**Server and Cron**

**dmidecode | less**: To get full Information of server

**fdisk -l | grep '^Disk /dev/' | grep -v loop**: To get Disk/Storage Info

**ls**: Display list of file & folder in current Directory

**clear** – clears the screen of the terminal.

**df -h:** To get Disk Space Information

**du**: Estimate file space usage

**free**: display memory usage

**uptime** – shows system uptime and load average

**cd xyz: C**hange Directory from current to xyz

**cd /:** Change/Back Directory from current to Previous

**cd .. :** Back the Previous Directory

**cd -:** Back the Previous Directory

**cd:** Back/Go To Home page

**pwd:** To get full path of current folder/file

**cp file\_to\_copy.txt new\_file.txt**: Make/create a copy file

**cp dir\_to\_copy new\_dir**: Make/create a copy Directory/Folder

**rm:** Remove File/Folder

**rf:** Remove forcefullyFile/Folder

**rm -rf \***: Remove forcefully All file/folder from current Directory

**rm -rf xyz-back-\***: Remove forcefully All file/folder from current Directory which start with xyz-back

**rm -rf xyz.log**: Remove forcefully file/folder from current Directory which name is like xyz.log

**du -h palmstorage 2>/dev/null | grep '[0-9\.]\+G'** : palmstorage(name of folder) like du -h engagementcode 2>/dev/null | grep ‘[0-9\.]\+G'

**du -a comviva | sort -n -r | head -n 20:** To get top 20 largest file in comvivafolder

**less comviva\_cron\_to\_trigger.log**: Read/View log file

**vim comviva\_XYZ\_publish.php**: View/Read the code of any file like **c**omviva\_XYZ\_publish.php. edit plain text files using efficient keybindings

**cat**: Display file contents.

**tac**: Output file contents, in reverse.

**more**: Display file contents one screen/page at a time

**less**: Similar to the more command with additional features.

**tail**: Used to display the tail end of a text file or piped data.

**:q enter:** Exit from viewer terminal

**exit**: Exit to home path from any where

**\xyz:** search text in file after open file through vim

**mkdir comviva\_backup:**  Create a Directory/Folder

**mv comviva/comviva\_cronr\_lms.log comviva\_backup**: To move the file(like comviva/comviva\_cronr\_lms.log) into another’s folder (like comviva\_backup)

**mv source\_file destination\_folder/**: Copy the file & pest it into Destination Folder

**top:** Get/View CPU Utilisation, shows an overall system view

**htop**: Get/View the CPU process utilisation

**pm2 list:** Get list of Active bot List. PM2 is a Node. js process manager that comes with a built-in load balancer

**pm2 stop/start {id}:** To start & stop the node base service on linux server

**crontab -l**: Get the list of cron/scheduler are running in server

**service crond status**: Check Cron is Running or Not

grep **master\_cronjob.php switch=recommended\_course** /var/log/cron: To check particular(master\_cronjob.php switch=recommended\_course) cron is running or not

**chkconfig crond on**

**service crond start:** Type both command to Run/Start the cron is not running

**tail -f /var/log/cron || cat /var/log/cron**: Verify cron is running by viewing log file, enter. || crond and cron jobs log file

**pgrep cron**: Find out if cron daemon is running or not, enter:

**kill 533494**: Kill The process by Process ID

**ping google.com**: Request a domain or IP address:

**history**: This command displays an enumerated list with the commands you’ve used in the past

**which:** The which command outputs the full path of shell commands(which php or which python)

**service nginx status:** To check the status of service status of nginx server. nginx.service - nginx - high performance web server

**service nginx stop:** To stop the service of nginx server

**service nginx start:** To start the service of nginx server

**status:** To check status of the service.

**start:** To Start the service.

**stop:** To stop the service.

**systemctl**: Listing Running Services Under SystemD in Linux

**systemctl --type=service**: To list all loaded services on your system (whether active; running, exited or failed, use the list-units subcommand and --type switch with a value of service

**systemctl --type=service --state=active**: list all loaded but active services, both running and those that have exited, you can add the --state option with a value of active, as follows.