INTRO(3) Manuel du Programmeur OpenBSD INTRO(3)

NOM

intro - introduction aux bibliothèques C

RÉSUMÉ

cc [*drapeaux*] *fichier* ... [**-lbibliothèque**]

DESCRIPTION

Les pages de manuel de la section 3 fournissent une vue d'ensemble des fonctions de la bibliothèque C, de leurs retours d'erreur, et d'autres définitions et concepts généraux. La plupart de ces fonctions sont disponibles depuis la bibliothèque C, *libc*. D'autres bibliothèques, comme la bibliothèque mathématique, *libm*, doivent être indiquées au moment de la compilation au moyen de l'option **-l** du compilateur.

Voici les diverses bibliothèques (suivies du drapeau de chargeur) :

- libc (**-lc**)

Fonctions de la bibliothèque C standard. Lorsqu'on utilise le compilateur C cc(1), il n'est pas nécessaire de fournir le drapeau de chargeur **-lc** pour ces fonctions. libc comprend en son sein plusieurs “bibliothèques” ou groupes de fonctions : les routines d'E/S standards, les routines de bases de données, les opérateurs de bits, les opérateurs de chaînes, les tests de caractères et les opérateurs de caractères, les routines de chiffrement DES, l'allocation de stockage, les fonctions liées au temps, la gestion des signaux, et plus encore.

- libcompat (**-lcompat**)

Fonctions qui sont obsolètes mais disponibles pour raison de compatibilité avec 4.3BSD. Il y a là en particulier un certain nombre d'interfaces d'appels système fournis dans les précédentes versions de BSD, incluses pour compatibilité du code source. L'usage de ces routines devrait, pour la plupart, être évité. L'entrée de page de manuel de chacune de ces routines de compatibilité indique la bonne interface à utiliser.

- libcrypto (**-lcrypto**)

La bibliothèque de cryptographie OpenSSL. Implémente un certain nombre d'algorithmes cryptographiques, fournissant ainsi des fonctionnalités telles que chiffrement symétrique, cryptographie à clef publique, et gestion de certificat. Voir crypto(3).

- libcurses (**-lcurses**)

libncurses (**-lncurses**)
libtermcap (**-ltermcap**)
libtermlib (**-ltermlib**)

Routines de gestion de l'écran indépendantes du terminal, pour les terminaux à deux dimensions et à affichage non-bitmap. Cette implémentation correspond à “new curses” et fait figure de remplacement de la bibliothèque curses classique de 4.2BSD. Les bibliothèques *libncurses*, *libtermcap* et *libtermlib* sont toutes des liens physiques vers *libcurses*. Cela n'est fait que pour des raisons de compatibilité ; les nouveaux programmes devraient se lier à **-lcurses**. Voir curses(3) et termcap(3).

- libdes (**-ldes**)

Implémentation de l'algorithme de chiffrement DES. Voir des_crypt(3).

- libedit (**-ledit**)

Fonctions génériques d'édition et d'historique de ligne, similaires à celles que l'on trouve dans sh(1). Les fonctions utilisant la bibliothèque *libedit* doivent être liées à la bibliothèque *libcurses*, c'est-à-dire **-ledit -lcurses**. Voir editline(3).

- libevent (**-levent**)

Fournit un mécanisme d'exécution d'une fonction lorsqu'un événement spécifique se produit sur un descripteur de fichier, ou après un laps de temps déterminé. Voir event(3).

- libexpat (**-lexpat**)

Bibliothèque de routines pour le parsing de documents XML.

- libform (**-lform**)

Infrastructures indépendantes du terminal pour la composition d'écran de formulaires sur les terminaux à cellule caractère. Les fonctions utilisant la bibliothèque *libform* doivent être liées à la bibliothèque *libcurses*, c'est-à-dire **-lform -lcurses**. Voir form(3).

