CLOUD COMPUTING

LAB 07

Submitted To:

Engr. Shoaib

Submitted By:

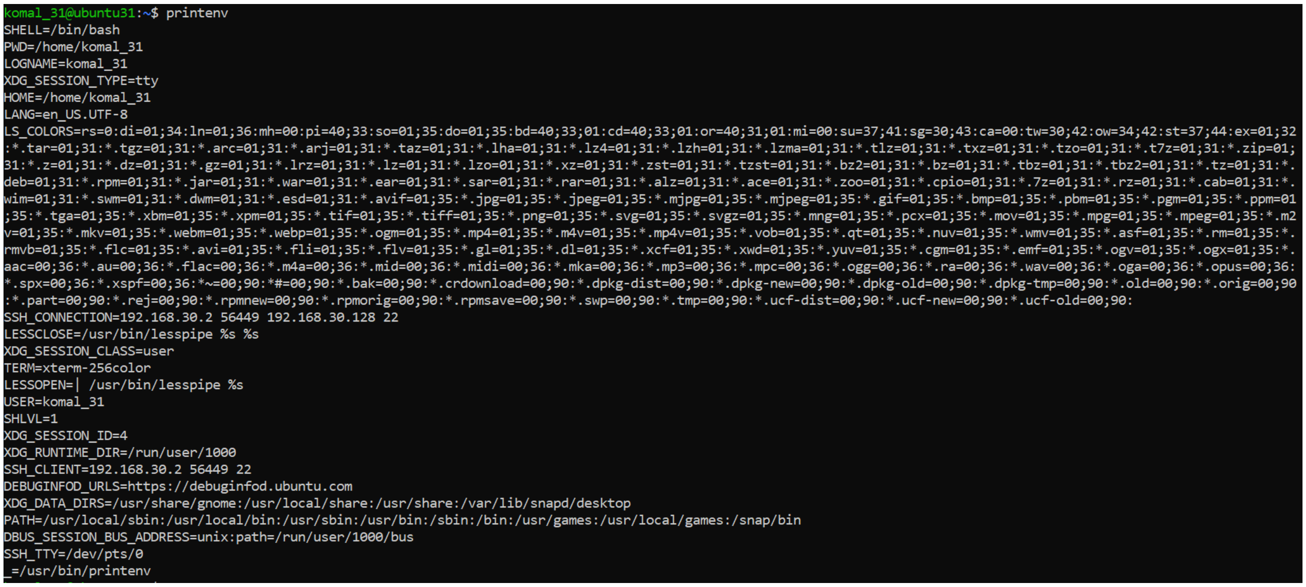
Komal Kashif

BSE V-A

2023-BSE-031

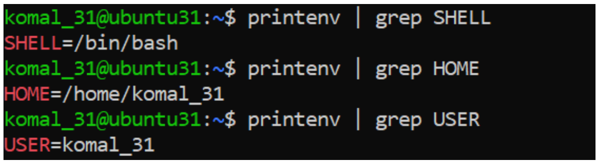
Task 1 — Print & filter environment variables

1. Print all environment variables:



```
komal_31@ubuntu31:~$ printenv
SHELL=/bin/bash
PWD=/home/komal_31
LOGNAME=komal_31
XDG_SESSION_TYPE=tty
HOME=/home/komal_31
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=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=00:tw=30;42:ow=34;42:st=37;44:ex=01;32
:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;
31:*.z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.
deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.
wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.avif=01;35:*.jpg=01;35:*.jpeg=01;35:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01
;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2
v=01;35:*.mkv=01;35:*.webm=01;35:*.webp=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*.rm=01;35:*.
rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.
aac=00;36:*.au=00;36:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:
*.spx=00;36:*.xspf=00;36:*~=00;90:*#=00;90:*.bak=00;90:*.crdownload=00;90:*.dpkg-dist=00;90:*.dpkg-new=00;90:*.dpkg-old=00;90:*.dpkg-tmp=00;90:*.old=00;90:*.orig=00;90
:*.part=00;90:*.rej=00;90:*.rpmnew=00;90:*.rpmorig=00;90:*.rpmsave=00;90:*.swp=00;90:*.tmp=00;90:*.ucf-dist=00;90:*.ucf-new=00;90:*.ucf-old=00;90:
SSH_CONNECTION=192.168.30.2 56449 192.168.30.128 22
LESSCLOSE=/usr/bin/lesspipe %s %s
XDG_SESSION_CLASS=user
TERM=xterm-256color
LESSOPEN=| /usr/bin/lesspipe %s
USER=komal_31
SHLVL=1
XDG_SESSION_ID=4
XDG_RUNTIME_DIR=/run/user/1000
SSH_CLIENT=192.168.30.2 56449 22
DEBUGINFOD_URLS=https://debuginfod.ubuntu.com
XDG_DATA_DIRS=/usr/share/gnome:/usr/local/share:/usr/share:/var/lib/snapd/desktop
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
SSH_TTY=/dev/pts/0
_=/usr/bin/printenv
```

2. Filter for SHELL, HOME and USER — run these greps together and capture one combined screenshot:



```
komal_31@ubuntu31:~$ printenv | grep SHELL
SHELL=/bin/bash
komal_31@ubuntu31:~$ printenv | grep HOME
HOME=/home/komal_31
komal_31@ubuntu31:~$ printenv | grep USER
USER=komal_31
```

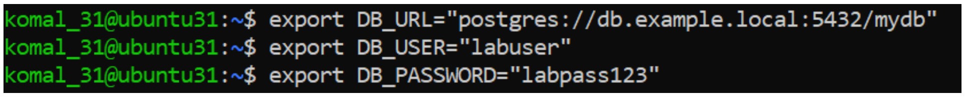
Task 2 — Export DB_* variables temporarily and observe scope

1. Define all DB_* variables

export DB_URL="postgres://db.example.local:5432/mydb"

export DB_USER="labuser"

export DB_PASSWORD="labpass123"

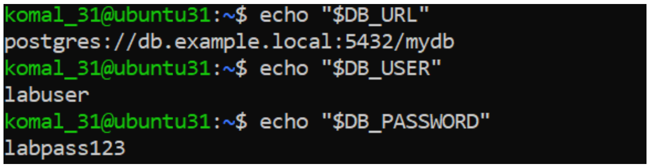


```
komal_31@ubuntu31:~$ export DB_URL="postgres://db.example.local:5432/mydb"
komal_31@ubuntu31:~$ export DB_USER="labuser"
komal_31@ubuntu31:~$ export DB_PASSWORD="labpass123"
```

2. Echo the three variables echo "$DB_URL"

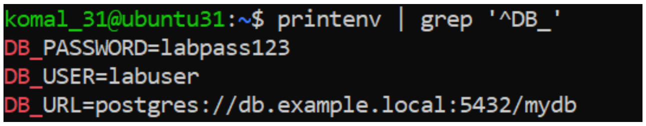
echo "$DB_USER"

echo "$DB_PASSWORD"



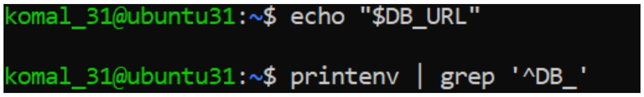
```
komal_31@ubuntu31:~$ echo "$DB_URL"
postgres://db.example.local:5432/mydb
komal_31@ubuntu31:~$ echo "$DB_USER"
labuser
komal_31@ubuntu31:~$ echo "$DB_PASSWORD"
labpass123
```

3. Show all DB_ variables with a single grep command



```
komal_31@ubuntu31:~$ printenv | grep '^DB_'
DB_PASSWORD=labpass123
DB_USER=labuser
DB_URL=postgres://db.example.local:5432/mydb
```

