Table of Contents

Database Maintenance

Various services are responsible for the correct functionality of the database.

Tracker Virtual machines

All services are installed into a virtual machine: dbloader-tracker

DB Loader

Database Loader purpose to insert data into Database.

DB Loader installation

- Install DB Loader:

```
sudo rpm -i cmsdbldr-0.1.0-1.noarch.rpm
```

- Add detector and database where you would like to upload or retrieve data:

```
e.g. for Tracker and CMSR database we execute: sudo /opt/cmsdbldr/bin/tool.sh -a trker cmsr
```

- Fill properties file which will be created after you execute command above:

```
sudo vim /opt/cmsdbldr/properties/trker_cmsr.properties
```

- Configuration example:

```
url=CMSR_LB
username=<Username>
password=<Password>
root-part-id=1000
condition-core-schema=CMS_TRK_CORE_COND
construct-core-schema=CMS_TRK_CORE_CONSTRUCT
managemnt-core-schema=CMS_TRK_CORE_MANAGEMNT
iov-core-schema=CMS_TRK_CORE_IOV_MGMNT
attribute-core-schema=CMS_TRK_CORE_ATTRIBUTE
view-ext-schema=CMS_TRK_TRKER_VIEW
construct-ext-schema=CMS_TRK_TRKER_CONSTRUCT
condition-ext-schema=CMS_TRK_TRKER_COND
operator-construct-permission=*
operator-condition-permission=*
operator-tracking-permission=*
log-level=INFO
api-port=8882
```

- After that execute this command again:

```
sudo /opt/cmsdbldr/bin/tool.sh -a trker cmsr
```

DB Loader maintenance

IMPORTANT: Every configuration to each database creates separate services and will be called like a configuration file when you registered it.

```
e.g. if we added configuration in previous steps: "sudo /opt/cmsdbldr/bin/tool.sh -a trker cmsr"
```

- Check service status:

```
sudo systemctl status cmsdbldr_trker_cmsr.service
```

- Stop Loader

```
sudo systemctl stop cmsdbldr_trker_cmsr.service
```

- Start Loader

```
sudo systemctl start cmsdbldr_trker_cmsr.service
```

- Restart Loader

```
sudo systemctl restart cmsdbldr_trker_cmsr.service
```

IMPORTANT: All log files are located:

```
/var/cmsdbldr/rest_*
```

DB Loader uninstall

If you have installed Resthub - you have to uninstall it in the first step. Because it requires DB Loader.

- Uninstall Loader Resthub

```
sudo rpm -e cmsdbldrapi
```

- Uninstall Loader

```
sudo rpm -e cmsdbldr
```

DB Loader Resthub

IMPORTANT: Resthub reads the same configuration as DB Loader. But if a user does not want to install data into Database. Just have a configuration to read data. From the configuration example above we should remove "API_PORT".

DB Loader Resthub installation

- Install Resthub

```
sudo rpm -i cmsdbldrapi-0.0.8-1.noarch.rpm
```

- Check service status

```
sudo systemctl status cmsdbldrapi
```

- Stop Resthub

```
sudo systemctl stop cmsdbldrapi
```

- Start Resthub

```
sudo systemctl start cmsdbldrapi
```

- Restart Resthub

```
sudo systemctl restart cmsdbldrapi
```

IMPORTANT: All log files are located:

```
/var/cmsdbldr/cmsdbldrapi.err
/var/cmsdbldr/cmsdbldrapi.out
```

VM Services Maintenance

All required services to start for CMSDB Loader and Resthub API.

CRON job for services/iptables

After VM reboot all iptables configuration is reset to default. As well as all services after reboot are down.

Below you can find instructions on how to setup CRON job to reload iptables after reboot and start all required services.

IPTABLES

After Resthub installation automatically added 2 rules to ip tables:

```
sudo iptables -I INPUT 5 -i eth0 -p tcp --dport 8113 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
sudo iptables -I INPUT -i eth0 -p tcp --dport 80 -m state --state NEW,ESTABLISHED -j ACCEPT
```

* First step is save IP tables configuration to external file in your own directory

```
sudo iptables-save > iptables.txt
```

* Move file to /home directory

```
sudo mv iptables.txt /home/iptables.txt
```

* Create CRONTAB job

```
sudo crontab -e
```

* In crontab file add these lines:

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin

@reboot iptables-restore < /home/iptables.txt
```

* After this save the file and reboot the VM

```
sudo reboot
```

* In order to check if everything went right you can try to log in into the machine again and execute simple grep for our ports form iptables

```
sudo /sbin/iptables -L -n | grep "dpt:8113 "
```

All Services for CMSDB Loader and Resthub

There are at least 4 services, which should be running. If the user already has cron job for e.g. iptables, he just has to edit the file. If not should check the section below how to export and create cron file. HTTPD, INCROND, CMSDBLoader (for each database separate instance), Resthub

*Find all DB Loader services. services should be like cmsdbldr_subsystem_database.service. You can list all services by command below:

```
sudo ls /etc/systemd/system
```

* Edit cron file and add all commands

```
sudo crontab -e
```

* We already have cron job for iptables, so just add additional lines and our file looks like:

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin

@reboot iptables-restore < /home/iptables.txt

@reboot systemctl start httpd
@reboot systemctl start incrond
@reboot /sbin/service cmsdbldrapi start
@reboot systemctl start  cmsdbldr_trker_int2r.service
@reboot systemctl start  cmsdbldr_trker_cmsr.service
```

This topic: Sandbox > OTDatabaseMaintenance
Topic revision: r7 - 2020-08-19 - AivarasSilale

or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback