MyLast

Emulator al comenzilor last/lastb

Universitatea din București

Gaitur Ruslan

Grupa 143

CUPRINS

1 Obiectiv

2 Sursa de date: Managementul AUTHLOG

3 Emularea *last*

4 Emularea *lastb*

5 Optiuni de filtrare

6 Analiza performanței

7 Interfața vizuală

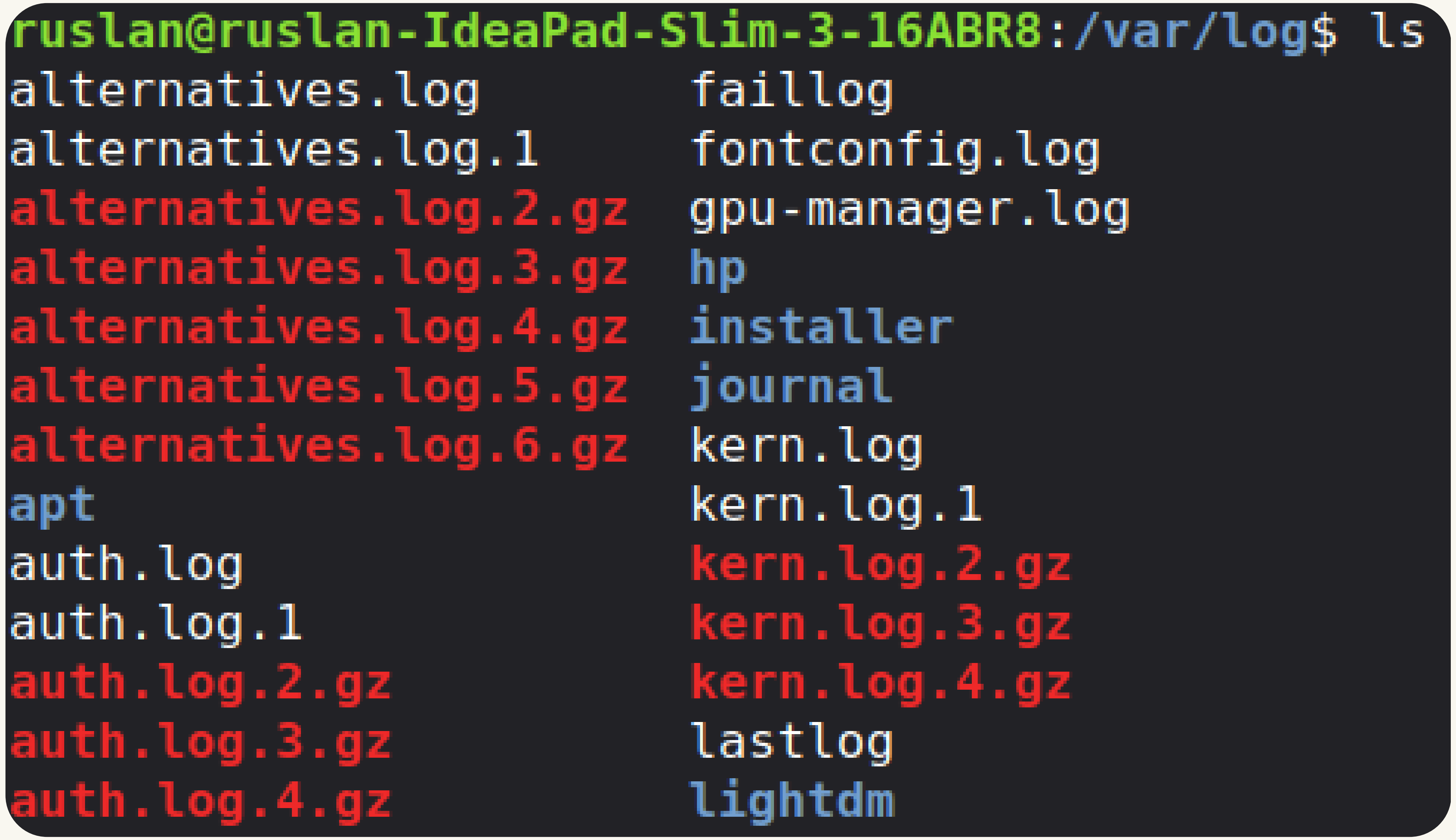
OBIECTIV

- Reproducerea funcționalității comenzilor standard last și lastb folosind un script Bash pur

Sursa de date

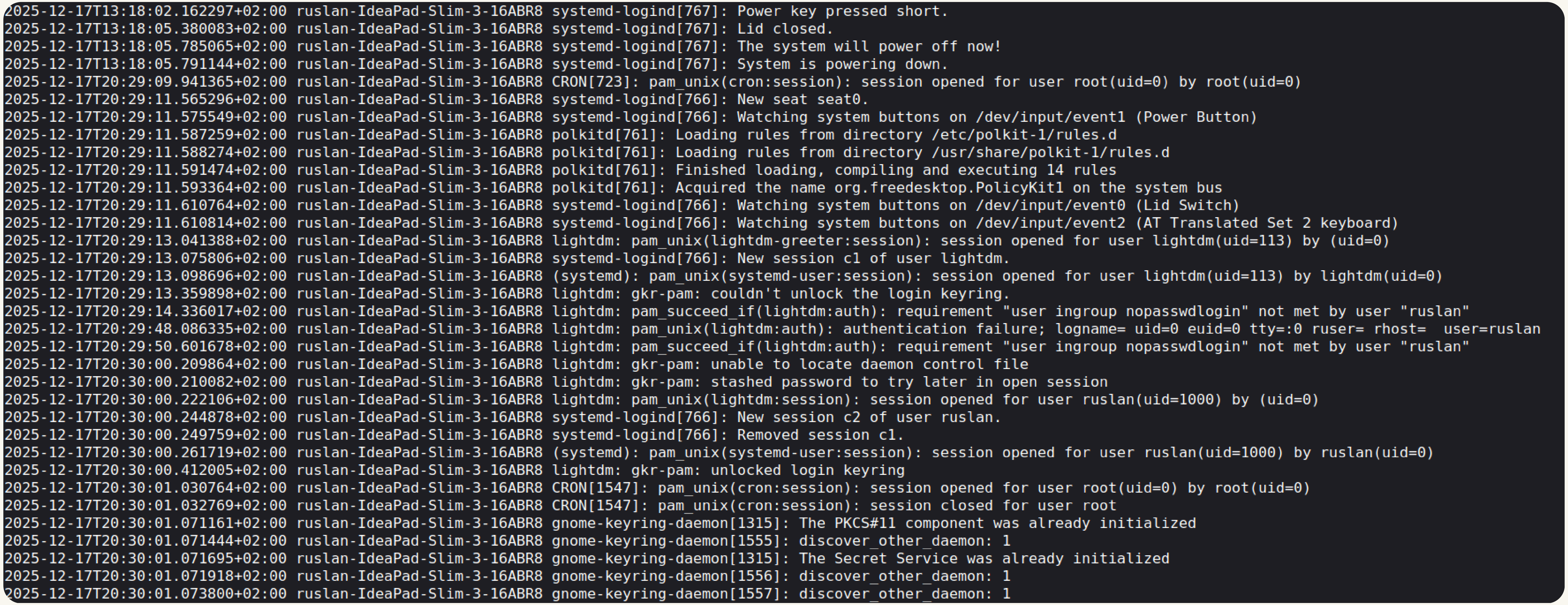
- Sursa de date pentru comenzile custom (*mylast, mylastb*) reprezintă jurnalele de sistem localizate în */var/log*
- Aceste jurnale se impart in 5 fișiere, dintre care 2 sunt arhivate in format *.gz*
- Dezarhivarea are loc folosind comanda *less*
- Toate 5 fișiere sunt concatenate într-un string, rezultatul fiind salvat in variabila *file*
- Variabila *file* reprezintă argumentul principal al comenzilor *mylast* și *mylastb*

