

## Лабораторна робота № 3

### Робота в оболонці **bash**, середовище оточення

#### Мета роботи:

- набути навичок налаштування програмного середовища Linux.

#### *Теоретичні відомості*

#### Командні оболонки ОС Unix

У системах Unix використовуються різні командні оболонки (command shells), які називаються також командними процесорами або інтерпретаторами команд. Командна оболонка забезпечує взаємодію між користувачем та ядром системи.

Серед них найбільш відомі і поширені:

- *sh* (*Bourne shell*) `/bin/sh` - оболонка Борна (не дуже зручна в роботі);
- *csh* (*C-shell*) `/bin/csh` - оболонка C (зручніша у порівнянні з *sh*, але несумісна з нею по командній мові);
- *ksh* (*Korn shell*) `/bin/ksh` - оболонка Корна (містить потужну командну мову, засновану на мові *sh*, та розвинені засоби інтерактивної роботи);
- *bash* (*Bourne-Again Shell*) `/bin/bash` - оболонка «Борна» (зручна для інтерактивної роботи, створена на основі *sh* і багато в чому з нею сумісна);
- *Z shell*, *zsh* `/bin/zsh` - одна з сучасних командних оболонок UNIX, що використовується безпосередньо як інтерактивна оболонка, або як скриптовий інтерпретатор. *Zsh* є розширенням *bourne shell* з великою кількістю поліпшень.

Тип оболонки, як правило, можна визначити за останнім символом запрошення:

- знак долара («\$») вказує на *sh*-сумісну оболонку (*sh*, *bash*, *ksh*),
- знак амперсанда «&» відповідає оболонці *csh*.

Однак у привілейованого користувача незалежно від командного процесора, який використовується, останнім символом запрошення зазвичай буває знак решітки «#».

#### Основними функціями командних оболонок є:

- організація діалогу з користувачем (введення команд);
- виконання внутрішніх команд;
- запуск зовнішніх програм;
- виконання командних файлів.

Командні мови в різних оболонках розрізняються, а стандартною прийнято вважати командну мову оболонки *bash*.

В графічній оболонці програма «Термінал», яка запускається комбінацією

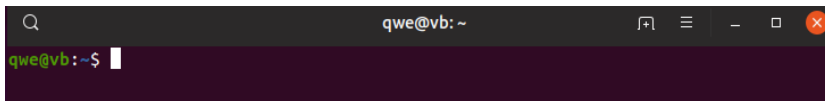


клавіш «Ctrl+Alt+T» або ярлик із загального меню програм, це важливий елемент операційної системи, який дозволяє запускати програми, створювати папки, копіювати і видаляти файли, встановлювати додатки і т.д. Системна утиліта, в яку ви передаєте ці команди, називається *Shell* або командна оболонка. За замовчуванням в Ubuntu використовується командна оболонка, яка називається *Bash*.

При вході в командний інтерпретатор відкривається вікно, в якому відображається:

*qwe* – ім'я облікового запису користувача, *vb* – ім'я комп'ютера, символ «:» - розділювач,

«~» - каталог виконання команди (*домашній каталог*).



## ***Синтаксис команд в Терміналі***

Команди Терміналу, як правило, складаються з назви програми, ключа і значення. В загальному вигляді виглядають так: **назва\_програми [-ключ] [значення]**.

**назва\_програми** - це ім'я виконуваного файлу з каталогів, записаних у змінну \$PATH (/bin, /sbin, /usr/bin, /usr/sbin, /usr/local/bin, /usr/local/sbin та ін.);

**[-ключ]** - опції програми, які може приймати виконувана програма;

**[значення]** - даний параметр може приймати в якості аргументу цифри, текст, спеціальні символи і навіть змінні.

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.***

Це змінні, які визначені лише для поточної сесії. Вони будуть зразу стерті після завершення сесії, будь то віддалений доступ або емулятор терміналу. Вони не зберігаються ні в яких файлах, а створюються і видаляються за допомогою спеціальних команд.

