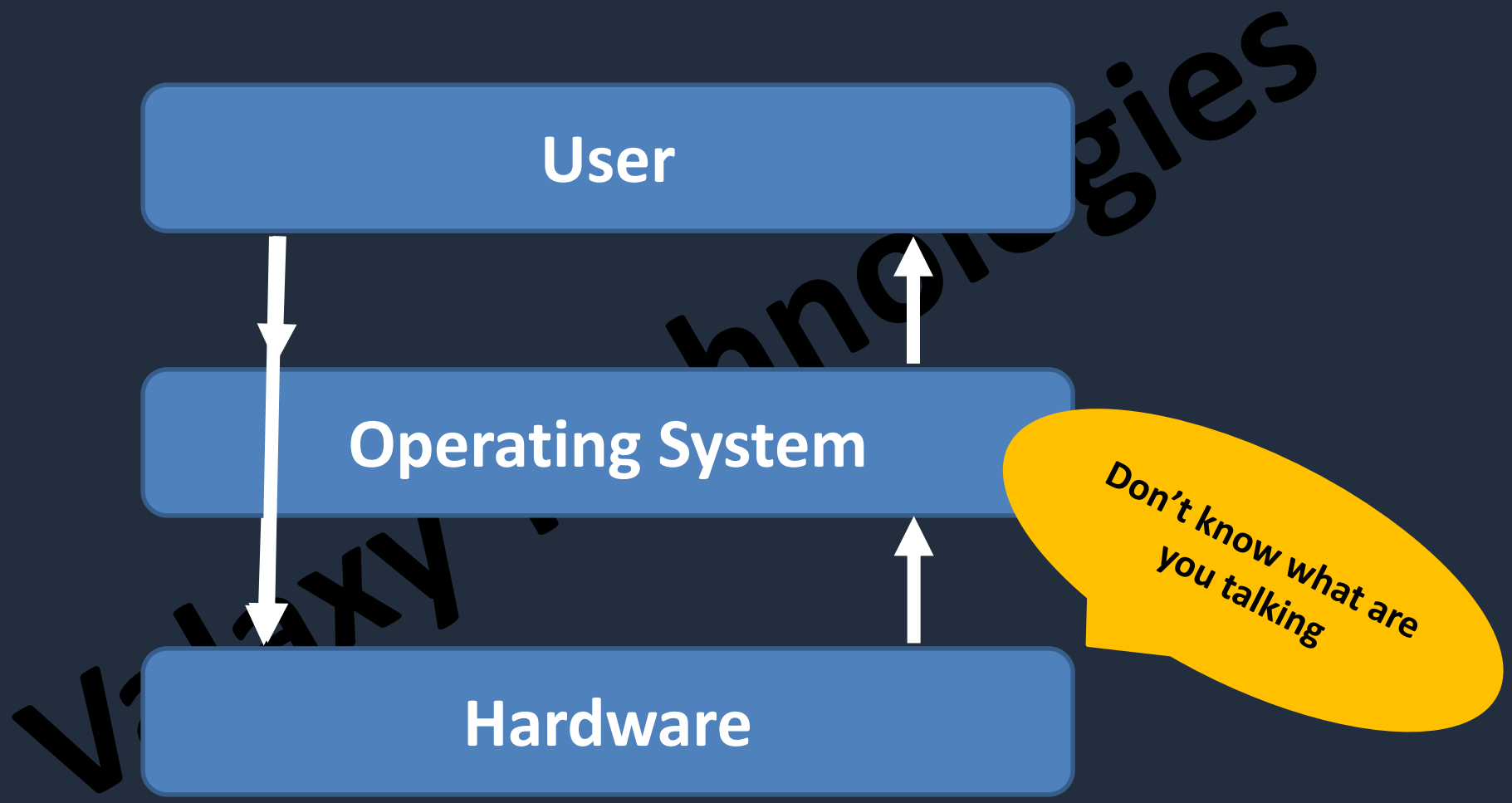
Linux

For Cloud & DevOps Engineers

What is Operating System?

