

OPSAWG
Internet-Draft
Intended status: Standards Track
Expires: 7 August 2026

Q. Ma, Ed.
Q. Wu
Huawei
M. Boucadair, Ed.
Orange
D. King
Lancaster University
3 February 2026

A YANG Data Model and RADIUS Extension for Policy-Based Network Access
Control
draft-ietf-opsawg-ucl-acl-12

Abstract

This document defines a YANG data model for policy-based network access control, which provides enforcement of network access control policies based on group identity. Additionally, the YANG data model defined in the document also extends ACLs (Access Control Lists) with date and time parameters to support schedule-aware policy enforcement.

Specifically in scenarios where network access is triggered by user authentication, this document defines a mechanism to ease the maintenance of the mapping between a user group identifier and a set of IP/MAC addresses to enforce policy-based network access control. Moreover, the document defines a Remote Authentication Dial-in User Service (RADIUS) attribute that is used to communicate the user group identifier as part of identification and authorization information.

Discussion Venues

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

Discussion of this document takes place on the Operations and Management Area Working Group Working Group mailing list (opsawg@ietf.org), which is archived at <https://mailarchive.ietf.org/arch/browse/opsawg/>.

Source for this draft and an issue tracker can be found at <https://github.com/boucadair/policy-based-network-acl>.

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 7 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	3
1.1. Editorial Note (To be removed by RFC Editor)	4
2. Conventions and Definitions	5
3. Sample Usage	6
4. Policy-based Network Access Control	6
4.1. Overview	7
4.2. Endpoint Group	9
4.2.1. User Group	10
4.2.2. Device Group	11
4.2.3. Application Group	11
4.3. Relations Between Different Endpoint Groups	12
5. The UCL Extension to the ACL Module	12
5.1. Module Overview	12
5.2. The "ietf-ucl-acl" YANG Module	14
6. User Access Control Group ID RADIUS Attribute	22
7. RADIUS Attributes	23
8. Operational Considerations	24
8.1. Deployment Options	24
8.2. Hardware/Software Implications	24
8.3. Mapping Consistency	24

9. Security Considerations	25
9.1. YANG	25
9.2. RADIUS	26
10. IANA Considerations	26
10.1. YANG	26
10.2. RADIUS	26
11. References	27
11.1. Normative References	27
11.2. Informative References	28
Appendix A. Examples Usage	31
A.1. Configuring the Controller Using Group based ACL	31
A.2. Configuring a PEP Using Group-based ACL	33
A.3. Configuring PEPs Using Address-based ACLs	37
Acknowledgments	41
Authors' Addresses	41

1. Introduction

With the increased adoption of remote access technologies (e.g., Virtual Private Networks (VPNs) and Bring Your Own Device (BYOD) policies), enterprises adopted more flexibility related to how, where, and when employees work and collaborate. However, more flexibility comes with increased risks. Enabling office flexibility (e.g., mobility across many access locations) introduces a set of challenges for large-scale enterprises compared to conventional network access management approaches. Examples of such challenges are listed below:

- * Endpoints do not have stable IP addresses. For example, Wireless LAN (WLAN) and VPN clients, as well as back-end Virtual Machine (VM)-based servers, can move; their IP addresses could change as a result. This means that relying on IP/transport fields (e.g., the 5-tuple) is inadequate to ensure consistent and efficient security policy enforcement. IP address-based policies may not be flexible enough to accommodate endpoints with volatile IP addresses.
- * With the massive adoption of teleworking, there is a need to apply different security policies to the same set of endpoints under different circumstances (e.g., prevent relay attacks against a local attachment point to the enterprise network). For example, network access might be granted based upon criteria such as users' access location, source network reputation, users' role, time-of-day, type of network device used (e.g., corporate issued device versus personal device), device's security posture, etc. This means that the network needs to recognize the endpoints' identity and their current context, and map the endpoints to their correct access grant to the network.

This document defines a YANG data model (Section 5.2) for policy-based network access control, which extends the IETF Access Control Lists (ACLs) module defined in [RFC8519]. This module can be used to ensure consistent enforcement of ACL policies based on the group identity. Additionally, the YANG data model defined in the document also extends ACLs with date and time parameters to support schedule-aware policy enforcement.

The ACL concept has been generalized to be device-nonspecific, and can be defined at network/administrative domain level [I-D.ietf-netmod-acl-extensions]. To allow for all ACL applications, the YANG module for policy-based network ACL defined in Section 5.2 does not limit how it can be used.

Specifically in scenarios where network access is triggered by user authentication, this document also defines a mechanism to establish a mapping between (1) the user group identifier (ID) and (2) common IP packet header fields and other encapsulating packet data (e.g., MAC address) to execute the policy-based access control. Additionally, the document defines a Remote Authentication Dial-in User Service (RADIUS) [RFC2865] attribute that is used to communicate the user group identifier as part of identification and authorization information (Section 6).

Although the document cites MAC addresses as an example in some sections, the document does not make assumptions about which identifiers are used to trigger ACLs. These examples should not be considered as recommendations. Readers should be aware that MAC-based ACLs can be bypassed by flushing out the MAC address. Other implications related to the change of MAC addresses are discussed in [RFC9797].

The document does not specify how to map the policy group identifiers to dedicated fields, Group-Based Policy (GBP) discussed in Section 6.2.3 of [RFC9638] provides an example of how that may be achieved.

1.1. Editorial Note (To be removed by RFC Editor)

Note to the RFC Editor: This section is to be removed prior to publication.

This document contains placeholder values that need to be replaced with finalized values at the time of publication. This note summarizes all of the substitutions that are needed. No other RFC Editor instructions are specified elsewhere in this document.

Please apply the following replacements:

- * XXXX --> the assigned RFC number for this document
- * SSSS --> the assigned RFC number for
[I-D.ietf-netmod-schedule-yang]
- * 2026-01-12 --> the actual date of the publication of this document

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.

The meanings of the symbols in tree diagrams are defined in [RFC8340].

The document uses the following term defined in [RFC3198]:

- * policy

The document uses the following terms defined in [RFC8519]:

- * Access Control Entry (ACE)
- * Access Control List (ACL)

The following definitions are used throughout this document:

- * Endpoint: Refers to an entity which could be an end-user, host device, or application that actually connects to a network.
- * Endpoint group: Refers to a group of endpoints that share common access control policies.
- * User group: A group of end-users who will be assigned the same network access policy. An end-user is defined as a person. Refer to Section 4.2.1 for more details.
- * device group: A collection of host devices that share a common access control policies. A host device provides compute, memory, storage, and networking capabilities and connects to a network. Host devices may be servers, Internet of Things (IoT) devices, etc. Refer to Section 4.2.2 for more details.
- * Application group: A collection of applications that share a

common access control policies. An application is a software program used for a specific service. Refer to Section 4.2.3 for more details.

- * Endpoint group identifier: An identifier used to represent the collective identity of an endpoint group. An endpoint group may include a user group, device group, or application group.
- * User group based Control List (UCL) data model: A YANG data model for policy-based network access control that specifies an extension to the "ietf-access-control-list" module [RFC8519]. It allows policy enforcement based on a group identifier, which can be used both at the network device level and at the network/administrative domain level.

