OpFlex support for NetFlow with OVS

New and Changed Information

The following table provides an overview of the significant changes to this guide up to this current release. The table does not provide an exhaustive list of all changes that are made to the guide or of the new features up to this release.

Table 1: New Features and Changed Behavior

Cisco APIC Release Version	Feature	Description
5.1(3)	OpFlex support for NetFlow with OVS	This guide became available.
5.2(1)	Change in CRD parameter	The `type` key has been changed to `flowType`. See Configuring OpFlex support for NetFlow with OVS on Kubernetes Setup, on page 4 section.

About OpFlex support for NetFlow with OVS

The OpFlex support for NetFlow with OVS feature enables the ability to sample traffic from compute nodes and analyze it through network analyzers.

Benefits of OpFlex support for NetFlow with OVS

The OpFlex support for NetFlow with OVS provides several benefits:

- The ability to get flow data directly from compute nodes providing visibility of local traffic.

- Easier network troubleshooting and security analysis.

OpFlex support for NetFlow with OVS Limitations and Restrictions

Be aware of the following issues when configuring OpFlex support for NetFlow with OVS:

- The current implementation enables NetFlow for all compute nodes in the VMM domain. There is no support for choosing a subset of compute nodes to export flow information from.

- There is no standard way to enable NetFlow in OpenStack setups. We recommend a new approach that requires the use of AIM to enable this.

- VMM Domain cannot be associated with more than one NetFlow VMM Exporter Policy.

Prerequisites for Configuring OpFlex support for NetFlow with OVS

You must complete the following tasks before you configure OpFlex support for NetFlow with OVS:

- You must have Cisco ACI release 5.1 or later and either have the Cisco ACI CNI plug-in or Cisco ACI ML2 plug-in installed.

• You must have Cisco ACI-CNI release 5.1 or later installed.

OpFlex support for NetFlow with OVS Configuration Workflow

This section describes a high-level overview of the tasks you perform to configure OpFlex support for NetFlow with OVS.

Procedure

Step 1   To configure OpFlex support for Netflow with OVS:

On OpenStack:

On Kubernetes:

Step 2   To verify that OpFlex is configured correctly for the Netflow OVS:

On OpenStack:

On Kubernetes:

Configuring OpFlex support for NetFlow with OVS on OpenStack Setup

This section describes how to configure OpFlex support for NetFlow with OVS on OpenStack setup.

The aimctl CLI tool must be run from the "ciscoaci_aim" docker container which lives in the OpenStack controller node.

In the case of multiple controllers, running the aimctl command on any one of the controllers to configure netflow is sufficient. All the other controllers will receive the WebSocket event to sync up.

Procedure

Step 1   To enter the container, enter the following command:

```
$ docker exec -itu root ciscoaci_aim bash
```

Step 2   You must enter the `aimctl` command to create a netflow session using name, netflow version, destination address and destination port:

```
$ aimctl manager netflow-vmm-exporter-pol-create <NAME> --ver <version> --dst_addr <dest_addr> \
--dst_port <dest_port>
```

Example:

```
$ aimctl manager netflow-vmm-exporter-pol-create test-netflow-session --ver v9 --dst_addr 1.1.1.1 \
--dst_port 2055
```

Step 3   You must enter the `aimctl` command to bind the netflow session to a VMM domain using domain type, domain name, and the netflow path created in step 1.

```
$ aimctl manager vmm-relation-to-exporter-pol-create <domain_type> <domain_name> <netflow_path>
```

Example:

```
$ aimctl manager vmm-relation-to-exporter-pol-create OpenStack osd16-fab \
uni/infra/vmmexporterpol-test-netflow-session
```

Configuring OpFlex support for NetFlow with OVS on Kubernetes Setup

This section describes how to configure OpFlex support for NetFlow with OVS on Kubernetes setup.

Procedure

Step 1   Verify the CRD is available, any faults and violations, enter the following commands:

```
$ kubectl get crd
$ kubectl describe crd netflowpolicies.aci.netflow
```

Step 2   Apply the Custom Resource yaml file with valid inputs, enter the following command:

```
$ kubectl apply -f <yaml_file>
```

Sample Custom Resource YAML file:

```
apiVersion: aci.netflow/v1alpha
kind: NetflowPolicy
metadata:
  name: netflow-policy
spec:
  flowSamplingPolicy:
    destIp: "172.28.184.76"
    destPort: 2055
    flowtype: "ipfix"
```

From Cisco ACI CNI release 5.2(1), the `type` key in the CRD has been changed to `flowType` (as indicated in the following example).

```
apiVersion: aci.netflow/v1alpha
kind: NetflowPolicy
metadata:
  name: netflow-policy
spec:
  flowSamplingPolicy:
    destIp: "172.28.184.76"
    destPort: 2055
    flowType: "ipfix"
```

Verifying that OpFlex is configured correctly for the NetFlow OVS on OpenStack

This section describes how to verify the OpFlex support for NetFlow with OVS on OpenStack setup.

