

Introduction aux systèmes GNU/Linux

S19E04 inetdoc.net



Philippe Latu / Université Toulouse 3

Document sous licence GNU FDL v1.3
<http://www.gnu.org/licenses/fdl.html>

Plan séance 4

- Séance 4 - Configuration du Système GNU/Linux
 - Utiliser les ressources du shell Bash
 - Identifier & gérer les processus
 - Gérer les permissions sur les fichiers et répertoires
 - Différencier les processus entre système hôte & conteneur
- Manipuler sur machine virtuelle & conteneur
 - Installer & configurer des services Internet
 - Repérer les propriétaires & les propriétés des processus correspondant

Shell Bash

- 1979 → SysV version 7 → Bourne Shell
- 1989 → GNU BASH → Bourne-again Shell → scripts
 - Interpréteur de commandes
 - IEEE POSIX P1003.2/ISO 9945.2 Shell and Tools standard
 - Environnement de développement
 - Fonctions et alias
 - Arithmétique et tableaux
 - Manipulations de chaînes de caractères
 - Documentation shell Bash
 - Advanced Bash Scripting
 - <http://www.tldp.org/LDP/abs/html/>

```
$ man sh
```

Documentation
syntaxe

```
$ help
```

Documentation
commandes internes

Shell Bash

- Éditions et corrections en ligne de commande
 - Auto-correction → commande `shopt -s dirspell`
 - Auto-completion → touche tabulation
 - Historique et rappel des commandes antérieures
 - Commande `history`
 - Séquences de touches
 - Ctrl+R, Ctrl+A, Ctrl+E
 - Shift+PageUp, Shift+PageDown, flèches haut et bas
- Séquencement entre les opérations
 - Tubes ou *pipes* → |
 - Enchaînements logiques → ; → && → ||
 - Commandes internes → jobs, suspend

Shell Bash

- Exemple de script shell
 - Nettoyage des bases de données du navigateur Mozilla Firefox

```
#!/bin/bash
```

Affecter une variable

```
database_tool=sqlite3
```

```
command=$(which $database_tool)
```

Tester la présence d'un programme exécutable

```
if [ ! -x "$command" ]; then
```

```
    echo "ERREUR : le programme $command est introuvable"
```

```
    exit 1
```

```
fi
```

Tester la présence d'un processus actif

```
if [ ! -z `pidof firefox-bin` ]; then
```

```
    echo "ERREUR : le navigateur est en cours d'exécution"
```

```
    exit 1
```

```
fi
```

Traiter tous les fichiers de bases de données dans une boucle

```
for base in ~/.mozilla/firefox/*.default/*.sqlite; do
```

```
    $command $base "VACUUM" && echo "Base $(basename $base .sqlite) nettoyée."
```

```
done
```

```
exit 0
```

