Network Working Group
Internet-Draft
Intended status: Experimental
Expires: 10 April 2026

Y. OIWA
AIST Japan
S. Kanno
Y. Sakemi
GMO CONNECT

7 October 2025

Secure Hybrid Network Monitoring - Path Characteristics Service
draft-oiwa-path-characteristics-service-00

Abstract

"Secure hybrid network monitoring - Problem statement" identifies challenges in securing and monitoring networks deployed across hybrid and mixed cloud environments. This document introduces the Path Characteristics Service (PCS), a framework that enables applications and operators to obtain and evaluate verifiable information about the characteristics of the network paths they use. It outlines a non-normative architecture and interfaces for PCS and explains how PCS can help address the identified challenges; it does not define protocol requirements.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 10 April 2026.

Copyright Notice

Table of Contents

1. Introduction

Virtualized resources such as cloud computing infrastructure are rapidly replacing traditional network and computing environments, including on-premises servers and locally managed clusters. In such infrastructures, the physical characteristics of resources - e.g., server location, local network topology, or the operators of network devices - are typically hidden from users in exchange for flexibility, redundancy, and cost benefits. At the same time, cryptographic protection mechanisms such as TLS or IPsec are widely used to secure communications into and out of these systems.

However, as identified in "Secure hybrid network monitoring - Problem statement" [I-D.oiwa-secure-hybrid-network], there remain many cases where application-level security depends not only on encrypted communication channels but also on specific properties of the underlying network and intermediate nodes. Examples include:

* Sensitivity to traffic analysis, where encrypted flows may still leak metadata;

* Legal or regulatory requirements mandating that certain properties (e.g., jurisdiction, physical location, or operational control) be verifiable;

* Threats such as Denial-of-Service (DoS) attacks, which cannot be prevented solely through encryption.

In non-virtualized, self-managed networks, operators can use existing mechanisms (e.g., NETCONF, path validation) to obtain status and operational information about network components. These mechanisms are not sufficient in modern hybrid or multi-cloud settings, where visibility into the underlying infrastructure is significantly limited.

To address these gaps, this document introduces the Path Characteristics Service (PCS) as a technical approach for continuously obtaining and verifying relevant characteristics of the network paths used in complex environments such as hybrid or multi-cloud deployments. PCS is intended to provide a common framework for securely obtaining, interpreting, and acting upon path-level information, thereby enabling high-security applications to maintain trust in the network even in the presence of virtualization and limited direct control.

This document builds upon the problem statement and gap analysis presented in [I-D.oiwa-secure-hybrid-network], and outlines a potential PCS architecture and its role in addressing the identified challenges.

PCS is a framework for obtaining, synthesizing, and evaluating verifiable information about the characteristics of network paths used by an application or tenant. In this document, "path characteristics" include properties that can affect security or compliance (e.g., jurisdictional residency, operator identity along the path, path segments and facilities traversed, transport security properties, or exposure to known risks). PCS defines:

* (1) mechanisms to obtain path-related evidence from multiple sources,

* (2) methods to reconstruct a coherent view of the path and its attributes from such evidence,

* (3) a policy-matching mechanism that evaluates whether a given path conforms to declarative requirements.

PCS does not mandate a specific transport or routing technology, is not itself a traffic-measurement protocol, and does not replace existing routing- or control-plane security mechanisms. Rather, it provides a verifiable interface and data model that higher-layer systems can use to reason about path trust and to trigger operational actions.

1.1. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 RFC2119 [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Design Principles

2.1. General

To overcome these problems, we propose to design a distributed architecture for assuring the network operation integrity for mixed and hybrid cloud applications. Such a system should:

* Have a modeling of the network infrastructure in two dimensions: one axis in parallel to the network paths and forwarding directions, and the other axis for the layers of protocols.

* Have enough knowledge on the complex dependency of software and protocols; not only the network packet-forwarding technologies but also surrounding areas such as addressing and DNS must be covered.

* Have explicit handling of tunneling and virtualization aspects, both on protocol level (e.g. VPNs, IP-IP, IPsec) and on infrastructure level (IaC, Network-as-a-Service, Wavelength Division Multiplexing, etc.)

* Consolidate operation information at each operator’s level and consider their pre-determined operation principles for evaluating integrity.

* Address management-oriented risks of infrastructure management, including non-network aspects.

