Illumio Core®

Compatible PCE Versions: 20.2.0, 21.1.0, 21.2.x

NEN

Version: 2.1.0

NEN Installation and Usage Guide

June 2022

22000-100-2.1.0

Legal Notices

Product Versions

NEN Version: 2.1.0

Compatible PCE Versions: 20.2.0, 21.1.0, 21.2.x

Standard versus LTS Releases

For information on Illumio software support for Standard and LTS releases, see Versions and Releases on the Illumio Support portal.

Resources

Legal information, see https://www.illumio.com/legal-information

Trademarks statements, see https://www.illumio.com/trademarks

Patent statements, see https://www.illumio.com/patents

License statements, see https://www.illumio.com/eula

Open source software utilized by the Illumio Core and their licenses, see *Open Source Licensing Disclosures* in the Illumio Core Technical Documentation portal.

Contact Information

To contact Illumio, go to https://www.illumio.com/contact-us

To contact the Illumio legal team, email us at legal@illumio.com

To contact the Illumio documentation team, email us at doc-feedback@illumio.com

Contents

Chapter 1

NEN Installation and Configuration

This chapter contains the following topics:

This section provides an overview of the NEN, describes how to install the NEN, and configure the supported switches.

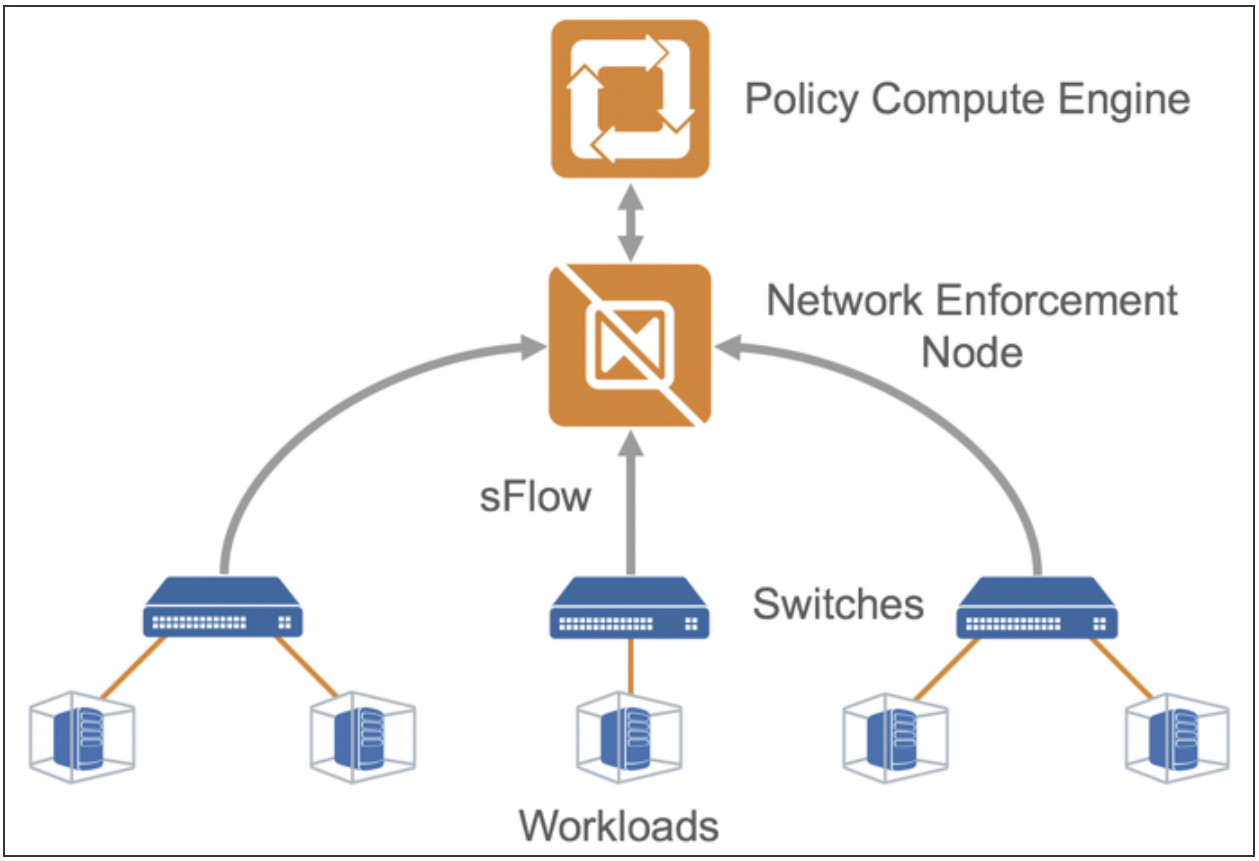
Overview of the NEN

The Illumio Network Enforcement Node (NEN) is the Illumio Core switch interface, which allows you to get visibility and enforcement on switches. The NEN has both switch and load balancer capabilities. Using the NEN, you can secure workloads that are attached to network switches. You can use the NEN to generate access control lists (ACLs) and load those on your switches to protect the ports to which your workloads are attached. Unlike the standard Illumio Virtual Enforcement Node (VEN) that is installed on workloads, the NEN is installed as a service on the Policy Compute Engine (PCE) or run as a standalone service. The NEN can manage multiple workloads and enforce policy

for those workloads. Every IP address associated with the managed network endpoints has one workload associated with it. Although, only one NEN will run per PCE cluster, the architecture supports multiple NENs, provided each is a standalone node within the PCE cluster.

Traffic Visualization

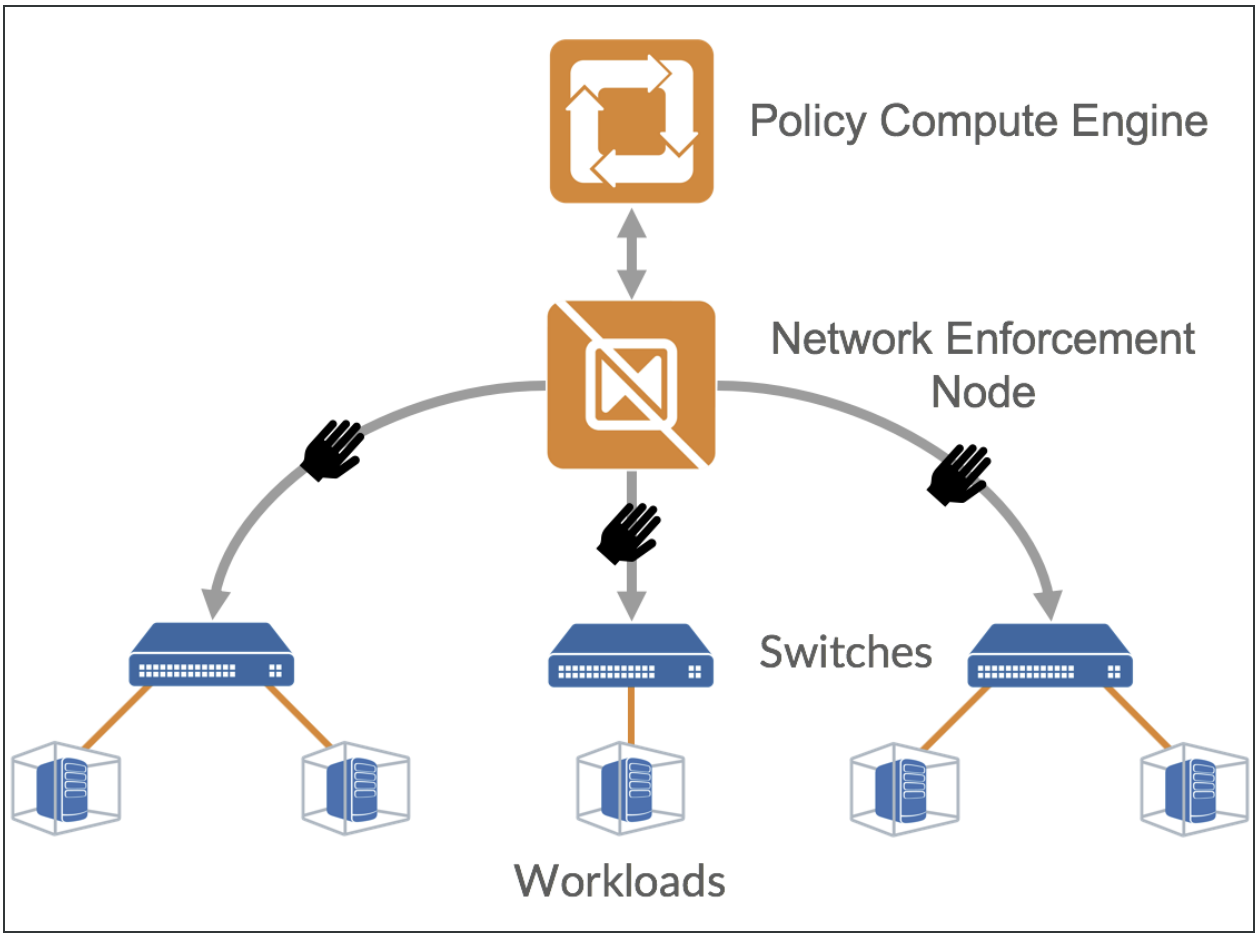
Visibility of communication across applications is critical for segmentation. The optimal method of getting visibility is to use a lightweight agent or a VEN, to report all inbound and outbound communications for each workload. In cases where a VEN cannot be installed, such as in legacy Windows machines, storage filers, or appliances, the NEN extends visualization capabilities to agentless workloads via the network. With the NEN, network administrators can configure their switches to send sFlow data to an sFlow collector, such as the NEN. An Illumio Core administrator can configure the NEN to listen for sFlow data from switches and associate workloads to those switches. The NEN receives sFlow data directly from the switches, summarizes it, and uploads it to the PCE. You can view this traffic flow in the Illumination® map and stream it out of the PCE through UDP in Splunk, CEF, or LEEF formats.