Procedure

Step 1   Log in to the Cisco APIC GUI, on the menu bar, choose **Fabric** > **Access Policies**.

Step 2   You need to confirm that the NetFlow policy created by you with the user inputs (dst_ip, version) has been pushed to the APIC. In the Navigation pane, choose **Policies** > **Interface** > **NetFlow** > **NetFlow Exporters for VM Networking** and click on one of the VMM external collector reachability.

Step 3   Verify the NetFlow policy has been successfully pushed to opflex agent with your given inputs. The dstAddr shown here will be the routable IP where the traffic will be received. Inside the opflex_agent container, enter the following commands:

```
# docker exec -itu root ciscoaci_opflex_agent bash
# gbp_inspect -prq NetflowExporterConfig
---. NetflowExporterConfig,/PolicyUniverse/PlatformConfig/
comp%2fprov-OpenStack%2fctrlr-%5bosd13-fab20%5d-osd13-fab20%2fsw-InsiemeLSOid/
NetflowExporterConfig/<new>
 {
    activeFlowTimeOut : 60
    dscp              : 44
    dstAddr           : 172.28.184.76
    dstPort           : 2055
    name              : <new>
    samplingRate      : 0
    srcAddr           : 0.0.0.0
    version           : 2 (v9)
 }
```

Step 4   The opflex agent uses OpenFlow to configure flows and pushes the NetFlow config to OVSDB. You need to verify the NetFlow policy with dest_IP and port, you have configured has been received by the OVS. On the compute nodes, check for **ovs-vsctl list ipfix**, enter the following commands:

```
$ ssh heat-admin@1.00.1.64
Last login: Thu Dec 3 14:50:02 2020 from 1.100.1.1
$ sudo -s
# ovs-vsctl list ipfix
_uuid :c3645755-5517-4a3e-84ac-8cc110254fa7
active_timeout      : 60
add_id_to_interface : false
engine_id           : []
engine_type         : []
external_ids        : []
targets             : ["172.28.184.76:2055"]
```

Verifying that OpFlex is configured correctly for the NetFlow OVS on Kubernetes

This section describes how to verify the OpFlex support for NetFlow with OVS on Kubernetes setup.

Procedure

Step 1   Log in to the Cisco APIC GUI, on the menu bar, choose **Virtual Networking** > **Kubernetes**.

Step 2   Verify that you see the NetFlow session is created on the APIC. In the Navigation pane, choose **Kubernetes** and click on the domain.

Step 3   Verify that the NetFlow policy has been successfully pushed to opflex agent with your given inputs. The dstAddr shown here will be the routable IP where the traffic is received. Inside the `opflex_agent` container, enter the following commands:

Example:

```
# kubectl exec -it -n aci-containers-system aci-containers-host-7nxfd -c opflex-agent /bin/sh
# gbp_inspect -prq NetflowExporterConfig
---. NetflowExporterConfig,/PolicyUniverse/PlatformConfig/
Comp%2fprov-Kubernetes%2fctrlr-%5bkube%5d-kube%2fsw-InsiemeLSOid/
NetflowExporterConfig/<new>
 {
   activeFlowTimeOut : 60
   dscp             : 44
   dstAddr          : 172.28.184.76
   dstPort          : 2055
   name             : <new>
   samplingRate     : 0
   srcAddr          : 0.0.0.0
   version          : 2 (v9)
 }
```

Step 4   Look up the pods, enter the following command:

```
$ kubectl get pods -A
```

Step 5   SSH into open-vswitch pod, enter the following command:

```
$ kubectl exec -it -n <NAMESPACE> <POD_NAME_HERE> /bin/sh
```

Example:

```
$ kubectl exec -it -n aci-containers-system  aci-containers-openvswitch-l2lxk /bin/sh
```

Step 6   Verify if the OVS has received the NetFlow config. Inside the pod, enter the following command:

```
$ ovs-vsctl list ipfix
```