Directorul /var/log



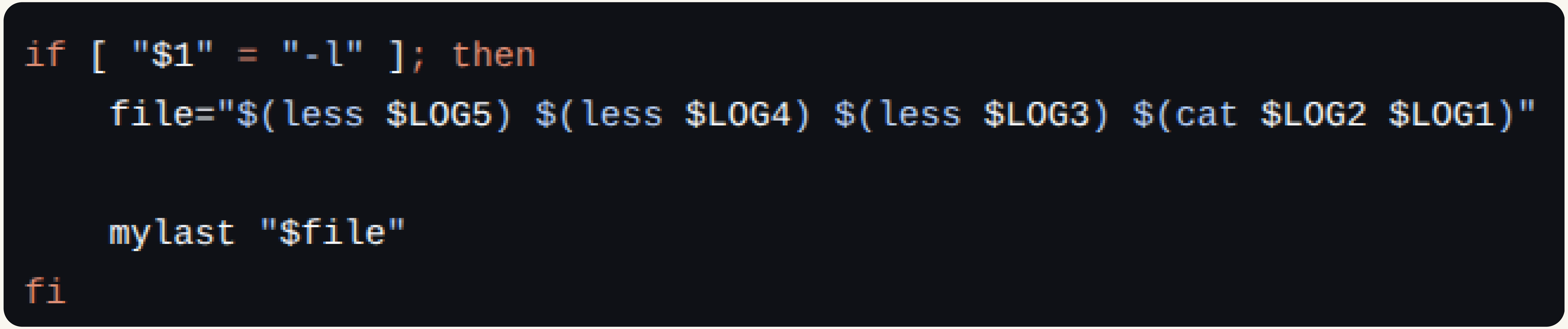
```
ruslan@ruslan-IdeaPad-Slim-3-16ABR8:/var/log$ ls
alternatives.log        faillog
alternatives.log.1      fontconfig.log
alternatives.log.2.gz   gpu-manager.log
alternatives.log.3.gz   hp
alternatives.log.4.gz   installer
alternatives.log.5.gz   journal
alternatives.log.6.gz   kern.log
apt                     kern.log.1
auth.log                kern.log.2.gz
auth.log.1              kern.log.3.gz
auth.log.2.gz           kern.log.4.gz
auth.log.3.gz           lastlog
auth.log.4.gz           lightdm
```

Structura authlog

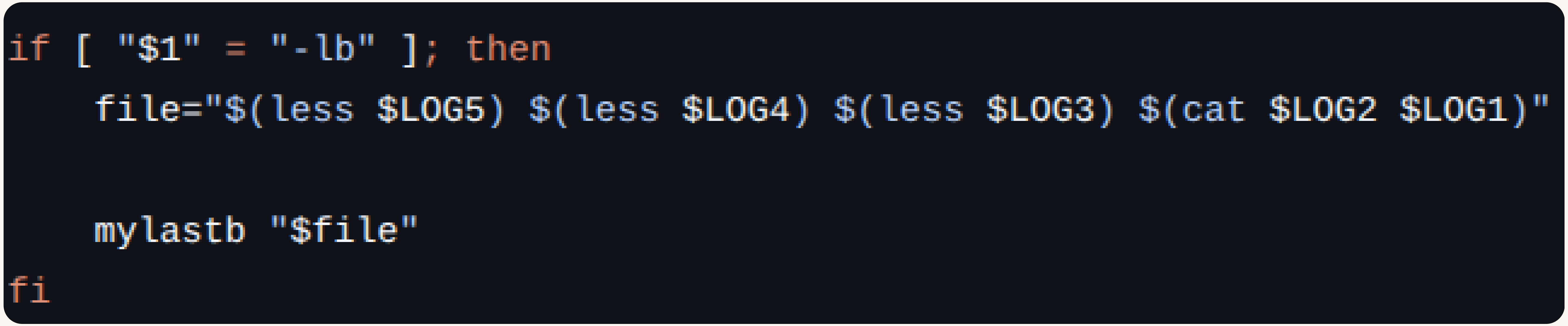


```
2025-12-17T13:18:02.162297+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[767]: Power key pressed short.
2025-12-17T13:18:05.380083+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[767]: Lid closed.
2025-12-17T13:18:05.785065+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[767]: The system will power off now!
2025-12-17T13:18:05.791144+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[767]: System is powering down.
2025-12-17T20:29:09.941365+02:00 ruslan-IdeaPad-Slim-3-16ABR8 CRON[723]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2025-12-17T20:29:11.565296+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: New seat seat0.
2025-12-17T20:29:11.575549+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: Watching system buttons on /dev/input/event1 (Power Button)
2025-12-17T20:29:11.587259+02:00 ruslan-IdeaPad-Slim-3-16ABR8 polkitd[761]: Loading rules from directory /etc/polkit-1/rules.d
2025-12-17T20:29:11.588274+02:00 ruslan-IdeaPad-Slim-3-16ABR8 polkitd[761]: Loading rules from directory /usr/share/polkit-1/rules.d
2025-12-17T20:29:11.591474+02:00 ruslan-IdeaPad-Slim-3-16ABR8 polkitd[761]: Finished loading, compiling and executing 14 rules
2025-12-17T20:29:11.593364+02:00 ruslan-IdeaPad-Slim-3-16ABR8 polkitd[761]: Acquired the name org.freedesktop.PolicyKit1 on the system bus
2025-12-17T20:29:11.610764+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: Watching system buttons on /dev/input/event0 (Lid Switch)
2025-12-17T20:29:11.610814+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: Watching system buttons on /dev/input/event2 (AT Translated Set 2 keyboard)
2025-12-17T20:29:13.041388+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: pam_unix(lightdm-greeter:session): session opened for user lightdm(uid=113) by (uid=0)
2025-12-17T20:29:13.075806+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: New session c1 of user lightdm.
2025-12-17T20:29:13.098696+02:00 ruslan-IdeaPad-Slim-3-16ABR8 (systemd): pam_unix(systemd-user:session): session opened for user lightdm(uid=113) by lightdm(uid=0)
2025-12-17T20:29:13.359898+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: gkr-pam: couldn't unlock the login keyring.
2025-12-17T20:29:14.336017+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: pam_succeed_if(lightdm:auth): requirement "user ingroup nopasswdlogin" not met by user "ruslan"
2025-12-17T20:29:48.086335+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: pam_unix(lightdm:auth): authentication failure; logname= uid=0 euid=0 tty=:0 ruser= rhost=  user=ruslan
2025-12-17T20:29:50.601678+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: pam_succeed_if(lightdm:auth): requirement "user ingroup nopasswdlogin" not met by user "ruslan"
2025-12-17T20:30:00.209864+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: gkr-pam: unable to locate daemon control file
2025-12-17T20:30:00.210082+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: gkr-pam: stashed password to try later in open session
2025-12-17T20:30:00.222106+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: pam_unix(lightdm:session): session opened for user ruslan(uid=1000) by (uid=0)
2025-12-17T20:30:00.244878+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: New session c2 of user ruslan.
2025-12-17T20:30:00.249759+02:00 ruslan-IdeaPad-Slim-3-16ABR8 systemd-logind[766]: Removed session c1.
2025-12-17T20:30:00.261719+02:00 ruslan-IdeaPad-Slim-3-16ABR8 (systemd): pam_unix(systemd-user:session): session opened for user ruslan(uid=1000) by ruslan(uid=0)
2025-12-17T20:30:00.412005+02:00 ruslan-IdeaPad-Slim-3-16ABR8 lightdm: gkr-pam: unlocked login keyring
2025-12-17T20:30:01.030764+02:00 ruslan-IdeaPad-Slim-3-16ABR8 CRON[1547]: pam_unix(cron:session): session opened for user root(uid=0) by root(uid=0)
2025-12-17T20:30:01.032769+02:00 ruslan-IdeaPad-Slim-3-16ABR8 CRON[1547]: pam_unix(cron:session): session closed for user root
2025-12-17T20:30:01.071161+02:00 ruslan-IdeaPad-Slim-3-16ABR8 gnome-keyring-daemon[1315]: The PKCS#11 component was already initialized
2025-12-17T20:30:01.071444+02:00 ruslan-IdeaPad-Slim-3-16ABR8 gnome-keyring-daemon[1555]: discover_other_daemon: 1
2025-12-17T20:30:01.071695+02:00 ruslan-IdeaPad-Slim-3-16ABR8 gnome-keyring-daemon[1315]: The Secret Service was already initialized
2025-12-17T20:30:01.071918+02:00 ruslan-IdeaPad-Slim-3-16ABR8 gnome-keyring-daemon[1556]: discover_other_daemon: 1
2025-12-17T20:30:01.073800+02:00 ruslan-IdeaPad-Slim-3-16ABR8 gnome-keyring-daemon[1557]: discover_other_daemon: 1
```