### ***2. Змінні користувача в середовищі оболонки.***

Ці змінні оболонки в Linux визначаються для конкретного користувача і завантажуються кожний раз, коли він входить в систему за допомогою локального терміналу, або ж підключається віддалено. Такі змінні зберігаються в файлах конфігурації: **.bashrc**, **.bash\_profile**, **.bash\_login**, **.profile** або в інших файлах, розміщених в директорії користувача. Ці файли є налаштуванням оболонки користувача. Вони складаються з команд **bash** і виконуються перед тим як запустити оболонку або завантажити систему.

Різниця цих файлів полягає у тому, коли вони виконуються, а саме:

**.profile** та **.bash\_profile** виконуються при вході в систему, тобто один раз - це при введенні логін користувача. Файл **.bashrc** виконується кожний раз, коли користувач відкриває нове вікно терміналу (аналог автозавантаження в Windows).

**Bash** також працює зі **скриптами** (сценаріями - **scripting language**), тобто списком команд, які записані у файл. Сценарій – це програма, яка працює з готовими компонентами.

### 3. Системні змінні оточення

Ці змінні доступні у всій системі та для всіх користувачів. Вони завантажуються при старті системи з системних файлів конфігурації: */etc/environment*, */etc/profile*, */etc/profile.d/\*.sh*, */etc/bash.bashrc*:

- */etc/environment* –системний файл конфігурації, який означає, що він використовується всіма користувачами. Він належить *root*, тому необхідно бути адміністратором і використовувати його *sudo* для зміни;

- *~/.profile* є одним із сценаріїв ініціалізації особистої оболонки вашого власного користувача. Кожний користувач має один такий файл і може його редагувати.

- */etc/profile* та */etc/profile.d/\*.sh* є сценаріями глобальної ініціалізації, які є еквівалентними *~/.profile* для кожного користувача. Глобальні сценарії виконуються раніше користувацьких сценаріїв; і *main /etc/profile* виконує усі *\*.sh* сценарії з директорії */etc/profile.d/* безпосередньо перед виходом;

- */etc/bash.bashrc* - це загальносистемна версія файлу *~/.bashrc*. ОС Ubuntu налаштована за замовчуванням для виконання цього файлу, коли користувач вводить оболонку або середовище робочого столу.

Файл */etc/environment* зазвичай містить тільки наступний рядок:

***PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games"***

Він встановлює змінну *PATH* для всіх користувачів в системі в це значення за замовчуванням, яке не повинно бути змінено основним способом. Принаймні, ви не повинні видалити будь - який з таких важливих шляхів, як */bin*, */sbin*, */usr/bin* та */usr/sbin* з нього.

Цей файл читається як один з перших файлів конфігурації кожною оболонкою кожного користувача, це не скрипт оболонки.

Середовище має вигляд рядка, що містить пари виду «ключ-значення».

Ключ – це ім'я змінної. Кілька значень, як правило, поділяються символом двокрапки (:). Кожна пара, в цілому, виглядає таким чином:

**КЛЮЧ = значення1: значення2: ...**

Якщо ж значення містить пробіли, потрібно використовувати подвійні лапки:

**КЛЮЧ = "значення з пробілами".**

В даному випадку під ключем маються на увазі змінні одного з двох існуючих видів: *змінні середовища* або *змінні оболонки*.

*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).

Зазвичай такі змінні позначаються за допомогою *великих літер*. Це допомагає користувачам розрізняти змінні середовища в інших контекстах.

У сценаріях зазвичай використовуються оголошення виду:

*ім'я\_змінної="значення змінної"*,

але конкретний синтаксис залежить від інтерпретатора, який використовується.

Для отримання значення змінної необхідно перед її ім'ям поставити символ долара. Також іноді потрібно її ім'я взяти у дужки (наприклад, в сценаріях утиліти *make*).

