



Ceci est un extrait électronique d'une publication de  
Diamond Editions :

<http://www.ed-diamond.com>

Retrouvez sur le site tous les anciens numéros en vente par  
correspondance ainsi que les tarifs d'abonnement.

Pour vous tenir au courant de l'actualité du magazine, visitez :

<http://www.gnulinuxmag.com>

Ainsi que :

<http://www.linux-pratique.com>

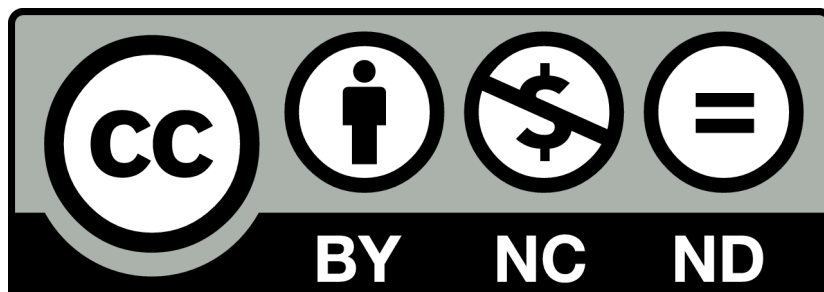
et

<http://www.miscmag.com>



Ceci est un extrait électronique d'une publication de Diamond Editions

<http://www.ed-diamond.com>



## Creative Commons

### Paternité - Pas d'Utilisation Commerciale - Pas de Modification 2.0 France

#### Vous êtes libres :

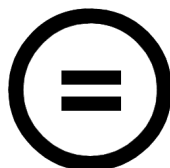
- de reproduire, distribuer et communiquer cette création au public.



**Paternité.** Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'oeuvre).



**Pas d'Utilisation Commerciale.** Vous n'avez pas le droit d'utiliser cette création à des fins commerciales.



**Pas de Modification.** Vous n'avez pas le droit de modifier, de transformer ou d'adapter cette création.

A chaque réutilisation ou distribution de cette création, vous devez faire apparaître clairement au public les conditions contractuelles de sa mise à disposition.

- Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits.
- Rien dans ce contrat ne diminue ou ne restreint le droit moral de l'auteur ou des auteurs.

Ceci est le Résumé Explicatif du Code Juridique. La version intégrale du contrat est attachée en fin de document et disponible sur :

<http://creativecommons.org/licenses/by-nc-nd/2.0/fr/legalcode>

### → Kernel Corner : 2.6.16, 2.6.17 WiFi et SysEnter

Matthieu Barthélemy et Éric Lacombe

**EN DEUX MOTS** L'objectif de cette nouvelle rubrique est de vous informer sur les grandes lignes de l'actualité du noyau Linux : le dernier sorti, mais aussi la version en préparation et le décryptage de leurs principales nouveautés.

Nous approfondirons aussi souvent que possible le sujet par quelques brèves expliquant une partie spécifique du noyau, une nouveauté méritant attention...

Bienvenue et bonne lecture !

L'actualité Linux est un sujet très riche et très intéressant, mais long à suivre et difficile à comprendre. Ceux qui ont fait l'expérience de parcourir la source d'information officielle le savent. Cette source est la *Linux Kernel Mailing List* (ou LKML pour les geeks qui se respectent), c'est-à-dire une liste de diffusion de mails, composée principalement de rapports de bugs, de bouts de codes les corrigeant (*patches*), de propositions d'ajout de fonctionnalités et des discussions parfois enflammées s'ensuivant. Le niveau technique est élevé et le trafic important : aux alentours de 8000 mails par mois. Je vous propose donc ici de faire une synthèse des nouveautés majeures du noyau stable actuel, la série 2.6.16.x et un tour d'horizon de ce qui se profile pour le 2.6.17 (qui sera probablement sorti à l'heure où vous lirez ces lignes).

#### Série 2.6.16 (depuis le 19 mars)

Cette mouture se distingue par l'ajout d'une nouvelle architecture supportée : les processeurs CELL d'IBM. Celui-ci, reposant sur un cœur principal secondé par plusieurs unités de calcul secondaires, équipera par exemple la console Playstation 3.

Du mouvement côté systèmes de fichiers ! On note tout d'abord l'apparition d'un petit nouveau : OCFS (*Oracle Clustering File System*) est un système de fichiers permettant des accès partagés (plusieurs clients accèdent aux mêmes données physiques). Il est dans ce sens similaire au bien connu NFS, mais a un avantage qui le place plutôt en comparaison avec des systèmes de fichiers comme GFS (*Global File System*). Il n'y a pas de relation client/serveur central, mais des nœuds accédant tous aux données sur le même support de stockage physique (avec, bien entendu, gestion des accès concurrents). Ceci ne met pas à l'abri d'un crash disque ou du système qui l'héberge, mais permet d'éviter les problèmes pouvant survenir sur la couche serveur des systèmes comme NFS. La différence avec NFS ne s'arrête pas là. Alors que NFS peut être vu comme une couche d'export de données prenant place au-dessus d'un système de fichiers dit « local » (ext2/3, reiser...), OCFS est un système de fichiers local, très proche de ext3 dans sa structure. Ainsi, pour l'utiliser, vous devrez lui réserver une partition dédiée et formatée via les outils spécifiques (*ocfsformat...*).

