Security Server Ubuntu 14.04 to 18.04 In-place Upgrade

This document describes the steps required for upgrading a stand-alone Security Server host from Ubuntu 14.04 LTS host to Ubuntu 18.04 LTS. The upgrade is a two-phase process; first from version 14.04 to 16.04 and then to version 18.04. Please read carefully through the whole document before starting the upgrade process. It is assumed that the reader is familiar with the Ubuntu Linux distribution and has experience of Ubuntu release upgrades.

Upgrading a system that uses a hardware security module has not been tested. Please verify that the HSM is compatible with Ubuntu 18.04 and check the HSM module documentation.

Alternative method: Upgrade Security Server to Ubuntu 18.04 Using a Configuration Backup.

Preparations

- Review the Ubuntu release notes and upgrade instructions:
    - 14.04 16.04 https://wiki.ubuntu.com/XenialXerus/ReleaseNotes
    - 16.04 18.04 https://wiki.ubuntu.com/BionicBeaver/ReleaseNotes
- Update all packages to the latest versions:

    - X-Road stable package repository: http://x-tee.ee/packages/live/xroad
    - X-Road test package repository: http://x-tee.ee/packages/test/xroad
    - The repository key: https://x-tee.ee/packages/xroad_repo.gpg
      ```
      sudo apt update && apt full-upgrade
      ```
- Ensure that the X-Road software version is 6.21.1/2.
- Recommended: Make sure that you have an up-to-date backup (or virtual machine snapshot) of the server and database.
- Use the admin UI to take a backup of the security server configuration and *download it to a safe location*

    - This makes it possible to restore the server configuration if the upgrade fails for some reason.
- Stop the security server:
  ```
  sudo stop xroad-proxy; sudo stop xroad-jetty; sudo stop xroad-confclient; sudo stop xroad-signer; sudo stop
  xroad-monitor; sudo stop xroad-opmonitor
  ```

- Optionally, take backup of the message log database. Without a backup, a failure in the upgrade process may cause loss of data.

    - See https://www.postgresql.org/docs/9.3/backup.html
    - Using pg_dump is a straightforward option, but can be problematic if the database is extremely large:
      ```
      sudo -iu postgres pg_dump -d messagelog -F c -f <dump_file>
      ```
    - *Reducing the size of the message log database dump*: By default the message log database keeps 30 days of message records but one can (temporarily) change the retention time in order to reduce the number of records in the database, thus reducing the size of the database dump. Changing the setting has no immediate effect since by default the message log cleanup is run twice a day (noon and midnight). See Security Server User Guide for more information.
- Optionally, take a backup of the archived message log files.
    - Note that some of the logged messages might not have been archived yet and exist only in the database.
- Optionally, take backup of the operational monitoring database (if xroad-opmonitoring is installed). Without a backup, a failure in the upgrade process may cause loss of data.

    - See https://www.postgresql.org/docs/9.3/backup.html
    - ```
      sudo -iu postgres pg_dump -d "op-monitor" -F c -f <dump_file>
      ```

Upgrading Ubuntu 14.04 to 18.04

The upgrade is a two-phase process; first from version 14.04 to 16.04 and then to version 18.04.

- Install update-manager-core if it is not already installed.
- Make sure the Prompt line in /etc/update-manager/release-upgrades is set to 'lts'
- Comment out x-tee repository in the file /etc/apt/sources.list.d/xroad.list:

  ```
  # deb http://x-tee.ee/packages/live/xroad trusty main
  ```

- Launch the upgrade tool with the command `sudo do-release-upgrade`
- Follow the on-screen instructions.

When the upgrade is finished, reboot when prompted (the security server processes won't start after reboot since 16.04 and 18.04 use systemd and the startup scripts are still in upstart format).

Repeat the above process to upgrade from 16.04 to 18.04.

Upgrading the X-Road software

Upgrade the database

- The Ubuntu upgrade process creates empty database instances that should be removed before the old database is upgraded to version 10.