Argumentul *file*



```
if [ "$1" = "-l" ]; then
    file="$(less $LOG5) $(less $LOG4) $(less $LOG3) $(cat $LOG2 $LOG1)"

    mylast "$file"
fi
```

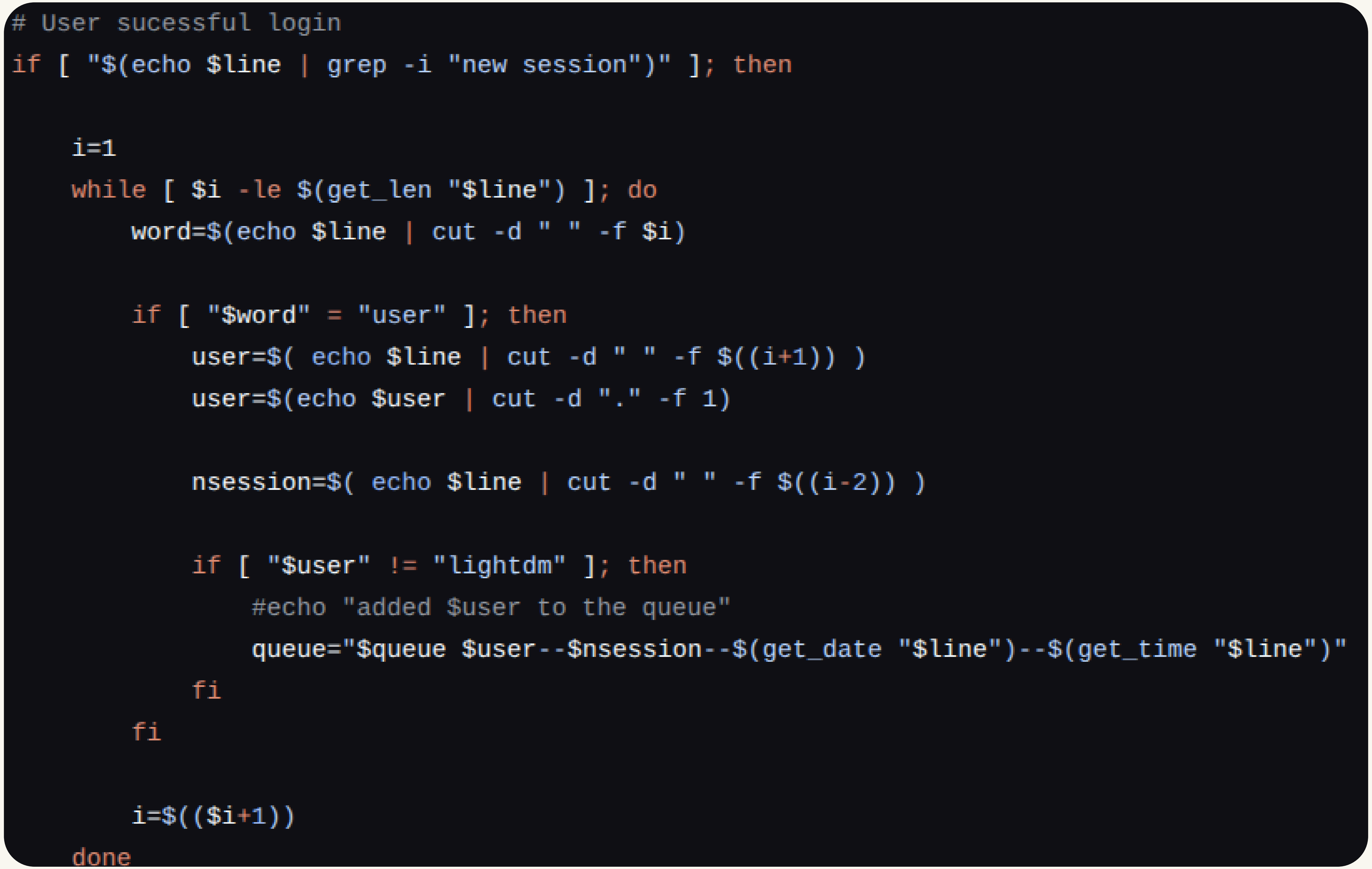


```
if [ "$1" = "-lb" ]; then
    file="$(less $LOG5) $(less $LOG4) $(less $LOG3) $(cat $LOG2 $LOG1)"

    mylastb "$file"
fi
```

Emularea last

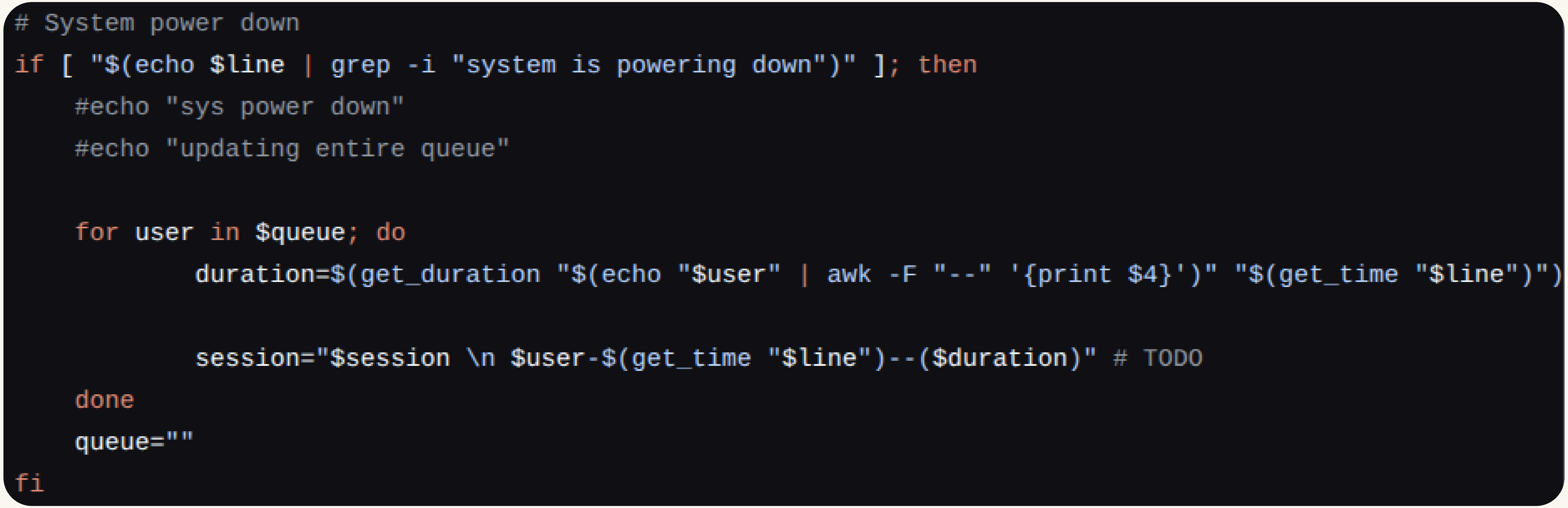
- Sunt definite variabilele *session* și *queue*
- Session - salvează sesiuni terminate, sesiunile se primesc doar din *queue*
- Queue - salvează sesiuni în curs de desfășurare la momentul citirii liniei
- Comanda mylast citește fiecare linie din *authlog* și verifică dacă: 1. A avut loc o autentificare de success a utilizatorului, 2. A avut loc “system power down”, 3. A avut loc o terminare forțată a sesiunii unui utilizator, 4. Utilizatorul s-a delogat din sistem
- Fiecare eveniment din acestea are sintaxa sa specifică in fișierul *authlog*
- Evenimentele de tip 1 adaugă sesiuni în *queue*
- Evenimentele de tip 2, 3, 4 scot sesiuni din *queue*, adăugându-le în *session*

