

9.2. grep, egrep, fgrep, gnu-grep, zgrep, zipgrep
=====

Die Kunst zu suchen oder suchet so werdet ihr finden.

Grep gibt es in den verschiedensten Ausführungen und Spielarten. Die verschiedenen "greps" unterscheiden sich in Optionen und in der Funktionalität und damit in der Geschwindigkeit des Suchens. Man sollte beim Durchmustern großer Datenbestände sich das für den jeweiligen Suchvorgang passende "grep" aussuchen!!

```
grep - Pattern mit einfachen regulären Ausdrücken
egrep - Pattern mit vollen regulären Ausdrücken
fgrep - Pattern mit Strings
gnu-grep - abhängig von Optionen
```

Zusätzlich gibt es noch Scripte (zgrep, zipgrep), die den Einsatz von grep bei gepackten Files unterstützen

In den verschiedenen Betriebssystemen ist oftmals unter dem gleichen Namen in verschiedenen Directories ein unterschiedliches "grep" versteckt.

z.B. Solaris:

```
/usr/bin/grep
/usr/xpg4/bin/grep,
/usr/local/bin/grep)
```

j-p bell

Seite 1

Allgemeine Funktionalität

- Die Programme der grep-Familie durchsuchen Files nach Pattern und geben entweder die Zeilen, die die Pattern enthalten oder die Zeilen, die die Pattern nicht enthalten, aus.
 - Je nach grep-Programm oder Option können Pattern ein einfacher String, ein "einfacher" regulärer Ausdruck, ein "komplizierter" regulärer Ausdruck sein.
 - Die Komplexiertheit des Pattern bestimmt wesentlich die Arbeitsgeschwindigkeit des "greps".
 - Es können mehrere Pattern spezifiziert werden, dann werden alle Zeilen ausgegeben, bei denen irgendein Pattern matchet.
 - Es kann kein, ein oder mehrere Files pro Kommandoaufruf spezifiziert werden. Wenn kein File spezifiziert wurde liest grep die Daten von der Standardeingabe.
- Probleme bereitet die Beschreibung der Pattern in Shellumgebungen, da \$, *, [, ^, |, (,), und \ gleichzeitig Shell-Metnzeichen und Elemente von regulären Ausdrücken sind.

j-p bell

Seite 2

```
SUN solaris 8
-----
/usr/bin/grep [-bchilnsvw] limited-regular-expression [filename ...]

/usr/xpg4/bin/grep [-E|-F] [-c|-l|-q] [-bhinsvwx] -e pattern_list ...
-f pattern_file ... ]|pattern [ file ... ]

/usr/bin/grep
/usr/bin/grep unterstützt begrenzte reguläre Ausdrücke (regexp(5))

/usr/xpg4/bin/grep
Mit Option -E werden volle reguläre Ausdrücke als Pattern
unterstützt.
Mit Option -F werden nur feste Strings als Pattern unterstützt.
Wird weder -E noch -F spezifiziert, werden einfache reguläre
Ausdrücke Pattern untersützt ( regexp(5))
```

Optionen für /usr/bin/grep und /usr/xpg4/bin/grep:

- b Gibt die Byte-Position jeder gefundenen Stelle mit aus
- Erstes Byte hat die Adresse 0.
- c Gibt nur die Gesamtzahl der gefundenen Stellen aus
- h Unterdrückt die Dateinamen vor jeder Fundstelle, wenn mehrere Files angegeben wurden
- i ignoriert Groß- und Kleinschreibung
- l gibt nur die Dateinamen mit Fundstellen aus, ohne Wiederholung
- n Gibt die Zeilennummer zu jeder Fundstelle aus (erste Zeile ist 1).
- s Keine Fehlermeldungen (silent)
- v gibt nur Zeilen aus, die das Pattern nicht enthalten
- w gibt nur Zeilen aus, die das Pattern als komplettes Wort enth

j-p bell

Seite 3

Optionen für /usr/xpg4/bin/grep

-e pattern_list

Spezifiziert ein oder mehrere Pattern. Die Pattern müssen durch <NL> getrennt sein. Achtung!! Pattern-Listen müssen durch Apostrophs geklammert sein.

