Enabling the OpFlex Drop Log Feature

Revised: February 22, 2021

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 Drop Log	This guide became available.

About OpFlex Drop Log

The OpFlex Drop Log feature logs any packet that gets dropped in the datapath. It is useful to know what policy dropped a packet while debugging flow drops. Overall policy logging can be useful to understand the policies in a datapath without looking at the configuration.

Currently tools such as ovs-appctl ofproto/trace that are provided by Open vSwitch (which allow tracing of a specific packet through the datapath) are being used to debug drops. The Drop log feature makes it easy to monitor drops at scale.

Policy logging is already available on ACI as an action in addition to permit and deny while specifying filters.

This feature extends the functionality to the compute, while adding policy-miss logging.

Benefits of OpFlex Drop Log

The OpFlex Drop Log provides several benefits:

- Allows logging of all dropped packets in the datapath due to policy miss.
- In Kubernetes, events are published to the corresponding pods, and from there any issue in traffic for which datapath is dropping can be noticed easily.
- If not on Kubernetes, then OpFlex logs will have all the packet drops logged and the IP addresses can be used to map the VMs involved in traffic.
- Support IPv4 (not option processing), IPv6, TCP, UDP, and Geneve with custom TLVs.

OpFlex Drop Log Limitations and Restrictions

Be aware of the following issues when configuring OpFlex Drop Log:

- Permit Logging is not supported.
- Drop action for policy is not available as a CRD in Kubernetes.
- Events are not supported on OpenStack, but OpFlex logs should be available.

Prerequisites for Configuring OpFlex Drop Log

You must complete the following tasks before you configure OpFlex Drop Log:

- You must have basic working knowledge about Cisco ACI CNI deployment or OpenStack ACI plug-in.
- Socket operations requires a minimum version of 1.58 for boost::asio.

OpFlex Drop Log Configuration Workflow

This section describes a high-level overview of the tasks you perform to configure OpFlex Drop Log:

1. You can either configure the OpFlex Drop Log natively or on Kubernetes.

   For more information see Configuring the OpFlex Drop Log Natively, on page 5 or Configuring the OpFlex Drop Log on Kubernetes, on page 3.

2. For OpenStack, configuring the OpFlex Drop Log natively workflow, would have been performed if using the Cisco installer with the exception of dynamically enabling drop log. This can be done by editing the `a.droplogcfg` file to have the following contents `{"drop-log-enable": true}`, instead of `false`.

3. Verify the configuration.

   For more information, see Verifying the OpFlex Drop Log on Non-Kubernetes Environment, on page 6 or Verifying the OpFlex Drop Log on Kubernetes, on page 3.

Configuring the OpFlex Drop Log on Kubernetes

This section describes how to configure the OpFlex Drop Log on Kubernetes.

Procedure

Step 1 In the acc-provision input file the configuration is automatically done by enabling the flag:

Example:

```
drop_log_config:
   enable: true
```

Step 2 In the context of opflex-agent running in Kubernetes, execute the acc-provision input file:

Example:

```
$ acc-provision -i input_file -f <flavor> > aci-containers.yaml
```

Verifying the OpFlex Drop Log on Kubernetes

This section describes how to verify the OpFlex Drop Log on Kubernetes.

Procedure

Check that the following knob is set in the /usr/local/etc/aci-containers/host-agent.conf file.

Example:

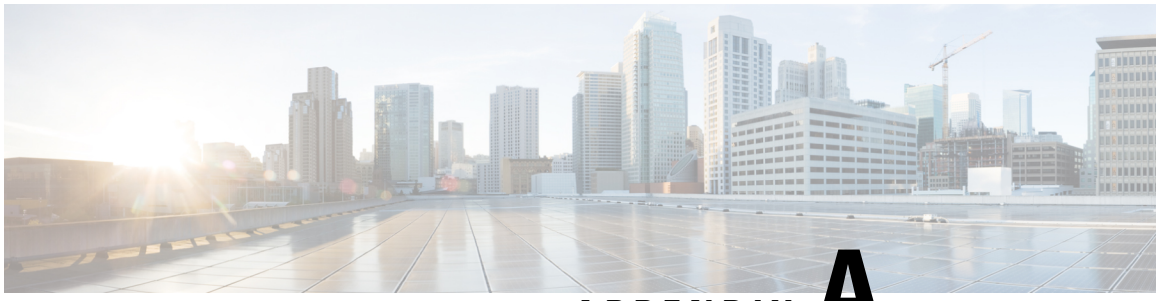
```
"enable-drop-log": true
```

This knob drives all the configuration mentioned previously for OpFlex.