Ensuite, le support des droits étendus *Access Control List* (ACL) POSIX pour les montages CIFS (système de partage de fichiers Microsoft) fait son apparition. Les transferts NFS sont, quant à eux, optimisés, gérant maintenant des blocs de données jusqu'à 1 Mo (contre 64K auparavant). Utile avec un serveur NFS adapté (Solaris par exemple).

Ce noyau est le premier à implémenter les « *high resolution timers* », de nouvelles structures de découpage du temps. Jusqu'alors, il ne pouvait (par défaut) gérer des intervalles inférieurs à 4ms (lié à la fréquence par défaut de l'interruption système, 250Hz), ce qui n'est pas gênant dans la majorité des cas, mais peut le devenir pour des applications temps réel et multimédias exigeantes ou pour certains pilotes.

La nouvelle API système *hrtimer*, facilement extensible, pourra autoriser de meilleures résolutions et expose des valeurs en nanosecondes. La fonction *nanosleep*, les *itimers* et les *posix-timers* l'exploitent dès à présent, même si elle utilise encore pour l'instant l'interruption système standard (fréquence configurable par les options *CONFIG\_HZ*). De nouveaux pilotes matériels ont été inclus, en particulier pour eagle (modems ADSL USB Sagem et Comtrend), pour les appareils photo numériques Nikon-CoolPix2000 et Olympus MAUSB-10.

On peut noter une nette amélioration de la gestion de l'énergie sur les périphériques SATA, supportant désormais d'être mis en veille ou arrêtés. Vous souhaitez savoir en un coup d'œil si une option est activée dans votre noyau ? Une nouvelle option (*KCONFIG\_PROC*) permet d'afficher l'équivalent du *.config*, la configuration étant sauvegardée directement dans l'image du noyau :

```
$ gzip -cd /proc/configfs.gz
```

Enfin, cette version marque la fin du support de la série 2.9 de GCC. Il faut désormais GCC >= 3.2 pour compiler correctement Linux.

## Série 2.6.17

La série 2.6.17 introduit de nouveaux appels système : `splice()` et `tee()`. Sans entrer dans des notions qui pourraient être complexes à comprendre, voici l'explication grossière : ces fonctions permettent de transférer des données d'une source vers une/des destination(s) via un tampon noyau et se distinguent d'une copie classique par le fait que les données ne transitent jamais par la mémoire classique (il n'y a pas de copie en espace utilisateur).

C'est donc au niveau des performances que réside tout leur intérêt ; dans leur principe, elles sont directement issues des fameux *pipes* Unix. `splice` transmet des données d'un tampon vers un descripteur de fichiers, tandis que `tee` transfère un *buffer* vers un ou plusieurs autre(s).

La fonction `sendfile()`, qui existe depuis longtemps et est également présente sur d'autres OS Unix, a un but similaire, mais fonctionne différemment et est limitée aux fichiers. `splice()` et `tee()`, quant à elles, peuvent manipuler des fichiers, des *streams* ou des *sockets*, à peu près n'importe quel flot de données donc.

Un transfert via `sendfile()` pouvant être vu comme un cas particulier de `splice()`, ce sera désormais un appel à cette dernière qui sera effectué (pour le 2.6.18 ?), représentant la première utilisation concrète de ces nouveaux appels « *zero-copy movement* ».

Reste ensuite aux applications pouvant en tirer partie à les employer ; pour reprendre un exemple concret donné par Linus, une puce d'encodage vidéo pourrait envoyer son flux simultanément et directement sur le disque pour sauvegarde et sur le réseau pour faire du *streaming*. D'une manière générale, tout transfert de données auquel un processus utilisateur n'a pas besoin d'accéder peut remplir ces conditions.

Une nouvelle architecture supportée fait son apparition : le Niagara, processeur multi-cœur, à l'instar du CELL. Le Niagara est le dernier-né de chez SUN, c'est une architecture de type SPARC. Il a en plus le bon goût d'être *open source*.

Si vous utilisez Linux sur des archis de type ARM, n'hésitez pas à consulter les évolutions du noyau 2.6.17 dans ce domaine, elles sont nombreuses (<http://www.kernelnewbies.org/LinuxChanges>).

Concernant la famille x86, une nouveauté de taille a été implémentée, nommée « SMP alternatives ». Concrètement, cette technique permet au noyau de prendre en charge l'ajout ou le retrait de CPU : tôt lors du *boot*, le code du noyau se modifie dynamiquement en chargeant depuis un fichier binaire (au format ELF) les fonctionnalités permettant la gestion du SMP.

Ces bouts de code contenant des instructions adaptées sont insérés en mémoire à l'endroit précis qui doit contenir le code UP/SMP si au moins deux CPU sont détectés. Conceptuellement, c'est un *patchage* automatique de l'image binaire d'un code en cours d'exécution par lui-même.

