

Network Inventory YANG WG
Internet-Draft
Intended status: Standards Track
Expires: 31 August 2026

M. Palmero
Independent
C. Cardona
NTT
D. Lopez
Telefonica
I. Busi
Huawei
27 February 2026

A YANG Module for Entitlement Inventory
draft-ietf-ivy-entitlement-inventory-02

Abstract

This document defines a YANG data model for managing software-based entitlements (licenses, authorization tokens, pay-as-you-go service credentials...) within a network inventory. The model represents the relationship between organizational entitlements, network element capabilities, and the constraints that entitlements impose on capability usage.

This data model enables operators to determine what capabilities their network elements possess, which capabilities are currently entitled for use, and what restrictions apply. The model supports both centralized entitlement management and device-local entitlement tracking for physical and virtual network elements.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at <https://dr2lopez.github.io/ivy-capability-entitlement/draft-ietf-ivy-entitlement-inventory.html>. Status information for this document may be found at <https://datatracker.ietf.org/doc/draft-ietf-ivy-entitlement-inventory/>.

Discussion of this document takes place on the Network Inventory YANG WG Working Group mailing list (<mailto:inventory-yang@ietf.org>), which is archived at <https://mailarchive.ietf.org/arch/browse/inventory-yang/>. Subscribe at <https://www.ietf.org/mailman/listinfo/inventory-yang/>.

Source for this draft and an issue tracker can be found at <https://github.com/dr2lopez/ivy-capability-entitlement>.

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 31 August 2026.

Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	4
1.1. Scope of the Entitlement Model	5
1.2. Entitlement Deployment Models	6
1.2.1. Entitlement Provisioning	6
2. Conventions and Definitions	7
3. Modeling Capabilities and Entitlements	8
3.1. Foundational model: NetworkElement-Entitlements-Capabilities and Restrictions	8
3.1.1. Progressive Model Complexity	9
3.2. Capabilities	10
3.2.1. Extending Capability Classes	12
3.3. Entitlements	15
3.3.1. Reverse Mapping from Entitlements to Capabilities . .	19
3.4. Entitlement Attachment	19

3.5.	Installed Entitlements	20
3.6.	Implementation Considerations	21
3.6.1.	Level 1: Centralized Entitlement Inventory	21
3.6.2.	Level 2: Installed Entitlements on Assets	21
3.6.3.	Level 3: Capabilities Reporting	21
3.6.4.	Level 4: Capability-Entitlement Linkage	22
3.6.5.	Level 5: Restrictions Reporting	22
3.7.	Model Definition	22
3.7.1.	Model tree	34
4.	Implementation Examples and Validation Scenarios	36
4.1.	Overview of Examples	36
4.2.	Basic Structure	37
4.2.1.	Scenario	37
4.2.2.	Operational Context	38
4.2.3.	JSON Example	38
4.3.	Expired License Handling	39
4.3.1.	Scenario	39
4.3.2.	Operational Context	40
4.3.3.	JSON Example	40
4.4.	Utilization Tracking with Restrictions	44
4.4.1.	Scenario	44
4.4.2.	Operational Context	44
4.4.3.	JSON Example	44
4.5.	Hierarchical Entitlements	49
4.5.1.	Scenario	49
4.5.2.	JSON Example	49
4.6.	License Pooling	56
4.6.1.	Scenario	56
4.6.2.	JSON Example	56
4.7.	Multi-Vendor Environment	62
4.7.1.	Scenario	62
4.7.2.	JSON Example	62
4.8.	Component-Level Entitlements	70
4.8.1.	Scenario	70
4.8.2.	JSON Example	70
4.9.	Capability Class Extension	78
4.9.1.	Scenario	78
4.9.2.	JSON Example	78
5.	Operational Considerations	79
5.1.	Entitlement Synchronization	79
5.2.	Entitlement Expiration Handling	80
5.3.	Performance Considerations	80
5.4.	Migration and Version Compatibility	80
6.	IANA Considerations	80
6.1.	URI Registration	80
6.2.	YANG Module Name Registration	80
7.	Security Considerations	81
7.1.	Entitlement Data Sensitivity	81

7.2. Entitlement Tampering	81
7.3. Information Disclosure	81
8. References	81
8.1. Normative References	81
8.2. Informative References	81
Acknowledgments	82
Authors' Addresses	82

1. Introduction

Network elements provide capabilities i.e., functions related to their role in the network, such as MPLS routing, advanced QoS, or bandwidth throughput, which operators use to build services. Many capabilities require an evidence item for the right to use them, issued by the network element vendor, for their activation. These evidence items are called entitlements, and can take different forms, such as software licenses, access tokens or credentials for as-a-service consumption.

This document defines a YANG data model for tracking entitlements and their relationship to capabilities. The model supports three operational use cases:

- * Tracking entitlements held by the organization, their scope, and assigned holders
- * Representing capabilities available on network elements and whether entitlements permit their use
- * Monitoring active capability usage and enforced restrictions

Operators use this information to answer: What can this device do? What is its entitlement-id to do? What restrictions apply?

As network technology evolves toward modular, software-defined, and virtualized architectures, managing the rights to activate specific functions becomes increasingly complex. These rights, granted via entitlements, must be tracked, aggregated, and matched to assets to ensure that services can be delivered using available capabilities. This complexity calls for structured, machine-readable models that represent which capabilities are available, permitted, and in use.

This draft provides a foundational YANG structure for representing these relationships as standardized data, complementing the network inventory module.

1.1. Scope of the Entitlement Model

The entitlement model provides an inventory of entitlements. This includes the entitled holders and the capabilities to which they are entitled. Additionally, it offers information into the restrictions of the operation of the different assets (network elements and components). In general, this model seeks to address the following questions:

- * What entitlements are administered/owned by the organization?
- * How are entitlements restricted to some assets and holders?
- * What entitlements are installed on each network asset?
- * What constraints do the current installed entitlements impose on the network assets' functionality?
- * Does the entitlement impose any kind of global restrictions? What are they?
- * What are the restrictions that each network element has due to the entitlements it holds locally?

In this document, the term "installed entitlements" refers to entitlements that have been assigned to a particular network asset. The act of installation may involve directly provisioning the entitlement on the device or component, or it may represent a logical assignment in a centralized system. Some entitlements may be assigned to multiple network assets up to a defined limit; such constraints can be modelled as global restrictions under the entitlement.

The model supports entitlement tracking and capability management. It is intentionally designed to be extensible through YANG augmentation. Organizations requiring vendor-specific entitlement features should augment this base model rather than modifying it directly.

This model focuses on operational inventory of entitlements and capabilities. The following are explicitly out of scope:

- * Commercial aspects of entitlement acquisition and pricing
- * Entitlement migration policies between devices (vendor-specific)
- * Per-user access control mechanisms (covered by separate access control standards)

This model focuses on the ability to use capabilities, not on access control mechanisms. For example, if a router cannot enable MPLS due to entitlement restrictions, it means the organization lacks the rights to use that capability—even if access to the device itself is available. This distinction is separate from, for instance, the ability of a specific user to configure MPLS due to access control limitations.

1.2. Entitlement Deployment Models

Entitlements can be deployed and managed in different ways depending on the operational environment and vendor implementation. The following deployment models are commonly encountered:

- * ***Local Installation***: The entitlement is installed directly on the network asset, which maintains knowledge of its entitlements and enforces capability restrictions locally. This is a common approach for devices that operate independently.
- * ***License Server***: Entitlements reside in an external (license) server, which may be deployed on-premises or in the cloud. Network assets communicate with the license server to verify entitlement status and capability permissions. This model supports centralized management and dynamic entitlement allocation.
- * ***Commercial Agreement***: In some deployments, entitlements exist purely as commercial agreements, and policy enforcement occurs outside the network asset. The network asset may operate without direct knowledge of the entitlement, relying on external systems for compliance tracking.

This model is designed to be exposed by both network elements and license services. It provides mechanisms for each system to express the information it knows while being clear about the information it does not have, primarily through the presence or absence of containers. A network element should contain certain entitlement information, a license service other information, and a telemetry monitoring system could gather data from both sources to provide a complete picture.

1.2.1. Entitlement Provisioning

This model is not intended for automatic discovery of entitlements or capabilities through the network elements themselves. Instead, it assumes that entitlements and their associations are either:

- * Provisioned in a license server or asset database;

- * Installed on individual devices and reported through management interfaces; or
- * Manually configured as part of an inventory process.

Future augmentations may explore capability discovery or telemetry-driven models, but they are out of scope of the current version.

2. 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.

- * ToBeUpdated(TBU) Open Issue for the IVY WG, to include:

<<Update Glossary under Network Inventory draft, [BaseInventory]. We need at least formal definitions of "capability" and "entitlement".>>

- * **Capability:** A discrete function, feature, or resource that a network element is technically capable of performing when properly entitled. Examples include MPLS routing, specific bandwidth throughput, or advanced QoS features.
- * **Entitlement:** A vendor-issued authorization (typically a license) that grants permission to activate and use one or more capabilities on specific network elements, potentially subject to constraints such as time limits, usage quotas, or scope restrictions.
- * **Installed Entitlement:** An entitlement that has been locally activated on a network element and is available for use by that element's capabilities.
- * **Capability Restriction:** A constraint imposed by an entitlement that limits how a capability can be used (e.g., bandwidth cap, concurrent user limit, geographic restriction).
- * **Network Asset:** A network element or a component within a network element. The model supports entitlements and capabilities at both levels. This term is used throughout the document when the concept applies equally to network elements and their components.

3. Modeling Capabilities and Entitlements

The model describes how to represent capabilities and the entitlements that enable them across inventoried network assets. Capabilities describe what an asset can do. Entitlements indicate whether those capabilities are allowed and under what conditions.

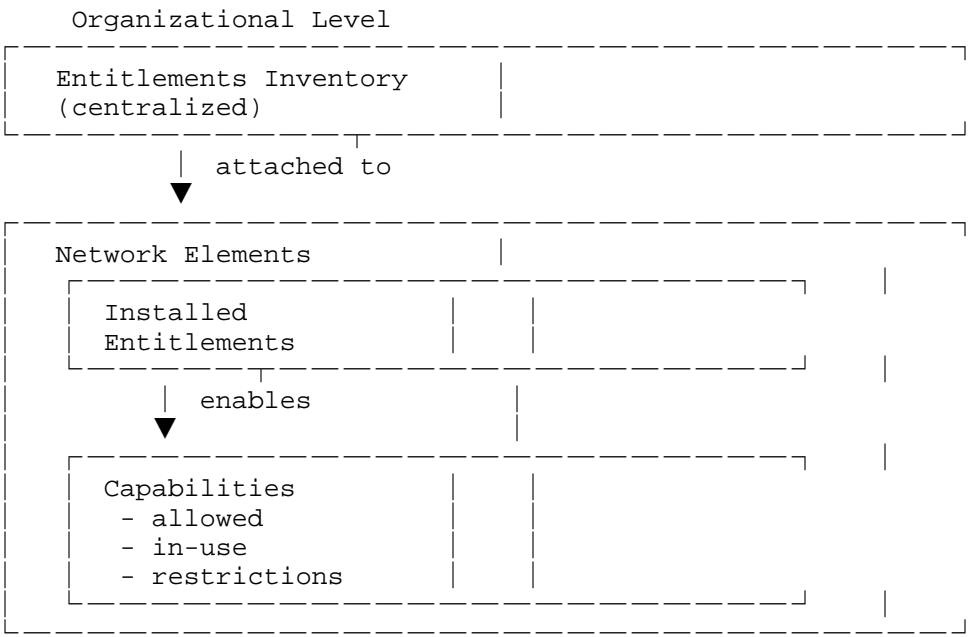


Figure 1: Relationship Between Entitlements and Capabilities

The following subsections describe how the model progressively builds upon the base network inventory to incorporate capabilities, entitlements, and their relationships. The model uses identity-based classes in multiple parts to enable extensibility, allowing implementations to derive custom types that reference external definitions when needed.

3.1. Foundational model: NetworkElement-Entitlements-Capabilities and Restrictions

To represent the complex relationships between network elements, capabilities, and entitlements, a foundational Network Inventory model should be built through a series of extensions. The following diagrams illustrate the progressive complexity of the approach, starting with simple network inventory extensions and culminating in a comprehensive model incorporating capabilities, entitlements, and

restrictions.

3.1.1. Progressive Model Complexity

Figure 2 depicts the initial step, highlighting the base network inventory and the areas to be extended: hardware, software, and entitlements. These extensions are necessary to properly model the relationships.