4. Close the bash session (e.g., exit) and reopen a new terminal. Verify the variables are gone by running the echo(s) and the grep together



```
komal_31@ubuntu31:~$ echo "$DB_URL"

komal_31@ubuntu31:~$ printenv | grep '^DB_'
```

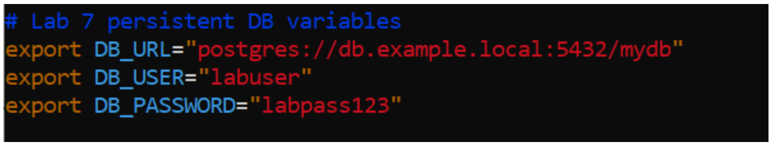
Task 3 — Make DB_* variables persistent in ~/.bashrc

1. Open ~/.bashrc in an editor and append the three export lines:

   export DB_URL="postgres://db.example.local:5432/mydb"

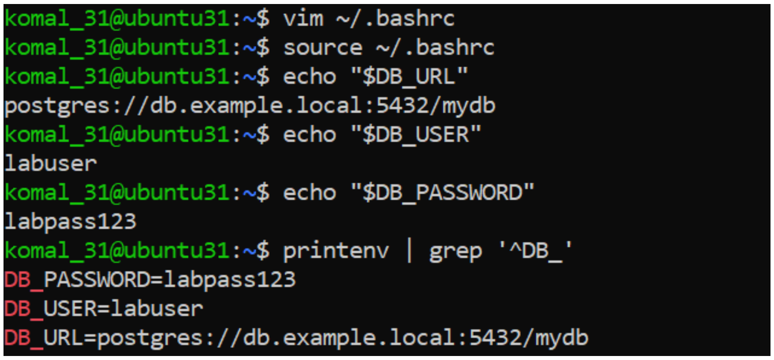
   export DB_USER="labuser"

   export DB_PASSWORD="labpass123"



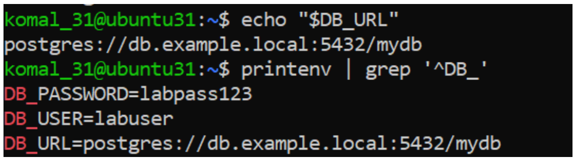
```
# Lab 7 persistent DB variables
export DB_URL="postgres://db.example.local:5432/mydb"
export DB_USER="labuser"
export DB_PASSWORD="labpass123"
```

2. Source ~/.bashrc and capture the source command in one screenshot together with the next verification commands (grouped): run source ~/.bashrc and then immediately run the three echoes and a single grep, capturing all of these in one screenshot:



```
komal_31@ubuntu31:~$ vim ~/.bashrc
komal_31@ubuntu31:~$ source ~/.bashrc
komal_31@ubuntu31:~$ echo "$DB_URL"
postgres://db.example.local:5432/mydb
komal_31@ubuntu31:~$ echo "$DB_USER"
labuser
komal_31@ubuntu31:~$ echo "$DB_PASSWORD"
labpass123
komal_31@ubuntu31:~$ printenv | grep '^DB_'
DB_PASSWORD=labpass123
DB_USER=labuser
DB_URL=postgres://db.example.local:5432/mydb
```

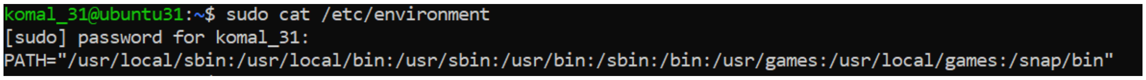
3. Close and reopen terminal. Verify persistence by running one echo and the grep together:



```
komal_31@ubuntu31:~$ echo "$DB_URL"
postgres://db.example.local:5432/mydb
komal_31@ubuntu31:~$ printenv | grep '^DB_'
DB_PASSWORD=labpass123
DB_USER=labuser
DB_URL=postgres://db.example.local:5432/mydb
```

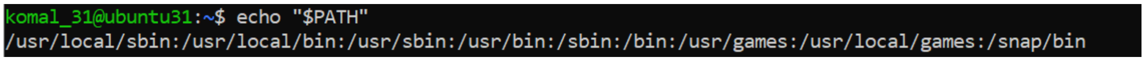
Task 4 — System-wide environment variable, welcome script, and PATH

1. View /etc/environment:



```
komal_31@ubuntu31:~$ sudo cat /etc/environment
[sudo] password for komal_31:
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"
```

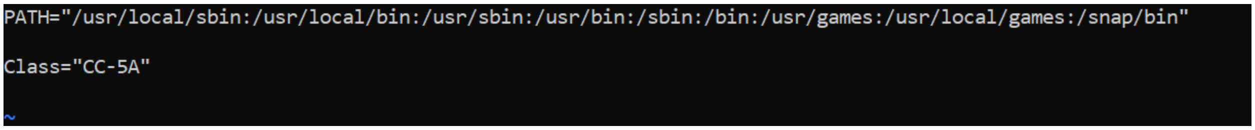
2. Show current PATH:



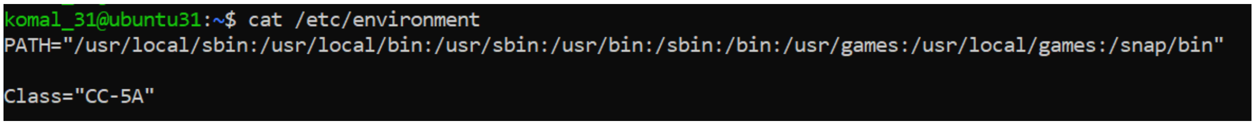
```
komal_31@ubuntu31:~$ echo "$PATH"
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

3. Edit /etc/environment and add Class:

# add line: Class="CC-<your_class_name>"

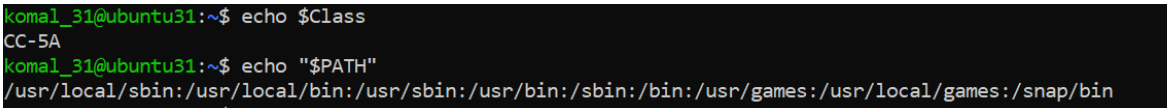


```
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"

Class="CC-5A"

~
```



```
komal_31@ubuntu31:~$ cat /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"

Class="CC-5A"
```

4. Re-login or open a new shell and show Class and PATH together (grouped prints): run echo $Class and echo $PATH



```
komal_31@ubuntu31:~$ echo $Class
CC-5A
komal_31@ubuntu31:~$ echo "$PATH"
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

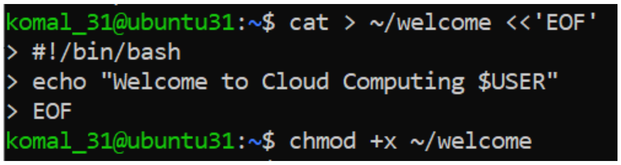
5. Create welcome script at your home directory (~/welcome) and make it executable

cat > ~/welcome <<'EOF'

#!/bin/bash

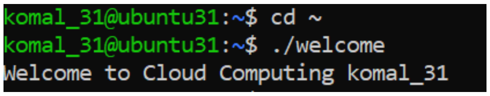
echo "Welcome to Cloud Computing $USER"

EOF



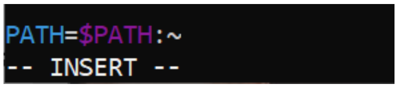
```
komal_31@ubuntu31:~$ cat > ~/welcome <<'EOF'
> #!/bin/bash
> echo "Welcome to Cloud Computing $USER"
> EOF
komal_31@ubuntu31:~$ chmod +x ~/welcome
```