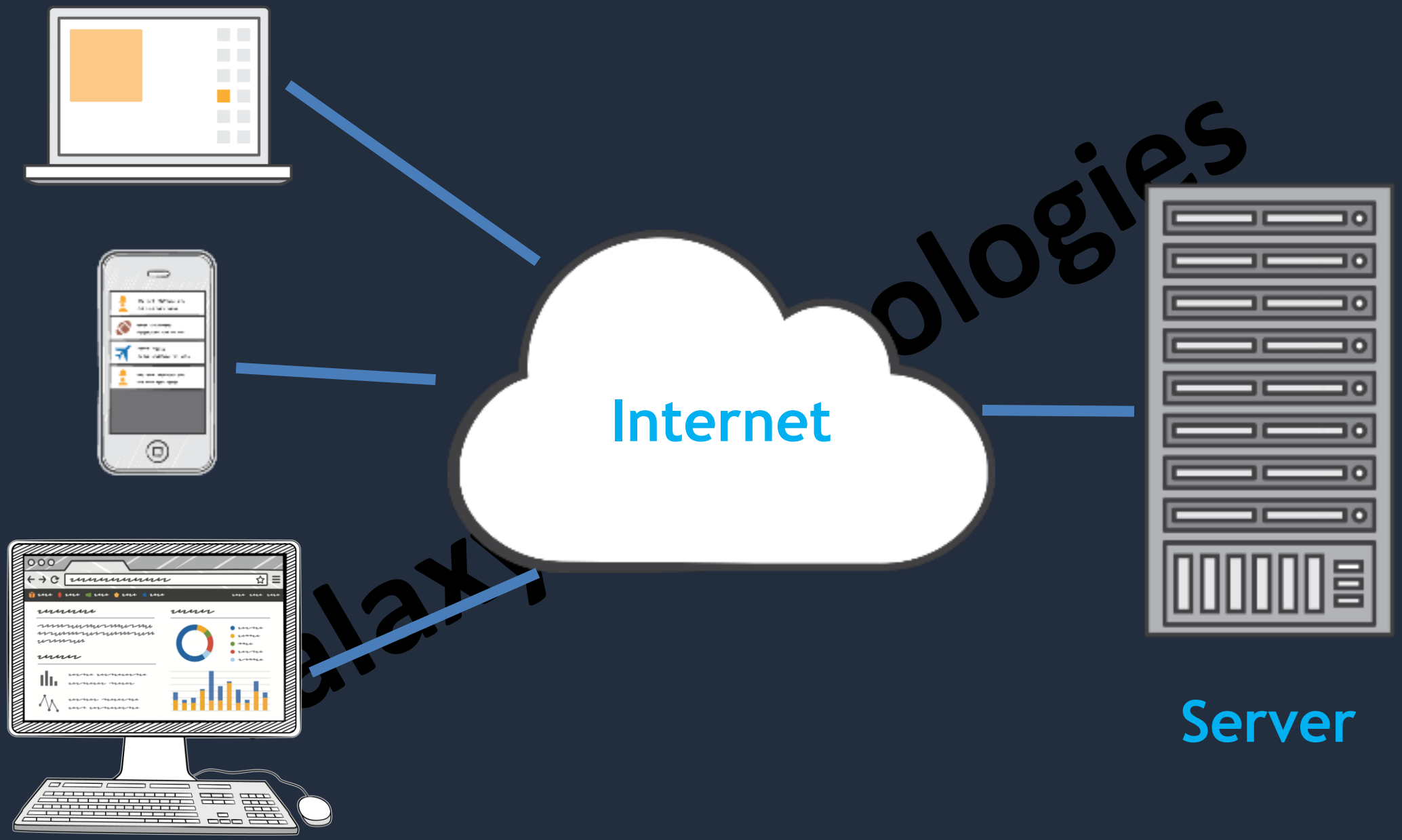
OS is an interface between user and the computer hardware

How It Works


User
Operating System
Hardware
Don’t know what are you talking

Operating Systems

- Windows
- OS X (MAC OS)
- IBM-AIX
- HP-UX
- Solaris
- Linux
  - RedHat, Ubuntu, fedora, Suse, Debian , cent, etc…


