

Міністерство освіти і науки України
НТУУ «КПІ ім. Ігоря Сікорського»
Навчально-науковий інститут атомної та теплової енергетики
Кафедра цифрових технологій в енергетиці

Лабораторна робота №3
з дисципліни «Операційна система UNIX»
Тема «Робота в оболонці bash, середовище оточення»
Варіант №22

Студента 2-го курсу НН ІАТЕ гр. ТР-12

Ковальова Олександра

Перевірила: д.т.н., проф. Левченко Л. О.

Мета роботи. Набути навичок налаштування програмного середовища Linux. Здобути практичні навички в роботі з командними оболонками.

Теоретична частина. Командна оболонка в Unix – це інтерфейс командного рядка, тобто виконує команди, які подає користувач, або які читаються з файлів. Такі файли з командами оболонки називаються сценаріями (скриптами, програмами) оболонки. Ці сценарії не компілюються, а інтерпретуються оболонкою. Це означає, що оболонка прочитує сценарій від початку до кінця, рядок за рядком, шукаючи зазначені там команди й виконуючи їх. На відміну від цього підходу, компілятор перетворює цілу програму до вигляду, придатного до виконання машиною — потім файл з таким кодом можна використати в сценарії оболонки. Характерна особливість мови оболонки — багато операцій, які в традиційних мовах програмування є вбудованими, виконуються з допомогою виклику зовнішніх програм.

Одним з завдань оболонки є забезпечення користувацького середовища, яке можна налаштувати за допомогою конфігураційних файлів.

UNIX надає на вибір декілька різних оболонок:

- **sh, або оболонка Борна (Bourne Shell)**

одна з перших оболонок, яка була використана у UNIX-подібних середовищах. Це базова оболонка, маленька та з невеликим набором можливостей. Вона є де-факто стандартною оболонкою, та присутня на кожній системі із UNIX.

- **bash, або нова оболонка Борна (Bourne Again Shell)**

стандартна оболонка Лінуks. У більшості користувачів Лінуks стандартною оболонкою встановлено саме bash. У деякому сенсі bash — це надбудова над sh, набір доповнень та додаткових модулів. Таким чином, нова оболонка Борна сумісна зі звичайною оболонкою Борна: команди, що працюють у sh, будуть працювати і у bash, але не обов'язково навпаки.

- **csh, або C-орієнтована оболонка (C Shell)**

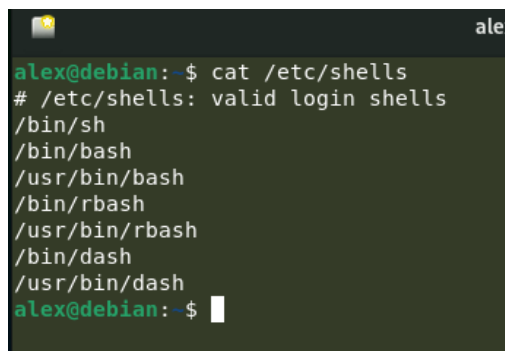
синтаксис цієї оболонки схожий із мовою програмування C. Зазвичай, цією оболонкою користуються програмісти.

- **ksh, або оболонка Корна (Korn Shell)**

була написана у початку 1980-х років та отримала найкращі на той час можливості оболонки Борна та csh. Є надбудовою над оболонкою Борна та гарантує виконання скриптів написаних для оболонки Борна. Стандартом POSIX є оболонка Корна з обмеженими можливостями.

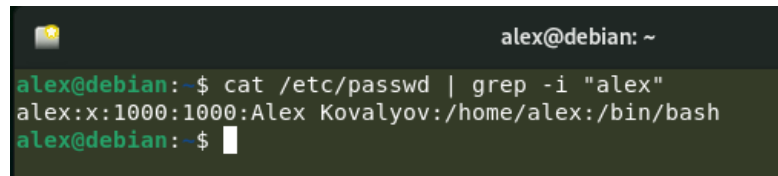
- **zsh** – сучасна оболонка, зворотно сумісна із bash.

У файлі /etc/shells знаходиться інформація про оболонки, присутні у системі:

A terminal window with a dark background and light green text. The prompt is 'alex@debian: ~'. The user has entered 'cat /etc/shells'. The output shows the list of valid login shells: '/bin/sh', '/bin/bash', '/usr/bin/bash', '/bin/rbash', '/usr/bin/rbash', '/bin/dash', and '/usr/bin/dash'. The prompt returns to 'alex@debian: ~' with a cursor.