3. Sample Usage

Access to some networks (e.g., enterprise networks) requires recognizing the endpoints' identities no matter how, where, and when they connect to the network resources. Then, the network maps the (connecting) endpoints to their access authorization rights. Such rights are defined using local policies. As discussed in Section 1, because (1) there is a large number of connecting endpoints and (2) an endpoint may have different source IP addresses in different network segments, deploying a network access control policy for each IP address or network segment requires a high overhead. An alternate approach is to configure endpoint groups to classify users, enterprise devices and applications and associate ACLs with endpoint groups so that endpoints in each group can share a group of ACL rules. This approach greatly reduces the overhead of the administrators and optimizes the ACL resources.

The network ACLs can be provisioned on devices using specific mechanisms, such as [RFC8519] or [I-D.ietf-netmod-acl-extensions].

Different policies may need to be applied in different contextual situations. For example, companies may restrict (or grant) employees access to specific internal or external resources during work hours, while another policy is adopted during off-hours and weekends. A network administrator may also require traffic-shaping (Section 2.3.3.3 of [RFC2475]) and policing (Section 2.3.3.4 of [RFC2475]) during peak hours in order to not affect other data services.

4. Policy-based Network Access Control

4.1. Overview

An example architecture of a system that provides real-time and consistent enforcement of access control policies is shown in Figure 1. This architecture illustrates a user-centric flow, which includes the following functional entities and interfaces:

- * A service orchestrator which coordinates the overall service, including security policies. The service may be connectivity or any other access to resources that can be hosted and offered by a network.
- * A Software-Defined Networking (SDN) [RFC7149] [RFC7426] controller which is responsible for maintaining endpoint-group based ACLs and mapping the endpoint group to the associated attributes information (e.g., IP/MAC address). An SDN controller also behaves as a Policy Decision Point (PDP) [RFC3198] and pushes the required access control policies to relevant Policy Enforcement Points (PEPs) [RFC3198]. A PDP is also known as "policy server" [RFC2753].

An SDN controller may interact with an Authentication, Authorization, and Accounting (AAA) [RFC3539] server or a Network Access Server (NAS) [RFC7542].

- * A Network Access Server (NAS) entity which handles authentication requests. The NAS interacts with an AAA server to complete user authentication using protocols like RADIUS [RFC2865]. When access is granted, the AAA server provides the group identifier (group ID) to which the user belongs when the user first logs onto the network. A new RADIUS attribute is defined in Section 6 for this purpose.
- * The AAA server provides a collection of authentication, authorization, and accounting functions. The AAA server is responsible for centralized user information management. The AAA server is preconfigured with user credentials (e.g., username and password), possible group identities and related user attributes (users may be divided into different groups based on different user attributes).
- * The Policy Enforcement Point (PEP) is the central entity which is responsible for enforcing appropriate access control policies. A first deployment scenario assumes that the SDN controller maps the group ID to the related common packet header and delivers IP/MAC address based ACL policies to the required PEPs. Another deployment scenario may require that PEPs map incoming packets to their associated source and/or destination endpoint-group IDs, and

acts upon the endpoint-group ID-based ACL policies (e.g., a group identifier may be carried in packet headers such as discussed in Section 6.2.3 of [RFC9638]).

Multiple PEPs may be involved in a network.

A PEP exposes a YANG-based interface (e.g., NETCONF [RFC6241]) to an SDN controller.

Figure 1 provides the overall architecture and procedure for policy-based access control management.

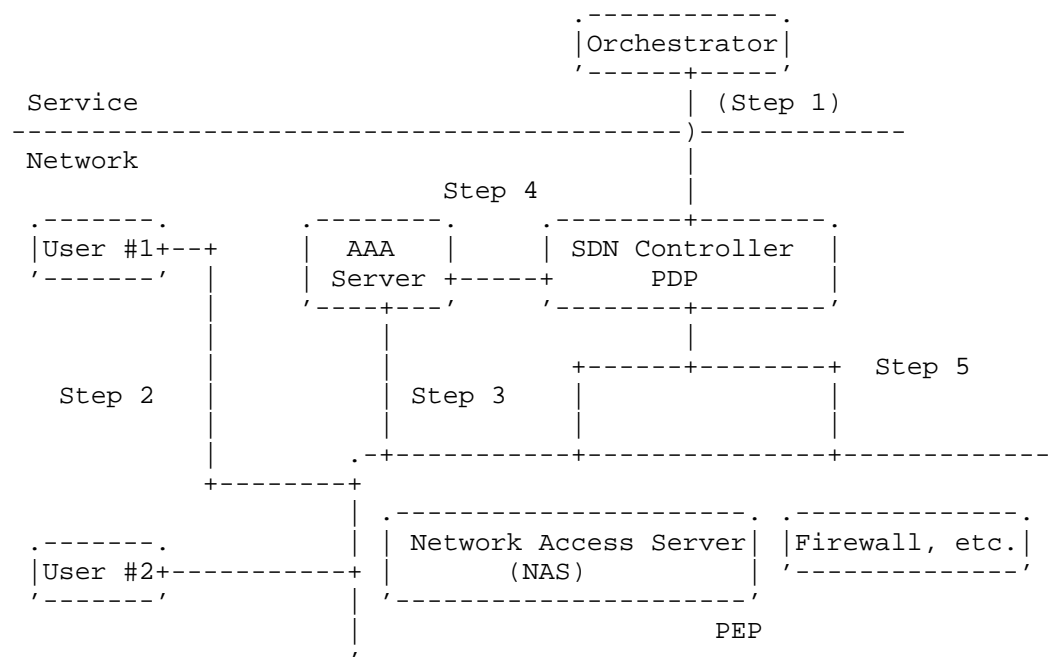


Figure 1: An Example Architecture for User Group-based Policy Management

In reference to Figure 1, the following typical flow is experienced:

Step 1: Administrators (or a service orchestrator) configure an SDN controller with network-level ACLs using the YANG module defined in Section 5.2. An example is provided in Appendix A.1.

Step 2: When a user first logs onto the network, they are required to be authenticated (e.g., using username and password) at the NAS.

Step 3: The authentication request is then relayed to the AAA server using a protocol such as RADIUS [RFC2865]. It is assumed that the AAA server has been appropriately configured to store user credentials, e.g., username, password, group information, and other user attributes. This document does not restrict what authentication method is used. Administrators may refer to, e.g., Section 7.4 of [I-D.ietf-radext-deprecating-radius] for authentication method recommendations. If the authentication request succeeds, the user is placed in a user group with the identifier returned to the NAS as the authentication result (see Section 6). If the authentication fails, the user is not assigned any user group, which also means that the user has no access (i.e., Access-Reject returned); or the user is assigned a special group with very limited access permissions for the network (as a function of the local policy). ACLs are enforced so that flows from that IP address are discarded (or rate-limited) by the network. In some implementations, the AAA server can be integrated with an SDN controller.

Step 4: Either the AAA server or the NAS notifies an SDN controller of the mapping between the user group ID and related common packet header attributes (e.g., IP/MAC address). The exact details of how such notification is performed are out of scope of this specification.