Extended Policy Model

The Illumio policy model encompasses workloads with native stateful firewalls built-in, such as Linux iptables or Windows Filtering Platform. Although all systems might not have a firewall built in, they still have segmentation requirements. To solve this use case, Illumio has extended its policy model to switches.

Illumio administrators can use the NEN to convert natural language policies into ACLs, which the switches understand natively. Your organization's teams that use Illumio Core can download ACLs from the PCE and provide them to the networking team for review before applying new policies to the switches.



NEN Limitations

This release is subject to the following limitations:

- You must provide a switch IP address and an interface traffic flow ID for interfaces that need to be monitored for sFlow data.
- The NEN discards sFlow data from an interface that it does not monitor.
- The Illumio Core generates only IPv4 ACLs that can be applied to either the L3/Routed interfaces or Switch Virtual Interface (SVI) for L2 interfaces when they are a member of a VLAN. Whenever ACLs are applied to the SVI, workloads within the same VLAN can freely communicate regardless of policy.

  This is a limitation of IPv4 ACLs on switches. Inter-VLAN or routed traffic will still be filtered by ACLs.

- For a Supercluster deployment, you can install the NEN only on the two database nodes of the Supercluster leader. You can install only one NEN and you cannot install it on a standalone PCE or non-Supercluster leader nodes.

PCE-based NEN Requirements

- Illumio-provided PCE 20.2.0 or later and NEN 2.1.0 (includes NFC) software packages.
- Cisco Nexus 9200 or 9300 or Arista 7000 series switch.
- Workloads that are directly attached to the switch on L2 or L3 ports or on port channels.

For the complete list of OS support for the NEN, see NEN OS Support and Package Dependencies on the Illumio Support portal.

NOTE:
The NEN targets top-of-rack (TOR) switches that are directly attached to the workload and not the core switches. For example, Cisco Nexus 9200 and 9300 (TOR) switches are supported, but the Cisco 9500 series switches are not supported.

Standalone NEN Hardware Requirements

To install a standalone NEN to support various numbers of Server Load Balancers and Virtual IPs, your hardware must meet the hardware requirements detailed in this section.

Server Load Balancers (SLBs)	Virtual IPs (VIPs)	Cores/Clock Speed[1]	RAM per Node[2]	Storage Device Size[3] and IOPS[4]	Network
Up to 6 SLBs	• Max 1,000 VIPs per SLB	• 2 cores • Intel® Xeon(R) CPU E5-2695 v4	8 GB	A single node including both core and data:	1 Gb Ethernet

Server Load Balancers (SLBs)	Virtual IPs (VIPs)	Cores/Clock Speed[1]	RAM per Node[2]	Storage Device Size[3] and IOPS[4]	Network
	• Max 3,000 VIPs across all SLBs	at 2.10GHz or equivalent		• 1 x 50 GB • 100 IOPS per device	
Up to 50 SLBs	• Max 1,000 VIPs per SLB • Max 12,000 VIPs across all SLBs	• 4 cores • Intel® Xeon(R) CPU E5-2695 v4 at 2.10GHz or equivalent	16 GB	A single node including both core and data: • 1 x 50 GB • 100 IOPS per device	1 Gb Ethernet

Footnotes:

[1] CPUs:

- The recommended number of cores is based only on physical cores from allocated CPUs, irrespective of hyper-threading or virtual cores. For example, in AWS one vCPU is only a single hyper-thread running on a physical core, which is half a core. 16 physical cores equates to 32 vCPUs in AWS.
- Full reservations for vCPU. No overcommit.

[2] Full reservations for vRAM. No overcommit.

[3] Additional disk notes:

- Storage requirements for network traffic data can increase rapidly as the amount of network traffic increases. Allocating a separate, large storage device for traffic data can accommodate these rapid changes without

potentially interrupting the service.

- Network File Systems (NFS) is not supported.

[4] Input/output operations per second (IOPS) are based on 8K random write operations. IOPS specified for an average of 300 flow summaries (80% unique `src_ip, dest_ip, dest_port, proto`) per workload every 10 minutes. Different traffic profiles might require higher IOPS.

Machine Resource Requirements for NEN VMs

Storage Device	Partition mount point	Size to Allocate
Device 1, Partition A	/	8 GB
Device 1, Partition B	/v/log	16 GB[1]
Device 1, Partition C	/var/lib/illumio-nen	Balance of Device 1

Footnote:

[1] The size of this partition assumes that PCE application logs and system logs are both stored in `/var/log/illumio-nen`. PCE application logs are stored in the `/var/log/illumio-pce` directory.

NEN Software

Download the NEN software from the Illumio's support site (login required) and install the NEN RPM package. See Install and Activate the NEN for information.

NOTE:
You can run the NEN as a service on a PCE SNC or MNC.

The self-signed certificates are supported in case of a PCE-NEN combined setup (NEN running on a PCE node). A standalone NEN cannot communicate with a PCE with self-signed certificates.

Support for Load Balancers

The Network Function Controller (NFC) is no longer supported from Illumio Core 19.3.0 release onwards. The load balancer interface has been moved from

the PCE in to the NEN. Since the NFC has been discontinued, you need the NEN to interface with the load balancer.

From the NEN 2.0.0 release onwards, the AVI Vantage load balancers are also supported. See Load Balancers and Virtual Servers for the NEN for information.

> **IMPORTANT:**
> The NEN 2.1.0 release supports up to 500 VIPs and up to 15 SLBs.

Recommended Skills

Illumio® recommends that you be familiar with:

- Your organization's security goals.
- Thorough understanding of Illumio Core.
- Your organization's network switches configuration and management.

Supported Switches and Configurations

The following switches are supported in this release:

- Cisco Nexus 9200 and 9300 series
- Arista 7000 series

Cisco Nexus 9000 Configuration

The following ACL and interface configurations are supported for the Illumio NEN integration:

ACL Implementation	Switch Interfaces	ACL Type
Router ACL (RACL) RACLs support both inbound and outbound enforcement.	• VLAN interface (SVI) • Layer 3 physical interface • Layer 3 port-channel interface	IPv4

IMPORTANT:
Cisco Nexus 9000 Series unsupported interface and ACL configurations

The NEN does not support:

- VLAN ACL (VACL) or Virtual Teletype (VTY) ACL as the ACL implementation
- VLAN trunk port (switchport mode trunk) or sub-interface as the switch interface
- MAC ACL type
- IPv6 ACL type
- PACLs for Layer 2 interfaces.

Administrative Access to the Switch

You or your network administrators need administrative access to your switches to configure them and load the NEN-generated ACLs.

NOTE:
The PCE and the NEN do not send any communication to the switch and never log into the switch. The PCE and the NEN do not require root or admin privileges on the switch.

Sufficient TCAM

Your switch's ternary content-addressable memory (TCAM) must be sufficient to store the IPv4 RACLs generated by the NEN.

NOTE:
Illumio does not provide a mechanism to check the TCAM depth or available memory for each platform. Your network or security administrators need to check whether the generated IP ACLs can be handled by the switch.

Enable sFlow

The NEN relies on sFlow to provide network traffic flow data for Illumination. Your switch must be configured with sFlow. See your vendor documentation for information.

Configure sFlow Output

The output of sFlow from the switch must be sent to the PCE so it can be monitored. The well-known port for sFlow is port UDP 6343. See Configure Switches for NEN for information.

Network Connectivity between Switches and NEN

The NEN listens for sFlow from the switches.

> IMPORTANT:
> Ensure that your network is configured to allow communication between your switches and the NEN.

Switch Information

You need to provide switch-related information in the PCE web console. See the table listed in Add Unmanaged Workloads and Switch Definitions in the PCE Web Console for information.

Install and Activate the NEN

This section describes how to install and activate the NEN on PCE data nodes.

Workflow for Setting up the NEN

This following is an overview of the steps required for working with the NEN:

1. Install the NEN software package on the PCE.
2. In the PCE web console:
    a. Define the switches.
    b. Create unmanaged workloads.

c. Assign those unmanaged workloads to switch interfaces.
d. Create security policy rules to protect the workloads attached to the switches.

3. Use the PCE REST API or the PCE web console to generate switch ACLs based on your organization's security policies.
4. Copy and paste the generated ACLs to configure the switch via the switch's command line.
5. Using the PCE REST API or the PCE web console, inform the PCE that the ACLs have been loaded.

Result: The PCE-generated ACLs on the switch will protect the target workloads.

Install the NEN on PCE data nodes

The RPM installation of the NEN requires `root` or `sudo` permissions on the PCE data nodes.

IMPORTANT:
If you are upgrading a PCE/NEN system from a pre-19.3.1 or from 20.1.0, you need to run the `sudo rpm -e illumio-pce-nen --noscripts` command before installing the 21.2.*x*/2.1.0 PCE/NEN RPMs.

To install the NEN on PCE data nodes:

1. Download the NEN rpm "`illumio-pce-nen-<version#-build#>.rpm`" from the Illumio support site.
2. Log into the PCE as root:

```
$ sudo su
```

3. Navigate to the `db0` and `db1` nodes and run the following command on each node:

```
rpm sudo rpm -ivh illumio-pce-nen-2.1.0-<build-number>.x86_64.rpm
```

To avail HA, you need to install the NEN on both the db nodes of the PCE.

