



# Module Linux B2

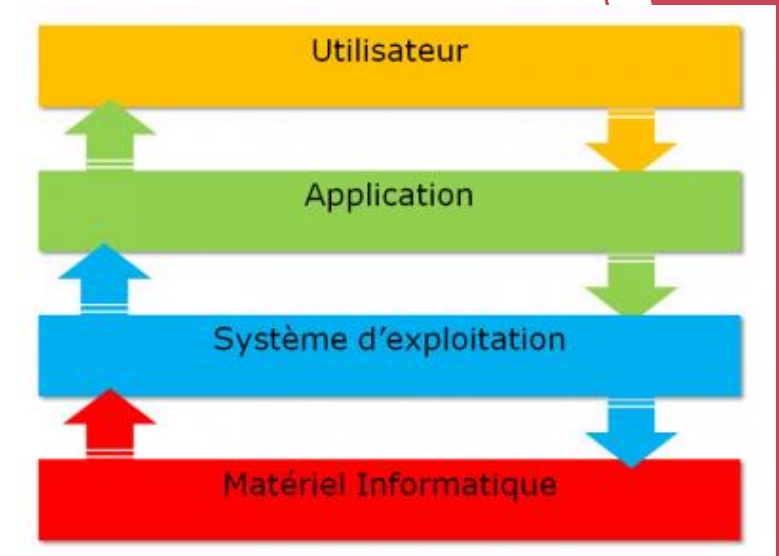
Unité de Formation Infrastructure et SI

# Prérequis

- ▶ Les compétences acquises lors du programme B1
  - ▶ Un e-learning est disponible si besoin de remise à niveau : <https://www.lynda.com/Red-Hat-Enterprise-Linux-tutorials/Cert-Prep-Red-Hat-Certified-System-Engineer-EX300/806170-2.html>
- ▶ Un poste de travail suffisant pour faire tourner une machine virtuelle

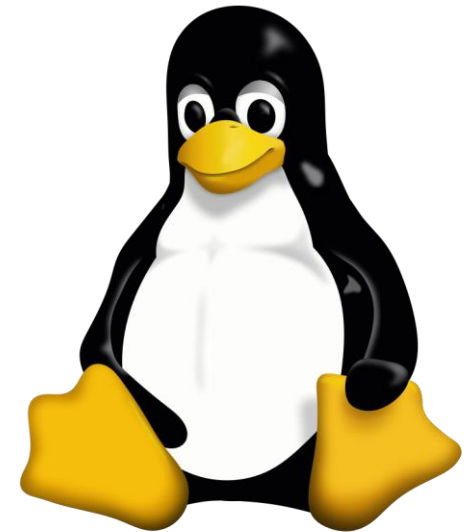
# 1. Éléments constituant un ordinateur

- ▶ CPU : Central Processing Unit
- ▶ Mémoires : RAM, ROM, mémoires de masse
- ▶ BIOS : Basic Input Output System
- ▶ BUS : ISA - PCI - PCI Express



## 2. Présentation de Linux

- ▶ GNU is Not Unix
- ▶ Langage : C et Assembleur
- ▶ Premier noyau Linux : 17 septembre 1991
- ▶ Linus Torvalds
- ▶ Système d'exploitation libre



# 3. Le noyau Linux

- ▶ Le kernel effectue fondamentalement 3 tâches indépendantes :
  - ▶ Il permet de charger les programmes les uns après les autres
  - ▶ Il émule une machine virtuelle
  - ▶ Il gère les ressources
- ▶ Modulaire à chaud
- ▶ Compilé et compilable selon le besoin

# 4. Le système Linux

- ▶ Multi-tâche
- ▶ Multi-session
- ▶ Multi-user
- ▶ Tout est fichier
- ▶ Système de fichiers hiérarchique
- ▶ Interpréteur de commande Shell

# 5. Démarrage du système

- ▶ Enchaînement d'opérations
  - ▶ Le BIOS définit le périphérique de démarrage et charge le MBR
  - ▶ Le MBR charge le programme de démarrage de l'OS : GRUB
  - ▶ GRUB charge le kernel et l'initramfs en mémoire vive
  - ▶ Le kernel utilise l'initramfs pour exécuter le programme init
  - ▶ Init initialise le reste du système
    - ▶ Chargement des modules/pilotes
    - ▶ Montage des partitions du système...
  - ▶ Le kernel effectue ensuite un « switch\_root » et appelle un second programme init

## 6. SysVinit

- ▶ Programme init : PID 1
- ▶ Démarre les autres programmes
  - ▶ Chargement dans l'ordre
- ▶ 7 niveaux de démarrage « runlevel »
  - ▶ 0 mode arrêt du système
  - ▶ 1 mode single user
  - ▶ 2 à 5 mode user
  - ▶ 6 mode redémarrage
- ▶ Mode Start avant le runlevel par défaut
  - ▶ Indique certains modules à charger, des paramètres du kernel, ...



# 7. Les runlevel

- ▶ On définit les services à démarrer et arrêter pour chaque runlevel dans les répertoires :
  - ▶ /etc/rcX.d (ou X est le numéro du runlevel)
- ▶ La syntaxe des liens vers les scripts est la suivante :
  - ▶ S01hostname.sh -> ../init.d/hostname.sh
  - ▶ S02udev -> ../init.d/udev
  - ▶ S09mountall.sh -> ../init.d/mountall.sh
  - ▶ K01apache2 -> ../init.d/apache2
  - ▶ K05umountnfs.sh -> ../init.d/umountnfs.sh

# 8. Systemd

- ▶ Remplaçant de SysVinit
- ▶ N'est plus basé sur les scripts mais sur des fichiers de configurations
  - ▶ service pour les processus
  - ▶ target pour grouper les unités
  - ▶ device pour les périphériques
  - ▶ mount pour les points de montage
  - ▶ timer pour activer les unités en fonction de timers
  - ▶ path pour les services en fonction des objets de système de fichier
- ▶ Options permettant la Gestion des dépendances
  - ▶ After, Before, Requires, ...
- ▶ Démarrage/Arrêt plus rapide
- ▶ Activation des services dynamiquement  $\neq$  SysVinit

# 9. Shutdown

- ▶ Permet un arrêt du système sûr
  - ▶ Notifie les utilisateurs connectés
  - ▶ Envoie une requête à init (init 0)
  - ▶ -r pour un redémarrage (init 6)
  - ▶ -c pour annuler un shutdown en cours (si TIME n'est pas passé)

# 10. Command Line Interface

- ▶ 3 flux standards
  - ▶ Clavier : périphérique d'entrée pour le canal 0 → **stdin**
  - ▶ Ecran : périphérique de sortie pour les canaux 1 et 2 → **stdout** et **stderr**
- ▶ Possibilité de rediriger les flux
  - ▶ `ls -R / >> ok 2>> nok`
  - ▶ `ls -R / >> log 2>&1`
  - ▶ `ls -R / 2>> /dev/null`
- ▶ Le pipe “|” redirige la sortie standard d'une commande vers l'entrée standard d'une seconde

# Pour aller plus loin

- ▶ Lynda :

- ▶ <https://www.lynda.com/Red-Hat-Enterprise-Linux-tutorials/Cert-Prep-Red-Hat-Certified-System-Engineer-EX300/806170-2.html>

- ▶ Formatux :

- ▶ <https://www.formatux.fr/>