

IPv6 Object Groups for ACLs

The IPv6 Object Groups for ACLs feature lets you classify users, devices, or protocols into groups and apply those groups to access control lists (ACLs) to create access control policies for those groups. This feature lets you use object groups instead of individual IP addresses, protocols, and ports, which are used in conventional ACLs. This feature allows multiple access control entries (ACEs), but now you can use each ACE to allow an entire group of users to access a group of servers or services or to deny them from doing so.

In large networks, the number of ACLs can be large (hundreds of lines) and difficult to configure and manage, especially if the ACLs frequently change. Object group-based ACLs are smaller, more readable, and easier to configure and manage than conventional ACLs, simplifying static and dynamic ACL deployments for large user access environments on Cisco IOS routers.

Cisco IOS Firewall benefits from object groups, because they simplify policy creation (for example, group A has access to group A services).

- Restrictions for IPv6 Object Groups for ACLs, on page 1
- Information About IPv6 Object Groups for ACLs, on page 2
- How to Configure Object Groups for ACLs, on page 3
- Configuration Examples for Object Groups for ACLs, on page 5
- Additional References for Object Groups for ACLs, on page 7
- Feature Information for IPv6 Object Groups for ACLs, on page 8

Restrictions for IPv6 Object Groups for ACLs

- Object group-based ACLs support only Layer 3 interfaces (such as routed interfaces and VLAN interfaces). Object group-based ACLs do not support Layer 2 features such as VLAN ACLs (VACLs) or port ACLs (PACLs).
- Object group-based ACLs are not supported with IPsec.
- The highest number of object group-based ACEs supported in an ACL is 2048.
- Empty object groups are automatically deleted.
- The object-group needs to be created before referencing it in the accesslist. An object-group cannot be deleted when it is referenced by other features, like access lists.
- Object groups that contain ACL entries are skipped, if an ACL match is performed for a packet flow.

- Use the same IPv6 address notation when you create and update an IPv6 network object group. If you create the object group with expanded addresses, use expanded addresses for later updates. If you create it with compressed addresses, use compressed addresses for later updates. The device treats an equivalent prefix in a different notation as a duplicate entry. The duplicate entry can cause the configuration commit or controller deployment to fail.

Information About IPv6 Object Groups for ACLs

You can configure conventional Access Contro list Entries (ACEs) and ACEs that refer to object groups in the same ACL.

You can use object group-based ACLs with quality of service (QoS) match criteria, Cisco IOS Firewall, Dynamic Host Configuration Protocol (DHCP), and any other features that use extended ACLs. In addition, you can use object group-based ACLs with multicast traffic.

In larger configurations, this feature reduces the storage needed in NVRAM, because when you use object groups in ACEs, you do not need to define an individual ACE for every address and protocol pairing.

Object Groups

An object group can contain a single object (such as a single IP address, network, or subnet) or multiple objects (such as a combination of multiple IP addresses, networks, or subnets).

A typical access control entry (ACE) allows a group of users to have access only to a specific group of servers. In an object group-based access control list (ACL), you can create a single ACE that uses an object group name instead of creating many ACEs (which requires each ACE to have a different IP address). A similar object group (such as a protocol port group) can be extended to provide access only to a set of applications for a user group. ACEs can have object groups for the source only, destination only, none, or both.

You can use object groups to separate the ownership of the components of an ACE. For example, each department in an organization controls its group membership, and the administrator owns the ACE itself to control which departments can contact one another.

IPv6 addresses and services (protocols) are treated as objects, which are then grouped into various object-groups as required. The two types of object groups are v6-network (for addresses) and v6-service (for protocols) object groups. You can do the nesting of object groups if required.

The object groups can be referenced in the place of protocol or source or destination address while configuring an IPv6 ACE. The ACE containing object group(s) is expanded into individual ACEs (for each object) and programmed into the hardware.

IPv6 network and service object-groups have their own config sub-modes in which the objects are added.

You can use object groups in features that use Cisco Policy Language (CPL) class maps.

This feature supports two types of object groups for grouping ACL parameters: network object groups and service object groups. Use these object groups to group IP addresses, protocols, protocol services (ports), and Internet Control Message Protocol (ICMP) types.

