

FAQ (Linux)

FAQ (Linux)

The information provided is mainly meant to address issues in Linux as it is the most commonly used server OS to run JET.

If you have issues in Windows Server OS, please feel free to contact us. We will update the information accordingly once we confirmed.



How to hide server default page or the root directory from the root of your website?

By default, the root of your website will show the server default page or the root directory. We would want to hide this from the normal users.

Step 1: Run the following commands:

```
$ sudo a2enmod rewrite
$ sudo systemctl restart apache2
```

Step 2: Edit the default config file:

If you wish to edit files from the graphical interface, add the following lines to the config file, which can be found at

</etc/apache2/sites-enabled/000-default.conf>

```
Redirect permanent / https://www.yourdomain.com/  
RedirectMatch ^/$ /JET/
```

```
RedirectMatch ^/$ /JET/
```

If you wish to edit files from the shell, follow the below steps:

Step 2.1: Open the default config file in vi

```
$ sudo vi /etc/apache2/sites-enabled/000-default.conf
```

Step 2.2: Add the below lines into this file

1. Press the key “i” on your keyboard to enter edit/insert mode on vi.

2. Insert the below text to the file:

```
Redirect permanent / https://www.yourdomain.com/  
RedirectMatch ^/$ /JET/
```

```
RedirectMatch ^/$ /JET/
```

Step 2.2: Save the file and quit vi

1. Press the “Esc” key on your keyboard to exit edit/insert mode on vi.
2. Type “:wq” to write the changes to the config file and quit.

[Back to Table of Contents](#)

How to ensure that the URL is not case sensitive?

For example, <https://www.url.com/TEST> should be the same as <https://www.url.com/tesT> and <https://www.url.com/test>.

Step 1: Edit the config file:

If you wish to edit files from the graphical interface, add the following lines to the config file, which can be found at

/etc/apache2/apache2.conf

```
LoadModule spelling_module /usr/lib/apache2/modules/mod_spelling.so
```

```
CheckSpelling On
```

```
CheckCaseOnly On
```

If you wish to edit files from the shell, follow the below steps:

Step 1.1: Open the default config file in vi

```
$ sudo vi /etc/apache2/apache2.conf
```

Step 1.2: Add the below lines into this file

1. Press the key “i” on your keyboard to enter edit/insert mode on vi.
2. Insert the below text to the file:

```
LoadModule speling_module /usr/lib/apache2/modules/mod_speling.so
```

```
CheckSpelling On
```

```
CheckCaseOnly On
```


Step 1.2: Save the file and quit vi

1. Press the “Esc” key on your keyboard to exit edit/insert mode on vi.
2. Type “:wq” to write the changes to the config file and quit.

Step 2: Restart the apache services

```
$ sudo service apache2 reload
```

[Back to Table of Contents](#)

How to setup auto OS updates in Linux?

Step 1: Create an account at <https://ubuntu.com>

Step 2: Copy and save the token from <https://ubuntu.com/pro/dashboard>

Step 3: Run the following commands on the server.

1. `sudo snap remove canonical-livepatch`
2. `sudo snap install canonical-livepatch`
3. `sudo apt update`
4. `sudo apt install ubuntu-advantage-tools`

5. `sudo ua attach`

Replace <token> with the token copied in Step 2.

After this is completed, your Linux server will automatically install essential updates in the background which will take effect the next time you restart the server.

[Back to Table of Contents](#)

How to enable crontab in linux?

What is crontab? Click here to find out.(https://www.tutorialspoint.com/unix_commands/crontab.htm)

Step 1: Open the crontab file to edit

```
$ sudo crontab -e
```

This will open the crontab file with the default editor vi

Step 2: Edit this file:

Step 2.1: Enter Edit/Insert mode on vi

1. Press the key “i” on your keyboard to enter edit/insert mode on vi.
2. Add the below line to the file:

```
* * * * * php -f /var/www/html/JET/scheduler.php
```

Step 2.2: Save the file and quit vi

1. Press the “Esc” key on your keyboard to exit edit/insert mode on vi.
2. Type “:wq” to write the changes to the crontab file and quit.

Step 3: Restart crontab

```
$ sudo /etc/init.d/cron restart
```

[Back to Table of Contents](#)

How do I make JET more secure in Linux?

