

---

# find

Outil de recherche de fichiers

## OPTIONS

- P** Ne jamais suivre les liens. C'est le mode par défaut.
- L** Suit les liens symboliques. cette option implique -noleaf. le prédicat -type matche toujours le type de fichier pointé par le lien
- H** Ne pas suivre les liens symboliques, excepté en traitant les arguments de la ligne de commande.
- D debugopts** Affiche des informations de diagnostique ; peut être utile pour diagnostiquer un problème. Les options valides sont :
  - exec** informations liées à -exec, -execdir, -ok et -okdir
  - opt** informations liées à l'optimisation de l'arborescence d'expression
  - rates** sommaire indiquant la fréquence de réussite et d'échec de chaque prédicat
  - search** Navigue dans l'arborescence en mode verbeux
  - stat** affiche des messages quand les fichiers sont examinés avec les appels système stat et lstat.
  - tree** Affiche l'arborescence d'expression sous sa forme originale et optimisée
- Olevel** Active l'optimisation de requête. find réordonne les tests pour accélérer l'exécution tout en préservant l'effet général, c'est à dire que les prédicats ayant des conséquences ne sont pas réordonnés relativement entre eux. Les optimisations sont formés à chaque niveau d'optimisation comme suit :
  - 0** Équivalent à l'optimisation 1
  - 1** Niveau par défaut. Les expressions sont réordonnés pour que les tests basés seulement sur les noms des fichiers (par exemple -name et -regex) sont effectués en premier
  - 2** Les tests -type ou -xtype sont effectués après les tests basés seulement sur les noms des fichiers, mais avant les tests qui nécessitent des informations sur l'inode. Dans de nombreux Unix récents, les types de fichiers sont retournés par readdir() et ces prédicats sont plus rapides à évaluer.
  - 3** L'ordre des tests est modifié pour que les tests rapides soient effectués en premier et les plus coûteux en derniers.

## Expression

La partie de la ligne de commande après la liste des points de départ est l'expression. C'est le type de spécification de requête décrivant comment les fichiers correspondent et ce que l'on fait avec ces fichiers. Une expression est composée d'une séquence d'éléments :

- tests** Les tests retournent vrai ou faux, généralement sur la base de propriétés d'un fichier.
- actions** Les actions ont des effets supplémentaire (comme afficher quelque chose) et retournent vrai ou faux.
- Options globales** Les options globales affectent les opérations de tests et les actions. Les options globales retournent toujours vrai.
- Options positionnelles** Affectent seulement les tests ou actions qui les suivent. Retournent toujours vrai.
- Opérateurs** Les opérateurs joignent des éléments dans l'expression.

## Options positionnelles

- 
- daystart** Mesure de temp (pour -amin, -atime, -cmin, -mmin, et -mtime) depuis le début du jour au-lieu des dernières 24 heures. N'affecte que les tests qui apparaissent ultérieurement dans la ligne de commande.
  - regex type** Change la syntaxe de l'expression régulière comprises par les tests -regex et -iregex apparaissant plus loin sur la ligne de commande. -regex type help affiche les types d'expression régulières connus.
  - warn, -nowarn** Active/désactive les messages d'alerte. Ces warning ne s'appliquent qu'à l'utilisation de la ligne de commande.

## Options globales

- d, -depth** Traite chaque contenu de répertoire avant le répertoire lui-même. -delete implique -depth
- ignore\_readdir\_race** N'affiche pas de messages d'erreur quand find ne parvient pas à obtenir les statistiques d'un fichier.
- maxdepth levels** niveau de sous-répertoires maximum à parcourir sous les points de départs. 0 signifie seulement le point de départ.
- mindepth levels** N'applique pas d'action ou test aux niveaux inférieurs. 1 signifie de traiter tous les fichiers excepté au point de départ.
- mount, -xdev** Ne descend pas les répertoires dans d'autres systèmes de fichier.
- noignore\_readdir\_race** Désactive l'effet de -ignore\_readdir\_race
- noleaf** N'optimise pas en assumant que les répertoires contenant 2 sous-répertoires de moins que leur nombre de lien dur. Cette option est nécessaire lors de recherches dans des systèmes de fichier qui ne suivent pas les conventions UNIX, tel que les systèmes de fichier CD-ROM, MS-DOS ou AFS.