- libfrtbegin (**-lfrtbegin**)

libg2c (**-lg2c**)
Routines de support pour les fonctions Fortran. Ces deux bibliothèques ont été séparées pour permettre la liaison de code Fortran avec d'autres langages sur les plateformes ELF. Voir g77(1).

- libgcc (**-lgcc**)

Support à l'exécution de GCC, comprenant l'arithmétique des nombres de type long, propolice, et le support des exceptions indépendantes du langage. Remarque : les utilisateurs n'ont normalement pas à se lier explicitement à cette bibliothèque.

- libgssapi (**-lgssapi**)

L'Interface de Programmation Applicative de Service Générique de Sécurité (Generic Security Service Application Program Interface, GSS-API) fournit des services de sécurité aux appelants, et ce de manière générique. Voir gssapi(3).

- libiberty (**-liberty**)

Ensemble de sous-routines manquantes dans d'autres systèmes d'exploitation, ainsi que le demangler C++ et d'autres fonctions utilisées par la chaîne d'outils GNU.

- libkadm5clnt (**-lkadm5clnt**)

Bibliothèque cliente d'administration de Kerberos, pour parler à une base de données Kerberos. Les clients communiquent via le réseau.

- libkadm5srv (**-lkadm5srv**)

Bibliothèque serveur d'administration de Kerberos, pour parler à une base de données Kerberos. Les serveurs parlent directement à la base de données.

- libkeynote (**-lkeynote**)

Bibliothèque système pour le système de gestion de confiance keynote. Les systèmes de gestion de confiance fournissent des mécanismes standards et à usage général pour spécifier des politiques et des crédits de sécurité au niveau d'une application. Les fonctions utilisant la bibliothèque *libkeynote* doivent être liées aux bibliothèques *libm* et *libcrypto*, c'est-à-dire **-lkeynote -lm -lcrypto**. Voir keynote(3) et keynote(4).

- libkrb5 (**-lkrb5**)

libasn1 (**-lasn1**)
libcom_err (**-lcom_err**)
libhdb (**-lhdb**)
libkafs (**-lkafs**)
Bibliothèques de Kerberos 5. Les bibliothèques *libasn1*, *libcom_err*, *libhdb* et *libkafs* sont toutes des liens physiques vers *libkrb5*. Voir krb5(3).

- libkvm (**-lkvm**)

Bibliothèque d'interface à la mémoire du noyau. Fournit une interface uniforme pour accéder aux images de mémoire virtuelle noyau, y compris les systèmes en fonctionnement et les dumps de crash. Voir kvm(3).

- libl (**-ll**)

libfl (**-lfl**)
La bibliothèque de lex(1), un générateur d'analyseur lexical. La bibliothèque *libfl* est un lien physique vers *libl*.

- libm (**-lm**)

Les fonctions mathématiques qui forment la bibliothèque C math, *libm*. Voir math(3).

- libmenu (**-lmenu**)

Infrastructures indépendantes du terminal pour la composition de menus systèmes sur les terminaux à cellule caractère. Les fonctions utilisant la bibliothèque *libmenu* doivent être liées à la bibliothèque *libcurses*, c'est-à-dire **-lmenu -lcurses**. Voir menu(3).

- libmilter (**-lmilter**)

L'API de filtrage de message de sendmail(8). Voir la documentation dans */usr/share/doc/html/milter/*.

- libobjc (**-lobjc**)

Bibliothèque pour Objective C, un sur-ensemble orienté objet du C ANSI. Utilisez cela pour compiler les programmes Objective C.

- libocurses (**-locurses**)

libotermcap (**-lotermcap**)
Routines pour fournir à l'utilisateur une méthode de mise à jour des écrans avec une optimisation raisonnable. La bibliothèque ocurses(3) est compatible avec la bibliothèque *curses* fournie dans 4.3. *libotermcap* est la bibliothèque *termcap* compatible 4.3, et consiste en un lien physique vers *libocurses*. Voir otermcap(3).