Step 1: Open the hosts file in vi

[How to use vi?](#)

```
$ sudo vim /etc/hosts.allow
```

Step 2: Add the following lines to his file:

```
sshd:localhost:allow  
sshd:192.168.0.:allow  
sshd:111.111.111.111:allow  
sshd:ALL:deny
```

Replace 111.111.111.111 with your own public IP

You can add multiple lines of this to allow more IPs to have access.

Step 2: Restart sshd

```
$ systemctl restart sshd
```


Step 3: Create a new user

For example, we want to create a new user 'jetuser'

```
$ sudo adduser jetuser  
$ usermod -aG sudo jetuser
```

Step 3: Edit the sshd config file

```
$ sudo vim /etc/ssh/sshd_config
```

Update the following line to disable root login:

```
PermitRootLogin no
```

Step 3: Restart sshd

```
$ systemctl restart sshd
```

[Back to Table of Contents](#)

How do I fix the timeout issue when there is a lot of data?

This can occur if there is a lot of record in the form and you are exporting the form data.

Step 1: Edit the file

```
$ sudo vim /etc/php/7.2/apache2/php.ini
```

Locate the following:

```
    max_input_time = 60  
max_execution_time = 60  
default_socket_timeout = 60
```

Change the numbers from 60 to 3600. This will increase the time given for the timeout to occur.

Step 2: Restart apache

```
$ sudo systemctl restart apache2
```

[Back to Table of Contents](#)

My OS partition ran out of space! What do I do?

Step 1: Delete files in /var/log/journal

```
$ sudo journalctl --vacuum-time=3d
```

Step 2: Remove update files

```
$ sudo apt-get autoclean
```

Step 3: Remove any unused packages

```
$ sudo apt autoremove
```

[Back to Table of Contents](#)

My data disk is gone after I restart the server

This can happen if Auto mount was not setup properly or there was a change in config due to cloud hosting updates.

Step 1: Confirm that the data disk is not mounted.

```
$ sudo df -TH
```

Step 2: Check for unmounted disk

```
$ sudo lsblk
```


Step 3: Mount the partition with the disk partition you found in Step 2, for example: /dev/vdb1

```
$ sudo mount /dev/vdb1 /data
```

Step 4: Check for Partition UUID

```
$ sudo blkid /dev/vdb1
```

```
[root@ecs-test-0001 ~]# blkid /dev/vdb1  
/dev/vdb1: UUID="0b3040e2-1367-4abb-841d-ddb0b92693df"  
TYPE="ext4?"
```

Step 5: Update fstab file

```
$ sudo vim /etc/fstab
```

Enter the below line at the end of the file.

```
UUID=0b3040e2-1367-4abb-841d-ddb0b92693df /data ext4  
defaults 0 2
```

[Back to Table of Contents](#)

How do I update php/apache

You will need to do this to keep the system up to date and be protected from vulnerabilities

In this example, php 7.4 branch is the latest php version.

Step 1: Add PPA

```
$ sudo apt install software-properties-common
```

```
$ sudo add-apt-repository ppa:ondrej/php  
$ sudo add-apt-repository ppa:ondrej/apache2  
$ sudo apt update
```

Step 2: Install latest php/apache

```
    $ sudo apt install php7.4  
$ sudo apt install apache2
```

Step 3: Install php Extensions using the following syntax

```
$ sudo apt install php7.4-extension_name
```

Step 4: There are some common php extensions that we can install.

```
$ sudo apt install php7.4-common php7.4-mysql php7.4-xml php7.4-xm  
lrpc php7.4-curl php7.4-gd php7.4-imagick php7.4-cli php7.4-dev php7.  
4-imap php7.4-mbstring php7.4-opcache php7.4-soap php7.4-zip php7.4-in  
tl -y
```

Step 5: Restart apache and check the versions to confirm

```
$ sudo systemctl restart apache2  
$ sudo php -v  
$ sudo apache2 -v
```

Take note that if you have changed some config files, you will have to check through all of those files to ensure that you still retain your customized config.

[Back to Table of Contents](#)

Apache/PHP version hiding?

When remote requests are sent to your Apache web server, by default, various information such as the php version, is sent along in server-generated documents back to the client.

The following steps would help to hide these information to protect your web server.

Apache:

Step 1: Run the following commands depending on which OS your server is using.