4. Restart both db0 and db1 nodes:

```
sudo -u ilo-pce illumio-pce-ctl restart
```

Result: The NEN is installed but not yet activated. To activate it, see Activate the NEN on PCE data nodes.

Activate the NEN on PCE data nodes

You can activate the NEN service using the PCE web console and the PCE command line.

1. Log into the PCE web console.
2. From the left navigation menu, choose **Workloads and VENS > Workloads**.
3. Click **Add > Pair Workload with Pairing Profile**.
4. Select the **default** from the 'Pick a Pairing Profile' drop-down options.
5. Copy the pairing **Key** value (alphanumeric).
6. Log in to PCE **data node 0** and run the following command:

```
$ sudo -u ilo-pce /opt/illumio-pce/illumio-nen-ctl activate <enter pairing key value>
```

NOTE:
The command is *not* `illumio-pce-ctl`, which is usually used. It is `illumio-nen-ctl`.

Result: The NEN is activated on the PCE.

Install Single RPM

From the NEN 2.1.0 release onwards, there is a single RPM that you can install on a standalone node or on a PCE/NEN setup.

If you are upgrading from PCE 19.3.0 or earlier to PCE 21.2.0 or later or from PCE 20.1.0 to PCE 21.2.0 or later and you have the NEN installed on a PCE, run the following command before installing the PCE RPMs and the NEN RPM:

```
rpm -e illumio-pce-nen --noscript
```

Standalone NEN

NOTE:
For standalone NEN hardware requirements, see Standalone NEN Hardware Requirements.

To install NEN as a standalone service:

1. Download the NEN Standalone rpm "`illumio-standalone-nen<version#-build#>.rpm`" from the Illumio support site.
2. Run the following command:

   ```
   # rpm -i illumio-standalone-nen-2.1.0-<build-number>.x86_64.rpm
   ```

3. Locate the runtime environment file sample at:

   ```
   /opt/illumio-nen/illumio/config/templates
   ```

4. Copy the file to the below location and update it with your pairing key and certificate information:

   ```
   /etc/illumio-nen/runtime_env.yml
   ```

5. Run the following commands:

   ```
   cd /opt/illumio-nen/
   ./illumio-nen-env setup
   sudo -u ilo-nen ./illumio-nen-ctl start -svw –runlevel 5
   ```

6. Activate the NEN:

```
sudo -u ilo-nen./illumio-nen-ctl activate --host <hostname>:8443 <enter
pairing key value from PCE Pairing Profile>
```

Enable Load Balancer

After installing the NEN RPM on the PCE and activating it, enable the load balancer (or SLB) functionality by running the following command on the NEN node (only db0):

```
sudo -u ilo-pce ./illumio-nen-ctl slb-enable
```

NOTE:
If you were previously using load balancer support, you do not need to run the above command because it will already be enabled.

For a standalone NEN, enable the load balancer functionality by running the following command:

```
sudo -u ilo-nen ./illumio-nen-ctl slb-enable
```

After the NEN is paired with the PCE, you will see a new item in the PCE UI main menu > **Infrastructure > Switches**.

Configure Switches for NEN

sFlow on the switch must be configured to send its output to the PCE. In addition, the sFlow-monitored interfaces on the switch must be configured in the NEN service via the PCE web console. If the NEN service receives sFlow information from an unrecognized or undefined network endpoint (or interface), it will reject that information. The NEN service continually aggregates the sFlow traffic and sends the aggregated information to the PCE traffic collector every 10 minutes.

NOTE:
sFlow is only a sampling protocol, so all the flows might not be recorded. If the default sampling rate is not sufficient for your use case, see your vendor documentation.

Configure sFlow on Cisco Switch

Use the following `(config)#` commands to configure sFlow on a Cisco 9000 series switch:

1. Enable sFlow:

```
(config)# feature sflow
```

2. In the following command, the `NEN_ip_address` variable is the IP address of the PCE:

```
(config)# sflow collector-ip NEN_ip_address vrf default
```

3. In the following command, the `switch_IP_address` variable is the IP address of the switch, which you will also use in the PCE web console. `switch_IP` is a management IP address.

```
(config)# sflow agent-ip switch_IP_address
```

4. In the following command, the `interface_name_to_monitor` variable is a mnemonic name that you have already defined on the switch for the interface, which you will also use in the PCE web console.

```
(config)# sflow data-source interface interface_name_to_monitor
```

5. Repeat the above `sflow data-source interface` command for all interfaces on the switch that you want to secure.

See Add Unmanaged Workloads and Switch Definitions in the PCE Web Console for information.

Example of sFlow Configuration for Cisco

```
nexus9000(config)# show run sflow

!Command: show running-config sflow

feature sflow

sflow collector-ip 10.10.10.1 vrf default
sflow agent-ip 10.20.20.1

sflow data-source interface Ethernet1/7
```

In this example:

- The IP address on the switch that can communicate with the PCE is 10.20.20.1.
- The PCE/NEN IP address (sFlow collector) is 10.10.10.1.
- A workload is directly attached to interface Ethernet 1/7.

Collect SNMP ifIndex Value for Cisco

When the switch reports sFlow to the NEN, it includes interface index details in the flow records. When the NEN receives sFlow, it parses the records and retains the records only for the interfaces you specify in the NEN configuration. You need to collect the `ifindex` IDs and add them to the NEN configuration later. You can determine your switches' SNMP ifIndex values using the following command:

```
# show interface snmp-ifindex
```

Manufacturer/Model	Command	Notes
Cisco 9000	In privileged mode: `show interface snmp-ifindex`	This command outputs the IFMIB (decimal) and the ifIndex (hex) values. You need the IFMIB

Manufacturer/Model	Command	Notes
		(decimal) value later. This value is required to configure **Monitor Traffic** for the NEN.

Example of Command Output

```
nexus9000# show interface snmp-ifindex
-------------------------------------------------------------------------------
Port   IFMIB      Ifindex (hex)
-------------------------------------------------------------------------------
mgmt0  83886080  (0x5000000)
Eth1/1 436207616 (0x1a000000)
Eth1/2 436208128 (0x1a000200)
Eth1/3 436208640 (0x1a000400)
Eth1/4 436209152 (0x1a000600)
Eth1/5 436209664 (0x1a000800)
Eth1/6 436210176 (0x1a000a00)
Eth1/7 436210688 (0x1a000c00)
Eth1/8 436211200 (0x1a000e00)
```

This example uses Ethernet 1/7 interface as an sFlow source interface. To enter the interface information in the PCE, collect the decimal value of the ifIndex. In case of the Cisco Nexus 9000, this value is in the **IFMIB** column of the command output. The command output above shows 436210688 as the IFMIB value for Ethernet 1/7 interface. This value is required to configure the **Monitor Traffic** field in the PCE configuration page.

Configure sFlow on Arista Switch

Use the following commands to configure sFlow on an Arista 7000 series switch:

1. Run sFlow (this command is similar to enabling sFlow on a Cisco switch):

```
sflow run
```

2. In the following command, the IP address is the destination PCE IP to which the sFlow information should be sent:

```
sflow destination 10.6.1.158
```

3. In the following command, the IP address is the source IP from where the sFlow information is sent:

```
sflow source 10.21.6.1
```

On an Arista switch, the list of sFlow command options are:

Command	Description
`destination`	Set sFlow collector destination.
`extension`	Configure sFlow extension settings.
`polling-interval`	Set polling interval (secs) for sFlow.
`qos`	Configure QoS parameters.
`run`	Run sFlow globally.
`sample`	Set sample characteristics for sFlow.
`source`	Set the source IP address.
`source-interface`	Configure the source interface for sFlow datagrams.
`vrf`	Configure VRFs.

Collect SNMP ifIndex Value for Arista

You can determine your Arista switches' SNMP ifIndex values using the following command:

```
arista7000# show snmp mib ifmib ifindex
Ethernet1: Ifindex = 1
Ethernet2: Ifindex = 2
```

```
Ethernet3: Ifindex = 3
Ethernet4: Ifindex = 4
Ethernet5: Ifindex = 5
Ethernet6: Ifindex = 6
Ethernet7: Ifindex = 7
Ethernet8: Ifindex = 8
```

Add Unmanaged Workloads and Switch Definitions in the PCE Web Console

To create a security policy, the switches and the workloads attached to them should be defined in the PCE web console as follows:

1. Log into the PCE web console.
2. Define the unmanaged workloads that are attached to the switch by selecting **Workloads and VENs > Workloads > Add > Add Unmanaged Workload**. You will associate these unmanaged workloads with their switches later.