- libossaudio (**-lossaudio**)

Fournit une émulation de l'interface audio OSS (Linux). Cela n'est utilisé que pour le portage de programmes. Voir ossaudio(3).

- libpanel (**-lpanel**)

Infrastructures indépendantes du terminal pour les fenêtres empilées sur les terminaux à cellule caractère. Les fonctions utilisant la bibliothèque *libpanel* doivent être liées à la bibliothèque *libcurses*, c'est-à-dire **-lpanel -lcurses**. Voir panel(3).

- libpcap (**-lpcap**)

Bibliothèque de capture de paquet. Tous les paquets du réseau, même ceux destinés aux autres hôtes, sont accessibles au travers de cette bibliothèque. Voir pcap(3).

- libperl (**-lperl**)

Routines de support pour perl(1).

- libpthread (**-pthread**)

API IEEE Std 1003.1-2001 (``POSIX'') pour les threads et l'ordonnanceur de threads. Les applications threadées devraient utiliser **-pthread** et non **-lpthread**. Voir pthreads(3). Remarque : les utilisateurs n'ont normalement pas à se lier explicitement à cette bibliothèque.

- libreadline (**-lreadline**)

Interface d'édition de ligne de commande. Voir readline(3).

- librpcsvc (**-lrpcsvc**)

Générée par rpcgen(1), contenant de petites fonctions pour de nombreux protocoles rpc(3) communs.

- libsectok (**-lsectok**)

Bibliothèque pour communiquer avec des cartes à puce ISO 7816. Voir sectok(3).

- libskey (**-lskey**)

Bibliothèque de support pour le toolkit d'authentification par mot de passe à usage unique (one time password, OTP) S/Key. Voir skey(3).

- libssl (**-lssl**)

La bibliothèque ssl OpenSSL implémente les protocoles Secure Sockets Layer (SSL v2/v3) et Transport Layer Security (TLS v1). Voir ssl(3).

- libstdc++ (**-lstdc++**)

Bibliothèque de sous-routine de GCC pour C++. Voir c++(1). Remarque : les utilisateurs n'ont normalement pas à se lier explicitement à cette bibliothèque.

- libsupc++ (**-lsupc++**)

(Systèmes GCC 3.3.x seulement.) Support pour le coeur du langage C++ (exceptions, new, typeinfo). Remarque : les utilisateurs n'ont normalement pas à se lier explicitement à cette bibliothèque.

- libusbhid (**-lusbhid**)

Routines pour extraire des données depuis des périphériques USB Human Interface Devices (HIDs). Voir usbhid(3).

- libutil (**-lutil**)

Fonctions système utilitaires. Il s'agit actuellement de check_expire(3), fmt_scaled(3), fparseln(3), getmaxpartitions(3), getrawpartition(3), login(3), login_fbtab(3), opendev(3), opendisk(3), openpty(3), pidfile(3), pw_init(3), pw_lock(3), readlabelfs(3) et uucplock(3).

- libwrap (**-lwrap**)

Bibliothèque de contrôle d'accès aux TCP wrappers. Voir hosts_access(3) et rfc1413(3).

- liby (**-ly**)

La bibliothèque pour yacc(1), un générateur de parser LALR.

- libz (**-lz**)

Bibliothèque de compression de données à usage général. Les fonctions de cette bibliothèque sont documentées dans compress(3). Le format de données est décrit dans les RFC 1950 à 1952.

Bibliothèques spécifiques à une plateforme :

- libalpha (**-lalpha**)

Fonctions d'E/S et d'accès mémoire pour Alpha. Voir inb(2).

- libamd64 (**-lamd64**)

Fonctions d'E/S et d'accès mémoire pour AMD64. Voir amd64_get_ioperm(2) et amd64_iopl(2).

- libarm (**-larm**)

Fonctions d'E/S et d'accès mémoire pour ARM. Voir arm_drain_writebuf(2) et arm_sync_icache(2).