A possible implementation of such a system could leverage distributed network security coordination between operators and users of cloud and network infrastructure. Rather than adopting a "disclose all" approach, this design would maintain both flexibility and efficiency for multi-cloud applications.

In particular, telemetry as defined in [RFC9232] can be utilized to clarify the state of monitored communications. By employing standardized telemetry mechanisms, it becomes possible to collect, aggregate, and share relevant operational data about network paths and security status without exposing sensitive internal details. This approach enables stakeholders to verify the integrity and security of communications across hybrid and multi-cloud environments, while respecting the confidentiality requirements of each operator.

In particular, PCS can clarify the status of monitored communications by utilizing telemetry defined in [RFC9232]. By adopting standardized telemetry information, it is possible to collect, aggregate, and share relevant operational data related to network paths and security status without extending the specifications of existing networks. This allows for the verification of communication integrity and security in hybrid and multi-cloud environments without exposing sensitive internal details. This approach respects the confidentiality requirements of each operator while enabling stakeholders to verify communication integrity and security.

2.2. System Capabilities

The Secure Hybrid Network Monitoring system SHOULD:

* Have the ability to state network security requirements from an infrastructure user to infrastructure providers. In a hybrid cloud or layered systems, it will include communications between operators of infrastructure/cloud systems.

* Have the ability to return status statement for the current provisional status against given requirements.

* Provide some choices on the transparency levels about the internals of the cloud service infrastructure.

* Have some traceability provisions for troubleshooting, if there are opacities in network status statement replies.

* Have enough consideration for various tunneling and virtualization technologies.

* Have a bidirectional interface to system-level security management systems, such as Continuous Diagnostics and Mitigations (CDM) dashboards.

3. Architecture Overview

The PCS architecture is organized into three conceptual layers, shown in {#fig-pcs-layers}:

* *PCS Layer* - Hosts PCS Servers and Clients, which exchange queries and responses regarding path characteristics. A PCS Server gathers data from one or more Telemetry Layer components, reconstructs path-level views, and evaluates them against applicable policies. PCS instances may also recursively query other PCS instances in adjacent administrative domains to obtain a broader or deeper view of the path.

* *Telemetry Layer* - Provides standardized access to measurements and topology information, collected by Telemetry Collectors from the underlying network. Depending on operator policy and technical constraints, this may include performance metrics, topology summaries, or security-related status. PCS uses defined APIs to query this layer, allowing integration with existing protocols (e.g., NETCONF, gNMI, BGP-LS, SNMP).

* *Network Layer* - Comprises the physical and virtual network elements such as routers, switches, links, VPN endpoints, and SDN-controlled domains. These elements generate raw operational data, which is exposed upward through the Telemetry Layer.

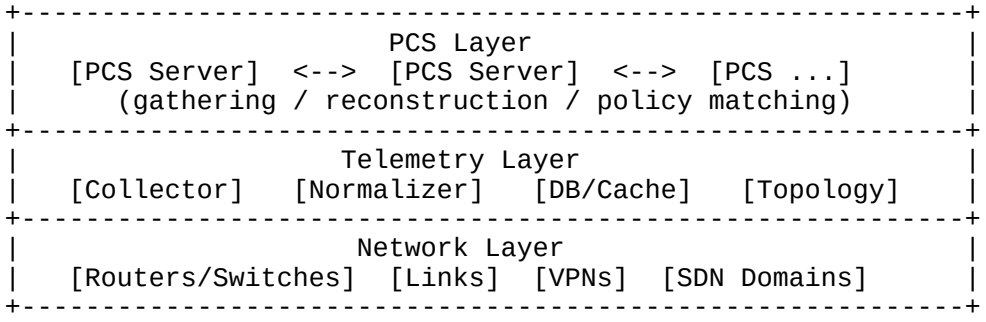

Figure 1: Three-layer model for secure hybrid network monitoring

3.1. Recursive Composition of PCS

PCS supports recursive composition, whereby a PCS Server MAY act as a client of other PCS Servers to obtain path-segment information across administrative boundaries. This enables path characteristics to be gathered at multiple granularities: *point-level* (e.g., device, hop, facility) and *plane-level* (e.g., segment, domain, cloud region). A recursive PCS deployment can therefore provide a coherent PathView even in deeply nested environments (e.g., VPN-over-VPN, SDN slices over WAN, multi-cloud overlays).