-e pattern

Spezifiziert ein einzelnes Pattern.

-E Unterstützung der vollen regulären Ausdrücke als Pattern.

-f pattern_file

pattern_file Datei enthält die Ausdrücke, nach denen gesucht w
Unterstützung von festen Strings als Pattern, keine regulären
Quiet-Modus. Keine Ausgabe. Return-Wert 0 wenn Zeile gefunden
gibt nur Zeilen aus, die das Wort als ganze Zeile enthalten

Operanden

file Dateiname der zu untersuchenden Datei

Bemerkungen

Die -e pattern_list Option hat die selbe Funktionalität wie mehrere
-e pattern Mehrfache -e und -f Optionen sind erlaubt.
-q Option bei mehreren Files: ein Auftreten reicht für Rückkehrwert 0

Filegröße: 2 GByte

Rückkehrwerte:

```
0 Mindestens eine Zeile gefunden
1 Keine Zeile gefunden
2 Syntaxfehler ode File nicht gefunden
```

j-p bell

Seite 4

Beispiele:

```
bell15% /usr/bin/grep -i -n bell passwd
bell15% /usr/bin/grep ^$ passwd
bell15% /usr/bin/grep -v . passwd
bell15% /usr/xpg4/bin/grep -E 'mbell
> cbell' passwd
bell15% /usr/xpg4/bin/grep -F -e 'mbell' -e 'cbell' passwd
bell15% /usr/xpg4/bin/grep -E '^cbell
^mbell' passwd
bell15% /usr/xpg4/bin/grep -F -w 'bell'
bell15% /usr/xpg4/bin/grep -F -x 'bell'
```

j-p bell

Seite 5

9.2.grep

SUN Solaris 2.8

```
/usr/bin/fgrep [-bchilnsvx] [-e pattern] [-f pattern-file]
                [pattern] [file ...]
/usr/xpg4/bin/fgrep [-bchilnsvx] [-e pattern] [-f pattern-file]
                [pattern] [file ...]
```

fgrep - fast grep. Als Pattern sind nur Strings zugelassen. Die Zeichen \$, *, [, ^, |, (,), und \ werden als Literale gewertet.

/usr/xpg4/bin/fgrep entspricht /usr/xpg4/bin/grep -F

Optionen:

- b Gibt die Byte-Position jeder gefundenen Stelle mit aus
- c Erstes Byte hat die Adresse 0.
- e Gibt nur die Gesamtzahl der gefundenen Stellen aus
- e pattern Spezifiziert ein einzelnen String (kann mehrfach auftreten).
- f pattern-file Spezifiziert ein File mit Strings
- h Unterdrückt die Dateinamen vor jeder Fundstelle, wenn mehrere Files angegeben wurden
- i ignoriert Groß- und Kleinschreibung
- l Gibt nur die Dateinamen mit Fundstellen aus, ohne Wiederholung
- n Gibt die Zeilennummer zu jeder Fundstelle aus (erste Zeile ist 1).
- s Keine Fehlermeldungen (silent)
- v gibt nur Zeilen aus, die den String nicht enthalten
- x gibt nur Zeilen aus, die den String als ganze Zeile enthalten

j-p bell

Seite 6

Operanden

file Dateiname der zu untersuchenden Datei

`/usr/bin/fgrep`
 pattern String

`/usr/xpg4/bin/fgrep`
 pattern String

Filegröße: 2 GByte

Rückkehrwerte:

- 0 Mindestens eine Zeile gefunden
- 1 Keine Zeile gefunden
- 2 Syntaxfehler oder File nicht gefunden

SUN Solaris 2.8

`/usr/bin/egrep [-bchilnsv] [-e pattern_list] [-f file] [pattern] [file ...]`

`/usr/xpg4/bin/egrep [-bchilnsvx] [-e pattern_list] [-f file] [pattern] [file`

`egrep - expression grep`

Erlaubt reguläre Ausdrücke als Suchpattern.