6. Run the script from your home directory using ./welcome:



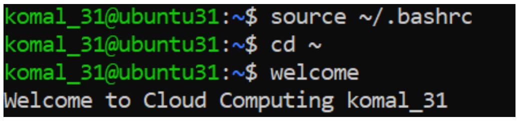
```
komal_31@ubuntu31:~$ cd ~
komal_31@ubuntu31:~$ ./welcome
Welcome to Cloud Computing komal_31
```

7. Add your home directory to PATH in ~/.bashrc.



```
PATH=$PATH:~
-- INSERT --
```

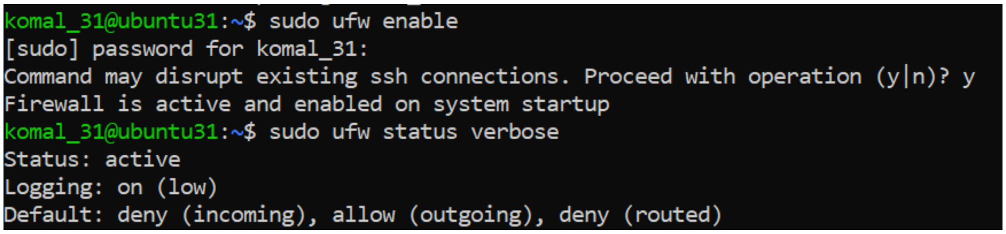
8. Apply the change and run welcome



```
komal_31@ubuntu31:~$ source ~/.bashrc
komal_31@ubuntu31:~$ cd ~
komal_31@ubuntu31:~$ welcome
Welcome to Cloud Computing komal_31
```

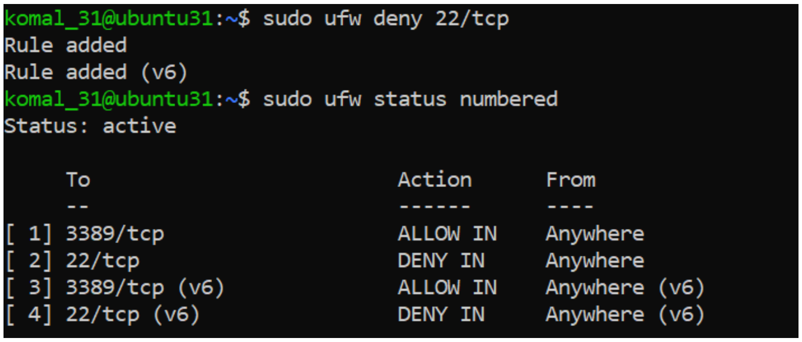
Task 5 — Block and allow SSH using ufw (firewall)

1. Enable ufw and show status



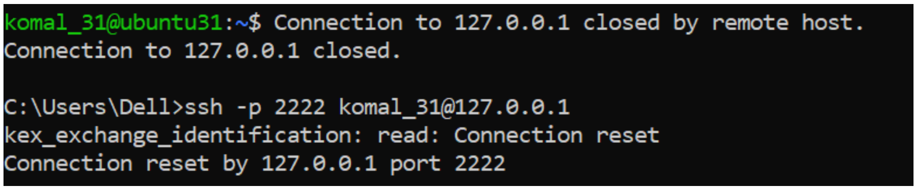
```
komal_31@ubuntu31:~$ sudo ufw enable
[sudo] password for komal_31:
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
komal_31@ubuntu31:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), deny (routed)
```

2. Deny TCP port 22 and show status



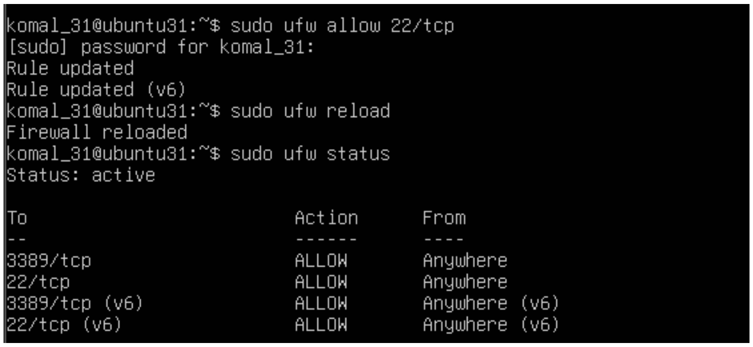
```
komal_31@ubuntu31:~$ sudo ufw deny 22/tcp
Rule added
Rule added (v6)
komal_31@ubuntu31:~$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 3389/tcp                   ALLOW IN    Anywhere
[ 2] 22/tcp                     DENY IN     Anywhere
[ 3] 3389/tcp (v6)              ALLOW IN    Anywhere (v6)
[ 4] 22/tcp (v6)                DENY IN     Anywhere (v6)
```

3. From Windows host attempt to SSH (expected to fail)



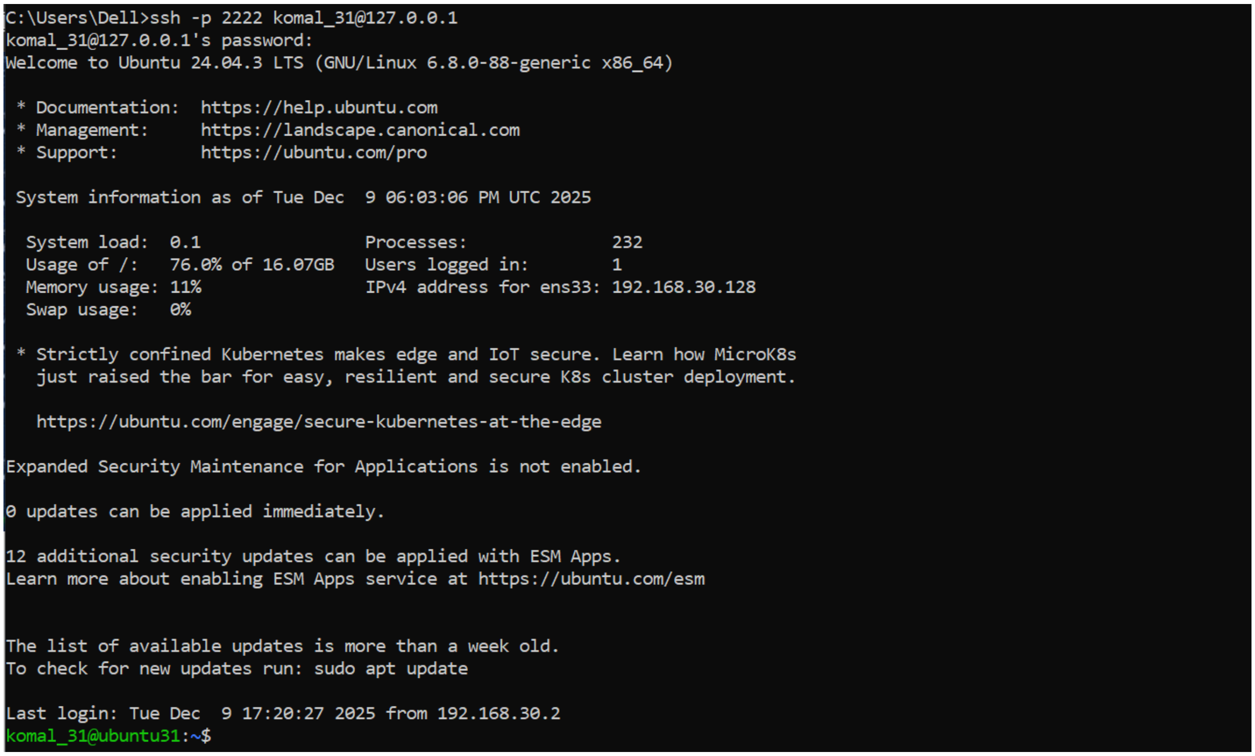
```
komal_31@ubuntu31:~$ Connection to 127.0.0.1 closed by remote host.
Connection to 127.0.0.1 closed.

C:\Users\Dell>ssh -p 2222 komal_31@127.0.0.1
kex_exchange_identification: read: Connection reset
Connection reset by 127.0.0.1 port 2222
```

