

SPRAWOZDANIE, WDSO, LAB4, KRZYSZTOF ZARĘBSKI GL20

Zadanie 4.1

Zamienia „g” z „s” w każdej linii zaczynającej się od „g”

- `/^g/` - znajduje linie zaczynającą się od „g”
- `/s/g/s/` - Pierwsze „s” to opcja edytora Sed, która podmienia to co jest podane w dwóch następnych parametrach, czyli zamienia „g” z „s”
- Ostatnie `/g` - sprawia, że wszystkie „g” w danej linii zostają podmienione, bez tego parametru zostanie zastąpione tylko pierwsze „g” w każdej linii
- `< we` - oznacza wykonaj polecenie na pliku `we`
- `> wy` - rozwiązanie wypisuje w pliku `wy`

Zadanie 4.2

W terminalu piszę komendę: `sed -n '/Aho/, /Peter/!p' pracownicy.dat`

```
ubuntu@ubuntu:~/Dokumenty/prograny$ sed -n '/Aho/, /Peter/!p' pracownicy.dat
Jan Nowak      3.5      30
Tom Blake     2.5      30
Anna Urbaniec  2.5       0
Marek Kowal
```

ubuntu@ubuntu:~/Dokumenty/prograny\$ █

Zadanie 4.3

W terminalu piszę komendę: `sed -n '/\nologin/!p' /etc/passwd | sort -dk5 -t:`

```
ubuntu@ubuntu:~$ sed -n '/nologin!p' /etc/passwd | sort -dk5 -t:
avahi-autoipd:x:105:113:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:117:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
colord:x:113:121:colord colour management daemon,,,:/var/lib/colord:/bin/false
dnsmasq:x:104:65534:dnsmasq,,,:/var/lib/misc:/bin/false
saned:x:108:115:/:home/saned:/bin/false
syslog:x:101:104:/:home/syslog:/bin/false
hplip:x:114:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:106:65534:Kernel Oops Tracking Daemon,,,:/bin/false
lightdm:x:112:118:Light Display Manager:/var/lib/lightdm:/bin/false
ubuntu:x:999:999:Live session user,,,:/home/ubuntu:/bin/bash
whoopsie:x:109:116:/:nonexistent:/bin/false
pulse:x:115:122:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:107:114:RealtimeKit,,,:/proc:/bin/false
root:x:0:0:root:/root:/bin/bash
speech-dispatcher:x:110:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
usbmux:x:103:46:usbmux daemon,,,:/home/usbmux:/bin/false
libuuid:x:100:101:/:var/lib/libuuid:
messagebus:x:102:106:/:var/run/dbus:/bin/false
```

Zadanie 4.4

Podpunkt A:

W terminalu piszę: `sed -i '1!{/^#/d}' firewall`

- `-i` zapisuje zmiany do oryginalnego pliku
- `1!` pomija pierwszą linie
- `{}` grupuje polecenia
- `/^#/d` - usuwa linie zaczynające się od #

```
#!/bin/bash
iptables -t filter -A INPUT -i eth0 -j ACCEPT

iptables -t filter -A INPUT -i eth0 -p udp -j ACCEPT

iptables -t filter -A INPUT -i eth0 -s 192.168.1.1/24 -p tcp --dport 22 -j ACCEPT
iptables -t filter -A INPUT -i eth0 -s 192.168.1.1/24 -p tcp --dport 80 -j ACCEPT
iptables -t filter -A INPUT -i eth0 -s 192.168.1.1/24 -p tcp --dport 443 -j ACCEPT
iptables -t filter -A FORWARD -i eth0 -s 192.168.1.1/24 -j ACCEPT
iptables -t filter -A INPUT -i eth0 -s 192.168.1.1/24 -p tcp --dport 22 -j ACCEPT

iptables -t filter -P FORWARD DROP
iptables -t filter -P INPUT DROP
iptables -t filter -P OUTPUT ACCEPT
```

Podpunkt B:

W terminalu piszę: `sed -i 's/-s 192.168.1.1/24/-d 10.1.2.1/24/' firewall`

- `-i` zapisuje zmiany do oryginalnego pliku
- `'s/'` - zamiana

- Trzeba zamienić znaki /, które mogą zostać zinterpretowane jako składnia polecenia sed, dlatego trzeba przed każdy / wpisać \ aby terminal wiedział, że nie jest to komenda do sed.