Internet
Server

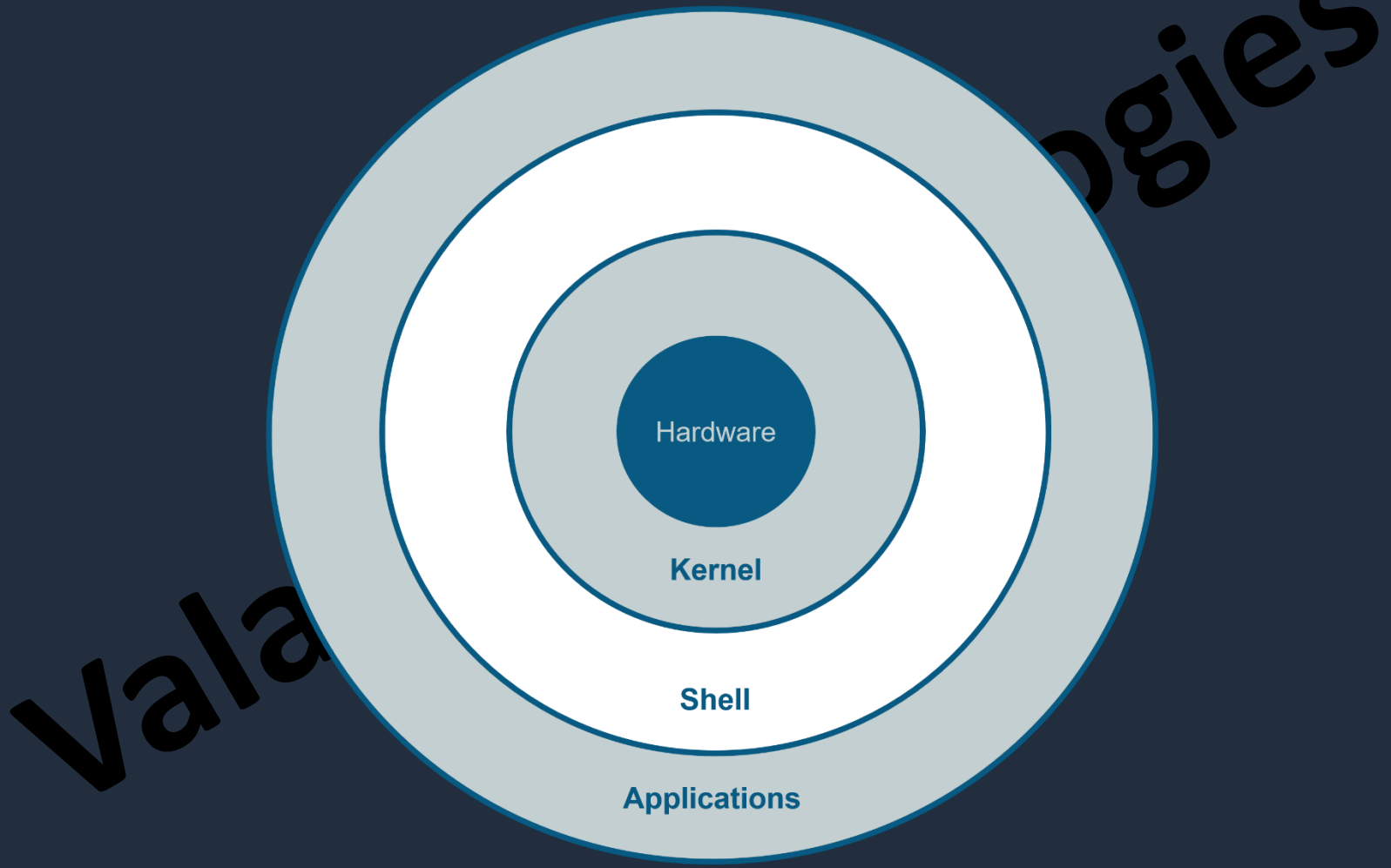
How to get a Linux system

- Install Linux OS directly in Laptop or Desktop
- Install VMware and create a VM
- Install Virtual Box and Create VM
- Provision a Linux VM on Cloud (AWS/Azure/GCP etc..)