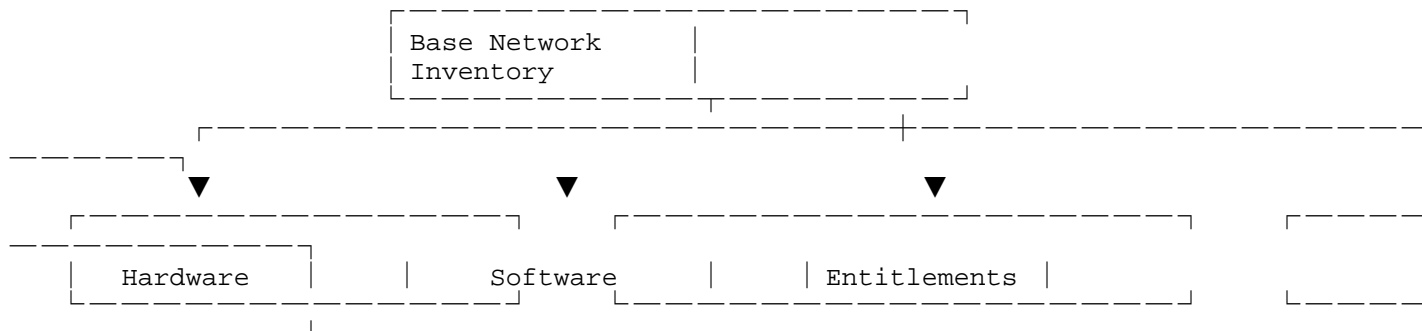


Figure 2: Base Network Inventory Entitlement extension

Figure 3 illustrates the initial relationship between network elements and entitlements, which is two-way: entitlements SHOULD be attached to NEs, and NEs SHOULD have entitlements installed.

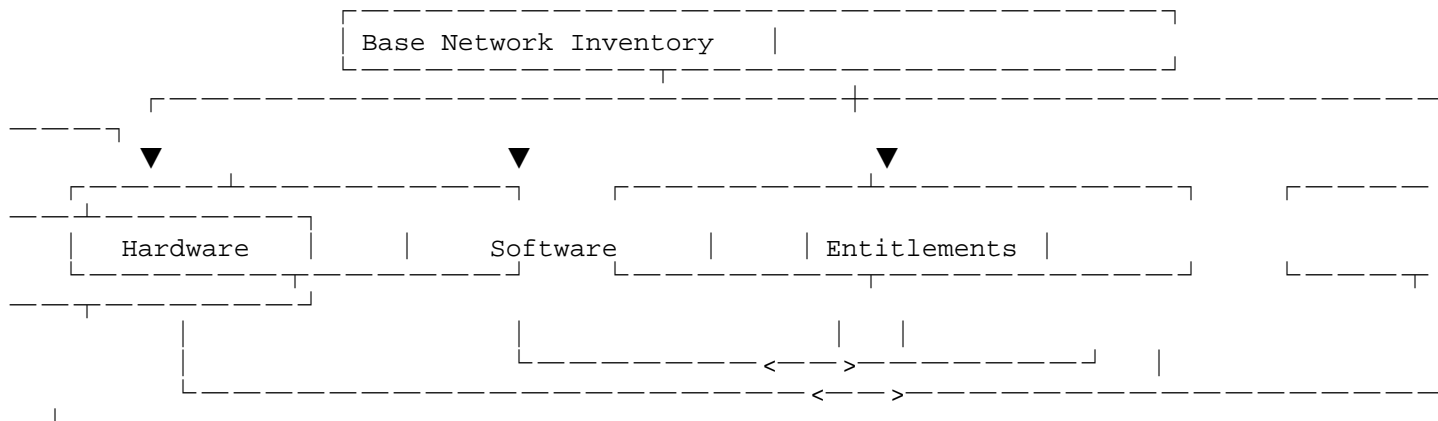


Figure 3: Relationship between entitlements and Base Inventory

Figure 4 depicts NE support capabilities by means of entitlements that authorize their use.

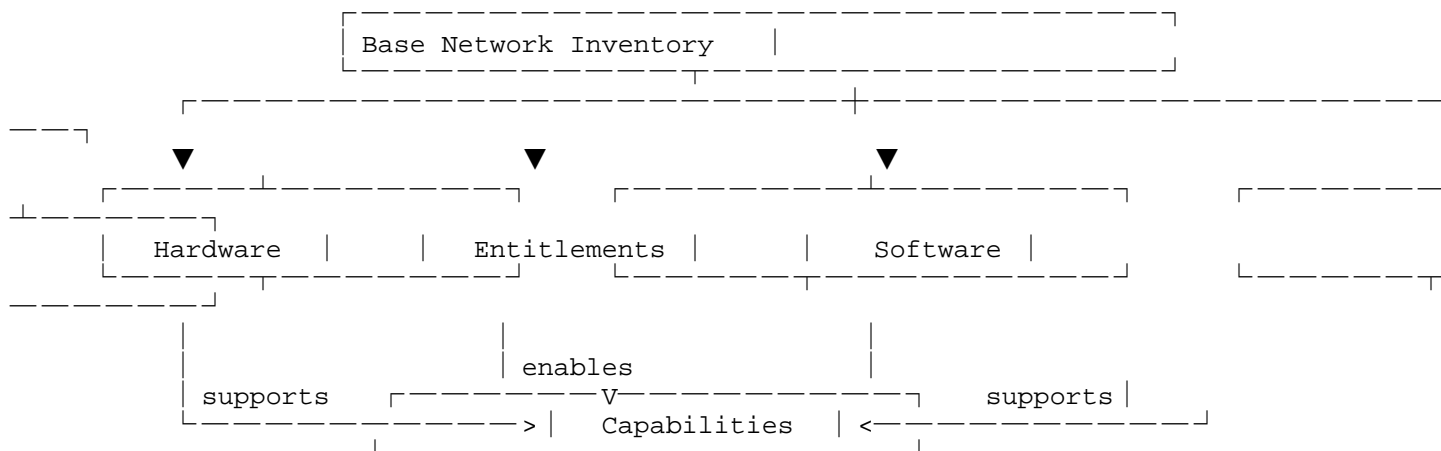


Figure 4: Capabilities integration with the Base Inventory

Finally, NE support capabilities thanks to entitlements that entitle them of their use under certain constraints as shown in Figure 5.

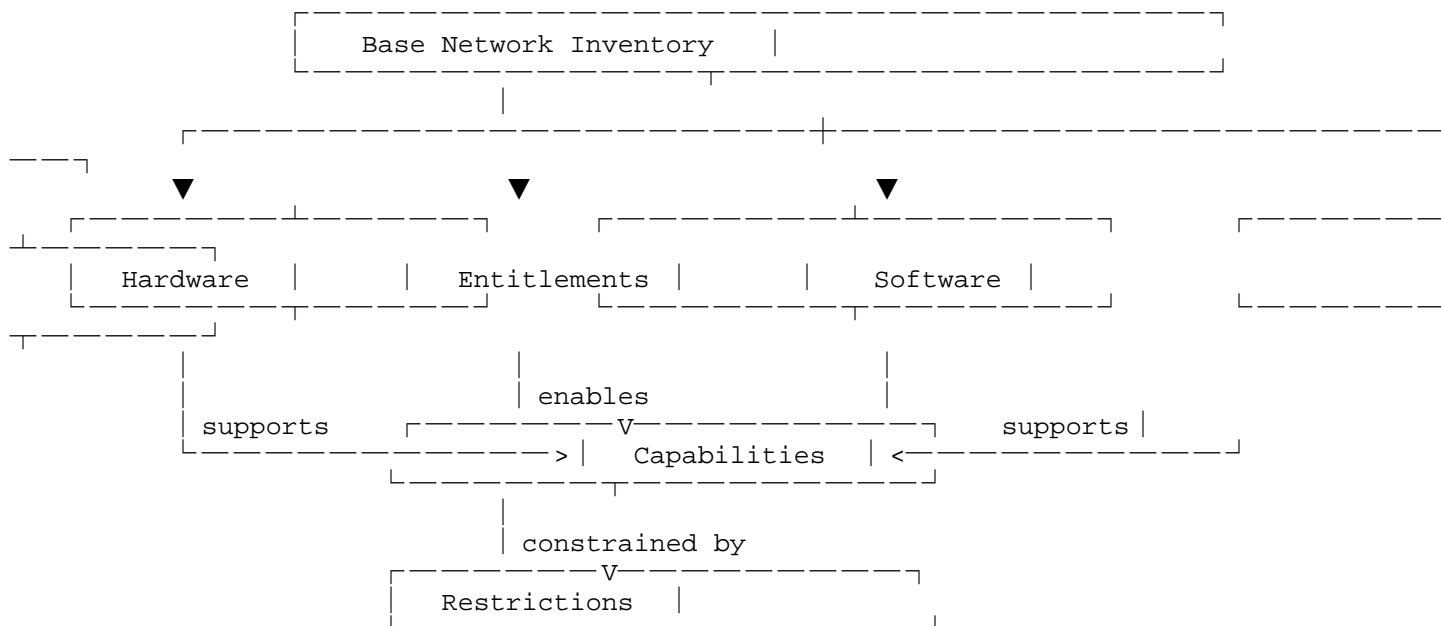


Figure 5: Complete model with restrictions

3.2. Capabilities

Capabilities are modeled by augmenting "network-element" in the "ietf-network-inventory" module in [BaseInventory] according to the following tree:

```

+--ro capabilities!
  +--ro capability-class* [capability-class]
    +--ro capability-class  identityref
    +--ro capability* [capability-id]
      +--ro capability-id      string
      +--ro extended-capability-description?  string
      +--ro entitlement-state!
      | +--ro allowed?  boolean
      | +--ro in-use?   boolean
      +--ro supporting-entitlements!
      | +--ro supporting-entitlement* [entitlement-id]
      | +--ro entitlement-id  -> ../../../../installed-entitlements/entit
lement/entitlement-id
      +--ro capability-restrictions!
        +--ro capability-restriction* [restriction-id]
          +--ro restriction-id  string
          +--ro description?    string
          +--ro resource-name?  string
          +--ro units?          string
          +--ro max-value?      int32
          +--ro current-value?  int32

```

For any given network asset, the capabilities list MAY include all potential capabilities advertised by the vendor, and MUST include those for which the network operator holds a valid entitlement—whether active or not.

This document does not define a complete theory of capabilities or their internal relationships; such work may be addressed elsewhere. Instead, the model provides a flexible framework through the use of identity-based capability classes:

- * ***Basic capability class***: The module defines basic-capability-description as a simple capability class using only identifiers and descriptions. This supports implementations that present capabilities as straightforward lists.
- * ***Extended capability classes***: For structured capability definitions, implementations derive new identities from capability-class. These reference external YANG modules where capabilities have formal structure and semantics. (TBU - See Section X for extension examples.)

This separation ensures that capability definitions can evolve independently of the entitlement inventory model, and that implementations can adopt capability models appropriate to their domain without modifications to this base module.

The granularity at which capabilities are defined is at the discretion of the vendor. A vendor MAY choose to advertise capabilities at a high level of abstraction, such as "Advanced Services", and consumers of this information should refer to vendor documentation to understand what specific functions are included. Alternatively, an implementation MAY enumerate capabilities at a finer granularity, listing individual protocols or features such as MPLS, BGP, or QoS. The model accommodates both approaches.

The capabilities of an inventoried network asset may be restricted based on the availability of proper entitlements. An entitlement manager should be interested in the capabilities available to be used on the network assets, and the capabilities that are currently available. The model includes this information by means of the "supporting entitlements" list, which references installed entitlements and includes potential restrictions related to the status of the entitlement. This allows organizations to monitor entitlement usage and avoid misconfigurations or exceeding permitted capability limits.

3.2.1. Extending Capability Classes

The capability-class identity provides an extension point for integrating external capability models. This module does not define domain-specific capability classes. Instead, extensions derive new capability classes that reference separate models where capabilities are formally defined.

The extension pattern involves two modules:

1. **Capability definition module**: An independent module defining capability concepts with its own structure (lists, containers, attributes). This module has no dependency on the entitlement inventory.
2. **Integration module**: An extension module that derives a new capability-class identity and augments the entitlement inventory to reference the capability definitions from the first module.

This pattern ensures that:

- * Capability models evolve independently of entitlement tracking.
- * Multiple capability domains can coexist (e.g., routing capabilities, security capabilities, QoS capabilities) each with their own defining module.

- * The entitlement inventory remains a thin integration layer rather than a repository of capability definitions.