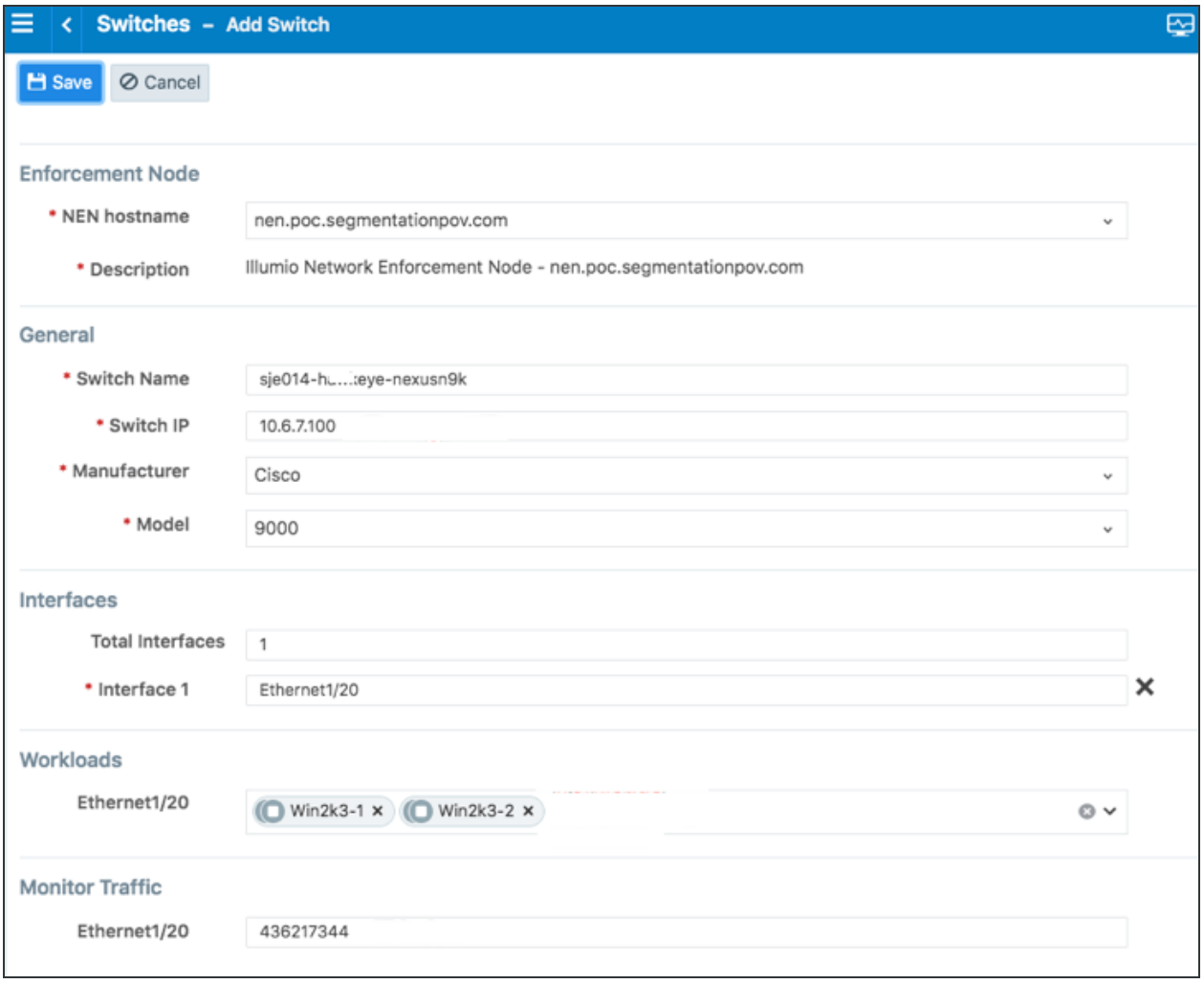
   See the *Security Policy Guide* for information on adding unmanaged workloads.
3. Define the switches and associated workloads, by selecting **Infrastructure > Switches**.
4. Click **Add**.
5. Enter the details in the displayed fields as described in the table below.
6. After entering or selecting values for all the required fields, click **Save**.

Fields in the **PCE web console > Infrastructure > Switches > Add Switch** page:

Field Name	Description	Required	Notes
NEN host-name	FQDN of the PCE that runs the NEN service	Yes	This field is populated with the FQDN of your PCE. You cannot edit this field.
Description	Description of the NEN service	Yes	This field is populated with "Illumio Network Enforcement Node" and the FQDN of your

Field Name	Description	Required	Notes
			PCE. You cannot edit this field.
Switch Name	A free-form, mnemonic name of your choice for the switch	Yes	Make this name easy to remember and distinguishable from other switch names.
Switch IP	IP address of the switch	Yes	Corresponds to `switch_IP_address` that you defined in Configure sFlow on Cisco Switch. It is also known as sflow agent-ip in Cisco switches.
Manufacturer	Name of the switch manufacturer	Yes	Select Cisco.
Model	Model number of the switch	Yes	Select 9000.
Interfaces	Defined interfaces on the switch	No	Corresponds to `interface_name_to_monitor` you defined on the switch and configured in Configure sFlow on Cisco Switch. This can be a custom string. You can also add interfaces that are not monitored by sFlow.
Workloads	Names of workloads connected to the switch's defined interfaces	No	Only those workloads assigned to the switch interfaces are secured. You can attach one or more workloads to an interface.
Monitor	SNMP ifIndex of	Yes/No	If the interface is monitored

Field Name	Description	Required	Notes
Traffic	the switch interface See Collect SNMP ifIndex Value for Cisco and Collect SNMP ifIndex Value for Arista.		by sFlow, the **Monitor Traffic** field is required.



NEN Configuration with the REST API

To manage network switches reporting data flows to the NEN and to get the generated ACLs to enforce policies based on what's been defined in the PCE,

you need to complete these tasks:

1. Get the list of switches and their details.
2. Generate the ACLs for one or all switches.
3. Print the ACLs in the desired format.

The sections below describe the manual steps, which can be inserted in any script to automate this process.

Get List of Switches and Details

To get the list of all the network switches registered against the NEN, run the following curl command:

```
curl -u api_xxx:xxx -H "Accept: application/json" -X
GET https://mypce.domain.io:8443/api/v2/orgs/1/network_devices
```

Result: Returns a list of switches with all the reported endpoints (ports, workloads) to the NEN.

Curl Command of Get List of Switches

```
curl -u api_
1853ebfcb1187acb4:9c2a381773a44e3a609448109278c02c4ec1fe597f9643af71a832c0a8b0c0d0
-H "Accept: application/json" -X GET
https://mypce.domain.io:8443/api/v2/orgs/1/network_devices
```

Response

```
[
   {
      "network_enforcement_node" : {
         "href" : "/orgs/1/network_enforcement_nodes/f64e78b7-2917-409f-9093-
9d6ddaa35799"
      },
      "href" : "/orgs/1/network_devices/f07a077a-70ad-4b57-b82a-f1d204fcfd99",
      "configure" : false,
```

```
      "network_endpoints" : [
         {
            "href" : "/orgs/1/network_devices/f07a077a-70ad-4b57-b82a-
f1d204fcfd99/network_endpoints/1ff6f037-d644-438e-ab32-019a45a7d8d5"
         },
         {
            "href" : "/orgs/1/network_devices/f07a077a-70ad-4b57-b82a-
f1d204fcfd99/network_endpoints/dd687e16-6998-4a39-8bde-a7fb445f18d9"
         },
         {
            "href" : "/orgs/1/network_devices/f07a077a-70ad-4b57-b82a-
f1d204fcfd99/network_endpoints/7345aed3-1fbd-4596-ada9-f6cbfb361dfe"
         },
         {
            "href" : "/orgs/1/network_devices/f07a077a-70ad-4b57-b82a-
f1d204fcfd99/network_endpoints/be58f614-7cc7-4132-a409-97ea8334dfeb"
         }
      ],
      "enforcement_instructions_data_timestamp" : "2019-05-06T15:45:02Z",
      "enforcement_instructions_data_href" : "/orgs/1/datafiles/49b11cf6-d6f9-
4efc-8cb2-c1a444cb9c02",
      "supported_endpoint_type" : "switch_port",
      "config" : {
         "model" : "9000",
         "name" : "cisco-n9k",
         "rules_format" : "cli",
         "ip_address" : "10.1.2.3",
         "device_type" : "switch",
         "manufacturer" : "Cisco"
      },
      "status" : "unmonitored"
   }
]
```


Generate ACLs for Switches

To generate ACLs for a specific switch registered against the NEN, run the following curl command:

```
curl -u api_xxx:xxx -H "Content-Type: application/json" -d {} -X POST
https://mypce.domain.io:8443/api/v2/orgs/1/network_devices/xxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxx/enforcement_instructions_request
```

NOTE:
Replace the `xxx-...-xxx` value with the value of the switch for which you intend to generate ACLs.

Curl Command Using Generate ACLs

```
curl -u api_
1853ebfcb1187acb4:9c2a381773a44e3a609448109278c02c4ec1fe597f9643af71a832c0a8b0c0d0
-H "Content-Type: application/json" -d {} -X POST https://mypce.domain.io:8443 -d
{} -X POST https://mypce.domain.io:8443/api/v2/orgs/1/network_devices/f07a077a-
70ad-4b57-b82a-f1d204fcfd99/enforcement_instructions_request
```

Result: Response with a 202 status code = Accepted.
The ACLs are generated on the PCE and are ready for use in a few seconds.

IMPORTANT:
API POST Requirements

While sending a POST request, you must include the header (`-H`) flag and the data (`-d`) flag. Even if you do not have any data to send, you must insert an empty data flag, as shown in the above example.

NOTE:
Illumio recommends that you insert a pause in any script to allow the PCE to generate the new ACLs for the specific switch. It takes approximately 30 seconds to generate all the ACLs.

The PCE will not send any update or acknowledgment to the REST client once it is finished generating the new ACLs for the switch.

Alternatively, you might want to generate ACLs for all the switches in your inventory to deliver them to your network team, by using the `all_devices: true` key-value pair in your JSON payload while sending the POST request.

