

Arbeiten mit der Shell Teil 4

Linux-Kurs der Unix-AG

Zinching Dang

6. Dezember 2017

UNIX
AG

TU Kaiserslautern

RH Regionales
Hochschul-
Rechenzentrum
Kaiserslautern RK



Wiederholung & Vertiefung

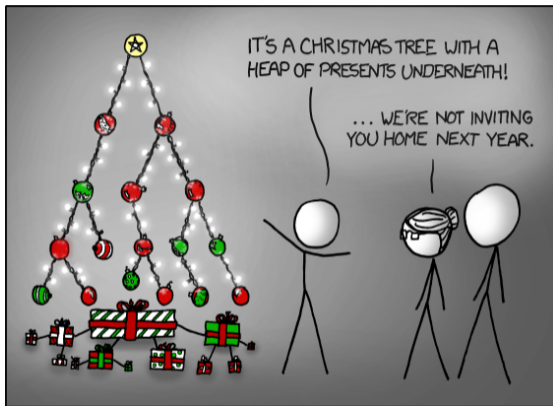
Regular Expressions

Suchen & Finden

Befehle & Optionen

Zusammenfassung & Ausblick

Übersicht



<https://xkcd.com/835>

Wiederholung & Vertiefung: Shellbefehle & Globbs

Platzhater

- ▶ `*` : beliebig viele Zeichen, außer `.` am Anfang
- ▶ `?` : ein beliebiges Zeichen, außer `.` am Anfang
- ▶ `[]` : definierte Liste von Zeichen

Verarbeiten mehrerer Dateien mit Globbs

- ▶ `rm VL*` löscht sämtliche Dateien die mit `VL` beginnen
- ▶ `ls -hl [SW]S1[5-6]` zeigt den Verzeichnisinhalt der Verzeichnisse `SS15` , `WS15` , `SS16` und `WS16` an

Wiederholung & Vertiefung: Shellbefehle & Globbs

Archivieren und Komprimieren

- ▶ `tar cf backup.tar Bilder/ Dokumente/`
 - ▶ erstellt aus `Bilder/` und `Dokumente/` ein Archiv `backup.tar`
- ▶ `tar cjf bu_16.tar.bz2 Bilder/*2016* Dokumente/[WS]S16`
 - ▶ mit Globbs und `bzip2` -Kompression
- ▶ `tar xf backup.tar.gz`
 - ▶ entpackt ein `gzip` -komprimiertes Archiv
- ▶ das „ - “ für Optionen kann weggelassen werden

Regular Expressions

Allgemeines

- ▶ funktionieren ähnlich wie Globs
- ▶ sind nicht komplett mit Globs kompatibel
- ▶ ermöglichen komplexe Suchmuster
- ▶ unterteilt in Platzhalter (Zeichen) und Wiederholungsangabe

Anwendungszwecke

- ▶ suchen (und ersetzen) in Texten

Regular Expressions

Platzhalter

- ▶ `.` : einzelnes beliebiges Zeichen
 - ▶ `a.c` passt z. B. auf: `aac` , `acc` , `aBc` , `a1c` , `a.c` , `a?c`
- ▶ `[]` : anzugebende Zeichenliste
 - ▶ `a[abc]c` passt auf: `aac` , `abc` , `acc` , aber nicht auf: `aAc` , `aaac`
 - ▶ `a[ab][cd]c` passt auf: `aacc` , `aadc` , `abcc` , `abdc` , aber nicht auf: `acdc` , `aabc`
- ▶ `[^]` : alles außer anzugebende Zeichenliste
 - ▶ `a[^abc]c` passt z. B. auf: `axc` , `aYc` , `a1c` , aber nicht auf: `aac` , `abc` , `acc`

Regular Expressions

Wiederholungsangabe

- ▶ `?` : einmal oder keinmal
 - ▶ `a[bd]?c` passt auf: `ac` , `abc` , `adc`
- ▶ `+` : mindestens einmal
 - ▶ `a[bd]+c` passt z. B. auf: `abc` , `adc` , `abbc` , `adbc` , `addc`
- ▶ `*` : beliebig oft
 - ▶ `a[bd]*c` passt z. B. auf: `ac` , `abc` , `adc` , `abbc` , `adbc` , `abddc`
- ▶ `{n}` : genau `n` mal
 - ▶ `a[bd]{2}c` passt auf: `abbc` , `abdc` , `adbc` , `abddc`

Regular Expressions

Weitere Beispiele

- ▶ `Ha*llo`
 - ▶ `Hallo` , `Hllo` , `Haaallo`
- ▶ `[a-z]+[0-9]{3}.`
 - ▶ `a123H` , `pp4321` , `uiae2468` , `abc123!`
- ▶ `[a-f0-9]{8}`
 - ▶ `deadbeef` , `0badf00d` , `abad1dea`

Übersicht: Shellbefehle

Neue Befehle

- ▶ `grep` – gibt Textzeilen aus, die auf ein Suchmuster passen
- ▶ `locate` – findet Dateien anhand von Datenbankeinträgen
- ▶ `whereis` – gibt den Pfad zu Befehlen an
- ▶ `find` – durchsucht rekursiv den Dateibaum

Allgemeines

- ▶ schneidet Zeilen aus, die auf das Suchmuster passen
- ▶ Suchmuster wird als Regular Expression angegeben
- ▶ standardmäßig nur „basic regular expression“
- ▶ Argumente: Suchmuster und Datei(en)

