Mise à jour du système Debian

Procédure de mise à jour

- Informer à l'avance les principaux utilisateurs de l'instance de la mise à jour en cours.
- Vérifier la présence d'une sauvegarde correcte et récente :
  - d'un snapshot de l'instance à mettre à jour sur l'interface d'OVH.
  - d'une sauvegarde via Borgmatic du système et des éventuelles base de données.
- Connectez vous en tant que *root* à l'instance depuis la console VNC du manager OVH. Cela permettra de surveiller l'instance lors des redémarrages.
- Vérifiez et notez :
  - le système d'exploitation actuellement installé sur votre machine : `lsb_release -a`
  - le noyau actuellement installé : `uname -mrs`
  - la version de Debian : `cat /etc/debian_version`
- Sauvegarder les éléments qui pourrait s'avérer nécessaire une fois la mise à jour lancée :
  - `cd /home/admin/dwl/`
  - `mkdir -p /home/admin/dwl/upgrade-debian-11-to-12`
  - `cd /home/admin/dwl/upgrade-debian-11-to-12`

  ```
  dpkg --get-selections "*" > system-$(lsb_release --codename|cut -f2)-$(date +"%Y-%m-%d")_installed_packages.txt
  tar -cjf system-$(lsb_release --codename|cut -f2)-$(date +"%Y-%m-%d")-var-lib-dpkg.tar.bz2  /var/lib/dpkg
  tar -cjf system-$(lsb_release --codename|cut -f2)-$(date +"%Y-%m-%d")-var-lib-apt-extended_states.tar.bz2 /var/lib/apt/extended_states
  tar -cjf system-$(lsb_release --codename|cut -f2)-$(date +"%Y-%m-%d")-etc.tar.bz2 /etc
  chmod 600 system-*
  chown admin: system-*
  ```

- En local sur votre machine, créer un dossier qui stockera les éléments les plus important avant la mise à jour : `mkdir -p ~/Data/sinp-<region>/system-upgrade/<instance>-srv/`
  - Placez vous dans ce dossier : `cd ~/Data/sinp-<region>/system-upgrade/<instance>-srv/`
  - Récupérer les éléments présent sur le serveur : `scp admin@<instance>-<region>-sinp:/home/admin/dwl/system-* .`
- Téléchargez et installez les derniers paquets et les mises à jour en attente sur votre version actuelle de Debian :

```
apt update
apt upgrade
apt full-upgrade
apt autoremove --purge
```

- Lister les paquets obsolètes pour les supprimer si possible avec les commandes suivantes :

```
apt list ~o
apt list ?obsolete
aptitude search '~o'
```

    - Suppression des paquets listés avec `apt remove <nom-paquet-1> <nom-paquet-1> <nom-paquet-…>`
- Supprimer les paquets n'appartenant pas aux dépôts officiel Debian et qui ne sont pas utile sur le serveur. Il faut bien sûr **garder les paquets nécessaire** (Ex. Docker, Postgreql). Utiliser la commande :

```
apt list --installed "?not(?origin(debian))"
aptitude search '?narrow(?installed, ?not(?origin(Debian)))'
```