`/usr/bin/egrep`

Erlaubt reguläre Ausdrücke mit folgenden

Einschränkungen:

```
\( und \),
\{ und \},
\< und \> und
\n
```

Erweiterungen:

1. `<Regulärer Ausdruck>+ - ein- oder mehrfaches Auftreten des Ausdrucks`
2. `<Regulärer Ausdruck>? - null- oder einfaches Auftreten des Ausdrucks`
3. `<Regulärer Ausdruck> | <Regulärer Ausdruck> | ..
 oder anstelle des | ein <NL> - matchet beim Auftreten eines
 der angegebenen Ausdrücke.`
4. `(<regulärer Ausdruck>)`

`/usr/xpg4/bin/egrep`
`/usr/xpg4/bin/egrep entspricht /usr/xpg4/bin/grep -E`

Optionen

/usr/bin/egrep und /usr/xpg4/bin/egrep :

- b Gibt die Byte-Position jeder gefundenen Stelle mit aus
Erstes Byte hat die Adresse 0.
- c Gibt nur die Gesamtzahl der gefundenen Stellen aus
- e pattern_list
Spezifiziert ein Liste von Ausdrücken
(kann mehrfach auftreten).
- f pattern-file
Spezifiziert ein File mit Ausdrücken
- h Unterdrückt die Dateinamen vor jeder Fundstelle, wenn mehrere
Files angegeben wurden
- i ignoriert Groß- und Kleinschreibung
- l gibt nur die Dateinamen mit Fundstellen aus, ohne
Wiederholungen.
- n Gibt die Zeilennummer zu jeder Fundstelle aus
(erste Zeile ist 1).
- s Keine Fehlermeldungen (silent)
- v gibt nur Zeilen aus, die den Ausdruck nicht enthalten

/usr/xpg4/bin/egrep

- x gibt nur Zeilen aus, die den Ausdruck als ganze Zeile
enthalten

j-p bell

Seite 9

Operanden

file Dateiname der zu untersuchenden Datei
pattern Ausdruck

Filegröße: 2 GByte

Rückkehrwerte:

- 0 Mindestens eine Zeile gefunden
- 1 Keine Zeile gefunden
- 2 Syntaxfehler ode File nicht gefunden

j-p bell

Seite 10

Linux (GNU)

grep, egrep, fgrep

```
grep [-[AB] NUM] [-CEFGVabchiLlnqrsvwxyUu] [-e PATTERN |
-f FILE] [-d ACTION] [--directories=ACTION] [--extended-regex]
[--fixed-strings] [--basic-regex] [--regexp=PATTERN]
[--file=FILE] [--ignore-case] [--word-regex] [--line-regex]
[--line-regex] [--no-messages] [--revert-match]
[--version] [--help] [--byte-offset] [--line-number]
[--with-filename] [--no-filename] [--quiet] [--silent]
[--text] [--files-without-match] [--files-with-matches]
[--count] [--before-context=NUM] [--after-context=NUM]
[--context] [--binary] [--unix-byte-offsets] [--recursive]
[ pattern ] [ files...]
```

Hauptoptionen

```
-G, --basic-regex
Pattern ist ein regulärer Standardausdruck (Standard)
-E, --extended-regex
Pattern ist ein erweiterter regulärer Ausdruck
(fast wie egrep)
-F, --fixed-strings
Pattern ist eine Menge Newline-getrennter Strings
(fgrep)
```