User login



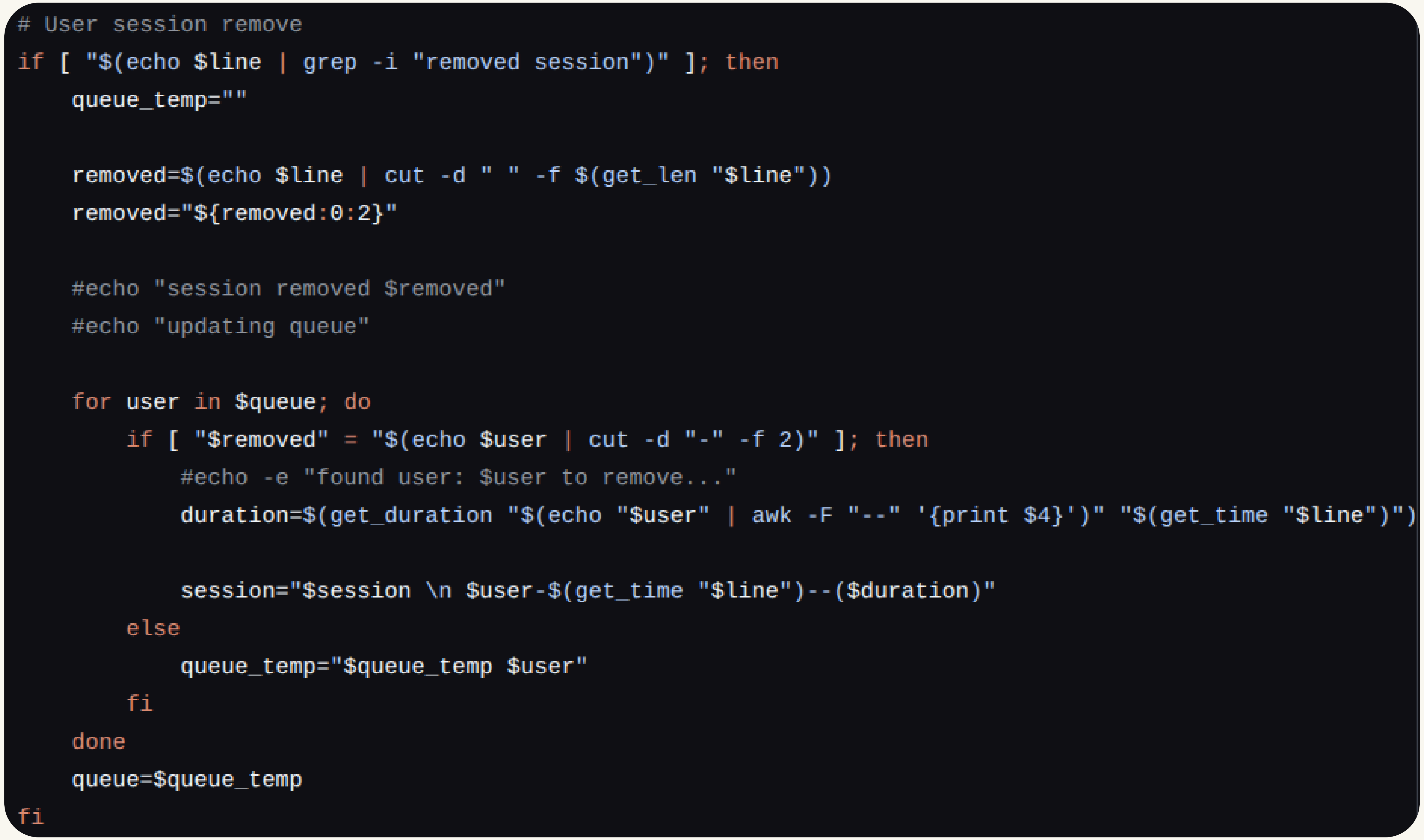
```
# User sucessful login
if [ "$(echo $line | grep -i "new session")" ]; then

    i=1
    while [ $i -le $(get_len "$line") ]; do
        word=$(echo $line | cut -d " " -f $i)

        if [ "$word" = "user" ]; then
            user=$( echo $line | cut -d " " -f $((i+1)) )
            user=$(echo $user | cut -d "." -f 1)

            nsession=$( echo $line | cut -d " " -f $((i-2)) )

            if [ "$user" != "lightdm" ]; then
                #echo "added $user to the queue"
                queue="$queue $user--$nsession--$(get_date "$line")--$(get_time "$line")"
            fi
        fi

        i=$(($i+1))
    done
```