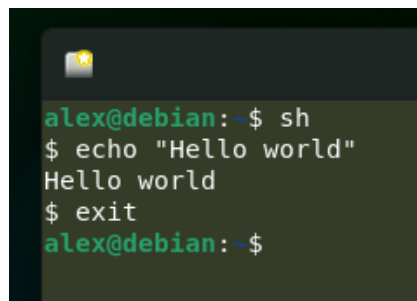
```
alex@debian: ~  
alex@debian:~$ cat /etc/shells  
# /etc/shells: valid login shells  
/bin/sh  
/bin/bash  
/usr/bin/bash  
/bin/rbash  
/usr/bin/rbash  
/bin/dash  
/usr/bin/dash  
alex@debian:~$
```

Оболонка користувача встановлюється у файлі /etc/passwd.



```
alex@debian: ~  
alex@debian:~$ cat /etc/passwd | grep -i "alex"  
alex:x:1000:1000:Alex Kovalyov:/home/alex:/bin/bash  
alex@debian:~$
```

Користувач може в будь-який момент запустити іншу оболонку. Наприклад, якщо він за замовчуванням працює під управлінням bash, а йому зручніше виконати якісь дії під оболонкою sh, він запускає цю оболонку. Для повернення під управління bash потрібно набрати команду exit.



```
alex@debian:~$ sh  
$ echo "Hello world"  
Hello world  
$ exit  
alex@debian:~$
```

Основними функціями командних оболонок є організація діалогу з користувачем (введення команд), виконання внутрішніх команд, запуск зовнішніх програм, виконання програмних файлів. Командні мови в різних оболонках розрізняються, а стандартною прийнято вважати командну мову оболонки bash.

Оболонка має свої налаштування та забезпечує доступ до ресурсів системи, користувач має свої налаштування при роботі з системою та своїми додатками. Для цього використовується середовище оточення.

Середовище оточення - це область, яка містить визначальні властивості системи у вигляді змінних, і це середовище оболонка будує при кожному запуску сесії. Такі змінні можуть містити загальні налаштування системи, параметри графічної або командної оболонки, дані про вподобання користувача, місце розміщення виконуваних файлів в системі, ім'я текстового редактора і багато іншого. Змінні оточення дозволяють простим і надійним способом передавати налаштування відразу для безлічі додатків.

Таким чином, змінні оточення в Linux - це спеціальні змінні, які визначені оболонкою і використовуються програмами під час виконання. Вони можуть визначатися системою і користувачем. Системні змінні оточення Linux визначаються системою і використовуються програмами системного рівня.

Змінні оточення бувають трьох типів:

1. Локальні змінні оточення.
2. Змінні користувача в середовищі оболонки.
3. Системні змінні оточення

Змінні середовища - це змінні, які були визначені для поточної оболонки і успадковуються усіма дочірніми оболонками або процесами. Змінні середовища використовуються для передачі інформації процесам, які запущені з оболонки.

Змінні оболонки - це змінні, які містяться виключно в оболонці, в якій вони були встановлені або визначені. Вони часто використовуються для відстеження поточних даних (наприклад, поточного робочого каталогу).

Хід роботи

Для початку, треба вивести змінні оболонки. Для цього потрібно використати команду `set`. Для зручності, введемо команду з ключем `-o`, який дозволяє вибрати певну опцію. Опцією є аргумент `"posix"`, тобто виведення за стандартом POSIX (тобто пропустить функції оболонки).