Step 5: Either group or IP/MAC address-based access control policies are maintained on relevant PEPs under the SDN controller's management. Whether the PEP enforces the group or IP/MAC address-based ACL is implementation specific. Both types of ACL policy may exist on the PEP. Appendix A.2 and Appendix A.3 elaborate on each case.

A similar flow applies to policy management based on other endpoint group types, such as device or application groups, except that the mapping between the group ID and related common packet header attributes (e.g., IP/MAC address) may be maintained on the SDN controller based on an inventory or an application registry. Particularly, the use of RADIUS exchanges is not required in such cases (Section 6).

Section 8 provides additional operational considerations.

4.2. Endpoint Group

4.2.1. User Group

The user group is determined by a set of predefined policy criteria (e.g., source IP address, geolocation data, time of day, or device certificate). It uses an identifier (user group ID) to represent the collective identity of a group of users. Users may be moved to different user groups if their composite attributes, environment, and/or local enterprise policy change.

A user is authenticated, and classified at the AAA server, and assigned to a user group. A user's group membership may change as aspects of the user change. For example, if the user group membership is determined solely by the source IP address, then a given user's group ID will change when the user is assigned a new IP address that falls outside of the range of addresses of the previous user group.

This document does not make any assumption about how user groups are defined. Such considerations are deployment specific and are out of scope. However, and for illustration purposes, Table 1 shows an example of how user group definitions may be characterized. User groups may share several common criteria. That is, user group criteria are not mutually exclusive. For example, the policy criteria of user groups R&D Regular and R&D BYOD may share the same set of users that belong to the R&D organization, and differ only in the type of clients (firm-issued clients vs. users' personal clients). Likewise, the same user may be assigned to different user groups depending on the time of day or the type of day (e.g., weekdays versus weekends), etc.

Group Name	Group ID	Group Description
R&D	foo-10	R&D employees
R&D BYOD	foo-11	Personal devices of R&D employees
Sales	foo-20	Sales employees
VIP	foo-30	VIP employees

Table 1: User Group Example

4.2.2. Device Group

The device group ID is an identifier that represents the collective identity of a group of enterprise end-devices. An enterprise device could be a server that hosts applications or software that deliver services to enterprise users. It could also be an enterprise IoT device that serves a limited purpose, e.g., a printer that allows users to scan and print. Table 2 shows an example of how device group definitions may be characterized.

Group Name	Group ID	Group Description
Workflow	bar-40	Workflow resource servers
R&D Resource	bar-50	R&D resource servers
Printer Resource	bar-60	Printer resources

Table 2: Device Group Example

Users accessing an enterprise device should be strictly controlled. Matching abstract device group ID instead of specified addresses in ACL policies helps shield the consequences of address change (e.g., back-end VM-based server migration).

4.2.3. Application Group

An application group is a collection of applications that share a common access control policies. A device may run multiple applications, and different policies might need to be applied to the applications and device. A single application may need to run on multiple devices/VMs/containers, the abstraction of application group eases the process of application migration. For example, the policy does not depend on the transport coordinates (i.e., 5-tuple). Table 3 shows an example of how application group definitions may be characterized.

Group Name	Group ID	Group Description
Audio/Video Streaming	baz-70	Audio/Video conferencing application
Instant Messaging	baz-80	Messaging application
document Collaboration	baz-90	Real-time document editing application

Table 3: Application Group Example

4.3. Relations Between Different Endpoint Groups

Policies enforcement can be targeted to different endpoint groups in different scenarios. For example, when a user connects to the network and accesses an application hosted on one or multiple devices, access policies may be applied to different user groups. In some cases, applications and devices may operate and run without requiring any user interventions, or they may require user authentication but access rules do not differentiate between different users. This enables policies to be applied to the application or device group. A device group can be used when there is only one single application running on the device or different applications running but with the same access control rules. If there is an application running on different devices/VMs/containers, it is simpler to apply a single policy to the application group.

5. The UCL Extension to the ACL Module

5.1. Module Overview

This module specifies an extension to the "ietf-access-control-list" module [RFC8519]. This extension adds endpoint groups so that an endpoint group identifier can be matched upon, and also enable access control policy activation based on date and time conditions.

Figure 2 provides the tree structure of the "ietf-ucl-acl" module.

```

module: ietf-ucl-acl

augment /acl:acls:
  +--rw endpoint-groups {ucl:group}?
    +--rw endpoint-group* [group-id]
      +--rw group-id      string
      +--rw group-type?   identityref

```

```

augment /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches:
  +--rw endpoint-group {ucl:match-on-group}?
    +--rw source-group-id?      group-id-reference
    +--rw destination-group-id? group-id-reference
augment /acl:acls/acl:acl/acl:aces/acl:ace:
  +--rw effective-schedule {ucl:schedule}?
    +--rw (schedule-type)?
      +--:(period)
        +--rw period
          +--rw period-description?      string
          +--rw period-start              yang:date-and-time
          +--rw time-zone-identifier?     sys:timezone-name
          +--rw (period-type)?
            +--:(explicit)
              | +--rw period-end?          yang:date-and-time
            +--:(duration)
              | +--rw duration?            duration
      +--:(recurrence)
        +--rw recurrence {schedule:icalendar-recurrence}?
          +--rw recurrence-first
            | +--rw start-time?            yang:date-and-time
            | +--rw duration?              duration
            | +--rw time-zone-identifier?  sys:timezone-name
          +--rw (recurrence-end)?
            +--:(until)
              | +--rw until?              yang:date-and-time
            +--:(count)
              | +--rw count?              uint32
          +--rw recurrence-description?    string
          +--rw frequency                  identityref
          +--rw interval?                  uint32
          +--rw period* [period-start]
            +--rw period-description?      string
            +--rw period-start              yang:date-and-time
            +--rw time-zone-identifier?     sys:timezone-name
            +--rw (period-type)?
              +--:(explicit)
                | +--rw period-end?        yang:date-and-time
              +--:(duration)
                | +--rw duration?          duration
          +--rw bysecond*                  uint32
          +--rw byminute*                  uint32
          +--rw byhour*                    uint32
          +--rw byday* [weekday]
            | +--rw direction*             int32
            | +--rw weekday                schedule:weekday
          +--rw bymonthday*                int32
          +--rw byyearday*                 int32

```

```

+--rw byyearweek*          int32
+--rw byyearmonth*         uint32
+--rw bysetpos*            int32
+--rw workweek-start?      schedule:weekday
+--rw exception-dates*     yang:date-and-time
```

Figure 2: UCL Extension

The first part of the "ietf-ucl-acl" module augments the "acl" list in the "ietf-access-control-list" module [RFC8519] with an "endpoint-groups" container having a list of "endpoint group" inside, each entry has a "group-id" that uniquely identifies the endpoint group and a "group-type" parameter to specify the endpoint group type.