Objects Allowed in Network Object Groups

A network object group is a group of any of the following objects:

- IPv6 address

- Host IPv6 addresses
- Other network object groups
- Subnets

Objects Allowed in Service Object Groups

A service object group is a group of any of the following objects:

- Source and destination protocol ports (such as Telnet or Simple Network Management Protocol [SNMP])
- Internet Control Message Protocol (ICMP) types (such as echo, echo-reply, or unreachables)
- Top-level protocols (such as Encapsulating Security Payload [ESP], TCP, or UDP)
- Other service object groups

ACLs Based on Object Groups

All features that use or reference conventional access control lists (ACLs) are compatible with object-group-based ACLs, and the feature interactions for conventional ACLs are the same with object-group-based ACLs. This feature extends the conventional ACLs to support object-group-based ACLs and also adds new keywords and the source and destination addresses and ports.

You can add, delete, or change objects in an object group membership list dynamically (without deleting and redefining the object group). Also, you can add, delete, or change objects in an object group membership list without redefining the ACL access control entry (ACE) that uses the object group. You can add objects to groups, delete them from groups, and then ensure that changes are correctly functioning within the object-group-based ACL without reapplying the ACL to the interface.

You can configure an object-group-based ACL multiple times with a source group only, a destination group only, or both source and destination groups.

You cannot delete an object group that is used within an ACL or a class-based policy language (CPL) policy.

How to Configure Object Groups for ACLs

To configure object groups for ACLs, you first create one or more object groups. These can be any combination of network object groups (groups that contain objects such as, host addresses and network addresses) or service object groups (which use operators such as **lt**, **eq**, **gt**, **neq**, and **range** with port numbers). Then, you create access control entries (ACEs) that apply a policy (such as **permit** or **deny**) to those object groups.

Configuring IPv6 Object Groups

Object Groups

The following object-groups are added:

```
Device# enable
Device# configure terminal
Device(config)# object-group ?
network     network group
```

```
security    security group
service     service group
v6-network  IPv6 network group
v6-service  IPv6 service group
```

Using Object Groups in IPv6 ACL

Object groups can be used in access-lists in 3 positions: protocol, source and destination IPv6 addresses

The following object-group options are added to existing protocol/address options.

```
Device(config-v6network-group)#?

Device(config-ipv6-acl)# [no] { permit | deny } [ <protocol options> | object-group
<v6service og name> ] { <source address options> | object-group <v6network OG
name> } { <destination address options> | object-group <v6network OG name> }
```

Creating an IPv6 Network Object Group

A network object group that contains a single object (such as a single IP address, a hostname, another network object group, or a subnet) or multiple objects with a network object-group-based ACL to create access control policies for the objects.

Note Use the same IPv6 address notation when you create and update the object group.

For example, *2000:0000:0000:0001::/64* and *2000:0:0:1::/64* identify the same prefix.

If you change the notation during an update, the device treats the prefix as a duplicate entry, and the operation will fail.

Perform the following steps to create IPv6 network object groups:

```
Device> enable
Device# configure terminal
Device(config)# object-group v6-network name
Device(config-v6network-group)# [no] { description <desc> | <x.x.x.x::x/prefix_len> |
host <x.x.x.x::x> | group-object <nested OG name> }


Device(config)#object-group v6-net ognet1
Device(config-v6network-group)#?


V6-Network object group configuration commands:
X:X:X:X::X/<0-128>  - IPv6 network address/prefix length
description         - Network object group description
exit                - Exit from object group configuration mode
group-object        - Nested object group
host                - Host address of group member
no                  - Negate or set default values of a command
```

Creating IPv6 Service Object Groups

Use a service object group to specify TCP and/or UDP ports or port ranges. When the service object group is associated with an access control list (ACL), this service object-group-based ACL can control access to ports.

Perform the following steps to create IPv6 service object group:

```
Device> enable
Device# configure terminal
Device(config)# object-group v6-service <name>
Device(config-v6service-group)# [no] {description <desc> | <0-255> | ahp | esp | hbh |icmp
 [<message type>]
| ipv6 | pcp | { <sctp | tcp | udp | tcp-udp> [source <src port options>]}
[<dest port options>] | group-object <nested OG name> }
Device(config-service-group)# end
```

```
Device# (config-v6service-group)#?
IPv6 Service object group configuration commands:
<0-255>         -  An IP protocol number
ahp             - Authentication Header Protocol
description     - Service object group description
esp             - Encapsulation Security Payload
exit            - Exit from object-group configuration mode
group-object    - Nested object group
hbh             - Hop by Hop options header
icmp            - Internet Control Message Protocol
ipv6            - Any Internet Protocol (v6)
no              - Negate or set default values of a command
pcp             - Payload Compression Protocol
sctp            - Streams Control Transmission Protocol
tcp             - Transmission Control Protocol
tcp-udp         - TCP or UDP protocol
udp             - User Datagram Protocol
```

