Utiliser des paquetages de Debian unstable via Ansible

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 octobre 2025

https://www.bortzmeyer.org/ansible-debian-unstable.html

Bon, c'est un article d'ultra-niche. Mais, comme j'ai eu du mal à faire fonctionner cette configuration, je me dis que je peux la décrire, et que peut-être quelqu'un·e trouvera cet article un jour via un moteur de recherche, et que cela lui servira. Donc, le problème était : soit un ensemble de machines Debian configurées via Ansible. Un paquetage dont j'avais besoin n'était que dans la version Debian <https://www.debian.org/doc/manuals/debian-handbook/sect.release-lifecycle.fr.html> "*unstable*". Comment l'installer via le module Ansible apt ?

D'abord, je détaille le problème. Pour une formation DNSSEC <https://www.afnic.fr/produits-services/formations/dnssec/>, je gère N machines virtuelles tournant sous Debian. Le paquetage d'OpenDNSSEC, qui était dans les précédentes versions de Debian, a été retiré de l'actuelle version "*stable*", la version 13, alias « trixie » <https://www.debian.org/News/2025/20250809>. La configuration que j'utilisais d'habitude avec Ansible et son module apt <https://docs.ansible.com/ansible/latest/collections/ansible/builtin/apt_module.html> n'est donc pas utilisable telle quelle pour installer OpenDNSSEC.

Il y a bien sûr plusieurs solutions à ce problème (mais pas celle des "*backports*" <https://backports.debian.org/>, OpenDNSSEC n'y est pas) :

— Ne pas utiliser de paquetage du tout, compiler OpenDNSSEC moi-même et l'installer sur les machines via d'autres modules Ansible qui permettent de copier exécutables et fichiers de configuration. Pas pratique ; cela fait perdre un des gros avantages de l'utilisation de Debian, son excellent système de paquetages.
— Utiliser le successeur d'OpenDNSSEC (qui est officiellement retiré), Cascade <https://blog.nlnetlabs.nl/cascade/>. Pas de paquetage (donc on revient au point précédent) et, en plus, Cascade est loin d'être utilisable.
— Utiliser la version unstable de Debian <https://www.debian.org/doc/manuals/debian-faq/choosing.fr.html#s3.1> au lieu de la "*stable*". Pour des travaux pratiques dans une formation, le risque d'utiliser "*unstable*" est moins grave qu'en production mais, quand même, je ne voudrais pas qu'un problème surgisse pendant les exercices.
— Créer un paquetage Debian sur une machine "*stable*" (en utilisant le source du paquetage de "*unstable*"), le copier vers les machines virtuelles et l'installer avec `dpkg -i`. À la réflexion, c'est sans doute ce que j'aurais dû faire, mais il n'est pas garanti que cela aurait bien marché.

J'ai finalement choisi une autre voie : utiliser "*stable*" mais permettre l'installation de paquetages qui existent dans "*unstable*". Bien que déconseillée par Debian <https://wiki.debian.org/DontBreakDebian#Don.27t_make_a_FrankenDebian>, cette méthode est assez banale et on trouve en ligne plein d'instructions sur comment la réaliser. Une configuration simple est, dans /etc/apt/sources.list.d/unstable.sou

```
Types: deb deb-src
URIs: mirror+file:///etc/apt/mirrors/debian.list
Suites: sid
Components: main
Signed-By: /usr/share/keyrings/debian-archive-keyring.gpg
```

(sid est le petit nom de la version "*unstable*". Si vous avez vu "*Toy Story*", vous savez pourquoi.) Avec cela, vous pouvez maintenant installer des paquetages d'"*unstable*" mais **attention**, vous ne voulez pas que ces paquetages remplacent vos paquetages "*stable*" existants ! Il faut donc aussi donner une basse priorité à "*unstable*", avec un /etc/apt/preferences.d/99unstable :

```
Package: *
Pin: release a=unstable
Pin-Priority: 1
```

Désormais, après le prochain apt update, un apt install quelquechose installera bien la version "*stable*" du paquetage « quelquechose », qui a une priorité supérieure. Et apt install opendnssec installera la version d'"*unstable*" puisqu'il n'y en a pas dans "*stable*".

Tout cela est classique et bien connu, et ça marche. Mais depuis Ansible, patatras. Le module apt d'Ansible prétend qu'il n'existe pas d'OpenDNSSEC, alors que ça marche en ligne de commande avec apt. Voici la configuration du module :

```
tasks:
  ...
  - name: install packages
    apt:   # https://docs.ansible.com/ansible/latest/collections/ansible/builtin/apt_module.html
      name: [truc, machin, opendnssec]
      state: present
```

C'est apparemment parce que Ansible ajoute à apt une option qui lui dit de n'utiliser que "*stable*" et je n'ai pas trouvé le moyen de couper cette option.

La solution est donc de faire deux tâches Ansible, une pour "*stable*" et une pour "*unstable*", en utilisant le paramètre default_release. Voici ma configuration complète et qui marche :

```
tasks:
  ...
  - name: "Copy apt sources"
    copy:
      src: apt-unstable.sources
      dest: /etc/apt/sources.list.d/unstable.sources
  - name: "Copy apt preferences"
    copy:
```

```
    src: apt-preferences.txt
    dest: /etc/apt/preferences.d/99unstable
- name: "Update Repository cache"
  apt: # https://docs.ansible.com/ansible/latest/collections/ansible/builtin/apt_module.html
    update_cache: true
    upgrade: dist
    cache_valid_time: 3600
    force_apt_get: true
- name: install packages
  apt:
    name: [truc, chose, machin]
    state: present
- name: install unstable packages
  apt:
    name: [opendnssec, softhsm2]
    default_release: sid
    state: latest
```

Notez le `state: latest`. Si on laisse à la valeur par défaut, `present`, Ansible produit un message d'erreur incohérent prétendant que `sid` n'est pas dans les sources.