"group-id" is defined as a string rather than unsigned integer (e.g., uint32) to accommodate deployments which require some identification hierarchy within a domain. Such a hierarchy is meant to ease coordination within an administrative domain. There might be cases where a domain needs to tag packets with the group they belong to. The tagging does not need to mirror exactly the "group id" used to populate the policy. How the "group-id" string is mapped to the tagging or field in the packet header in encapsulation scenario is outside the scope of this document. Augmentation may be considered in the future to cover encapsulation considerations.

The second part of the "ietf-ucl-acl" module augments the "matches" container in the "ietf-access-control-list" module [RFC8519] so that a source and/or destination endpoint group index can be referenced as the match criteria.

The third part of the module augments the "ace" list in the "ietf-access-control-list" module [RFC8519] with date and time specific parameters to allow ACE to be activated based on a date/time condition. Two types of time range are defined, which reuse "recurrence" and "period" groupings defined in the "ietf-schedule" YANG module in [I-D.ietf-netmod-schedule-yang], respectively.

5.2. The "ietf-ucl-acl" YANG Module

This module imports types and groupings defined in the "ietf-schedule" module [I-D.ietf-netmod-schedule-yang]. It also augments the "ietf-access-control-list" module (Section 4.1 of [RFC8519]).

```
<CODE BEGINS> file "ietf-ucl-acl@2026-01-12.yang"
module ietf-ucl-acl {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-ucl-acl";
  prefix ucl;

  import ietf-access-control-list {
    prefix acl;
    reference
      "RFC 8519: YANG Data Model for Network Access
        Control Lists (ACLs)";
  }
  import ietf-schedule {
    prefix schedule;
    reference
      "RFC SSSS: A Common YANG Data Model for Scheduling";
  }

  organization
    "IETF OPSWG Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
    WG List: <mailto:opsawg@ietf.org>

    Editor:   Qiufang Ma
              <mailto:maqiufang1@huawei.com>
    Author:   Qin Wu
              <mailto:bill.wu@huawei.com>
    Editor:   Mohamed Boucadair
              <mailto:mohamed.boucadair@orange.com>
    Author:   Daniel King
              <mailto:d.king@lancaster.ac.uk>";

  description
    "The User group-based Control List (UCL) YANG module augments
    the IETF access control lists (ACLs) module and is meant to
    ensure consistent enforcement of ACL policies based on
    the group identity.

    Copyright (c) 2026 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).
```

All revisions of IETF and IANA published modules can be found at the YANG Parameters registry group (<https://www.iana.org/assignments/yang-parameters>).

This version of this YANG module is part of RFC XXXX; see the RFC itself for full legal notices.";

```
revision 2026-01-12 {
  description
    "Initial revision.";
  reference
    "RFC XXXX: A YANG Data Model and RADIUS Extension for
      Policy-based Network Access Control";
}

feature schedule {
  description
    "Indicates support of schedule-based ACEs.";
}

feature match-on-group {
  description
    "Indicates support of matching on endpoint groups.";
}

feature group {
  if-feature "ucl:match-on-group";
  description
    "Indicates support of group-based ACLs.";
}

feature mixed-ipv4-group {
  if-feature "acl:match-on-ipv4 and ucl:match-on-group";
  description
    "IPv4 and group ACL combinations supported.";
}

feature mixed-ipv6-group {
  if-feature "acl:match-on-ipv6 and ucl:match-on-group";
  description
    "IPv6 and group ACL combinations supported.";
}

feature mixed-ipv4-ipv6-group {
  if-feature "acl:match-on-ipv4 and acl:match-on-ipv6 and "
    + "ucl:match-on-group";
  description
    "IPv4, IPv6, and group ACL combinations supported.";
```



```

}

feature mixed-eth-group {
  if-feature "acl:match-on-eth and ucl:match-on-group";
  description
    "Eth and group ACL combinations supported.";
}

feature mixed-eth-ipv4-group {
  if-feature "acl:match-on-eth and acl:match-on-ipv4 and "
    + "ucl:match-on-group";
  description
    "Eth, IPv4, and group ACL combinations supported.";
}

feature mixed-eth-ipv6-group {
  if-feature "acl:match-on-eth and acl:match-on-ipv6 and "
    + "ucl:match-on-group";
  description
    "Eth, IPv6, and group ACL combinations supported.";
}

feature mixed-eth-ipv4-ipv6-group {
  if-feature "acl:match-on-eth and acl:match-on-ipv4 and "
    + "acl:match-on-ipv6 and ucl:match-on-group";
  description
    "Eth, IPv4, IPv6, and group ACL combinations supported.";
}

identity group-acl-type {
  if-feature "group";
  base acl:acl-base;
  description
    "An Access Control List (ACL) that matches based on an endpoint
    group identity, which can represent the collective identity of
    a group of authenticated users, end-devices, or applications.
    An endpoint group identity may be carried in the outer/inner
    packet header (e.g., via NVO3 encapsulation), or may not
    correspond to any field in the packet header. Matching on
    Layer 4 header fields may also exist in the Access Control
    Entries (ACEs).";
}

identity mixed-ipv4-group-type {
  if-feature "mixed-ipv4-group";
  base acl:ipv4-acl-type;
  base ucl:group-acl-type;
  description