To generate ACLs for all the switches registered against the NEN, run the following curl command:

```
curl -u api_xxx:xxx -H "Content-Type: application/json" -d '{"all_devices": true}'
-X POST https://mypce.domain.io:8443/api/v2/orgs/1/network_devices/multi_
enforcement_instructions_request
```

Get List of ACLs

To download ACLs for a specific switch registered against the NEN, get the updated value of the `enforcement_instructions_data_href` key. This value keeps changing because each time the PCE generates new ACLs for a switch, it is considered to be a new datafile.

1. To get the updated `enforcement_instructions_data_href` value for a network switch, run the following curl command:

   ```
   curl -u api_xxx:xxx -H "Accept: application/json" -X
   GET https://mypce.domain.io:8443/api/v2/orgs/1/network_devices/enforcement_
   instructions_data_href'
   ```

   The above command returns a list of switches. You have to then parse the JSON output and filter on the `enforcement_instructions_data_href` key to get the updated value. You can use the JQ tool to filter outputs on any JSON file.

2. After you retrieve the updated value, use it in the following curl command to get the generated ACLs:

   ```
   curl -u api_xxx:xxx -H "Accept: application/json" -X
   GET https://mypce.domain.io:8443/api/v2/orgs/1/datafiles/xxxxxx-xxxx-xxxx-
   xxxx-xxxxxxxxxxx
   ```

Replace the `xxx-...-xxx` value with the value of the `enforcement_instructions_data_href` key that you got by running the previous GET request.

Example of Get List of ACLs

```
curl -u api_
1853ebfcb1187acb4:9c2a381773a44e3a609448109278c02c4ec1fe597f9643af71a832c0a8b0c0d0
-H "Accept: application/json" -X GET
https://mypce.domain.io:8443/api/v2/orgs/1/network_devices/enforcement_
instructions_data_href' "/orgs/1/datafiles/d1bdbb23-60c4-439e-bd74-ca0d03d959a7"
```

Output:

```
no ip access-list ILLUMIO_ACLS-Ethernet1-21-Inbound
p access-list ILLUMIO_ACLS-Ethernet1-21-Inbound
!---Inbound ACL Rules ---
       permit ip host 10.10.100.201 host 10.10.100.202
       permit ip host 10.10.100.201 host 10.10.100.203
       permit ip host 10.10.100.201 host 10.10.100.204
       permit tcp any any established
       permit udp any eq 68 any eq 67
       permit udp any range 1024 65535 any eq 53
       exit
                        ...
no ip access-list ILLUMIO_ACLS-VLAN-20-Outbound
ip access-list ILLUMIO_ACLS-VLAN-20-Outbound
!---Outbound ACL Rules ---
       permit ip host 10.10.100.201 host 10.10.100.204
       permit ip host 10.10.100.202 host 10.10.100.204
       permit ip host 10.10.100.203 host 10.10.100.204
       permit tcp any any established
       permit udp any eq 67 any eq 68
       permit udp any eq 53 any range 1024 65535
       exit
```

Load Balancers and Virtual Servers for the NEN

Illumio Core supports activation of enforcement on F5 BIG-IP Local Traffic Manager (LTM), BIG-IP Advanced Firewall Manager (AFM), and AVI Vantage systems.

> IMPORTANT:
> From Illumio Core 19.3.0 release onwards, the Network Function Controller (NFC) is no longer supported. The F5 interface has been moved from the PCE in to the Network Enforcement Node (NEN).

Since the NFC has been discontinued, you need the NEN to interface with Load Balancers. The NEN has both switch and load balancer capabilities.

Concepts

- **Load balancer (SLB)**: Is either a physical machine or a virtual machine performing load balancing functions. An SLB object represents a standalone device or an HA Pair and includes management of IP/port, user-/password, and so on. These values are used by an Illumio NEN to read information from and manage the device. In case of HA, it may include multiple SLB devices.
- **Illumio Virtual Server**: Is the same as an F5 Virtual Server.
- **Discovered Virtual Server**: An Illumio NEN queries the load balancer for VIPs and specifies the client-facing VIP with port + protocol combination.
- **Created Virtual Server**: Is a provisionable policy object with labels used in policy writing. In the UI, the Virtual Server creation process is called VIP Management. Virtual Server providers (backend servers) are specified using labels and can optionally specify backend port independently of the port used by the VIP.
- **VIP**: Is a virtual IP or a local IP (a front-end IP that clients can connect to).
- **SNAT pool**: Is a group of IPs that the Virtual Servers use to connect to the backend servers. A Virtual Server can only have a single VIP connected to it, on a single port. It can also be accessed by the SLBs local IPs.

Load Balancers

Illumio Coresupports activation of enforcement on F5 BIG-IP Local Traffic Manager (LTM), BIG-IP Advanced Firewall Manager (AFM), and AVI Vantage systems.

> IMPORTANT:
> From the Illumio Core 19.3.0 release onwards, the Network Function Controller (NFC) is no longer supported. The F5 interface has been moved from the PCE in to the Network Enforcement Node (NEN).

Since the NFC has been discontinued, you need the NEN to interface with Load Balancers. The NEN has both switch and load balancer capabilities.

By applying labels to your load balancer's virtual servers, you can write rules that allows client workloads in front of the load balancer to communicate with the virtual IP address of the load balancer's virtual servers. By adding labels to the pool members behind a virtual sever, you can allow communication from the load balancer to the members of the pool. The source for this communication is determined by the load balancer. The Illumio Core programs policies on the load balancer to enforce security policy. The PCE uses the load balancer's REST APIs to read and write security policies to configure security rules.

The PCE supports configuration of two load balancer units if they are configured in Active/Standby or Active/Active modes. The PCE needs to be configured with the user name and password of an administrative user who has read-write access to all configurations on the load balancer.

The PCE configures new objects on the load balancer and does not alter any existing configurations. When an Illumio-created object in the load balancer configuration is modified, the PCE detects it as tampering and modifies the configuration back to the intended state so that the correct security policy is enforced.

The Illumio Core dynamically adjusts policies on the load balancer based on application and topology changes in the datacenter so that the Illumio Core

can enforce consistent security policy on load balancers across the datacenter and cloud environments, as well as show the application traffic in Illumination. The Illumio Core keeps track of the policy it programmed and reconfigures policy if it was altered on the load balancer manually or by other means.

The Illumio Core makes use of the following constructs on load balancers:

- **LTM:** iRules on LTM provide capability to restrict application access. When LTM is used as enforcement mechanism, the Illumio Core uses virtual-server based iRules and Datagroup Lists.
- **AFM:** AFM provides stateful firewalling on BIG-IP. When AFM is used as an enforcement mechanism, the Illumio Core uses Network Firewall policies in the virtual server section and address-lists in the network firewall.
- **AVI**: The Illumio Core uses the Network Security Policy rules to program AVI Vantage.

NOTE:
Configuring two F5 units in Active/Standby mode is supported. However, F5 vCMP or F5 clustering is not supported.

F5 BIG-IP Requirements

The Illumio Core uses its REST API to program F5 load balancers, which means that F5 needs to be running a software version that supports REST-API. The requirements include:

- BIG-IP 11.5.x or higher
- Utilize SNAT or Auto-map mode

AVI Vantage Requirements

- AVI Vantage 18.2.3 or higher

Configure Load Balancers

You can add a load balancer using the PCE web console. However, before you add a load balancer, you need to pair the NEN with the load balancer

functionality enabled with the PCE.

NOTE:
A load balancer does not need to be provisioned to work. However, the virtual servers you associate with this load balancer do need to be provisioned.

1. From the PCE web console menu, choose **Infrastructure** > **Load Balancers**.
2. Click **Add**.
3. Specify a name for the load balancer and provide a description.
4. From *Device Type*, select appropriate load balancer device type.
5. From number of devices, select **(1) Standard** or **(2) HA Pair**.

   The load balancer details are displayed.
6. Specify the following settings to enable the PCE to connect to the load balancer:
   - Management IP address or FQDN of the load balancer
   - Port on which to connect
   - Username
   - Password
7. Select **verify TLS** to verify the trust of the TLS certificate provided by the load balancer before connecting to it.
8. Click **Save**.

About Virtual Servers

Virtual servers in the Illumio Core contain two parts:

- A virtual IP address (VIP) and port through which the service is exposed
- Local IP address(es) used to communicate with backend servers (pool members).

A virtual server is similar to a workload. It can be assigned labels and has IP addresses, but does not report traffic to the Illumio Core. Each virtual server has only one VIP. The local IP addresses are used as a source IP address for

connections to the pool members (backend servers) when the virtual server is operating in SNAT mode or Auto mode. These IP addresses are likely to be shared by multiple virtual servers on the server load balancer.

A virtual server is identified by a set of labels. The consumers and providers for a virtual server can be assigned different labels, which could place them in the same group or a different group in Illumination. See **Groups in Illumination** in the *Visualization Guide* for information.

Providers are allowed to have an incomplete label set (for example, only a Location label), so the providers can be in all groups within the specified location. As a result, a single virtual server can have providers in any group or in any number of groups in Illumination.