Power down



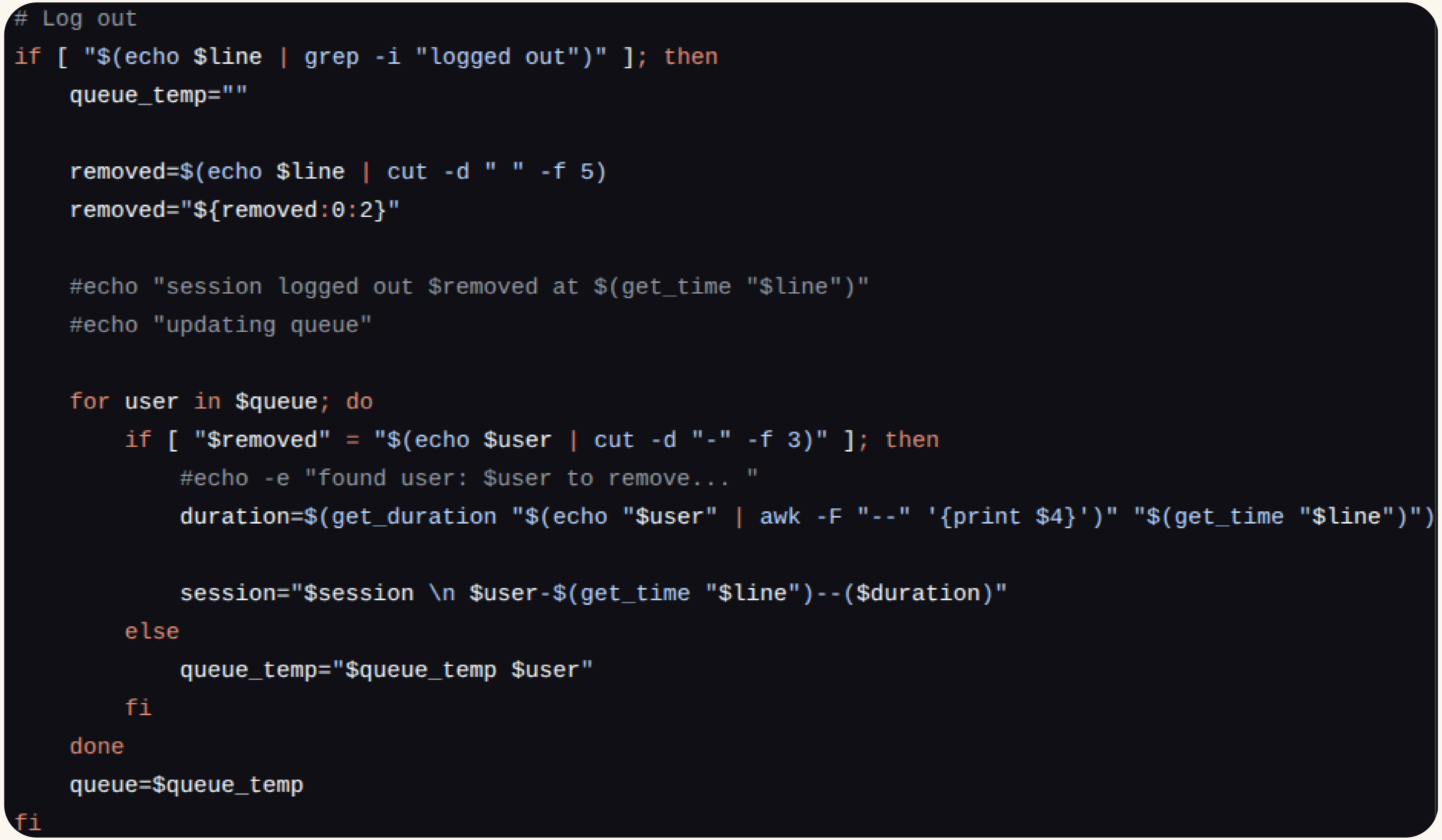
```
# System power down
if [ "$(echo $line | grep -i "system is powering down")" ]; then
    #echo "sys power down"
    #echo "updating entire queue"

    for user in $queue; do
            duration=$(get_duration "$(echo "$user" | awk -F "--" '{print $4}')" "$(get_time "$line")")

            session="$session \n $user-$(get_time "$line")--($duration)" # TODO
    done
    queue=""
fi
```

User session remove



```
# User session remove
if [ "$(echo $line | grep -i "removed session")" ]; then
    queue_temp=""

    removed=$(echo $line | cut -d " " -f $(get_len "$line"))
    removed="${removed:0:2}"

    #echo "session removed $removed"
    #echo "updating queue"

    for user in $queue; do
        if [ "$removed" = "$(echo $user | cut -d "-" -f 2)" ]; then
            #echo -e "found user: $user to remove..."
            duration=$(get_duration "$(echo "$user" | awk -F "--" '{print $4}')" "$(get_time "$line")")

            session="$session \n $user-$(get_time "$line")--($duration)"
        else
            queue_temp="$queue_temp $user"
        fi
    done
    queue=$queue_temp
fi
```

User logout

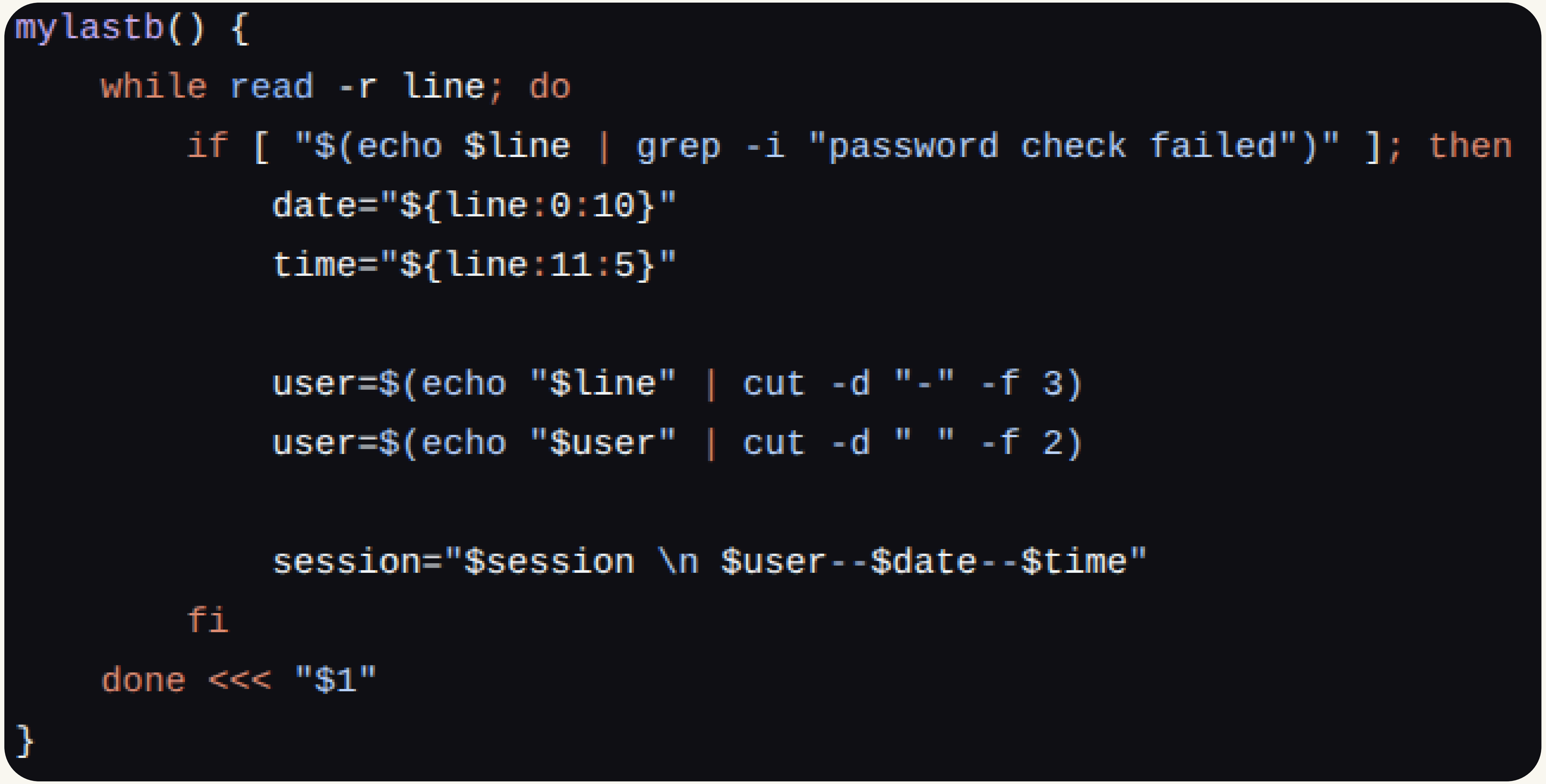


```
# Log out
if [ "$(echo $line | grep -i "logged out")" ]; then
    queue_temp=""

    removed=$(echo $line | cut -d " " -f 5)
    removed="${removed:0:2}"

    #echo "session logged out $removed at $(get_time "$line")"
    #echo "updating queue"

    for user in $queue; do
        if [ "$removed" = "$(echo $user | cut -d "-" -f 3)" ]; then
            #echo -e "found user: $user to remove... "
            duration=$(get_duration "$(echo "$user" | awk -F "--" '{print $4}')" "$(get_time "$line")")

            session="$session \n $user-$(get_time "$line")--($duration)"
        else
            queue_temp="$queue_temp $user"
        fi
    done
    queue=$queue_temp
fi
```

Emularea lastb

- Se citește linie cu linie din *file*
- Sunt de interes liniile care denotă eșecul autentificării, după enunțul “password check failed”
- Se adaugă eșecul direct în variabila *session*