### ***Виведення змінних оболонки (set) і середовища (env або printenv)***

Кожна сесія відстежує свої змінні оболонки і середовища. Вивести їх можна кількома способами. Щоб переглянути список всіх змінних середовища, використовуйте команди *env* або *printenv*. За замовчуванням команди виведуть однаковий результат *env*

```
qwe@vb:~$ env
SHELL=/bin/bash
SESSION_MANAGER=local/vb:@/tmp/.ICE-unix/1363,unix/vb:/tmp/.ICE-unix/1363
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GTK_IM_MODULE=ibus
LANGUAGE=ru_UA:ru
QT4_IM_MODULE=xim
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
SSH_AGENT_PID=1508
GTK_MODULES=gail:atk-bridge
XDG_SEAT=seat0
PWD=/home/qwe
LOGNAME=qwe
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=x11
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
GDMSESSION=ubuntu
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
_=/usr/bin/env
```

### ***printenv***

```
qwe@vb:~$ printenv
SHELL=/bin/bash
SESSION_MANAGER=local/vb:@/tmp/.ICE-unix/1363,unix/vb:/tmp/.ICE-unix/1363
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GTK_IM_MODULE=ibus
LANGUAGE=ru_UA:ru
QT4_IM_MODULE=xim
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
SSH_AGENT_PID=1508
GTK_MODULES=gail:atk-bridge
XDG_SEAT=seat0
PWD=/home/qwe
LOGNAME=qwe
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=x11
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
GDMSESSION=ubuntu
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
_=/usr/bin/printenv
```

Це типовий приклад результату, який виведений командами *env* та *printenv*. Ці команди відрізняються лише кількома індивідуальними функціями. Наприклад, *printenv* може запитувати значення окремих змінних:

*printenv SHELL*

/bin/bash

Команда *env* дозволяє змінювати середовище, в якій запуснені програми, передаючи набір значень змінних в команду:

```
env VAR1="blahblah" command_to_run command_options
```

Дочірні процеси зазвичай успадковують змінні середовища батьківського процесу, що дає можливість змінювати значення або вносити додаткові змінні для дочірніх процесів. При виведенні команди *printenv* багато змінних середовища створені за допомогою системних файлів і процесів без втручання користувача.

Для перегляду змінних оболонки використовується команда *set*. При введенні без додаткових параметрів команда *set* виводить список всіх змінних оболонки, змінних середовища, локальних змінних і функцій оболонки:

```
set
BASH=/usr/bin/bash
BASHOPTS=checkwinsize:cmdhist:expand_aliases:extglob:extquote:force_
ignore:histappend:interactive_comments:login_shell:progcomp:promptvars:
sourcepath
BASH_ALIASES=()
BASH_ARGC=()
BASH_ARGV=()
BASH_CMDS=()
. . .
```

Цей список досить довгий. Для виведення списку у більш зручному форматі, відкрийте його за допомогою програми-пейджера: **set | less**.

```
BASH=/usr/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote
:force_ignores:globasciiranges:histappend:interactive_comments:progcomp:promptva
rs:sourcepath
BASH_ALIASES=()
BASH_ARGC=([0]="0")
BASH_ARGV=()
BASH_CMDS=()
BASH_COMPLETION_VERSION=([0]="2" [1]="8")
BASH_LINENO=()
BASH_SOURCE=()
BASH_VERSION=([0]="5" [1]="0" [2]="3" [3]="1" [4]="release" [5]="x86_64-pc-linu
x-gnu")
BASH_VERSION='5.0.3(1)-release'
CLUTTER_IM_MODULE=xim
COLORTERM=truecolor
COLUMNS=80
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
DESKTOP_SESSION=ubuntu
DIRSTACK=()
DISPLAY=:0
EUID=1000
GDMSESSION=ubuntu
```

Список містить величезну кількість додаткової інформації, яка в даний момент не потрібна (наприклад, деякі функції *bash*). Щоб у більш зручному форматі вивести результат, потрібно запустити команду *set* в режимі *POSIX*, який пропускає функції оболонки. Це потрібно виконати в субоболонці, щоб не змінити поточне середовище:

***set -o posix; set***

Команда виведе усі змінні середовища і оболонки.