Virtual Server Members and Labels

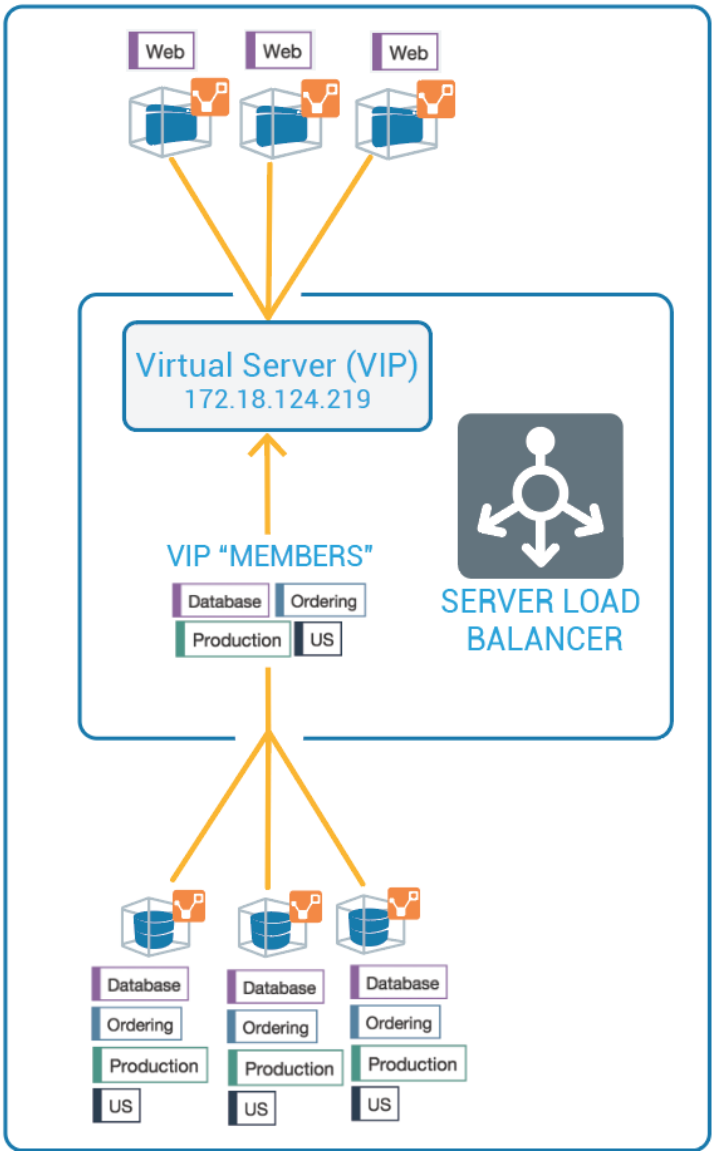
The Illumio Core allows you to write rules to allow communication with workloads managed by a load balancer using labels.

Virtual Server Members

When you configure load balancers in the PCE, it connects to the load balancer using the Illumio Core REST API. The PCE reads all the load balancer virtual servers configurations and populates the Discovered Virtual Servers tab of a load balancer's details page. Any virtual servers associated with the load balancer can be converted to a managed virtual server for use with the PCE. When you configure the virtual server in the PCE web console, you can apply labels to the virtual servers. After configuring a virtual server, you can write a rule that allows other clients to communicate with it.

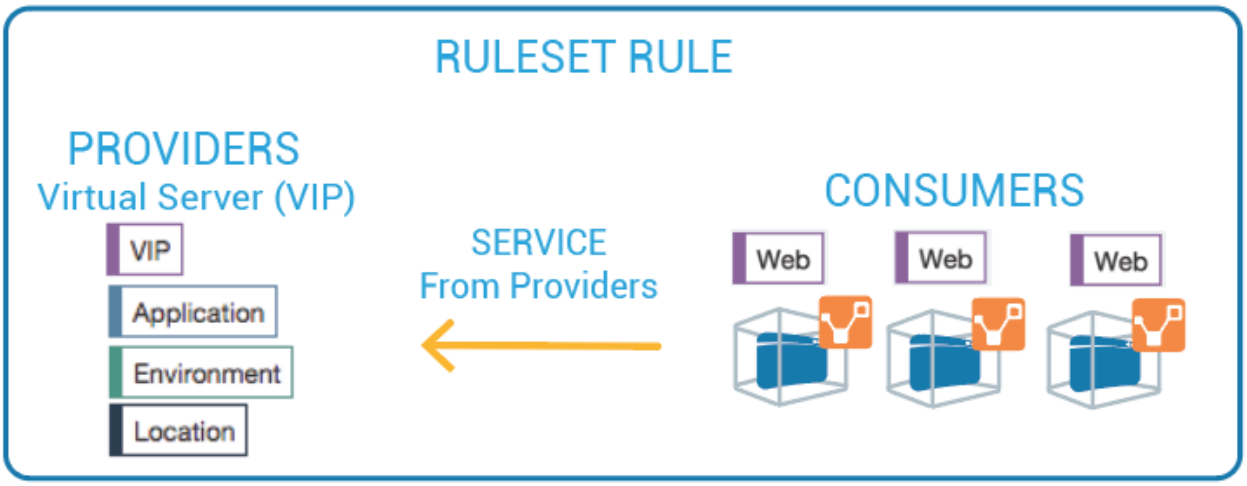
The members behind a virtual server are specified by configuring a set of labels in the virtual server configuration. A set of four Illumio labels can be applied on the Virtual Server Members tab, which is used to match the same set of labels on workloads in the virtual server's pool. If any of the workloads in the virtual server pool share the same set of four labels specified under the Virtual Server Members tab, then any rule you write with the virtual server also applies to the workload members.

In this diagram, you can see how the workloads that belong to the virtual server pool have the same labels specified on the Virtual Server Members tab:



Ruleset Rule for Virtual Server

This diagram illustrates the rule you can write after you label a virtual server and its members:



Configure Virtual Servers

After adding a load balancer to the PCE, you can manage its virtual servers. For each virtual server, you can add a complete set of the four Illumio labels: Role, Application, Environment, and Location. Adding labels to the virtual server enables you to add it in a rule.

You add the four Illumio labels to the Virtual Server's Members tab. When the labels specified under Virtual Server Members match labels of workloads in the virtual server pool, any rule you write with the virtual server are applied to the workload members.

Configuring a load balancer's virtual servers consists of these three settings:

- **Enforced or Not Enforced:** When you select Enforced, any rules you write using the labels associated with the virtual servers and any of its members are enacted. Selecting Not Enforced disables the labels and any policy written that affects the virtual server or its members is disabled.
- **Service:** Select the service to use for the rules that allow access to the virtual server. For example, HTTPD 80 TCP.
- **Labels:** You must apply one each of the four Illumio labels to the virtual server: Role, Application, Environment, and Location. Assigning labels enables the virtual server to be used in rules.

NOTE:
Virtual servers are considered a security policy item, so any changes to a virtual server configuration must be provisioned before any of those changes take effect and become active.

Virtual Server Limitations

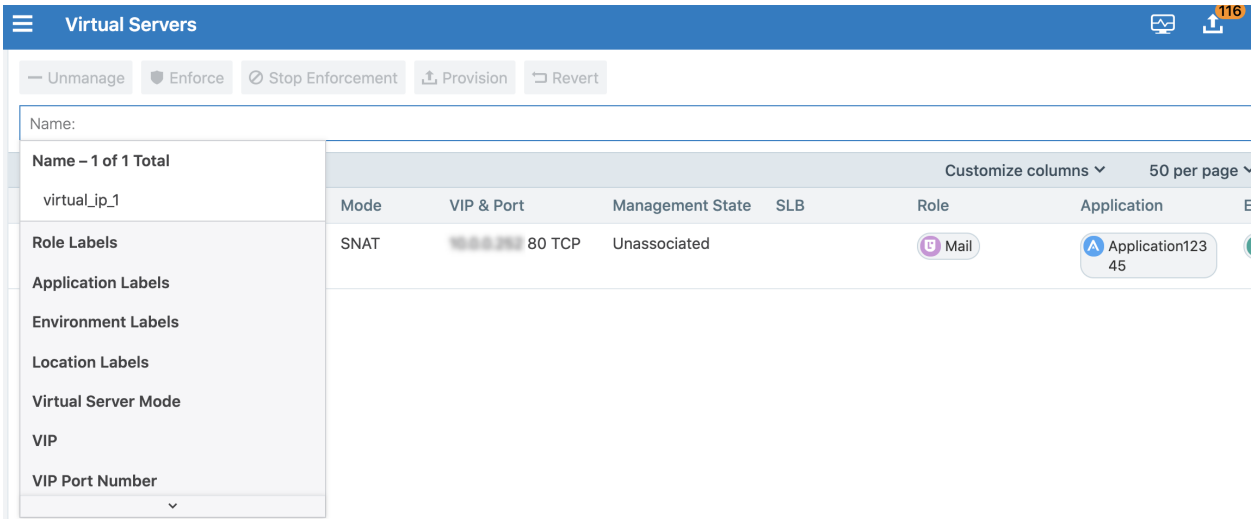
- Illumination does not support location-level and application-level maps.
- If a single SNAT pool is shared between multiple virtual servers, the Illumination map does not render correctly.
- SNAT and Auto-map modes of F5 virtual servers are supported. Transparent mode is not supported.

NOTE:
Before any virtual server configuration can go into effect, you need to provision your changes. See **Provisioning** in the *Security Policy Guide* for information.