Structura comenzii

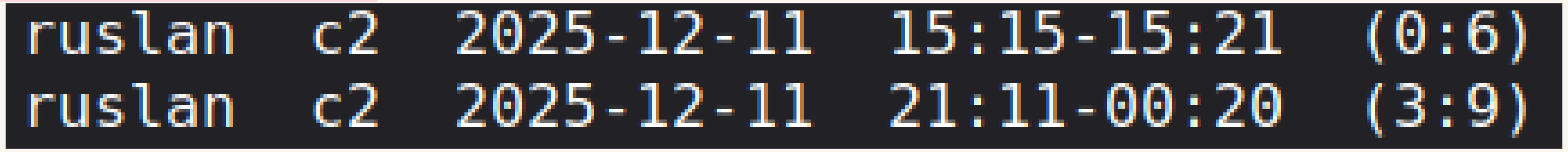


```
mylastb() {
    while read -r line; do
        if [ "$(echo $line | grep -i "password check failed")" ]; then
            date="${line:0:10}"
            time="${line:11:5}"

            user=$(echo "$line" | cut -d "-" -f 3)
            user=$(echo "$user" | cut -d " " -f 2)

            session="$session \n $user--$date--$time"
        fi
    done <<< "$1"
}
```

Opțiuni de filtrare

-n

Afișează doar n linii din cele mai recente sesiuni



```
ruslan   c2   2025-12-11   15:15-15:21   (0:6)
ruslan   c2   2025-12-11   21:11-00:20   (3:9)
```

-s

Afișează toate sesiunile după o anumită dată

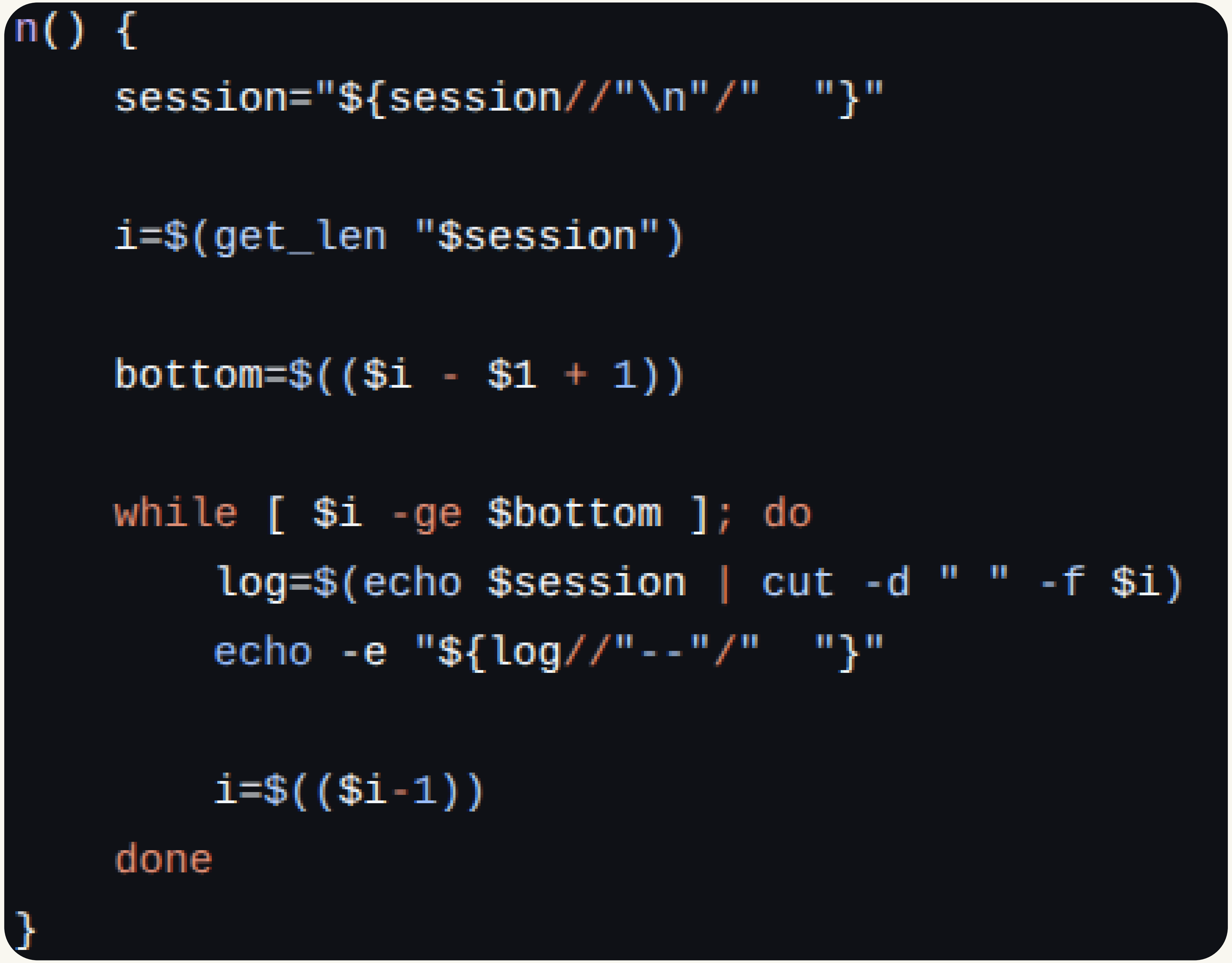
-t

Afișează toate sesiunile înainte de o anumită dată

-p

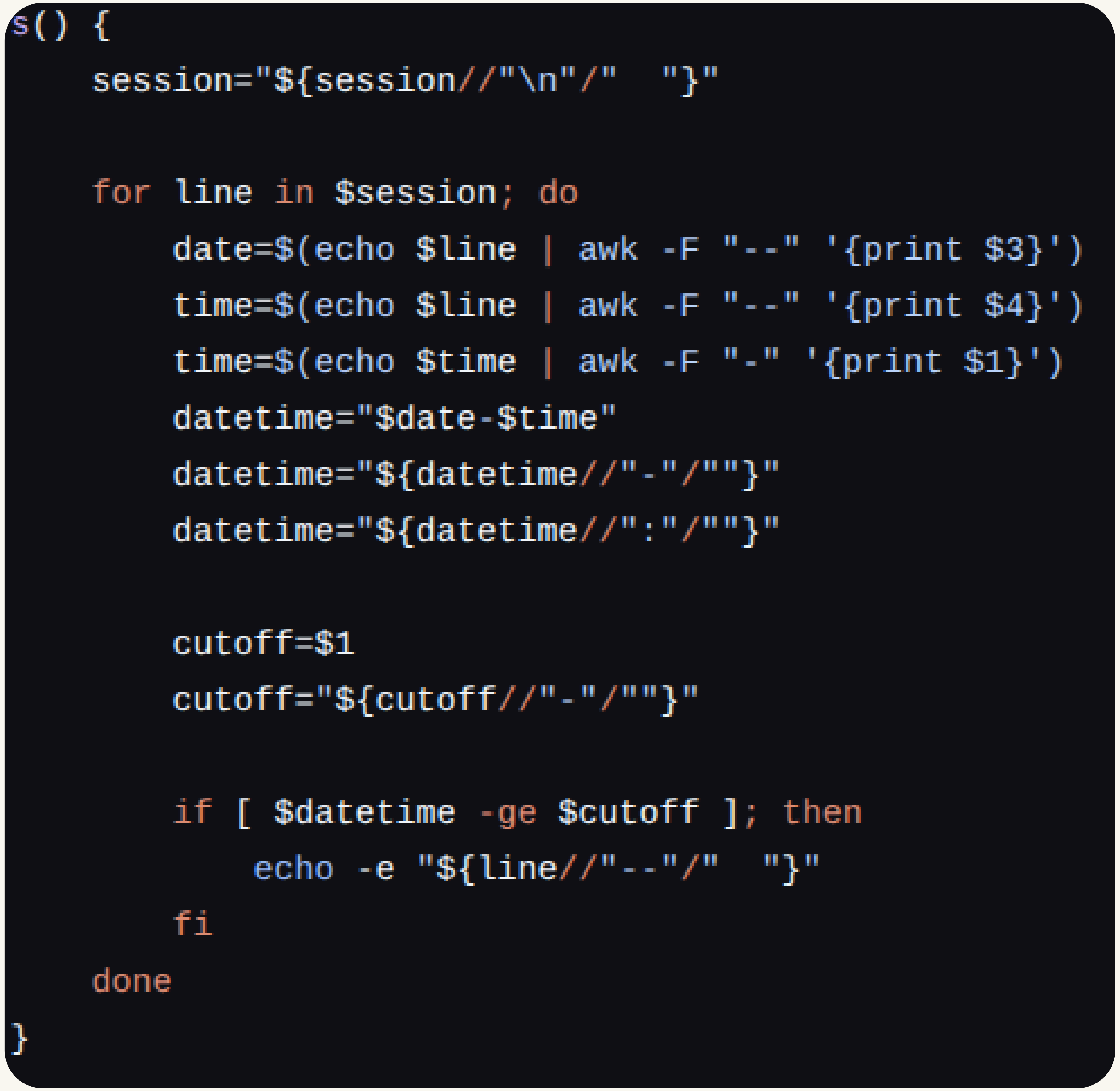
Afișează toate sesiunile active la momentul respectiv