## Tests

- +n** plus grand que n
- n** plus petit que n
- n** exactement n
- amin n** le fichier a été accédé il y a n minutes
- anewer file** le dernier accès est plus récent de sa dernière modification
- atime n** le fichier a été accédé il y a n\*24 heures. +1 signifie un accès il y a au moins 2 jours.
- cmin n** Le status du fichier a été changé il y a n minutes
- cnewer file** Le status du fichier a été changé plus récemment que sa modification. Si le fichier est un lien symbolique et que l'option -H ou -L est présente, le temps du fichier pointé est toujours utilisé
- ctime n** Le status du fichier a été changé il y a n\*24 heure.
- empty** le fichier est vide et est un fichier régulier ou un répertoire
- executable** le bit de permission est mis. Prend en compte les listes de contrôles d'accès et autres permissions que -perm ignore. Utilise access(2)
- false** Toujours faux
- fstype type** Le fichier est sur système de fichier de type spécifié.
- gid n** Le GID du fichier est n
- group name** le groupe propriétaire est le nom spécifié
- iname pattern** Comme -lname, mais le match est insensible à la casse. Si -L est présent, ce test retourne faux sauf si le lien symbolique est cassé
- iname pattern** comme -name, mais de match est insensible à la casse.
- inum n** le fichier a le numéro d'inode n
- ipath pattern, -iwholename pattern** comme -path, mais le match est insensible à la casse.
- iregex pattern** comme -regex, mais le match est insensible à la casse.
- links n** le fichier a n liens

- 
- lname pattern** le fichier est un lien symbolique dont le contenu match le motif. Si -L est présent, ce test retourne faux sauf si le lien symbolique est cassé
  - mmin n** Les données du fichier ont été modifiés il y a n minutes
  - mtime n** Les données du fichier ont été modifiés il y a n\*24 heures.
  - name pattern** La base du nom de fichier ( le chemin avec les répertoires enlevés) correspond au motif.
  - newer file** Le fichier a été modifié plus récemment que le fichier. avec un lien symbolique et -L ou -H, utilise la cible.
  - newerXY reference** Réussis si l'horodatage X du fichier à considérer est plus récent que l'horodatage Y du fichier référence. Les lettres X et Y peuvent être une des lettres suivantes :
    - a** atime du fichier référence
    - B** Date de création du fichier référence
    - c** ctime du fichier référence
    - m** mtime du fichier référence
    - t** Référence est interprétée directement comme un temps
  - nogroup** Aucun groupe ne correspond au GID du fichier
  - nouser** Aucun utilisateur ne correspond au UID du fichier
  - path pattern, -wholenamename pattern** Le nom du fichier matche le pattern.
  - perm mode** Les bits de permission du fichier sont exactement le mode spécifié.
  - perm -mode** Tous les bits de permissions sont définis pour le fichier.
  - perm /mode** Un des bits de permission sont définis pour le fichier
  - readable** Match les fichiers accessibles en lecture. Prend en compte les listes de contrôle d'accès et d'autres permissions
  - regex pattern** Le nom du fichier match le motif. C'est un match sur le chemin entier
  - samefile name** Le fichier réfère au même inode que name. avec -L, inclus les liens symboliques
  - size n [cwbkMG]** Le fichier utilise n unités d'espace, arrondi. Les suffixes suivants peuvent être utilisés :
    - b** block de 512 octets
    - c** Octet
    - w** Mots de 2 octets
    - k** Kio
    - M** Mio
    - G** Gio
  - true** Toujours vrai
  - type c** Le fichier est de type c (utiliser des virgules pour spécifier plusieurs types) :
    - b** Périphérique block
    - c** Périphérique caractère
    - d** répertoire
    - p** pipe nommé
    - f** fichier régulier
    - l** liens symboliques
    - s** socket
  - uid n** L'UID du fichier est n
  - used n** Le fichier a été accédé n jours après que son status ait été changé
  - user uname** Le propriétaire du fichier est uname (UID permis)
  - writable** Matche les fichiers accessible en écriture. Prend en compte les listes de contrôle d'accès et d'autres permissions
  - xtype c** Idem à -type sauf si le fichier est un lien symbolique. Pour ces liens : si -H ou -P est spécifié, vrai si le fichier est un lien vers un fichier de type c ; si -L est donné, vrai si c est 'l'.
  - context pattern** matche le contexte de sécurité du fichier.