The following example module defines capability concepts for a specific domain:

```
module example-capability-framework {
  yang-version 1.1;
  namespace "urn:example:capability-framework";
  prefix excap;

  organization
    "Example Organization";
  description
    "Example module defining a list of capabilities.";

  revision 2025-12-05 {
    description
      "Initial version.";
  }

  container capabilities {
    description
      "Container for capability definitions.";

    list capability {
      key "capability-id";
      description
        "List of capability definitions.";

      leaf capability-id {
        type string;
        description
          "Unique identifier for the capability.";
      }

      leaf description {
        type string;
        description
          "Human-readable description of the capability.";
      }
    }
  }
}
```

The following extension module extends the capability-class identity and augments the entitlement inventory to reference the capability definitions from the module above:

```
module example-capability-extension {
  yang-version 1.1;
  namespace "urn:example:capability-extension";
  prefix excapext;

  import ietf-entitlement-inventory {
    prefix ei;
  }
  import ietf-network-inventory {
    prefix inv;
  }
  import example-capability-framework {
    prefix excap;
  }

  organization
    "Example Organization";
  description
    "Example module that extends capability-class and adds
    a reference to capability definitions in another module.";

  revision 2025-12-05 {
    description
      "Initial version.";
  }

  identity example-capability-class {
    base ei:capability-class;
    description
      "Capability class that references the example
      capability framework.";
  }

  augment "/inv:network-inventory/inv:network-elements"
    + "/inv:network-element/ei:capabilities"
    + "/ei:capability-class/ei:capability" {
    when "derived-from-or-self(..ei:capability-class,"
      + "'excapext:example-capability-class')";
    description
      "Adds a reference to capability definitions.";

    leaf capability-ref {
      type leafref {
        path "/excap:capabilities/excap:capability"
          + "/excap:capability-id";
      }
      description
        "Reference to a capability definition in the
```

```
        example-capability-framework module.";
    }
}
```

This pattern allows capability definitions to evolve independently while maintaining a clean integration with the entitlement inventory through the capability-class identity mechanism.

3.3. Entitlements

The entitlement modeling augments "network-inventory" in the ietf-network-inventory module in [BaseInventory] with a top-level entitlements container according to the following tree:

```

+--ro entitlements!
  +--ro entitlement* [entitlement-id]
    +--ro entitlement-id      string
    +--ro product-id?        string
    +--ro sku?                string
    +--ro vendor?            string
    +--ro part-number?       string
    +--ro state?              entitlement-state-t
    +--ro renewal-profile
      | +--ro activation-date?  yang:date-and-time
      | +--ro start-date?      yang:date-and-time
      | +--ro expiration-date? yang:date-and-time
    +--ro restrictions!
      | +--ro restriction* [restriction-id]
      | | +--ro restriction-id  string
      | | +--ro description?    string
      | | +--ro resource-name?  string
      | | +--ro units?          string
      | | +--ro max-value?      int32
      | | +--ro current-value?  int32
    +--ro parent-entitlement-uid? -> ../../entitlement/entitlement-id
    +--ro entitlement-attachment
      +--ro universal-access?  boolean
      +--ro holders
        | +--ro organizations_names
        | | +--ro organizations*  string
        | +--ro users_names
        | | +--ro users*          string
      +--ro assets
        +--ro elements
          | +--ro network-elements* -> /inv:network-inventory/network-elements/netw
ork-element/ne-id
        +--ro components
          +--ro component* [network-element component-id]
            +--ro network-element -> /inv:network-inventory/network-elements/ne
twork-element/ne-id
            +--ro component-id    -> /inv:network-inventory/network-elements/ne
twork-element[inv:ne-id=current()/../network-element]/components/component/component-id

```

Figure 6 depicts the relationship between the Entitlement Inventory model and other models. The Entitlement Inventory model enhances the model defined in the base network inventory model with entitlement-specific attributes and centralized entitlement management capabilities.

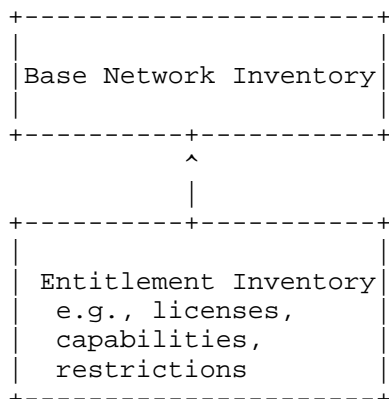


Figure 6: Relationship of Entitlement Inventory Model to Other Inventory Models

Entitlements MUST be listed at the top level, directly under the network-inventory container. This is required because organizations may own entitlements that are not yet assigned to any network asset. Such entitlements exist in a pending state, available for future assignment or installation when the organization decides to allocate them to specific assets.

Entitlements may be listed without explicitly identifying the assets (network elements or components) they apply to. Entitlements are linked to network assets in multiple ways: (1) When entitlements are created for specific assets (i.e., they should only be installed on those), then those assets are specified under the entitlement's attachment section. (2) When an entitlement is installed on a network asset, it appears in the asset's installed-entitlements list. (3) When an installed entitlement enables capabilities, the asset's capabilities will reference the installed entitlement via the supporting-entitlements list.

The base network inventory model includes both network elements and components within them. A network element is an abstraction that typically represents a complete device such as a router or switch. For single-chassis devices, entitlements are typically associated with the network element itself rather than with individual chassis components. However, certain deployment scenarios involve multi-chassis systems, such as stacked switches or optical network elements—where multiple physical units operate as a single logical network element. In these cases, each component may have its own commercial identity (such as a serial number) while the collection behaves as one network element.

Entitlements are typically assigned based on commercial identifiers, often targeting serial numbers. The model supports linking entitlements to both network elements and individual components. However, component-level entitlement tracking is RECOMMENDED only when necessary—specifically when each component has its own set of capability limitations that must be managed independently. Examples include:

- * Individual switches in a stack, where each unit has separate entitlements;
- * Individual chassis in a multi-chassis network element, such as optical equipment; or
- * Pay-as-you-grow routers where line cards have independent entitlement requirements.

In the YANG model, both network elements and components are supported by providing augmentations to each.

Entitlements and network assets are linked in the model in multiple ways. Entitlements at the network-inventory level should be attached to network assets through their attachment mechanism, representing organizational entitlements. Network assets have their own installed-entitlements that may be derived from the centralized entitlements or assigned directly. The capabilities of network assets reference these installed entitlements through their supporting-entitlements lists. The former addresses the case of a centralized license server or inventory system, while the latter represents entitlements that are actively entitling the asset's capabilities. An installed entitlement that is not referenced by any capability means that it is active on the asset but not currently in use.

Entitlements are managed both centrally at the network-inventory level and at the asset level through installed-entitlements. Network assets reference their installed entitlements through their capabilities' supporting-entitlements lists. For instance, a license server or inventory system should list an entitlement at the top level, which then gets installed on specific network assets where the capabilities reference the active entitlement. Each installed entitlement references its centralized entitlement directly via the entitlement-id leafref. For hierarchical or pooled entitlements (e.g., a base license with add-on upgrades), the "parent-entitlement-uid" field in the centralized entitlement catalog links child entitlements to their parent. Proper identification of entitlements is imperative to ensure consistency across systems, enabling monitoring systems to recognize when multiple locations reference related entitlements.

3.3.1. Reverse Mapping from Entitlements to Capabilities

While the model includes links from capabilities to supporting entitlements, some inventory operators may need to evaluate entitlements independently and identify the capabilities they enable.

To support this, implementers may use the "product-id" or "capability-class" metadata along with external references or catalogs. Implementations requiring reverse mapping (identifying capabilities enabled by a specific entitlement) may leverage vendor-specific augmentations or external entitlement catalogs. Standardization of such reverse mappings is outside the scope of this document.

3.4. Entitlement Attachment

The "entitlement" container holds a container called "entitlement-attachment" which relates how the entitlement is operationally linked to holders or network assets. Note that there is a difference between an entitlement being attached to a network asset and an entitlement being installed on the asset. In the former, the license was explicitly associated with one or more assets. Some licenses actually can be open but have a limited number of installations. Other licenses should be openly constrained to a geographic location. We are not dealing with these complex cases now, but the container can be expanded for this in the future.

The model accommodates listing entitlements acquired by the organization but not yet applied or utilized by any actor/asset at the network-inventory level. For these pending entitlements, they can be managed centrally without requiring individual network assets to be aware of their existence.

Some entitlements are inherently associated with a holder, such as organization or a user. For example, a software license may be directly attached to a user. Also, the use of a network device may come with a basic license provided solely to an organization. Some entitlements could be assigned to a more abstract description of holders, such as people under a jurisdiction or a geographical area. The model contains basic information about this, but it can be extended in the future to be more descriptive.

While attachment is optional, the model should be capable of expressing attachment in various scenarios. The model can be expanded to list to which network assets an entitlement is aimed for, when this link is more vague, such as a site license (e.g., network assets located in a specific site), or more open licenses (e.g., free software for all users subscribed to a streaming platform).

The current model does not provide information on whether an entitlement can be reassigned to other network assets. Such scenarios fall under the "what if" category, which is not covered by this model.

3.5. Installed Entitlements

Since capabilities are optional in network assets, the model also provides an augmentation to track entitlements that are installed directly on network assets. This augmentation of "network-element" and "component" in the "ietf-network-inventory" module provides local entitlement storage according to the following tree:

```
+--ro installed-entitlements!
  +--ro entitlement* [entitlement-id]
    +--ro entitlement-id    -> /inv:network-inventory/ei:entitlements/entitlement/entitlement-id
    +--ro in-use?          boolean
```

The installed entitlements represent references to entitlements that are currently active and entitling the network asset. The "entitlement-id" field provides a direct reference to the centralized entitlement at the network-inventory level.

This structure allows network assets to track which entitlements are actively granting them rights, while maintaining the ability to trace relationships to organization-wide entitlement policies.

When entitlements are installed at the component level (e.g., line cards), implementations MAY also list them at the parent network-element level to provide a consolidated view of all entitlements active on the device. Management systems should recognize when an entitlement-id appears at both levels and treat them as the same license instance to avoid double-counting. This point requires further exploration in future instances of this document.

3.6. Implementation Considerations

The model is designed to support partial implementations. Not all systems need to implement every container or feature. The use of presence containers throughout the model allows implementations to signal which parts of the model they support. An implementation that does not populate a presence container indicates that it cannot report that information.

The following progression describes how implementations can adopt the model incrementally, from basic entitlement tracking to full capability and restriction reporting:

3.6.1. Level 1: Centralized Entitlement Inventory

The minimal implementation populates the top-level entitlements container under network-inventory. This provides a centralized catalog of all entitlements owned or managed by the organization, including their identifiers, vendors, states, and validity periods.

At this level, the system answers: What entitlements does the organization have?

3.6.2. Level 2: Installed Entitlements on Assets

Building on Level 1, implementations can populate the installed-entitlements container on network elements and/or components. This tracks which entitlements are currently active and entitling each network asset, by referencing the centralized entitlement catalog.

At this level, the system additionally answers: Which entitlements are actively entitling which assets?

3.6.3. Level 3: Capabilities Reporting

Implementations that can report device capabilities populate the capabilities container on network elements and/or components. This lists what functions each asset can perform, organized by capability class.

At this level, the system additionally answers: What can each asset do?

3.6.4. Level 4: Capability-Entitlement Linkage

Advanced implementations populate the supporting-entitlements container within each capability. This links capabilities to the installed entitlements that enable them, along with the entitlement-state container indicating whether each capability is allowed and in use.

When a capability lists multiple supporting entitlements, the entitlement-state/allowed field MUST reflect the combined effect of all required entitlements. If any required entitlement is missing, expired, or revoked, allowed should be false. The in-use field indicates whether the capability is currently operational.

At this level, the system additionally answers: Which entitlements enable which capabilities? What is allowed and what is in use?

3.6.5. Level 5: Restrictions Reporting

Full implementations populate restriction information at two levels:

- * The restrictions container under each entitlement for global restrictions (e.g., total allowed installations, aggregate usage limits)
- * The capability-restrictions container within each capability for capability-specific limits (e.g., maximum throughput, connection limits)

At this level, the system additionally answers: What constraints apply to entitlements and capabilities? What are the current usage levels?

Implementations SHOULD document which levels they support and any deviations from this progression.