3.1.1. PCS-to-PCS Query Flow

At a high level, a PCS Server receiving a client query proceeds as follows (non-normative):

1. *Scope decomposition:* Partition the end-to-end path into one or more *sub-scopes* (e.g., by AS/domain, tunnel boundary, cloud region).

2. *Local evidence collection:* For sub-scopes under local administration, gather evidence from the Telemetry Layer and reconstruct local PathView fragments.

3. *Federated queries:* For external sub-scopes, issue *PCS-to-PCS* sub-queries to authoritative PCS Servers for those domains, specifying the requested properties, freshness, and privacy constraints.

4. *Composition:* Merge local and federated fragments into a single PathView, preserving provenance and confidence metadata.

5. *Policy evaluation:* Apply policy matching on the composed PathView and return the result to the client.

A simplified message sketch (non-normative):

```
Client -> PCS(Server-G): Query{target, required_properties, freshness, recursion_limit}
PCS(Server-G) -> PCS(Server-A): SubQuery{scope=A, required_properties..., freshness, trai
l}
PCS(Server-A) -> Telemetry(A): gather()
PCS(Server-G) <- PCS(Server-A): PathView{A}, evidence_refs, confidence
PCS(Server-G) -> PCS(Server-B): SubQuery{scope=B, ...}
PCS(Server-G) <- PCS(Server-B): PathView{B}, ...
Client <- PCS(Server-G): Result{PathView{A-B}, policy_decision, trace, completeness}
```

3.1.2. Stop Conditions and Loop Prevention

To avoid unbounded recursion, a PCS Server MUST implement explicit stop conditions and loop prevention. The following mechanisms are RECOMMENDED:

* *Recursion limit:* A request header recursion_limit (non-negative integer). Each PCS Server decrements it when forwarding sub-queries. If it reaches zero, further delegation MUST NOT occur; the server returns incomplete=true for unexplored scopes.

* *Visited set / trail:* Each sub-query SHOULD carry an ordered trail including {request_id, caller_domain, scope}. A PCS Server MUST detect a cycle when its own domain/scope appears again and terminate delegation for that branch.

* *Scope contraction:* A PCS Server SHOULD only delegate the *minimal* missing scope. Overlapping or redundant sub-queries SHOULD be coalesced to reduce fan-out.

* *Cache with freshness:* Sub-query results MAY be cached with explicit issued_at/exp (or fresh-until) metadata. Reuse is allowed only if freshness requirements are met.

* *Cut failure semantics:* When a sub-scope cannot be explored (timeout, policy denial, or recursion limit), the composed PathView MUST mark the segment as opaque and set completeness=partial.

3.1.3. Security, Trust, and Provenance in Recursion

* *Authentication:* PCS-to-PCS exchanges MUST be mutually authenticated (e.g., mTLS or equivalent).

* *Authorization and least disclosure:* Responding servers MAY honor privacy constraints (e.g., return *plane-level* summaries instead of *point-level* details) while still signing the returned assertions.

* *Provenance:* Each returned fragment SHOULD include a signed provenance block (issuer, scope, time, signature). When composing, the caller MUST preserve the chain of provenance for third-party verification.

* *Non-amplification:* A PCS Server MUST NOT act as an open relay. Rate limits and request shaping SHOULD apply to delegated queries.

3.1.4. Result Composition and Conflict Handling

When multiple fragments overlap or disagree:

* *Priority rules:* Prefer fragments issued by the authoritative domain for that scope. Otherwise, prefer newer and higher-confidence evidence.

* *Conflict marking:* If conflicts remain, the composed PathView MUST annotate the affected elements with conflict=true and lower confidence.

* *Granularity reconciliation:* Plane-level evidence MAY satisfy policies that require only aggregate properties; point-level evidence is required when policies demand specific node attributes.

3.1.5. Example Fields (non-normative)

A sub-query MAY include:

```
{ "scope": {"domain":"AS65001", "segment":"vpn:1234"},
"required_properties":
["jurisdiction","operator","tunnel.integrity"], "freshness":
{"max_age":"300s"}, "recursion_limit": 2, "privacy":
{"granularity":"plane"}, "trail":
[{"domain":"example.net","req":"abc123"}] }
```

A fragment response MAY include:

```
{ "scope": {"domain":"AS65001"}, "path_view": {...}, "provenance": {"
issuer":"pcs.as65001.net","issued_at":"2025-08-
15T02:10Z","sig":"..."}, "completeness": "partial", "confidence":
0.87 }
```

3.1.6. Non-Goals

Recursive PCS does *not* attempt to: * control routing or steer traffic; * force disclosure of internal topology beyond the responder’s policy; * guarantee global completeness in the presence of non-cooperating domains.