Le premier bénéfice des « SMP alternatives » est que les distributions Linux n'auront plus à livrer un noyau distinct pour systèmes mono et multiprocesseurs et qu'il n'y aura plus besoin de recompiler le noyau d'un serveur après lui avoir ajouté un deuxième CPU. En se projetant plus loin, on peut également voir l'intérêt de cette fonctionnalité combinée au support du « CPU *hotplug* » : un processeur ajouté à chaud serait immédiatement pris en compte et utilisé par le système, sans interrompre les processus en cours !

Ceci vaut également pour les environnements virtualisés ; certains (dont Xen) permettant d'affecter à chaud les ressources matérielles aux instances de systèmes virtuels, on peut imaginer de répartir une instance Linux momentanément très chargée sur un ou plusieurs processeurs dédiés : le noyau de l'instance virtualisée les utiliserait alors immédiatement (passage en direct du mode UP au mode SMP) et reviendrait en mode UP, toujours sans interruption, quand l'administrateur du système de virtualisation le déciderait.

Un sujet à suivre donc ! En restant dans le domaine des processeurs, ce concept d'auto-adaptation du code en fonction du contexte pourrait également amener à ne fournir qu'un seul noyau pour plusieurs déclinaisons de CPU : votre distribution, qui livre actuellement un noyau générique fonctionnant sur une grande partie de la famille de votre processeur, vous permet ainsi de l'installer sur une grande variété de systèmes, au prix que les optimisations spécifiques au CPU que vous possédez ne seront pas utilisées. Les *alternatives* offriraient le choix au noyau de charger les fonctionnalités adaptées à votre processeur ! Pour se la jouer philosophique, seul le futur nous dira ce que nous pouvons attendre des implémentations supplémentaires de « code auto-adaptatif ».

Comme d'habitude, de nouveaux périphériques sont gérés, en particulier les cartes WIFI à base de puces Broadcom BCM43xx. Les possesseurs de nombre de portables PC (Acer, Dell ..) et Mac (*Airport Extreme*) sauront apprécier cette nouvelle à sa juste valeur.

En outre, une pile SoftMac générique fait son apparition (en attendant Devicescape ?). Les puces graphiques ATI RS350 (circuit vidéo intégré à la carte mère) sont maintenant supportées ; le nouveau pilote Radeon a, quant à lui, été amélioré.

Ceci devrait profiter aux anciennes cartes PCI comme aux nouveaux matériels de type r300, sous condition que le driver côté Xorg évolue.

A noter que l'ancien, déjà marqué obsolète, est désormais retiré du noyau officiel. Les détenteurs de contrôleurs RAID SATA Promise FastTrak TX4300 (puces PDC40719) pourront maintenant l'exploiter sous Linux pour la première fois. Un bond en avant est également fait pour le support de cartes de type DVB (acquisition vidéo numérique).

La partie systèmes de fichiers n'est pas en reste : le RAID5 logiciel permet le *reshape*, c'est-à-dire l'ajout de volumes, « à chaud » et sans bloquer les opérations de lecture/écriture en cours, y compris lors du *sync*. A noter que pour en bénéficier, il faudra mettre à jour *mdadm*.

Pour savoir ce qui se passe sur vos systèmes de fichiers distants et locaux, deux outils de statistiques détaillées font leur apparition : le premier est l'entrée `/proc/self/mountstats`.

Pour l'instant, il semble qu'il n'y ait que le client NFS qui ait implémenté des statistiques vraiment complètes.

Le second permet de voir en direct ce qui se passe sur vos disques locaux ; il nécessite `debugfs` (et l'option `BLK_DEV_IO_TRACE` est à activer dans le noyau) et les outils utilisateur `blktrace`. Je pense qu'ici un exemple vaut mieux qu'une longue explication.

```
$ git clone git://brick.kernel.dk/data/git/blktrace.git
$ cd blktrace
$ make
```

Passons *root*, puis activons `debugfs` si besoin :

```
# make install
# mkdir /debug
# mount -t debugfs none /debug/
```

On lance l'outil `btrace` sur la partition souhaitée :

```
# btrace /dev/hda
```

S'affiche alors en direct la trace de l'activité sur la partition choisie, ce qui ressemble à :

```
3,0 0 226 15.004000000 0 UT R [swapper] 1
3,0 0 227 15.004000000 7 U R [kblockd/0] 1
3,0 0 241 15.104000000 4004 D WB 112292344+8
[konqueror]
```

Interrompons la trace avec `[CTRL+C]` :

```
CPU0 (3,0):
Reads Queued: 0, 0KiB Writes Queued: 55, 304KiB
Read Dispatches: 0, 0KiB Write Dispatches: 34,
220KiB
Reads Requeued: 0 Writes Requeued: 0
Reads Completed: 0, , 0KiB Writes Completed: 34,
220KiB
Read Merges: 0 Write Merges: 21
Read depth: 0 Write depth: 1
IO unplugs: 12 Timer unplugs: 4
Throughput (R/W): 0KiB/s / 14KiB/s
Events (3,0): 242 entries
Skips: 0 forward (0 - 0.0%)
```

Les options de l'outil sont nombreuses (cf. README) et très utiles. Côté réseau, Netfilter gère maintenant la *connection tracking* et la translation d'adresses (NAT) pour les protocoles H323 (utilisé par beaucoup de logiciels de VoIP), H.245 *tunnelling* et RTP/RTCP.