3.7. Model Definition

```
module ietf-entitlement-inventory {  
  yang-version 1.1;  
  namespace "urn:ietf:params:xml:ns:yang:ietf-entitlement-inventory";  
  prefix ei;  
  
  import ietf-yang-types {  
    prefix yang;
```

```
}
import ietf-network-inventory {
  prefix inv;
}

organization
  "IETF IVY Working Group";
contact
  "WG Web:    <https://datatracker.ietf.org/wg/ivy/>
  WG List:    <mailto:inventory-yang@ietf.org>

  Author:     Marisol Palmero

  Author:     Camilo Cardona

  Author:     Diego Lopez

  Author:     Italo Busi
  ";
description
  "A YANG module for Entitlement Inventory, as per
  draft-ietf-ivy-entitlement-inventory-01.

  Copyright (c) 2025 IETF Trust and the persons identified as
  authors of the code. All rights reserved.

  Redistribution and use in source and binary forms, with or
  without modification, is permitted pursuant to, and subject to
  the license terms contained in, the Revised BSD License set
  forth in Section 4.c of the IETF Trust's Legal Provisions
  Relating to IETF Documents
  (https://trustee.ietf.org/license-info).

  This version of this YANG module is part of RFC XXXX
  (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
  for full legal notices.

  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 (RFC 2119) (RFC 8174) when, and only when,
  they appear in all capitals, as shown here.
  ";

revision 2025-10-20 {
  description
    "First full draft version for
    draft-ietf-ivy-entitlement-inventory";
```

```
    reference
      "draft-ietf-ivy-entitlement-inventory-01";
  }

  identity capability-class {
    description
      "Base identity for capability classes.";
  }

  identity basic-capability-description {
    base capability-class;
    description
      "Basic capability class for general capability descriptions.";
  }

  typedef entitlement-state-t {
    type enumeration {
      enum active {
        description
          "Entitlement is active.";
      }
      enum expired {
        description
          "Entitlement is expired.";
      }
      enum pending {
        description
          "Entitlement is pending activation.";
      }
      enum revoked {
        description
          "Entitlement is revoked.";
      }
    }
    description
      "State of the entitlement.";
  }

  grouping restriction-fields {
    description
      "Common fields for describing restrictions or limits.
       Used both for capability-level restrictions and
       entitlement-level global restrictions.";
    leaf description {
      type string;
      description
        "Human-readable description of the restriction.";
    }
  }
```



```
leaf resource-name {
  type string;
  description
    "Optional name of the physical or network resource
    being restricted (e.g., 'bandwidth', 'throughput',
    'storage', 'memory').";
}
leaf units {
  type string;
  description
    "Units for the restriction values (e.g., 'Mbps',
    'connections', 'tunnels').";
}
leaf max-value {
  type int32;
  description
    "Maximum permitted value for this restriction.";
}
leaf current-value {
  type int32;
  description
    "Current usage or consumption of this restricted
    resource at query time.";
}
}

grouping installed-entitlements-group {
  description
    "Grouping for installed entitlements that can be applied to
    network elements or components (generally called asset
    over this document).";
  container installed-entitlements {
    presence
      "The presence of this container means the information system
      that exposes this model knows of the installed entitlements
      of the asset that it populates.
      An empty list of entitlements would then mean
      that no entitlement is installed in this asset.";
    config false;
    description
      "Entitlements currently active and entitling this asset.";
    list entitlement {
      key "entitlement-id";
      description
        "List of entitlements actively entitling this asset.
        Each entitlement references a global listed entitlement.";
      leaf entitlement-id {
        type leafref {
```

```
        path "/inv:network-inventory/ei:entitlements"
          + "/ei:entitlement/ei:entitlement-id";
      }
      description
        "Reference to centralized entitlement.";
    }
    leaf in-use {
      type boolean;
      description
        "Informs whether the entitlement is actively used,
        besides being installed. If existing, and if the
        capabilities list exist and the information system
        supports setting their entitlement-state, this
        information MUST be consistent with it. Meaning, this
        should be in-use if any capability it supports is
        in-use, or false otherwise. The no presence of this
        leaf means that the information system cannot express
        this information";
    }
  }
}

grouping capabilities-group {
  description
    "Grouping for capabilities that can be applied to assets.
    Capabilities represent what the
    asset can do, potentially restricted by entitlements.";
  container capabilities {
    presence
      "The presence of this container means the information system
      that exposes this model is aware of and can report the
      capabilities of this asset
      (i.e. network element or component).
      An empty list of capability classes would mean that the
      element has no capabilities configured or available.";
    config false;
    description
      "Container for capabilities of this asset.";
    list capability-class {
      key "capability-class";
      description
        "List of capability classes supported by this asset. Each
        class groups related capabilities.";
      leaf capability-class {
        type identityref {
          base capability-class;
        }
      }
    }
  }
}
```

```
    description
      "Identifier for the capability class using an identity
      reference.";
  }
  list capability {
    key "capability-id";
    description
      "Individual capability within this class. Represents a
      specific function or feature that the element may
      perform.";
    leaf capability-id {
      type string;
      description
        "Unique identifier for this capability.";
    }
    leaf extended-capability-description {
      type string;
      description
        "Extended capability description.";
    }
  }
  container entitlement-state {
    presence
      "The presence of this container indicates the system
      can report whether this capability is allowed and/or
      in use based on entitlement status.";
    description
      "Reports whether this capability is permitted by
      entitlements and whether it is currently in active
      use.";
    leaf allowed {
      type boolean;
      description
        "Whether the capability is allowed by entitlements.";
    }
    leaf in-use {
      type boolean;
      description
        "Whether the capability is currently in use.";
    }
  }
  container supporting-entitlements {
    presence
      "The presence of this container indicates the system
      can report the entitlement(s) supporting
      the use of this capability by the asset to its
      current allowed state. this container
      should not exist if the system cannot report this.
      An empty list of supporting-entitlement means
```

```
        the capability requires no special
        entitlement to be provided.";
    description
        "List of installed entitlements that
        enable or support this capability.";
    list supporting-entitlement {
        key "entitlement-id";
        description
            "List of installed entitlements
            that enable or support this capability. The
            capability may require one or more
            entitlements to be allowed and in use.";
        leaf entitlement-id {
            type leafref {
                path "../..../..../installed-entitlements"
                + "/entitlement/entitlement-id";
            }
            description
                "Reference to an installed entitlement
                supporting this capability.";
        }
    }
}
container capability-restrictions {
    presence
        "The presence of this container indicates that the
        system can report the current capability restrictions.
        If present, an empty list of
        capability-restriction means the capability
        has no restriction.";
    description
        "Restrictions or limits imposed on this capability by
        entitlements.";
    list capability-restriction {
        key "restriction-id";
        description
            "Restrictions or limits imposed on this capability by
            entitlements.";
        leaf restriction-id {
            type string;
            description
                "Unique identifier for this
                capability restriction.";
        }
        uses restriction-fields;
    }
}
```

```
    }  
  }  
}  
  
augment "/inv:network-inventory/inv:network-elements"  
  + "/inv:network-element" {  
  description  
    "Augments network elements with installed entitlements tracking  
    which entitlements are currently active and entitling the  
    device.";  
  uses installed-entitlements-group;  
}  
  
augment "/inv:network-inventory/inv:network-elements"  
  + "/inv:network-element/inv:components/inv:component" {  
  description  
    "Augments network element components with installed  
    entitlements for component-level tracking.";  
  uses installed-entitlements-group;  
}  
  
augment "/inv:network-inventory/inv:network-elements"  
  + "/inv:network-element" {  
  description  
    "Augments network elements with capabilities information,  
    describing what functions the element can perform and their  
    entitlement status.";  
  uses capabilities-group;  
}  
  
augment "/inv:network-inventory/inv:network-elements"  
  + "/inv:network-element/inv:components/inv:component" {  
  description  
    "Augments network element components with capabilities for  
    component-level feature tracking and entitlement  
    restrictions.";  
  uses capabilities-group;  
}  
  
augment "/inv:network-inventory" {  
  description  
    "Augments the network inventory with a centralized entitlements  
    catalog. This provides organization-wide visibility of all  
    acquired entitlements, their holders, validity periods, and  
    asset associations.";  
  container entitlements {  
    presence  
      "The presence of this container indicates the system
```

```
    maintains and can report the organizational entitlement
    catalog. An empty list means the organization has no
    entitlements defined.";
config false;
description
  "Top-level container for organizational entitlements.";
list entitlement {
  key "entitlement-id";
  description
    "List of entitlements owned or managed by the organization.
    Each entitlement represents a license, right, or
    permission to use specific capabilities, potentially with
    restrictions on scope, time, or usage.";
  leaf entitlement-id {
    type string;
    description
      "Unique entitlement identifier.";
  }
  leaf product-id {
    type string;
    description
      "Product identifier for this entitlement.";
  }
  leaf sku {
    type string;
    description
      "Stock Keeping Unit - vendor's catalog/ordering number
      for this entitlement. Used for procurement and asset
      management integration.";
  }
  leaf vendor {
    type string;
    description
      "Vendor or issuer of this entitlement. Identifies the
      license provider.";
  }
  leaf part-number {
    type string;
    description
      "Manufacturer's part number. May differ from SKU in
      distribution channels.";
  }
  leaf state {
    type entitlement-state-t;
    description
      "Current state of the entitlement.";
  }
  container renewal-profile {
```

```
description
  "Renewal and validity information for the entitlement.";
leaf activation-date {
  type yang:date-and-time;
  description
    "Date when entitlement was activated.";
}
leaf start-date {
  type yang:date-and-time;
  description
    "Start date of entitlement validity.";
}
leaf expiration-date {
  type yang:date-and-time;
  description
    "Expiration date of the entitlement.";
}
}
container restrictions {
  presence
    "The presence of this container means the
    system can provide information of global restrictions
    for this entitlement. An empty list will then
    mean that the entitlement has no global restriction.";
  description
    "Global restrictions imposed by this entitlement.";
  list restriction {
    key "restriction-id";
    description
      "List of restrictions that apply globally to this
      entitlement across all assets and holders. These may
      include usage limits, quotas, or other constraints on
      how the entitlement can be utilized.";
    leaf restriction-id {
      type string;
      description
        "Unique restriction identifier.";
    }
    uses restriction-fields;
  }
}
leaf parent-entitlement-uid {
  type leafref {
    path "../..//entitlement/entitlement-id";
  }
  must '.. != ../entitlement-id' {
    error-message
      "An entitlement cannot reference itself as its
```

```
        parent.";
    }
    description
        "Reference to parent entitlement if this is derived.";
}
container entitlement-attachment {
    description
        "Defines how the entitlement is attached to holders and
        assets.";
    leaf universal-access {
        type boolean;
        description
            "True if entitlement has universal access.";
    }
    container holders {
        description
            "Holders of this entitlement.
            This is for information purposes only, it
            does not apply any restrictions on who can
            use or not the asset where assigned
            to the entitlement.";
        container organizations_names {
            description
                "Organization holders.";
            leaf-list organizations {
                type string;
                description
                    "List of organization names.";
            }
        }
        container users_names {
            description
                "User holders.";
            leaf-list users {
                type string;
                description
                    "List of user names.";
            }
        }
    }
}
container assets {
    description
        "Assets to which this entitlement is attached.";
    container elements {
        description
            "Network elements covered by this entitlement.";
        leaf-list network-elements {
            type leafref {
```



```

        path "/inv:network-inventory"
          + "/inv:network-elements/inv:network-element"
          + "/inv:ne-id";
      }
    description
      "References to network elements covered by this entitlement. When specified, this entitlement applies to the listed network elements.";
  }
}
container components {
  description
    "Individual components covered by this entitlement.";
  list component {
    key "network-element component-id";
    description
      "List of specific components to which this entitlement applies. Allows fine-grained entitlement assignment at the component level rather than entire network elements.";
    leaf network-element {
      type leafref {
        path "/inv:network-inventory"
          + "/inv:network-elements"
          + "/inv:network-element/inv:ne-id";
      }
      description
        "Reference to network element.";
    }
    leaf component-id {
      type leafref {
        path "/inv:network-inventory"
          + "/inv:network-elements"
          + "/inv:network-element"
          + "[inv:ne-id=current()/../network-element]"
          + "/inv:components/"
          + "inv:component/inv:component-id";
      }
      description
        "Reference to component within the specified network element.";
    }
  }
}
}
}
}
}
}
```

```

    }
  }
}

```

3.7.1. Model tree

```
module: ietf-entitlement-inventory
```

```

augment /inv:network-inventory/inv:network-elements/inv:network-element:
  +--ro installed-entitlements!
    +--ro entitlement* [entitlement-id]
      +--ro entitlement-id    -> /inv:network-inventory/ei:entitlements/entitlement/e
ntitlement-id
      +--ro in-use?          boolean
    augment /inv:network-inventory/inv:network-elements/inv:network-element/inv:components/
inv:component:
  +--ro installed-entitlements!
    +--ro entitlement* [entitlement-id]
      +--ro entitlement-id    -> /inv:network-inventory/ei:entitlements/entitlement/e
ntitlement-id
      +--ro in-use?          boolean
    augment /inv:network-inventory/inv:network-elements/inv:network-element:
  +--ro capabilities!
    +--ro capability-class* [capability-class]
      +--ro capability-class  identityref
      +--ro capability* [capability-id]
        +--ro capability-id    string
        +--ro extended-capability-description?  string
        +--ro entitlement-state!
          | +--ro allowed?    boolean
          | +--ro in-use?    boolean
        +--ro supporting-entitlements!
          | +--ro supporting-entitlement* [entitlement-id]
          |   +--ro entitlement-id    -> ../../../../../../../../../../installed-entitlements/en
titlement/entitlement-id
        +--ro capability-restrictions!
          +--ro capability-restriction* [restriction-id]
            +--ro restriction-id    string
            +--ro description?      string
            +--ro resource-name?    string
            +--ro units?            string
            +--ro max-value?        int32
            +--ro current-value?    int32
    augment /inv:network-inventory/inv:network-elements/inv:network-element/inv:components/
inv:component:
  +--ro capabilities!
    +--ro capability-class* [capability-class]
      +--ro capability-class  identityref
      +--ro capability* [capability-id]
        +--ro capability-id    string
        +--ro extended-capability-description?  string
        +--ro entitlement-state!
          | +--ro allowed?    boolean
          | +--ro in-use?    boolean

