Linux

- Bash-Befehle

Bash-Befehle

Beim Arbeiten mit der Bash (aka Shell oder Konsole) treten wiederkehrende Aufgaben auf, wofür wir hier Befehle sammeln.

Systeminfos

Freier Plattenplatz: *df -h*

Prozessorinfos: *cat /proc/cpuinfo*

`(Anzahl cores: |grep 'model name' /proc/cpuinfo | wc -l)`

Server-Laufzeit, eingeloggte user, Auslastung (CPU-Load letzte Minute, 5 Minuten und 15 Minuten):
uptime

Linux-Version (RedHat/CentOS): *cat /etc/redhat-release*

Linux-Version (generisch): *cat /etc/*release*

Prozesse

Prozessübersicht und Systeminfos: *top*

Weitere Details zu Prozessen; die <PID> (z.B. 460831) von top nehmen:
ps -ef | grep <PID> (z.B.: *ps -ef | grep 460831)*

Kurzform:
pmap -d <PID>

Prozessinfos mit Beschriftung:
ps -p <PID> -o pid,vsz=MEMORY -o user,group=GROUP -o comm,args=ARGS

RAM-Verbrauch in MB:
ps -o rss= -p <PID> | awk '{print $1/1024}'

noch mehr Prozess-Details:
cat /proc/<PID>/status

Prozesse, die aktuell mehr als 10% CPU verbrauchen inkl. command-Ursprung (z.B. /cli/cron.php):
*(echo "PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND"; \
top -b -n 1 -c | awk 'NR>7 && $9 > 10' | sort -rn -k 9) | column -t*

Prozesse mit mehr als 10% in letzten 30 Minuten:
*(echo "PID USER %CPU %MEM VSZ RSS ELAPSED PPID COMMAND" && *
*ps -eo pid,user,%cpu,%mem,vsz,rss,etime,ppid,args --sort=-%cpu | *
awk 'NR>1 && $3 > 10 && ($7 ~ /^[0-9]{1,2}:[0-5][0-9]$/ || $7 ~ /^([0-9]+-)?0:[0-5][0-9]:[0-5][0-9]$/)') | column -t

Hilfreiche Parameter

Pipe (|): Der Pipe-Operator (Pipe = Kurzform für Pipeline) leitet die Ausgabe eines Befehls direkt an einen anderen Befehl weiter. Der zweite Befehl kann das Ergebnis bzw. die Ausgabe des ersten Befehls weiterverarbeiten. Es können auch mehr als zwei Pipes kombiniert werden. Beispiele:
Via grep werden die Zeilen aus file.txt angezeigt, welche "bash" enthalten. Dies wird an wc -l weitergegeben zum Zählen (wc=word count), wie oft "bash" vorkommt. Kommt es 10x vor, lautet die Ausgabe "10":
cat file.txt | grep 'bash' | wc -l

Nur Zeilen ausgeben, die ein "G" enthalten, also GB-Größe haben:
sudo ls -lath /vol/moodlebackup | grep 'G'

Zeilen ausgeben mit "G" und "3.", wobei der Punkt mit dem \ als Sonderzeichen maskiert wird (escapen); "grep -v" schließt Ausdrücke aus, "3.mbz" wird damit aus der Ausgabe entfernt:
sudo ls -lath /vol/moodlebackup | grep 'G' | grep '3\.' | grep -v '3\.mbz'

Zeilen ausgeben mit "G" und "1.", "grep -E" filtert nach mehreren Ausdrücken mit | getrennt, hier 2020, 2021 und 2022:
sudo ls -lath /vol/moodlebackup | grep 'G' | grep '1\.' | grep -v '1\.mbz' | grep -E '2020|2021|2022'

Zählen der Zeilen der Ausgabe, also z.B. wie viele Dateien ls in Summe ergibt:
| wc -l

Links / Quellen

Administration über Kommandozeile (Moodle Docs)

Autor: Klaus Steitz, Technische Universität Darmstadt