

systemd

1 Prise en main

- Q1.** Lancez `systemctl status`
- (a) Quel est le statut général de votre système ?
 - (b) Dans la hiérarchie des `cgroups`, identifiez celui correspondant à votre session utilisateur
- Q2.** Visualisez les `cgroups` actifs avec `systemd-cgtop`. Triez par nombre de processus.
- (a) *Question pour la fin du TP* : pour voir des statistiques détaillées par `cgroup`, il faut activer des options d'accounting. Activez-les, réessayez.
- Q3.** Visualisez l'état d'un service avec `systemctl status ssh`. Si vous ne voyez pas les dernières lignes de log, vérifiez que vous êtes dans le groupe `systemd-journal`.
- Q4.** Redémarrez un service avec `systemctl restart ssh`. Confirmez avec `systemctl status ssh`
- Q5.** Vérifiez que la complétion automatique fonctionne dans le shell (par exemple, essayez `systemctl status ss<TAB>`). Sinon, installez `bash-completion` et réessayez.
- Q6.** Explorez le statut de votre système :
- (a) `systemctl` ou `systemctl list-units` : les *units* actifs
 - (b) uniquement les services : `systemctl -t service`
 - (c) les units dans l'état failed : `systemctl --state failed`

2 Analyse du boot

- Q7.** Quel a été votre temps de boot? \leadsto `systemd-analyze time`
- Q8.** Analysez votre boot, essayez d'identifier les services lents à démarrer : `systemd-analyze plot > output.svg` ; `firefox output.svg`
- Q9.** Identifiez le chemin critique de votre boot : `systemd-analyze critical-chain`

3 Votre premier service systemd : faked

faked est un faux service, très basique, que nous allons installer et configurer avec `systemd`.

- Q10.** Récupérez *faked* :
- ```
wget -O faked https://github.com/lnussbaum/slides-lectures/raw/master/faked
puis : chmod a+rx faked
```
- Q11.** Lisez (et comprenez) le code de *faked*
- Q12.** Lancez *faked* (`./faked`). Avec `ps` (`ps fx` par exemple), regardez les processus créés par *faked*. Constatez qu'un `CTRL+C` ne suffit pas à tuer les processus fils.
- Q13.** Déplacez *faked* dans `/usr/local/sbin`. Créez un fichier `.service` pour *faked*. Inspirez vous d'un service existant (`systemctl cat atd`, par exemple) et simplifiez éventuellement.
- Q14.** Rechargez les fichiers de configuration avec `systemctl daemon-reload`
- Q15.** Vérifiez que vous pouvez démarrer le service à la main, voir son statut, l'arrêter. Vérifiez qu'un arrêt tue bien tous les processus fils.

**Q16.** Activez le démarrage au boot, vérifiez.

## 4 Utilisation du journal

**Q17.** Consultez les entrées du journal depuis le dernier boot : `journalctl -b`

**Q18.** Consultez les entrées du journal en format natif : `journalctl -o verbose`

**Q19.** Limitez vous aux entrées concernant un service, à un PID, etc. Vérifiez que la complétion fonctionne :  
`journalctl <TAB><TAB>`

**Q20.** Lancez `journalctl -f` dans un terminal, redémarrez *faked* dans un autre terminal, vérifiez que l'affichage est mis à jour en temps réel.

**Q21.** Par défaut, le journal n'est pas persistant dans *Debian jessie* : il est effacé à chaque démarrage. Consultez `/usr/share/doc/systemd/README.Debian.gz` pour le rendre persistant. Vérifiez que les données sont bien écrites dans `/var/log/journal`, et qu'après un reboot, vous pouvez consulter les entrées du boot précédent.

## 5 Améliorez faked

**Q22.** Modifiez *faked* pour utiliser le système de notification pour remonter des informations vers `systemd` :

- (a) Consultez `systemd-notify(1)`
- (b) Ajoutez `NotifyAccess=all` à la configuration du service *faked*
- (c) Dans *faked*, ajoutez des appels `system("systemd-notify --status '....'")` aux différentes étapes du démarrage
- (d) Vérifiez que vous récupérez bien le statut de *faked* dans `systemctl status faked` (ligne *Status* :)

**Q23.** Utilisez le système de notification pour indiquer la fin du démarrage de *faked* :

- (a) Ajoutez `Type=notify` à votre configuration de service
- (b) Dans *faked*, ajoutez un appel à `system("systemd-notify --ready --status '....'")` à la fin du démarrage
- (c) Vérifiez que `systemctl start faked` ou `systemctl restart faked` attend bien la fin du démarrage, au lieu de retourner immédiatement. Les services qui doivent démarrer après *faked* attendront ainsi la fin du démarrage de *faked*.

**Q24.** La sortie (*stdout*, *stderr*) de *faked* est récupérée automatiquement par le journal. Il est possible de formater les messages pour indiquer un niveau de priorité, en écrivant par exemple : `<1>Message de niveau "emergency"`.

- (a) Modifiez *faked* pour afficher des messages avec différents niveaux de priorité
- (b) Vérifiez avec `journalctl` qu'ils sont bien pris en compte (les messages les plus importants – valeur de préfixe la plus faible – sont affichés en rouge dans `systemctl status`)

**Q25.** Demandez à `systemd` de redémarrer automatiquement *faked* en cas de plantage.

- (a) Ajoutez `Restart=on-failure` dans la configuration de votre service.
- (b) Faites crasher *faked* en faisant `kill -USR1 le_pid_de_faked`