What Is The "Routerless IXP" Peering Model

Historically, the Netflix Open Connect CDN has offered two partnership models:

A) **Embedded Open Connect Appliances** (OCAS): In this model, ISPs control and operate the router that is connected to the embedded OCAs in their own network at their data center.

B) **SFI Peering** (Public Or Private): In this model, Netflix controls and operates the routers and OCAs that are located at global data center facilities and public internet exchange fabrics (IXPs). We peer at those locations both through public sessions over IXs or with dedicated PNIs between the ISP and Netflix routers.

In some locations, building a traditional SFI peering site may not be viable for any number of reasons. To serve these locations better, Netflix now offers a third deployment model option called **Routerless IXP**.

The **Routerless IXP** model works as follows: The IXP operator provides adequate connectivity to the internet to allow the OCA(s) at their data center site to fill with content on a daily basis and to communicate with our control plane services that run in Amazon Web Services (AWS). The IXP operator establishes a BGP peering session to the OCA(s) using a public ASN and IP addresses. The OCA(s) forward all traffic to the IXP router, which then peers with route servers or individual customers. ***Interested ISPs with a presence at the IXP can opt-in to peer with Netflix at these locations.*** This is the opposite of our typical model at an IXP with a public peering fabric, where members would need to opt-out if they did not want our routes to be sent to them. ***The goal of the opt-in mechanism is to ensure that ISPs who have established structure in their route announcements can avoid any unexpected churn from these new Netflix deployments***.

How Can Member ISPs At The IXP Opt-In To The Routerless IX Implementation

ISP members at an IXP where Netflix has deployed the Routerless IXP model **must explicitly opt-in** to receiving Netflix content. To opt in, the ISP needs to:

A) Set a Netflix-specified BGP community (40027:40000) on some or all of the routes that they have advertised to the IXP operator's router (or indirectly, to the IXP route server).

For more general information about BGP communities, see this article.

It is important to reiterate that the choice to opt-in belongs to the individual member ISP, not the IXP operator. As outlined in our contract with the IXP operator, the IXP operator may not opt their members in by adding this community.

Example Configurations For Participating ISPs

The following examples illustrate how to apply the opt-in BGP community. In these examples we shall use the actual route server details:

ASN	37386	
	IPv4	**IPv6**
RS01	196.223.25.11	2001:43f8:130::11
RS02	196.223.25.12	2001:43f8:130::12

The IXP participant (ISP), for the purposes of this example, is assumed to have IP address 196.223.25.254 / 2001:43f8:130::254 and AS 64500

Cisco Example

```
!
! Display BGP communities in XXX:YYY format:
ip bgp-community new-format
!
! Prefix list of what IPv4 prefixes you are advertising to the route server
ip prefix-list IX-ADVERTISE-IPV4-OUT seq 10 permit <Your IPv4 Range>


!
! Route map to add the 40027:40000 community to all advertised routes
! and to ensure only routes in the prefix list are advertised (to avoid route leaks)
route-map IX-RS-IPV4-OUT permit 10
   match ip address prefix-list IX-ADVERTISE-IPV4-OUT
   set community 40027:40000

! Prefix list of what IPv6 prefixes you are advertising to the route server
ipv6 prefix-list IX-ADVERTISE-IPV6-OUT seq 10 permit <Your IPv6 Range>


!
! Route map to add the 40027:40000 community to all advertised routes
! and to ensure only routes in the prefix list are advertised (to avoid route leaks)
route-map IX-RS-IPV6-OUT permit 10
   match ipv6 address prefix-list IX-ADVERTISE-IPV6-OUT
   set community 40027:40000


!
! BGP configuration for route server
router bgp 64500
   no bgp enforce-first-as
   neighbor196.223.25.11 remote-as 37386
   neighbor196.223.25.11 description IX Route Server
   neighbor 2001:43f8:130::11 remote-as 37386
   neighbor 2001:43f8:130::11 description IX Route Server


!
! Address family configurations - note the "send-community" ensures
! that communities are properly sent and that the route-map causes
! the route-map to get called to add the prefixes
address-family ipv4
   neighbor196.223.25.11 activate
   neighbor196.223.25.11 send-community both
   neighbor196.223.25.11 route-map IX-RS-IPV4-OUT out
   exit-address-family
!
address-family ipv6
   neighbor 2001:43f8:130::11 activate
   neighbor 2001:43f8:130::11 send-community both
   neighbor 2001:43f8:130::11 route-map IX-RS-IPV6-OUT out
   exit-address-family
```

