



Architecture d'un système d'exploitation

(ASE 2017-2018)

TD1

L'ordonnanceur du noyau Linux

marc.perache@cea.fr
hugo.taboada.contact@gmail.com

I Fonctionnement de l'ordonnanceur

Nous vous proposons d'étudier le fonctionnement d'un ordonnanceur. Pour cela, nous allons parcourir le code source dans *kernel/sched/*. Le but de ce TD est de démystifier le code du kernel ainsi que de comprendre de quelle façon est codé un ordonnanceur.

Q.1: Quand avons-nous besoin de l'ordonnanceur ?

Q.2: Dans le code source, trouvez le point d'entrée de `sched_yield()`. À quoi correspond cette fonction ?

La fonction **printk** est une fonction permettant d'afficher des messages depuis le code dans le kernel. Nous pouvons visualiser les messages avec la commande **dmesg**.

Q.3: Insérez un `printk("current=%p",current)` avant et après l'appel à la fonction `schedule()`. Écrivez un code utilisateur avec un appel à `sched_yield()`. Regardez les messages avec `dmesg`. Que remarquez-vous ?

Q.4: Donnez l'enchaînement des fonctions depuis ce point d'entrée.

Q.5: Quelle est la fonction principale de l'ordonnanceur ?

Q.6: Pourquoi la préemption est-elle désactivé dans l'ordonnanceur ?

Q.7: Pourquoi le processus *next* libère-t-il le verrou de la run queue pris par le processus précédent ?

Q.8: Dans quelle fonction est choisi le prochain processus à exécuter ?

Q.9: Décrivez le fonctionnement de l'ordonnanceur. (explication + schéma)

Questions bonus :

Q.10: Trouvez le code du changement de contexte en assembleur et l'expliquer.

Q.11: Pourquoi cet ordonnanceur est bien adapté au temps partagé (plus de processus que de coeurs). Est-il adapté au HPC ?

II Abstractions dans l'ordonnanceur

Maintenant, nous allons étudier la façon dont les différents algorithmes d'ordonnancement sont reliés à l'ordonnanceur.

Q.12: Nous avons vu dans la question **Q.8**, la fonction permettant de sélectionner le prochain processus à exécuter. Quel mécanisme de programmation est utilisé pour relier les algorithmes d'ordonnancement et l'ordonnanceur ?

Q.13: Citez différents algorithmes utilisés dans le noyau linux. **Bonus : Expliquez-les.**

III Completely Fair Scheduling

Actuellement, l'algorithme utilisé par défaut dans le noyau linux est Completely Fair Scheduling (CFS).

Q.14: Dans quel fichier est implémenté cet ordonnanceur ?

Q.15: Sur quelle structure de donnée est basée cet algorithme ?

Q.16: Comment fonctionne-t-il ?