---

# Actions

- delete** Supprime des fichiers; vrai si la suppression réussie.
- exec command**; Exécute la commande. vrai si 0 est retourné. La chaîne '{}' est remplacée par le nom du fichier courant.
- exec command {} +** Variante, mais la ligne de commande est construite en ajoutant chaque fichier sélectionné à la fin. Seule une instance de '{}' est permise. Retourne toujours vrai
- execdir command**; **-execdir command {} +** comme -exec, mais la commande est lancée pour le sous-répertoire contenant le fichier correspondant, qui est normalement le répertoire courant. C'est une méthode plus sécurisée pour invoquer des commandes. Comme avec -exec, la forme '+ ' permet de traiter plusieurs fichiers à la fois.
- fs file** Vrai; comme -ls mais écrit dans fichier comme -fprint. le fichier de sortie est toujours créé, même si le prédicat n'est jamais matché.
- fprint file** Vrai; affiche le nom du fichier dans le fichier spécifié. Si le fichier n'existe pas, il est créé. S'il existe, il est tronqué. /dev/stdout et /dev/stderr sont gérés spécialement
- fprint0 file** Vrai; comme -print0 mais écrit dans le fichier comme -fprint.
- fprintf file format** Vrai; comme -printf mais écrit dans le fichier comme -fprint.
- ls** Vrai; liste les fichiers courants au format ls -dils sur la sortie standard. Les compteurs de block sont des blocks de 1K, sauf si POSIXLY\_CORRECT est définis, auquel cas les blocks de 512 octets sont utilisés
- ok command**; Comme -exec mais demande à l'utilisateur avant.
- okdir command**; comme -execdir, mais demande à l'utilisateur avant.
- print** Vrai; affiche le nom du fichier sur la sortie standard, suivi d'une nouvelle ligne, à la différence de print0
- print0** Vrai; affiche le nom de fichier sur la sortie standard, suivi par un caractère null. Correspond à l'option -0 de xargs
- printf format** Vrai; affiche le format sur la sortie standard. Interprète les directives '\ ' et '% '.