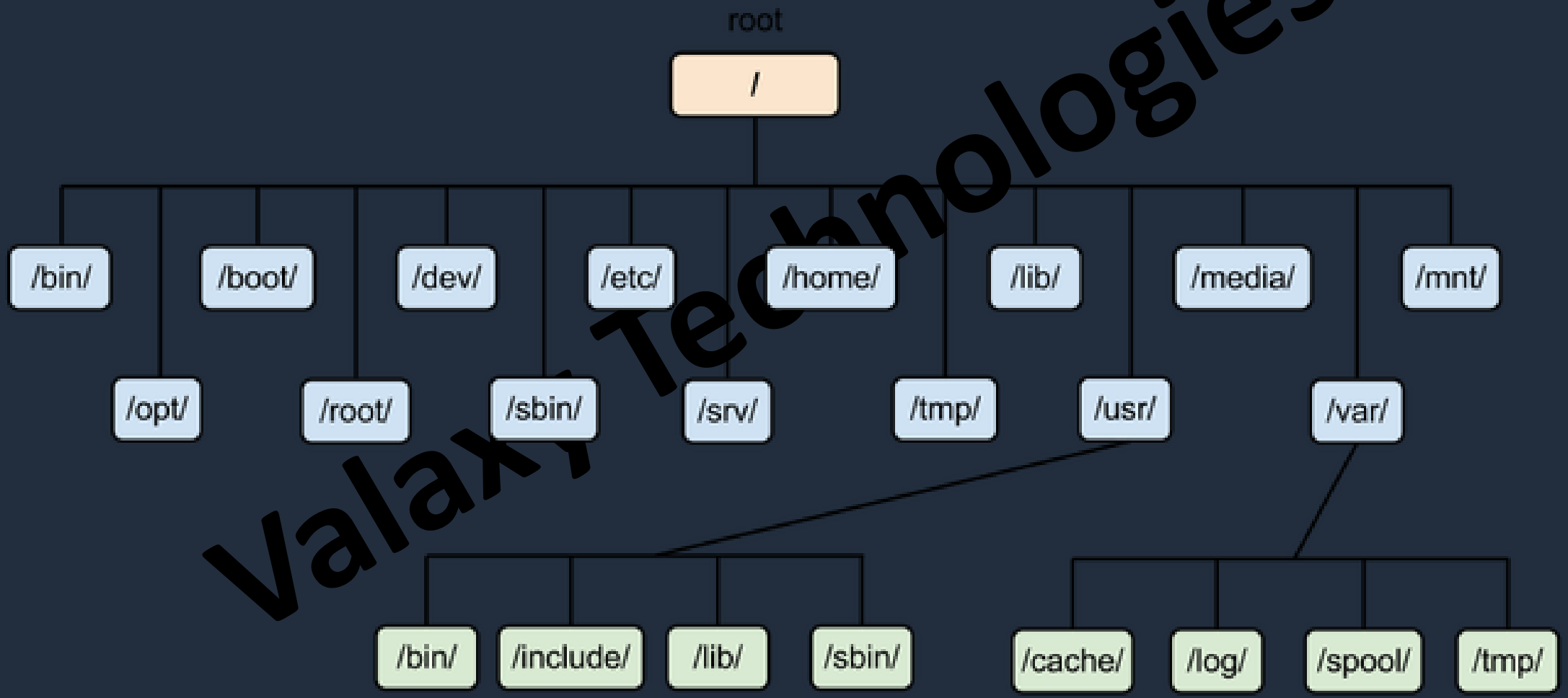
Why Linux

- Free
- Stability
- Secure
- Community Support

Linux Architecture



Linux Filesystem Hierarchy



Create An
AWS Free Tier Account

Creating an Linux EC2 Instance

Connect to a Linux EC2 Instance

Linux Filesystem Hierarchy

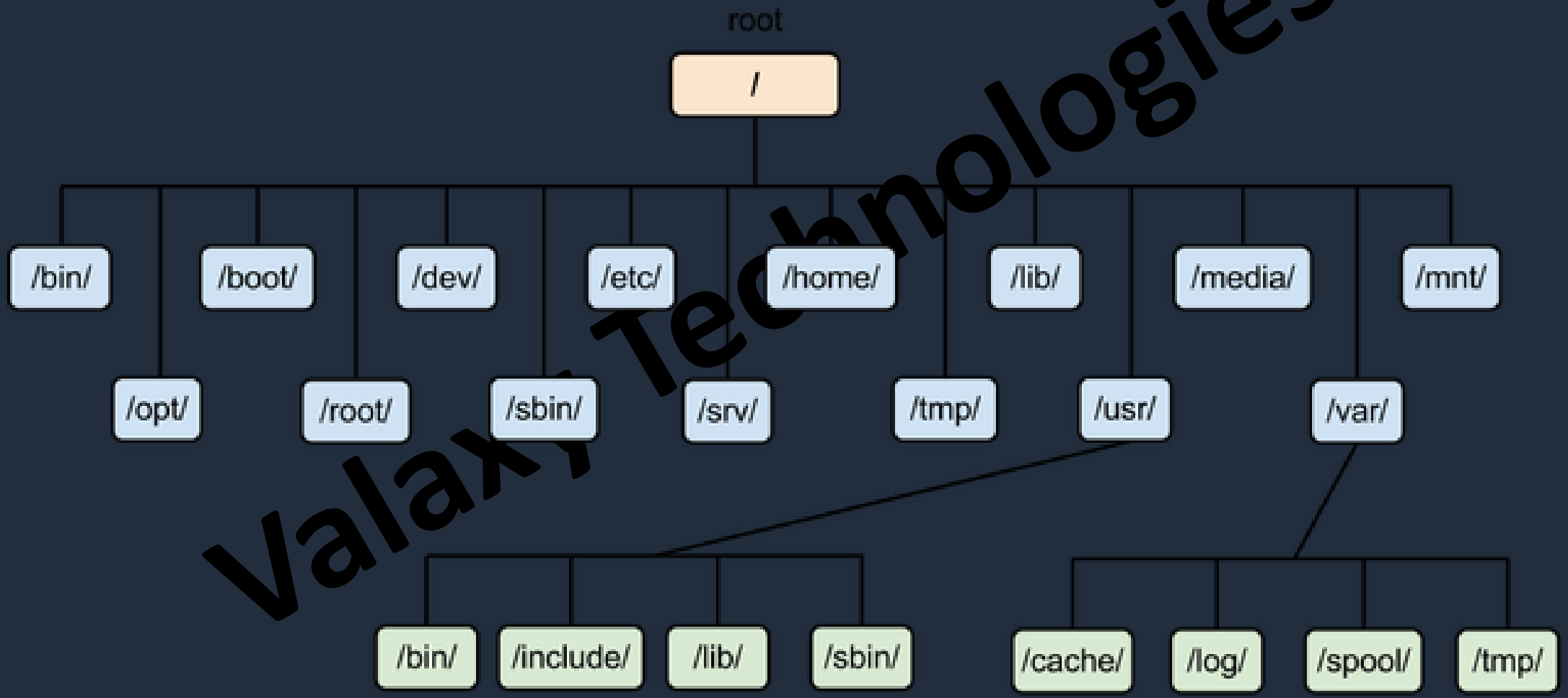
Linux Filesystem Hierarchy

Directory Name	Description
/	This is top level directory It is parent directory for all other directories It is called as ROOT directory It is represented by forward slash (/) C:\ of windows
/root	it is home directory for root user (super user) It provides working environment for root user C:\Documents and Settings\Administrator
/home	it is home directory for other users It provide working environment for other users (other than root) c:\Documents and Settings\username

Linux Filesystem Hierarchy

Directory Name	Description
/usr	by default softwares are installed in /usr directory (UNIX Sharable Resources) c:\program files
/bin	it contains commands used by all users (Binary files)
/sbin	it contains commands used by only Super User (root) (Super user's binary files)
/var	it is containing variable data like mails, log files

Linux Filesystem Hierarchy



Linux Basic commands

Command	Description
date	Show the current date and time
cal	Show this month's calendar
uptime	Show current uptime
whoami	who you are logged in as
finger	Display information about user
users / id	Shows user information
man command	Shows manual of command
username	Shows your user name
who / w	display who is online

View files

Command	Description
ls	directory listing
cat filename	view file content
less	view a file page by page
more	output the contents of *file*
head	output the first 10 lines of *file*
tail	output the last 10 lines of *file*
page	display file page by page

Create & Delete file/directory

Command	Description
touch	create a 0 bites file
cat > filename	create file and allow to write
nano	create a file if filename doesn’t exist
vi	create a file if filename doesn’t exist
rm	remove a file
mkdir	Create a directory
rmdir	Remove a empty directory
rm -rf	Remove a directory

Managing files or directories

Command	Description
cp	Copy a file
mv	Move a file
find	Find a file
grep	Search for a pattern in a file
cd	Switch between directories
diff	Find content difference in 2 files
sed	search and replace particular pattern
chmod	Change file permissions
chown	Change Ownership of a file
file	Show what kind of file it is

System Management

Command	Description
history	list all commands executed by a user
free	Free memory of a server
/proc/meminfo	Displays memory information
/proc/cpuinfo	Displays CPU information