In addition to the logs getting printed under opflex-agent, the event will be logged to the pod in question. If both source and destination pods are present on the same node, only the source pod will have the event. Repeated events to the same pod are rate limited to one every 2 minutes and dropped before publishing if the event could not be published within 10 minutes of the event timestamp.

Example of an event under a pod:

```
noiro@f1-compute-1:~/noiro$ kubectl describe pod busybox
Name:         busybox
Namespace:    default
Priority:     0
Node:         f1-compute-1/1.100.101.11
Start Time:   Wed, 01 Apr 2020 15:21:02 -0400
Labels:       <none>
Annotations:  kubectl.kubernetes.io/last-applied-configuration:
                {"apiVersion":"v1","kind":"Pod","metadata":{"annotations":{"opflex.cisco.com/endpoint-group":"{ \"tenant\":\"kube\", \"app-profile\":\"aci...
              opflex.cisco.com/endpoint-group: { "tenant":"kube", "app-profile":"aci-containers-kube", "name":"aci-containers-drop" }
Status:       Running
IP:           10.2.8.9
IPs:
  IP:  10.2.8.9
Containers:
  busybox:
    Container ID:  docker://b758d754c0df57a8968319d852e9e89deac58a631548c59c03ee534b1f0c3c72
    Image:         busybox
    Image ID:      docker-pullable://busybox@sha256:b26cd013274a657b86e706210ddd5cc1f82f50155791199d29b9e86e935ce135
    Port:          <none>
    Host Port:     <none>
    Command:
      sleep
      3600
    State:          Running
      Started:      Wed, 01 Apr 2020 15:21:04 -0400
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-w7fxf (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  default-token-w7fxf:
    Type:        Secret (a volume populated by a Secret)
    SecretName:  default-token-w7fxf
    Optional:    false
QoS Class:       BestEffort
Node-Selectors:  kubernetes.io/hostname=f1-compute-1
Tolerations:     :NoSchedule
                 node.kubernetes.io/not-ready:NoExecute for 300s
                 node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type     Reason                                 Age    From                  Message
  ----     ------                                 ----   ----                  -------
  Normal   Scheduled                              5m50s  default-scheduler     Successfully assigned default/busybox to f1-compute-1
  Normal   Pulled                                 5m49s  kubelet, f1-compute-1  Container image "busybox" already present on machine
  Normal   Created                                5m48s  kubelet, f1-compute-1  Created container busybox
  Normal   Started                                5m48s  kubelet, f1-compute-1  Started container busybox
  Warning  Int-POL_TABLE(Policy Drop)             4m7s   aci-containers-host    IPv4 packet from 10.2.8.9 to 10.2.8.7 was dropped
  Warning  Int-PORT_SECURITY_TABLE(Security Drop)  18s    aci-containers-host    ARP packet from 10.2.8.9 to 10.2.8.7 was dropped
```



APPENDIX A

Appendix

Configuring the OpFlex Drop Log Natively

This section describes how to configure the OpFlex Drop Log natively on opflex-agent.

The following configuration is for the case where OpFlex is running independently.

Procedure

Step 1 Create Geneve interfaces for both bridges' integration and access, enter the following commands:

Example:

```
$ ovs-vsctl add-port br-int gen1 -- set interface gen1 type=geneve options:remote_ip=flow options:key=1
$ ovs-vsctl set interface gen1 ingress_policing_rate=1000 ovs-vsctl set interface gen1
ingress_policing_burst=100
$ ovs-vsctl add-port br-access gen2 -- set interface gen2 type=geneve options:remote_ip=flow
options:key=2
$ ovs-vsctl set interface gen2 ingress_policing_rate=1000 ovs-vsctl set interface gen2
ingress_policing_burst=100
```

Take note in the opflex-agent configuration block where gen1 and gen2 are as shown below:

Example:

```
"drop-log": {
        // Encapsulate drop-log traffic with GENEVE.
        "geneve" : {
           // The name of the drop-log tunnel integration interface in OVS
           "int-br-iface": "gen1",
           // The name of the drop-log tunnel access interface in OVS
           "access-br-iface": "gen2",
           // Remote IP address that the packet should be sent to.
           "remote-ip" : "192.168.1.2"
        }
    },

    "drop-log-config-sources": {
            // Filesystem path to monitor for drop log control
            // Default: no drop log service
            "filesystem": ["/var/lib/opflex-agent-ovs/droplog"]
    }
```

Step 2 A drop log file needs to be created in the path and noted as the drop-log-config source with a .droplogcfg extension with the following contents:

Example:

```
{
        "drop-log-enable": true
    }
```

This value can be toggled to false to disable drop-log feature which is immediate.

