opFlow 3 Operations Guide

Operational Status Report

You can get to this from "Views -> Operational Status Report"

High Volume mode sample:

To read a loadCycle: "**Load Time**: 42.73s, **Insertion Time**: 21.08s, **Filter Time**: 0.15s, **Endpoint Time**: 10.41s, Flows: 56075, Conversations: 21246, Skipped Filtered Flows: 53, Unique IPs: 8960"

total processing time = "Load Time + Filter Time + Endpoint Time" (Load time includes Insertion Time, it is broken out to help see DB performance)
total processing time = 42.73 + 0.15 + 10.41
total processing time = 53.29s

Date	Activity	Type	Status	Details	Stats
2016-04-19T11:12:13	loadCycle	completed	ok	Process 27739 done with file(s) nfcapd.201604191109	Load Time: 42.73s, Insertion Time: 21.08s, Filter Time: 0.15s, Endpoint Time: 10.41s, Flows: 56075, Conversations: 21246, Skipped Filtered Flows: 53, Unique IPs: 8960
2016-04-19T11:10:02	loadCycle	completed	ok	Process 27684 done with file(s) nfcapd.201604191107	Load Time: 37.42s, Insertion Time: 18.82s, Filter Time: 0.13s, Endpoint Time: 5.37s, Flows: 48445, Conversations: 18372, Skipped Filtered Flows: 57, Unique IPs: 8472
2016-04-19T11:08:30	loadCycle	completed	ok	Process 27576 done with file(s) nfcapd.201604191105	Load Time: 55.97s, Insertion Time: 33.9s, Filter Time: 0.24s, Endpoint Time: 10.17s, Flows: 56434, Conversations: 21374, Skipped Filtered Flows: 51, Unique IPs: 8788
2016-04-19T11:06:28	loadCycle	completed	ok	Process 27438 done with file(s) nfcapd.201604191103	Load Time: 54.56s, Insertion Time: 34.03s, Filter Time: 0.14s, Endpoint Time: 10.56s, Flows: 55158, Conversations: 19285, Skipped Filtered Flows: 45, Unique IPs: 8975

Low Volume mode example:

To read a loadCycle: "**Load Time**: 223.72s **Summarize Time**: 5.05s Aggregation Time: 3.18s **Filter Time**: 133.78s **Endpoint Time**: 23.65s Flows: 202482 Conversations: 16355 Unique IPs: 5221" the

total processing time = "Load Time + Summarize Time + Filter Time + Endpoint Time" (Summarise Time includes Aggregation Time, it is broken out to see DB performance)
total processing time = 223.72 + 5.05 + 133.78 + 23.65
total processing time = 386.2s

Low Volume mode does not list the insertion time

Also note:
Skipped Empty Flows - flows that were skipped because they had 0 bytes
Skipped Filtered Flows - flows that were skipped because they matched the config setting opflow_drop_endpoints

opflow-cli.pl - Manage opFlow from the CLI

opflow-cli.pl allows you to run setup routines, create reports, manually load flow files and generally run CRUD operations on endpoints/apps/agents/filters.

Run opflow-cli.pl -h to get help:

```
Usage: opflow-cli.exe act=operation [option=A...] [param=X...]
opflow-cli.exe act=setup-db [drop=true] [usepercent=NN] [dryrun=false/true]
opflow-cli.exe act=setup-iana [url=...|file=...]
opflow-cli.exe act=(purge|purge-raw|purge-endpoints|purge-indices)
opflow-cli.exe act=load-flows file=...

opflow-cli.exe act=list-endpoints [searchprop=value...]
value can be regex:...

opflow-cli.exe act=create-endpoint property=value...
opflow-cli.exe act=show-endpoint ip=address
opflow-cli.exe act=update-endpoint ip=address entry.propname=value...
opflow-cli.exe act=delete-endpoint ip=address

opflow-cli.exe act=list-apps [searchprop=value...]
opflow-cli.exe act=(create-app|update-app) number=N protocol=M name=X description=Y
opflow-cli.exe act=delete-app number=N protocol=M

opflow-cli.exe act=list-agents
opflow-cli.exe act=update-agent agent=X [in_if=Y out_if=Z] admin_status=(active|inactive)
opflow-cli.exe act=sync-agent-node-data

opflow-cli.exe act=list-filters [include_inactive=0/1]
opflow-cli.exe act=show-filter name=F
opflow-cli.exe act=disable-filter name=F
opflow-cli.exe act=create-filter name=F [description=D] propA=X,Y,Z [propB=...]
properties: agent application endpoint proto src_ip dst_ip src_port dst_port
filter expressions: A,B,C for list of choices, regex:string supported
for all properties except agent and proto.
proto must be number or list of numbers. agent must be agent_ip
(for all interfaces), or agent_ip:in_ifidx:out_ifidx.

opflow-cli.exe act=create-report [param=...]

options:
quiet=1: suppress output, only set exit code
debug=1: more verbose debugging output
```

Manage Agents

Agents are a list of the IP addresses from which flows are received. The System -> Manage Agents GUI function enables you to see each agent IP address as well as the node that has been associated with it . The association to an NMIS node enables opFlow to look up the interface indexes which the traffic is flowing to and from, these associations are automatically updated once an hour and can also be triggered at any time by using the System -> Sync Agent/Node data GUI function. Please refer to: opCharts/NMIS integration for information on configuring opFlow to connect to opCharts.

opFlow allows to select which flow agents (and in/out interfaces) your opFlow instance should accept data from, this is currently done in the opflow-cli.pl program only.