- libi386 (**-li386**)

Fonctions d'E/S et d'accès mémoire pour i386. Voir i386_get_ioperm(2), i386_get_ldt(2), i386_iopl(2) et i386_vm86(2).

TYPES DE BIBLIOTHÈQUES

Les bibliothèques système sont situées dans */usr/lib*. Habituellement, une bibliothèque présentera quelques variantes :

```
        libc.a
        libc.so.30.1
        libc_p.a
        libc_pic.a
```

Les bibliothèques portant le suffixe “.a” sont statiques. Si on lie un programme à une telle bibliothèque, alors tout le code de la bibliothèque sera lié dans le binaire. Cela signifie que le binaire peut être lancé même si les bibliothèques ne sont pas disponibles. Mais cela peut être inefficace en terme d'utilisation de la mémoire. On peut demander au compilateur C, cc(1), de lier statiquement en spécifiant le drapeau **-static**.

Les bibliothèques portant un suffixe “.so.X.Y” sont des bibliothèques dynamiques. Quand le code est compilé dynamiquement, le code de la bibliothèque dont l'application a besoin n'est pas lié dans le binaire. Au lieu de cela, on ajoute des structures de données contenant des informations sur les bibliothèques dynamiques avec lesquelles il faut se lier. Quand le binaire est exécuté, l'éditeur de lien à l'exécution ld.so(1) lit ces structures de données, et les charge à une adresse virtuelle en utilisant l'appel système mmap(2).

“X” représente le numéro majeur de la bibliothèque, et “Y” représente le numéro mineur. En général, un binaire sera capable d'utiliser une bibliothèque dynamique portant un numéro mineur différent, mais le numéro majeur doit correspondre. Dans l'exemple ci-dessus, un binaire lié à une bibliothèque de numéro mineur “3” pourrait être lié à libc.so.30.1, tandis qu'un binaire lié une bibliothèque de numéro majeur “31” ne le pourrait pas.

Les avantages des bibliothèques dynamiques sont que plusieurs instances du même programme peuvent partager l'espace d'adressage, et que la taille physique du binaire est plus réduite. Les inconvénients sont la complexité ajoutée qui découle du chargement dynamique des bibliothèques, et le temps supplémentaire pris pour charger les bibliothèques. Bien entendu, si les bibliothèques ne sont pas disponibles, le binaire ne pourra pas s'exécuter. On peut demander au compilateur C, cc(1), de lier statiquement en spécifiant le drapeau **-shared**, bien que sur les systèmes qui le supportent ce sera le comportement par défaut, et ce ne sera donc pas nécessaire de le spécifier.

Les bibliothèques portant le suffixe “_p.a” sont des bibliothèques de profilage. Elles contiennent des informations supplémentaires et convenant aux programmes qui analysent des indicateurs tels que la vitesse d'exécution ou les compteurs d'appels. Cela peut alors être interprété par des utilitaires tels que gprof(1). On peut demander au compilateur C, cc(1), de générer le code de profilage, ou de se

lier aux bibliothèques de profilage, en spécifiant le drapeau **-pg**.

Les bibliothèques portant le suffixe “_pic.a” contiennent du code indépendant de la position (position-independent code, PIC). Normalement, les compilateurs produisent du code relocalisable. Le code relocalisable doit être modifié à l'exécution, selon l'endroit en mémoire où il sera lancé. Le code PIC n'a pas besoin d'être modifié à l'exécution, mais il est moins efficace que le code relocalisable. Le code PIC est utilisé par les bibliothèques partagées, ce qui peut les rendre plus lentes. On peut demander au compilateur C, cc(1), de générer du code PIC ou de se lier aux bibliothèques PIC en spécifiant les drapeaux **-fpic** ou **-fPIC**.