Ce récapitulatif des deux derniers noyaux est loin d'être exhaustif. Je ne peux que vous inviter à compléter cette lecture par les sites traitant de l'actualité du noyau, tels que L.W.N ([www.lwn.net](http://www.lwn.net)), kernelnewbies ([www.kernelnewbies.org](http://www.kernelnewbies.org)), et aussi kerneltrap ([www.kerneltrap.org](http://www.kerneltrap.org)).

## Focus sur la pile WIFI Devicescape

Une nouvelle a fait beaucoup de bruit : la société Devicescape, auteur d'une pile complète pour périphériques de type 802.11.x, jusqu'alors propriétaire, a décidé de l'offrir dans sa dernière version à la communauté sous licence GPL. Elle intègre les protections MAC, le WEP et le WPA, permet de faire de la QoS, telle que définie par la spécification 802.11 (voix, vidéo, « au mieux » et « en arrière-plan »). Elle peut notamment piloter les puces Atheros, Broadcom et Intel IXP4xx, matériels dont l'installation n'est pas toujours possible (ou les fonctionnalités incomplètes) actuellement sous Linux. Alors, c'est pour quand dans les noyaux officiels ?

Le débat sur l'état du support du WiFi dans le noyau Linux n'est pas nouveau, l'idée d'intégrer une pile complète pouvant factoriser toutes les fonctionnalités

pouvant être requises pour tous les types de *chipsets* a été l'objet d'un long *troll* sur la LKML début décembre 2005. Dès le début Jiri Benc, développeur chez SUSE plaide pour l'intégration de celle de Devicescape. Jeff Garzi, l'ancien développeur en charge du WiFi, s'y oppose, préférant une évolution progressive du code existant déjà dans le kernel.

Il se retrouvera progressivement isolé dans sa position. L'intégration dans le noyau officiel ne sera pas immédiate, pour plusieurs raisons : premièrement, il existe déjà des couches WIFI plus ou moins complètes dans le noyau, comme SoftMAC, utilisée pour les puces BroadCom et Prism54.

Les développeurs ne veulent pas multiplier le code redondant. Deuxièmement, il faut l'adapter aux conventions de codage de Linux pour qu'elle puisse

être incluse (abandon de l'utilisation de `/proc` par la pile, déplacement de code en espace utilisateur). De plus, il faudrait la rendre plus modulaire, afin de ne pouvoir charger que les parties utiles à un matériel donné. John Linville, récemment nommé « mainteneur WIFI » a créé une nouvelle branche de test dans lequel il inclut la pile Devicescape.

Un bon signe pour qu'elle soit officiellement intégrée. Mais si celle-ci fournit les principales fonctions requises pour faire fonctionner beaucoup de chipsets (en particulier ceux déléguant des tâches à l'OS et au driver au lieu de les gérer en hardware, cas des chips SoftMAC), il reste toujours la couche qui fait la jonction pile->matériel à développer spécifiquement pour chaque puce.

Pour tester la branche Linville (les puces de type rt2X00 et bcm43XX utiliseront alors Devicescape) :

```
$ git clone git://git.kernel.org/pub/scm\
  /linux/kernel/git/linville\
  /wireless-dev.git
```

Attention, près de 120Mo seront téléchargés !

```
$ cd wireless-dev
```

...et effectuer les classiques :

```
make [menu/old/g/x]config, make bzImage modules modules_
install.
```

(en choisissant les options relatives à Devicescape et aux chipsets concernés). Il va de soi que tout ceci est hautement expérimental.

## Les appels système et l'instruction SYSENER

Les appels système dans Linux jusqu'à la version 2.5.53 (décembre 2002) utilisaient le chemin des interruptions logicielles.

Lorsqu'un processus en espace utilisateur souhaitait demander un service au noyau, il faisait appel à la `libc`, cette dernière levait l'interruption 0x80 pour provoquer le passage en ring 0 du processeur (i.e. « mode tout puissant » ou encore « mode noyau ») ; et donner la main au noyau (fonction `system_call`) pour que celui-ci récupère via les registres généraux le numéro de l'appel système à effectuer, ainsi que ses paramètres.

Lorsque le noyau avait fini son travail, il utilisait l'instruction IRET pour rendre la main au processus en mode utilisateur.

Regardons plus précisément ce qui se passe lors de la levée d'une interruption. Une lecture de l'IDT (*Interrupt Descriptor Table*) se produit en mémoire pour récupérer l'entrée qui correspond à l'interruption levée.

Elle identifie alors l'entrée dans la GDT (*Global Descriptor Table*) correspondant au segment de code. Pour le cas des appels système, cette lecture est en fait accélérée par le cache du processeur, puisque c'est toujours la même ligne d'interruption qui est utilisée.

Cependant, des cycles de lecture dans le cache et des vérifications d'autorisation d'accès sont nécessaires à chaque fois qu'un passage en mode noyau via l'interruption 0x80 se produit.