**\a** alert (bell)

**\b** backspace, ascii 8

**\c** n'affiche pas la suite

**\f** form feed

**\n** nouvelle ligne

**\r** retour charriot

**\t** tabulation horizontale

**\v** Tabulation verticale

**\0** ASCII NUL

**\** le caractère '\ '

**\nnn** Le caractère 8bits dont la valeur est nnn en octal

**%%** le caractère %

**%a** Date de dernier accès au fichier, au format retourné par la fonction ctime

**%Ak** Date de dernier accès au fichier, au format spécifié par k

**@** Secondes depuis le 1 Janvier 1970, 00 :00 GMT, avec partie fractionnelle.

**H** Heure (00..23)

**I** Heure (01..12)

**k** Heure (0..23)

**l** heure (1..12)

**M** minute (00..59)

**p** AM ou PM local

**r** Temps, 12 heures (hh :mm :ss [AP]M)

**S** Secondes (00.00 .. 61.00). Il y a une partie fractionnelle

**T** Heure, 24 heure (hh :mm :ss.xxxxxxxxx)

**+** Date et heure, séparé par un '+', par exemple '2010-04-23+17 :45 :45.0'.

---

**X** Temps local (H :M :S). Le second champ inclus une partie fractionnelle

**Z** Timezone

**a** Nom de la semaine abrégée (Sun..Sat)

**A** Nom complet de la semaine (Sunday..Saturday)

**b** Nom du mois abrégée (Jan..Dec)

**B** Nom complet du mois (January..December)

**c** Date et heure locale (Sat Nov 04 12 :02 :33 EST 1989). Le format est le même que pour `ctime(3)`

**d** Jour du mois (01..31)

**D** date (mm/dd/yy)

**h** Idem à b

**j** Jour de l'année (001..366)

**m** mois (01..12)

**U** Numéro de la semaine avec dimanche comme premier jour de la semaine (00..53)

**w** Jour de la semaine (0..6)

**W** Numéro de semaine avec lundi comme premier jour de la semaine (00..53)

**x** Représentation de date locale (mm/dd/yy)

**y** 2 derniers chiffres de l'année (00..99)

**Y** Année (1970..)

**%b** quantité d'espaces utilisé par ce fichier en blocks de 512 octets.

**%c** Date de dernier changement du status du fichier au format retourné par la fonction `ctime`

**%Ck** Date de dernier changement du status du fichier au format spécifié par k

**%d** Profondeur du fichier dans l'arborescence des répertoires, 0 signifiant le point de départ

**%D** Numéro de périphérique sur lequel le fichier existe ( le champ `st_dev` de la struct `stat`), en décimal.

**%f** Nom du fichier avec les répertoires enlevés (`basename`)

**%F** Type de système de fichier sur lequel le fichier réside

**%g** Nom du groupe du fichier

**%G** GID du fichier

**%h** Chemin du fichier (`dirname`)

**%H** Point de départ sous lequel le fichier a été trouvé

**%i** numéro d'inode du fichier

**%k** Quantité d'espaces disque utilisé par ce fichier en blocks de 1Kio.

**%l** Objet de lien symbolique (chaîne vide si le fichier n'est pas un lien symbolique)

**%m** bits de permissions du fichier en octal.

**%M** Permissions du fichier sous forme symbolique, comme `ls`.

**%n** Nombre de liens dur vers le fichier

**%p** Nom du fichier

**%P** Nom du fichier avec le nom du point de départ sous lequel il a été trouvé supprimé

**%s** Taille du fichier en octets

**%S** `BLOCKSIZE*st_blocks / st_size`. fichier sans `sparse`.

**%t** Date de dernière modification du fichier au format retourné par la fonction `ctime`

**%Tk** Date de dernière modification du fichier au format k

**%u** Nom propriétaire du fichier

**%U** UID du propriétaire du fichier

**%y** Type du fichier, comme `ls -l`. U=unknown type

**%Y** Type de fichier, et suit les liens symboliques L=loop, N=nonexistent

**%Z** Contexte de sécurité du fichier

---

`%{ % [ %(` Réserve

**-prune** Vrai ; si le fichier est un répertoire, ne descend pas dedans. si `-depth` est donné, false.

**-quit** Quitte immédiatement.

## Opérateurs

( **expr** ) Forcer la précedence. Vu que les parenthèse sont spéciales, il est nécessaire des les échapper : `\(...)`

**! EXPR** vrai si EXPR est faux

**-not expr** Idem à !, mais non conforme POSIX

**expr1 expr2** 2 expressions sur une ligne sont jointes avec un AND implicite ; expr2 n'est pas évalué si expr1 est faux

**expr1 -a expr2** idem

**expr1 -and expr2** idem, mais non conforme POSIX

**expr1 -o expr2** OU ; expr2 n'est pas évalué si expr1 est vrai

**expr1 -or expr2** idem, mais non conforme POSIX

**expr1 , expr2** Liste, les 2 expressions sont toujours évaluées. La valeur de expr1 est ignorée ; la valeur de la liste est la valeur de expr2.