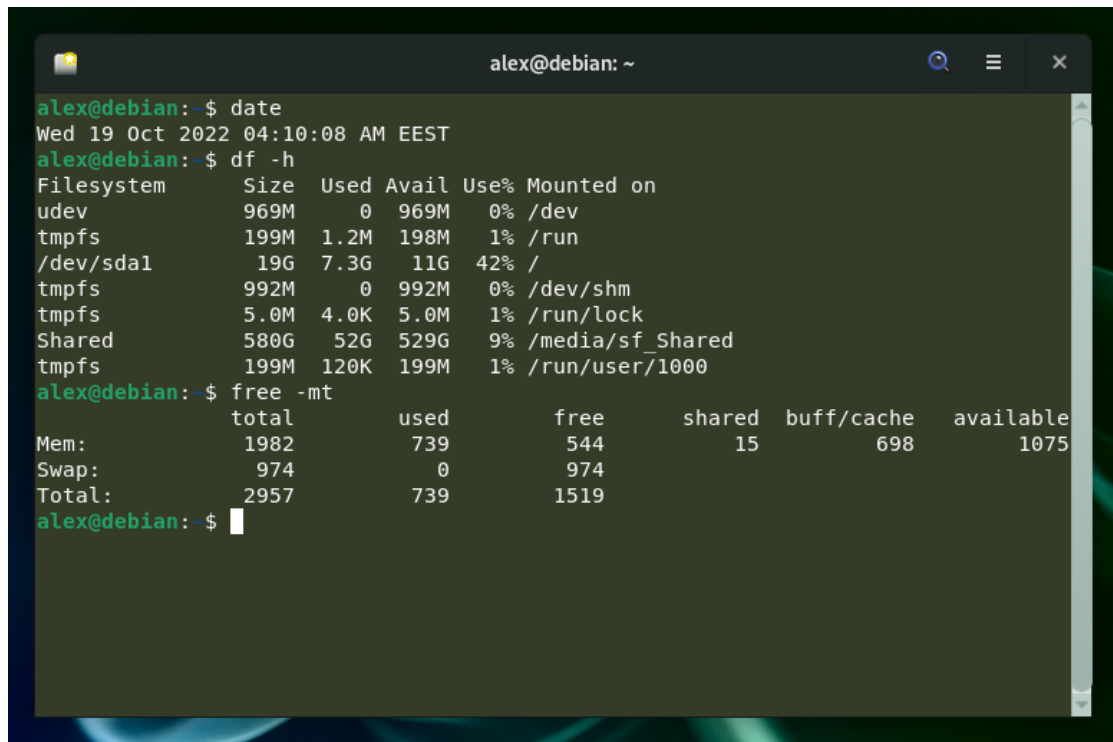
```
alex@debian:~$ set -o posix; set
BASH=/usr/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote
:force_ignore:globasciiranges:histappend:interactive_comments:progcomp:promptva
rs:sourcepath
BASH_ALIASES=()
BASH_ARGC=([0]="0")
BASH_ARGV=()
BASH_CMDS=()
BASH_COMPLETION_VERSION=([0]="2" [1]="11")
BASH_LINENO=()
BASH_SOURCE=()
BASH_VERSION=([0]="5" [1]="1" [2]="4" [3]="1" [4]="release" [5]="x86_64-pc-linu
x-gnu")
BASH_VERSION='5.1.4(1)-release'
COLORTERM=truecolor
COLUMNS=80
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
DESKTOP_SESSION=gnome
DIRSTACK=()
DISPLAY=:0
EUID=1000
GDMSESSION=gnome
GDM_LANG=en_US.UTF-8
```

Наступним кроком потрібно вивести змінні середовища. Це можна зробити за допомогою команд `env` та `printenv`. Ці команди відрізняються декількома індивідуальними функціями, наприклад `printenv` може запитувати значення окремих змінних, а команда `env` дозволяє змінювати середовище, в якій запущені програми, передаючи набір значень змінних в команду.

```
alex@debian:~$ printenv
SHELL=/bin/bash
SESSION_MANAGER=local/debian:@/tmp/.ICE-unix/781,unix/debian:/tmp/.ICE-unix/781
QT_ACCESSIBILITY=1
COLORTERM=truecolor
SSH_AGENT_LAUNCHER=openssh
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
LANGUAGE=en_US:en
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=gnome
GTK_MODULES=gail:atk-bridge
PWD=/home/alex
LOGNAME=alex
XDG_SESSION_DESKTOP=gnome
XDG_SESSION_TYPE=wayland
XAUTHTORITY=/run/user/1000/.mutter-Xwaylandauth.0DQ2T1
GDM_LANG=en_US.UTF-8
HOME=/home/alex
USERNAME=alex
IM_CONFIG_PHASE=1
LANG=en_US.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd
```

Тут можна побачити основні змінні середовища. Наприклад, можна побачити яка командна оболонка використовується на даний момент, яка встановлена тема для терміналу, ім'я користувача, розкладку та інше.

Виведемо поточну дату за допомогою команди `date`, обсяг вільного простору на диску за допомогою команди `df` (disk file system) та обсяг оперативної пам'яті. В утиліті `df` ключ `-h` використовується щоб вивести розмір в мегабайтах та гігабайтах. Для команди `free` використовуємо ключі `-m` та `-t`, тобто показати розмір в мегабайтах та показувати результат в скороченому варіанті відповідно.



```
alex@debian: ~  
alex@debian:~$ date  
Wed 19 Oct 2022 04:10:08 AM EEST  
alex@debian:~$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
udev            969M   0 969M   0% /dev  
tmpfs           199M  1.2M 198M   1% /run  
/dev/sda1       19G   7.3G 11G  42% /  
tmpfs           992M   0 992M   0% /dev/shm  
tmpfs           5.0M   4.0K 5.0M   1% /run/lock  
Shared         580G   52G 529G   9% /media/sf_Shared  
tmpfs           199M  120K 199M   1% /run/user/1000  
alex@debian:~$ free -mt  
              total        used        free      shared  buff/cache   available  
Mem:         1982          739          544           15          698        1075  
Swap:          974           0           974  
Total:       2957          739        1519  
alex@debian:~$
```