# 2 SITES INCONTOURNABLES



Toute l'actualité du magazine sur :

[www.gnulinuxmag.com](http://www.gnulinuxmag.com)

Abonnements et anciens numéros en vente sur :

[www.ed-diamond.com](http://www.ed-diamond.com)



Pour accélérer ce chemin critique, une idée fut que le CPU puisse être *hardcodé* pour toujours aboutir à un point unique dans le segment de code noyau lors d'un appel système.

Nous évitons ainsi les lectures en mémoire. C'est ce que propose l'instruction **SYSENTER** depuis le Pentium 2 (et assimilés). L'utilisation de cette instruction pour le passage en mode noyau depuis un appel système entraîne de réelles performances.

Voyons à présent comment s'utilise cette instruction et comment elle est mise en œuvre dans Linux depuis la version 2.5.53 (option configurable du noyau). **SYSENTER** s'appuie sur des registres spécifiques (MSR, « *Model-Specific Register* ») pour son fonctionnement : **SYSENTER\_CS\_MSR**, **SYSENTER\_EIP\_MSR** et **SYSENTER\_ESP\_MSR**.

Dans le cas de Linux, ces registres sont chargés lors de l'initialisation du noyau. Le premier contient le sélecteur de segment du code noyau, le deuxième contient l'offset dans le segment de code amenant à la routine de gestion des appels système et le dernier pointe sur la pile à utiliser.

Lorsque l'instruction **SYSENTER** est exécutée, le processeur effectue les étapes suivantes :

- ▶ 1 Il charge les registres **CS** et **EIP** avec les valeurs que nous lui avons fournies.
- ▶ 2 Il charge le registre **SS** (*Stack Segment*) grâce à la valeur de **SYSENTER\_CS\_MSR** auquel il ajoute 8 (ce qui correspond à l'entrée « *Kernel Data Segment* » de la GDT).
- ▶ 3 Il charge le registre **ESP** avec la valeur que nous lui avons fournie.
- ▶ 4 Il passe en ring 0.
- ▶ 5 Il exécute la fonction noyau correspondante (l'adresse est alors dans **EIP**).

Pour utiliser cette instruction, Linux mappe à l'avant-dernière page de l'espace d'adressage d'un nouveau processus (0xffffe000 sur 32 bits), un binaire **ELF** contenant le code suivant :

```
__kernel_vsycall:
pushl %ecx
pushl %edx
pushl %ebp
movl %esp, %ebp
sysenter
```

Lors d'un appel système, la **libc**, après avoir chargé les registres généraux avec le numéro du service noyau et ses paramètres, va effectuer un appel à l'endroit du code précédent (au lieu de lever l'interruption 0x80). Le processeur va donc sauvegarder certains registres sur la pile utilisateur du processus courant (nous allons voir pourquoi), puis exécuter **SYSENTER**. Le contrôle est alors donné à la fonction noyau chargée de gérer les appels système.

Elle s'occupe en premier lieu de charger dans le registre **ESP** (contenant jusque-là, la valeur prédéfinie dans **SYSENTER\_ESP\_MSR**) l'adresse de la pile noyau associée au processus s'exécutant. Ensuite, elle exécute le service demandé (via un appel dans la *syscall table*).

Lorsque le service se termine et que le noyau n'a pas d'autre travail en attente (gestionnaire de signal à exécuter par exemple), il est temps de repasser en mode utilisateur.

Pour cela l'instruction **SYSEXIT** est nécessaire. Elle commence par charger **CS** avec la valeur de **SYSENTER\_CS\_MSR** plus 16 (correspondant à l'entrée « *User Code Segment* » de la GDT). **SS** est ensuite chargée par le même type d'opération.

Elle continue en chargeant **EIP** et **ESP** avec les valeurs contenues respectivement dans les registres **EDX** et **ECX**, pour finalement passer en ring 3 et exécuter le code à l'adresse donnée par **EIP**.

Quelques explications supplémentaires s'imposent. **ECX** et **EDX** sont chargés avant le passage en mode utilisateur par le noyau de façon à ce que lors du **SYSEXIT**, le flux d'exécution soit bien détourné au début du code suivant (se trouvant un peu après `__kernel_vsycall`) :

```
popl %ebp
popl %edx
popl %ecx
ret
```

Les valeurs initiales des registres **ECX** et **EDX** (pouvant contenir certains paramètres de l'appel système exécuté) sont donc rétablies.

Vous vous demandez certainement pourquoi **EBP** (pouvant contenir aussi un paramètre) est conservé alors qu'il n'est pas utilisé par **SYSEXIT**.

En fait, le noyau s'en sert pour conserver l'adresse de la base de la pile noyau associée au processus s'exécutant.

A cet endroit, se trouve une structure d'informations importantes sur ce processus. Il est donc nécessaire de sauvegarder la valeur de **EBP** avant que le noyau s'amuse avec ;)

Au final, après le **RET** la **libc** reprend la main pour ensuite continuer l'exécution du processus.

Pour les lecteurs souhaitant plus d'informations sur ce que fait le noyau après **SYSENTER**, je vous conseille de regarder dans les sources du noyau le fichier `arch/i386/kernel/entry.S` à partir de `sysenter_entry`.