```

```

    +--ro supporting-entitlements!
    |   +--ro supporting-entitlement* [entitlement-id]
    |   |   +--ro entitlement-id    -> ../../../../installed-entitlements/en
entitlement/entitlement-id
    +--ro capability-restrictions!
    |   +--ro capability-restriction* [restriction-id]
    |   |   +--ro restriction-id    string
    |   |   +--ro description?     string
    |   |   +--ro resource-name?   string
    |   |   +--ro units?          string
    |   |   +--ro max-value?      int32
    |   |   +--ro current-value?  int32
augment /inv:network-inventory:
+--ro entitlements!
+--ro entitlement* [entitlement-id]
+--ro entitlement-id    string
+--ro product-id?      string
+--ro sku?             string
+--ro vendor?          string
+--ro part-number?     string
+--ro state?           entitlement-state-t
+--ro renewal-profile
|   +--ro activation-date? yang:date-and-time
|   +--ro start-date?     yang:date-and-time
|   +--ro expiration-date? yang:date-and-time
+--ro restrictions!
|   +--ro restriction* [restriction-id]
|   |   +--ro restriction-id    string
|   |   +--ro description?     string
|   |   +--ro resource-name?   string
|   |   +--ro units?          string
|   |   +--ro max-value?      int32
|   |   +--ro current-value?  int32
+--ro parent-entitlement-uid? -> ../../entitlement/entitlement-id
+--ro entitlement-attachment
+--ro universal-access?   boolean
+--ro holders
|   +--ro organizations_names
|   |   +--ro organizations*   string
|   +--ro users_names
|   |   +--ro users*          string
+--ro assets
+--ro elements
|   +--ro network-elements* -> /inv:network-inventory/network-elements/n
network-element/ne-id
+--ro components
+--ro component* [network-element component-id]
+--ro network-element    -> /inv:network-inventory/network-elements
/network-element/ne-id
+--ro component-id       -> /inv:network-inventory/network-elements
/network-element[inv:ne-id=current()/../network-element]/components/component/component-i
d

```

4. Implementation Examples and Validation Scenarios

This section provides a progressive, from basic to advanced, series of validated JSON examples demonstrating practical implementation patterns for the entitlement inventory model. The examples are organized from simple to more complex, enabling implementers to:

1. Understand core concepts through minimal working examples.
2. Explore operational scenarios.
3. Identify implementation patterns for common use cases.
4. Validate their own implementations against canonical examples.

Each example: - Addresses specific operational questions - Builds upon concepts introduced in previous examples - Includes contextual explanation of design choices - Provides JSON that validates against the ietf-entitlement-inventory YANG module.

In order to use the examples: - Start with Basic Structure Example to understand fundamental relationships - Progress through examples based on your deployment scenario - Refer to the YANG module trees introduced in the draft, for complete model structure

4.1. Overview of Examples

The following table summarizes the examples provided in this section and the primary concepts each demonstrates:

Example	Title	Complexity	Key Concepts	Operational Question Addressed
1	Basic Structure	Simple	Fundamental relationships, entitlement states	What are the core components of the model?
2	Expired License Handling	Simple	Lifecycle management, state transitions	How does the model handle expired entitlements?
3	Utilization Tracking	Moderate	Restrictions, usage monitoring	What constraints apply and how

				to track usage?
4	Hierarchical Entitlements	Moderate	Parent-child relationships, tiered licensing	How to model license upgrades and dependencies?
5	License Pooling	Advanced	Shared entitlements, multi-device allocation	How to manage pooled licenses across devices?
6	Multi-Vendor Environment	Advanced	Heterogeneous networks, vendor diversity	How to unify entitlements across vendors?
7	Component-Level Entitlements	Advanced	Modular devices, granular licensing	How to track entitlements for device components?
8	Capability Class Extension	Expert	Extensibility, external references	How to integrate custom capability models?

Table 1

Legend: - Simple: Foundational concepts, minimal complexity -
 Moderate: Multi-component scenarios, intermediate concepts -
 Advanced: Complex deployments, advanced patterns - Expert:
 Extensibility and customization

4.2. Basic Structure

4.2.1. Scenario

A network operator has purchased a single routing license for a router. The license enables basic routing capabilities. This represents the simplest possible deployment: one device, one entitlement, one capability.

4.2.2. Operational Context

This example answers the fundamental questions: - What entitlements does the organization own? - Which device is this entitlement installed on? - What capability does this entitlement enable? - Is the capability currently allowed and in-use? This is based on the entitlement-state field.

4.2.3. JSON Example

```
{
  "ietf-network-inventory:network-inventory": {
    "network-elements": {
      "network-element": [
        {
          "ne-id": "router-1",
          "components": {
            "component": [
              {
                "component-id": "chassis-router-1",
                "class": "iana-hardware:chassis"
              }
            ]
          },
        },
      ],
      "ietf-entitlement-inventory:installed-entitlements": {
        "entitlement": [
          {
            "entitlement-id": "ent-1"
          }
        ]
      },
      "ietf-entitlement-inventory:capabilities": {
        "capability-class": [
          {
            "capability-class": "basic-capability-description",
            "capability": [
              {
                "capability-id": "generic-routing-functions",
                "extended-capability-description": "Basic routing capabilities",
                "entitlement-state": {
                  "allowed": true,
                  "in-use": true
                },
                "supporting-entitlements": {
                  "supporting-entitlement": [
                    {
                      "entitlement-id": "ent-1"
                    }
                  ]
                }
              }
            ]
          }
        ]
      }
    }
  }
}
```

```

    ]
  }
}
]
}
]
}
],
"ietf-entitlement-inventory:entitlements": {
  "entitlement": [
    {
      "entitlement-id": "ent-1",
      "product-id": "prod-1",
      "state": "active",
      "renewal-profile": {
        "activation-date": "2025-01-01T00:00:00Z",
        "expiration-date": "2026-01-01T00:00:00Z"
      },
      "entitlement-attachment": {
        "universal-access": false,
        "holders": {
          "organizations_names": {
            "organizations": ["org-1"]
          }
        },
        "assets": {
          "elements": {
            "network-elements": ["router-1"]
          }
        }
      }
    }
  ]
}
}

```

4.3. Expired License Handling

4.3.1. Scenario

The basic structure example showed a healthy state where an active entitlement enables a capability. However, entitlements have lifecycles, they can expire, be revoked, or become inactive. This example demonstrates how the model represents these state transitions and their impact on capabilities.

This example demonstrates how the model handles entitlement lifecycle states. An expired security entitlement results in capabilities being disallowed (allowed: false), while an active routing entitlement keeps its capabilities enabled. The installed-entitlements list shows in-use status reflecting actual capability usage.

4.3.2. Operational Context

Based on the state comparison: Active vs Expired, there is an operational impact with the corresponding risk analysis.

Aspect	Impact	Remediation
Security capabilities	Disabled, features stopped	Renew ent-sec-001 or purchase new license
Routing capabilities	Unaffected, continue operating	Monitor expiration date (2025-06-30)
Device operation	Continues with reduced functionality	Plan renewal before 2025-06-30
Compliance risk	Potential breach if security required	Immediate action if security is mandatory

Table 2

Implementation considerations should consider: - Do not delete the entitlement record (preserve for audit) - Do not immediately remove installed-entitlement (keep for renewal) - Do not affect unrelated entitlements on the same device

4.3.3. JSON Example

```
{
  "ietf-network-inventory:network-inventory": {
    "ietf-entitlement-inventory:entitlements": {
      "entitlement": [
        {
          "entitlement-id": "security-features",
          "product-id": "SEC-ADVANCED-1Y",
          "state": "expired",
          "renewal-profile": {
            "start-date": "2023-10-01T00:00:00Z",
            "activation-date": "2023-10-01T00:00:00Z",

```



```

        "expiration-date": "2024-10-01T00:00:00Z"
    },
    "entitlement-attachment": {
        "universal-access": false,
        "holders": {
            "organizations_names": {
                "organizations": ["org-1"]
            }
        },
        "assets": {
            "elements": {
                "network-elements": ["edge-router-12"]
            }
        }
    }
},
{
    "entitlement-id": "basic-routing-active",
    "product-id": "ROUTING-BASE-3Y",
    "state": "active",
    "renewal-profile": {
        "start-date": "2024-01-01T00:00:00Z",
        "activation-date": "2024-01-01T00:00:00Z",
        "expiration-date": "2027-01-01T00:00:00Z"
    },
    "entitlement-attachment": {
        "universal-access": false,
        "holders": {
            "organizations_names": {
                "organizations": ["org-1"]
            }
        },
        "assets": {
            "elements": {
                "network-elements": ["edge-router-12"]
            }
        }
    }
}
]
},
"network-elements": {
    "network-element": [
        {
            "ne-id": "edge-router-12",
            "components": {
                "component": [
                    {

```

```
        "component-id": "main-chassis",
        "class": "iana-hardware:chassis"
    }
]
},
"ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
        {
            "entitlement-id": "security-features",
            "in-use": false
        },
        {
            "entitlement-id": "basic-routing-active",
            "in-use": true
        }
    ]
},
"ietf-entitlement-inventory:capabilities": {
    "capability-class": [
        {
            "capability-class": "basic-capability-description",
            "capability": [
                {
                    "capability-id": "stateful-firewall",
                    "extended-capability-description": "Stateful firewall",
                    "entitlement-state": {
                        "allowed": false,
                        "in-use": false
                    },
                    "supporting-entitlements": {
                        "supporting-entitlement": [
                            {
                                "entitlement-id": "security-features"
                            }
                        ]
                    },
                    "capability-restrictions": {
                        "capability-restriction": [
                            {
                                "restriction-id": "firewall-sessions",
                                "description": "Maximum concurrent firewall sessions",
                                "resource-name": "sessions",
                                "units": "connections",
                                "max-value": 50000,
                                "current-value": 0
                            }
                        ]
                    }
                }
            ]
        }
    ]
}
```

```

    },
    {
      "capability-id": "ipsec-vpn",
      "extended-capability-description": "IPSec VPN tunnels",
      "entitlement-state": {
        "allowed": false,
        "in-use": false
      },
      "supporting-entitlements": {
        "supporting-entitlement": [
          {
            "entitlement-id": "security-features"
          }
        ]
      },
      "capability-restrictions": {
        "capability-restriction": [
          {
            "restriction-id": "vpn-tunnels",
            "description": "Maximum VPN tunnels",
            "resource-name": "tunnels",
            "units": "tunnels",
            "max-value": 100,
            "current-value": 0
          }
        ]
      }
    }
  ],
  {
    "capability-id": "ospf-routing",
    "extended-capability-description": "OSPF",
    "entitlement-state": {
      "allowed": true,
      "in-use": true
    },
    "supporting-entitlements": {
      "supporting-entitlement": [
        {
          "entitlement-id": "basic-routing-active"
        }
      ]
    },
    "capability-restrictions": {
      "capability-restriction": [
        {
          "restriction-id": "ospf-neighbors",
          "description": "Maximum OSPF neighbor adjacencies, just to give an
example :)",
          "resource-name": "neighbors",

```

```

        "units": "adjacencies",
        "max-value": 50,
        "current-value": 8
      }
    ]
  }
}

```

4.4. Utilization Tracking with Restrictions

4.4.1. Scenario

This example shows comprehensive utilization tracking across multiple capabilities. Each capability includes capability-restrictions with current-value and max-value fields, enabling organizations to monitor resource consumption against licensed limits. This addresses the question: "What constraints apply and what are current usage levels?"

