

# Architecture d'un Système d'Exploitation

## Devoir Surveillé

2018 /2019



### 1 Introduction

Ce devoir surveillé comporte 5 parties chacune se rapportant au cours et aux TPs que vous avez eus durant le semestre :

- les composants d'un système d'exploitation
- allocateur mémoire / Ordonnanceur de tâches
- sécurité
- systèmes de fichiers locaux
- analyse de crash et debugging du noyau / Optimisations du système d'exploitation

### 2 Notation

Chacune des 5 parties sera notées sur 20 points, puis l'ensemble (100 points) sera ramené sur 20 pour donner la note de ce contrôle.

En ce qui concerne les QCM, la notation est la suivante :

- la totalité des points pour la réponse juste
- 0 point en l'absence de réponse
- -1 point en cas de réponse fausse

### Rappel

La note de ce contrôle sera moyennée avec la moyenne des Tps pour donner la note de l'UE.

### 3 Déroulement du DS

Durée : 1h45

Format :

**FAIRE 5 copies différentes correspondant aux 5 parties du DS**

**Ni le cours ni les téléphones portables ne sont autorisés pendant l'épreuve.**

# Le sujet

## 1 Les composants d'un système d'exploitation (20 points)

### Questions libres :

1.1 A quoi sert un système d'exploitation ? (7 points)

1.2 En se basant sur ce qui a été vu en cours, décrivez les grandes étapes du démarrage du système d'exploitation Linux. (13 points)

## 2 Allocateur mémoire / Ordonnanceur de tâches (20 points)

### QCM:

2.1 Qu'est-ce que le mot clé static ? (2 points)

- a) une déclaration de donnée constante
- b) une déclaration de variable sur le tas
- c) une déclaration de variable sur la pile
- d) une déclaration de variable globale

2.2 Qu'est-ce qu'une Huge Page ? (2 points)

- a) une page de plus de 4Ko
- b) une page qui n'utilise pas les TLBs
- c) une page en adressage physique
- d) une page dimensionné pour les caches

2.3 Qu'est-ce qu'une page virtuelle ? (2 points)

- a) une page qui n'existe pas vraiment
- b) une page vue du programmeur
- c) une page contenant des zéros
- d) un type de page conçu pour les machines virtuelles

2.4 Qu'est-ce qu'une page physique ? (2 points)

- a) une page accessible uniquement en espace noyau
- b) une page optimisé pour tenir dans les caches
- c) une page vue du système d'exploitation
- d) une page contenant des zéros

### Question libre :

2.5 Comment fonctionne l'ordonnanceur de Linux ? (12 points)

### 3 Sécurité (20 points)

Accessibilité  
Intégrité  
non-repudiation

#### Questions libres :

- 3.1 Donnez les quatre concepts de base de la sécurité informatique avec un exemple pour chacun d'eux, applicable aux environnements HPC. (7 points)
- 3.2 Expliquez ce qu'est le principe de sécurité en profondeur et donnez deux exemples de mesures techniques mettant en œuvre ce principe. (7 points)
- 3.3 Expliquez les différences entre identification et authentification. Donnez deux exemples d'authentification simple et un exemple d'authentification à double facteur. (6 points)

### 4 Systèmes de fichiers locaux (20 points)

FS  
LFS  
NTFS  
XFS

#### Questions libres :

- 4.1 Donnez 3 systèmes de fichiers locaux, en expliquant pour quel média il a été conçu (10 points)
- 4.2 Expliquez ce qu'est la VFS et ce qu'elle apporte (10 points)

interface

5 Analyse de crash et débbuging du noyau / Optimisations du système d'exploitation (20 points)

QCM :

5.1 Dans quel section les allocations dynamiques sont effectuées ? (2 points)

- a) La section bss.
- b) La section heap.
- c) La section txt.
- d) La section data.

5.2 Quelle commande est utilisée pour visualiser les syscalls d'un programme ? (2 points)

- a) strace.
- b) gdb.
- c) cscope
- d) ulimit -c.

5.3 Laquelle de ces affirmations est fausse ? (2 points)

- a) L'outil crash permet de débbuger le noyau.
- b) L'outil perf permet de visualiser le comportement d'un processus particulier.
- c) Il est impossible de contrôler les messages de debug du kernel.
- d) DIS est une commande de l'outil crash.

5.4 Laquelle de ces affirmations est vraie ? (4 points)

- a) Un syscall est exécuté exclusivement en espace utilisateur.
- b) L'appel à un syscall en assembleur utilise exactement les mêmes registres quelle que soit l'architecture du processeur.
- c) L'appel au syscall « exit(int status) » ne déclenche aucun code.
- d) L'appel système « int dup(int fildes) ; » est défini avec la macro SYSCALL\_DEFINE1.

**Question libre :**

5.5 Décrivez les différentes étapes lors de l'exécution de ce programme, dans l'espace utilisateur et dans l'espace noyau (10 points) :

```
$ cat prog.c
void sub(void){
    return;
}
int main(void){
    sub();
    exit(0);
}
$ gcc prog.c -o prog
$ ./prog
```