4. Allow SSH back and reload, then show status (group allow, reload, status in one screenshot if run together).



```
komal_31@ubuntu31:~$ sudo ufw allow 22/tcp
[sudo] password for komal_31:
Rule updated
Rule updated (v6)
komal_31@ubuntu31:~$ sudo ufw reload
Firewall reloaded
komal_31@ubuntu31:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
3389/tcp                   ALLOW       Anywhere
22/tcp                     ALLOW       Anywhere
3389/tcp (v6)              ALLOW       Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

5. From Windows host attempt SSH again (should succeed)



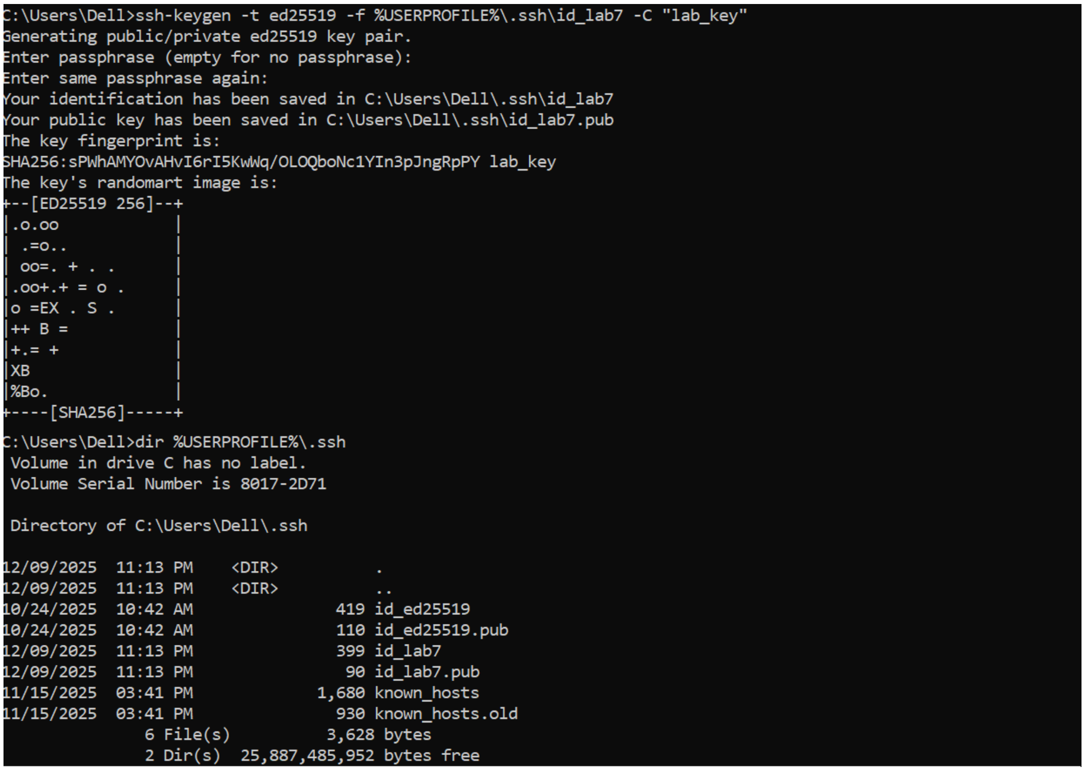
```
C:\Users\Dell>ssh -p 2222 komal_31@127.0.0.1
komal_31@127.0.0.1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Dec  9 06:03:06 PM UTC 2025

  System load:  0.1               Processes:             232
  Usage of /:   76.0% of 16.07GB   Users logged in:       1
  Memory usage: 11%               IPv4 address for ens33: 192.168.30.128
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

12 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Dec  9 17:20:27 2025 from 192.168.30.2
komal_31@ubuntu31:~$
```

Task 6 — Configure SSH key-based login from Windows host

A. On Windows host (client) — group related client actions:

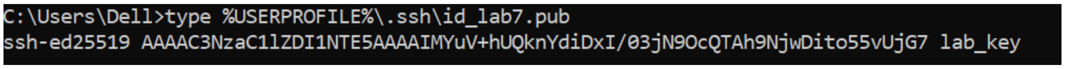
1. Generate ed25519 key pair (if needed) and show the generated files in one screenshot (run ssh-keygen and then list ~/.ssh):

   ssh-keygen -t ed25519 -f ~/.ssh/id_lab7 -C "lab_key"



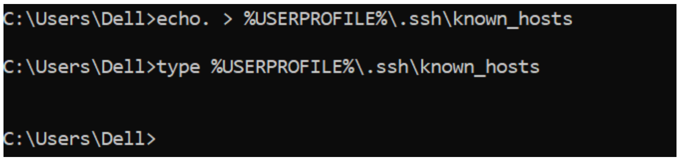
```
C:\Users\Dell>ssh-keygen -t ed25519 -f %USERPROFILE%\.ssh\id_lab7 -C "lab_key"
Generating public/private ed25519 key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Dell\.ssh\id_lab7
Your public key has been saved in C:\Users\Dell\.ssh\id_lab7.pub
The key fingerprint is:
SHA256:sPWhAMYOvAHvI6rI5KwWq/OLOQboNc1YIn3pJngRpPY lab_key
The key's randomart image is:
+--[ED25519 256]--+
|.o.oo            |
| .=o..           |
| oo=. + . .      |
|.oo+.+ = o .     |
|o =EX . S .      |
|++ B =           |
|+.= +            |
|XB               |
|%Bo.             |
+----[SHA256]-----+
C:\Users\Dell>dir %USERPROFILE%\.ssh
 Volume in drive C has no label.
 Volume Serial Number is 8017-2D71

