, “API Gateway Performance Benchmark”, https://medium.com/code-beyond/api-gateway-performance-benchmark-407500194c76, June 2025.

[7] “API Gateway Performance Benchmarks with KrakenD”, https://www.krakend.io/docs/benchmarks/, October 2016.

[8] “API Gateway Benchmarking with KrakenD”, https://www.krakend.io/docs/benchmarks/api-gateway-benchmark/, October 2016.

[9] “Tyk Performance Benchmarks”, https://tyk.io/performance-benchmarks/.

[10] V. Bertocci, "RFC 9068: JSON Web Token (JWT) Profile for OAuth 2.0 Access Tokens", 2021.

MODULAR INTEGRATED SUSTAINABLE DATACENTER

Rijksdienst voor Ondernemend Nederland

8ra CLOUD-EDGE CONTINUUM

This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)

Appendices

A.1 List of Acronyms

ABAC	Attribute-Based Access Control
API	Application Programming Interface
IAM	Identity and Access Management
IAMS	Identity and Access Management Services
JWT	JSON Web Token
LDAP	Lightweight Directory Access Protocol
mTLS	Mutual Transport Layer Security
MISD	Modular Integrated Sustainable Datacenters
OIDC	OpenID Connect
OPA	Open Policy Agent
PEP	Policy Enforcement Point
PKI	Public Key Infrastructure
RBAC	Role-Based Access Control
RTT	Round Trip Time
SDN	Software-Defined Networking
SPIFFE	Secure Production Identity Framework for Everyone
SPIRE	SPIFFE Runtime Environment
ZTA	Zero Trust Architecture

MODULAR INTEGRATED SUSTAINABLE DATACENTER
Rijksdienst voor Ondernemend Nederland
8ra CLOUD-EDGE CONTINUUM
This activity is conducted in the scope of Modular Integrated Sustainable Datacenter (MISD) project which received funding from the Netherlands Enterprise Agency (RVO) under the 8ra initiative (IPCEI-CIS)