Step 3 Drop-log pruning can be configured by adding the prune filters. The following example shows all the fields supported in pruning and the syntax. More filter entries can be added by appending to the list under the “drop-log-pruning” section.

Example:

```
{
    "drop-log-enable": true,
    “drop-log-pruning”: {
    “filter1”: {
 "name":"filter1","sip":"1.2.3.4","dip":"5.6.7.0/24","smac":
"00:01:02:03:04:05/FF:FF:00:00:00:00","dmac":"06:07:08:09:0A:0B/FF:FF:FF:FF:FF:FF",
"ip_proto":6, "sport":12000,”dport":13000 }
     }
}
```

Step 4 Add an iptables rule on the host to redirect drop-log packets to the listener socket:

Example:

```
$ iptables -t nat -A OUTPUT -p udp --dport 6081 -j DNAT --to 127.0.0.1:50000
```

Step 5 To export packet events to a Unix server socker, note the name of the socket here:

Example:

```
"packet-event-notif": {
        "socket-name": ["/usr/local/var/run/packet-event-notification.sock"]
    }
```

Verifying the OpFlex Drop Log on Non-Kubernetes Environment

This section describes how to verify the OpFlex Drop Log on non-Kubernetes environment.

Procedure

Step 1 Confirm that Geneve interfaces are corresponding to the configuration that has been created in OVSDB. Where opflex-agent is running, enter the following command:

Example:

```
$ ovs-vsctl show
595c1b80-bb13-48d7-8be2-b7276cf8c460
    Bridge br-access
        Port br-access
            Interface br-access
                type: internal
        Port "gen2"
            Interface "gen2"
```

```
            type: geneve
            options: {key="2", remote_ip=flow}
    Bridge br-int
        fail_mode: secure
        Port "gen1"
            Interface "gen1"
                type: geneve
                options: {key="1", remote_ip=flow}
        Port br-int
            Interface br-int
                type: internal
    ovs_version: "2.12.0"
```

Step 2 Confirm that the iptables rule is present on the host:

Example:

```
$ ~/opflex/agent-ovs$ iptables -L -t nat
Chain OUTPUT (policy ACCEPT)
target   prot   opt   source     destination
DOCKER   all    --    anywhere   !127.0.0.0/8   ADDRTYPE match dst-type LOCAL
DNAT     udp    --    anywhere   anywhere       udp dpt:6081 to:127.0.0.1:50000
```

Step 3 Look for the following logs in opflex-agent before any dropped packets appear:

Example:

```
[info] [ovs/IntFlowManager.cpp:277:setDropLog] DropLog port set to gen1 tunnel
destination: 192.168.1.2:6081
[info] [ovs/AccessFlowManager.cpp:170:setDropLog] DropLog port set to gen2 tunnel
destination: 192.168.1.2:6081
[info] [ovs/PacketLogHandler.cpp:110:startListener] PacketLogHandler started!
[info] [./ovs/include/PacketLogHandler.h:127:LocalClient] Packet Event socket set to
/usr/local/var/run/aci-containers-packet-event-notification.sock
[info] [ovs/PacketLogHandler.cpp:32:run] PacketEventExporter started!
```

Step 4 Dynamic enabling of drop-logs is indicated by the following log:

Example:

```
[info] [ovs/IntFlowManager.cpp:407:packetDropLogConfigUpdated] Droplog mode set to unfiltered
```

An example of the logs printed:

```
[info] [ovs/PacketLogHandler.cpp:197:parseLog] Int-PORT_SECURITY_TABLE
MAC=88:1d:fc:f2:fb:59:00:22:bd:f8:19:ff:Qtag QTAG=0 SRC=10.5.8.1 DST=10.5.8.3 LEN=100
DSCP=56 TTL=64 ID=39239 FLAGS=0 FRAG=0 PROTO=ICMP TYPE=8 CODE=0 ID=14769 SEQ=38261
```

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA 95134-1706
USA

Asia Pacific Headquarters
CiscoSystems(USA)Pte.Ltd.
Singapore

Europe Headquarters
CiscoSystemsInternationalBV
Amsterdam,TheNetherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.