

Référence rapide grep

Correspondance de motifs, regex, recherche récursive, contexte, filtrage

Utilisation de base

Lancer grep

```
grep "pattern" file.txt # chercher dans un fichier
grep "error" *.log     # chercher dans plusieurs
                        # fichiers
grep "hello" file1.txt file2.txt # liste de fichiers explicite
cat file.txt | grep "pattern" # entrée par tube
dmesg | grep -i "usb"     # filtrer la sortie d'une
                        # commande
```

Options courantes

- i** Correspondance insensible à la casse
- v** Inverser — afficher les lignes non correspondantes
- c** Afficher le nombre de lignes correspondantes
- n** Afficher les numéros de ligne
- l** Lister uniquement les noms de fichiers avec des correspondances
- L** Lister les noms de fichiers sans correspondance
- w** Correspondance de mots entiers uniquement
- x** Correspondance de lignes entières uniquement

Motifs regex

Expressions régulières de base (BRE)

- .** N'importe quel caractère unique
- *** Zéro ou plusieurs occurrences de l'élément précédent
- ^** Début de ligne
- \$** Fin de ligne
- [abc]** Classe de caractères — l'un de a, b, c
- [^abc]** Classe niée — tout sauf a, b, c
- [a-z]** Plage — lettres minuscules
- \<, \>** Limites de mot (GNU)
- \(\), \1** Groupe de capture et rétro-référence

Exemples BRE

```
grep '^#' file.conf # lignes commençant par #
grep 'error$' file.log # lignes finissant par error
grep '^$' file.txt # lignes vides
grep 'col[ou]r' file.txt # correspondre color ou colour
```

Regex étendue

Expressions régulières étendues (ERE)

- +** Un ou plusieurs de l'élément précédent
- ?** Zéro ou un de l'élément précédent
- {n}** Exactement n répétitions
- {n,m}** Entre n et m répétitions
- (a|b)** Alternance — correspondre a ou b
- ()** Regroupement (sans barre oblique inverse)

Exemples ERE

```
grep -E '[0-9]{3}-[0-9]{4}' f # motif de numéro de téléphone
grep -E '(error|warn|fatal)' f # plusieurs motifs
grep -E '[A-Z][a-z]+' f # mots commençant par une
                        # majuscule
grep -P '\d{1,3}\.\d{1,3}' f # regex Perl : fragments IP
```

Lignes de contexte

Exemples de contexte

```
grep -B 3 "error" app.log # 3 lignes avant la
                        # correspondance
grep -A 5 "FAIL" test.log # 5 lignes après la
                        # correspondance
grep -C 2 "crash" kern.log # 2 lignes avant et après
                        # correspondance
grep --group-separator="---" -C 1 "err" f # séparateur personnalisé
```

Options de contexte

- B N** Afficher N lignes avant chaque correspondance
- A N** Afficher N lignes après chaque correspondance
- C N** Afficher N lignes avant et après (contexte)
- group-separator=str** Séparateur entre les groupes (par défaut - -)
- color=auto** Mettre en surbrillance les correspondances dans le terminal

Recherche récursive

Exemples récurifs

```
grep -r "TODO" . # récurif depuis le répertoire
                # courant
grep -rn "FIXME" src/ # récurif avec numéros de ligne
grep -r --include="*.py" "import" . # uniquement les fichiers .py
grep -r --exclude="*.log" "error" . # ignorer les fichiers .log
grep -r --exclude-dir=node_modules "require" .
```

Options récursives

- r / --recursive** Chercher dans les répertoires récursivement
- R** Comme **-r** mais suit les liens symboliques
- include=glob** Chercher uniquement les fichiers correspondant au glob
- exclude=glob** Ignorer les fichiers correspondant au glob
- exclude-dir=dir** Ignorer les répertoires correspondant au nom
- include-dir=dir** Chercher uniquement dans les répertoires correspondants

