

Distribution EOLE - T che #37304

Sc nario # 37182 (Termin  (Sprint)): Upgrade-Auto H py 2.9 → 2.10

 tude

26/01/2026 16:24 - Benjamin Bohard

Statut:	Ferm�	D�but:	01/10/2022
Priorit�:	Normal	Ech�ance:	
Assign� �:	Benjamin Bohard	% r�alis�:	100%
Version cible:	Livraison Cadoles - MEN 31/10/2025 (40)	Temps estim�:	0.00 heure
		Temps pass�:	0.00 heure
Description			

Historique

#1 - 26/01/2026 16:25 - Benjamin Bohard

- Statut chang  de Nouveau   En cours

#2 - 26/01/2026 16:30 - Benjamin Bohard

Le paquet contenant les migrations est accessible   l'adresse : https://downloads.opennebula.io/repo/7.0/Ubuntu/24.04/pool/opennebula/opennebula-migration_7.0.1-1_all.deb.

Il ne peut pas  tre install  directement puisqu'il d clare une d pendance sur la version 7 de OpenNebula et que la version install e sur EOLE 2.10 est la 6.10.

Pour passer de la 2.9   la 2.10, seuls les fichiers de migration 6.6.0_to_6.8.0.rb et 6.8.0_to_6.10.0.rb sont n cessaires.  a repr sente 4 fichiers au plus : 2 dans /usr/lib/one/ruby/onedb/local et 2 dans /usr/lib/one/ruby/onedb/shared).   v rifier si les deux sont n cessaires dans le contexte d'Upgrade-Auto.

  l'instance, apr s Upgrade-Auto, le script /usr/share/eole/posttemplate/90-one-db fonctionne comme attendu.

#3 - 26/01/2026 16:34 - Benjamin Bohard

Le service openvswitch-switch ne d marre pas.
Le d marrage   la main renvoie

Failed to start openvswitch-switch.service: Unit netplan-apply.service is masked.

Le service netplan-apply.service ne semble pas disponible mais ne semble pas non plus  tre une d pendance directe du service openvswitch-switch.service

```
# /usr/lib/systemd/system/openvswitch-switch.service
[Unit]
Description=Open vSwitch
Before=network.target
After=network-pre.target ovssdb-server.service ovs-vswitchd.service
PartOf=network.target
Requires=ovssdb-server.service
Requires=ovs-vswitchd.service
```

```
[Service]
Type=oneshot
ExecStart=/bin/true
ExecReload=/usr/share/ovnswitch/scripts/ovs-systemd-reload
ExecStop=/bin/true
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
Also=ovs-record-hostname.service
```

#4 - 26/01/2026 16:36 - Benjamin Bohard

Le démarrage du service `ovs-vswitchd.service` débloque l'instance (blocage à l'étape `/usr/share/eole/postservice/29-ovs-mng` comme indiqué dans le commentaire du scénario).

#5 - 09/02/2026 11:27 - Benjamin Bohard

Pour le cas simple de montée de version d'un nœud isolé, les trois observations ont été effectuées :

- les fichiers de migration sont fonctionnels (4 fichiers nécessaires, mode de distribution non décidé encore) ;
- le service `ovs-vswitchd` peut-être déclaré dans le dictionnaire `29_one-master.xml` en 2.10 (ce service doit être démarré dans le cadre de reconfigure pour que ce dernier ne soit pas bloqué) ;
- les services `opennebula-gate` et `opennebula-flow` posent problème (`systemd` abandonne leur démarrage pour des raisons encore inconnues).

Pour le cas plus complexe de montée de version d'un cluster, il semble manquer une procédure (et les scripts qui permettraient de la mettre en œuvre). Voici ce qui a été réalisé (mais non complété) dans le cadre de ce scénario :

- un script en `pre_upgrade` qui effectue les tâches suivantes :
 - vérification de l'état des VM (certains états sont incompatibles avec la migration de la base) ;
 - vérification du rôle du nœud (la procédure de `opennebula` impose de migrer le leader (base de référence) et `EOLE` fait l'amalgame entre leader et index 0 dans le cluster) ;
 - désactivation des nœuds et de la zone pour ne plus permettre d'éditer la base de données.
- un script en `post_upgrade` qui effectue les tâches suivantes lorsqu'exécuté sur le serveur d'index 0 :
 - montée de version de la base de données (originellement effectuée lors de l'instance suivant l'`Upgrade-Auto`) ;
 - sauvegarde de la base de données ;
 - envoi de la base de données sur les autres nœuds du cluster en attente de mise à jour.
- le même script qui effectue les tâches suivantes lorsqu'exécuté sur les autres nœuds du cluster :
 - utilisation de la sauvegarde envoyée par le serveur d'index 0 pour écraser sa propre base.

Cette procédure n'est pas validée. Voici les observations effectuées à ce stade :

- tous les nœuds devraient disposer de la même base au moment de redémarrer les services ;
- dans l'étape de création du cluster, le nœud d'index 0 initie le cluster alors que les autres serveurs sont disponibles (configuration et services opérationnels) ;
- dans l'étape d'`Upgrade-Auto`, le nœud d'index 0 devrait également attendre que les autres nœuds soient disponibles avant de démarrer les services ;
- il faut découpler la propagation de la base du redémarrage des services et des opérations de synchronisation pour se laisser le temps de préparer les nœuds follower (d'où le déplacement de certains traitements en `post-upgrade`).

Quelques pistes qui n'ont pas été approfondies :

- conditionner les opérations de synchronisation des nœuds pour permettre l'instance des nœuds indépendamment les uns des autres ;
- casser le lien entre index du serveur et rôle dans le cluster (ce rôle n'est pas attribuable manuellement et la conjonction rôle de leader / index 0 n'est pas garantie).

#6 - 09/02/2026 11:32 - Benjamin Bohard

- Fichier 99-hapy-ha ajouté
- Fichier 00-hapy-ha ajouté

#7 - 10/03/2026 10:56 - Benjamin Bohard

- Statut changé de En cours à À valider

#8 - 11/03/2026 11:27 - Ludwig Seys

- Statut changé de À valider à Résolu

#9 - 12/03/2026 10:23 - Joël Cuissinat

- Statut changé de Résolu à Fermé
- % réalisé changé de 0 à 100
- Restant à faire (heures) mis à 0.0

Fichiers

99-hapy-ha	631 octets	09/02/2026	Benjamin Bohard
00-hapy-ha	2,55 ko	09/02/2026	Benjamin Bohard