

NOM

acct – Fichier de comptabilité des processus

SYNOPSIS

```
#include <sys/acct.h>
```

DESCRIPTION

Si le noyau est compilé avec l'option de comptabilité des processus (**CONFIG_BSD_PROCESS_ACCT**) activée, alors l'appel **acct(2)** démarre la comptabilité. Par exemple :

```
acct("/var/log/pacct");
```

Quand la comptabilité des processus est active, le noyau écrit dans le fichier indiqué un enregistrement à chaque mort d'un processus. Cet enregistrement contient des informations concernant le processus terminé, décrites dans *<sys/acct.h>* ainsi :

```
#define ACCT_COMM 16

typedef u_int16_t comp_t;

struct acct {
    char ac_flag; /* Attribut de comptabilité */
    u_int16_t ac_uid; /* UID comptabilisé */
    u_int16_t ac_gid; /* GID comptabilisé */
    u_int16_t ac_tty; /* Terminal de contrôle */
    u_int32_t ac_btime; /* Heure de création du processus */
                    /* (secondes depuis 01/01/1970) */
    comp_t ac_utime; /* Temps CPU en mode utilisateur */
    comp_t ac_stime; /* Temps CPU en mode noyau */
    comp_t ac_etime; /* Temps d'attente */
    comp_t ac_mem; /* Moyenne mémoire utilisée (ko) */
    comp_t ac_io; /* Caractères transmis (inutilisé) */
    comp_t ac_rw; /* Blocs lus ou écrits (inutilisé) */
    comp_t ac_minflt; /* Nb fautes de pages mineures */
    comp_t ac_majflt; /* Nb fautes de pages majeures */
    comp_t ac_swaps; /* Nb de swaps (inutilisé) */
    u_int32_t ac_exitcode; /* Code de retour - voir wait(2) */
    char ac_comm[ACCT_COMM+1];
                    /* Nom de la commande (nom de base
                    de la dernière commande exécutée
                    terminé par zéro) */
    char ac_pad[X]; /* Remplissage */
};

enum { /* Bits qui peuvent se trouver dans ac_flag */
    AFORK = 0x01, /* A exécuté fork mais pas exec */
    ASU = 0x02, /* A utilisé les privilèges superutilisateur */
    ACORE = 0x08, /* A créé un fichier core */
    AXSIG = 0x10 /* Tué par un signal */
};
```

Le type de donnée *comp_t* est une valeur réelle constituée d'un exposant en base 8 sur 3 bits et d'une mantisse sur 13 bits. Une valeur *c*, de ce type peut être convertie en entier (long) ainsi :

```
v = (c & 0x1fff) << (((c >> 13) & 0x7) * 3);
```

Les champs *ac_utime*, *ac_stime*, et *ac_etime* mesurent les temps en Â« tops d'horloge Â». Divisez ces valeurs par *sysconf(_SC_CLK_TCK)* pour obtenir des secondes.

Format des fichiers de comptabilit  version 3

Depuis le noyau 2.6.8, une autre version du fichier peut  tre  ventuellement produite, si l'option **CONFIG_BSD_PROCESS_ACCT_V3** est d finie lors de la compilation du noyau. Lorsque cette option est active, les enregistrements  crits dans le fichier de comptabilit  contiennent des champs suppl men-taires et la longueur des champs *c_uid* et *ac_gid* est  largie de 16   32 bits (en accord avec l' lar-gissement des UID et GID dans les noyaux Linux 2.4 et ult rieur). Les enregistrements sont d finis ainsi :

```
struct acct_v3 {
    char ac_flag; /* Attribut de comptabilit  */
    char ac_version; /* Vaut toujours ACCT_VERSION (3) */
    u_int16_t ac_tty; /* Terminal de contr le */
    u_int32_t ac_exitcode; /* Code de retour - voir wait(2) */
    u_int32_t ac_uid; /* UID r el */
    u_int32_t ac_gid; /* GID r el */
    u_int32_t ac_pid; /* PID */
    u_int32_t ac_ppid; /* PPID (PID du parent) */
    u_int32_t ac_btime; /* Heure de cr ation du processus */
                        (secondes depuis 01/01/1970) */
    comp_t ac_etime; /* Temps  coul  */
    comp_t ac_utime; /* Temps CPU en mode utilisateur */
    comp_t ac_stime; /* Temps CPU en mode noyau */
    comp_t ac_mem; /* Moyenne m moire utilis e (ko) */
    comp_t ac_io; /* Caract res transmis (inutilis ) */
    comp_t ac_rw; /* Blocs lus ou  crits (inutilis ) */
    comp_t ac_minflt; /* Nb fautes de pages mineures */
    comp_t ac_majflt; /* Nb fautes de pages majeures */
    comp_t ac_swaps; /* Nb de swaps (inutilis ) */
    char ac_comm[ACCT_COMM+1];
                        /* Nom de la commande */
    char ac_pad[X]; /* Remplissage */
};
```

VERSIONS

La structure *acct_v3* est d finie dans la *glibc* depuis la version 2.6.

CONFORMIT 

La comptabilit  des processus provient de BSD. Bien qu'elle soit impl ment e sur la plupart des syst mes, elle n'est pas normalis e et varie parfois entre les syst mes.

NOTES

Les enregistrements dans le fichier de comptabilit  sont class s en fonction de l'heure de fin des processus.

Dans les noyaux jusqu'au 2.6.9 (inclus), un enregistrement s par   tait  crit pour chaque thread utilisant la biblioth que NPTL. Depuis Linux 2.6.10, un enregistrement unique est  crit pour l'ensemble du processus   la fin de son dernier thread.

Le fichier *proc/sys/kernel/acct* d crit dans **proc(5)**, configure le comportement de la comptabilit  des processus lorsque l'espace disque commence   manquer.

VOIR AUSSI

lastcomm(1), **acct(2)**, **accton(8)**, **sa(8)**

TRADUCTION

Ce document est une traduction réalisée par Christophe Blaess <<http://www.blaess.fr/christophe/>> le 2 juillet 2008.

L'équipe de traduction a fait le maximum pour réaliser une adaptation française de qualité. La version anglaise la plus à jour de ce document est toujours consultable via la commande : `< LANG=C man 5 acct >`. N'hésitez pas à signaler à l'auteur ou au traducteur, selon le cas, toute erreur dans cette page de manuel.