j-p bell

Seite 11

9.2.grep

7.4.2017

zusätzliche Optionen:

```
-A NUM, --after-context=NUM
NUM Zeilen von vorausgehendem Kontext anzeigen
-B NUM, --before-context=NUM
NUM Zeilen von folgendem Kontext anzeigen
-C, --context[=NUM]
NUM (Standard ist 2) Zeilen Kontext anzeigen,
wenn nicht mit -A oder -B anderes festgelegt ist
-NUM
Wie --context=NUM
-V, --version
Versionsnummer ausgeben und beenden
-b, --byte-offset
Byte-Offset zusammen mit Auszeilen anzeigen
-c, --count
Nur Zahl der passenden Zeilen pro Datei ausgeben, normale
Ausgabe unterdrücken
Mit -v, --revert- wird die Zahl der nicht passenden Zeilen
ausgegeben
-d ACTION, --directories=ACTION
Verarbeitung von Verzeichnissen festlegen.
ACTION ist 'read', 'recurse', oder 'skip'.
read - lesen wie ein normales File
skip - übergehen ohne Fehlernachrichten
recurse - durchmustern aller Files des Directories,
rekursiv - identisch mit Option -r
-e PATTERN, --regexp=PATTERN
PATTERN als regulären Ausdruck verwenden
Sinnvoll für Pattern, die mit "--" beginnen
```

j-p bell

Seite 12

```

-f FILE, --file=FILE
  Pattern aus Datei FILE lesen
-h, --no-filename
  Anzeige des Dateiname unterdrücken bei der Eingabe von
  mehreren Files
-i, --ignore-case
  Unterschied zwischen Groß- und Kleinschreibung ignorieren.
-L, --files-without-match
  Nur Namen von Eingabe-Dateien ausgeben, die keine Übereinstimmung e
  Unterdrückung der normalen Ausgabe.
-l, --files-with-matches
  Nur Namen von Eingabe-Dateien mit Übereinstimmungen ausgeben.
  Unterdrückung der normalen Ausgabe.
-n, --line-number
  Zeilennummer mit Ausgabezeilen anzeigen.
-q, --quiet, --silent
  Alle normalen Ausgaben unterdrücken.
-r, --recursive
  Alle Directories recursiv lesen. Wie Option -d recurse.
-s, --no-messages
  Fehlermeldungen unterdrücken.
-a, --text
  Keine Unterdrückung von Ausgabezeilen, die binäre Daten
  enthalten. Normalerweise werden binäre Files nicht
  durchsucht.
-v, --invert-match
  Nicht-passende Zeilen anzeigen.
-w, --word-regexp
  Nur Ausgabe von Zeilen, bei denen das Pattern auf ganze
  Wörter paßt.

```

j-p bell

Seite 13

```

-x, --line-regexp
  Nur Ausgabe von Zeilen, bei denen das Pattern auf ganze
  Zeilen paßt.
-Y
  Synonym für -i.
-U, --binary
  CR-Zeichen am Zeilenende belassen (für MSDOS-Files)
-u, --unix-byte-offsets
  Offsets ausgeben, als fehlten CR-Zeichen(für MSDOS-Files)

```

Argumente:

| pattern | Suchmuster |
|---------|------------|
|---------|------------|

| | |
|------|-------------------------------------|
| file | Files, die durchsucht werden sollen |
|------|-------------------------------------|

Rückkehrwerte:

| | |
|---|--------------------------------------|
| 0 | Mindestens eine Zeile gefunden |
| 1 | Keine Zeile gefunden |
| 2 | Syntaxfehler ode File nicht gefunden |

j-p bell

Seite 14

```
zgrep - search possibly compressed files for a regular expression

zgrep [ grep_options ] [ -e ] pattern filename...
```

Zgrep is used to invoke the grep on compressed or gzip'ed files. All options specified are passed directly to grep. If no file is specified, then the standard input is decompressed if necessary and fed to grep. Otherwise the given files are uncompressed if necessary and fed to grep.

If zgrep is invoked as zegrep or zfgrep then egrep or fgrep is used instead of grep. If the GREP environment variable is set, zgrep uses it as the grep program to be invoked. For example:

```
for sh: GREP=fgrep zgrep string files
for csh: (setenv GREP fgrep; zgrep string files)
```

```
zipgrep - search files in a ZIP archive for lines matching a pattern
```

```
zipgrep [egrep_options] pattern file[.zip] [file(s)...] [-x xfile(s)...]
```

zipgrep will search files within a ZIP archive for lines matching the given string or pattern. zipgrep is a shell script and requires egrep(1) and unzip(1L) to function. Its output is identical to that of egrep(1).

Argumente

All options prior to the ZIP archive filename are passed to egrep.