Wichtige Optionen

- ▶ `-E` – „**e**xtended regexes“: erweiterte Regular Expressions, macht dasselbe wie der Befehl `egrep`
- ▶ `-r` – „**r**“ecursive: rekursiv Dateien nach Suchmuster durchsuchen
- ▶ `-v` – „in**v**ert“: Ausgabe umkehren (alles, was nicht auf das Suchmuster passt)

Beispiele

- ▶ `egrep Blubbs meineDatei`
 - ▶ zeigt alle Zeilen an, die das Wort `Blubbs` enthalten
- ▶ `egrep -v Blubbs meineDatei`
 - ▶ zeigt alle Zeilen an, die das Wort `Blubbs` nicht enthalten
- ▶ `egrep [0-9]{5}`
 - ▶ zeigt alle Zeilen an, die eine fünfstellige Zahl enthalten

Dateien anhand einer Datenbank finden – locate

Allgemeines

- ▶ indiziert regelmäßig das Dateisystem
- ▶ gibt den absoluten Pfad von gefundenen Dateien an
- ▶ Argumente: Suchmuster als Glob

Wichtige Optionen

- ▶ `-i` – „**i**gnore case“: Groß- und Kleinschreibung ignorieren

Dateien anhand einer Datenbank finden – locate

Hinweise

- ▶ wird kein Glob-Symbol im Suchmuster verwendet, so werden „`*`“ an den Anfang und an das Ende angehängen
 - ▶ `locate README` wird als `locate *README*` interpretiert
- ▶ neu angelegte Dateien werden nicht gefunden, da sie erst noch indiziert werden müssen
- ▶ Indizierung kann mit dem Befehl `updatedb` erzwungen werden

Allgemeines

- ▶ gibt den absoluten Pfad zu Befehlen und der zugehörigen Man-Page an
- ▶ hat keine nennenswerten Optionen
- ▶ Argument: eine oder mehrere Befehle

Dateibaum durchsuchen – find

Allgemeines

- ▶ sucht rekursiv nach Dateien
- ▶ Startpunkt als erstes Argument
- ▶ Suchkriterien als weitere Argument
- ▶ mehrere Suchkriterien durch Aneinanderreihung
- ▶ mit „ ! “ kann ein Kriterium invertiert werden

Dateibaum durchsuchen – find

Suchkriterien

- ▶ `-iname` : kompletter Dateiname, ohne Groß-/Kleinschreibung
 - ▶ Anführungszeichen (`"`) bei Verwendung von Globs
- ▶ `-size` : Dateigröße (Suffixe wie `k` , `M` oder `G` erlaubt)
 - ▶ `+n` : größer als `n`
 - ▶ `-n` : kleiner als `n`
 - ▶ `n` : genau `n`
- ▶ `-type` : Dateityp
 - ▶ `f` : normale Datei
 - ▶ `d` : Verzeichniss

Dateibaum durchsuchen – find

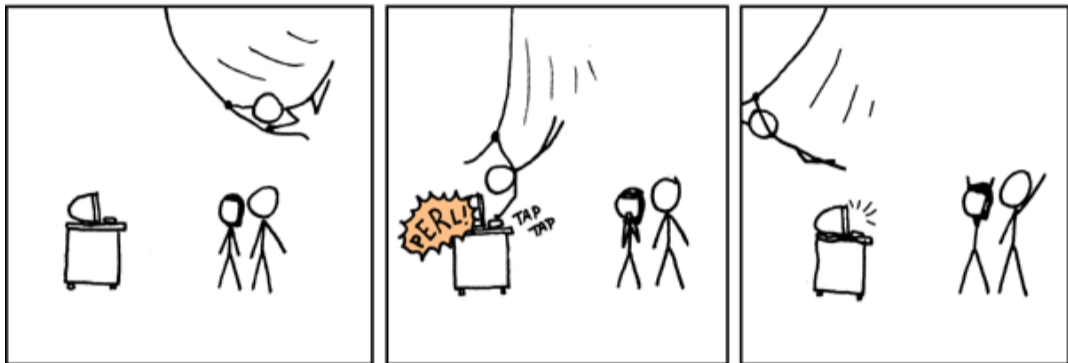
Beispiele

- ▶ `find . -iname "*LiNuXkUrS*"`
 - ▶ sucht im aktuellen Verzeichnis rekursiv nach Dateien und Verzeichnissen die „ linuxkurs “ enthalten und zeigt diese an
- ▶ `find Downloads/ -type f -size +1G -iname "*.iso"`
 - ▶ zeigt alle Dateien an, die größer als 1GB sind und auf `.iso` enden
- ▶ `find . ! -iname "*.pdf"`
 - ▶ zeigt alles außer `.pdf` -Dateien an

Wichtige Befehle & Optionen

Befehl	Optionen	Funktion
<code>grep</code>	<code>-E</code> , <code>-r</code> , <code>-v</code>	Zeilen anzeigen, die auf das Suchmuster passen
<code>locate</code>	<code>-i</code>	Dateien anhand von Datenbank finden
<code>whereis</code>		Dateipfad zu Befehlen anzeigen
<code>find</code>	<code>-iname</code> , <code>-size</code> , <code>-type</code>	Dateien rekursiv suchen: nach Name nach Dateigröße nach Dateityp

Comic zum Thema



<https://xkcd.com/208>

Zusammenfassung & Ausblick

Zusammenfassung

- ▶ komplexe Suchmuster mit Regular Expressions (regex)
- ▶ Dateien suchen und finden

Nächstes Mal

- ▶ weitere Shellbefehle
- ▶ Informationen zum Dateisystem auslesen