Filter the Virtual Server List

You can filter the Virtual Servers list by using the properties filter at the top of the list. For example, you can filter and search by label. You can also filter and search by the following objects:

- Virtual server mode
- Virtual IP address, the VIP port number, or VIP Protocol
- Server Load Balancer



Configure a Load Balancer's Virtual Servers

1. From the PCE web console menu, choose **Infrastructure** > **Load Balancers**.
2. Select the load balancer for which you want to configure virtual servers.
3. Select the **Virtual Servers** tab.
4. Select one of the load balancer's virtual servers and click **Manage**.
5. Select one of the virtual servers and click **Edit**.
6. Enter a name and description for the virtual server.
7. To enable the virtual server's policy, select **Enforced**.
8. Select a service to associate with the virtual server. The service selected enables that service to be used in rules you write for this virtual server.
9. Select one each of the four labels to assign to the virtual server.
10. Click **Save**.
11. Before any virtual servers can go into effect, they must be provisioned.

Chapter 2

Security Policy and the NEN

This chapter contains the following topics:

This section describes how to create security policy and apply those policies on the switches for use with the NEN.

Create and Apply Policy with NEN

To apply the security policy, you need to:

1. Create the policy and generate ACLs for loading onto the switch.
2. Load the generated ACLs onto the switch.
3. Inform the PCE that the ACLs have been loaded onto the switch.

Create Security Policy

In the PCE web console, create label-based policies for the workloads that are bound to the switch ports. For information on how to create policies, see the *Security Policy Guide*.

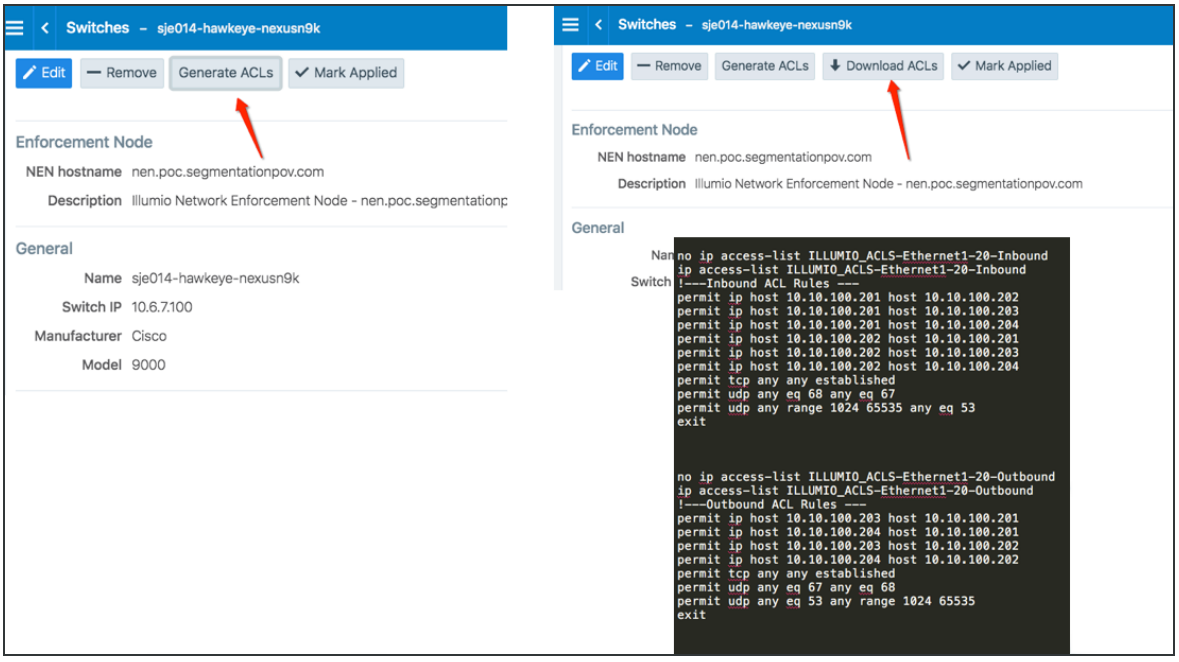
NOTE:
Make sure you provision the policies before generating the ACLs.

Generate and Download ACLs

After you have created new policies for the NEN-managed workloads and provisioned them in the PCE, you can generate the ACLs. The PCE writes those ACLs to its local files.

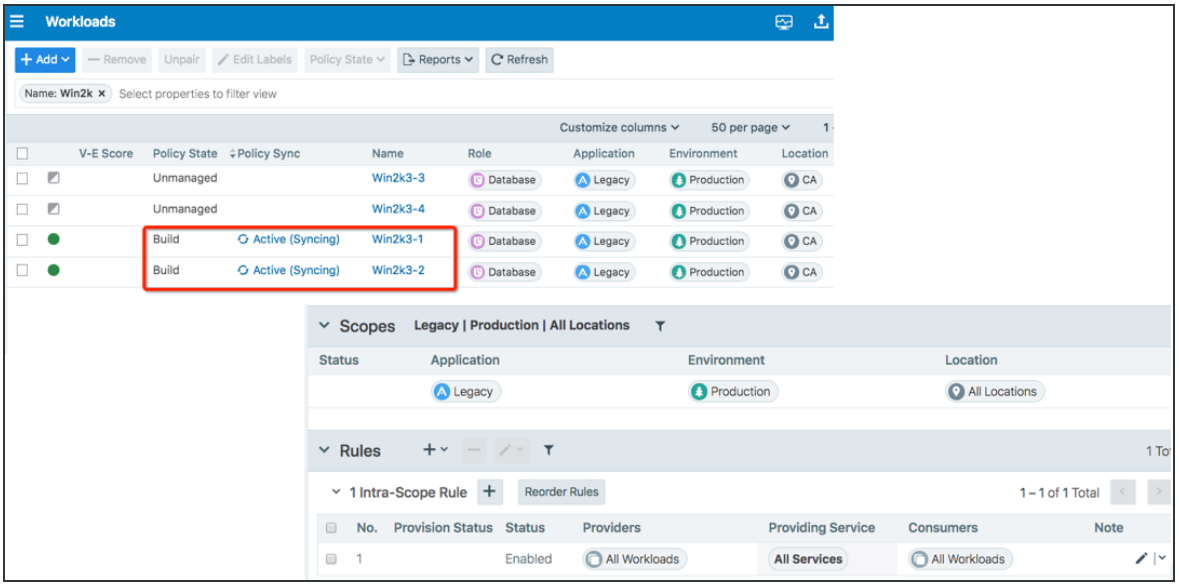
Create the associated switch ACLs as follows:

1. Log into the PCE web console.
2. From the PCE web console menu, choose **Infrastructure** > **Switches**.
3. Select the switch.
4. On the Switches page, click **Generate ACLs**. It takes a moment for the ACL generation to complete.
5. After the ACLs have been generated, click **Download ACLs** to download a .txt file of the updated ACLs from your web browser.
6. Go to the Downloads folder on your computer and open the .txt file. This file contains a list of inbound and outbound ACLs.



The workloads associated with the switch remain in an **Active (Syncing)** policy sync state until you click the **Mark Applied** button on the Switches page.

NOTE:
You might see a “Syncing” notification appear in the PCE web console until you mark the ACLs as **Applied**.



Apply ACLs on the Switch

After generating the ACLs from the NEN, you can copy the text of the ACLs from the PCE-generated files and paste them into the switch's command line to configure the ACLs on the switch. Each interface per direction requires a separate text file.

Example of Inbound and Outbound ACLs

The following ACLs are generated for only Ethernet 1/7 interface of a Cisco Nexus 9000:

```
no ip access-list ILLUMIO_ACLS-Eth1-7-Inbound
ip access-list ILLUMIO_ACLS-Eth1-7-Inbound
!---Inbound ACL Rules ---
permit tcp any any established
permit udp any eq 68 any eq 67
permit udp any range 1024 65535 any eq 53
exit
```

```
no ip access-list ILLUMIO_ACLS-Eth1-7-Outbound
ip access-list ILLUMIO_ACLS-Eth1-7-Outbound
!---Outbound ACL Rules ---
permit tcp host 10.6.4.94 host 10.0.17.17 eq 5432
permit tcp any any established
permit udp any eq 67 any eq 68
permit udp any eq 53 any range 1024 65535
exit
```

NOTE:
By default, the NEN generates ACLs to allow basic infrastructure services such as DHCP and DNS from all IP addresses in addition to the policies defined on the PCE. You cannot prevent DNS and DHCP ACLs from being generated by the NEN. If your network administrator does not want DHCP or DNS rules added to the switch, you can remove those ACL lines while copying the ACLs over to the switch.

To configure the ACLs on to the switch, the network administrator must log into the Cisco Nexus 9000 command line and go into configuration mode. Once in configuration mode, the ACLs can be copied from the NEN in to the switch command line as shown in this example:

```
nexus9000# conf
Enter configuration commands, one per line. End with CNTL/Z.
nexus9000(config)# no ip access-list ILLUMIO_ACLS-Eth1-7-Inbound
nexus9000(config)# ip access-list ILLUMIO_ACLS-Eth1-7-Inbound
nexus9000(config-acl)# !---Inbound ACL Rules ---
nexus9000(config-acl)# permit icmp host 10.0.17.17 0.0.0.0/0
nexus9000(config-acl)# permit tcp any any established
nexus9000(config-acl)# permit udp any eq 68 any eq 67
nexus9000(config-acl)# permit udp any range 1024 65535 any eq 53
nexus9000(config-acl)# exit
nexus9000(config)#
nexus9000(config)#
```

```
nexus9000(config)#
nexus9000(config)# no ip access-list ILLUMIO_ACLS-Eth1-7-Outbound
nexus9000(config)# ip access-list ILLUMIO_ACLS-Eth1-7-Outbound
nexus9000(config-acl)# !---Outbound ACL Rules ---
nexus9000(config-acl)# permit tcp host 10.6.4.94 host 10.0.17.17 eq 5432
nexus9000(config-acl)# permit tcp 0.0.0.0/0 host 10.0.17.17 eq 22
nexus9000(config-acl)# permit tcp any any established
nexus9000(config-acl)# permit udp any eq 67 any eq 68
nexus9000(config-acl)# permit udp any eq 53 any range 1024 65535
nexus9000(config-acl)# exit
```

After the ACLs have been added to the switch, they must be applied to an interface on the switch as either a PACL or a RACL. To take advantage of both inbound and outbound ACLs, this example uses RACLs. If the workload is directly attached to a Layer 2 interface (switchport mode access), you must apply the RACL to the VLAN/SVI interface. If the workload is directly attached to a Layer 3 interface (no switchport), the ACL can be directly applied to the physical interface (or port).

