



DEVOIR N°2 DE RECHERCHE DE GROUPE

programmation système

Membres du groupe :

- KINGA RANDO Pheguord**
- BOUYEDI MBAGWU Yannick**

chargé de cours: Dr. Davy MOUSSAVOU

Les appels système UNIX sont essentiels pour la gestion des systèmes de fichiers. Voici comment les utiliser dans un programme en C:

1. mount(2), umount(2), getfsstat(2):

- **mount(2):** Utilisé pour monter un système de fichiers.
- **umount(2):** Utilisé pour démonter un système de fichiers.
- **getfsstat(2):** Récupère des informations sur les systèmes de fichiers montés.
- **Exemple en C:** Ces appels sont généralement utilisés par des programmes système ou des scripts pour gérer les points de montage.

2. stat(2), lstat(2), fsstat(2):

- **stat(2):** Récupère des informations sur un fichier.
- **lstat(2):** Similaire à stat, mais ne suit pas les liens symboliques.
- **fsstat(2):** Récupère des informations sur le système de fichiers.
- **Exemple en C:** Utilisé pour obtenir des informations comme la taille, les permissions, etc.

3. chmod(2), lchmod(2), fchmod(2):

- **chmod(2):** Change les permissions d'un fichier.
- **lchmod(2):** Change les permissions sans suivre les liens symboliques.
- **fchmod(2):** Change les permissions d'un fichier via un descripteur de fichier.
- **Exemple en C:** Utilisé pour modifier les permissions d'accès.

4. chown(2), lchown(2), fchown(2):

- **chown(2):** Change le propriétaire et le groupe d'un fichier.
- **lchown(2):** Similaire à chown, mais ne suit pas les liens symboliques.

- **fchown(2):** Change le propriétaire

et le groupe via un descripteur de fichier.

- **Exemple en C:** Utilisé pour changer la propriété des fichiers.

5. **utimes(2), lutimes(2), futimes(2):**

- **utimes (2):** Change les horodatages d'accès et de modification d'un fichier.
- **lutimes(2):** Similaire à utimes, mais ne suit pas les liens symboliques.
- **futimes(2):** Change les horodatages via un descripteur de fichier.
- **Exemple en C:** Utilisé pour modifier les horodatages des fichiers.

6. **link(2), unlink(2) :**

- **link(2):** Crée un lien physique vers un fichier.
- **unlink(2):** Supprime un lien physique vers un fichier.

- **Exemple en C:** Utilisé pour créer ou supprimer des liens.

7. **rename(2):**

- **rename(2):** Renomme ou déplace un fichier.
- **Exemple en C:** Utilisé pour renommer ou déplacer des fichiers.

8. **symlink(2), readlink(2) :**

- **symlink(2):** Crée un lien symbolique.
- **readlink(2):** Lit le contenu d'un lien symbolique.

- **Exemple en C: Utilisé pour gérer les liens symboliques.**

9. opendir(3), readdir(3), closedir(3):

- **opendir(3): Ouvre un répertoire.**
- **readdir(3): Lit les entrées d'un répertoire.**
- **closedir(3): Ferme un répertoire.**

- **Exemple en C: Utilisé pour parcourir les fichiers dans un répertoire.**

10. chdir(2), fchdir(2) :

- **chdir(2): Change le répertoire de travail courant.**
- **fchdir(2): Change le répertoire de travail courant via un descripteur de fichier.**

Exemple en C: Utilisé pour changer le répertoire courant.

Ces appels système sont généralement utilisés dans des programmes en C pour manipuler des fichiers et des répertoires. Chaque appel a des paramètres spécifiques et des valeurs de retour qui doivent être gérés correctement pour éviter des erreurs.