

Clean-up process for opFlow that has gotten too large

In the situation that opFlow has gotten too large for the disk/partitions it is installed on the following process can be followed to clean up all flow /conversation data (but keep other data, like reports).

There are two types of data that need to be cleaned up:

1. Raw flow data (nfdump files)
2. Database data

Before starting

It's best to shut all daemons down before starting.

```
service nfdump stop
service opflowd stop
service omkd stop
service mongod stop
```

Raw flow data

The opFlow installer adds a cron job that cleans these files up. It uses the config variable `opflow_raw_files_age_days` and purges any raw flows that are older than the number of days specified.

```
# purge the raw nfdump input files once daily
23 4 * * * root /usr/local/omk/bin/opflow-cli.pl act=purge-raw quiet=true
```

To clean up manually, find the directory the files are saved into by looking at the config file `omk/conf/opCommon.nmis`:

```
# where nfdump inputs are expected, and saved dailies are kept
'<opflow_dir>' => '/var/lib/nfdump',
```

The default is listed in the code block above. To clean out all files simply delete all files in the directory listed

```
rm -rf /var/lib/nfdump/* # NOTE: be sure this is the directory in the config found above
```

Database data

By default, all opFlow data is stored in a database named 'flows', this is configurable and `opCommon.nmis` has this setting which defines the dbname used `'opflow_db_name' => "flows"`.

NOTE about DB size: when re-creating the database using `opflow-cli`, the following config variables will be used to determine how large to make the collections `'opflow_db_conversations_collection_size' => '16106127360' (15G)`, `'opflow_db_flows_collection_size' => 5368709120 (5G)`. These can be changed or overridden on the CLI by specifying `usepercent=NN`, which calculates the size the db will use based off of the percentage of disk specified.

There are two options to clean up database data.

1. Remove all data, start from scratch. (See note above regarding db size)

```
# drop and re-create with correct size
/usr/local/omk/bin/opflow-cli.pl act=setup-db drop=true
# add auth to new db
/usr/local/omk/bin/setup_mongodb.pl
```

2. Drop flow/conversation data, keep all other data.

- a. First Back up data to keep

```
mongodump -u opUserRW -p op42flow42 -d flows -c customapps -o .
mongodump -u opUserRW -p op42flow42 -d flows -c endpoints -o .
mongodump -u opUserRW -p op42flow42 -d flows -c filters -o .
mongodump -u opUserRW -p op42flow42 -d flows -c iana -o .
mongodump -u opUserRW -p op42flow42 -d flows -c report_data -o .
```

b. Drop the database and re-create it. (See note above regarding db size)

```
# drop and re-create with correct size
/usr/local/omk/bin/opflow-cli.pl act=setup-db drop=true
# add auth to new db
/usr/local/omk/bin/setup_mongodb.pl
```

c. Restore data

```
/usr/local/mongodb/bin/mongorestore -u opUserRW -p op42flow42 -d flows -c customapps customapps.
bson
/usr/local/mongodb/bin/mongorestore -u opUserRW -p op42flow42 -d flows -c endpoints endpoints.bson
/usr/local/mongodb/bin/mongorestore -u opUserRW -p op42flow42 -d flows -c filters filters.bson
/usr/local/mongodb/bin/mongorestore -u opUserRW -p op42flow42 -d flows -c iana iana.bson
/usr/local/mongodb/bin/mongorestore -u opUserRW -p op42flow42 -d flows -c report_data report_data.
bson
```

Last step

Start all daemons back up.

```
service mongod start
service nfdump start
service opflowd start
service omkd start
```