### Основні змінні середовища і оболонки

Деякі особливо корисні змінні середовища і оболонки використовуються дуже часто. Нижче наведено **список основних змінних середовища**:

• SHELL - описує оболонку, яка є інтерпретатором введених команд. У більшості випадків за замовчуванням встановлена оболонка **bash**, але це значення можна змінити в разі потреби.

```
SHELL=/bin/bash
```

• TERM - вказує вид терміналу, який емулюється при запуску оболонки. Залежно від операційних вимог можна емулювати різні апаратні термінали.

```
TERM=xterm-256color
```

• LOGNAME - реєстраційне ім'я користувача.

```
LOGNAME=qwe
```

• USER - поточний користувач.

```
USER=qwe
```

• PWD - поточний робочий каталог.

```
PWD=/home/qwe
```

• LS\_COLORS - визначає кольорові коди, які використовуються для кольорового виведення результату команди **ls**. Таке виведення допомагає користувачеві швидше прочитати результат команди (наприклад, швидко розрізнити типи файлів).

```
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=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01
```

• PATH - список каталогів, до яких звертається система при виконанні команд. Коли користувач запускає команду, система перевіряє ці каталоги у зазначеному порядку в пошуках виконуваного файлу.

```
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

• LANG - поточні налаштування мови та локалізації, в тому числі кодування символів.

```
LANG=ru_UA.UTF-8
```

• HOME - ім'я домашнього каталогу поточного користувача.

```
HOME=/home/qwe
```

• \_ : остання виконана команда

```
_=/usr/bin/printenv
```

### Список основних змінних оболонки:

• BASHOPTS: список опцій, які використовуються при виконанні **bash**.

```
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote:force_fignore:globasciiranges:histappend:interactive_comments:progcomp:promptvars:sourcpath
```

• BASH\_VERSION: запущена версія **bash** в зрозумілій формі.

```
BASH_VERSION='5.0.3(1)-release'
```

• BASH\_VERSINFO: версія **bash** в машиночитасьому форматі.

```
BASH_VERSINFO=( [0]="5" [1]="0" [2]="3" [3]="1" [4]="release" [5]="x86_64-pc-linux-gnu" )
```

• COLUMNS: визначає ширину виведення у стовбцях.

```
COLUMNS=108
```

• DIRSTACK: стек каталогів, доступних командам **pushd** і **popd**.

```
DIRSTACK=()
```

• HISTFILESIZE: максимальна кількість рядків, що міститься у файлі історії команд.

```
HISTFILESIZE=2000
```

• HISTSIZE: Кількість команд, які необхідно запам'ятовувати в списку історії.

```
HISTSIZE=1000
```

• HOSTNAME: поточне ім'я хоста.

```
HOSTNAME=vb
```

• IFS: Внутрішній розділювач полів введення у командному рядку. За замовчуванням встановлено пробіл.

```
IFS=''
```



• PS1: визначає рядок первинного запрошення - вид командного рядка при запуску сесії оболонки. Змінна PS2 встановлює рядок вторинного запрошення, якщо команда займає декілька рядків.

```
PS1='\\[\\e\\0;\\u@\\h: \\w\\a\\]\\${debian_chroot:+(\\$debian_chroot)}\\[\\033[01;32m\\]\\u@\\h\\[\\033[00m\\]:\\[\\033[01;34m\\]\\w\\[\\033[00m\\]\\$ '
```

• SHELLOPTS: параметри оболонки, які можна встановити за допомогою *set*.

