Praktikum Betriebssysteme

Aufgabe 1 – UNIX (Linux)

# 1.

## a)

strg-C schließt zb die Shell

cat hello.c schreibt zb den Source Code auf die Konsole

cd / geht zum Home Ordner

cd .. geht ein Verzeichnis hoch

chmod +ux hello lässt mich die Datei hello ausführen

cp -i file1 file2 kopiert file1 nach file2 und fragt mich vorher, wenn file2 existiert

date gibt aktuelles Datum ausführe

df Datensysteminformatoinen werden angezeigt

echo gibt String aus

env gibt alle Umgebungsvariablen aus

exit wie strg-c

export gibt

find .../git/haw.bai3.bs -hello.c findet die Datei hello.c in git/haw.bai3.bs

grep -r sucht rekursiv den String in dem File

ls -l gibt eine lange Liste von Dateinamen im aktuellenn Verzeichnis aus

man gibt eine Anleitung zum nachfolgenden Befehl

mkdir erzeugt ein Verzeichnis

more zeigt eine Textdatei Seitenweise her

mv benennt Datei um oder verschiebt

PROG führt Programm aus

ps -efa gibt eine Prozessliste aus

pstree wie ps nur als Baum

VAR=VALUE VAR bekommt den Wert VALUE

>FILE schreibe stdout in FILE

$VAR String VAR durch aktuellen Wert ersetzen

## b)

|  |  |  |
| --- | --- | --- |
| $HOME | bash: /home/thorben: Ist ein Verzeichnis |  |
| $PATH | bash: /home/thorben/bin:/home/thorben/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin: Datei oder Verzeichnis nicht gefunden |  |
| $UID | 1000: Befehl nicht gefunden. |  |
| $USER | thorben: Befehl nicht gefunden. |  |
| cd $HOME | Geht ins oberste Verzeichnis  Alternative: cd home/thorben | cd |
| ↑↓ strg-d | Navigiert zum letzten/nächsten Befehl  Schließt Terminal |  |
| .bashrc |  |  |

# 2.

## a) frename.sh:

#!/bin/bash

# Hängt für alle Dateien im angegebenen Verzeichnis dir

#die Zeichenkette string an den aktuellen Dateinamen an

#(Umbenennung aller Dateien im Verzeichnis, Unterverzeichnisse bleiben unberührt).

# Thorben Schomacker

# Ferdinand Trendelenburg

# 19.02.2018

echo "Umbenennung im Verzeichnis: $1 durch anhaengen von $2 an den Dateinamen"

for entry in "$1"/\*

do

#umbenennen von "alerName" in "alerName" + das was angefügt werden soll

mv "./$entry" "$entry$2"

# Terminalausgabe

echo "$entry wurde umbenannt in $entry$2"

done

## b) try\_host.sh

#!/bin/bash

#

# Thorben Schomacker

# Ferdinand Trendelenburg

# 22.02.2018

echo "Ping wird durchgefuehrt"

light\_red='\e[1;91m%s\e[0m\n'

light\_green='\e[1;92m%s\e[0m\n'

case $1 in

-s)

# pingen im Zeitintervall von $2 sekunden

ping -i $2 $3

# wenn Ergebniss = 0

if [ "$?" -eq 0 ]; then

printf "$light\_green" "$3 OK"

else

printf "$light\_red" "$3 FAILED"

fi

;;

-h)

# einmaliges pingen

ping -c 1 $2

# wenn Ergebniss = 0

if [ "$?" -eq 0 ]; then

printf "$light\_green" "$2 OK"

else

printf "$light\_red" "$2 FAILED"

fi

;;

\*) echo Falscher input ;;

esac

# 3. mkfile

/\*

\* @author: Thorben Schomacker

\* @author: Ferdinand Trendelenburg

\*

\*/

#include <stdio.h>

#include <string.h>

#include <fcntl.h>

int main(void) {

//Variablen initialisieren und einige deklarieren

FILE \*fp;

int size = 30;

char filename[size];

int mode = 0700;

//eigabe des Namens der neuen textdatei

fgets(filename, size, fp);

//zeilenumbruch wegschneiden

strtok(filename, "\n");

//'./' und '.txt' anfügen und datei erstellen mit dem in mode gespeicherten berechtigungen

creat(strcat(strcat("./", filename), ".txt"), mode);

//datei schließen

fclose(fp);

//ausgaben

printf("Name der neuen Datei: %s\n", filename);

printf("Die Datei bsp1 wurde erfolgreich angelegt!");

return(0);

}