For Debian/Ubuntu systems:

```
$ sudo vi /etc/apache2/apache2.conf
```

For RHEL/CentOS systems

```
$ sudo vi /etc/httpd/conf/httpd.conf
```

Step 2: Add/modify the following lines.


```
ServerTokens Prod  
ServerSignature Off
```

Step 3: Restart the Apache service

```
$ sudo systemctl restart apache2
```

PHP:

Step 1: Edit php.ini

```
expose_php = off
```

[Back to Table of Contents](#)

How to migrate MYSQL database?

MySQL database is normally located in the System Disk of the Linux partition.

As you use JET, the data will grow and it may exceed

the limits of the System Disk partition.

As such, you should move the MySQL database to the data disk.

The below steps is based on Ubuntu Linux servers but it should be more or less the same in other Linux servers as well.

Do adjust for any differences and if you are not sure, you can contact the Cloud provider for further assistance.

Note: It is advisable to do a backup of the database before proceeding with the below. (SQL file backups are recommended)

Step 1: Verify the directory of MySQL database.

```
$ sudo mysql -u root -p
```

Enter the MySQL password.

```
mysql> select @@datadir;
```

The output will determine the MySQL database directory which is usually [/var/lib/mysql/](#)

Step 2: Stop MySQL service

```
$ sudo systemctl stop mysql
```

This will stop the service.

```
$ sudo systemctl status mysql
```

This will allow you to confirm the status of the service.

Step 3: Copy MySQL database to another location

```
$ sudo rsync -av /var/lib/mysql /data/database
```

In this case, /data/database folder is where I want to copy to.

Step 4: Rename the old database folder

```
$ sudo mv /var/lib/mysql /var/lib/mysql.bak
```

This will save the backup temporarily until we confirm that the MySQL migration is successful.

Step 5: Point the MySQL service to the new directory.

```
$ sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
```

Edit this file using vi.

```
$ datadir=/data/database/mysql
```

Locate the “datadir=” line and update it with the new path.

Step 6: Configure AppArmor Access Control Rules

```
$ sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
```


Edit this file using vi.

```
alias /var/lib/mysql/ -> /data/database/mysql/
```

Add this line and save.

```
$ sudo systemctl restart apparmor
```

Restart AppArmor service.

Step 7: Restart MySQL service

```
$ sudo mkdir /var/lib/mysql/mysql -p
```

Create a dummy folder in place of the default MySQL directory.

```
$ sudo systemctl start mysql
```

Start MySQL service.

```
$ sudo systemctl status mysql
```

Confirm that the service has started.

Step 8: Step 7 may sometimes fail if you took the chance to upgrade MySQL while the service is down.

In this case, you will need to re-initialize your database with this command:

```
mysql_install_db
```

before starting from step 3 again.

Step 9: Verify the directory of MySQL database again.

```
$ sudo mysql -u root -p
```

Enter the MySQL password.

```
mysql> select @@datadir;
```

The output will determine the MySQL database directory which is /data/database/

Step 10: Cleanup of temporary files/folders.

Do this after verifying the data integrity of existing data.

```
$ sudo rm -Rf /var/lib/mysql.bak
```

Remove the old MySQL folder we previously rename.

```
$ sudo systemctl restart mysql
```

Restart the service once more to confirm.

[Back to Table of Contents](#)

How to install SSL certificate in Linux

Enable SSL in Apache

```
$ sudo a2enmod ssl
```

```
user@server:/etc/apache2$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Module socache_shmcb already enabled
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and
create self-signed certificates.
To activate the new configuration, you need to run:
  service apache2 restart
```

Update config file

Step 1: Navigate to `/etc/apache2/sites-enabled/`

Step 2: Edit the file “000-default.conf”

Step 3: Comment out all existing lines inside with a ‘#’

Replace the filepaths shown above with the appropriate path that is used for your linux server.

Step 4: Add the below to the file.

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combine
SSLEngine on
SSLCertificateFile /etc/ssl/certificate.crt
SSLCertificateKeyFile /etc/ssl/certificate.key
SSLCertificateChainFile /etc/ssl/chain.crt
```

Step 5: Replace the filepaths shown above with the appropriate path that is used for your linux server.