- Chercher et supprimer les anciens fichier de configuration devenus inutiles : `find /etc -name "*.dpkg-*" -o -name "*.ucf-*" -o -name "*.merge-error"`
    - Utilisation de `diff` pour voir les différences et application des changements aux fichiers de config existants.
    - Suppression du fichier `.dpkg-*` après correction.
- Vérifier l'absence de paquet dans un mauvais état d'installation :
    - `dpkg --audit`
    - `dpkg --get-selections "*" > ~/curr-pkgs.txt`
    - Chercher les paquets "hold" : `dpkg --get-selections | grep 'hold$'`
- Redémarrez le système pour installer le dernier noyau Linux et les autres mises à jour : `systemctl reboot`
- Vérifier les sources de paquets extérieurs à la distribution Debian dans : `ll /etc/apt/sources.list.d/` et aussi dans le fichier `vi /etc/apt/sources.list`
    - Si des sources extérieures de paquets extérieures à Debian sont référencées dans le fichier *sources.list* les déplacer dans des fichiers indépendant dans le dossier */etc/apt/sources.list.d/*.
        - Normalement, nous devrions avoir : docker et postgresql.
    - Il faudra les mettre à jour également vers la nouvelle version de Debian lorsque nous le ferons pour le fichier principal : *sources.list* ⇒ changement du nom de la version de Debian.
- Mettez à jour les dépôts du fichier *sources.list* vers ceux de la nouvelle version de Debian :
    - Debian 11 vers 12 ⇒ *bullseye* par *bookworm* : `sed -i 's/bullseye/bookworm/g' /etc/apt/sources.list`
    - Debian 10 vers 11 ⇒ *buster* par *bullseye* : `sed -i 's/buster/bullseye/g' /etc/apt/sources.list`
- Vérifier que tout a bien été remplacé correctement : `vi /etc/apt/sources.list`
- Remplacer également les occurrences de l'ancien nom de version par le nouveau dans tous les fichiers du dossier `/etc/apt/sources.list.d/`
- ⚠ IMPORTANT : Lancer une session *screen* pour pouvoir retrouver l'état de la mise à jour en cas de déconnexion réseau : `screen -S upgrade-system`
- Mettez à jour les paquets :`apt update`
- Effectuer une mise à niveau minimale (met à niveau tous les paquets existants sans installer ni supprimer d'autres paquets) : `apt upgrade --without-new-pkgs`
- Au cours du processus de mise à niveau, vous recevrez sans doute diverses questions comme « Voulez-vous redémarrer le service ? » ou bien « Que faire avec ce fichier de configuration ? ». D'une manière générale, répondez Oui pour redémarrez les services et appuyez sur Entrée pour conserver les fichier de configurations existants, sauf si vous savez ce que vous faites. Les fichiers de config proposés sont retrouvables par la suite et vont permettre de rechercher les

différences pour les appliquer si nécessaire.
- Effectuer la mise à niveau complète : `apt full-upgrade`
- Une fois la mise à niveau vers la nouvelle version de Debian terminée, redémarrez votre machine : `systemctl reboot`
- Vérifiez que la nouvelle version de Debian a bien été installé : `lsb_release -a` et `cat /etc/debian_version`
- Supprimez les paquets obsolètes : `apt autoremove --purge`
- Vérifier que tous les paquets sont bien à jour. Il reste quelques fois des paquets nécessitant une mise à jour :

```
apt update
apt upgrade
apt full-upgrade
apt autoremove --purge
```

- Plutôt que d'attendre la prochaine mise à jour de Debian, vous pouvez dès maintenant chercher et supprimer les anciens fichier de configuration devenus inutiles : `find /etc -name "*.dpkg-*" -o -name "*.ucf-*" -o -name "*.merge-error"`
  - Utilisation de `diff` pour voir les différences et application des changements aux fichiers de config existants. L'idée est de garder les modifications que nous avons fait et d'introduire les changements effectués par le mainteneur du paquet...
  - Lorsqu'il n'y a plus que nos changements comme différence, nous pouvons supprimer le fichier `.dpkg-*`.

Procédure post mise à jour

- A priori, l'utilisation de iptables-legacy comme alternative d'Iptables force la désinstallation de Firewalld et de Docker-CE avant la mise à jour. Il et donc nécessaire de les ré-installer ensuite.
- Ré-installation de *Firewalld* : `apt install firewalld`
  - Mise à jour de l'alternative iptables : `update-alternatives --config iptables` choisir "iptables-legacy"
  - Il ne semble pas nécessaire de recréer les règles du parefeu qui semble continuer à utiliser "iptables-legacy".
- Ré-installation de *Docker*, en suivant la même procédure que pour l'installation initiale
  - Relancer les stacks *Docker Compose* : '' docker compose down ; docker compose up -d''
  - Vérifier avec *Portainer* (https://manager.<domaine-sin>/) le bon fonctionnement des containers.
- La mise à jour du système semble rendre non-fonctionnel les *venv* Python créé sous l'ancienne version du système. Il faut donc les recréer :
  - Recréation du venv du script /opt/srvstatus :
    - Sauvegarder l'ancien venv : `mv venv "venv.save$(date +"%Y-%m-%d")"`
    - Réinstaller en suivant la documentation (inule de cloner à nouveau, mettre à jour seulement avec `git pull`).