You can use `opflow-cli.pl` to view the list of agents and interfaces, and set any of them inactive or active. If an agent and in/out interface combination is set inactive, then opFlow will ignore flows from this agent and involving these in/out interfaces. There is also a "wildcard" agent record available: if you set that to inactive, then it **overrides** all interface-specific settings and no flows will be accepted from this agent (no matter what interfaces are involved). Please note that `opflow-cli.pl` does show the invidual agent+interface records even if the wildcard record is set to inactive and thus is overriding them.

If you only want to disable flows coming in that match a particular in and out interface combination, then you should set that specific record inactive and leave the wildcard record active.

License Count

Our development team has provided the following query that will produce the nodes/interfaces that are being used to calculate interface usage.

```
mongo -u opUserRW -p op42flow42 --host 127.0.0.1 flows --eval 'db.getCollection("agents").aggregate([  {
"$match" : {"in_if" : { "$ne" : "*" },"out_if" : { "$ne" : "*" }, "admin_status" : "active"   }},{ "$group" :
{_id : "$agent","in_if" : { "$addToSet" : "$in_if" },"out_if" : { "$addToSet" : "$out_if"}}}, { "$project" : {
_id : 1,interfaces : { "$setUnion" : [ "$in_if","$out_if" ] }}} ]);'
```

This command may be used from the command line. From our test machine it results in the following:

```
[root@opmantek bin]# mongo -u opUserRW -p op42flow42 --host 127.0.0.1 flows --eval 'db.getCollection("agents").
aggregate([  { "$match" : {"in_if" : { "$ne" : "*" },"out_if" : { "$ne" : "*" }, "admin_status" : "active"   }},
{ "$group" : {_id : "$agent","in_if" : { "$addToSet" : "$in_if" },"out_if" : { "$addToSet" : "$out_if"}}}, {
"$project" : {  _id : 1,interfaces : { "$setUnion" : [ "$in_if","$out_if" ] }}} ]);'
MongoDB shell version: 3.2.16
connecting to: 127.0.0.1:27017/flows
{ "_id" : "10.8.0.1", "interfaces" : [ 0 ] }
{ "_id" : "192.168.10.71", "interfaces" : [ 3, 10, 9, 2, 8, 5, 6 ] }
{ "_id" : "10.10.1.1", "interfaces" : [ 5, 4, 0, 1, 2, 3 ] }
```

The _id value is the source IP address of the sFlow sample, This can be considered the Node. The numbers in the interfaces field are the SNMP indexes of the subject interfaces.

Note: Disabling all flows in+out of an interface will remove it from the licensing count (lowering the used count by one interface). opFlow 3.0.2 requires each combo to be disabled, just disabling the wildcard record will not remove the interfaces from the licensing count. The GUI refreshes the license count every 5 minutes, restart omkd if you would like to see the most up-to-date count immediately.

Get a list of known Agents

/usr/local/omk/bin/opflow-cli.pl act=list-agents

e.g.

```
[root@server:/usr/local/omk/bin]#(2) ./opflow-cli.pl act=list-agents
opflow-cli.pl Version 3.1.0

Copyright (C) 2015 Opmantek Limited (www.opmantek.com)
This program comes with ABSOLUTELY NO WARRANTY;
See www.opmantek.com or email contact@opmantek.com
opFlow 1.0 is licensed to Opmantek for 50 Interfaces

Agent IP        In Intf        Out Intf        Active        Last Seen
120.29.0.102        2        1        active        Tue Apr 19 13:53:57 2016
120.29.0.102        1        2        active        Tue Apr 19 13:53:57 2016
120.29.0.102        1        0        active        Tue Apr 19 13:53:57 2016
120.29.0.102        *        *        active        Tue Apr 19 13:53:57 2016
192.168.88.254        8        1        active        Tue Apr 19 13:40:01 2016
192.168.88.254        2        1        active        Tue Apr 19 13:53:57 2016
192.168.88.254        10        1        active        Tue Apr 19 13:53:57 2016
192.168.88.254        1        8        active        Tue Apr 19 13:40:01 2016
```

Stop Processing Flows from an Agent/Interface

To disable processing flows from an agent, disable all agent+in_if+out_if entries, or the wildcard record (i.e. by not passing `in_if` and `out_if`). Here is an example:

```
# disable the whole agent
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 admin_status=inactive

# disable an interface: disable all combinations of in and out interface that involve the one you don't want
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 in_if=2 out_if=1 admin_status=inactive
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 in_if=1 out_if=2 admin_status=inactive
```

Start Processing Flows from an Agent/Interface

Note: Enabling an agent which has individual interface records disabled will not enable those interface records as well.

```
# enable a whole agent,
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 admin_status=inactive


# enable flows in both directions to add a single interface back, only a single direction can be enabled if
desired
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 in_if=2 out_if=1 admin_status=active
/usr/local/omk/bin/opflow-cli.pl act=update-agent agent=120.29.0.102 in_if=1 out_if=2 admin_status=active
```

Manage Filters

Filters allow you to create pre-defined searches that will load quickly in the GUI. A filter must be in place before the flows arrive as the flows are tagged with the filter when they are processed. Any flows that match the filter but arrived before the filter was created will not be displayed.

Note: Agents are filtered automatically, there is no need to create extra filters for them.

Options for filtering rules: agent|application|endpoint|proto|src_ip|dst_ip|src_port|dst_port

Create a Filter Based on Application

```
/usr/local/omk/bin/opflow-cli.pl act=create-filter name=HTTP application=http
/usr/local/omk/bin/opflow-cli.pl act=create-filter name=HTTPS application=https
```