

## Урок 4. Bash, скрипты и автоматизация

1. Написать скрипт, который удаляет из текстового файла пустые строки и заменяет маленькие символы на большие (воспользуйтесь tr или sed).
2. Изменить скрипт мониторинга лога, используя утилиту tailf, чтобы он выводил сообщения при попытке неудачной аутентификации пользователя /var/log/auth.log, отслеживая сообщения примерно такого вида: May 16 19:45:52 vlamp login[102782]: FAILED LOGIN (1) on '/dev/tty3' FOR 'user', Authentication failure Проверить скрипт, выполнив ошибочную регистрацию с виртуального терминала.
3. Создать скрипт, который создаст директории для нескольких годов (2010 — 2017), в них — поддиректории для месяцев (от 01 до 12), и в каждый из них запишет несколько файлов с произвольными записями (например, 001.txt, содержащий текст «Файл 001», 002.txt с текстом Файл 002) и т. д.

### 1. Написать скрипт, который удаляет из текстового файла пустые строки и заменяет маленькие символы на большие (воспользуйтесь tr или sed).

- nano file1 „hello world ... !” Strg + x
- cp file1tr
- nano script1.sh
  - #!/bin/bash
  - sed '/#\\|^\$\\| \*#/d' ./file1 | sed -e 's/(.\*)/\\U\\1/' > file1-2
  - cat file1tr | tr -s '\\n' | tr 'a-z' 'A-Z' > file1-2tr
    - OR... | tr [:lower:] [:upper] > file1-2tr
- (sudo) chmod +x script1.sh
- (sudo) sh script1.sh (or ./script1.sh)

```

gb@PC1:~/4les$ nano file1
gb@PC1:~/4les$ cp file1 file1tr
gb@PC1:~/4les$ nano script1.sh
gb@PC1:~/4les$ sh script1.sh
gb@PC1:~/4les$ nano script1.sh
gb@PC1:~/4les$ sh script1.sh
gb@PC1:~/4les$ ls -la
insgesamt 28
drwxr-xr-x 2 gb gb 4096  2. Nov 23:28 .
drwxr-xr-x 4 gb gb 4096  2. Nov 22:59 ..
-rw-r--r-- 1 gb gb 68  2. Nov 23:16 file1
-rw-r--r-- 1 gb gb 66  2. Nov 23:30 file1-2
-rw-r--r-- 1 gb gb 66  2. Nov 23:30 file1-2tr
-rw-r--r-- 1 gb gb 68  2. Nov 23:17 file1tr
-rwxr-xr-x 1 gb gb 130  2. Nov 23:28 script1.sh
gb@PC1:~/4les$ cat file1
hello world
Hello World!
hello world
world
!
3. Создать скрипт, который создаст директории для нескольких годов (2010 — 2017), в них
gb@PC1:~/4les$ cat file1-2 file1-2tr
HELLO WORLD
HELLO WORLD!
HELLO . . . . . WORLD
WORLD
!
HELLO WORLD
HELLO WORLD!
HELLO . . . . . WORLD
WORLD
!

```

2. Изменить скрипт мониторинга лога, используя утилиту tailf, чтобы он выводил сообщения при попытке неудачной аутентификации пользователя /var/log/auth.log, отслеживая сообщения примерно такого вида: May 16 19:45:52 vlamp login[102782]: FAILED LOGIN (1) on '/dev/tty3' FOR 'user', Authentication failure  
Проверить скрипт, выполнив ошибочную регистрацию с виртуального терминала.

- `sudo tail -f /var/log/auth.log`
- `sudo sudo tail -f /var/log/auth.log | grep "authentication failure"`
- `nano script2.sh`

```
#!/bin/bash
# ** Output text like:
# May 16 19:45:52 vlamp login[102782]: FAILED LOGIN (1) on '/dev/tty3' FOR 'user',
Authentication failure
# cut -d" " -f1-4 ... # echo "$(date '+%b %d %T')"
# sudo tail -f /var/log/auth.log | grep "authentication failure"
#...
#grep FAIL | echo "$(date '+%b %d %T') $(logname) login[$(id -u)]: FAILED LOGIN on
'\dev\$(who | cut -d" " -f6))' FOR $(whoami), Authentication failure"
#...
# write to file ...
# grep "$(date '+%b %e')" $LOG_FILE | grep "Authentication failure for user" >
loginfailure_$(date +%m-%d-%y)
```

```
LOG_FILE='/var/log/auth.log'
tail -f $LOG_FILE | grep "authentication failure" && > loginfailure_$(date +%m-%d-%y)
```

- `(sudo) chmod +x script2.sh`
- `(sudo) sh script2.sh`

The screenshot shows three terminal windows. The top window displays the output of `tail -f /var/log/auth.log | grep FAIL`, showing several failed login attempts for user1. The bottom-left window shows the execution of `sh script2.sh`, which outputs a failed login message for user1. The bottom-right window shows a user attempting to switch to root using `su - root`, which results in an authentication failure.

```
└─$ tail -f /var/log/auth.log | grep FAIL
Nov  4 09:43:24 [localhost] su[84511]: FAILED SU (to user1) kali on pts/3
Nov  4 09:44:25 [localhost] su[84767]: FAILED SU (to user1) kali on pts/3
Nov  4 09:44:53 [localhost] su[84884]: FAILED SU (to user1) kali on pts/3
Nov  4 09:46:39 [localhost] su[85388]: FAILED SU (to root) user1 on pts/3

┌─(kali@kali)-[~/les4]
└─$ sh script2.sh
Nov 04 09:46:37 kali login[1000]: FAILED LOGIN on '\dev\tty7' FOR kali, Authentication failure

┌─(user1@kali)-[~]
└─$ su - root
Password:
su: Authentication failure

┌─(user1@kali)-[~]
└─$
```

- ```
nano script3.sh
#!/bin/bash
for file in ./les4/{2010..2017}/{01..12}/{001..005}; do
    mkdir -p "$(dirname $file)" && echo "Файл $(echo $file | cut -d '/' -f 5)" > "$file.txt"
done
```
- ```
(sudo) chmod +x script3.sh
```
- ```
(sudo) sh script3.sh
```

| Links | Datent                    | Befehl | Optionen     | Rechts        |
|-------|---------------------------|--------|--------------|---------------|
| ...   | 3 Unix Ubuntu/4/4les/les4 | -[.]>  | ...          | B PC/3 Unix U |
| .n    | Name                      | Größe  | Modifikation | .n            |
| /..   | UBERVZ                    | Nov 3  | 20:38        | /..           |
| /2010 | 4096                      | Nov 3  | 20:38        | /les4         |
| /2011 | 4096                      | Nov 3  | 20:38        | file1         |
| /2012 | 4096                      | Nov 3  | 20:38        | file1-2       |
| /2013 | 4096                      | Nov 3  | 20:38        | file1-2tr     |
| /2014 | 4096                      | Nov 3  | 20:38        | file1tr       |
| /2015 | 4096                      | Nov 3  | 20:38        | *script1.sh   |
| /2016 | 4096                      | Nov 3  | 20:38        | *script3.sh   |
| /2017 | 4096                      | Nov 3  | 20:38        |               |