Non-normative note: Recursive PCS enables operators to obtain *point-level* evidence where permitted, while still producing *plane-level* assertions when detailed disclosure is not available, thus delivering useful policy decisions even in deeply nested hybrid environments.

4. Roles and Responsibilities

This section defines the main roles in a PCS-enabled environment. The roles align with the three-layer model described in {#fig-pcs-layers}.

* *Network Operator / Domain Owner* - Responsible for the physical and virtual network infrastructure. They control the disclosure policy for topology and telemetry data, and may operate or delegate the Telemetry Layer components. They are the authoritative source of truth for the properties of network elements within their administrative domain.

* *Telemetry Collector / Aggregator* - Operates within the Telemetry Layer to gather and normalize data from network devices and services. Collectors may use standardized protocols (e.g., NETCONF, gNMI, BGP-LS, SNMP) or vendor-specific mechanisms. Aggregators combine data from multiple collectors or domains, and may provide cached or summarized information to PCS Servers.

* *PCS Server* - Hosts the PCS functionality, including querying the Telemetry Layer, reconstructing path-level characteristics, and applying policy matching. A PCS Server may recursively query other PCS Servers in adjacent domains to extend its view beyond local telemetry.

* *PCS Client* - An application or system component that requests path characteristics from a PCS Server. Clients may use the information to make operational or security decisions, such as selecting a compliant path or avoiding a path with undesirable properties.

5. PCS Main Functions

The PCS Layer provides three main functional stages when processing a path characteristics query:

* *Gathering* - Collects relevant evidence from the Telemetry Layer and, where applicable, from other PCS Servers via recursive queries. Evidence may include metrics, topology fragments, operational status, and security-relevant events. Data sources can be heterogeneous and may vary in freshness and granularity.

* *Reconstruction* - Builds an end-to-end "PathView" from the gathered evidence. Reconstruction may operate at different levels of abstraction:

  - *Point-level view* - Characteristics for each individual network element (e.g., router, link).

  - *Domain-level view* - Aggregated characteristics for an entire administrative domain or SDN segment. The reconstruction process resolves conflicts, fills gaps using partial data, and ensures that the resulting PathView is internally consistent.

* *Policy Matching* - Compares the reconstructed PathView against a set of predefined policies provided by the PCS Client. Policies may include security requirements (e.g., encryption in transit, jurisdiction constraints), performance thresholds (e.g., maximum latency), or operational constraints (e.g., avoiding specific domains). The PCS Server returns a compliance result, possibly with annotations explaining non-compliance or uncertainty.

6. Path Characteristics Service (PCS)

The Path Characteristics Service (PCS) provides an authenticated and access-controlled endpoint for requesting and receiving status statements regarding the characteristics of network paths. PCS is typically operated by network operators or connectivity providers, or it may also be offered by third-party service providers or cloud operators. It answers queries about the real-time or recent status of network paths to authenticated clients.

In multi-stakeholder environments, such as hybrid or multi-cloud deployments, a PCS Server may query other PCS Servers operated by different providers. This recursive gathering enables a PCS to return aggregated and policy-filtered status information to the requesting client.

6.1. Identification and Authentication

* PCS endpoints MUST be access-controlled and confidentiality-protected using secure protocols (e.g., TLS 1.3 or later).

* Clients MUST be strongly authenticated, for example via OAuth 2.1, mutual TLS (mTLS), or OpenID Connect.

* Authentication credentials SHOULD be bound to a specific connectivity channel, such as:

  - Physical (layer-1) leased lines,

  - Layer-2 segments (e.g., VLAN, VXLAN),

  - Virtual private network (VPN) tunnels,

  - SD-WAN overlay paths.

* If multiple connectivity channels exist under a single business contract, multiple identifiers may be associated with a single authentication session (TBD: operational policy).