uname -a	show kernel information
du	show directory space usage
whereis	show possible locations of *app*
which	show which *app* will be run by default

Networking

Command	Description
hostname	lists host name of the server
ping <ip>	availability of destination server over the network
wget	download packages/softwars onto Linux system
ifconfig	lists IP address(es) of the server
telnet	connect to remote host / check port availability status
curl	acces the application as from browser

Port Numbers

Port Number	Service
21	FTP
22	SSH
23	TELNET
25	SMTP
53	DNS
80	HTTP
443	HTTPS

Software Management

yum is the primary tool for getting, installing, deleting, querying, and managing RedHat Enterprise Linux RPM software packages from official RedHat software repositories, as well as other third-party repositories.

Commands:

```
yum install <package name>
yum remove <package name>
yum update <package name>
yum info <package name>
yum list available
yum list installed
```

Services

service - This controls the starting and stopping of services

chkconfig - This controls which services are set to start on boot

```
#service <name of the service> status --- To check the status of the service
#service <name of the service> start --- To start the service
#service <name of the service > stop --- To stop a service
#service <name of the service> reload --- To reload the service
#service <name of the service> restart --- To restart the service

#chkconfig --list --- To check the availability of service
#chkconfig <service> on --- To make the service available after restart
#chkconfig <service> off --- To make the service unavailable after restart
```

Process Management

- When you start a program or running an application in Linux, it actually run as a process
- A Linux process (a daemon), running in foreground or in the background, uses memory and CPU resources.

Command	Description
ps -ef	list the process which are running in the system
kill / kill -9	kill a process or service
fg	run the program in the foreground
bg	Run the service in the back group
top	List top 20 process which are consuming more CPU

Networking

IP Address: An IP address can be thought of as being similar to a phone number. Just as every person who communicates with a telephone is using a phone with a unique phone number, every computer that is on the Internet has a unique IP address. Not only on internet but within an organization every computer is assigned an IP address so that they can communicate with each other.

Command:

```
ifconfig -a
ip addr
```

Runlevels

Looks at the /etc/inittab file to decide the Linux run level.