Comptage et listage

Exemples de comptage et listage

```
grep -c "error" *.log # compter les correspondances par
                    # fichier
grep -l "TODO" src/*.py # lister les fichiers avec des
                    # TODO
grep -L "test" src/*.py # fichiers sans "test"
grep -o "http[^ ]*" page.html # extraire uniquement les parties
                    # correspondantes
grep -c ' ' file.txt # compter le total des lignes
                    # (comme wc -l)
```

Options de sortie

- c** Afficher le nombre de lignes correspondantes par fichier
- l** Afficher uniquement les noms des fichiers avec des correspondances
- L** Afficher uniquement les noms des fichiers sans correspondances
- o** Afficher uniquement les parties correspondantes des lignes
- H / -h** Afficher / masquer le préfixe du nom de fichier
- Z** Sortie délimitée par null (pour xargs -0)

Correspondance inversée

Inversion et exclusion

```
grep -v "^#" config.conf # supprimer les lignes de
                        # commentaires
grep -v "^$" file.txt # supprimer les lignes vides
grep -v -e "debug" -e "trace" app.log # exclure deux motifs
grep -v "pattern" f | grep "other" # chaîne : PAS A, puis B
```

Stratégies de filtrage

- v** Inverser — sélectionner les lignes non correspondantes
- v avec -e pipe chain** Exclure plusieurs motifs
Chaîtrer les appels grep pour un filtrage complexe
- grep -v '^\$' | grep -v '^#'** Supprimer les lignes vides et les commentaires
- v avec -c** Compter les lignes non correspondantes

Motifs multiples

Exemples de motifs multiples

```
grep -e "error" -e "warning" app.log
grep -E "error|warning|fatal" app.log
grep -f patterns.txt file.txt # motifs depuis un fichier
grep -w -e "GET" -e "POST" access.log
```

Options de motifs

- e pattern** Spécifier un motif (utilisable plusieurs fois)
- f file** Lire les motifs depuis un fichier (un par ligne)
- E 'a|b|c'** Alternance ERE pour plusieurs motifs
- F** Chaînes fixes — pas de regex, correspondance plus rapide
- G** Regex de base (mode par défaut)
- P** Regex compatible Perl (PCRE)

Performance

Conseils de performance

- F (fgrep)** Mode chaîne fixe — le plus rapide pour les littéraux
- LC_ALL=C grep** Contourner la locale pour un gain de vitesse 2-10x sur les données ASCII
- include/--exclude** Réduire les fichiers cherchés avant ouverture
- m N** S'arrêter après N correspondances par fichier
- q** Mode silencieux — quitter à la première correspondance (pour scripts)
- ripgrep (rg)** Remplacement compatible ; plus rapide sur les grands dépôts

Exemples de performance

```
LC_ALL=C grep -F "exact string" huge.log
grep -r -m 1 "needle" /var/log/ # s'arrêter après le premier
                        # résultat
grep -rq "pattern" . && echo "found" # test booléen
grep -r --include="*.go" "func main" .
```

Modèles courants

Lignes de commande utiles

```
grep -rn "TODO\|FIXME\|HACK" src/ # trouver les marqueurs de
                        # code
grep -oP '(?<=)[^]+(?=)' f # extraire les chaînes entre
                        # guillemets
grep -E '^$' f | wc -l # compter les lignes vides
grep -c ' ' *.py | sort -t: -k2 -rn # trier les fichiers par
                        # nombre de lignes
grep -rn --include="*.yaml" "password" . # auditer pour les secrets
```

Référence rapide grep

Recettes

Adresses IP	<code>grep -oE '[0-9]{1,3}(\.[0-9]{1,3}){3}'</code>
Adresses email	<code>grep -oE '[a-zA-Z0-9._%+-]+@[a-z.-]+'</code>
URL	<code>grep -oE 'https?://[^\]+'</code>
Lignes entre marqueurs	<code>grep -A999 'START' f grep -B999 'END'</code>
Correspondances uniques	<code>grep -oE 'pattern' f sort -u</code>
Comptage par motif	<code>grep -c 'pat1' f; grep -c 'pat2' f</code>