 Directory of C:\Users\Dell\.ssh

12/09/2025  11:13 PM    <DIR>          .
12/09/2025  11:13 PM    <DIR>          ..
10/24/2025  10:42 AM               419 id_ed25519
10/24/2025  10:42 AM               110 id_ed25519.pub
12/09/2025  11:13 PM               399 id_lab7
12/09/2025  11:13 PM                90 id_lab7.pub
11/15/2025  03:41 PM             1,680 known_hosts
11/15/2025  03:41 PM               930 known_hosts.old
               6 File(s)          3,628 bytes
               2 Dir(s)  25,887,485,952 bytes free
```

2. Show the public key content



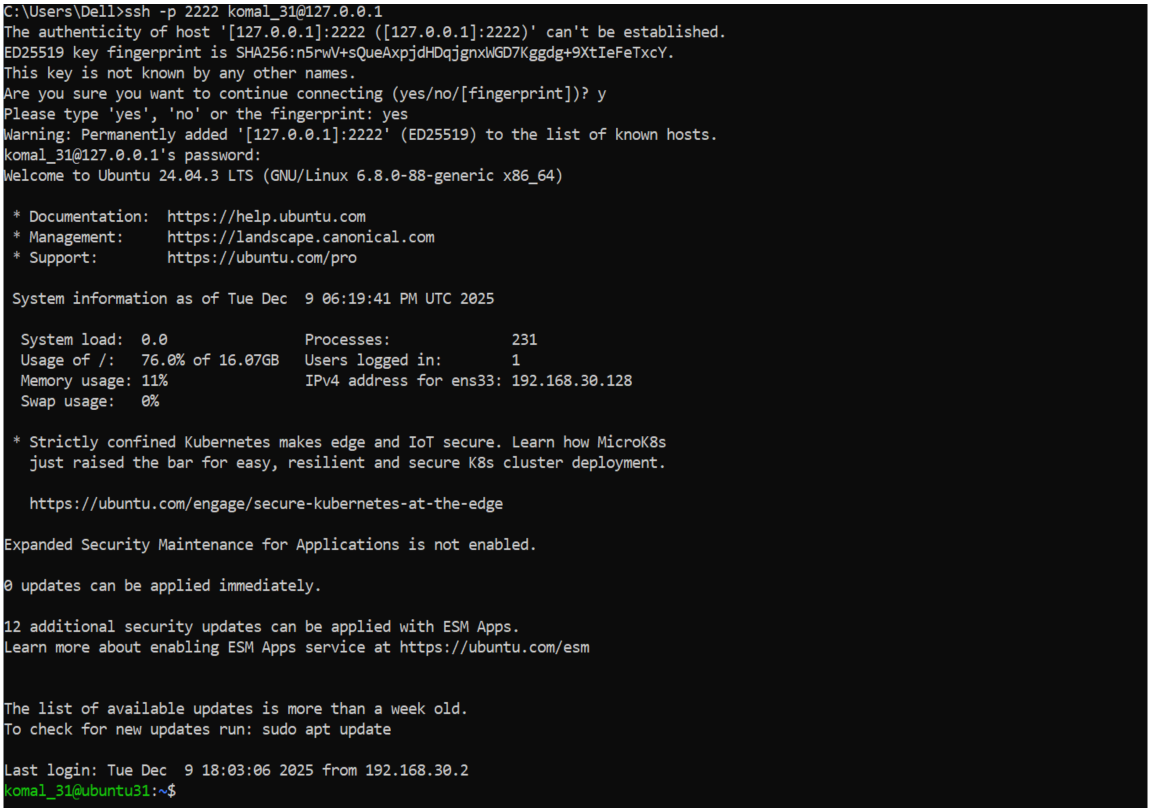
```
C:\Users\Dell>type %USERPROFILE%\.ssh\id_lab7.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMYuV+hUQknYdiDxI/03jN9OcQTAh9NjwDito55vUjG7 lab_key
```

3. Clear the known_hosts file content and verify it is empty:



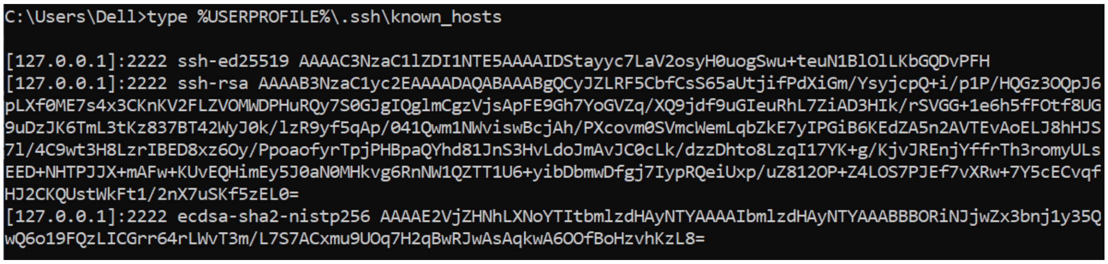
```
C:\Users\Dell>echo. > %USERPROFILE%\.ssh\known_hosts

C:\Users\Dell>type %USERPROFILE%\.ssh\known_hosts


C:\Users\Dell>
```

4. Connect to the Ubuntu server using the standard SSH command (this will prompt to accept the server host key because known_hosts is empty).



```
C:\Users\Dell>ssh -p 2222 komal_31@127.0.0.1
The authenticity of host '[127.0.0.1]:2222 ([127.0.0.1]:2222)' can't be established.
ED25519 key fingerprint is SHA256:n5rwV+sQueAxpjdHDqjgnxWGD7Kggdg+9XtIeFeTxcY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '[127.0.0.1]:2222' (ED25519) to the list of known hosts.
komal_31@127.0.0.1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Dec  9 06:19:41 PM UTC 2025

  System load:  0.0               Processes:             231
  Usage of /:   76.0% of 16.07GB   Users logged in:       1
  Memory usage: 11%               IPv4 address for ens33: 192.168.30.128
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

12 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Dec  9 18:03:06 2025 from 192.168.30.2
komal_31@ubuntu31:~$
```

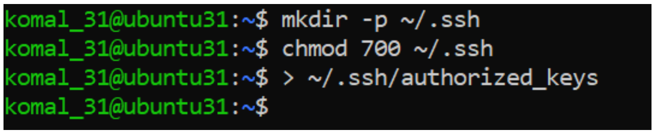
5. After the successful connection, view the known_hosts file to show the server host key was added