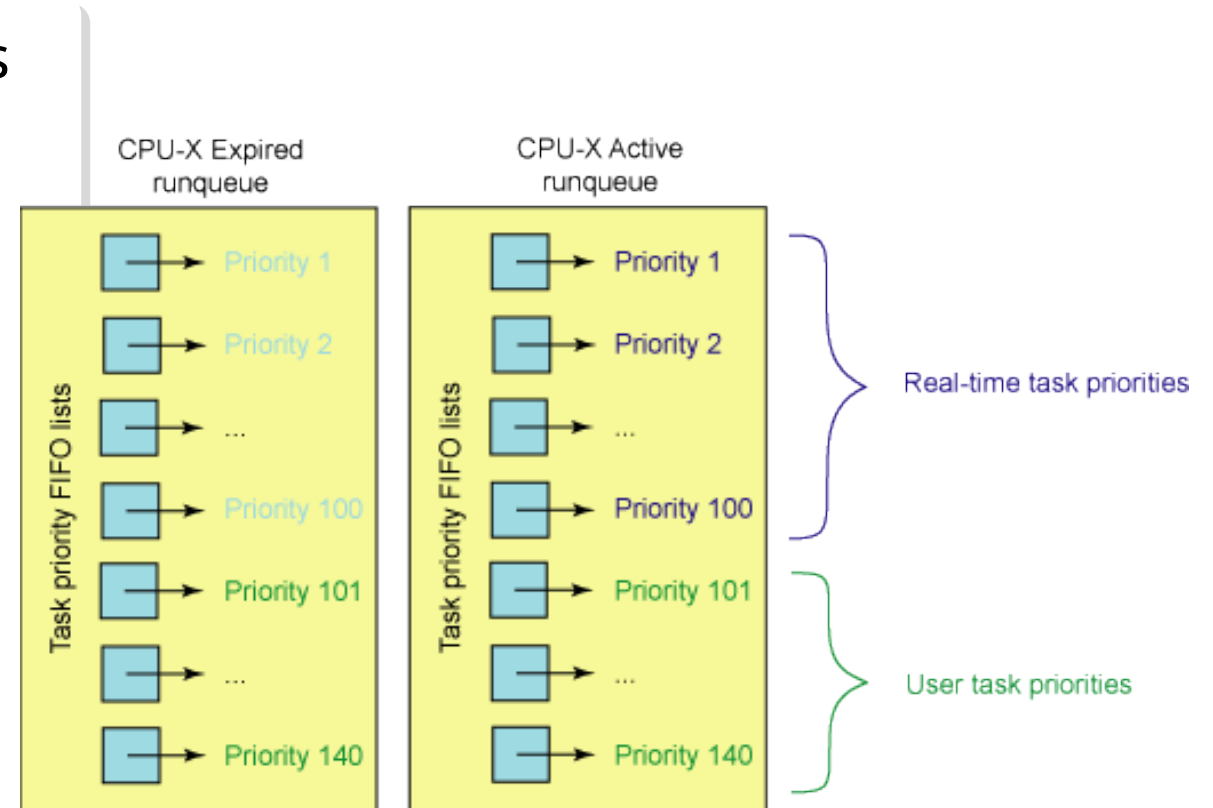
Shell Bash

▪ Applications

- Comment lister l'historique des commandes ?
- Quel est l'effet de la séquence de touche Ctrl+E ?
- Comment rappeler une commande dans l'historique ?
- Quel est l'effet de la séquence de touche Ctrl+D ?
- Comment accéder à la documentation de la commande intégrée shopt ?
- Comment activer l'auto-correction à l'ouverture d'un shell Bash ?
 - https://www.gnu.org/software/bash/manual/html_node/The-Shopt-Builtin.html
- Comment ajouter la commande 'z' à l'ouverture d'un shell Bash ?
 - Quelques indications
 - Consulter l'adresse : <https://github.com/rupa/z>
 - Utiliser la commande 'git' pour cloner le dépôt
 - Utiliser la variable d'environnement \$HOME

Processus = programme en cours d'exécution

- Fonction Unix de base → partage des ressources du système entre différents programmes
- Ordonnanceur (*Scheduler*) → attribution des tranches de temps processeur en fonction du type de tâche
- Fonctions multi-tâches préemptives du noyau Linux
 - Planification de l'exécution des processus
 - Contrôle au début et à la fin de chaque tranche de temps processeur



Processus = programme en cours d'exécution

- Gestion des processus

- Qui est le propriétaire d'un processus ?

- Quelles sont les ressources utilisées par un processus ?

- Comment changer le niveau de priorité d'un processus ?

- Comment tuer un processus défectueux ?

- Commandes de gestion des processus

- Qui, quoi, combien : `ps`, `w`, `top`, `htop`, `iostat`

- Priorités : `nice`, `renice`, `ionice`

- Signalisation et arrêt : `kill`, `killall`

- Utilisation mémoire : `free -m`, `vmstat -w -a -S M`, `vmstat -s`

```
1 [||||] 6.3% 5 [||||] 4.5%
2 [|||||] 10.0% 6 [||] 2.6%
3 [||||] 8.0% 7 [|||||] 9.8%
4 [||||] 8.3% 8 [||||] 7.5%
Mem[|||||||||||||||||] 2.58G/7.83G Tasks: 128, 443 thr; 1 running
Swp[|] 31.4M/14.9G Load average: 0.18 0.19 0.18
Uptime: 05:51:15

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
26955 phil 20 0 1578M 255M 148M S 5.7 3.2 0:20.45 /usr/bin/plasmashell
```


Processus

▪ Applications

▪ Commande `ps`

- Comment visualiser les processus, les propriétaires et les terminaux ?
- Quelle est la signification des 4 options dans la commande 'ps faux' ?

▪ Commandes `kill` et `killall`

- Quelle est la signification du terme signal ?
- Comment relancer un processus ?
- Comment tuer un processus «en force» ?

▪ Processus ou services inutiles







- Comment supprimer un service inutile de façon permanente ?
- Comment caractériser le gain en occupation mémoire correspondant ?

