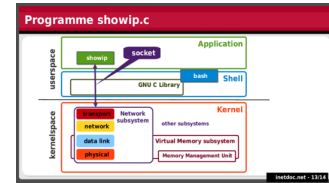


## Architecture système

Quelques éléments d'architecture système, [ODP], [PDF]

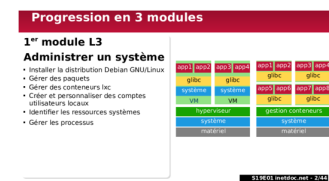
Présentation sur les relations entre l'architecture du système d'exploitation GNU/Linux, le développement de programmes et l'utilisation des appels système via les bibliothèques standard. La dernière vue en forme d'interrogation pose la problématique de l'introduction à la représentation des données dans un système à microprocesseur.



## Introduction aux systèmes GNU/Linux

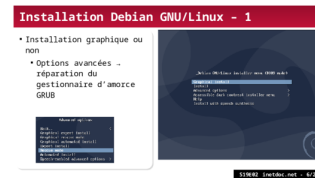
Séance 1 : Du système Unix aux distributions GNU/Linux, [ODP], [PDF]

Le but de la première séance, en plus de la présentation du programme, est d'introduire l'histoire du développement du logiciel libre et donner quelques éléments d'analyse sur les choix de systèmes et d'applications. On passe en revue l'histoire du système Unix, du projet GNU et du noyau Linux. On présente ensuite les caractéristiques des méthodes de développement des projets Open Source. Enfin, on donne la méthode de travail pour les séances suivantes.



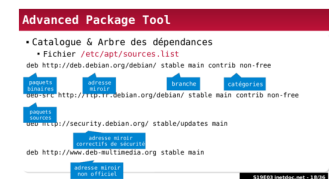
Séance 2 : Installation du Système Debian GNU/Linux, [ODP], [PDF]

La seconde séance est consacrée à l'installation du système GNU/Linux. On utilise la distribution **Debian GNU/Linux**. Elle permet de détailler les différentes étapes d'une installation système : configuration réseau, stratégie de partitionnement, composants du système de base et sélection de jeux de paquets d'applications. On introduit ensuite quelques bases sur l'identification du matériel d'une architecture classique.



Séance 3 : Environnements graphiques - gestion de paquets, [ODP], [PDF]

La troisième séance est consacrée à la présentation des caractéristiques générales des environnements graphiques sur les systèmes GNU/Linux et à la gestion de paquets avec les outils de la bibliothèque Advanced Package Tool de la distribution Debian GNU/Linux. On détaille les opérations usuelles nécessaires au maintien d'une configuration.



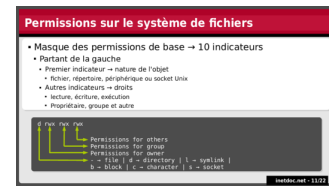
### Captures vidéo

1. Identifier les caractéristiques des serveurs graphiques sur GNU/Linux - 15:05
2. Tester les différents types d'URLs avec le gestionnaire de fichiers KDE - 4:28
3. Identifier les caractéristiques d'un gestionnaire de paquets - 13:38
4. Gérer les paquets avec aptitude en mode console  
Partie 1 : Mettre à jour un système complet - 15:07  
Partie 2 : Manipuler un paquet individuel - 09:35

5. [Gérer les paquets avec aptitude en ligne de commande](#) - 15:07
6. [Identifier les sources des gestionnaires de paquets](#) - 2:26

Séance 4 : Shell Bash - processus - permissions sur les fichiers et répertoires - compilation d'une application, [\[ODP\]](#), [\[PDF\]](#)

La quatrième séance aborde les tâches d'administration de premier niveau. Il s'agit de découvrir les ressources du shell BASH, de visualiser et manipuler les processus et de gérer les permissions sur l'arborescence du système de fichiers. L'autre objectif est de montrer les étapes usuelles de la compilation d'une application à partir de ses sources. C'est une tâche assez ardue qui doit motiver les étudiants à revenir vers la gestion de paquets.

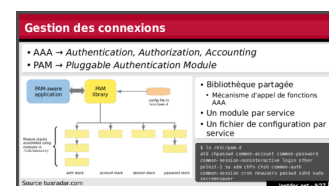


Captures vidéo

1. Utiliser les ressources du shell Bash
  - Partie 1 : [Introduction](#) - 2:45
  - Partie 2 : [Reconnaître les éléments de la syntaxe d'un script Bash](#) - 9:22
  - Partie 3 : [Utiliser le shell avec le gestionnaire de fenêtres byobu](#) - 16:12
2. Identifier et gérer les processus
  - Partie 1 : [Introduction](#) - 8:45
  - Partie 2 : [Utiliser htop, iotop, kill et killall](#) - 20:19
3. Gérer les permissions sur les objets de l'arborescence système
  - Partie 1 : [Introduction](#) - 10:20
  - Partie 2 : [Recenser les types d'objets de l'arborescence et écrire le programme hello.sh](#) - 18:52
  - Partie 3 : [Présentation des permissions étendues](#) - 9:21
  - Partie 4 : [Gérer les permissions étendues lors de la création d'une arborescence de site Web statique](#) - 15:05

Séance 5 : Comptes utilisateurs – journalisation et planification, [\[ODP\]](#), [\[PDF\]](#)

Dans la cinquième séance, on aborde dans un premier temps la gestion et la personnalisation des comptes utilisateurs. Dans un deuxième temps, on présente le fonctionnement de l'authentification PAM. Dans un troisième temps, on exploite les journaux système. Enfin, on observe le fonctionnement de la planification périodique des tâches.



