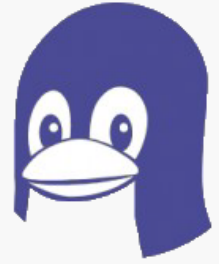


```
# copy .config from the host linux
cp /boot/config-`uname -r`* .config
```

**Formez votre équipe au debug kernel !**



```
# make sure at least below options are enabled
```

```
vim .config
```

```
CONFIG_DEBUG_KERNEL=y
```

```
CONFIG_FRAME_POINTER=y
```

```
CONFIG_KGDB=y
```

**Formation : Debug Kernel**

**Référence : DEBK**

**Durée : 2 jours**

**Niveau : expérimenté**

**Support de cours : français**

**Lieu : Paris ou sur site**

**Tarif : 1400 € HT**

## Nos engagements pédagogiques !

- Des salles équipées de vidéoprojecteurs et de PCs tout équipés
- 50% de cours théorique
- 50% d'exercices pratiques
- Des formateurs qualifiés et **certifiés**
- Chefs de projets, ingénieurs dans l'industrie
- Ayant tous plus de 10 ans d'expérience dans l'animation de formations et la réalisation de projets

## Renseignement et inscription :

Contactez votre conseiller  
formation Styrel

Tél. : +33 7 83 07 61 67

[formation@styrel.fr](mailto:formation@styrel.fr)

## Description

La formation **Debug Kernel** propose d'explorer avec les participants, le système qui sous-tend le fonctionnement du noyau LINUX pour mieux l'appréhender et connaître les sources d'information liées.

## Public concerné

Ce cours s'adresse aux développeurs expérimentés.

## Objectifs de la formation

A l'issue de la formation, le stagiaire sera capable de :

- Savoir collecter de manière exhaustive les informations liées à un dysfonctionnement du noyau
- Savoir analyser les informations recueillies

## Prérequis

Expérience de programmation sous Linux.

## Programme de la formation

### ➤ Systèmes de fichiers et debug

- Système de fichiers virtuel procfs
- Système de fichiers virtuel sysfs
- Collecter des informations de debug avec debugfs
- Stocker des informations de manière persistente avec pstore

### ➤ Debug user space

- Récupérer un core dump
- Utiliser gdb
- Détection de head corruption avec heap / alloc

### ➤ Erreurs kernel et dialogue avec le noyau

- cktrace
- warn
- Kernel tainted – liste des flags

## Styrel : bien plus qu'une formation !

- Intégrateur en Informatique Industrielle depuis 30 ans
- Une vision terrain indépendante de tout constructeur
- Des solutions complètes, du logiciel au matériel, en passant par le service

## Nos certifications

N° d'Agrément Formation Continue :  
11 91 02 737 91

# DataDock



- oops
- panic
- bug

### ➤ Configurer son kernel pour améliorer le debug

- debug info
- kdump / kexec
- Configuration de spin lock, mutex, utilisation de locks printk

### ➤ Les outils de debug kernel

- System.map
- Mettre en place une console série
- Spécificités de l'utilisation d'une console série sous Xen
- Mise en place d'une netconsole
- Utiliser qemu pour debugger
- kgdb (port série)
- crash / kdump
- De l'importance de l'appareil photo
- racing / ftrace
- Quelques paramètres kernel utiles :
- panic=oops, vga=, earlyprintk=, ignore\_loglevel, initcall\_debug, log\_buf\_len

### ➤ Analyser les informations recueillies

- Identifier des adresses mémoire avec addr2line
- gdb, le couteau suisse du débogage
- Un outil d'analyse dédié au kernel : crash
- Outil d'aide à l'analyse : printk
- Définir un format de message avec pr\_\*
- Extraire le device et son driver avec dev\_\*
- printk versus dev\_\* ?

## Évaluation et suivi de la formation

A l'issue de chaque demi-journée de formation, une feuille d'émargement est signée, à la fois par le stagiaire et le formateur. Des exercices pratiques de programmation en cohérence avec les objectifs de ce programme, permettent de vérifier l'acquisition des compétences tout au long de la session. Une attestation de fin de formation est remise au stagiaire lui permettant de faire valoir l'acquisition de ses nouvelles compétences.