# Creative Commons

## Paternité - Pas d'Utilisation Commerciale - Pas de Modification 2.0

Creative Commons n'est pas un cabinet d'avocats et ne fournit pas de services de conseil juridique. La distribution de la présente version de ce contrat ne crée aucune relation juridique entre les parties au contrat présenté ci-après et Creative Commons. Creative Commons fournit cette offre de contrat-type en l'état, à seule fin d'information. Creative Commons ne saurait être tenu responsable des éventuels préjudices résultant du contenu ou de l'utilisation de ce contrat.

### Contrat

L'Oeuvre (telle que définie ci-dessous) est mise à disposition selon les termes du présent contrat appelé Contrat Public Creative Commons (dénommé ici « CPCC » ou « Contrat »). L'Oeuvre est protégée par le droit de la propriété littéraire et artistique (droit d'auteur, droits voisins, droits des producteurs de bases de données) ou toute autre loi applicable. Toute utilisation de l'Oeuvre autrement qu'explicitement autorisée selon ce Contrat ou le droit applicable est interdite.

L'exercice sur l'Oeuvre de tout droit proposé par le présent contrat vaut acceptation de celui-ci. Selon les termes et les obligations du présent contrat, la partie Offrante propose à la partie Acceptante l'exercice de certains droits présentés ci-après, et l'Acceptant en approuve les termes et conditions d'utilisation.

### 1. Définitions

- a. « **Oeuvre** » : oeuvre de l'esprit protégeable par le droit de la propriété littéraire et artistique ou toute loi applicable et qui est mise à disposition selon les termes du présent Contrat.
- b. « **Oeuvre dite Collective** » : une oeuvre dans laquelle l'oeuvre, dans sa forme intégrale et non modifiée, est assemblée en un ensemble collectif avec d'autres contributions qui constituent en elles-mêmes des oeuvres séparées et indépendantes. Constituent notamment des Oeuvres dites Collectives les publications périodiques, les anthologies ou les encyclopédies. Aux termes de la présente autorisation, une oeuvre qui constitue une Oeuvre dite Collective ne sera pas considérée comme une Oeuvre dite Dérivée (telle que définie ci-après).
- c. « **Oeuvre dite Dérivée** » : une oeuvre créée soit à partir de l'Oeuvre seule, soit à partir de l'Oeuvre et d'autres oeuvres préexistantes. Constituent notamment des Oeuvres dites Dérivées les traductions, les arrangements musicaux, les adaptations théâtrales, littéraires ou cinématographiques, les enregistrements sonores, les reproductions par un art ou un procédé quelconque, les résumés, ou toute autre forme sous laquelle l'Oeuvre puisse être remaniée, modifiée, transformée ou adaptée, à l'exception d'une oeuvre qui constitue une Oeuvre dite Collective. Une Oeuvre dite Collective ne sera pas considérée comme une Oeuvre dite Dérivée aux termes du présent Contrat. Dans le cas où l'Oeuvre serait une composition musicale ou un enregistrement sonore, la synchronisation de l'oeuvre avec une image animée sera considérée comme une Oeuvre dite Dérivée pour les propos de ce Contrat.
- d. « **Auteur original** » : la ou les personnes physiques qui ont créé l'Oeuvre.
- e. « **Offrant** » : la ou les personne(s) physique(s) ou morale(s) qui proposent la mise à disposition de l'Oeuvre selon les termes du présent Contrat.
- f. « **Acceptant** » : la personne physique ou morale qui accepte le présent contrat et exerce des droits sans en avoir violé les termes au préalable ou qui a reçu l'autorisation expresse de l'Offrant d'exercer des droits dans le cadre du présent contrat malgré une précédente violation de ce contrat.

**2. Exceptions aux droits exclusifs.** Aucune disposition de ce contrat n'a pour intention de réduire, limiter ou restreindre les prérogatives issues des exceptions aux droits, de l'épuisement des droits ou d'autres limitations aux droits exclusifs des ayants droit selon le droit de la propriété littéraire et artistique ou les autres lois applicables.

**3. Autorisation.** Soumis aux termes et conditions définis dans cette autorisation, et ceci pendant toute la durée de protection de l'Oeuvre par le droit de la propriété littéraire et artistique ou le droit applicable, l'Offrant accorde à l'Acceptant l'autorisation mondiale d'exercer à titre gratuit et non exclusif les droits suivants :

- a. reproduire l'Oeuvre, incorporer l'Oeuvre dans une ou plusieurs Oeuvres dites Collectives et reproduire l'Oeuvre telle qu'incorporée dans lesdites Oeuvres dites Collectives;
- b. distribuer des exemplaires ou enregistrements, présenter, représenter ou communiquer l'Oeuvre au public par tout procédé technique, y compris incorporée dans des Oeuvres Collectives;
- c. lorsque l'Oeuvre est une base de données, extraire et réutiliser des parties substantielles de l'Oeuvre.