Captures vidéo

1. Gérer les comptes utilisateurs locaux
  - Partie 1 : [Introduction](#) - 10:42
  - Partie 2 : [Créer un compte utilisateur local avec adduser](#) - 6:00
  - Partie 3 : [Gérer les permissions sur les répertoires utilisateurs](#) - 8:33
  - Partie 4 : [Affecter un utilisateur à des groupes](#) - 3:35
  - Partie 5 : [Attribuer les permissions pour le développement de site Web dynamique](#) - 17:44
2. Gérer la personnalisation et l'authentification
  - Partie 1 : [Introduction](#) - 6:28
  - Partie 2 : [Affecter le masque utilisateur umask à l'ouverture de session](#) - 3:38

Partie 3 : [Gérer les changements d'identité avec su et sudo](#) - 3:10

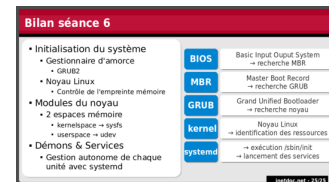
3. [Exploiter les messages systèmes avec rsyslog](#) - 10:07

4. [Gérer la planification des tâches avec cron](#) - 7:19

Séance 6 : Analyse de l'initialisation du système et des services, [\[ODP\]](#), [\[PDF\]](#)

La sixième et dernière séance de cette série présente les étapes de l'initialisation du système. On identifie la sélection d'un noyau avec le gestionnaire d'amorce. On différencie les espaces mémoire utilisateur et noyau pour analyser la gestion des périphériques matériels et le chargement des modules.

Les manipulations sur `systemd`, le gestionnaire de démarrage des services, permettent de reconnaître les services en cours d'exécution et les contraintes d'accès aux ressources matérielles associées. On utilise les groupes de contrôle `cgroups` et la journalisation `journalctl`.



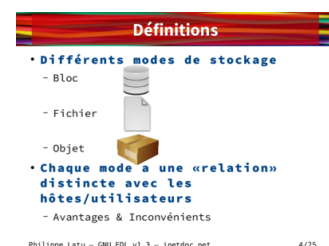
Captures vidéo

1. [Présenter les étapes de l'initialisation d'un système GNU/Linux](#) - 7:26
2. [Identifier les composants d'un noyau Linux et lire le contenu d'un disque RAM initial](#) - 5:24
3. [Installer un nouveau noyau et identifier le rôle du gestionnaire d'amorce](#) - 5:07
4. [Différencier les espaces mémoire noyau et utilisateur](#) - 5:49
5. [Analyser la représentation des périphériques à partir des informations des systèmes de fichiers virtuels](#) - 3:56
6. [Illustrer la représentation d'un périphérique de stockage par le démon udev](#) - 3:29
7. [Faire la correspondance entre un périphérique matériel et son module logiciel](#) - 3:02
8. [Définir le rôle du gestionnaire de services systemd](#) - 3:26
9. [Définir les objets de l'arborescence du gestionnaire de services ainsi que le contrôle des ressources](#) - 3:53
10. [Analyser l'arborescence du gestionnaire de service avec systemctl](#) - 5:02
11. [Contrôler les ressources mémoire et CPU allouées à un service Web apache](#) - 9:00
12. [Définir le rôle de la journalisation fournie avec le gestionnaire de services](#) - 2:20
13. [Exploiter les informations de la journalisation du gestionnaire de services avec journalctl](#) - 11:32

## Administration système en réseau

Stockage Réseau, [\[ODP\]](#), [\[PDF\]](#)

Cette présentation est une introduction aux notions de base sur le stockage de données dans un système d'information. Elle débute par les définitions DAS, SAN, NAS et le mode mode objet. Elle introduit ensuite les distinctions entre les profils d'accès au différents types de stockage. Enfin, elle présente les protocoles de stockage sur IP et la notion de volume logique. Elle se termine par le scénario de travaux pratiques à suivre.



## Systèmes de fichiers réseau, [ODP], [PDF]

Cette présentation est une introduction sur les systèmes de fichiers réseau et au stockage objet. Partant d'une topologie type associant les réseaux SAN et NAS, on introduit le rôle du sous-système Virtual File System du noyau qui rend les accès au stockage transparents du point de vue de l'espace utilisateur. On présente ensuite le mécanisme des appels de procédures distants (RPC). Ce même mécanisme est utilisé par les deux principaux systèmes de fichiers réseau NFS et SMB. Seules les bibliothèques de procédures diffèrent entre les deux systèmes de fichiers. La présentation se termine par les caractéristiques du stockage objet. L'architecture Ceph est utilisée comme exemple de ce mode de stockage qui s'adresse directement aux applications.