Podpunkt C:

W terminalu piszemy: `sed -i 's/-s 10.1.1.1\24//' firewall`

```
#!/bin/bash
iptables -t filter -A INPUT -i eth0 -j ACCEPT
#iptables -t filter -A FORWARD -i eth2 -j ACCEPT
#iptables -t filter -A INPUT -i eth3 -j ACCEPT

#-----TRANSPARENT PROXY-----
#iptables -t nat -A PREROUTING -i eth0 -s 192.168.1.1/24 -p tcp --dport 80 -j R$
#iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port$
#iptables -t nat -A PREROUTING -i eth0 -s 10.1.2.1/24 -p tcp --dport 80 -j REDIS$
#-----KONIEC PROXY-----

#DHCP+SMB UDP
iptables -t filter -A INPUT -i eth0 -p udp -j ACCEPT

#SMB TCP 139
iptables -t filter -A INPUT -i eth0 -d 10.1.2.1/24 -p tcp --dport -j ACCEPT

#PROXY
iptables -t filter -A INPUT -i eth0 -d 10.1.2.1/24 -p tcp --dport -j ACCEPT
```

Zadanie 4.5

Tworzymy skrypt .awk powtarzający po każdej linii bez żadnego warunku, wykorzystując `gsub()`, aby zamienić ciągi znaków na `"jpg"` i pisze w formacie `[stara nazwa] [nowa nazwa.jpg]` -

W terminalu piszę:

`"find . -type f -name *.jpeg -o -name *.JPG -o -name *.JPEG | awk -f script.awk | xargs -n2 mv"`

Polecenie `find` szuka pliku (`-type f`) o nazwie (`-name`), która kończy się rozszerzeniem `.jpeg` albo `.JPG` albo `.JPEG`. Następnie wynik tego polecenia leci do skryptu `.awk` i wynik tego polecenia przesyłam do polecenia `mv` wraz z `xargs` gdzie parametr `-n2` określa maksymalną liczbę argumentów.

```
{
    str = $1
    gsub(/jpeg|JPG|JPEG/, "jpg")
    str = str " " $1
    print str
}
```

```
plik1.jpg
plik2.jpeg
plik3.JPG
plik4.JPEG
plik.jpg
```

Po uruchomieniu skryptu:

```
plik1.jpg
plik2.jpg
plik3.jpg
plik4.jpg
plik.jpg
```

Zadanie 4.6

W terminalu piszę: `ifconfig | awk -f skrypt.awk > interfaces.txt`

Wykorzystując polecenie `ifconfig` przekazuje do skryptu i wynik tego polecenia wypisuje w pliku `interfaces.txt`.

```
ubuntu@ubuntu:~$ ifconfig | awk -f skrypt.awk > interfaces.txt
ubuntu@ubuntu:~$ cat skrypt.awk
#!/bin/awk

BEGIN{
    printf "int_name\tmac_address\t\tip_address\n"
}

/\w+ +Link/ && !/lo +Link/ {
    printf "%s\t\t", $1
}

/inet addr:[0-9.]+/ && !/inet addr:127.0.0.1/ {
    printf "%s\n", substr($2, 6)
}

/HWaddr [0-9a-f:]+/ {
    printf "%s\t", $5
}
}
```

Podgląd pliku `interfaces.txt`

```
ubuntu@ubuntu:~$ cat interfaces.txt
int_name      mac_address      ip_address
eth0          08:00:00:00:00:00 10.0
ubuntu@ubuntu:~$
```

Zadanie 4.7

Utworzyłem plik *zadanie.txt* z liczbami

2

3

5

```
ubuntu@ubuntu:~$ cat zadanie.txt
2
3
5
ubuntu@ubuntu:~$
```

Wnętrze skryptu:

```
#!/bin/awk

/[0-9]+/ {
    for(i=0;i<$1;i++)
        printf "+"
    }
    printf "\n"
}
```

Każdą liczbę z pliku *zadanie.txt* skrypt zastąpi znakiem +

W terminalu piszę `awk -f skrypt.awk zadanie.txt`

```
ubuntu@ubuntu:~$ awk -f skrypt.awk zadanie.txt
++
+++
+++++
```

Zadanie 4.8

Wnętrze skryptu:

```
ubuntu@ubuntu:~$ cat skrypt48.awk
#!/bin/awk

{
    out = ""
    for(i=1; i<=NF; i++){
        if($i != 0){
            if(NF-i != 0){
                out = out$i"x^"NF-i"+"
            }else{
                out =out$i"+"
            }
        }
    }
    out = substr(out,0,length(out)-1)
    printf "%s\n", out
}
ubuntu@ubuntu:~$
```

Podobnie jak w poprzednim zadaniu używam komendy `awk -f skrypt48.awk plik48.txt`

Po uruchomieniu skryptu sprawdzam zawartość `plik48.txt`:

```
ubuntu@ubuntu:~$ cat plik48.txt
4 5 0 2
ubuntu@ubuntu:~$
```