```

```
    "An ACL that contains a mix of entries that match on fields
    in the IPv4 header and endpoint group identities, which can
    represent the collective identity of a group of authenticated
    users, end-devices, or applications. Matching on Layer 4
    header fields may also exist in the ACEs.";
}

identity mixed-ipv6-group-type {
    if-feature "mixed-ipv6-group";
    base acl:ipv6-acl-type;
    base ucl:group-acl-type;
    description
        "An ACL that contains a mix of entries that match on fields
        in the IPv6 header and endpoint group identities, which can
        represent the collective identity of a group of authenticated
        users, end-devices, or applications. Matching on Layer 4
        header fields may also exist in the ACEs.";
}

identity mixed-ipv4-ipv6-group-type {
    if-feature "mixed-ipv4-ipv6-group";
    base acl:ipv4-acl-type;
    base acl:ipv6-acl-type;
    base ucl:group-acl-type;
    description
        "An ACL that contains a mix of entries that match on fields
        in the IPv4 header, IPv6 header, and endpoint group
        identities, which can represent the collective identity of a
        group of authenticated users, end-devices, or applications.
        Matching on Layer 4 header fields may also exist in the
        ACEs.";
}

identity mixed-eth-group-type {
    if-feature "mixed-eth-group";
    base acl:eth-acl-type;
    base ucl:group-acl-type;
    description
        "An ACL that contains a mix of entries that match on fields
        in the Ethernet header and endpoint group identities,
        which can represent the collective identity of a group of
        authenticated users, end-devices, or applications. Matching
        on Layer 4 header fields may also exist in the ACEs.";
}

identity mixed-eth-ipv4-group-type {
    if-feature "mixed-eth-ipv4-group";
    base acl:eth-acl-type;
```

```
base acl:ipv4-acl-type;
base ucl:group-acl-type;
description
    "An ACL that contains a mix of entries that match on fields
    in the Ethernet header, IPv4 header, and endpoint group
    identities, which can represent the collective identity of a
    group of authenticated users, end-devices or applications.
    Matching on Layer 4 header fields may also exist in the
    ACEs.";
}

identity mixed-eth-ipv6-group-type {
    if-feature "mixed-eth-ipv6-group";
    base acl:eth-acl-type;
    base acl:ipv6-acl-type;
    base ucl:group-acl-type;
    description
        "An ACL that contains a mix of entries that match on fields
        in the Ethernet header, IPv6 header, and endpoint group
        identities, which can represent the collective identity of
        a group of authenticated users, end-devices or applications.
        Matching on Layer 4 header fields may also exist in the
        ACEs.";
}

identity mixed-eth-ipv4-ipv6-group-type {
    if-feature "mixed-eth-ipv4-ipv6-group";
    base acl:eth-acl-type;
    base acl:ipv4-acl-type;
    base acl:ipv6-acl-type;
    base ucl:group-acl-type;
    description
        "An ACL that contains a mix of entries that match on fields
        in the Ethernet header, IPv4 header, IPv6 header, and endpoint
        group identities, which can represent the collective identity
        of a group of authenticated users, end-devices or
        applications. Matching on Layer 4 header fields may also exist
        in the ACEs.";
}

identity endpoint-group-type {
    description
        "Identity for the type of endpoint group.";
}

identity user-group {
    base ucl:endpoint-group-type;
    description
```

```

    "Identity for the user endpoint group type.";
}

identity device-group {
  base ucl:endpoint-group-type;
  description
    "Identity for the device endpoint group type.";
}

identity application-group {
  base ucl:endpoint-group-type;
  description
    "Identity for the application endpoint group type.";
}

typedef group-id-reference {
  type leafref {
    path "/acl:acls/ucl:endpoint-groups"
      + "/ucl:endpoint-group/ucl:group-id";
  }
  description
    "Defines a reference to a group identifier.";
}

augment "/acl:acls" {
  if-feature "ucl:group";
  description
    "Adds a container for endpoint group definition.";
  container endpoint-groups {
    description
      "Defines a container for the endpoint group list.";
    list endpoint-group {
      key "group-id";
      description
        "Definition of the endpoint group list.";
      leaf group-id {
        type string {
          length "1..64";
        }
        description
          "The endpoint group identifier that uniquely identifies
            an endpoint group.";
      }
      leaf group-type {
        type identityref {
          base endpoint-group-type;
        }
        description

```

```

        "Specifies the endpoint group type.";
    }
}
}

augment "/acl:acls/acl:acl/acl:aces/acl:ace/acl:matches" {
    if-feature "ucl:match-on-group";
    description
        "Specifies how a source and/or destination endpoint group
        index can be referenced as the match criteria in the ACEs.";
    container endpoint-group {
        when "derived-from-or-self(/acl:acls/acl:acl/acl:type, "
            + "'ucl:group-acl-type')";
        description
            "Adds new match types.";
        leaf source-group-id {
            type group-id-reference;
            description
                "The matched source endpoint group ID.";
        }
        leaf destination-group-id {
            type group-id-reference;
            description
                "The matched destination endpoint group ID.";
        }
    }
}

augment "/acl:acls/acl:acl/acl:aces/acl:ace" {
    if-feature "ucl:schedule";
    description
        "Adds schedule parameters to allow the ACE to take effect
        based on date and time.";
    container effective-schedule {
        description
            "Defines when the access control entry rules
            are applied based on date and time condition.

            If it is not configured, the access control entry
            is immediately and always applied.";
        choice schedule-type {
            description
                "Choice based on the type of the time range.";
            container period {
                description
                    "The ACE is applied based on a precise period of
                    time.";
            }
        }
    }
}

```

```

        uses schedule:period-of-time;
    }
    container recurrence {
        if-feature "schedule:icalendar-recurrence";
        description
            "The ACE is applied based on a recurrence rule.";
        uses schedule:icalendar-recurrence;
    }
}
}
}
}
<CODE ENDS>
```

The User-Access-Group-ID Attribute MUST NOT appear in any other RADIUS packet.

The User-Access-Group-ID Attribute is structured as follows:

Type
TBA1

Length
This field indicates the total length, in octets, of all fields of this attribute, including the Type, Length, Extended-Type, and the "Value".

The Length MUST be at most 67 octets. The maximum length is 67 octets to accommodate the maximum group ID of 64 octets plus one octet for Type, one octet for Length, and one octet for Extended-Length.

Data Type
string (Section 3.5 of [RFC8044])

Value
This field contains the user group ID.

7. RADIUS Attributes

Table 4 provides a guide as what type of RADIUS packets that may contain User-Access-Group-ID Attribute, and in what quantity.

Access-Request	Access-Accept	Access-Reject	Access-Challenge	Attribute
0+	0+	0	0	User-Access-Group-ID
Accounting-Request	CoA-Request	CoA-ACK	CoA-NACK	Attribute
0+	0+	0	0	User-Access-Group-ID

Table 4: Table of Attributes

Notation for Table 4:

0 This attribute MUST NOT be present in packet.

0+ Zero or more instances of this attribute MAY be present in packet.

8. Operational Considerations

8.1. Deployment Options

The UCL model can be implemented in different ways.

In some cases, the UCL data model is implemented at the network/administrative domain level with an SDN controller maintaining the dynamical mapping from endpoint group ID to IP/transport fields (e.g., the 5-tuple) and programming the PEPs with IP address/5-tuple based ACLs. In such cases, PEPs do not require to implement specific logic (including hardware) compared to the enforcement of conventional ACLs.

It is possible for the UCL data model to be implemented at the device level. While it eliminates the need for an SDN controller to interact frequently with the PEPs for reasons like the user's context of network connection change or VM/application migration, dedicated hardware/software support might be needed for PEPs to understand the endpoint group identifier. In scenarios where the NAS behaves as the PEP which acquires the source and/or destination endpoint group ID from the AAA server, ACL policy enforcement based on the group identity without being encapsulated into packet headers might affect the forwarding performance. Implementations need to evaluate the operational tradeoff (flexibility brought to the network vs. the complexity of implementation) carefully. Such assessment is out of scope for this document.

8.2. Hardware/Software Implications

Some devices may not have built-in capabilities to enforce group-based match policies. Hardware or software upgrades may be required to support such feature by involved PEPs.

8.3. Mapping Consistency

The specification requires that adequate setup is put in place to map a Group ID to packets fields, typically managed by a controller. Special care should be taken to ensure that such mapping is appropriately enforced when distinct mechanisms (RADIUS, etc.) are supported in network.

9. Security Considerations

9.1. YANG

This section is modeled after the template described in Section 3.7 of [I-D.ietf-netmod-rfc8407bis].

The "ietf-ucl-acl" YANG module defines a data model that is designed to be accessed via YANG-based management protocols such as NETCONF [RFC6241] and RESTCONF [RFC8040]. These YANG-based management protocols (1) have to use a secure transport layer (e.g., SSH [RFC4252], TLS [RFC8446], and QUIC [RFC9000]) and (2) have to use mutual authentication.