```
SHELLOPTS=braceexpand:emacs:hashall:histexpand:history:interactive-comments:monitor:posix
```

• UID: унікальний ідентифікатор поточного користувача.

```
UID=1000
```

Приклад ~/.profile

```
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ] ; then
    PATH="$HOME/bin:$PATH"
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/.local/bin" ] ; then
    PATH="$HOME/.local/bin:$PATH"
fi
```

Команда *date* виводить поточну дату.

Утиліта *grep* використовується для пошук рядка у тексті за шаблоном. В якості шаблону використовуються регулярні вирази. Ключі цієї утиліти можна подивитися за допомогою команди *man grep*. Наприклад, *grep 'word' filename*.

Облікові записи користувачів знаходяться у файлі /etc/passwd. Для пошуку свого облікового запису можна скористатися командою *grep qwe /etc/passwd*.

```
qwe@vb:~$ grep qwe /etc/passwd
qwe:x:1000:1000:qwe,,,:/home/qwe:/bin/bash
```

Команда *df* показує обсяг вільного простору на диску для всіх файлових систем.

Приклад використання: *df -h*

- *h*: показує об'єм диска в кіло-і мегабайтах, а не байтах.

Команда *free* показує обсяг вільної та зайнятої оперативної пам'яті.

Приклад використання: *free -mt*

- *m*: показує обсяг в мегабайтах, а не кілобайтах; - *g*: показує обсяг в гігабайтах;

- *t*: відображає підсумковий рядок.

У випадку, коли виникає потреба послідовно виконати декілька команд, є можливість записати виклики декількох команд в одному рядку. Роздільником команд є «;» (крапка з комою). При цьому для виконання наступної команди не треба чекати завершення попередньої.

Синтаксис наступний: **command\_1; command\_2; command\_3**

У випадку, коли необхідно виконання декількох команд в одному рядку і умовою є виконання наступної команди лише в тому випадку, якщо успішно відпрацює попередня. Синтаксис наступний: **command\_1 && command\_2**

Наприклад, **sudo apt update && sudo apt upgrade**

Слід пам'ятати, що при випадковому натисканні комбінації «Ctrl+S» відбувається блокування терміналу. Для переведення терміналу в робочий стан треба натиснути комбінацію клавіш «Ctrl+Q».

За допомогою символів «!!» можна викликати всю попередню команду. Наприклад, для виконання команди з правами суперкористувача можна набрати **sudo !!**

Для призупинення виконання команди, яка виконується, необхідно натиснути «Ctrl+C».

Команда **history** виводить команди сеансу користувача.

### **Завдання:**

1. Ознайомитися з теоретичними матеріалом по лабораторній роботі.
2. Опанувати команди по роботі зі змінними оболонки та середовища.
3. Підготувати звіт для викладача про виконання лабораторної роботи і представити його.

### **Хід виконання роботи**

1. Вивести змінні оболонки.
2. Вивести змінні середовища.
3. Вивести поточну дату, обсяг вільного простору на диску, обсяг оперативної пам'яті.
4. За допомогою команди *history* виведіть команди, які ви використовували.
5. Описати можливості *Z shell* оболонки.

### **Підготувати звіт**

1. Описати хід виконання поставлених завдань, надаючи знімок екрану (screenshot).
2. Висновки по роботі.

### **Контрольні питання**

1. Що таке командна оболонка?
2. Які функції виконує командна оболонка?
3. Що таке середовище оточення?
4. Що таке змінні оточення?
5. Які типи змінних оточення вам відомі?
6. Які основні змінні середовища?
7. Які основні змінні оболонки?

### **Література**

1. Роббинс А. Bash. Карманный справочник системного администратора. СПб.: ООО "Альфа-книга". 2017. 152 с.
2. Колисниченко Д. Н. Руководство по командам и shell-программированию в Linux. СПб.: БХВ-Петербург. 2011. 288 с.