Les droits mentionnés ci-dessus peuvent être exercés sur tous les supports, médias, procédés techniques et formats. Les droits ci-dessus incluent le droit d'effectuer les modifications nécessaires techniquement à l'exercice des droits dans d'autres formats et procédés techniques. L'exercice de tous les droits qui ne sont pas expressément autorisés par l'Offrant ou dont il n'aurait pas la gestion demeure réservé, notamment les mécanismes de gestion collective obligatoire applicables décrits à l'article 4(d).

**4. Restrictions.** L'autorisation accordée par l'article 3 est expressément assujettie et limitée par le respect des restrictions suivantes :

- a. L'Acceptant peut reproduire, distribuer, représenter ou communiquer au public l'Oeuvre y compris par voie numérique uniquement selon les termes de ce Contrat. L'Acceptant doit inclure une copie ou l'adresse Internet (Identifiant Uniforme de Ressource) du présent Contrat à toute reproduction ou enregistrement de l'Oeuvre que l'Acceptant distribue, représente ou communique au public y compris par voie numérique. L'Acceptant ne peut pas offrir ou imposer de conditions d'utilisation de l'Oeuvre qui altèrent ou restreignent les termes du présent Contrat ou l'exercice des droits qui y sont accordés au bénéficiaire. L'Acceptant ne peut pas céder de droits sur l'Oeuvre. L'Acceptant doit conserver intactes toutes les informations qui renvoient à ce Contrat et à l'exonération de responsabilité. L'Acceptant ne peut pas reproduire, distribuer, représenter ou communiquer au public l'Oeuvre, y compris par voie numérique, en utilisant une mesure technique de contrôle d'accès ou de contrôle d'utilisation qui serait contradictoire avec les termes de cet Accord contractuel. Les mentions ci-dessus s'appliquent à l'Oeuvre telle qu'incorporée dans une Oeuvre dite Collective, mais, en dehors de l'Oeuvre en elle-même, ne soumettent pas l'Oeuvre dite Collective, aux termes du présent Contrat. Si l'Acceptant crée une Oeuvre dite Collective, à la demande de tout Offrant, il devra, dans la mesure du possible, retirer de l'Oeuvre dite Collective toute référence au dit Offrant, comme demandé. Si l'Acceptant crée une Oeuvre dite Collective, à la demande de tout Auteur, il devra, dans la mesure du possible, retirer de l'Oeuvre dite Collective toute référence au dit Auteur, comme demandé.



- b. L'Acceptant ne peut exercer aucun des droits conférés par l'article 3 avec l'intention ou l'objectif d'obtenir un profit commercial ou une compensation financière personnelle. L'échange de l'Oeuvre avec d'autres Oeuvres protégées par le droit de la propriété littéraire et artistique par le partage électronique de fichiers, ou par tout autre moyen, n'est pas considéré comme un échange avec l'intention ou l'objectif d'un profit commercial ou d'une compensation financière personnelle, dans la mesure où aucun paiement ou compensation financière n'intervient en relation avec l'échange d'Oeuvres protégées.
- c. Si l'Acceptant reproduit, distribue, représente ou communique l'Oeuvre au public, y compris par voie numérique, il doit conserver intactes toutes les informations sur le régime des droits et en attribuer la paternité à l'Auteur Original, de manière raisonnable au regard du médium ou au moyen utilisé. Il doit communiquer le nom de l'Auteur Original ou son éventuel pseudonyme s'il est indiqué ; le titre de l'Oeuvre Originale s'il est indiqué ; dans la mesure du possible, l'adresse Internet ou l'Identifiant Uniforme de Ressource (URI), s'il existe, spécifié par l'Offrant comme associé à l'Oeuvre, à moins que cette adresse ne renvoie pas aux informations légales (paternité et conditions d'utilisation de l'Oeuvre). Ces obligations d'attribution de paternité doivent être exécutées de manière raisonnable. Cependant, dans le cas d'une Oeuvre dite Collective, ces informations doivent, au minimum, apparaître à la place et de manière aussi visible que celles à laquelle apparaissent les informations de même nature.
- d. Dans le cas où une utilisation de l'Oeuvre serait soumise à un régime légal de gestion collective obligatoire, l'Offrant se réserve le droit exclusif de collecter ces redevances par l'intermédiaire de la société de perception et de répartition des droits compétente. Sont notamment concernés la radiodiffusion et la communication dans un lieu public de phonogrammes publiés à des fins de commerce, certains cas de retransmission par câble et satellite, la copie privée d'Oeuvres fixées sur phonogrammes ou vidéogrammes, la reproduction par reprographie.