Filtrul *-n*



```
n() {
    session="${session//"\n"/"  "}"

    i=$(get_len "$session")

    bottom=$(($i - $1 + 1))

    while [ $i -ge $bottom ]; do
        log=$(echo $session | cut -d " " -f $i)
        echo -e "${log//"--"/"  "}"

        i=$(($i-1))
    done
}
```

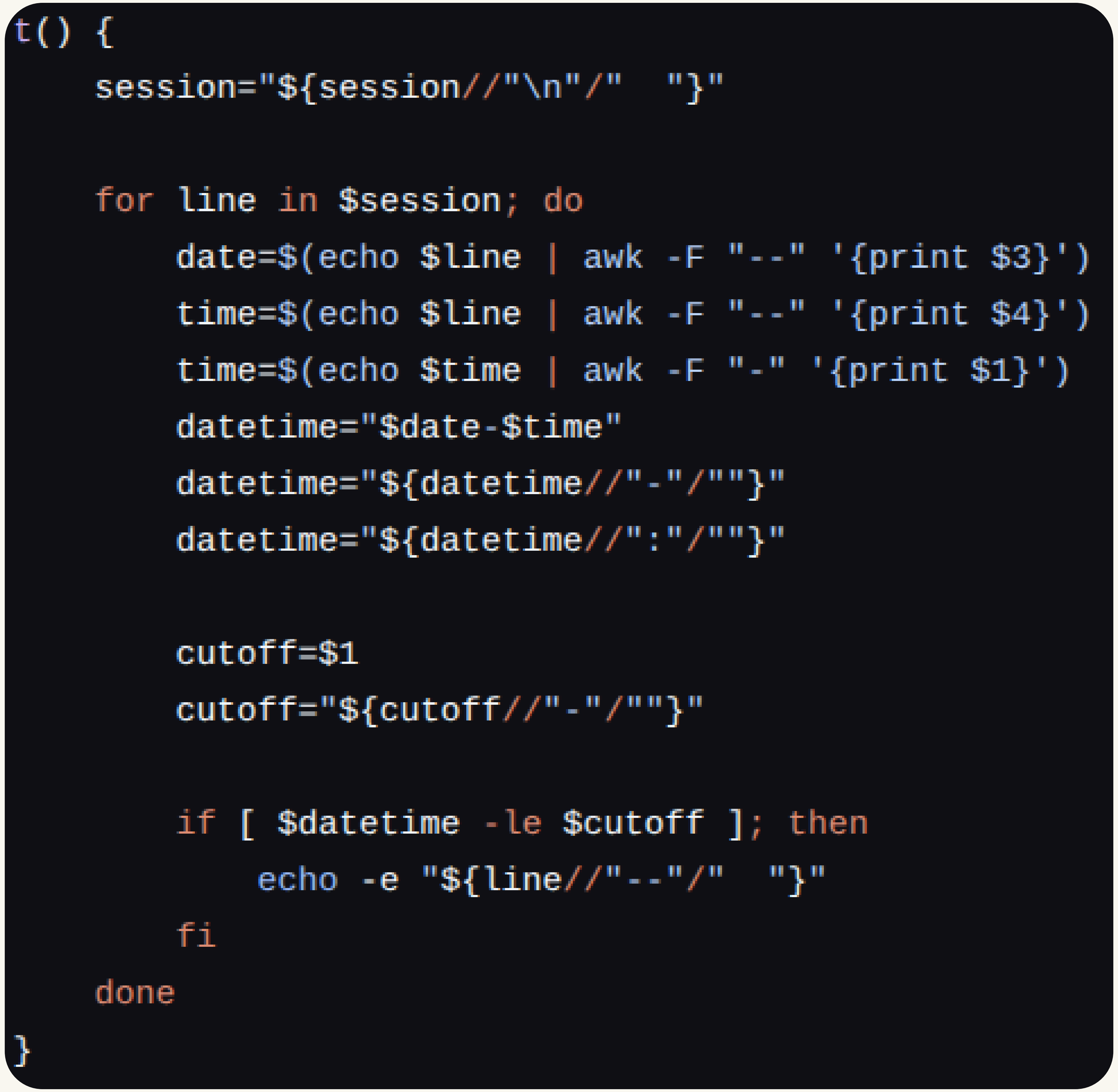
Filtrul -s



```
s() {
    session="${session//"\n"/"  "}"

    for line in $session; do
        date=$(echo $line | awk -F "--" '{print $3}')
        time=$(echo $line | awk -F "--" '{print $4}')
        time=$(echo $time | awk -F "-" '{print $1}')
        datetime="$date-$time"
        datetime="${datetime//"-"/""}"
        datetime="${datetime//":"/""}"

        cutoff=$1
        cutoff="${cutoff//"-"/""}"

        if [ $datetime -ge $cutoff ]; then
            echo -e "${line//"--"/"  "}"
        fi
    done
}
```