Serveur db-srv

- Désinstaller la dernière version de Postgresql (16) qui s'est installé automatiquement :
  - voir les version actuelles de Postgresql qui fonctionnent : `pg_lsclusters`
  - chercher les paquets Postgresql installés : `apt list --installed|grep`

`postgresql`

- Corriger les collations des bases de données Postgresql :
  - Ressource : Doc Postgresql ALTER COLLATION
  - Passer en utilisateur *postgres* : `su - postgres`
  - Lancer *Psql* : `psql`
  - Lister les problèmes de collations vis à vis des bases de données :

```
SELECT datname,
  datcollate,
  datcollversion,
  pg_database_collation_actual_version(oid)
FROM pg_database;
```

  - Pour chaque base avec une collation *datcollversion* différente de *pg_database_collation_actual_version*, il faut :
    - Basculer sur la base (Ex. pour *geonature2db*) : `\c geonature2db`
    - Liste les éventuels objets à reconstruire :

```
SELECT pg_describe_object(refclassid, refobjid, refobjsubid)
AS "Collation",
  pg_describe_object(classid, objid, objsubid) AS "Object"
FROM pg_depend d JOIN pg_collation c
  ON refclassid = 'pg_collation'::regclass AND refobjid =
c.oid
WHERE c.collversion <> pg_collation_actual_version(c.oid)
ORDER BY 1, 2;
```

    - Mettre à jour la collation, une fois les objets reconstruits :

```
ALTER DATABASE geonature2db REFRESH COLLATION VERSION;
```

    - La base *template0* ne doit pas avoir de collation. Pour Debian 12, nous devrions au final avoir quelque chose comme ceci:

```
   datname    | datcollate  | datcollversion |
pg_database_collation_actual_version
--------------+-------------+----------------+----------------
----------------------
 geonature2db | fr_FR.UTF-8 | 2.36           | 2.36
 template1    | fr_FR.UTF-8 | 2.36           | 2.36
 admin        | fr_FR.UTF-8 | 2.36           | 2.36
 gnatlas      | fr_FR.UTF-8 | 2.36           | 2.36
 postgres     | fr_FR.UTF-8 | 2.36           | 2.36
 telegraf     | fr_FR.UTF-8 | 2.36           | 2.36
 template0    | fr_FR.UTF-8 |                | 2.36
(7 lignes)
```

    - Il existe également des scripts Bash capable d'automatiser cette mise à jour des collations.
- Pour le SINP PACA, il a été nécessaire de se connecter à *web-srv* pour relancer les services SystemD de GeoNature, GeoNature-Atlas, TaxHub et UsersHub. L'arrêt de la base de données Postgresql semble leurs poser problème dans certains cas.

- Installer Pipx qui remplace l'utilisation de `pip install --user` à partir de Debian 12 :
  - En tant que *root* installer le paquet : `sudo apt install pipx`
  - Se connecter en tant que *geonat* sur *db-srv* : '' ssh geonat@db-<sinp-region>-sinp ''
  - Configurer les chemins d'accès : `pipx ensurepath`
  - Installer Pipenv : `pipx install pipenv`
  - Relancer le terminal pour prendre en compte les changement ou essayer : `source ~/.bashrc`
  - Vérifier que Pipx est fonctionnel : `pipx --version`
- Réinstaller le venv avec Pipenv dans `/home/geonat/data/import-parser/` :
  - Supprimer l'ancienne installation du venv : `cd ~/.local/share/virtualenvs/ ; rm -f import-parser-* '' * Retourner dans le dossier du Parser : ''cd ~/data/import-parser/`
  - Ré-installer le venv et les dépendances : '' pipenv install ''
- Réinstaller le venv avec Pipenv dans `/home/geonat/data/gn2pg/` :
  - Supprimer l'ancienne installation du venv : `cd ~/.local/share/virtualenvs/ ; rm -f gn2pg-* '' * Retourner dans le dossier du Parser : ''cd ~/data/gn2pg/`
  - Ré-installer le venv et les dépendances : '' pipenv install ''

Serveur web-srv

- Recréer les *venv* de :
  - GeoNature : `cd install; ./01_install_backend.sh`
    - Installer le SDK Sentry dans le venv de GN : `pip install --upgrade sentry-sdk[flask]`
    - Pour Debian 12 et Python 3.11, nécessite de modifier le fichier `backend/requirements-common.txt` :

```
lxml==4.9.4
fiona==1.8.22
```

