Purging of old data in opEvents

Introduction

Version 2.0.0 (and newer) of opEvents provides a new mechanism for expiring old data from the database. The purging is totally optional and no old data will be removed unless you explicitely configure opEvents to do so.

Important, Versions of opEvents Prior to 2.4.2 will not have a purging policy configured by default, the installer will prompt to enable the default policy.

What can be purged?

opEvents can expire four different types of old data independently:

- Summary reports
- Events and Event Actions
- Raw Logs
- Archve Logs

Configuration

Your desired purging policy is defined by setting one or more of the following four configuration properties in `conf/opCommon.nmis`. Here is the commented example from `install/opCommon.nmis`:

```
'opevents' => {
  # lots of other directives...
  # how long should things be kept in the db? format NN[dhm]MM[hm]..., plain N means N minutes. 0 or undef
means no purging.
  'opevents_reports_purge_older_than' => undef,     # covers summary reports
  'opevents_events_purge_older_than' => undef,      # covers events and actions
  'opevents_rawlogs_purge_older_than' => undef,     # covers raw logs
  'opevents_archivelogs_purge_older_than' => undef, # covers archive logs
```

The configuration is pretty straightforward:

1. no value, the value 0 or the special value `undef` means no purging whatsoever.
2. a purely numeric value is interpreted to mean "purge entries that are older than so many **minutes**".
3. the system understands combinations of the units "d", "h" and "m", in any order and without any delimiting spaces.
   For example, `31d12h` means "purge data older than 31 days and 12 hours" (as does `12h31d`).

Activation

The expiration of old data is performed by `opeventsd.pl` if and when it is started with the argument `act=purge`. You can also instruct it to only tell you how many entries a purge run *would* remove (without removing anything) by giving the arguments `act=purge simulate=true`.

By default the installer for opEvents 2.0 will create a suitable cron schedule in `/etc/cron.d/opevents` which triggers this action once weekly, but you can of course modify this to your liking.

And example of cron job for opEvents, for purging and report tasks:

```
# this cron schedule controls opEvents db purging and reporting
#
# m h dom month dow user command
#
# purge the database once weekly
12 3 * * 5              root /usr/local/opmojo/bin/opevents-cli.pl act=purge
# create daily summary reports
0 9 * * *               root /usr/local/opmojo/bin/opevents-cli.pl act=create-report daily=true
0 15 * * *              root /usr/local/opmojo/bin/opevents-cli.pl act=create-report daily=true
# create weekly summary reports
32 3 * * 1              root /usr/local/opmojo/bin/opevents-cli.pl act=create-report weekly=true
# create monthly summary reports
43 4 1 * *              root /usr/local/opmojo/bin/opevents-cli.pl act=create-report monthly=true
```

Related Topics

- How To - opEvents - Manually Purge Events From OMK Database