Filtrul

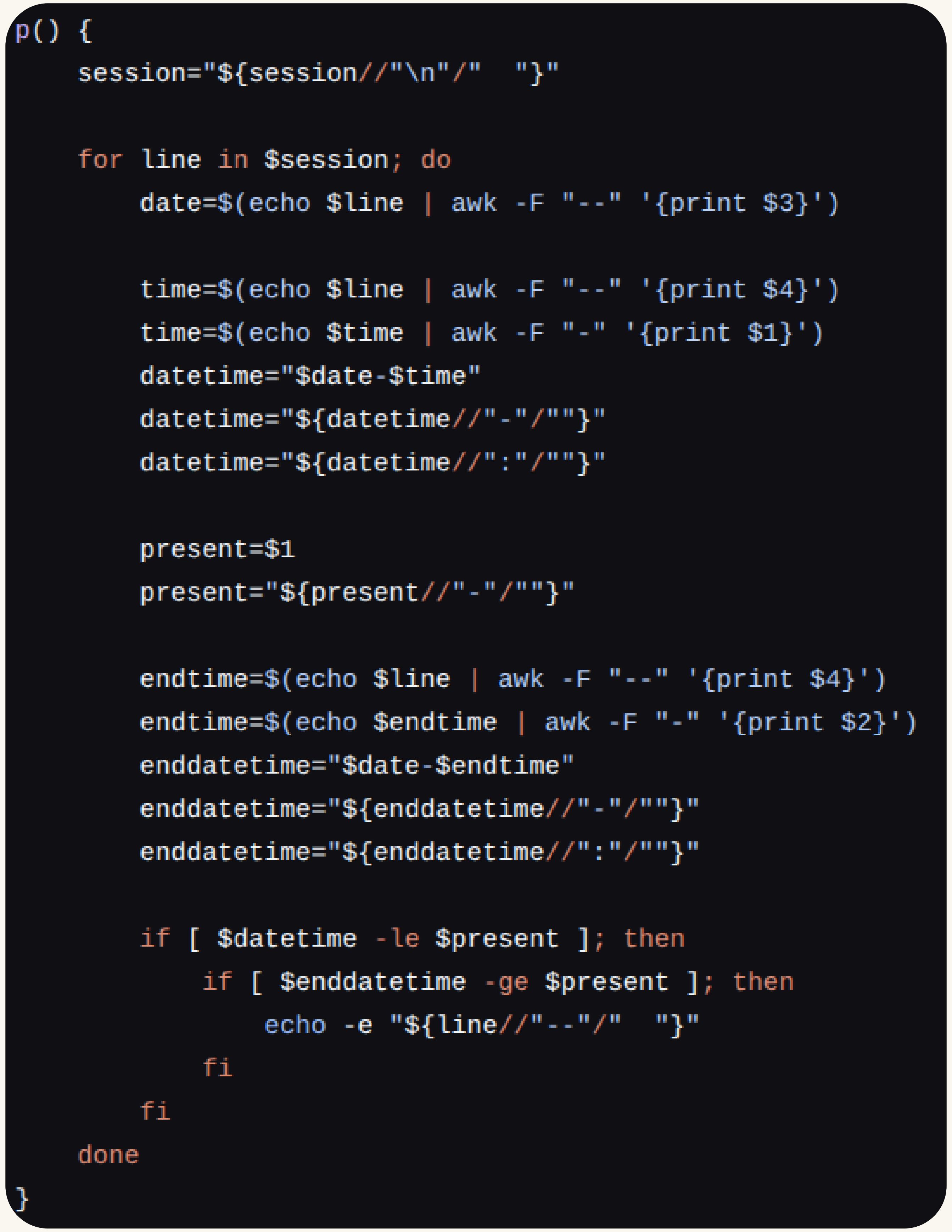
-t



```
t() {
    session="${session//"\n"/"  "}"

    for line in $session; do
        date=$(echo $line | awk -F "--" '{print $3}')
        time=$(echo $line | awk -F "--" '{print $4}')
        time=$(echo $time | awk -F "-" '{print $1}')
        datetime="$date-$time"
        datetime="${datetime//"-"/""}"
        datetime="${datetime//":"/""}"

        cutoff=$1
        cutoff="${cutoff//"-"/""}"

        if [ $datetime -le $cutoff ]; then
            echo -e "${line//"--"/"  "}"
        fi
    done
}
```

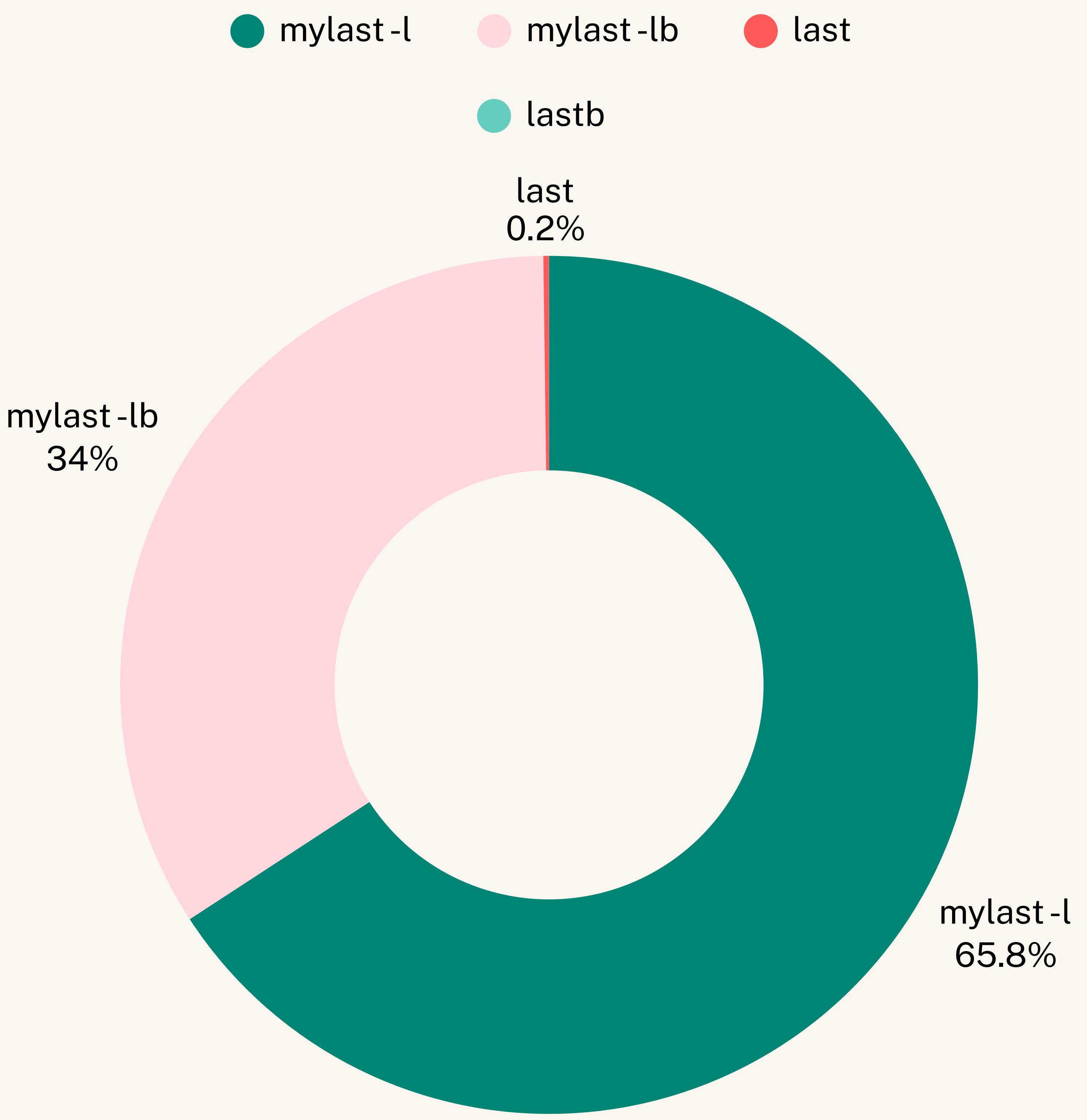
Filtrul

-p



```
p() {
    session="${session//"\n"/"  "}"

    for line in $session; do
        date=$(echo $line | awk -F "--" '{print $3}')

        time=$(echo $line | awk -F "--" '{print $4}')
        time=$(echo $time | awk -F "-" '{print $1}')
        datetime="$date-$time"
        datetime="${datetime//"-"/""}"
        datetime="${datetime//":"/""}"

        present=$1
        present="${present//"-"/""}"

        endtime=$(echo $line | awk -F "--" '{print $4}')
        endtime=$(echo $endtime | awk -F "-" '{print $2}')
        enddatetime="$date-$endtime"
        enddatetime="${enddatetime//"-"/""}"
        enddatetime="${enddatetime//":"/""}"

        if [ $datetime -le $present ]; then
            if [ $enddatetime -ge $present ]; then
                echo -e "${line//"--"/"  "}"
            fi
        fi
    done
}
```

Analiza performanței



Final