```
C:\Users\Dell>type %USERPROFILE%\.ssh\known_hosts

[127.0.0.1]:2222 ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIDStayyc7LaV2osyH0uogSwu+teuN1BlOlLKbGQDvPFH
[127.0.0.1]:2222 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCyJZLRF5CbfCsS65aUtjifPdXiGm/YsyjcpQ+i/p1P/HQGz3OQpJ6
pLXf0ME7s4x3CKnKV2FLZVOMWDPHuRQy7S0GJgIQglmCgzVjsApFE9Gh7YoGVZq/XQ9jdf9uGIeuRhL7ZiAD3HIk/rSVGG+1e6h5fFOtf8UG
9uDzJK6TmL3tKz837BT42WyJ0k/lzR9yf5qAp/041Qwm1NWviswBcjAh/PXcovm0SVmcWemLqbZkE7yIPGiB6KEdZA5n2AVTEvAoELJ8hHJS
7l/4C9wt3H8LzrIBED8xz6Oy/PpoaofyrTpjPHBpaQYhd81JnS3HvLdoJmAvJC0cLk/dzzDhto8LzqI17YK+g/KjvJREnjYffrTh3romyULs
EED+NHTPJJX+mAFw+KUvEQHimEy5J0aN0MHkvg6RnNW1QZTT1U6+yibDbmwDfgj7IypRQeiUxp/uZ812OP+Z4LOS7PJEf7vXRw+7Y5cECvqf
HJ2CKQUstWkFt1/2nX7uSKf5zEL0=
[127.0.0.1]:2222 ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBORiNJjwZx3bnj1y35Q
wQ6o19FQzLICGrr64rLWvT3m/L7S7ACxmu9UOq7H2qBwRJwAsAqkwA6OOfBoHzvhKzL8=
```

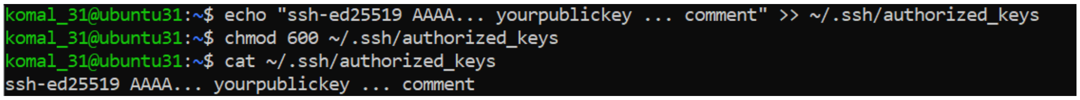
B. On Ubuntu server — group related server-side commands:

1. Prepare the ~/.ssh directory and clear authorized_keys (this will create the directory if missing, set the correct directory permissions, and truncate the authorized_keys file).



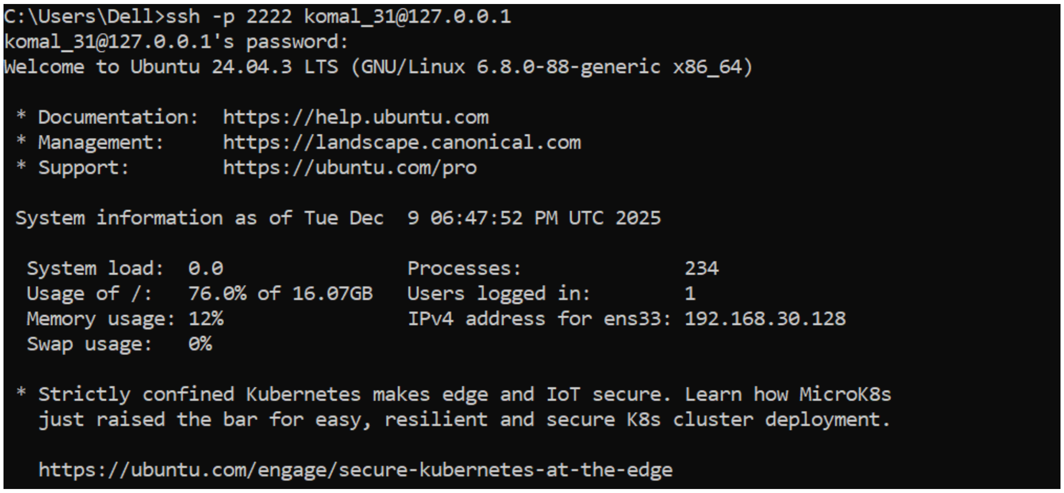
```
komal_31@ubuntu31:~$ mkdir -p ~/.ssh
komal_31@ubuntu31:~$ chmod 700 ~/.ssh
komal_31@ubuntu31:~$ > ~/.ssh/authorized_keys
komal_31@ubuntu31:~$
```

2. Append the public key, set file permissions, and show the resulting authorized_keys



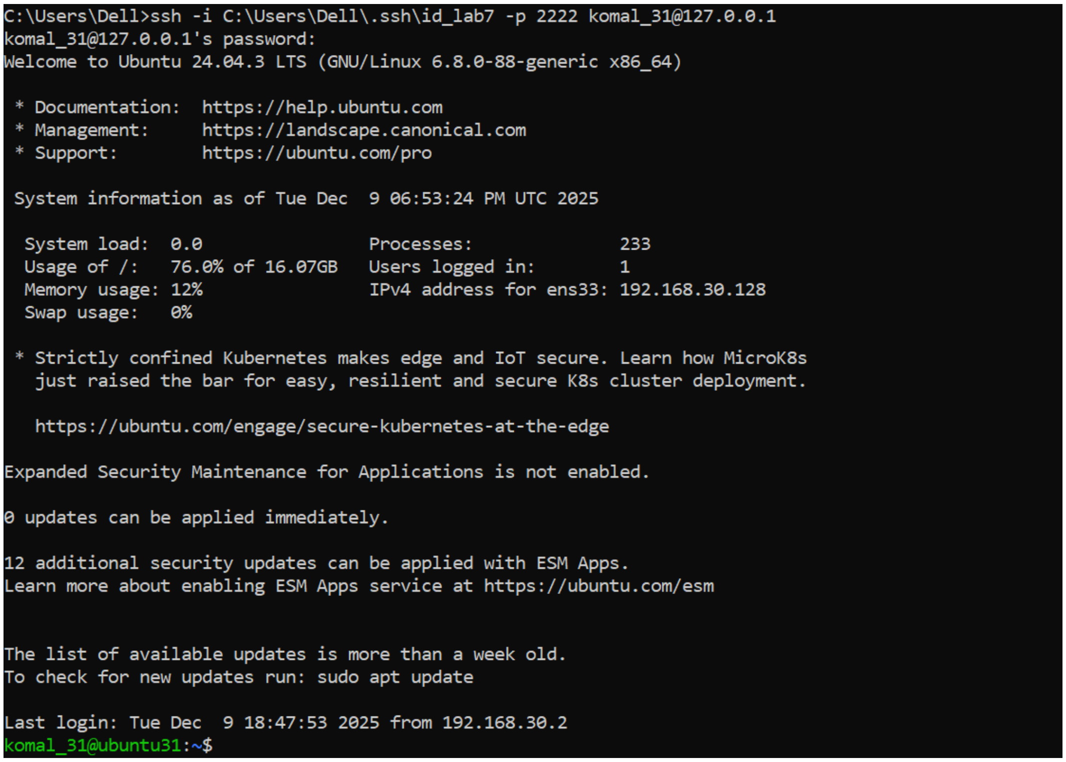
```
komal_31@ubuntu31:~$ echo "ssh-ed25519 AAAA... yourpublickey ... comment" >> ~/.ssh/authorized_keys
komal_31@ubuntu31:~$ chmod 600 ~/.ssh/authorized_keys
komal_31@ubuntu31:~$ cat ~/.ssh/authorized_keys
ssh-ed25519 AAAA... yourpublickey ... comment
```

3. From Windows host test password less login:



```
C:\Users\Dell>ssh -p 2222 komal_31@127.0.0.1
komal_31@127.0.0.1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Dec  9 06:47:52 PM UTC 2025

  System load:  0.0               Processes:             234
  Usage of /:   76.0% of 16.07GB   Users logged in:       1
  Memory usage: 12%               IPv4 address for ens33: 192.168.30.128
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge
```

4. Also demonstrate explicit identity usage



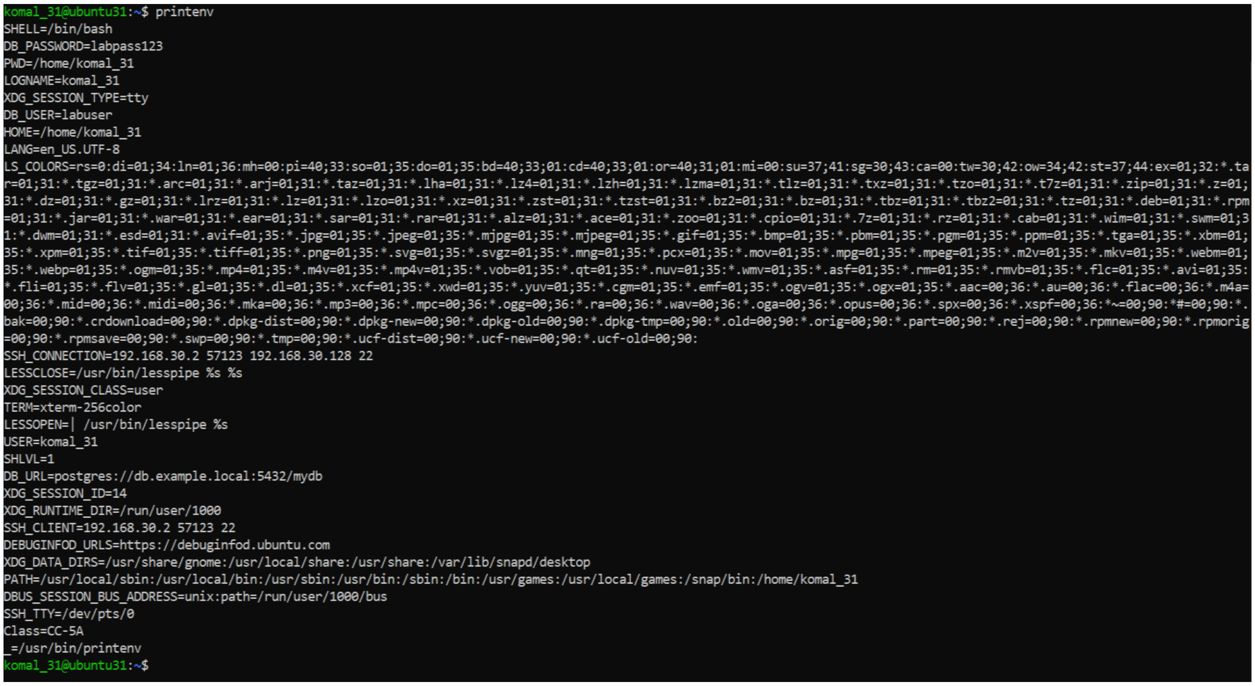
```
C:\Users\Dell>ssh -i C:\Users\Dell\.ssh\id_lab7 -p 2222 komal_31@127.0.0.1
komal_31@127.0.0.1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Tue Dec  9 06:53:24 PM UTC 2025

  System load:  0.0               Processes:             233
  Usage of /:   76.0% of 16.07GB   Users logged in:       1
  Memory usage: 12%               IPv4 address for ens33: 192.168.30.128
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

12 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Dec  9 18:47:53 2025 from 192.168.30.2
komal_31@ubuntu31:~$
```

Exam Evaluation Questions

Q1: Quick Environment Audit

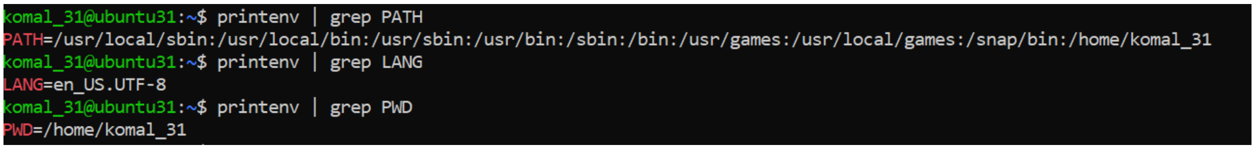
- Objective: Demonstrate you can inspect the current environment and extract a few key variables.
- Actions & evidence:

    i. Run a single command to display environment variables and capture its output.