## Noms de fichier non-usuels

De nombreuses actions de find résultent dans l'affichage de données qui sont sous le contrôle d'autres utilisateurs. Cela inclus les noms de fichiers, tailles, dates de modification, etc. Les noms de fichier sont potentiellement un problème vu qu'ils peuvent contenir n'importe quel caractère excepté `\0` et `/`. Les caractères non-usuels peuvent créer des effets indésirables. (par exemple, changer les paramètres des fonctions clé de certains terminaux).

**-print0, -fprint0** Affiche toujours le nom de fichier exacte, inchangé, même si la sortie va dans un terminal.

**-ls, -fls** Les caractères non usuels sont toujours échappés, les espaces blanc échappés, et les guillemets doubles sont affichés en style C échappé.

**-printf, -fprintf** Si la sortie n'est pas un terminal, elle est affichée tel quelle. Sinon, le résultat dépend des directives utilisées. `%D`, `%F`, `%g`, `%G`, `%H`, `%Y`, et `%y` expandent aux valeurs qui sont sous le contrôle du propriétaire du fichier, et sont affichés tel quels. `%a`, `%b`, `%c`, `%d`, `%i`, `%k`, `%m`, `%M`, `%n`, `%s`, `%t`, `%u` et `%U` ont des valeurs sous le contrôle du propriétaire du fichier mais qui ne peuvent pas être utilisés pour envoyer des données arbitraires au terminal, et sont donc affichés tel quels. `%f`, `%h`, `%l`, `%p` et `%P` sont quotés.

**-print, -fprint** Le quoting est géré de la même manière que pour `-printf` et `-fprintf`.

**-ok et -okdir** Affiche les noms de fichier tels quels

## Variables d'environnement

**LANG** Spécifie la locale

**LC\_ALL** Spécifie la locale

**LC\_COLLATE** Spécifie la locale pour la catégorie LC\_COLLATE

**LC\_CTYPE** Langage à utiliser pour déterminer le jeu de caractères

**LC\_MESSAGES** Spécifie la locale pour la catégorie LC\_MESSAGES

**NLSPATH** Détermine l'emplacement des catalogues de message d'internationalisation

**PATH** Les chemins de recherche pour les commandes

**POSIXLY\_CORRECT** Strictement Conforme à POSIX

---

**TZ** Affecte le timezone utilisé

## Exemples

Trouver les fichiers nommés core sous /tmp et les supprimer

```
find /tmp -name core -type f -print | xargs /bin/rm -f
```

Lancer file sur tous les fichiers sous le répertoire courant.

```
find . -type f -exec file '{}' \;
```

traverser le système de fichier, lister les fichiers setuid et les répertoires dans /root/suid.txt et les grands fichiers dans /root/big.txt

```
find /\( -perm -4000 -fprintf /root/suid.txt '%#m %u %p\n' \) , \(-size +100M -fprintf /root/big.txt '%-10s %p\n' \)
```

Recherche les fichiers dans le home qui ont été modifiés dans les dernière 24 heures

```
find $HOME -mtime 0
```

Rechercher les fichiers qui ont les permissions de lecture et écriture pour leur propriétaire, groupe, mais seulement en lecture pour les autres.

```
find . -perm 664
```

Rechercher les fichier en écriture pour tout le monde

```
find . -perm /222
```

```
find . -perm /220
```

```
find . -perm /u+w,g+w
```

```
find . -perm /u=wg=w
```

Rechercher les fichier en lecture pour tout le monde, ayant au moins un bit d'écriture mis, mais non exécutable

```
find . -perm -444 -perm /222! -perm /111
```

```
find . -perm -a+r -perm /a+w! -perm /a+x
```

Copier le contenu de /source-dir dans /dest-dir, mais omet les fichiers et répertoires nommés .snapshot et leur contenu. Omet également les fichier ou répertoires se terminant par ~, mais pas leur contenu.

```
cd /source-dir ; find . -name .snapshot -prune -o \( ! -name *~ -print0 \) | cpio -pmd0 /dest-dir
```

## Codes de sortie

find quitte avec un status de 0 si tous les fichier ont été traité avec succès, Supérieur à 0 si une erreur se produit.