4.4.2. Operational Context

4.4.3. JSON Example

```

{
  "ietf-network-inventory:network-inventory": {
    "network-elements": {
      "network-element": [
        {
          "ne-id": "enterprise-router-5",
          "components": {
            "component": [
              {
                "component-id": "main-chassis",
                "class": "iana-hardware:chassis"
              }
            ]
          }
        }
      ],
      "ietf-entitlement-inventory:installed-entitlements": {
        "entitlement": [
          {

```

```
    "entitlement-id": "security-suite-ent",
    "in-use": true
  },
  {
    "entitlement-id": "advanced-routing-ent",
    "in-use": true
  },
  {
    "entitlement-id": "voice-gateway-ent",
    "in-use": false
  }
]
},
"ietf-entitlement-inventory:capabilities": {
  "capability-class": [
    {
      "capability-class": "basic-capability-description",
      "capability": [
        {
          "capability-id": "firewall",
          "extended-capability-description": "firewall",
          "entitlement-state": {
            "allowed": true,
            "in-use": true
          },
          "supporting-entitlements": {
            "supporting-entitlement": [
              {
                "entitlement-id": "security-suite-ent"
              }
            ]
          },
          "capability-restrictions": {
            "capability-restriction": [
              {
                "restriction-id": "concurrent-sessions",
                "description": "Maximum concurrent firewall sessions",
                "resource-name": "sessions",
                "units": "connections",
                "max-value": 100000,
                "current-value": 45000
              }
            ]
          }
        }
      ]
    },
    {
      "capability-id": "vpn",
      "extended-capability-description": "IPSec VPN tunnels",
```

```

    "entitlement-state": {
      "allowed": true,
      "in-use": true
    },
    "supporting-entitlements": {
      "supporting-entitlement": [
        {
          "entitlement-id": "security-suite-ent"
        }
      ]
    },
    "capability-restrictions": {
      "capability-restriction": [
        {
          "restriction-id": "tunnel-count",
          "description": "Maximum VPN tunnels",
          "resource-name": "tunnels",
          "units": "count",
          "max-value": 500,
          "current-value": 120
        }
      ]
    }
  },
  {
    "capability-id": "bgp-advanced",
    "extended-capability-description": "Advanced BGP features including rou
te reflector",
    "entitlement-state": {
      "allowed": true,
      "in-use": true
    },
    "supporting-entitlements": {
      "supporting-entitlement": [
        {
          "entitlement-id": "advanced-routing-ent"
        }
      ]
    },
    "capability-restrictions": {
      "capability-restriction": [
        {
          "restriction-id": "bgp-peers",
          "description": "Maximum BGP peer sessions",
          "resource-name": "peers",
          "units": "sessions",
          "max-value": 200,
          "current-value": 75
        }
      ]
    }
  }

```

```

    ]
  }
}
]
}
]
}
],
"ietf-entitlement-inventory:entitlements": {
  "entitlement": [
    {
      "entitlement-id": "security-suite-ent",
      "product-id": "SEC-SUITE-ENTERPRISE-001",
      "state": "active",
      "renewal-profile": {
        "start-date": "2024-06-01T00:00:00Z",
        "activation-date": "2024-06-15T00:00:00Z",
        "expiration-date": "2025-06-15T00:00:00Z"
      },
      "entitlement-attachment": {
        "universal-access": false,
        "holders": {
          "organizations_names": {
            "organizations": ["Enterprise Corp"]
          }
        },
        "assets": {
          "elements": {
            "network-elements": ["enterprise-router-5"]
          }
        }
      }
    },
    {
      "entitlement-id": "advanced-routing-ent",
      "product-id": "ROUTING-ADVANCED-001",
      "state": "active",
      "renewal-profile": {
        "start-date": "2024-06-01T00:00:00Z",
        "activation-date": "2024-06-15T00:00:00Z",
        "expiration-date": "2025-06-15T00:00:00Z"
      },
      "entitlement-attachment": {
        "universal-access": false,
        "holders": {
          "organizations_names": {

```

```

        "organizations": ["Enterprise Corp"]
    },
    "assets": {
        "elements": {
            "network-elements": ["enterprise-router-5"]
        }
    }
},
{
    "entitlement-id": "voice-gateway-ent",
    "product-id": "VOICE-GW-PREMIUM-001",
    "state": "active",
    "renewal-profile": {
        "start-date": "2024-12-01T00:00:00Z",
        "activation-date": "2024-12-15T00:00:00Z",
        "expiration-date": "2025-12-15T00:00:00Z"
    },
    "entitlement-attachment": {
        "universal-access": false,
        "holders": {
            "organizations_names": {
                "organizations": ["Enterprise Corp"]
            },
            "users_names": {
                "users": ["telecom-admin"]
            }
        },
        "assets": {
            "elements": {
                "network-elements": ["enterprise-router-5"]
            }
        }
    },
    "restrictions": {
        "restriction": [
            {
                "restriction-id": "voice-channels",
                "description": "Maximum concurrent voice channels",
                "units": "channels",
                "max-value": 100,
                "current-value": 0
            }
        ]
    }
}
]

```



```

    }
  }
}

```

4.5. Hierarchical Entitlements

4.5.1. Scenario

This example demonstrates the parent-entitlement-uid mechanism for modeling entitlement hierarchies. A base "bronze" entitlement provides foundational capabilities, while a "silver" upgrade entitlement (referencing the bronze as parent) adds advanced features. This pattern supports tiered licensing models.

4.5.2. JSON Example

```

{
  "ietf-network-inventory:network-inventory": {
    "ietf-entitlement-inventory:entitlements": {
      "entitlement": [
        {
          "entitlement-id": "bronze-routing-base",
          "product-id": "ROUTER-BRONZE-BASE",
          "state": "active",
          "renewal-profile": {
            "start-date": "2024-01-01T00:00:00Z",
            "activation-date": "2024-01-15T00:00:00Z",
            "expiration-date": "2027-01-15T00:00:00Z"
          },
          "entitlement-attachment": {
            "universal-access": false,
            "holders": {
              "organizations_names": {
                "organizations": [
                  "Enterprise Networks"
                ]
              }
            },
            "assets": {
              "elements": {
                "network-elements": [
                  "branch-router-1",
                  "branch-router-2"
                ]
              }
            }
          }
        }
      ]
    }
  }
}

```

```

    {
      "entitlement-id": "silver-routing-upgrade",
      "product-id": "ROUTER-SILVER-UPGRADE",
      "parent-entitlement-uid": "bronze-routing-base",
      "state": "active",
      "renewal-profile": {
        "start-date": "2025-06-01T00:00:00Z",
        "activation-date": "2025-06-15T00:00:00Z",
        "expiration-date": "2027-01-15T00:00:00Z"
      },
      "entitlement-attachment": {
        "universal-access": false,
        "holders": {
          "organizations_names": {
            "organizations": [
              "Enterprise Networks"
            ]
          }
        },
        "assets": {
          "elements": {
            "network-elements": [
              "branch-router-2"
            ]
          }
        }
      },
    },
  ],
},
"network-elements": {
  "network-element": [
    {
      "ne-id": "branch-router-1",
      "components": {
        "component": [
          {
            "component-id": "main-unit",
            "class": "iana-hardware:chassis"
          }
        ]
      },
    },
  ],
  "ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
      {
        "entitlement-id": "bronze-routing-base",
        "in-use": true
      }
    ]
  }
}

```

```
    ]
  },
  "ietf-entitlement-inventory:capabilities": {
    "capability-class": [
      {
        "capability-class": "basic-capability-description",
        "capability": [
          {
            "capability-id": "ospf-routing",
            "extended-capability-description": "OSPF dynamic routing protocol",
            "entitlement-state": {
              "allowed": true,
              "in-use": true
            },
            "supporting-entitlements": {
              "supporting-entitlement": [
                {
                  "entitlement-id": "bronze-routing-base"
                }
              ]
            },
            "capability-restrictions": {
              "capability-restriction": [
                {
                  "restriction-id": "ospf-areas",
                  "description": "Maximum OSPF areas",
                  "resource-name": "routing-areas",
                  "units": "areas",
                  "max-value": 10,
                  "current-value": 3
                }
              ]
            }
          }
        ]
      }
    ],
    {
      "capability-id": "static-routing",
      "extended-capability-description": "Static route configuration",
      "entitlement-state": {
        "allowed": true,
        "in-use": true
      },
      "supporting-entitlements": {
        "supporting-entitlement": [
          {
            "entitlement-id": "bronze-routing-base"
          }
        ]
      }
    }
  ],
}
```

```

        "capability-restrictions": {
          "capability-restriction": [
            {
              "restriction-id": "static-routes",
              "description": "Maximum static routes",
              "resource-name": "routes",
              "units": "routes",
              "max-value": 500,
              "current-value": 127
            }
          ]
        }
      ],
    },
    {
      "ne-id": "branch-router-2",
      "components": {
        "component": [
          {
            "component-id": "main-unit",
            "class": "iana-hardware:chassis"
          }
        ]
      },
      "ietf-entitlement-inventory:installed-entitlements": {
        "entitlement": [
          {
            "entitlement-id": "bronze-routing-base",
            "in-use": true
          },
          {
            "entitlement-id": "silver-routing-upgrade",
            "in-use": true
          }
        ]
      },
      "ietf-entitlement-inventory:capabilities": {
        "capability-class": [
          {
            "capability-class": "basic-capability-description",
            "capability": [
              {
                "capability-id": "ospf-routing",
                "extended-capability-description": "OSPF dynamic routing protocol",

```

```
"entitlement-state": {
  "allowed": true,
  "in-use": true
},
"supporting-entitlements": {
  "supporting-entitlement": [
    {
      "entitlement-id": "bronze-routing-base"
    }
  ]
},
"capability-restrictions": {
  "capability-restriction": [
    {
      "restriction-id": "ospf-areas",
      "description": "Maximum OSPF areas",
      "resource-name": "routing-areas",
      "units": "areas",
      "max-value": 10,
      "current-value": 5
    }
  ]
}
},
{
  "capability-id": "static-routing",
  "extended-capability-description": "Static route configuration",
  "entitlement-state": {
    "allowed": true,
    "in-use": true
  },
  "supporting-entitlements": {
    "supporting-entitlement": [
      {
        "entitlement-id": "bronze-routing-base"
      }
    ]
  },
  "capability-restrictions": {
    "capability-restriction": [
      {
        "restriction-id": "static-routes",
        "description": "Maximum static routes",
        "resource-name": "routes",
        "units": "routes",
        "max-value": 500,
        "current-value": 89
      }
    ]
  }
}
```

```

    ]
  }
},
{
  "capability-id": "bgp-routing",
  "extended-capability-description": "BGP routing protocol with route p
olicies",
  "entitlement-state": {
    "allowed": true,
    "in-use": true
  },
  "supporting-entitlements": {
    "supporting-entitlement": [
      {
        "entitlement-id": "bronze-routing-base"
      },
      {
        "entitlement-id": "silver-routing-upgrade"
      }
    ]
  },
  "capability-restrictions": {
    "capability-restriction": [
      {
        "restriction-id": "bgp-peers",
        "description": "Maximum BGP peer sessions",
        "resource-name": "bgp-sessions",
        "units": "peers",
        "max-value": 100,
        "current-value": 24
      }
    ]
  }
},
{
  "capability-id": "mpls",
  "extended-capability-description": "MPLS label switching",
  "entitlement-state": {
    "allowed": true,
    "in-use": true
  },
  "supporting-entitlements": {
    "supporting-entitlement": [
      {
        "entitlement-id": "bronze-routing-base"
      },
      {
        "entitlement-id": "silver-routing-upgrade"
      }
    ]
  }
}

```

```

    ]
  },
  "capability-restrictions": {
    "capability-restriction": [
      {
        "restriction-id": "mpls-lsps",
        "description": "Maximum MPLS label-switched paths",
        "resource-name": "lsps",
        "units": "paths",
        "max-value": 200,
        "current-value": 87
      }
    ]
  }
},
{
  "capability-id": "advanced-qos",
  "extended-capability-description": "Advanced QoS with traffic shaping",
  "entitlement-state": {
    "allowed": true,
    "in-use": false
  },
  "supporting-entitlements": {
    "supporting-entitlement": [
      {
        "entitlement-id": "bronze-routing-base"
      },
      {
        "entitlement-id": "silver-routing-upgrade"
      }
    ]
  },
  "capability-restrictions": {
    "capability-restriction": [
      {
        "restriction-id": "qos-classes",
        "description": "Maximum QoS traffic classes",
        "resource-name": "qos-classes",
        "units": "classes",
        "max-value": 16,
        "current-value": 0
      }
    ]
  }
}
]

```

```

    }
  }
]
}
}
}

```

4.6. License Pooling

4.6.1. Scenario

This example shows how shared entitlements can be installed across multiple network elements. A pool-based license is defined once at the network-inventory level with global restrictions (total seats), then installed on multiple routers. Each router's capabilities reference the shared entitlement, and individual capability-restrictions track per-device usage against the pool.