```
komal_31@ubuntu31:~$ printenv
SHELL=/bin/bash
DB_PASSWORD=labpass123
PWD=/home/komal_31
LOGNAME=komal_31
XDG_SESSION_TYPE=tty
DB_USER=labuser
HOME=/home/komal_31
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=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=00:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.ta
r=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;
31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm
=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.wim=01;31:*.swm=01;3
1:*.dwm=01;31:*.esd=01;31:*.avif=01;35:*.jpg=01;35:*.jpeg=01;35:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;
35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;
35:*.webp=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:
*.fli=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00;36:*.m4a=
00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:*.spx=00;36:*.xspf=00;36:*~=00;90:*#=00;90:*.
bak=00;90:*.crdownload=00;90:*.dpkg-dist=00;90:*.dpkg-new=00;90:*.dpkg-old=00;90:*.dpkg-tmp=00;90:*.old=00;90:*.orig=00;90:*.part=00;90:*.rej=00;90:*.rpmnew=00;90:*.rpmorig
=00;90:*.rpmsave=00;90:*.swp=00;90:*.tmp=00;90:*.ucf-dist=00;90:*.ucf-new=00;90:*.ucf-old=00;90:
SSH_CONNECTION=192.168.30.2 57123 192.168.30.128 22
LESSCLOSE=/usr/bin/lesspipe %s %s
XDG_SESSION_CLASS=user
TERM=xterm-256color
LESSOPEN=| /usr/bin/lesspipe %s
USER=komal_31
SHLVL=1
DB_URL=postgres://db.example.local:5432/mydb
XDG_SESSION_ID=14
XDG_RUNTIME_DIR=/run/user/1000
SSH_CLIENT=192.168.30.2 57123 22
DEBUGINFOD_URLS=https://debuginfod.ubuntu.com
XDG_DATA_DIRS=/usr/share/gnome:/usr/local/share:/usr/share:/var/lib/snapd/desktop
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/home/komal_31
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
SSH_TTY=/dev/pts/0
Class=CC-5A
_=/usr/bin/printenv
komal_31@ubuntu31:~$
```

    ii. In the same terminal session, run three filters (one per line) to show values for PATH, LANG, and PWD

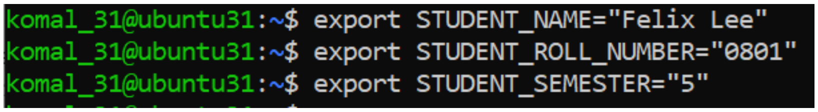


```
komal_31@ubuntu31:~$ printenv | grep PATH
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/home/komal_31
komal_31@ubuntu31:~$ printenv | grep LANG
LANG=en_US.UTF-8
komal_31@ubuntu31:~$ printenv | grep PWD
PWD=/home/komal_31
```

Q2: Short-lived Student Info

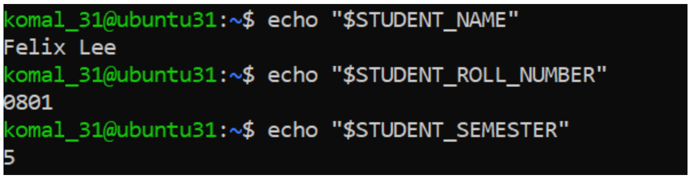
- Objective: Show how temporary environment variables behave (session-scoped).
- Actions & evidence:

    i. In one terminal, set three variables (STUDENT_NAME, STUDENT_ROLL_NUMBER, STUDENT_SEMESTER) using export



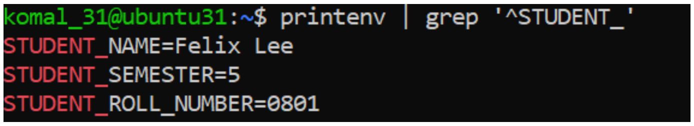
```
komal_31@ubuntu31:~$ export STUDENT_NAME="Felix Lee"
komal_31@ubuntu31:~$ export STUDENT_ROLL_NUMBER="0801"
komal_31@ubuntu31:~$ export STUDENT_SEMESTER="5"
```

ii. Still in the same session, print the three values with echo (grouped)



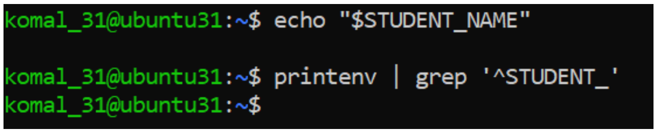
```
komal_31@ubuntu31:~$ echo "$STUDENT_NAME"
Felix Lee
komal_31@ubuntu31:~$ echo "$STUDENT_ROLL_NUMBER"
0801
komal_31@ubuntu31:~$ echo "$STUDENT_SEMESTER"
5
```

iii. Use a single printenv|grep command to list any STUDENT_ variables and capture the result.



```
komal_31@ubuntu31:~$ printenv | grep '^STUDENT_'
STUDENT_NAME=Felix Lee
STUDENT_SEMESTER=5
STUDENT_ROLL_NUMBER=0801
```

iv. Exit that shell, open a fresh terminal, and show that the STUDENT_ variables are not set (use echo and printenv|grep together)

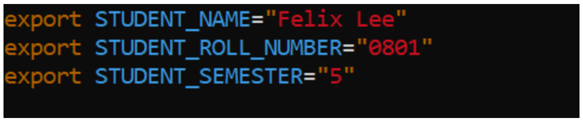


```
komal_31@ubuntu31:~$ echo "$STUDENT_NAME"

komal_31@ubuntu31:~$ printenv | grep '^STUDENT_'
komal_31@ubuntu31:~$
```

Q3: Make It Sticky (Persistence Check for Student Info)

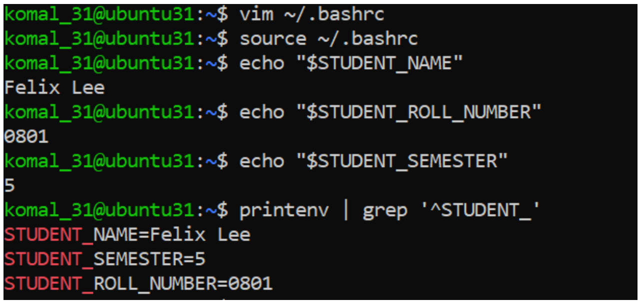
- Objective: Demonstrate persistence of environment variables across sessions via shell configuration.
- Actions & evidence:

i. Edit ~/.bashrc and append the three STUDENT_* exports.



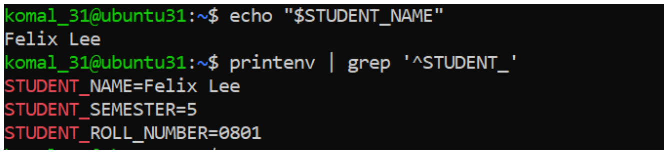
```
export STUDENT_NAME="Felix Lee"
export STUDENT_ROLL_NUMBER="0801"
export STUDENT_SEMESTER="5"
```

ii. Reload your shell config with a single command and then verify the three variables and show printenv | grep '^STUDENT_'