Permissions sur le système de fichiers

- Un **masque** de permissions pour tout fichier ou répertoire

- Exemple : commande **ls -lA**



```
$ ls -lA
drwx----- 2 etu etu 4096 Jan 1 00:04 mail
drwxrwxr-x 3 etu etu 4096 Nov 18 09:54 public_html
-rw-rw---- 1 etu etu 136430 Feb 6 16:52 trash.file
```

 **permissions**  **owner**  **group**  **size**  **date & time**  **name**

- Un utilisateur peut appartenir à plusieurs groupes

- Exemple : commande **id**

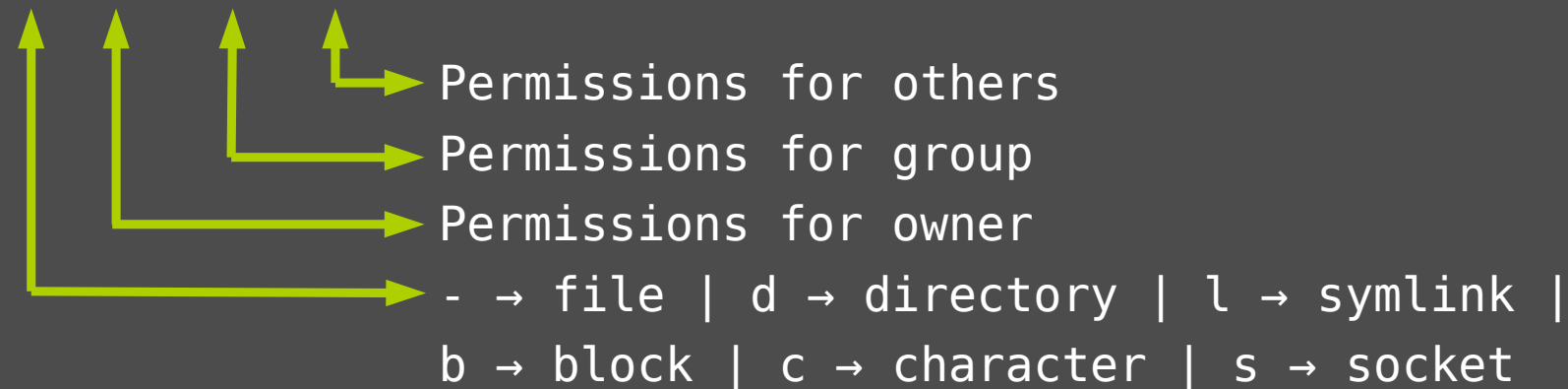
```
$ id
uid=1000(etu) gid=1000(etu) groups=1000(etu),29(audio)
```

 **private uid & gid**  **public group**

Permissions sur le système de fichiers

- Masque des permissions de base → 10 indicateurs
 - Partant de la gauche
 - Premier indicateur → nature de l'objet
 - fichier, répertoire, périphérique ou socket Unix
 - Autres indicateurs → droits
 - lecture, écriture, exécution
 - Propriétaire, groupe et autre

d rwx rwx rwx



Permissions sur le système de fichiers

- Visualisation/Édition du masque des permissions
- Commandes usuelles
 - `ls` → visualisation
 - `chown`, `chgrp` → changement de propriétaire ou de groupe
 - `chmod` → changement de masque
 - `umask` → masque utilisateur utilisé par défaut lors de la création d'objets
- Codage des permissions
 - Notation littérale
 - `r` → read → droit de lecture
 - `w` → write → droit d'écriture
 - `x` → execute → droit d'exécution
 - Notation en octal
 - `r` → 2^2 → 4
 - `w` → 2^1 → 2
 - `x` → 2^0 → 1

```
$ touch emptyfile
$ ls -l emptyfile
-rw- r-- r-- 1 etu etu 0 mai 16 11:20 emptyfile
420 400 400
6 4 4 → 644
$ chmod +x emptyfile
-rwx r-x r-x 1 etu etu 0 mai 16 11:20 emptyfile
421 401 401
7 5 5 → 755 → chmod +x = chmod 755
```

Permissions sur le système de fichiers

▪ Applications

▪ À l'aide de la commande `'ls'`, donner un exemple de :

- programme exécutable
- lien symbolique
- périphérique en mode caractère
- périphérique en mode bloc
- socket Unix

▪ Permissions sur les fichiers et répertoires

- Donner les valeurs numériques des masques d'un fichier de données et d'un programme
- Quel est l'effet de l'instruction `'umask 0027'` ?

▪ Shell script Bash «Hello, World!»