Following are the available run levels

- 0 – halt
- 1 – Single user mode
- 2 – Multiuser, without NFS
- 3 – Full multiuser mode
- 4 – unused
- 5 – X11
- 6 – reboot

Archiving files or directories

Command	Description
gzip	Create a compressed file
gunzip	Unzip a file
tar	extract tar file

Crontab

In any operating system, it is possible to create jobs that you want to reoccur. This process known as ***job scheduling***, is usually done based on user-defined jobs. For RedHat or any other Linux, this process is handled by the cron service or a daemon called **crond**, which can be used to schedule tasks

Commands:

```
crontab -l
crontab -e
```

Field	Description	Allowed Value
MIN	Minute field	0 to 59
HOUR	Hour field	0 to 23
DOM	Day of the month	1-31
MON	Month field	1-12
DOW	Day of the week	0-6
CMD		any command

Crontab examples

Execute a job at 8:30 on everyday morning

```
30 8 * * * Command
```

Execute a job at 2:00 PM on every Saturday

```
00 14 * * 6 Command
```

Execute a job at 12:00 AM on 1st July

```
00 00 01 06 * Command
```

Execute a job at 3:30 PM on Every month 25th

```
30 15 25 * * Command
```

Copy file between servers

Windows to Linux

```
Mobaxterm or winscp
```

Linux to Linux

SCP (secure copy) is a command-line utility that allows you to securely copy files and directories between two systems.

```
scp source_file_name username@destination_host:destination_folder
```

Example: `scp file1 root@10.20.30.40:/tmp`

```
scp root@10.20.30.40:/tmp /home/ec2-user/
```

Link Files

There are 2 types of link files.

Soft link and Hard link

Soft link	Hard link
SHORTCUT FILE	BACKUP FILE
Size of link file is equal to no. of characters in the name of original file	Size of both file is same
if original file is deleted, link is broken and data is lost	If original file is deleted then also link will contain data
`Command: ln -s <src_file> <dest_file>`	`Command: ln <src_file> <dest_file>`

I/O Redirection

Redirection is a process where we can copy the output of any command(s), file(s) into a new file. There are two ways of redirecting the output into a file.

Using **>** or **>>** **filename** after the command, and

Examples:

```
cat file1 > file2
cat file1 >> file2
cat file1 file2 > file3
```

SSH

The SSH protocol (also referred to as Secure Shell) is a method for secure remote login from one computer to another. It provides several alternative options for strong authentication, and it protects the communications security and integrity with strong encryption.

Port Number	:	22
Daemon	:	sshd
Conf file	:	/etc/ssh/sshd_config

HTTP

Port Number	:	80
Daemon	:	httpd
Conf file	:	/etc/httpd/conf/httpd.conf

File Permissions

Access modes are different on file and directory

permission	Files	Directory
r	Open the file	'ls' the contents of dir
w	Write, edit, append, delete file	Add/Del/Rename contents of dir
x	To run a command/shell script	To enter into dir using 'cd'

File types

Symbol	Type of file
-	Normal file
b	Block file (Harddisk, Floppy disk)
c	Character file (Keyboard, Mouse)
d	Directory
l	link files (short cut)

Grep command

Grep stands for Global Regular Expression Print.

It is used to pick out the required expression from the file and print the output.

```
Syntax: grep <patron> filename
```

Sed command

stands for **stream editor**, which is used to search a word in the file and replace it with the word required to be in the output.

Note: it will only modify the output, but there will be no change in the original file.

Examples:

```
sed 's/old_text/new_text/' file_name
sed 's/old_text/new_text/g' file_name
sed -i 's/old_text/new_text/' file_name
sed -n '5,10p' file_name
sed '10,20d' file_name
```

find command

find command is used to find the files or directory's path, it is exactly like the find option in windows where you can search for a file.

```
Syntax: find / -option filename
```

Option	Usage
-name	For searching a file with its name
-user	For files whose owner is a particular user
-group	For files belonging to particular group