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., "config true", which is the default). All writable data nodes are likely to be sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) and delete operations to these data nodes without proper protection or authentication can have a negative effect on network operations. The following subtrees and data nodes have particular sensitivities/vulnerabilities:

- * /acl:acls/ucl:endpoint-groups/ucl:endpoint-group: This list specifies all the endpoint group entries. Unauthorized write access to this list can allow intruders to modify the entries so as to forge an endpoint group that does not exist or maliciously delete an existing endpoint group, which could be used to craft an attack.
- * /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches/ucl:endpoint-group: This subtree specifies a source and/or endpoint group index as match criteria in the ACEs. Unauthorized write access to this data node may allow intruders to modify the group identity so as to permit access that should not be permitted, or deny access that should be permitted.
- * /acl:acls/acl:acl/acl:aces/acl:ace/ucl:effective-schedule: It specifies the scheduling of ACLs. Unauthorized write access to this data node may allow intruders to alter it. This may lead to service disruption or unavailability. Strict access control must be implemented for write operations on this subtree to ensure that only authorized users can modify it.

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. Specifically, the following subtrees and data nodes have particular sensitivities/vulnerabilities:

- * /acl:acls/acl:acl/acl:aces/acl:ace/ucl:effective-schedule: It specifies when the access control entry rules are applied. An unauthorized read access of the list will allow the attacker to determine which rules are applied, to better craft an attack.

9.2. RADIUS

RADIUS-related security considerations are discussed in [RFC2865]. An effort to deprecating insecure practices in RADIUS is provided in [I-D.ietf-radext-deprecating-radius].

This document targets deployments where a trusted relationship is in place between the RADIUS client and server with communication optionally secured by IPsec or Transport Layer Security (TLS) [RFC6614][I-D.ietf-radext-radiusdtls-bis].

10. IANA Considerations

10.1. YANG

This document registers the following URIs in the "IETF XML Registry" [RFC3688].

URI: urn:ietf:params:xml:ns:yang:ietf-ucl-acl
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.

This document registers the following YANG modules in the "YANG Module Names" registry [RFC6020].

name:	ietf-ucl-acl
prefix:	ucl
namespace:	urn:ietf:params:xml:ns:yang:ietf-ucl-acl
maintained by IANA?	N
reference:	RFC XXXX

10.2. RADIUS

This document requests IANA to assign a new RADIUS attribute type from the IANA registry "Radius Attribute Types" [RADIUS-Types]:

Value	Description	Data Type	Reference
TBA1	User-Access-Group-ID	string	This-Document

Table 5: RADIUS Attribute

11. References

11.1. Normative References

- [I-D.ietf-netmod-schedule-yang]
Ma, Q., Wu, Q., Boucadair, M., and D. King, "A Common YANG Data Model for Scheduling", Work in Progress, Internet-Draft, draft-ietf-netmod-schedule-yang-10, 7 August 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-netmod-schedule-yang-10>>.
- [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>>.
- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, DOI 10.17487/RFC2865, June 2000, <<https://www.rfc-editor.org/rfc/rfc2865>>.
- [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>>.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <<https://www.rfc-editor.org/rfc/rfc6241>>.
- [RFC6929] DeKok, A. and A. Lior, "Remote Authentication Dial In User Service (RADIUS) Protocol Extensions", RFC 6929, DOI 10.17487/RFC6929, April 2013, <<https://www.rfc-editor.org/rfc/rfc6929>>.

- [RFC8044] DeKok, A., "Data Types in RADIUS", RFC 8044, DOI 10.17487/RFC8044, January 2017, <<https://www.rfc-editor.org/rfc/rfc8044>>.
- [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>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/rfc/rfc8341>>.
- [RFC8519] Jethanandani, M., Agarwal, S., Huang, L., and D. Blair, "YANG Data Model for Network Access Control Lists (ACLs)", RFC 8519, DOI 10.17487/RFC8519, March 2019, <<https://www.rfc-editor.org/rfc/rfc8519>>.

11.2. Informative References

- [I-D.ietf-netmod-acl-extensions]
de Dios, O. G., Barguil, S., Boucadair, M., and Q. Wu, "Extensions to the Access Control Lists (ACLs) YANG Model", Work in Progress, Internet-Draft, draft-ietf-netmod-acl-extensions-17, 3 April 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-netmod-acl-extensions-17>>.
- [I-D.ietf-netmod-rfc8407bis]
Bierman, A., Boucadair, M., and Q. Wu, "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", Work in Progress, Internet-Draft, draft-ietf-netmod-rfc8407bis-28, 5 June 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-netmod-rfc8407bis-28>>.
- [I-D.ietf-radext-deprecating-radius]
DeKok, A., "Deprecating Insecure Practices in RADIUS", Work in Progress, Internet-Draft, draft-ietf-radext-deprecating-radius-08, 6 November 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-radext-deprecating-radius-08>>.

[I-D.ietf-radext-radiusdtls-bis]

Rieckers, J., Cullen, M., and S. Winter, "RadSec: RADIUS over Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", Work in Progress, Internet-Draft, draft-ietf-radext-radiusdtls-bis-12, 19 January 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-radext-radiusdtls-bis-12>>.

[I-D.smith-vxlan-group-policy]

Smith, M. and L. Kreeger, "VXLAN Group Policy Option", Work in Progress, Internet-Draft, draft-smith-vxlan-group-policy-05, 22 October 2018, <<https://datatracker.ietf.org/doc/html/draft-smith-vxlan-group-policy-05>>.

[I-D.yizhou-anima-ip-to-access-control-groups]

Li, Y., Shen, L., and Y. Zhou, "Autonomic IP Address To Access Control Group ID Mapping", Work in Progress, Internet-Draft, draft-yizhou-anima-ip-to-access-control-groups-02, 15 November 2021, <<https://datatracker.ietf.org/doc/html/draft-yizhou-anima-ip-to-access-control-groups-02>>.

[I-D.you-i2nsf-user-group-based-policy]

You, J., Zarny, M., Jacquenet, C., Boucadair, M., Li, Y., Strassner, J., and S. Majee, "User-group-based Security Policy for Service Layer", Work in Progress, Internet-Draft, draft-you-i2nsf-user-group-based-policy-02, 8 July 2016, <<https://datatracker.ietf.org/doc/html/draft-you-i2nsf-user-group-based-policy-02>>.

[RADIUS-Types]

IANA, "RADIUS Types", <<https://www.iana.org/assignments/radius-types>>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <<https://www.rfc-editor.org/rfc/rfc2475>>.

[RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework for Policy-based Admission Control", RFC 2753, DOI 10.17487/RFC2753, January 2000, <<https://www.rfc-editor.org/rfc/rfc2753>>.