- List the database instances with `pg_lsclusters`, and drop the extra ones (likely 9.5 and 10)
  Do not remove the version 9.3 (main) running on port 5432
  ```
  sudo pg_dropcluster --stop <version, e.g. 10> main
  ```

  > **Example**
  >
  > ```
  > $ sudo pg_lsclusters
  > Ver Cluster Port Status Owner    Data directory              Log file
  > 9.3 main    5432 online postgres /var/lib/postgresql/9.3/main /var/log/postgresql/postgresql-9.3-main.log
  > 9.5 main    5433 online postgres /var/lib/postgresql/9.5/main /var/log/postgresql/postgresql-9.5-main.log
  > 10  main    5434 online postgres /var/lib/postgresql/10/main  /var/log/postgresql/postgresql-10-main.log
  >
  > $ sudo pg_dropcluster --stop 9.5 main
  > $ sudo pg_dropcluster --stop 10 main
  > $ sudo pg_lsclusters
  > Ver Cluster Port Status Owner    Data directory              Log file
  > 9.3 main    5432 online postgres /var/lib/postgresql/9.3/main /var/log/postgresql/postgresql-9.3-main.log
  > ```

- Upgrade the old database to version 10.

  Ubuntu's pg_upgradecluster helper script by default dumps the old database, which can require a lot of time and free disk space if the database is large. The --method=upgrade uses pg_upgrade instead and the --link option avoids copying the data files.
  ⚠️ Using --link requires that the new and old database are in the same filesystem. See https://www.postgresql.org/docs/10/pgupgrade.html for details and recovery instructions.

  ```
  sudo pg_upgradecluster --method=upgrade --link 9.3 main
  ```

  > **Example**
  >
  > ```
  > $ sudo pg_upgradecluster --method=upgrade --link 9.3 main
  >
  > $ sudo pg_lsclusters
  > Ver Cluster Port Status Owner    Data directory              Log file
  > 9.3 main    5433 down   postgres /var/lib/postgresql/9.3/main /var/log/postgresql/postgresql-9.3-main.log
  > 10  main    5432 online postgres /var/lib/postgresql/10/main  /var/log/postgresql/postgresql-10-main.log
  > ```

Update the X-Road package repository to point to the Ubuntu 18.04 packages

- Edit the apt repository in the /etc/apt/sources.list.d/xroad.list to use bionic version (in TEST environment substitude /live/xroad with /test/xroad):

  ```
  deb http://x-tee.ee/packages/live/xroad bionic main
  ```

- (Optional) Remove the now unnecessary openjdk-r-ppa-trusty.* and nginx-stable-trusty.* -files from /etc/apt/sources.list.d/xroad.list

Upgrade the packages

```
sudo apt update && sudo apt full-upgrade
```

Warnings like the following during the upgrade can be ignored:
Failed to stop xroad-jetty.service: Unit xroad-jetty.service not loaded.
invoke-rc.d: initscript xroad-jetty, action "stop" failed.
dpkg: warning: old xroad-jetty9 package pre-removal script subprocess returned error exit status 5

Verify that the security server services are running and the system is responding

```
systemctl list-units "xroad-*" "nginx.service" "postgresql@*"

UNIT                        LOAD    ACTIVE SUB     DESCRIPTION
nginx.service               loaded  active running A high performance web server and a reverse proxy server
postgresql@10-main.service  loaded  active running PostgreSQL Cluster 10-main
xroad-confclient.service    loaded  active running X-Road confclient
xroad-jetty.service         loaded  active running X-Road Jetty server
xroad-monitor.service       loaded  active running X-Road Monitor*
xroad-opmonitor.service     loaded  active running X-Road opmonitor daemon*
xroad-proxy.service         loaded  active running X-Road Proxy
xroad-signer.service        loaded  active running X-Road signer
```

* if the monitoring addon(s) are installed

Drop old database and obsolete packages

After verifying that the database upgrade was successful, drop the old database and remove obsolete PostgreSQL packages.

```
sudo pg_dropcluster 9.3 main
sudo apt purge postgresql-9.3 postgresql-9.5
sudo apt autoremove
```

Troubleshooting

- If recovery from a upgrade failure is not possible and there is a backup/snapshot available, one can restore it and start over.
- If all else fails, do a clean Ubuntu 18.04 installation and restore the security server configuration from the backup. See **Security Server to Ubuntu 18.04 Using a Configuration Backup**