- Créer le fichier script `hello.sh` et le rendre exécutable

```
#!/bin/bash
```

```
echo "Hello, World!"
```

Permissions sur le système de fichiers

- Masque étendu : 3 bits supplémentaires → extension des permissions
 - **SUID** : Set User ID bit
 - **SGID** : Set Group ID bit
 - directory **Sticky bit**
- Ces 3 bits prennent la place du bit d'exécution **x**
 - Pour le propriétaire du fichier
 - **s** indique qu'il a aussi le droit d'exécution
 - **S** indique qu'il n'a pas le droit d'exécution
 - Pour le groupe du fichier
 - **s** indique qu'il a aussi le droit d'exécution
 - **S** indique qu'il n'a pas le droit d'exécution
 - Directory Sticky bit
 - Utile pour les répertoires partagés
 - Un utilisateur ne peut effacer que les fichiers qu'il a créé

Permissions sur le système de fichiers

▪ Applications

- Quel est le rôle du masque étendu pour les objets suivants ?

`/usr/bin/passwd` et `/tmp`

- Comment configurer un environnement de développement Web statique ?

- Installer le paquet `task-web-server`

- Fixer la valeur du masque utilisateur à `0027`

- Créer le répertoire `/var/www/newhtml` avec le masque `rwxr-s---`

L'utilisateur propriétaire doit être `etu` et le groupe propriétaire `www-data`

- Créer un fichier `index.html` dans ce nouveau répertoire

- Ajouter une entrée `newsite` dans `/etc/hosts` avec l'adresse `127.0.0.2`

- Copier le fichier `/etc/apache2/sites-available/000-default.conf` en `newsite.conf`

- Activer le nouveau site avec la commande `a2ensite`

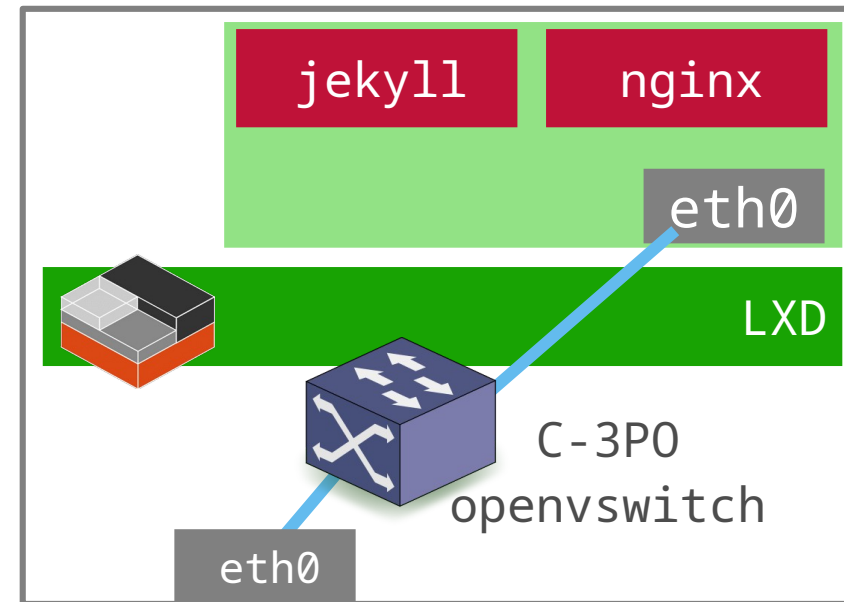
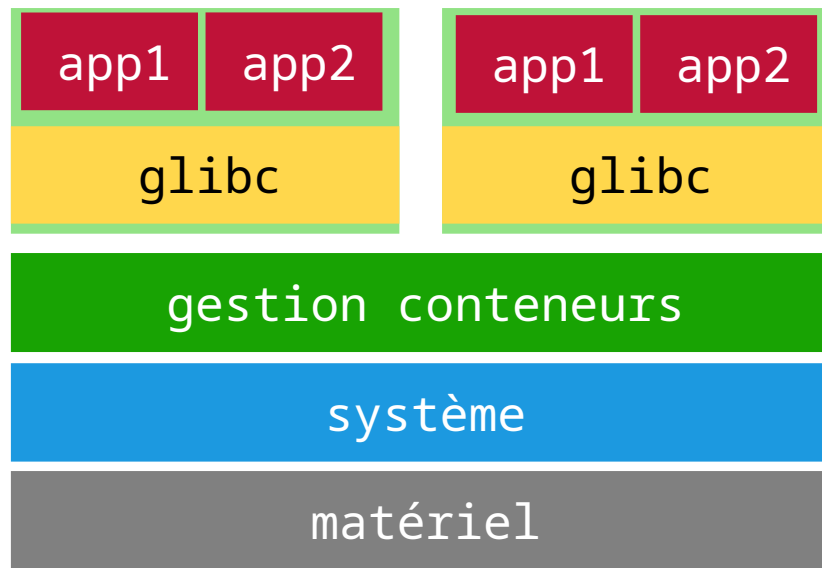
Recharger la configuration du service