- [RFC3198] Westerinen, A., Schnizlein, J., Strassner, J., Scherling, M., Quinn, B., Herzog, S., Huynh, A., Carlson, M., Perry, J., and S. Waldbusser, "Terminology for Policy-Based Management", RFC 3198, DOI 10.17487/RFC3198, November 2001, <<https://www.rfc-editor.org/rfc/rfc3198>>.
- [RFC3539] Aboba, B. and J. Wood, "Authentication, Authorization and Accounting (AAA) Transport Profile", RFC 3539, DOI 10.17487/RFC3539, June 2003, <<https://www.rfc-editor.org/rfc/rfc3539>>.
- [RFC4252] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, DOI 10.17487/RFC4252, January 2006, <<https://www.rfc-editor.org/rfc/rfc4252>>.
- [RFC6614] Winter, S., McCauley, M., Venaas, S., and K. Wierenga, "Transport Layer Security (TLS) Encryption for RADIUS", RFC 6614, DOI 10.17487/RFC6614, May 2012, <<https://www.rfc-editor.org/rfc/rfc6614>>.
- [RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", RFC 7149, DOI 10.17487/RFC7149, March 2014, <<https://www.rfc-editor.org/rfc/rfc7149>>.
- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", RFC 7426, DOI 10.17487/RFC7426, January 2015, <<https://www.rfc-editor.org/rfc/rfc7426>>.
- [RFC7542] DeKok, A., "The Network Access Identifier", RFC 7542, DOI 10.17487/RFC7542, May 2015, <<https://www.rfc-editor.org/rfc/rfc7542>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/rfc/rfc8040>>.
- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/rfc/rfc8340>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/rfc/rfc8446>>.

- [RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <<https://www.rfc-editor.org/rfc/rfc9000>>.
- [RFC9638] Boutros, S. and D. Eastlake 3rd, Ed., "Network Virtualization over Layer 3 (NVO3) Encapsulation Considerations", RFC 9638, DOI 10.17487/RFC9638, September 2024, <<https://www.rfc-editor.org/rfc/rfc9638>>.
- [RFC9797] Henry, J. and Y. Lee, "Randomized and Changing Media Access Control (MAC) Addresses: Context, Network Impacts, and Use Cases", RFC 9797, DOI 10.17487/RFC9797, June 2025, <<https://www.rfc-editor.org/rfc/rfc9797>>.

Appendix A. Examples Usage

A.1. Configuring the Controller Using Group based ACL

Let's consider an organization that would like to manage the access of R&D employees that bring personally owned devices (BYOD) into the workplace.

The access requirements are as follows:

- * Permit traffic from R&D BYOD of employees, destined to R&D employees' devices every work day from 8:00:00 to 18:00:00 UTC, starting in January 1st, 2026.
- * Deny traffic from R&D BYOD of employees, destined to finance servers located in the enterprise DC network starting at 8:30:00 of January 20, 2026 with an offset of -08:00 from UTC (Pacific Standard Time) and ending at 18:00:00 in Pacific Standard Time on December 31, 2026.

The example shown in Figure 3 illustrates the configuration of an SDN controller using the group-based ACL:

```
{
  "ietf-access-control-list:acls": {
    "ietf-ucl-acl:endpoint-groups": {
      "endpoint-group": [
        {
          "group-id": "R&D",
          "group-type": "ietf-ucl-acl:user-group"
        },
        {
          "group-id": "R&D BYOD",
```

```
    "group-type": "ietf-uc1-acl:user-group"
  },
  {
    "group-id": "finance server",
    "group-type": "ietf-uc1-acl:device-group"
  }
]
},
"acl": [
  {
    "name": "sample-group-acl",
    "type": "ietf-uc1-acl:group-acl-type",
    "aces": {
      "ace": [
        {
          "name": "rule1",
          "matches": {
            "ietf-uc1-acl:endpoint-group": {
              "source-group-id": "R&D BYOD",
              "destination-group-id": "R&D"
            }
          },
          "actions": {
            "forwarding": "ietf-access-control-list:accept"
          },
          "ietf-uc1-acl:effective-schedule": {
            "recurrence": {
              "recurrence-first": {
                "start-time": "2026-01-01T08:00:00Z",
                "duration": "PT10:00:00"
              },
              "frequency": "ietf-schedule:daily",
              "byday": [
                {
                  "weekday": "monday"
                },
                {
                  "weekday": "tuesday"
                },
                {
                  "weekday": "wednesday"
                },
                {
                  "weekday": "thursday"
                },
                {
                  "weekday": "friday"
                }
              ]
            }
          }
        }
      ]
    }
  }
]
```


The examples in this section do not intend to be exhaustive. In particular, explicit IP addresses ("destination-ipv4-network" or "destination-ipv6-network") are provided only for one single rule to illustrate how the mapping between a group ID and IP addresses is translated into an ACL rule entry.

```
{
  "ietf-access-control-list:acls": {
    "ietf-ucl-acl:endpoint-groups": {
      "endpoint-group": [
        {
          "group-id": "R&D",
          "group-type": "ietf-ucl-acl:user-group"
        },
        {
          "group-id": "R&D BYOD",
          "group-type": "ietf-ucl-acl:user-group"
        }
      ]
    },
    "acl": [
      {
        "name": "sample-ucl-ipv4",
        "type": "ietf-ucl-acl:mixed-ipv4-group-type",
        "aces": {
          "ace": [
            {
              "name": "rule1",
              "matches": {
                "ietf-ucl-acl:endpoint-group": {
                  "source-group-id": "R&D BYOD",
                  "destination-group-id": "R&D"
                }
              },
              "actions": {
                "forwarding": "ietf-access-control-list:accept"
              },
              "ietf-ucl-acl:effective-schedule": {
                "recurrence": {
                  "recurrence-first": {
                    "start-time": "2026-01-01T08:00:00Z",
                    "duration": "PT10:00:00"
                  },
                  "frequency": "ietf-schedule:daily",
                  "byday": [
                    {
                      "weekday": "monday"
                    }
                  ]
                }
              }
            }
          ]
        }
      }
    ]
  }
}
```

```

        {
            "weekday": "tuesday"
        },
        {
            "weekday": "wednesday"
        },
        {
            "weekday": "thursday"
        },
        {
            "weekday": "friday"
        }
    ]
}
},
{
    "name": "rule2",
    "matches": {
        "ietf-uc1-acl:endpoint-group": {
            "source-group-id": "R&D BYOD"
        },
        "ipv4": {
            "destination-ipv4-network": "203.0.113.1/24"
        }
    },
    "actions": {
        "forwarding": "ietf-access-control-list:reject"
    },
    "ietf-uc1-acl:effective-schedule": {
        "period": {
            "period-start": "2026-01-20T08:30:00-08:00",
            "period-end": "2026-12-31T18:00:00-08:00"
        }
    }
}
]
}
}
}
```

Figure 4: Example of PEP Configuration

Figure 5 shows an example of the same policy but with a destination IPv6 prefix.

```

{
  "ietf-access-control-list:acls": {
    "ietf-ucl-acl:endpoint-groups": {
      "endpoint-group": [
        {
          "group-id": "R&D",
          "group-type": "ietf-ucl-acl:user-group"
        },
        {
          "group-id": "R&D BYOD",
          "group-type": "ietf-ucl-acl:user-group"
        }
      ]
    },
    "acl": [
      {
        "name": "sample-ucl-ipv6",
        "type": "ietf-ucl-acl:mixed-ipv6-group-type",
        "aces": {
          "ace": [
            {
              "name": "rule1",
              "matches": {
                "ietf-ucl-acl:endpoint-group": {
                  "source-group-id": "R&D BYOD",
                  "destination-group-id": "R&D"
                }
              },
              "actions": {
                "forwarding": "ietf-access-control-list:accept"
              },
              "ietf-ucl-acl:effective-schedule": {
                "recurrence": {
                  "recurrence-first": {
                    "start-time": "2026-01-01T08:00:00Z",
                    "duration": "PT10:00:00"
                  },
                  "frequency": "ietf-schedule:daily",
                  "byday": [
                    {
                      "weekday": "monday"
                    },
                    {
                      "weekday": "tuesday"
                    },
                    {
                      "weekday": "wednesday"
                    }
                  ]
                }
              }
            }
          ]
        }
      ]
    }
  }
}