Arista Example

```
!
! Prefix list of what IPv4 prefixes you are advertising to the route server
ip prefix-list IX-ADVERTISE-IPV4-OUT seq 10 permit <Your IPv4 Range>

!
! Route map to add the 40027:40000 community to all advertised routes and to ensure
! only routes in the prefix list are advertised (to avoid route leaks)
route-map IX-RS-IPV4-OUT permit 10
match ip address prefix-list IX-ADVERTISE-IPV4-OUT
set community 40027:40000

!
! Prefix list of what IPv6 prefixes you are advertising to the route server
ipv6 prefix-list IX-ADVERTISE-IPV6-OUT seq 10 permit <Your IPv6 Range>

!
! Route map to add the 40027:40000 community to all advertised routes and to ensure
! only routes in the prefix list are advertised (to avoid route leaks)
route-map IX-RS-IPV6-OUT permit 10
match ipv6 address prefix-list IX-ADVERTISE-IPV6-OUT
set community 40027:40000

!
! BGP configuration for route server
router bgp 64500
neighbor196.223.25.11 remote-as 37386
neighbor 196.223.25.11 description IX Route Server
neighbor 196.223.25.11 route-map IX-RS-IPV4-OUT out
no neighbor 196.223.25.11 enforce-first-as
neighbor 196.223.25.11 send-community standard
neighbor 2001:43f8:130::11 remote-as 37386
neighbor 2001:43f8:130::11 description IX Route Server
neighbor 2001:43f8:130::11 route-map IX-RS-IPV6-OUT out
no neighbor 2001:43f8:130::11 enforce-first-as
neighbor 2001:43f8:130::11 send-community standard
!
address-family ipv6 neighbor 2001:43f8:130::11 activate
!
```

Juniper Example

```
config@juniper# show protocols bgp
group IX-RS-IPV4 {
    type external;
    description "IX IPv4 Route Servers";
    family inet {
        unicast;
    }
    export IX-RS-IPV4-OUT;
    peer-as 37386;
    neighbor 196.223.25.11 {
        description "IX Route Server";
    }
}
group IX-RS-IPV6 {
    type external;
    description "IX IPv6 Route Servers";
    family inet {
        unicast;
    }
    export IX-RS-IPV6-OUT;
    peer-as 37386;
    neighbor 2001:43f8:130::11 {
        description "IX Route Server";
    }
}

config@juniper# show policy-options
prefix-list IX-ADVERTISE-IPV4-OUT {
    <prefixes>
}
prefix-list IX-ADVERTISE-IPV6-OUT {
    <prefixes>
}

policy-statement IX-RS-IPV4-OUT {
    term exports {
        from {
            rib inet.0;
            prefix-list IX-ADVERTISE-IPV4-OUT;
        }
        then {
            community add NETFLIX_ENABLE;
            accept;
        }
    }
    term defaultdeny {
        then reject;
    }
}
policy-statement IX-RS-IPV6-OUT {
    term exports {
        from {
            rib inet6.0;
            prefix-list IX-ADVERTISE-IPV6-OUT;
        }
        then {
            community add NETFLIX_ENABLE;
            accept;
        }
    }
    term defaultdeny {
        then reject;
    }
}
community NETFLIX_ENABLE members 40027:40000;
```