6.2. Subscription for Status Statements

PCS protocols SHOULD support both: * *Streaming (push)* - Clients subscribe to a query and receive updates when relevant changes occur. * *Polling (pull)* - Clients periodically retrieve the current status, with configurable intervals.

Subscription parameters (e.g., polling interval, event triggers, maximum update rate) SHOULD be negotiable between the PCS Client and PCS Server.

7. PCS Query

A PCS Query is the primary mechanism by which a PCS Client requests path characteristics information from a PCS Server.

A query typically includes: - *Target Path* - The path or set of candidate paths for which characteristics are requested, identified by endpoints, AS-paths, or other topology identifiers. - *Requested Characteristics* - Specific metrics or properties to be returned (e.g., jurisdiction, encryption status, latency). - *Policy Set (optional)* - Policies to be applied for Policy Matching, expressed in a structured form understood by the PCS Server. - *Query Constraints* - Optional parameters such as maximum acceptable data age, recursion depth, or partial data acceptance.

The PCS Server processes the query by: 1. Gathering evidence from the Telemetry Layer and, if necessary, from other PCS Servers. 2. Reconstructing the PathView from the gathered data. 3. Performing Policy Matching if a policy set is provided.

The server returns a *PCS Response* that may contain: - The reconstructed PathView (point-level and/or domain-level view). - Policy Matching results (pass/fail/unknown) for each policy. - Metadata such as data age, sources used, and recursion depth reached.

Depending on deployment and query complexity, PCS queries may be handled synchronously or asynchronously. In the asynchronous case, the initial response includes a job identifier that can be used to retrieve results later.

Access to PCS Query endpoints MUST be subject to authentication and authorization controls, and responses MAY apply redaction or aggregation according to the policies of the Network Operator or Domain Owner.

7.1. Connectivity Properties

* List of desired connectivity properties declares what kind of network nodes (both network nodes and edges) the communication packets will be allowed to flow over.

7.2. Properties for nodes

Possible property requests for a network node will include at least:

* operator
* geo-location
* supplier
* model
* hardware ID
* the name and version of the running software
* the security status of the node
* the security status of the operator
* required assurance level (see below)

7.3. Properties for edges

Network edges may be categorized into:

* A physical network edge
* A network tunnel
* A software-defined network

Possible property requests for a physical network edge will include at least:

* operator
* geo-location
* the protocol type of the physical network
* the security status of the operator
* required assurance level (see below)

Possible property requests for a network tunnel will include at least:

* operator
* geo-location
* (nested) path property request for the underlying network
* the identification of the tunnel
* the protocol type
* the strength of the integrity/confidentiality protection
* the security status of the tunnel
* the name and version of the software realizing the tunnel
* the security status of the operator
* required assurance level (see below)

Possible property requests for a software-defined network will include at least:

* operator
* geo-location
* (nested) path property request for the underlying network
* the name and version of the software realizing the network
* the security status of the network-defining software
* the security status of the operator
* required assurance level (see below)

7.4. Status statement and assurance levels

A status statement, which is a response to the query, will contain either evidence or a guarantee of the required network properties. There will be several types of assurance levels or types of status statement to be returned.

7.4.1. Traced present status statement

For traced status statement, the query will typically contain a requirement for specific node suppliers and types. The answer will contain a recorded trace of the path, signed with each traversed network nodes with their identifications. The information will ensure that the property is satisfied only at the present time. This type of status statement will require dedicated support for packet traces in every network node.

7.4.2. Transparent present status statement

For transparent status statement, the response will contain a list of traversed nodes and edges with their properties (as requested in the query). If the query contains requirements for networks operated by third parties (i.e. involving cascaded queries to other PCSs), the status statement will contain sub-status statement received from the third parties. The information will ensure that the property is satisfied only at the present time.

7.4.3. Traceable opaque present status statement

For traceable opaque status statement, the response will contain an opaque ID for the response. That ID has to correspond to the trace information which can be used by operators to identify the records for troubleshooting in the future. The information will ensure that the property is satisfied only at the present time.

7.4.4. Opaque present status statement

For opaque status statement, the response will contain just a positive or negative answer to the question. The information will ensure that the property is satisfied only at the present time.

7.4.5. Traceable opaque future status statement

For traceable opaque future status statement, the response will contain an opaque ID for the response. That ID has to correspond to the trace information which can be used by operators to identify the records for troubleshooting in the future. The information will ensure that the network is controlled in the way that the required property is kept satisfied, even when dynamic routing has been changed.