```

```
{
    {
        "weekday": "thursday"
    },
    {
        "weekday": "friday"
    }
]
}
},
{
    "name": "rule2",
    "matches": {
        "ietf-uc1-acl:endpoint-group": {
            "source-group-id": "R&D BYOD"
        },
        "ipv6": {
            "destination-ipv6-network": "2001:db8:1234::/64"
        }
    },
    "actions": {
        "forwarding": "ietf-access-control-list:reject"
    },
    "ietf-uc1-acl:effective-schedule": {
        "period": {
            "period-start": "2026-01-20T08:30:00-08:00",
            "period-end": "2026-12-31T18:00:00-08:00"
        }
    }
}
]
```

Figure 5: Example of PEP Configuration (ipv6)

A.3. Configuring PEPs Using Address-based ACLs

The section describes an example of configuring a PEP using IP address based ACL. IP address based access control policies could be applied to the PEP that may not understand the group information, e.g., firewall.

Assume an employee in the R&D department accesses the network wirelessly from a non-corporate laptop. The SDN controller associates the user group to which the employee belongs with the user address according to step 1 to 4 in Section 4.1.

Assume the mapping between device group ID and IP addresses is predefined or acquired via device authentication. Figure 6 shows an IPv4 address based ACL configuration delivered from the controller to the PEP. This example is consistent with the example presented in Appendix A.1.

```
{
  "ietf-access-control-list:acls": {
    "acl": [
      {
        "name": "sample-acl-ipv4",
        "type": "ietf-access-control-list:ipv4-acl-type",
        "aces": {
          "ace": [
            {
              "name": "rule1",
              "matches": {
                "ipv4": {
                  "destination-ipv4-network": "192.168.2.1/24",
                  "source-ipv4-network": "192.168.1.1/24"
                }
              },
              "actions": {
                "forwarding": "ietf-access-control-list:accept"
              },
              "ietf-ucl-acl:effective-schedule": {
                "recurrence": {
                  "recurrence-first": {
                    "start-time": "2026-01-01T08:00:00Z",
                    "duration": "PT10:00:00"
                  },
                  "frequency": "ietf-schedule:daily",
                  "byday": [
                    {
                      "weekday": "monday"
                    },
                    {
                      "weekday": "tuesday"
                    },
                    {
                      "weekday": "wednesday"
                    }
                  ]
                }
              }
            }
          ]
        }
      }
    ]
  }
}
```

```
        "weekday": "thursday"
      },
      {
        "weekday": "friday"
      }
    ]
  }
},
{
  "name": "rule2",
  "matches": {
    "ipv4": {
      "destination-ipv4-network": "203.0.113.1/24",
      "source-ipv4-network": "192.168.1.1/24"
    }
  },
  "actions": {
    "forwarding": "ietf-access-control-list:reject"
  },
  "ietf-ucl-acl:effective-schedule": {
    "period": {
      "period-start": "2026-01-20T08:30:00-08:00",
      "period-end": "2026-12-31T18:00:00-08:00"
    }
  }
}
]
}
```

Figure 6: Example of PEP Configuration

Figure 7 shows an example of the same policy but with a destination IPv6 prefix.

```
{
  "ietf-access-control-list:acls": {
    "acl": [
      {
        "name": "sample-acl-ipv6",
        "type": "ietf-access-control-list:ipv6-acl-type",
        "aces": {
          "ace": [
            {
              "action": "deny",
              "protocol": "ipv6",
              "source-address": "1::1",
              "destination-address": "2::2",
              "port": 80,
              "protocol-specific": {}
            }
          ]
        }
      }
    ]
  }
}
```

```
"name": "rule1",
"matches": {
  "ipv6": {
    "destination-ipv6-network": "2001:db8::1/64",
    "source-ipv6-network": "2001:db8::2:1/64"
  }
},
"actions": {
  "forwarding": "ietf-access-control-list:accept"
},
"ietf-uc1-acl:effective-schedule": {
  "recurrence": {
    "recurrence-first": {
      "start-time": "2026-01-01T08:00:00Z",
      "duration": "PT10:00:00"
    },
    "frequency": "ietf-schedule:daily",
    "byday": [
      {
        "weekday": "monday"
      },
      {
        "weekday": "tuesday"
      },
      {
        "weekday": "wednesday"
      },
      {
        "weekday": "thursday"
      },
      {
        "weekday": "friday"
      }
    ]
  }
},
{
  "name": "rule2",
  "matches": {
    "ipv6": {
      "destination-ipv6-network": "2001:db8:1234::/64",
      "source-ipv6-network": "2001:db8::2:1/64"
    }
  },
  "actions": {
    "forwarding": "ietf-access-control-list:reject"
  },
}
```



```

    "ietf-ucl-acl:effective-schedule": {
      "period": {
        "period-start": "2026-01-20T08:30:00-08:00",
        "period-end": "2026-12-31T18:00:00-08:00"
      }
    }
  ]
}

```

Figure 7: Example of PEP Configuration (IPv6)

Acknowledgments

This work has benefited from the discussions of User-group-based Security Policy over the years. In particular, [I-D.you-i2nsf-user-group-based-policy] and [I-D.yizhou-anima-ip-to-access-control-groups] provided mechanisms to establish a mapping between the IP address/prefix of users and access control group IDs.

Jianjie You, Myo Zarny, Christian Jacquenet, Mohamed Boucadair, and Yizhou Li contributed to an earlier version of [I-D.you-i2nsf-user-group-based-policy]. We would like to thank the authors of that Internet-Draft on modern network access control mechanisms for material that assisted in thinking about this document.

Thanks to Joe Clarke, Bill Fenner, Benoît Claise, Rob Wilton, David Somers-Harris, Alan Dekok, Heikki Vatainen, Wen Xiang, Wei Wang, Hongwei Li, and Jensen Zhang for their review and comments.

Thanks to Dhruv Dhody for the OPSDIR review, Alexander Pelov for INTDIR review, Valery Smyslov for the SECDIR review, and Acee Lindem for the YANGDOCTORS review.

Thanks to Mahesh Jethanandani for the AD review.

Authors' Addresses

Qiufang Ma (editor)
Huawei
101 Software Avenue, Yuhua District
Jiangsu
210012
China
Email: maqiufang1@huawei.com

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Jiangsu
210012
China
Email: bill.wu@huawei.com

Mohamed Boucadair (editor)
Orange
35000 Rennes
France
Email: mohamed.boucadair@orange.com

Daniel King
Lancaster University
United Kingdom
Email: d.king@lancaster.ac.uk