File Permissions

Permissions are applied at 3 levels

- Owner or User level
- Group level
- Others level

Permissions are applied in 3 ways

- r – Read only
- w – Write/Edit/Append/Delete
- x – Execute/Run

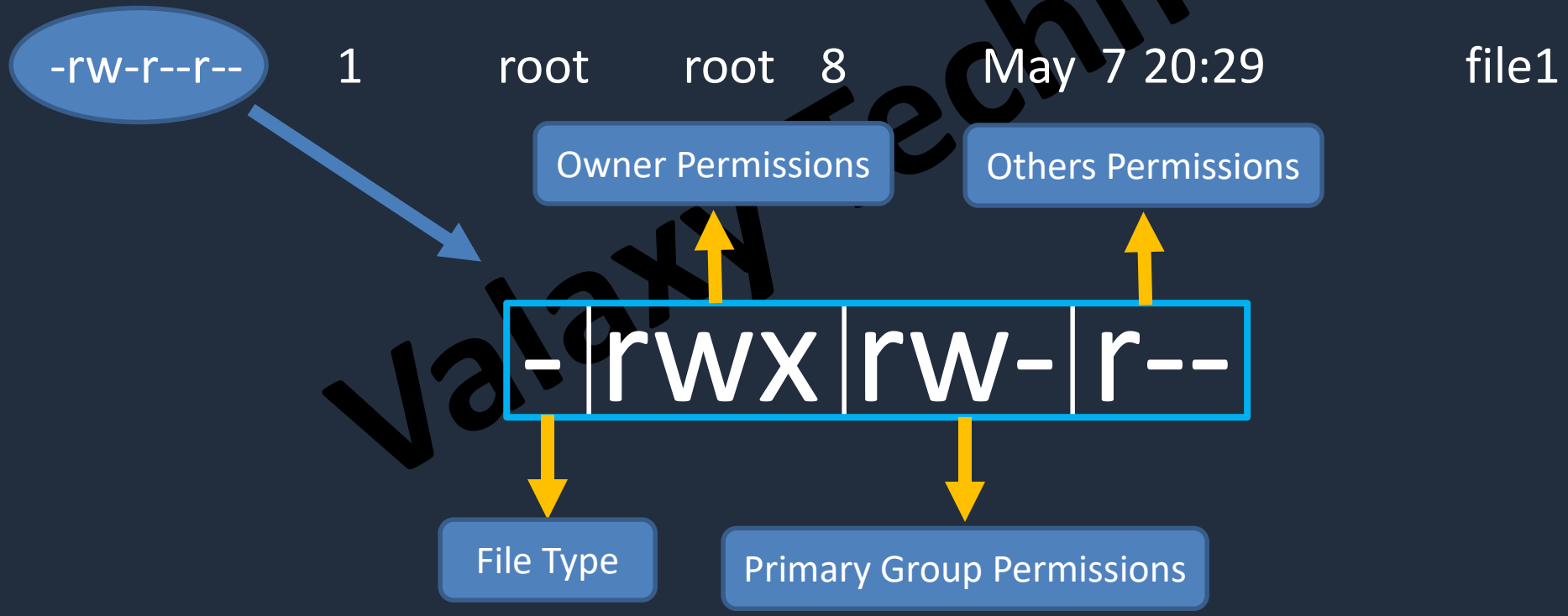
File Permissions

Access modes are different on file and directory

permission	Files	Directory
r	Open the file	'ls' the contents of dir
w	Write, edit, append, delete file	Add/Del/Rename contents of dir
x	To run a command/shell script	To enter into dir using 'cd'