## 5. Garantie et exonération de responsabilité

- a. En mettant l'Oeuvre à la disposition du public selon les termes de ce Contrat, l'Offrant déclare de bonne foi qu'à sa connaissance et dans les limites d'une enquête raisonnable :
  - i. L'Offrant a obtenu tous les droits sur l'Oeuvre nécessaires pour pouvoir autoriser l'exercice des droits accordés par le présent Contrat, et permettre la jouissance paisible et l'exercice licite de ces droits, ceci sans que l'Acceptant n'ait aucune obligation de verser de rémunération ou tout autre paiement ou droits, dans la limite des mécanismes de gestion collective obligatoire applicables décrits à l'article 4(e);
- b. L'Oeuvre n'est constitutive ni d'une violation des droits de tiers, notamment du droit de la propriété littéraire et artistique, du droit des marques, du droit de l'information, du droit civil ou de tout autre droit, ni de diffamation, de violation de la vie privée ou de tout autre préjudice délictuel à l'égard de toute tierce partie.
- c. A l'exception des situations expressément mentionnées dans le présent Contrat ou dans un autre accord écrit, ou exigées par la loi applicable, l'Oeuvre est mise à disposition en l'état sans garantie d'aucune sorte, qu'elle soit expresse ou tacite, y compris à l'égard du contenu ou de l'exactitude de l'Oeuvre.

**6. Limitation de responsabilité.** A l'exception des garanties d'ordre public imposées par la loi applicable et des réparations imposées par le régime de la responsabilité vis-à-vis d'un tiers en raison de la violation des garanties prévues par l'article 5 du présent contrat, l'Offrant ne sera en aucun cas tenu responsable vis-à-vis de l'Acceptant, sur la base d'aucune théorie légale ni en raison d'aucun préjudice direct, indirect, matériel ou moral, résultant de l'exécution du présent Contrat ou de l'utilisation de l'Oeuvre, y compris dans l'hypothèse où l'Offrant avait connaissance de la possible existence d'un tel préjudice.

## 7. Résiliation

- a. Tout manquement aux termes du contrat par l'Acceptant entraîne la résiliation automatique du Contrat et la fin des droits qui en découlent. Cependant, le contrat conserve ses effets envers les personnes physiques ou morales qui ont reçu de la part de l'Acceptant, en exécution du présent contrat, la mise à disposition d'Oeuvres dites Dérivées, ou d'Oeuvres dites Collectives, ceci tant qu'elles respectent pleinement leurs obligations. Les sections 1, 2, 5, 6 et 7 du contrat continuent à s'appliquer après la résiliation de celui-ci.
- b. Dans les limites indiquées ci-dessus, le présent Contrat s'applique pendant toute la durée de protection de l'Oeuvre selon le droit applicable. Néanmoins, l'Offrant se réserve à tout moment le droit d'exploiter l'Oeuvre sous des conditions contractuelles différentes, ou d'en cesser la diffusion; cependant, le recours à cette option ne doit pas conduire à retirer les effets du présent Contrat (ou de tout contrat qui a été ou doit être accordé selon les termes de ce Contrat), et ce Contrat continuera à s'appliquer dans tous ses effets jusqu'à ce que sa résiliation intervienne dans les conditions décrites ci-dessus.

## 8. Divers

- a. A chaque reproduction ou communication au public par voie numérique de l'Oeuvre ou d'une Oeuvre dite Collective par l'Acceptant, l'Offrant propose au bénéficiaire une offre de mise à disposition de l'Oeuvre dans des termes et conditions identiques à ceux accordés à la partie Acceptante dans le présent Contrat.
- b. La nullité ou l'inapplicabilité d'une quelconque disposition de ce Contrat au regard de la loi applicable n'affecte pas celle des autres dispositions qui resteront pleinement valides et applicables. Sans action additionnelle par les parties à cet accord, lesdites dispositions devront être interprétées dans la mesure minimum nécessaire à leur validité et leur applicabilité.
- c. Aucune limite, renonciation ou modification des termes ou dispositions du présent Contrat ne pourra être acceptée sans le consentement écrit et signé de la partie compétente.
- d. Ce Contrat constitue le seul accord entre les parties à propos de l'Oeuvre mise ici à disposition. Il n'existe aucun élément annexe, accord supplémentaire ou mandat portant sur cette Oeuvre en dehors des éléments mentionnés ici. L'Offrant ne sera tenu par aucune disposition supplémentaire qui pourrait apparaître dans une quelconque communication en provenance de l'Acceptant. Ce Contrat ne peut être modifié sans l'accord mutuel écrit de l'Offrant et de l'Acceptant.
- e. Le droit applicable est le droit français.

Creative Commons n'est pas partie à ce Contrat et n'offre aucune forme de garantie relative à l'Oeuvre. Creative Commons décline toute responsabilité à l'égard de l'Acceptant ou de toute autre partie, quel que soit le fondement légal de cette responsabilité et quel que soit le préjudice subi, direct, indirect, matériel ou moral, qui surviendrait en rapport avec le présent Contrat. Cependant, si Creative Commons s'est expressément identifié comme Offrant pour mettre une Oeuvre à disposition selon les termes de ce Contrat, Creative Commons jouira de tous les droits et obligations d'un Offrant.

A l'exception des fins limitées à informer le public que l'Oeuvre est mise à disposition sous CPCC, aucune des parties n'utilisera la marque « Creative Commons » ou toute autre indication ou logo afférent sans le consentement préalable écrit de Creative Commons. Toute utilisation autorisée devra être effectuée en conformité avec les lignes directrices de Creative Commons à jour au moment de l'utilisation, telles qu'elles sont disponibles sur son site Internet ou sur simple demande.

Creative Commons peut être contacté à <http://creativecommons.org/>.