  - TaxHub :
    - Pour Debian 12 et Python 3.11, nécessite de modifier le fichier `requirements-common.txt` :

```
psycopg2==2.9.5
```

  - UsersHub :
    - Pour Debian 12 et Python 3.11, nécessite de modifier le fichier `requirements-common.txt` :

```
Jinja2==2.11.3
psycopg2==2.9.5
MarkupSafe==2.0.1
```

  - GeoNature-Atlas :
    - Pour Debian 12 et Python 3.11, nécessite de modifier le fichier `requirements.txt` :

```
greenlet==1.1.3
```

    - Relancer les services SystemD de GeoNature, GeoNature-Atlas, TaxHub et UsersHub.

Serveur bkp-srv

- Recréer les *venv* suivant :
    - `/opt/srvstatus/venv` : voir la section Installer le script
    - `/root/borg-env` : voir la section Installer Borg sur Bkp-Srv
    - `/home/backups/borg-env/` : voir la section Installer Borg sur Bkp-Srv
    - `/home/backups/osc-env/` : voir la section Installation d'OpenStack Clien OCS

Mise à jour de Debian 11 vers 12

Ressources

- How to upgrade Debian 11 to Debian 12 bookworm using CLI

Notes

- Résultat de `lsb_release -a` :

```
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux 11 (bullseye)
Release:        11
Codename:       bullseye
```

- Résultat de `uname -mrs` :

```
Linux 5.10.0-27-cloud-amd64 x86_64
```

- Résultat de `cat /etc/debian_version` :

```
11.8
```

- Résultat de `apt list ~o` :

```
gcc-8-base/now 8.3.0-6 amd64  [installé, local]
libapt-inst2.0/now 1.8.2.3 amd64  [installé, local]
libapt-pkg5.0/now 1.8.2.3 amd64  [installé, local]
libboost-iostreams1.67.0/now 1.67.0-13+deb10u1 amd64  [installé, local]
libboost-system1.67.0/now 1.67.0-13+deb10u1 amd64  [installé, local]
libcwidget3v5/now 0.5.17-11 amd64  [installé, local]
libdns-export1104/now 1:9.11.5.P4+dfsg-5.1+deb10u9 amd64  [installé,
local]
libffi6/now 3.2.1-9 amd64  [installé, local]
libgcc1/now 1:8.3.0-6 amd64  [installé, local]
libhogweed4/now 3.4.1-1+deb10u1 amd64  [installé, local]
libicu63/now 63.1-6+deb10u3 amd64  [installé, local]
```

```
libip4tc0/now 1.8.2-4 amd64  [installé, local]
libip6tc0/now 1.8.2-4 amd64  [installé, local]
libisc-export1100/now 1:9.11.5.P4+dfsg-5.1+deb10u9 amd64  [installé,
local]
libjson-c3/now 0.12.1+ds-2+deb10u1 amd64  [installé, local]
libmpdec2/now 2.4.2-2 amd64  [installé, local]
libnettle6/now 3.4.1-1+deb10u1 amd64  [installé, local]
libprocps7/now 2:3.3.15-2 amd64  [installé, local]
libpython3.7-minimal/now 3.7.3-2+deb10u6 amd64  [installé, local]
libpython3.7-stdlib/now 3.7.3-2+deb10u6 amd64  [installé, local]
libreadline7/now 7.0-5 amd64  [installé, local]
linux-image-4.19.0-26-cloud-amd64/now 4.19.304-1 amd64  [installé,
local]
python3.7-minimal/now 3.7.3-2+deb10u6 amd64  [installé, local]
```

- Résultat de `find /etc -name "*.dpkg-*" -o -name "*.ucf-*" -o -name "*.merge-error"` :

```
/etc/default/rkhunter.ucf-dist
/etc/firewalld/firewalld.conf.dpkg-dist
/etc/ca-certificates.conf.dpkg-old
/etc/ssh/sshd_config.ucf-dist
/etc/sysctl.conf.dpkg-dist
/etc/rkhunter.conf.dpkg-dist
/etc/cron.daily/bsdmainutils.dpkg-remove
/etc/cron.daily/rkhunter.dpkg-dist
/etc/apt/apt.conf.d/50unattended-upgrades.ucf-dist
/etc/cloud/cloud.cfg.dpkg-dist
```