4.6.2. JSON Example

```

{
  "ietf-network-inventory:network-inventory": {
    "network-elements": {
      "network-element": [
        {
          "ne-id": "datacenter-router-1",
          "components": {
            "component": [
              {
                "component-id": "main-chassis",
                "class": "iana-hardware:chassis"
              }
            ]
          },
          "ietf-entitlement-inventory:installed-entitlements": {
            "entitlement": [
              {
                "entitlement-id": "enterprise-license-pool"
              },
              {
                "entitlement-id": "advanced-security-pool"
              }
            ]
          },
          "ietf-entitlement-inventory:capabilities": {
            "capability-class": [
              {
                "capability-class": "basic-capability-description",

```



```
"capability": [
  {
    "capability-id": "enterprise-routing",
    "extended-capability-description": "Enterprise routing protocols",
    "entitlement-state": {
      "allowed": true,
      "in-use": true
    },
    "supporting-entitlements": {
      "supporting-entitlement": [
        {
          "entitlement-id": "enterprise-license-pool"
        }
      ]
    },
    "capability-restrictions": {
      "capability-restriction": [
        {
          "restriction-id": "bgp-peers",
          "description": "Maximum BGP peers",
          "resource-name": "bgp-sessions",
          "units": "peers",
          "max-value": 500,
          "current-value": 245
        }
      ]
    }
  },
  {
    "capability-id": "advanced-firewall",
    "extended-capability-description": "Enterprise firewall",
    "entitlement-state": {
      "allowed": true,
      "in-use": true
    },
    "supporting-entitlements": {
      "supporting-entitlement": [
        {
          "entitlement-id": "advanced-security-pool"
        }
      ]
    },
    "capability-restrictions": {
      "capability-restriction": [
        {
          "restriction-id": "firewall-throughput",
          "description": "Maximum firewall throughput",
          "resource-name": "throughput",
```

```

        "units": "Gbps",
        "max-value": 40,
        "current-value": 28
      }
    ]
  }
}
},
{
  "ne-id": "datacenter-router-2",
  "components": {
    "component": [
      {
        "component-id": "main-chassis",
        "class": "iana-hardware:chassis"
      }
    ]
  },
  "ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
      {
        "entitlement-id": "enterprise-license-pool"
      }
    ]
  },
  "ietf-entitlement-inventory:capabilities": {
    "capability-class": [
      {
        "capability-class": "basic-capability-description",
        "capability": [
          {
            "capability-id": "enterprise-routing",
            "extended-capability-description": "Enterprise routing protocol",
            "entitlement-state": {
              "allowed": true,
              "in-use": true
            },
            "supporting-entitlements": {
              "supporting-entitlement": [
                {
                  "entitlement-id": "enterprise-license-pool"
                }
              ]
            }
          }
        ]
      }
    ]
  },
},

```

```

        "capability-restrictions": {
          "capability-restriction": [
            {
              "restriction-id": "bgp-peers",
              "description": "Maximum BGP peers",
              "resource-name": "bgp-sessions",
              "units": "peers",
              "max-value": 500,
              "current-value": 178
            }
          ]
        }
      ],
    },
  },
  {
    "ne-id": "branch-router-1",
    "components": {
      "component": [
        {
          "component-id": "main-unit",
          "class": "iana-hardware:chassis"
        }
      ]
    },
    "ietf-entitlement-inventory:installed-entitlements": {
      "entitlement": [
        {
          "entitlement-id": "advanced-security-pool"
        }
      ]
    },
    "ietf-entitlement-inventory:capabilities": {
      "capability-class": [
        {
          "capability-class": "basic-capability-description",
          "capability": [
            {
              "capability-id": "advanced-firewall",
              "extended-capability-description": "Enterprise firewall",
              "entitlement-state": {
                "allowed": true,
                "in-use": true
              },
              "supporting-entitlements": {

```

```

        "supporting-entitlement": [
            {
                "entitlement-id": "advanced-security-pool"
            }
        ],
        "capability-restrictions": {
            "capability-restriction": [
                {
                    "restriction-id": "firewall-throughput",
                    "description": "Maximum firewall throughput",
                    "resource-name": "throughput",
                    "units": "Gbps",
                    "max-value": 10,
                    "current-value": 7
                }
            ]
        }
    }
}

},
"ietf-entitlement-inventory:entitlements": {
    "entitlement": [
        {
            "entitlement-id": "enterprise-license-pool",
            "product-id": "ENT-ROUTER-POOL-100",
            "state": "active",
            "renewal-profile": {
                "start-date": "2025-01-01T00:00:00Z",
                "activation-date": "2025-01-15T00:00:00Z",
                "expiration-date": "2026-01-15T00:00:00Z"
            },
            "restrictions": {
                "restriction": [
                    {
                        "restriction-id": "license-consumption",
                        "description": "Enterprise router licenses consumed from pool",
                        "units": "licenses",
                        "max-value": 100,
                        "current-value": 87
                    }
                ]
            }
        }
    ],
},

```

```

    "entitlement-attachment": {
      "universal-access": false,
      "holders": {
        "organizations_names": {
          "organizations": [
            "Company-A"
          ]
        }
      },
      "assets": {
        "elements": {
          "network-elements": [
            "datacenter-router-1",
            "datacenter-router-2"
          ]
        }
      }
    },
    {
      "entitlement-id": "advanced-security-pool",
      "product-id": "SEC-FIREWALL-POOL-25",
      "state": "active",
      "renewal-profile": {
        "start-date": "2025-03-01T00:00:00Z",
        "activation-date": "2025-03-01T00:00:00Z",
        "expiration-date": "2026-03-01T00:00:00Z"
      },
      "restrictions": {
        "restriction": [
          {
            "restriction-id": "license-consumption",
            "description": "Security licenses consumed from pool (high utilization)",
            "units": "licenses",
            "max-value": 25,
            "current-value": 21
          },
          {
            "restriction-id": "total-throughput",
            "description": "Aggregate firewall throughput across all devices (real-time snapshot)",
            "resource-name": "throughput",
            "units": "Gbps",
            "max-value": 100,
            "current-value": 50
          }
        ]
      },
      "entitlement-attachment": {

```

```

    "universal-access": false,
    "holders": {
      "organizations_names": {
        "organizations": [
          "Company-A"
        ]
      }
    },
    "assets": {
      "elements": {
        "network-elements": [
          "datacenter-router-1",
          "branch-router-1"
        ]
      }
    }
  }
}

```

4.7. Multi-Vendor Environment

4.7.1. Scenario

This example illustrates entitlement management in a heterogeneous network with devices from multiple vendors. Each vendor may use different licensing models (consumption-based, perpetual, subscription), but the unified model captures all entitlements consistently. The example shows how organizations gain visibility across their entire multi-vendor infrastructure.

4.7.2. JSON Example

```

{
  "ietf-network-inventory:network-inventory": {
    "network-elements": {
      "network-element": [
        {
          "ne-id": "vendor-a-router-hq-1",
          "components": {
            "component": [
              {
                "component-id": "chassis",
                "class": "iana-hardware:chassis"
              }
            ]
          }
        }
      ]
    }
  }
}

```

```

    ]
  },
  "ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
      {
        "entitlement-id": "vendor-a-sdwan-consumption"
      }
    ]
  },
  "ietf-entitlement-inventory:capabilities": {
    "capability-class": [
      {
        "capability-class": "basic-capability-description",
        "capability": [
          {
            "capability-id": "sd-wan",
            "extended-capability-description": "SD-WAN with consumption-based bil
ling",
            "entitlement-state": {
              "allowed": true,
              "in-use": true
            },
            "supporting-entitlements": {
              "supporting-entitlement": [
                {
                  "entitlement-id": "vendor-a-sdwan-consumption"
                }
              ]
            },
            "capability-restrictions": {
              "capability-restriction": [
                {
                  "restriction-id": "active-tunnels",
                  "description": "Current active SD-WAN tunnels",
                  "resource-name": "tunnels",
                  "units": "count",
                  "max-value": 100,
                  "current-value": 45
                }
              ]
            }
          }
        ]
      }
    ]
  },
  {
    "ne-id": "vendor-b-switch-dc-1",

```

```
"components": {
  "component": [
    {
      "component-id": "main-unit",
      "class": "iana-hardware:chassis"
    }
  ]
},
"ietf-entitlement-inventory:installed-entitlements": {
  "entitlement": [
    {
      "entitlement-id": "vendor-b-datacenter-perpetual"
    },
    {
      "entitlement-id": "vendor-b-support-subscription"
    }
  ]
},
"ietf-entitlement-inventory:capabilities": {
  "capability-class": [
    {
      "capability-class": "basic-capability-description",
      "capability": [
        {
          "capability-id": "evpn-vxlan",
          "extended-capability-description": "EVPN-VXLAN overlay",
          "entitlement-state": {
            "allowed": true,
            "in-use": true
          },
          "supporting-entitlements": {
            "supporting-entitlement": [
              {
                "entitlement-id": "vendor-b-datacenter-perpetual"
              },
              {
                "entitlement-id": "vendor-b-support-subscription"
              }
            ]
          },
          "capability-restrictions": {
            "capability-restriction": [
              {
                "restriction-id": "vxlan-tunnels",
                "description": "Maximum VXLAN tunnel endpoints",
                "resource-name": "vteps",
                "units": "endpoints",
                "max-value": 500,
```



```

        "current-value": 234
      }
    ]
  }
}
],
{
  "ne-id": "vendor-c-switch-dc-2",
  "components": {
    "component": [
      {
        "component-id": "chassis",
        "class": "iana-hardware:chassis"
      }
    ]
  },
  "ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
      {
        "entitlement-id": "vendor-c-telemetry-tier-standard"
      }
    ]
  },
  "ietf-entitlement-inventory:capabilities": {
    "capability-class": [
      {
        "capability-class": "basic-capability-description",
        "capability": [
          {
            "capability-id": "streaming-telemetry",
            "extended-capability-description": "Streaming telemetry tier",
            "entitlement-state": {
              "allowed": true,
              "in-use": true
            },
            "supporting-entitlements": {
              "supporting-entitlement": [
                {
                  "entitlement-id": "vendor-c-telemetry-tier-standard"
                }
              ]
            },
            "capability-restrictions": {
              "capability-restriction": [

```

```

        {
          "restriction-id": "telemetry-streams",
          "description": "Maximum concurrent telemetry streams",
          "resource-name": "streams",
          "units": "streams",
          "max-value": 200,
          "current-value": 87
        }
      ]
    }
  ]
}
],
},
"ietf-entitlement-inventory:entitlements": {
  "entitlement": [
    {
      "entitlement-id": "vendor-a-sdwan-consumption",
      "product-id": "SDWAN-CONSUMPTION-BILLING",
      "sku": "L-SDWAN-CONSUMPTION",
      "vendor": "Vendor-A",
      "part-number": "SDWAN-CONSUMPTION-LIC",
      "state": "active",
      "renewal-profile": {
        "start-date": "2025-01-01T00:00:00Z",
        "activation-date": "2025-01-01T00:00:00Z"
      },
      "restrictions": {
        "restriction": [
          {
            "restriction-id": "monthly-bandwidth-consumed",
            "description": "Total bandwidth consumed this billing period",
            "resource-name": "bandwidth",
            "units": "GB",
            "max-value": 10000,
            "current-value": 7234
          }
        ]
      },
      "entitlement-attachment": {
        "universal-access": false,
        "holders": {
          "organizations_names": {
            "organizations": [

```

```

        "Enterprise IT Dept"
    ]
},
"users_names": {
    "users": [
        "network-admin"
    ]
}
},
"assets": {
    "elements": {
        "network-elements": [
            "vendor-a-router-hq-1"
        ]
    }
}
},
{
    "entitlement-id": "vendor-b-datacenter-perpetual",
    "product-id": "DC-EVPN-VXLAN-PERPETUAL",
    "sku": "S-EVPN-PERM",
    "vendor": "Vendor-B",
    "part-number": "DC-EVPN-PERPETUAL-LIC",
    "state": "active",
    "renewal-profile": {
        "activation-date": "2023-03-15T00:00:00Z"
    },
    "entitlement-attachment": {
        "universal-access": false,
        "holders": {
            "organizations_names": {
                "organizations": [
                    "Enterprise IT Dept"
                ]
            },
            "users_names": {
                "users": [
                    "datacenter-ops"
                ]
            }
        },
        "assets": {
            "elements": {
                "network-elements": [
                    "vendor-b-switch-dc-1"
                ]
            }
        }
    }
}

```

```
    }
  }
},
{
  "entitlement-id": "vendor-b-support-subscription",
  "product-id": "DC-SUPPORT-ANNUAL",
  "sku": "S-SUPPORT-1Y",
  "vendor": "Vendor-B",
  "part-number": "DC-SUPPORT-SUB-1Y",
  "state": "active",
  "renewal-profile": {
    "start-date": "2024-10-01T00:00:00Z",
    "activation-date": "2024-10-01T00:00:00Z",
    "expiration-date": "2025-10-01T00:00:00Z"
  },
  "entitlement-attachment": {
    "universal-access": false,
    "holders": {
      "organizations_names": {
        "organizations": [
          "Enterprise IT Dept"
        ]
      },
      "users_names": {
        "users": [
          "datacenter-ops"
        ]
      }
    },
    "assets": {
      "elements": {
        "network-elements": [
          "vendor-b-switch-dc-1"
        ]
      }
    }
  },
  "entitlement-id": "vendor-c-telemetry-tier-standard",
  "product-id": "TELEMETRY-STD-50DEV-1Y",
  "sku": "TELEM-STD-50-1Y",
  "vendor": "Vendor-C",
  "part-number": "TELEM-STD-TIER-1Y",
  "state": "active",
  "renewal-profile": {
    "start-date": "2025-01-01T00:00:00Z",
    "activation-date": "2025-01-01T00:00:00Z",
```

```

    "expiration-date": "2026-01-01T00:00:00Z"
  },
  "restrictions": {
    "restriction": [
      {
        "restriction-id": "subscribed-device-count",
        "description": "Device count in subscribed tier",
        "units": "devices",
        "max-value": 50,
        "current-value": 50
      },
      {
        "restriction-id": "current-device-count",
        "description": "Actual devices currently managed (may exceed tier for ove
rage billing)",
        "units": "devices",
        "max-value": 150,
        "current-value": 63
      }
    ]
  },
  "entitlement-attachment": {
    "universal-access": false,
    "holders": {
      "organizations_names": {
        "organizations": [
          "Enterprise IT Dept"
        ]
      },
      "users_names": {
        "users": [
          "datacenter-ops",
          "noc-team"
        ]
      }
    }
  },
  "assets": {
    "elements": {
      "network-elements": [
        "vendor-c-switch-dc-2"
      ]
    }
  }
}

```

4.8. Component-Level Entitlements

4.8.1. Scenario

This example demonstrates entitlement tracking at the component level within a modular network element. Individual line cards have their own port licenses, while the chassis has system-level entitlements. This addresses scenarios where different components within the same device have independent entitlement requirements, such as pay-as-you-grow deployments.

