

NOM

execl, execlp, execl, execv, execvp - Exécuter un programme.

SYNOPSIS

```
#include <unistd.h>

extern char **environ;

int execl (const char *path, const char *arg, ...);
int execlp (const char *file, const char *arg, ...);
int execl (const char *path, const char *arg , ..., char * const
envp[]);
int execv (const char *path, char *const argv[]);
int execvp (const char *file, char *const argv[]);
```

DESCRIPTION

La famille de fonctions exec remplace l'image mémoire du processus en cours par un nouveau processus. Les fonctions décrites dans cette page sont en réalité des frontaux pour l'appel système execve(2). (Voir la page de execve pour des informations détaillées sur le remplacement du processus en cours.)

L'argument initial de toutes ces fonctions est le chemin d'accès du fichier à exécuter.

Les arguments const char *arg ainsi que les points de suspension des fonctions execl, execlp, et execl peuvent être vues comme des arg0, arg1, ..., argn. Ensemble ils décrivent une liste d'un ou plusieurs pointeurs sur des chaînes de caractères terminées par des caractères nuls, qui constituent les arguments disponibles pour le programme à exécuter. Par convention le premier argument doit pointer sur le nom du fichier associé au programme à exécuter. La liste des arguments doit se terminer par un pointeur NULL.

Les fonctions execv et execvp utilisent un tableau de pointeurs sur des chaînes de caractères terminées par des caractères nuls, qui constituent les arguments disponibles pour le programme à exécuter. Par convention le premier argument doit pointer sur le nom du fichier associé au programme à exécuter. Le tableau de pointeur doit se terminer par un pointeur NULL.

La fonction execl peut également indiquer l'environnement du processus à exécuter en faisant suivre le pointeur NULL qui termine la liste d'arguments, ou le pointeur NULL de la table par un paramètre supplémentaire. Ce paramètre est un tableau de pointeurs sur des chaînes de caractères terminées par des caractères nuls, qui doit se terminer par un pointeur NULL. Les autres fonctions fournissent au nouveau processus l'environnement constitué par la variable externe environ.

Certaines de ces fonctions ont une sémantique spécifique.

Les fonctions execlp et execvp agiront comme le shell dans la recherche du fichier exécutable si le nom fourni ne contient pas de slash (/). Le chemin de recherche est spécifié dans la variable d'environnement PATH. Si cette variable n'est pas définie, le chemin par défaut sera `"/bin:/usr/bin:"`. De plus certaines erreurs sont traitées de manière spécifique.

Si la permission d'accès au fichier est refusée (execve renvoie EACCES), ces fonctions continueront à parcourir le reste du chemin de recherche. Si aucun fichier n'est trouvé, elle reviendront, et errno contiendra le code d'erreur EACCES.

Si l'en-tête d'un fichier n'est pas reconnu (execve renvoie ENOEXEC), ces fonctions exécuteront un shell avec le chemin d'accès au fichier en tant que premier argument. Si ceci échoue, aucune recherche supplémentaire n'est effectuée.

VALEUR RENVOYÉE

Si l'une quelconque des fonctions exec revient à l'appelant, c'est qu'une erreur a eu lieu. La valeur de retour est -1, et errno contient le code d'erreur.

FICHIERS

/bin/sh

ERREURS

Toutes ces fonctions peuvent échouer et positionner errno sur n'importe quelle erreur décrite dans la fonction execve(2).

VOIR AUSSI

sh(1), execve(2), fork(2), environ(5), ptrace(2)

COMPATIBILITÉ

Sur certains systèmes, le chemin de recherche par défaut (employé si l'environnement ne contient pas la variable PATH) contient le répertoire courant à la suite de /bin et /usr/bin, afin d'éviter les chevaux de Troie. Linux utilise encore le répertoire courant en premier dans ce chemin de recherche par défaut.

Le comportement de execlp et execvp lorsqu'une erreur se produit est une pratique historique mais n'a traditionnellement jamais été documenté, ni spécifié dans le standard POSIX. Lorsque l'erreur ETXTBSY se déclenche (fichier exécutable ouvert en écriture par un autre processus), BSD (et peut-être d'autres systèmes) endorment le processus appelant puis réitérent l'opération. Linux considère cette situation comme une erreur grave, et revient immédiatement.

Habituellement, les fonctions execlp et execvp ignoraient toutes les erreurs sauf celles décrites ci-dessus, ENOMEM et E2BIG. Désormais, elles reviennent à l'appelant si une autre erreur que celles-ci se produit.

CONFORMITÉ

execl, execv, execl, execlp et execvp sont conformes à IEEE Std1003.1-88 (''POSIX.1'').

TRADUCTION

Christophe Blaess, 1997.