À l'exception des bibliothèques dynamiques, les bibliothèques sont générées en utilisant l'utilitaire ar(1). Les bibliothèques renferment un index de leur contenu, stocké en son sein même. L'index dresse la liste de chaque symbole défini par un membre de la bibliothèque, c'est-à-dire un fichier objet relocalisable. Cela accélère le liage à la bibliothèque, et permet aux routines de la bibliothèque de s'appeler les unes les autres quelque soit leur placement à l'intérieur de la bibliothèque. L'index est créé par ranlib(1) et peut être visionné en utilisant nm(1).

La construction des versions PIC des bibliothèques et des bibliothèques dynamiques peut être empêchée en fixant la variable NOPIC dans */etc/mk.conf*. La construction des versions de profilage des bibliothèques et/ou des bibliothèques dynamiques peut être empêchée en fixant la variable NOPROFILE dans */etc/mk.conf*. Voir mk.conf(5) pour de plus amples détails.

FICHIERS

```
/usr/lib/libasn1.a
/usr/lib/libc.a
/usr/lib/libcom_err.a
/usr/lib/libcompat.a
/usr/lib/libcrypto.a
/usr/lib/libcurses.a
/usr/lib/libdes.a
/usr/lib/libedit.a
/usr/lib/libevent.a
/usr/lib/libexpat.a
/usr/lib/libfl.a
/usr/lib/libform.a
/usr/lib/libfrtbegin.a
/usr/lib/libg2c.a
/usr/lib/gcc-lib/${ARCH}-unknown-openbsd${OSREV}/${GCCREV}/fpic/libgcc.a
/usr/lib/libgssapi.a
/usr/lib/libhdb.a
/usr/lib/libiberty.a
/usr/lib/libkadm5clnt.a
/usr/lib/libkadm5srv.a
/usr/lib/libkafs.a
/usr/lib/libkeynote.a
/usr/lib/libkrb5.a
/usr/lib/libkvm.a
/usr/lib/libl.a
/usr/lib/libm.a
```

```
/usr/lib/libmenu.a
/usr/lib/libmilter.a
/usr/lib/libncurses.a
/usr/lib/libobjc.a
/usr/lib/libocurses.a
/usr/lib/libossaudio.a
/usr/lib/libotermcap.a
/usr/lib/libpanel.a
/usr/lib/libpcap.a
/usr/lib/libperl.a
/usr/lib/libpthread.a
/usr/lib/libreadline.a
/usr/lib/librpcsvc.a
/usr/lib/libsectok.a
/usr/lib/libskey.a
/usr/lib/libssl.a
/usr/lib/libstdc++.a
/usr/lib/libsupc++.a
/usr/lib/libtermcap.a
/usr/lib/libtermlib.a
/usr/lib/libusbhid.a
/usr/lib/libutil.a
/usr/lib/libwrap.a
/usr/lib/liby.a
/usr/lib/libz.a
```

```
/usr/lib/libalpha.a
/usr/lib/libamd64.a
/usr/lib/libarm.a
/usr/lib/libi386.a
```

VOIR AUSSI

ar(1), c++(1), cc(1), g77(1), gcc-local(1), gprof(1), ld(1), ld.so(1), lex(1), nm(1), perl(1), ranlib(1), yacc(1), intro(2), compress(3), crypto(3), curses(3), des_crypt(3), editline(3), event(3), form(3), hosts_access(3), keynote(3), kvm(3), math(3), menu(3), ocurses(3), ossaudio(3), panel(3), pcap(3), pthreads(3), readline(3), rfc1413(3), rpc(3), sectok(3), skey(3), ssl(3), stdio(3), termcap(3), usbhid(3), keynote(4), mk.conf(5)

HISTORIQUE

Une page de manuel **intro** est apparue dans AT&T UNIX Version 7.

OpenBSD 4.4 26 Août 2008 6

From:
http://www.mouet-mouet.net/doku.php/doku.php/ - **mouet-mouet !**

Permanent link:
http://www.mouet-mouet.net/doku.php/doku.php/doku.php?id=maxime:openbsd:manpages-fr:3:intro

Last update: **2021/10/08 00:17**