4.8.2. JSON Example

```
{
  "ietf-network-inventory:network-inventory": {
    "ietf-entitlement-inventory:entitlements": {
      "entitlement": [
        {
          "entitlement-id": "base-system-license",
          "product-id": "ROUTER-BASE-2025",
          "state": "active",
          "renewal-profile": {
            "activation-date": "2025-01-01T00:00:00Z",
            "start-date": "2025-01-01T00:00:00Z",
            "expiration-date": "2026-01-01T00:00:00Z"
          },
          "entitlement-attachment": {
            "universal-access": false,
            "holders": {
              "organizations_names": {
                "organizations": [
                  "corp-a"
                ]
              }
            }
          },
          "assets": {
            "elements": {
              "network-elements": [
                "modular-router-dc1"
              ]
            }
          }
        },
        {
          "entitlement-id": "advanced-routing-license",
          "product-id": "NET-ADV-ROUTE-100",
          "state": "active",
```

```
"renewal-profile": {
  "activation-date": "2025-01-15T00:00:00Z",
  "start-date": "2025-01-15T00:00:00Z",
  "expiration-date": "2026-01-15T00:00:00Z"
},
"entitlement-attachment": {
  "universal-access": false,
  "holders": {
    "organizations_names": {
      "organizations": [
        "corp-a"
      ]
    }
  },
  "assets": {
    "elements": {
      "network-elements": [
        "modular-router-dc1"
      ]
    }
  }
},
{
  "entitlement-id": "port-license-100g-slot1",
  "product-id": "PORT-LIC-100G-8PORT",
  "state": "active",
  "renewal-profile": {
    "activation-date": "2025-02-01T00:00:00Z",
    "start-date": "2025-02-01T00:00:00Z",
    "expiration-date": "2026-02-01T00:00:00Z"
  },
  "entitlement-attachment": {
    "universal-access": false,
    "holders": {
      "organizations_names": {
        "organizations": [
          "corp-a"
        ]
      },
      "users_names": {
        "users": [
          "admin"
        ]
      }
    },
    "assets": {
      "components": {
```

```

        "component": [
            {
                "network-element": "modular-router-dc1",
                "component-id": "linecard-slot-1"
            }
        ]
    }
}
},
{
    "entitlement-id": "port-license-100g-slot2",
    "product-id": "PORT-LIC-100G-4PORT",
    "state": "active",
    "renewal-profile": {
        "activation-date": "2025-02-15T00:00:00Z",
        "start-date": "2025-02-15T00:00:00Z",
        "expiration-date": "2026-02-15T00:00:00Z"
    },
    "entitlement-attachment": {
        "universal-access": false,
        "holders": {
            "organizations_names": {
                "organizations": [
                    "corp-a"
                ]
            },
            "users_names": {
                "users": [
                    "admin"
                ]
            }
        },
        "assets": {
            "components": {
                "component": [
                    {
                        "network-element": "modular-router-dc1",
                        "component-id": "linecard-slot-2"
                    }
                ]
            }
        }
    },
    {
        "entitlement-id": "crypto-accelerator-license",
        "product-id": "SEC-CRYPTO-ACC",

```



```

    "state": "active",
    "renewal-profile": {
      "activation-date": "2025-03-01T00:00:00Z",
      "start-date": "2025-03-01T00:00:00Z",
      "expiration-date": "2026-03-01T00:00:00Z"
    },
    "entitlement-attachment": {
      "universal-access": false,
      "holders": {
        "organizations_names": {
          "organizations": [
            "corp-a"
          ]
        },
        "users_names": {
          "users": [
            "security-admin"
          ]
        }
      },
      "assets": {
        "components": {
          "component": [
            {
              "network-element": "modular-router-dc1",
              "component-id": "security-module"
            }
          ]
        }
      }
    },
    "network-elements": {
      "network-element": [
        {
          "ne-id": "modular-router-dc1",
          "components": {
            "component": [
              {
                "component-id": "chassis-main",
                "class": "iana-hardware:chassis"
              },
              {
                "component-id": "linecard-slot-1",
                "class": "iana-hardware:module",
                "ietf-entitlement-inventory:installed-entitlements": {

```

```

        "entitlement": [
            {
                "entitlement-id": "port-license-100g-slot1"
            }
        ],
    },
    "ietf-entitlement-inventory:capabilities": {
        "capability-class": [
            {
                "capability-class": "basic-capability-description",
                "capability": [
                    {
                        "capability-id": "high-speed-ports-1-8",
                        "extended-capability-description": "Enable 100G ports 1-8 on li
necard",

                        "entitlement-state": {
                            "allowed": true,
                            "in-use": true
                        },
                        "supporting-entitlements": {
                            "supporting-entitlement": [
                                {
                                    "entitlement-id": "port-license-100g-slot1"
                                }
                            ]
                        },
                        "capability-restrictions": {
                            "capability-restriction": [
                                {
                                    "restriction-id": "port-count",
                                    "description": "Number of active ports",
                                    "resource-name": "ports",
                                    "units": "count",
                                    "max-value": 8,
                                    "current-value": 8
                                }
                            ]
                        }
                    }
                ]
            }
        ]
    },
    {
        "component-id": "linecard-slot-2",
        "class": "iana-hardware:module",
        "ietf-entitlement-inventory:installed-entitlements": {
            "entitlement": [

```

```

        {
          "entitlement-id": "port-license-100g-slot2"
        }
      ]
    },
    "ietf-entitlement-inventory:capabilities": {
      "capability-class": [
        {
          "capability-class": "basic-capability-description",
          "capability": [
            {
              "capability-id": "high-speed-ports-1-4",
              "extended-capability-description": "Enable 100G ports 1-4 on li
necard",
              "entitlement-state": {
                "allowed": true,
                "in-use": true
              },
              "supporting-entitlements": {
                "supporting-entitlement": [
                  {
                    "entitlement-id": "port-license-100g-slot2"
                  }
                ]
              },
              "capability-restrictions": {
                "capability-restriction": [
                  {
                    "restriction-id": "port-count",
                    "description": "Number of active ports",
                    "resource-name": "ports",
                    "units": "count",
                    "max-value": 4,
                    "current-value": 4
                  }
                ]
              }
            }
          ]
        }
      ]
    },
    "component-id": "security-module",
    "class": "iana-hardware:module",
    "ietf-entitlement-inventory:installed-entitlements": {
      "entitlement": [
        {

```

```

        "entitlement-id": "crypto-accelerator-license"
      }
    ]
  },
  "ietf-entitlement-inventory:capabilities": {
    "capability-class": [
      {
        "capability-class": "basic-capability-description",
        "capability": [
          {
            "capability-id": "hardware-encryption",
            "extended-capability-description": "Hardware-accelerated encryp
tion",
            "entitlement-state": {
              "allowed": true,
              "in-use": true
            },
            "supporting-entitlements": {
              "supporting-entitlement": [
                {
                  "entitlement-id": "crypto-accelerator-license"
                }
              ]
            },
            "capability-restrictions": {
              "capability-restriction": [
                {
                  "restriction-id": "crypto-throughput",
                  "description": "Maximum encryption throughput",
                  "resource-name": "throughput",
                  "units": "Gbps",
                  "max-value": 100,
                  "current-value": 65
                }
              ]
            }
          }
        ]
      }
    ]
  },
  "ietf-entitlement-inventory:installed-entitlements": {
    "entitlement": [
      {
        "entitlement-id": "base-system-license"
      }
    ],
  },

```

```

        {
            "entitlement-id": "advanced-routing-license"
        },
        {
            "entitlement-id": "port-license-100g-slot1"
        },
        {
            "entitlement-id": "port-license-100g-slot2"
        },
        {
            "entitlement-id": "crypto-accelerator-license"
        }
    ]
},
"ietf-entitlement-inventory:capabilities": {
    "capability-class": [
        {
            "capability-class": "basic-capability-description",
            "capability": [
                {
                    "capability-id": "routing-protocols",
                    "extended-capability-description": "Advanced routing protocols (BGP,
OSPF, IS-IS)",
                    "entitlement-state": {
                        "allowed": true,
                        "in-use": true
                    },
                    "supporting-entitlements": {
                        "supporting-entitlement": [
                            {
                                "entitlement-id": "advanced-routing-license"
                            }
                        ]
                    },
                    "capability-restrictions": {
                        "capability-restriction": [
                            {
                                "restriction-id": "max-routes",
                                "description": "Maximum routing table entries",
                                "resource-name": "routing-table",
                                "units": "entries",
                                "max-value": 1000000,
                                "current-value": 450000
                            }
                        ]
                    }
                }
            ]
        }
    ]
}

```

```
    ]
  }
}
]
```

4.9. Capability Class Extension

4.9.1. Scenario

This example demonstrates extending the capability-class identity to reference external capability definitions. The example-capability-extension module derives a new capability class and augments the model to reference capabilities defined in a separate module. This pattern allows domain-specific capability models to integrate cleanly with entitlement tracking.

4.9.2. JSON Example

```

{
  "example-capability-framework:capabilities": {
    "capability": [
      {
        "capability-id": "cap-routing-basic",
        "description": "Basic routing functionality"
      },
      {
        "capability-id": "cap-routing-advanced",
        "description": "Advanced routing with BGP and OSPF"
      }
    ]
  },
  "ietf-network-inventory:network-inventory": {
    "network-elements": {
      "network-element": [
        {
          "ne-id": "device-1",
          "ietf-entitlement-inventory:capabilities": {
            "capability-class": [
              {
                "capability-class": "example-capability-extension:example-capability-class",
                "capability": [
                  {
                    "capability-id": "routing",
                    "example-capability-extension:capability-ref": "cap-routing-basic"
                  }
                ]
              }
            ]
          }
        }
      ]
    }
  }
}

```

5. Operational Considerations

5.1. Entitlement Synchronization

When entitlements are managed both centrally and locally, implementations SHOULD provide mechanisms to detect inconsistencies between:

- * Centralized entitlement records
- * Locally installed entitlements

- * Actual capability usage

5.2. Entitlement Expiration Handling

Network elements SHOULD generate notifications when installed entitlements are approaching expiration. The notification timing and handling is implementation-specific but SHOULD provide sufficient lead time for renewal.

5.3. Performance Considerations

Implementations tracking large numbers of entitlements SHOULD consider:

- * Caching strategies for frequently accessed entitlement data
- * Efficient indexing of entitlement-to-capability mappings
- * Minimizing overhead of entitlement validation checks

5.4. Migration and Version Compatibility

When migrating from vendor-specific entitlement systems, implementers should consider mapping strategies that preserve entitlement relationships while adopting this standard model.

6. IANA Considerations

This document registers one URI in the "IETF XML Registry" [RFC3688] and one YANG module in the "YANG Module Names" registry [RFC6020].

6.1. URI Registration

IANA is requested to register the following URI in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

URI: urn:ietf:params:xml:ns:yang:ietf-entitlement-inventory
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

6.2. YANG Module Name Registration

IANA is requested to register the following entry in the "YANG Module Names" registry [RFC6020]:

Name: ietf-entitlement-inventory
Namespace: urn:ietf:params:xml:ns:yang:ietf-entitlement-inventory
Prefix: ei
Maintained by IANA: N
Reference: RFC XXXX

7. Security Considerations

7.1. Entitlement Data Sensitivity

Implementations MUST protect entitlement data with appropriate access controls consistent with organizational security policies.

7.2. Entitlement Tampering

Implementations SHOULD use cryptographic signatures or similar mechanisms to verify entitlement integrity. Network elements SHOULD validate entitlements before activating capabilities.

7.3. Information Disclosure

Access to entitlement inventory data SHOULD be restricted to authorized personnel. Consider implementing role-based access controls that limit visibility based on operational need.

8. References

8.1. 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>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/rfc/rfc3688>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <<https://www.rfc-editor.org/rfc/rfc6020>>.
- [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>>.

8.2. Informative References

[BaseInventory]

Yu, C., Belotti, S., Bouquier, J., Peruzzini, F., and P. Bedard, "A Base YANG Data Model for Network Inventory", Work in Progress, Internet-Draft, draft-ietf-ivy-network-inventory-yang-14, 5 February 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-ivy-network-inventory-yang-14>>.

Acknowledgments

This document is based on work partially funded by the EU Horizon Europe projects ACROSS (grant 101097122), ROBUST-6G (grant 101139068), iTrust6G (grant 101139198), MARE (grant 101191436), and CYBERNEMO (grant 101168182).

Authors' Addresses

Marisol Palmero
Independent
Email: marisol.ietf@gmail.com

Camilo Cardona
NTT
Email: camilo@gin.ntt.net

Diego Lopez
Telefonica
Email: diego.r.lopez@telefonica.com

Italo Busi
Huawei
Email: italo.busi@huawei.com