- Tester l'ouverture de la page Web et retrouver les traces dans les journaux du service Web

Application → Serveur jekyll - 0

▪ Objectifs

- Transformer le système hôte en commutateur/routeur réseau
- Installer de gestionnaire de conteneurs LXD
- Créer un conteneur 'jekyll'



Application → Serveur jekyll - 1

- Installation des paquets, du service snapd et de LXD
 - Installer les paquets openvswitch-switch et snapd
 - Installer de gestionnaire de conteneurs LXD
 - Attention !
 - L'utilisateur 'etu' doit appartenir au groupe système 'lxd'
 - La liste des chemins de recherche d'applications doit contenir '/snap/bin'

```
etu@vm0:~$ sudo apt install openvswitch-switch snapd
etu@vm0:~$ sudo snap install lxd
etu@vm0:~$ sudo adduser etu lxd
Déconnexion / Reconnexion

etu@vm0:~$ id | grep -o lxd
lxd
etu@vm0:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games:/snap/bin
```

Application → Serveur jekyll - 2

▪ Interconnexion réseau

1. Désactiver la configuration système initiale

```
$ sudo ifdown eth0
```

2. Éditer le fichier `/etc/network/interfaces`

3. Activer la nouvelle configuration

```
$ sudo ifup C-3P0
```

▪ Vérifier l'état des interfaces

```
$ ip addr ls
```

```
etu@vm0:~$ cat /etc/network/interfaces
# This file describes the network
# interfaces available on your system
# and how to activate them.
# For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-ovs C-3P0
iface C-3P0 inet dhcp
        ovs_type OVSBridge
        ovs_ports eth0

allow-C-3P0 eth0
iface eth0 inet manual
        ovs_bridge C-3P0
        ovs_type OVSPort
        up ip link set dev $IFACE up
        down ip link set dev $IFACE down
```

Application → Serveur jekyll - 3

▪ Gestion des conteneurs

1. Configuration initiale

```
$ lxd init
```

2. Éditer le profil 'default'

```
$ lxc profile device set default eth0 \  
nictype bridged
```

3. Créer le conteneur 'jekyll'

```
$ lxc launch images:debian/bullseye jekyll  
$ lxc ls
```

```
etu@vm0:~$ lxd init --dump  
config: {}  
networks: []  
storage_pools:  
- config:  
  size: 15GB  
  source:  
/var/snap/lxd/common/lxd/disks/default.img  
  description: ""  
  name: default  
  driver: btrfs  
profiles:  
- config: {}  
  description: Default LXD profile  
  devices:  
  eth0:  
    name: eth0  
    nictype: bridged  
    parent: C-3P0  
    type: nic  
  root:  
    path: /  
    pool: default  
    type: disk  
  name: default
```

Application → Serveur jekyll - 5

▪ Conteneur jekyll

- Installer les paquets

```
lxc exec jekyll -- apt update
```

```
lxc exec jekyll -- apt install jekyll nginx htop
```

- Configurer le *reverse proxy* nginx

```
lxc exec jekyll -- unlink /etc/nginx/sites-enabled/default
```

```
lxc exec jekyll -- vim /etc/nginx/sites-enabled/reverse.conf
```

```
lxc exec jekyll -- systemctl restart nginx
```

```
lxc exec jekyll -- ss -tan
```

```
lxc exec jekyll -- jekyll serve
```

▪ Système hôte

- Consulter la page web

```
wget -O /dev/null 'adresse IP du conteneur'
```

- **Identifier les processus nginx et jekyll avec htop dans le conteneur et sur le système hôte**

```
server {  
    listen 80;  
    listen [::]:80;  
    location / {  
        proxy_pass http://127.0.0.1:4000;  
    }  
}
```

Bilan séance 4

- Shell Bash
 - Interpréteur de commandes aux fonctions étendues
 - Interface de base de l'administration système → contexte infrastructure
- Processus
 - Tout programme en cours d'exécution a l'«identité» de son propriétaire
- Droits sur le système de fichiers
 - Principes de gestion des droits Unix
 - Compromis efficacité/simplicité
- **Compétences essentielles en administration système**
 - Identifier les processus et leurs propriétaires
 - Différencier l'appartenance des processus entre système hôte et conteneur ou machine virtuelle

Défi pour la prochaine séance !

- Quelle est la relation entre uid/gid et subuid/subgid ?
- Consulter la page à l'adresse ci-dessous
 - <https://ubuntu.com/blog/custom-user-mappings-in-lxd-containers>
 - Quel est le nom de l'utilisateur normal qui obtient la valeur subuid 100000 ?
 - À quel uid cette valeur correspond dans le conteneur ?