Optional Command to Verify the Interface Configuration

```
# show run interface ethernet x/y
```

Example of Command Usage for Ethernet 1/7

```
nexus9000(config)# show run int eth1/7

!Command: show running-config interface Ethernet1/7
!Time: Tue Oct 16 17:32:52 2018

version 7.0(3)I5(1)

interface Ethernet1/7
switchport
switchport access vlan 17
no shutdown
```

This interface is a Layer 2 interface and is part of VLAN 17. You must apply this to VLAN 17 interface (SVI).

NOTE:
For L2 interfaces, the VLAN must have a Switch Virtual Interface (SVI).

```
nexus9000(config)# interface vlan 17
nexus9000(config-if)# ip access-group ILLUMIO_ACLS-Eth1-7-Inbound in
nexus9000(config-if)# ip access-group ILLUMIO_ACLS-Eth1-7-Outbound out
```

Result: The ACLs are now configured on the switch and any communication through VLAN 17 will be denied if it is not permitted in the ACLs.

Mark ACLs as Applied

You have to inform the PCE after you have loaded the ACLs because the NEN service does not configure the generated ACLs on the switch.

1. Log into the PCE web console.
2. From the PCE web console menu, choose **Infrastructure** > **Switches**.
3. Select the switch.
4. On the Switch page, click **Mark Applied**.

Write SLB Policy

Writing a policy for a load balancer is similar to writing a policy for a workload, except for the following differences:

- Leave the service as unspecified and it will be automatically determined by the port and protocol of the discovered VIP.
- Specify “Uses Virtual Services” in the rule.

A rule that is provided between a virtual server (or its labels) and a set of consumers implicitly programs two sets of rules:

- Rules between the consuming workloads or labels and the frontend VIP of the F5 on the discovered VIP port and protocol: Traffic flows between consuming workloads and the VIP are enforced on both ends if the virtual server is managed and enforced.
- Rules between the F5 pool and the virtual server providers on the service specified in the virtual server object (usually All Services): These rules are enforced for inbound traffic to the virtual server provider if the virtual server provider workloads are enforced.

SLB Methods

The SLB APIs are used to enable automation for F5 policy management.

Functionality	HTTP	URI
Get the list of SLBs	`GET`	`[api_version][org_href]/slbs`
Get a specified SLB	`GET`	`[api_version][org_href]/slb-s/:uuid`
Create an SLB object	`POST`	`[api_version][org_href]/slbs`

SLB Parameters

The parameters for the SLB methods are:

Parameter	Description	Type
`name`	The short friendly name of the server load balancer	String
`nfc`	Network Function Controller managing this SLB	String
`device_type`	Device type of the server load balancer	String
`devices`	Configuration and runtime state of the devices backing this SLB Network VF.	String

Configure an SLB Object

To configure an SLB object:

- **Step 1. Create an SLB object and instruct NEN to sync with it.**

```
POST /api/v2/orgs/{org id}/slbs
```

```
{

        "devices" : [

        {

                "config" : {

                        "username" : "admin",

                        "port" : 443,

                        "credential" : "admin",           # never replayed in northb
API

                        "host" : "10.2.32.6",

                        "credential_type" : "password",

                        "check_certificate" : false

                }

        }

],

        "device_type" :“F5 Big-IP LTM”

        "name" : "Illumio Test SLB"

}
```

- Step 2. GET an SLB response.

```
GET /orgs/{org id}/slbs/{UUID of SLB object
```

```
{

       "name" : "Illumio Test SLB",

       "devices" : [

       {

                "status" : {"connection_state" : "pending"},    # will become
successful when NEN syncs w/ device

                "href" : "/orgs/1/slb_devices/9349ff36-ab38-42bf-909a-
eb5aa3baf187",

                "config" : {

                "username" : "admin",

                "check_certificate" : false,

                "credential_type" : "password",

                "host" : "10.2.32.6",

                "credential" : null,

                "port" : 443

       }

}

],

       "href" : "/orgs/1/slbs/8a82a1b0-c2ce-43ec-abf7-77bd8a3fd22c",
```

```
        "device_type" : "f5_bigip_afm"

        [ ... ]  # created_at, updated_at, etc.

}
```

Step 3. GET a list of Discovered Virtual Servers.

```
GET /orgs/1/discovered_virtual_servers
```

```
{

        "snat_type" : "snat_pool",

        "dvs_identifier" : "d3b784c2fd24ad364c5adb3319169bd2",

        "mode" : "snat",

        "vip_port" : {"port" : 8080, "protocol" : 6, "vip" : "172.16.27.88" },

        "service_checks" : [{"protocol" : 1}],

        "name" : "Common/QL_VIP_1",

        "slb" : {

                "href" : "/orgs/1/slbs/8a82a1b0-c2ce-43ec-abf7-77bd8a3fd22c"

        },

        "snat_pool_ips" : ["172.16.26.27", "172.16.26.18", "172.16.27.18"],

        "local_ips" : ["172.16.26.18", "172.16.27.18"],

        "href" : "/orgs/1/discovered_virtual_servers/2c460b98-2176-4a44-9ba4-
e77f3eacd0f1"
```

```
        [ ... ]  # created_at, updated_at, etc.

}
```

Step 4. Manage a VIP by creating a Virtual Server object.

```
POST /orgs/1/sec_policy/draft/virtual_servers
```

```
{

        "name" : "Common/chris-VIP1",

        "service" : {

                "href" : "/orgs/1/sec_policy/draft/services/1"

        },

        "labels" : [],

        "providers" : [],

        "mode" : "unmanaged",  # enforced

        "discovered_virtual_server" : {

                "href" : "/orgs/1/discovered_virtual_servers/23338ceb-7580-466a-bbcf-
a645b82ce97b"

        }

}
```

Step 5. Modify the enforcement mode, labels, and backend/provider labels of the Virtual Server.

```
PUT /orgs/1/sec_policy/draft/virtual_servers/84bae9dd-f1f6-4322-bffc-
f07354b0622a
```

```
{

       "mode” : "enforced",

       "labels” : [{"href” : "/orgs/1/labels/448"}, {"href” :
"/orgs/1/labels/444"}],  # any RAEL tuple

       "providers” : [{"label":{"href":"/orgs/1/labels/449"}}]   # note:
providers may have different labels

}
```

Step 6. Provision the Virtual Server into an active policy.

```
POST /orgs/1/sec_policy
```

```
{

       "update_description” : "Provision my first VS",

       "change_subset” : {

              "virtual_servers” : [{"href” : "/orgs/1/sec_policy/draft/virtual_
servers/84bae9dd-f1f6-4322-bffc-f07354b0622a"}]

       }

}

/orgs/1/sec_policy/draft/virtual_servers/84bae9dd-f1f6-4322-bffc-
f07354b0622a
```

```
/orgs/1/sec_policy/active/virtual_servers/84bae9dd-f1f6-4322-bffc-
f07354b0622a
```

Step 7. Write rules that apply to the Virtual Server.

```
POST /orgs/1/sec_policy/draft/rule_sets/1480/sec_rules
```

```
{

        "enabled" : true,

        "providers" : [

                {"label" : {"href" : "/orgs/1/labels/444"}},

                {"label" : {"href" : "/orgs/1/labels/448"}}

        ],

        "resolve_labels_as" : {

                "consumers" : ["workloads"],

                "providers" : ["virtual_services"]    # NOTE: Must be virtual_services

        },

        "consumers" : [

                {"actors" : "ams"}  # All Workloads

        ]

}
{
```

```
        "consumers" : [

                {"label" : {"href" : "/orgs/1/labels/444"}}

        ],

        "providers" : [

        {

                "virtual_server" :

                {"href" : "/orgs/1/sec_policy/draft/virtual_servers/84bae9dd-f1f6-4322-
bffc-f07354b0622a"}

        }

],

"enabled" : true,

"resolve_labels_as" : {

"consumers" : ["workloads"],

"providers" : ["virtual_services"]

        }
```

Remove Filtering

Some types of virtual servers are not visible, such as those without default server pools. From the NEN 2.1.0 release onwards, you can do filtering related to such virtual servers. You can see VIPs that do not have a pool associated with them, use UDP, or are not SNAT/Auto-SNAT.

To view all types of virtual servers configured on the F5 load balancers, you need to enter specific commands during the NEN installation (on a NEN by NEN basis). These commands will disable (enabled by default) the built-in filter running on the NEN on the Leader PCE cluster.

1. Navigate to the directory:

```
/opt/illumio-pce/
```

2. Enter:

```
sudo -su ilo-pce ./illumio-nen-ctl slb-enable --virtual-server-filtering
disabled
```

3. Restart the NEN on both db0 and db1 nodes:

```
sudo -u ilo-pce ./illumio-pce-ctl restart
```