```
komal_31@ubuntu31:~$ vim ~/.bashrc
komal_31@ubuntu31:~$ source ~/.bashrc
komal_31@ubuntu31:~$ echo "$STUDENT_NAME"
Felix Lee
komal_31@ubuntu31:~$ echo "$STUDENT_ROLL_NUMBER"
0801
komal_31@ubuntu31:~$ echo "$STUDENT_SEMESTER"
5
komal_31@ubuntu31:~$ printenv | grep '^STUDENT_'
STUDENT_NAME=Felix Lee
STUDENT_SEMESTER=5
STUDENT_ROLL_NUMBER=0801
```

iii. Close and re-open a terminal and demonstrate the STUDENT_NAME variable is available (echo and printenv grep together)

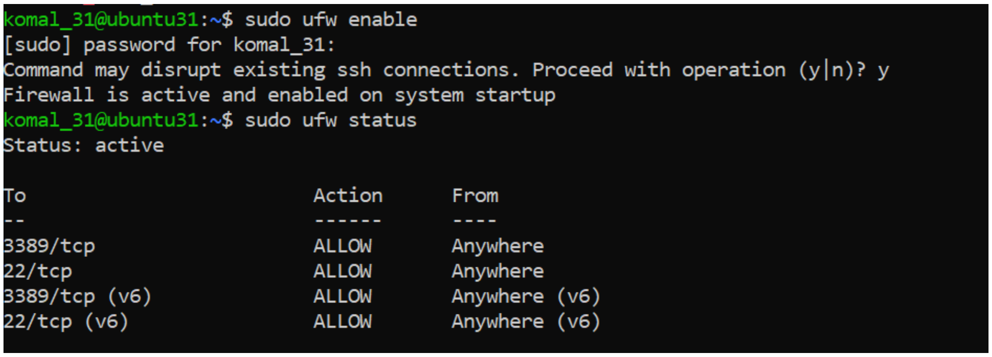


```
komal_31@ubuntu31:~$ echo "$STUDENT_NAME"
Felix Lee
komal_31@ubuntu31:~$ printenv | grep '^STUDENT_'
STUDENT_NAME=Felix Lee
STUDENT_SEMESTER=5
STUDENT_ROLL_NUMBER=0801
```

Q4: Firewall Rules: Block and Restore Ping (ICMP)

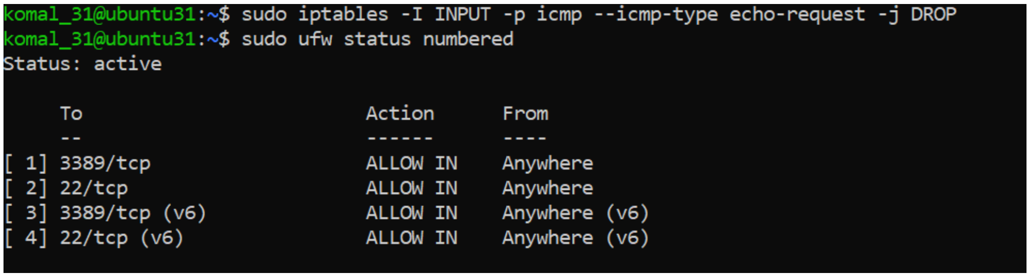
- Objective: Demonstrate you can block ping (ICMP echo) traffic using ufw and then re-allow it; show effect from a client.
- Actions & evidence:

i. Enable ufw



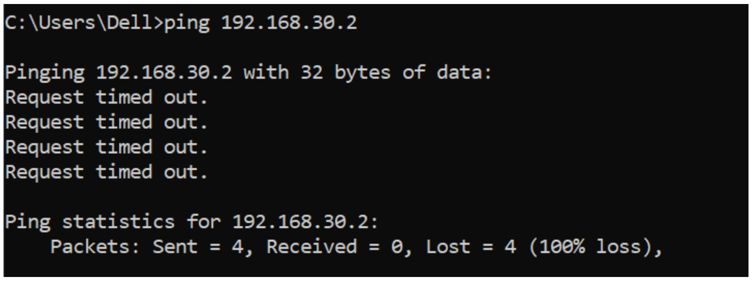
```
komal_31@ubuntu31:~$ sudo ufw enable
[sudo] password for komal_31:
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
komal_31@ubuntu31:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
3389/tcp                   ALLOW       Anywhere
22/tcp                     ALLOW       Anywhere
3389/tcp (v6)              ALLOW       Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

ii. Add a rule to block ping (ICMP echo) and show ufw status numbered



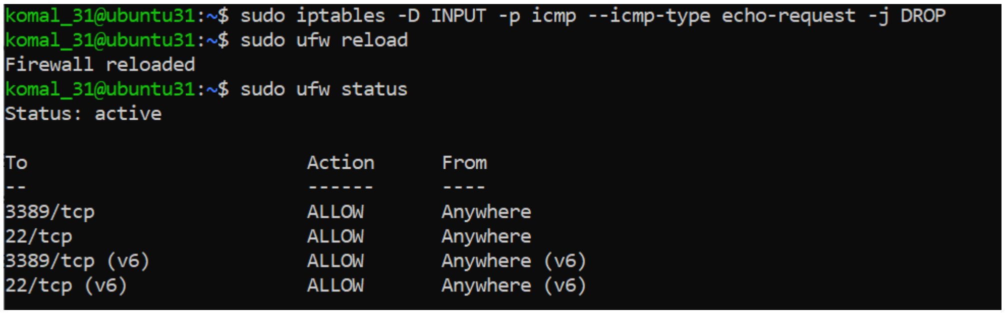
```
komal_31@ubuntu31:~$ sudo iptables -I INPUT -p icmp --icmp-type echo-request -j DROP
komal_31@ubuntu31:~$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 3389/tcp                   ALLOW IN    Anywhere
[ 2] 22/tcp                     ALLOW IN    Anywhere
[ 3] 3389/tcp (v6)              ALLOW IN    Anywhere (v6)
[ 4] 22/tcp (v6)                ALLOW IN    Anywhere (v6)
```

iii. From your Windows host (or another client), attempt to ping the server while the rule is active and capture the blocked/failing ping



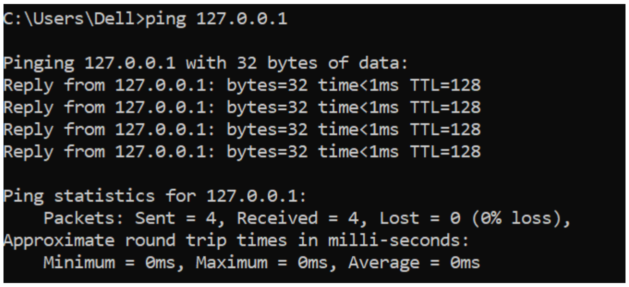
```
C:\Users\Dell>ping 192.168.30.2

Pinging 192.168.30.2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.30.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

iv. Re-allow ping (ICMP) (or remove the deny rule)



```
komal_31@ubuntu31:~$ sudo iptables -D INPUT -p icmp --icmp-type echo-request -j DROP
komal_31@ubuntu31:~$ sudo ufw reload
Firewall reloaded
komal_31@ubuntu31:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
3389/tcp                   ALLOW       Anywhere
22/tcp                     ALLOW       Anywhere
3389/tcp (v6)              ALLOW       Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

v. From the client, ping the server again



```
C:\Users\Dell>ping 127.0.0.1

Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```