Verifying IPv6 Object Groups for ACLs

Perform the following steps to verfiy IPv6 object groups for ACLs:

```
Device# enable
Device# show running int <name>-------to check if ACL is applied on the interface
Device# show object-group object-group-name --------to check if configured object groups
are referenced
Device# show ipv6 access-list --------to check the configured ACL
```

The above mentioned show commands display the contents of the named or numbered access list or object group-based ACL (or for all access lists and object group-based ACLs if no name is entered).

Configuration Examples for Object Groups for ACLs

Example: Creating an IPv6 Network Object Group

The following example shows how to create an IPv6 network object group named v6-network ognet1:

```
Device> enable
Device# configure terminal
Device(config)# object-group v6-network ognet1
Device(config-v6-network-group)# 1:1:2::0/32
Device(config-v6-network-group)# host AB:233::23D5
Device(config-v6-network-group)# exit
```

The following example shows how to create a network object group named v6-network ognet2, which contains a host, a subnet, and an existing object group (child) as objects:

```
Device> enable
Device# configure terminal
Device(config)# object-group network v6-network ognet2
Device(config-v6network-group)# 1:2:3::4/36
Device(config-v6network-group)# host AABB::CCDD
Device(config-v6network-group)# group-object ognet1
Device(config-v6network-group)# exit
```

Example: Creating a IPv6 Service Object Group

The following example shows how to create a service object group named v6-service ogserv1, which contains several ICMP, TCP, UDP, and TCP-UDP protocols as objects:

```
Device> enable
Device# configure terminal
Device(config)# object-group service v6-service ogserv1
Device(config-v6service-group)# icmp unreachable
Device(config-v6service-group)# tcp smtp
Device(config-v6service-group)# tcp telnet
Device(config-v6service-group)# tcp source range 3000 4000 telnet
Device(config-v6service-group)# pcp
Device(config-v6service-group)# udp domain
Device(config-v6service-group)# hph
Device(config-v6service-group)# exit
```

Example: Creating an IPv6 Object Group-Based ACL

The following example shows how to create an IPv6 object-group-based ACL that permits packets:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 access-list ogacl1
Device(config-ipv6-acl)# permit object-group ogserv1 5:6:7::5/56 object-group ognet1
Device(config-ipv6-acl)# deny ip object-group ognet2 object-group ognet3
Device(config-ipv6-acl)# permit ipv6 any any
Device(config-ipv6-acl)# exit
```

Example: Verifying IPv6 Object Groups for ACLs

The following example shows how to display all object groups:

```
Device# show object-group

V6-Network object group ognet1
1:1:2::/32
host AB:233::23D5
V6-Network object group ognet2
1:2:3::4/36
host AABB::CCDD
group-object ognet1
```

```
V6-Network object group ognet3
host 1::1
host 1::2
host 1::3
V6-Service object group ogserv1
icmp unreachable
tcp source range 3000 4000 eq telnet
pcp
hbh
```

The following example shows how to display information about IPv6 object-group-based ACL:

```
Device# show ipv6 access-list
IPv6 access list ogacl1
      permit object-group ogserv1 5:6:7::/56 object-group ognet1 sequence 10
      deny ipv6 object-group ognet2 object-group ognet3 sequence 20
      permit ipv6 any any sequence 30
```

Additional References for Object Groups for ACLs

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	• Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
ACL configuration guide	*Security Configuration Guide: Access Control Lists*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 Object Groups for ACLs

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for Object Groups for ACLs

Feature Name	Releases	Feature Information
IPv6 Object Groups for ACLs	Cisco IOS XE Release 16.11.1	The IPv6 Object Groups for ACLs feature lets you classify users, devices, or protocols into groups and apply them to access control lists (ACLs) to create access control policies for those groups. This feature lets you use object groups instead of individual IP addresses, protocols, and ports, which are used in conventional ACLs. This feature allows multiple access control entries (ACEs), but now you can use each ACE to allow an entire group of users to access a group of servers or services or to deny them from doing so.