7.4.6. Opaque future status statement

For opaque status statement, the response will contain just a positive or negative answer to the query. The information will ensure that the network is controlled in the way that the required property is kept satisfied, even when dynamic routing has been changed.

7.5. Things to be considered:

* How to measure the security level of operators
  - Standards or de-facto standards for status sharing with security dashboards
* Details on specifications for real-world properties such as operators, suppliers, models, and geo-locations
* How to integrate and monitor application-level dynamic routing (e.g. DNS)
* Possible more-detailed specifications for network topology requirements
* Possible integration with RPKI and other global-level managements

8. Use cases

Secure Hybrid Network Monitoring with PCS will be shown with specific examples using several use cases.

8.1. Case 1: Data Residency / Sovereignty Compliance

Certain applications must ensure that communication paths remain entirely within a given legal jurisdiction. For example, a financial institution may require that all customer data traffic remains within Japan or the EU, avoiding any transit through networks located in other countries. PCS can verify this by reconstructing the network path and confirming that all intermediate hops are located within the allowed jurisdictions. If a violation is detected (e.g., a hop located outside the allowed set), the PCS Client can take preventive actions such as rejecting the path or raising an alert.

PCS role: * Gather geolocation and jurisdiction information for each hop in the path from telemetry sources. * Reconstruct the complete PathView with jurisdictional annotations. * Apply policy matching to ensure ’jurisdiction’ {allowed jurisdictions}‘.

8.2. Case 2: Critical Infrastructure Operator Validation

Some sectors, such as healthcare or energy, require that only approved network operators be involved in the transport of sensitive data. For example, a healthcare information system may require that all intermediate networks along a path are operated by organizations on a predefined whitelist. PCS can validate this by retrieving operator identifiers for each path segment and checking them against the policy.

PCS role: * Obtain operator identifiers and relevant cryptographic assertions from telemetry sources (e.g., RPKI, operator registries). * Reconstruct the PathView including operator information for each segment. * Apply policy matching to ensure ’operator’ {approved operators}‘.

8.3. Case 3: Incident Forensics and Audit

When a security incident occurs, operators may need to reconstruct and verify the exact network path taken by affected communications at the time of the incident. For example, during an investigation, a signed PathView from PCS can be used as part of an evidence package to demonstrate which networks were traversed and whether the path changed unexpectedly. This can also support regulatory audits that require verifiable historical path data.

PCS role: * Store PathViews with associated evidence, cryptographic signatures, and timestamps. * Provide mechanisms for retrieving historical PathViews for a given time window. * Allow independent verification of historical evidence using signatures and trust anchors.

9. Security Considerations

The proposed "Secure Hybrid Network Monitoring - Path Characteristics Service" itself introduces several security considerations that have to be addressed:

* *Information Disclosure*: The system must carefully balance the need for security monitoring with the protection of sensitive infrastructure information.

* *Trust Model*: Clear trust relationships must be established between different stakeholders in the hybrid cloud environment.

* *Authentication and Authorization*: Proper mechanisms must be in place to ensure only authorized entities can access monitoring information.

* *Integrity Protection*: The monitoring data itself must be protected from tampering or manipulation.

10. IANA Considerations

Currently, we assume that IANA actions are not necessary, but we plan to make corrections as necessary.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

11.2. Informative References

[ISO-IEC-5140] ISO/IEC, "Information technology - Cloud computing - Multi-cloud and hybrid cloud vocabulary", ISO/IEC 5140:2024, 2024.

[I-D.oiwa-secure-hybrid-network] Oiwa, Y., "Securing hybrid network - criteria and requirements", Work in Progress, Internet-Draft, draft-

oiwa-secure-hybrid-network-01, 4 November 2024,
<https://datatracker.ietf.org/doc/html/draft-oiwa-secure-hybrid-network-01>.

Acknowledgments

This work is based on results obtained from a project JPNP23013 commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

Authors' Addresses

Yutaka OIWA
National Institute of Advanced Industrial Science and Technology (AIST)
Email: y.oiwa@aist.go.jp

Satoru Kanno
GMO CONNECT Inc.
Email: kanno@gmo-connect.jp

Yumi Sakemi
GMO CONNECT Inc.
Email: sakemi-yumi@gmo-connect.jp