Виведемо історію команд за допомогою команди `history`.



```
212 cat /etc/passwd  
213 clear  
214 cat /etc/passwd | grep -i "alex"  
215 csh  
216 clear  
217 sh  
218 clear  
219 set  
220 clear  
221 man set  
222 set --help  
223 clear  
224 set -o posix; set  
225 clear  
226 set --help  
227 clear  
228 printenv  
229 clear  
230 date  
231 df -h  
232 free -mt  
233 man df  
234 history  
alex@debian:~$
```

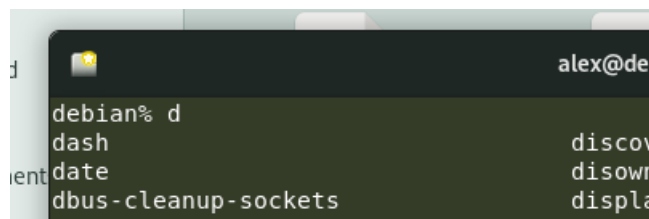
Наступним кроком є робота з оболонкою zsh. Але вона не встановлена, тому треба її скачати. Для початку, знайдемо пакет з цією оболонкою.

```
alex@debian:~$ sudo apt search '^zsh$'
Sorting... Done
Full Text Search... Done
zsh/stable,stable-security 5.8-6+deb11u1 amd64
  shell with lots of features
```

Завантажуємо за допомогою пакетного менеджера apt.

```
alex@debian:~$ sudo apt install zsh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  zsh-common
Suggested packages:
  zsh-doc
The following NEW packages will be installed:
  zsh zsh-common
0 upgraded, 2 newly installed, 0 to remove and 2 not upgraded.
Need to get 4,849 kB of archives.
After this operation, 18.3 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://deb.debian.org/debian bullseye/main amd64 zsh-common all
```

Ця оболонка заснована на bash, і у неї є певні переваги. Наприклад, автодоповнення всіх команд, перехід по папкам без команди cd, можливість встановлення тем, плагінів, та багато чого іншого.



```
alex@deb
debian% d
dash      discov
date      disown
dbus-cleanup-sockets display
```

Контрольні запитання:

1) Що таке командна оболонка?

Командна оболонка - це окремий програмний продукт, який забезпечує прямий зв'язок між користувачем і операційною системою. Основна оболонка в більшості дистрибутивів Linux – bash. Ще існують такі оболонки як sh, zsh, csh, та інші.

2) Які функції виконує командна оболонка?

Основними функціями командних оболонок є організація діалогу з користувачем (введення команд), виконання внутрішніх команд, запуск зовнішніх програм, виконання програмних файлів.

3) Що таке середовище оточення?

Середовище оточення – це область, яка містить визначальні властивості системи у вигляді змінних, і це середовище оболонка буде при кожному запуску сесії. Такі змінні можуть містити загальні налаштування системи, параметри графічної або командної оболонки, дані про вподобання

користувача, місце розміщення виконуваних файлів в системі, ім'я текстового редактора і багато іншого. Змінні оточення дозволяють простим і надійним способом передавати налаштування відразу для безлічі додатків.

4) Що таке змінні оточення?

Це спеціальні змінні, які визначені оболонкою і використовуються програмами під час виконання. Вони можуть визначатися системою і користувачем. Системні змінні оточення Linux визначаються системою і використовуються програмами системного рівня.

5) Які типи змінних оточення вам відомі?

Локальні змінні оточення (визначені лише для поточної сесії), змінні користувача в середовищі оболонки (визначаються для конкретного користувача і завантажуються кожний раз, коли він входить в систему за допомогою локального терміналу, або ж підключається віддалено), та системні змінні оточення.

6) Які основні змінні середовища?

SHELL (вказує шлях до оболонки), TERM (вид терміналу), LOGNAME (реєстраційне ім'я користувача), USER (поточний користувач), PWD (поточний каталог), PATH (список каталогів до яких звертається оболонка при спробі запустити програму) та інші.

7) Які основні змінні оболонки?

BASH_VERSION (версія оболонки), COLUMNS (ширина виведення тексту), UID (унікальний ідентифікатор користувача) та інші.

Висновок: за результатами виконання цієї лабораторної роботи було ознайомлено з такими командними оболонками як bash, zsh, sh та інші. Був закладений фундамент у розумінні будови цих оболонок, що таке змінні середовища, оболонки, і так далі. Була проведена робота з базовими командами вищевказаних оболонок.