- Fichier `sources.list` pour Debian 12 :

```
# Main
deb http://deb.debian.org/debian bookworm main
deb-src http://deb.debian.org/debian bookworm main
# Security
deb http://security.debian.org/debian-security bookworm-security
main
deb-src http://security.debian.org/debian-security bookworm-
security main
# Update
deb http://deb.debian.org/debian bookworm-updates main
deb-src http://deb.debian.org/debian bookworm-updates main
# Backports
deb http://deb.debian.org/debian bookworm-backports main contrib
non-free
deb-src http://deb.debian.org/debian bookworm-backports main
contrib non-free
```

Mise à jour de Debian 10 vers 11

Ressources

- Debian - Upgrades from Debian 10 (buster)
- Mettre à jour Debian 10 vers Debian 11
- OVH Documentation - Mettre à jour votre système d'exploitation

Notes

- Résultat de `lsb_release -a` :

```
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux 10 (buster)
Release:    10
Codename:   buster
```

- Résultat de `uname -mrs` :
  - *db-srv* :
    - SINP-PACA :

      ```
      Linux 4.19.0-22-cloud-amd64 x86_64
      ```

    - SINP-AURA :

      ```
      Linux 4.19.0-23-cloud-amd64 x86_64
      ```

  - *web-srv* :
    - SINP-PACA :

      ```
      Linux 4.19.0-24-cloud-amd64 x86_64
      ```

    - SINP-AURA :

      ```
      Linux 4.19.0-16-cloud-amd64 x86_64
      ```

  - *bkp-srv* :
    - SINP-AURA :

      ```
      Linux 4.19.0-25-cloud-amd64 x86_64
      ```

- Résultat de `cat /etc/debian_version` :

```
10.13
```

- Résultat de `aptitude search '?narrow(?installed, ?not(?origin(Debian)))`:
  - SINP AURA :
    - web-srv:

```
i   containerd.io
- An open and reliable container runtime
i A docker-buildx-plugin
- Docker Buildx cli plugin.
i   docker-ce
- Docker: the open-source application container engine
i   docker-ce-cli
- Docker CLI: the open-source application container engine
i A docker-ce-rootless-extras
- Rootless support for Docker.
i   docker-compose-plugin
- Docker Compose (V2) plugin for the Docker CLI.
i A linux-image-4.19.0-16-cloud-amd64
- Linux 4.19 for x86-64 cloud (signed)
```

- Fichier `sources.list` de Debian 11 :

```
# Main
deb http://deb.debian.org/debian bullseye main
deb-src http://deb.debian.org/debian bullseye main
# Security
deb http://security.debian.org/debian-security bullseye-security main
deb-src http://security.debian.org/debian-security bullseye-security main
# Update
deb http://deb.debian.org/debian bullseye-updates main
deb-src http://deb.debian.org/debian bullseye-updates main
# Backports
deb http://deb.debian.org/debian bullseye-backports main contrib non-free
deb-src http://deb.debian.org/debian bullseye-backports main contrib non-free
```

- Résultat de `find /etc -name "*.dpkg-*" -o -name "*.ucf-*" -o -name "*.merge-error"`:
  - SINP AURA :
    - web-srv :

```
/etc/default/rkhunter.ucf-dist
/etc/firewalld/firewalld.conf.dpkg-dist
/etc/ca-certificates.conf.dpkg-old
/etc/ssh/sshd_config.ucf-dist
/etc/rkhunter.conf.dpkg-dist
/etc/cron.daily/bsdmainutils.dpkg-remove
/etc/cron.daily/rkhunter.dpkg-dist
/etc/apt/apt.conf.d/50unattended-upgrades.ucf-dist
/etc/cloud/cloud.cfg.dpkg-dist
/etc/nginx/sites-available/default.dpkg-dist
/etc/nginx/nginx.conf.dpkg-dist
```

From:
https://sinp-wiki.cbn-alpin.fr/ - **CBNA SINP**

Permanent link:
https://sinp-wiki.cbn-alpin.fr/serveurs/mise-a-jour/systeme-debian?rev=1763892940

Last update: **2025/11/23 10:15**