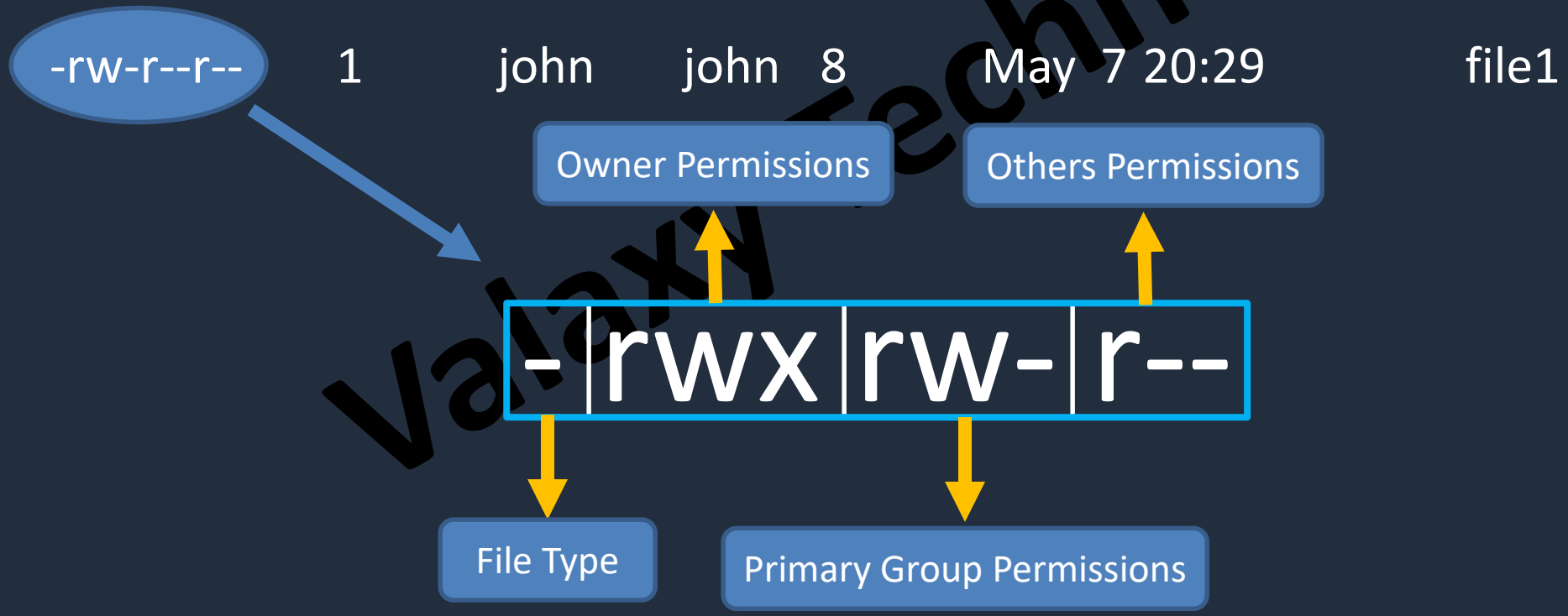
File Permissions

Unix/Linux files have 8 attributes that can be seen with **ls -l** command.

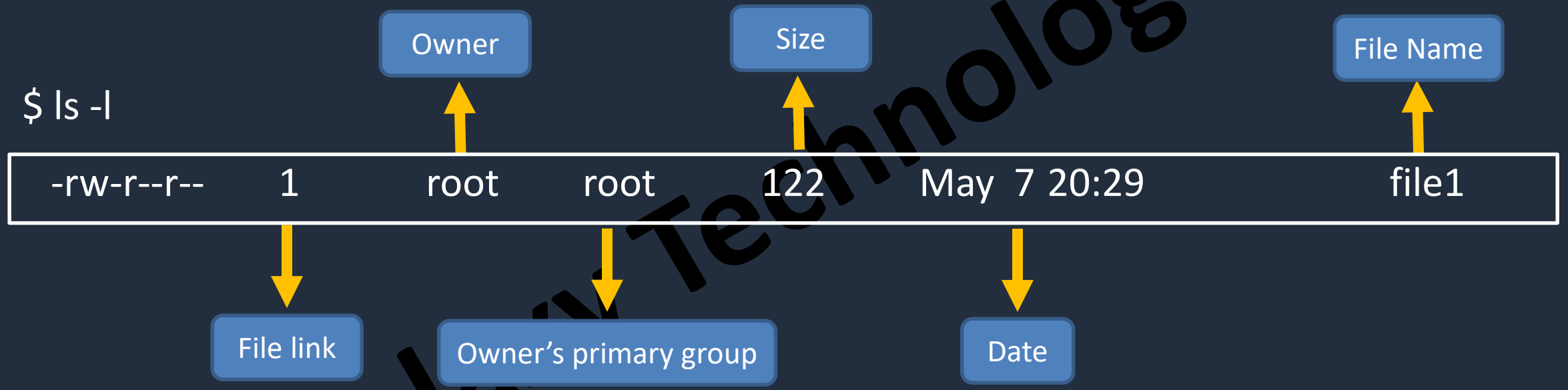


File Permissions

Unix/Linux files have 8 attributes that can be seen with **ls -l** command.



File Permissions



File Permissions

Permission can be set on any file/dir by using two methods:-

- Symbolic method (ugo)
- Absolute method (numbers)

Symbolic method	Absolute method
# chmod [who] [+/-/=] [permissions] file Who: To whom the permissions to be assigned Permissions: User/owner (u); group (g); others (o) Example: #chmod u=rwx,g=rw,o=r <file_name> #chmod ugo=rwx <file_name>	we use numbers instead of using symbols Read – 4 Write – 2 Execute – 1 chmod 764 <file_name> Chmod 777 <file_name>

User Management

- In Linux there are three types of users.
- 1. **Super or root user:** User is the most powerful user. He is the administrator user.
- 2. **System user:** Users created by the softwares or applications.
- 3. **Normal user:** Normal users are the users created by root user.

Type	Example	Home Directory	Shell
Super User	Root	/root	/bin/bash
System User	ftp, ssh, apache	/var/ftp, etc	/sbin/nologin
Normal user	visitor, ec2-user	/home/username	/bin/bash

User Creation

Whenever a user is created in Linux, below things happen by default.

- A home directory is created(/home/username)
- unique UID & GID are given to user
- An entry in /etc/passwd

User Creation

The syntax for creating a user in Linux is

```
# useradd <option> <username>
```

Options are:

- -u user id
- -G Secondary group id
- -g primary group id
- -d home directory
- -c comment
- -s shell

Example : during docker setup we should add user to a docker group.