```
ServerTokens Prod
ServerSignature Off
```

Restart the apache2 service

```
$ systemctl restart apache2
```

Note: The config file mentioned above may differ in different Linux servers.

[Back to Table of Contents](#)

Update Ciphersuite

Open the config file for apache2

```
sudo vim /etc/apache2/sites-enabled/000-default.conf
```

Insert the following within the VirtualHost 443 section

```
SSLCipherSuite          ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES
```

```
256-GCM-SHA384: ECDHE-ECDSA-AES128-SHA: ECDHE-ECDSA-AES256-SHA: ECDHE-ECDSA-AES128-SHA256: ECDHE-ECDSA-AES256-SHA384: ECDHE-RSA-AES128-GCM-SHA256: ECDHE-RSA-AES256-GCM-SHA384; DHE-RSA-AES128-GCM-SHA256: DHE-RSA-AES256-GCM-SHA384
SSLProxyCipherSuite          ECDHE-ECDSA-AES128-GCM-SHA256: ECDHE-ECDSA-AES256-GCM-SHA384: ECDHE-ECDSA-AES128-SHA: ECDHE-ECDSA-AES256-SHA: ECDHE-ECDSA-AES128-SHA256: ECDHE-ECDSA-AES256-SHA384: ECDHE-RSA-AES128-GCM-SHA256: ECDHE-RSA-AES256-GCM-SHA384; DHE-RSA-AES128-GCM-SHA256: DHE-RSA-AES256-GCM-SHA384
SSLProtocol                  all -SSLv3 -TLSv1 -TLSv1.1
SSLProxyProtocol             all -SSLv3 -TLSv1 -TLSv1.1
```

Restart the apache2 service

```
$ systemctl restart apache2
```

[Back to Table of Contents](#)

Php.ini config

Open the php.ini file

```
sudo vim /etc/php/*.*/apache2/php.ini
```

Take note that *.* is to be replaced with the php version that the server is currently installed with.

It can be 7.4, 8.0 or 8.1 based on the latest versions of php at this moment.

Update existing lines with the following

```
1. expose_php = Off
2. max_execution_time = 3600
3. max_input_time = 3600
4. post_max_size = 2000M
5. upload_max_filesize = 2000M
6. default_socket_timeout = 3600
7. session.cache_expire = 3600
8. date.timezone = Asia/Singapore
9. session.cookie_httponly = 1
10. session.use_only_cookies = 1
11. session.cookie_secure = 1
```

Note that 2000M and 3600 can be replaced with other values suitable for your server.

These values are picked to provide a longer timeout and larger file upload size.

Restart the apache2 service

```
$ systemctl restart apache2
```

[Back to Table of Contents](#)

Transfer of files to Linux

Install pscp in windows.

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

We need to install either Putty or just PSCP from the above link.

Once that is done, we can access the 'pscp' command from cmd.

PSCP command

```
pscp file.txt root@192.168.42.1:/var/www/html
```

- file.txt – the file that was to be transferred to linux
- root – the username of the account to use to transfer to linux
- 192.168.42.1 – the IP address of the linux machine (to be replaced with your own IP)
- /var/www/html – the file directory where you want to transfer your file to.

After entering the above command, you will be prompted to enter the password.

[Back to Table of Contents](#)

Issue that you may encounter on OS major version upgrade

OS upgrade is an overall upgrade of not just the OS but also the various software/services.

MySQL

If you have a second disk drive and had followed our guide to redirect MySQL to that location, you will encounter an issue during OS upgrade.

The OS upgrade will update your MySQL version and it will try to read into the default MySQL directory. It will then generate an error alert when it could not find it.

All we need to do in this case is to restore back the config so that it points to the default directory and we can setup for the second disk drive after the update is completed.

```
$ vim /etc/mysql/mysql.conf.d/mysqld.cnf
```

Edit the file above.

Ensure that the line containing `datadir` follows the below.

```
datadir = /var/lib/mysql
```

Apache2

During OS upgrade, it will prompt us if we want to replace the edited apache config files with the one from the new version.

As it is a version update, we would want to install the config files of the new version so that we do not miss out anything that may be added due to the new version.

We will then need to update the new config files with our required config.

Php

OS upgrade will always install the latest php with regards to the OS version you are updating to.

JET Workflow currently only works with php version 7.4

So there is a need to change the default running php version to 7.4

```
$ sudo add-apt-repository -y ppa:ondrej/php
$ sudo apt update
$ sudo apt install php7.4

$ sudo a2dismod php8.1
$ systemctl restart apache2

$ sudo a2enmod php7.4
$ sudo update-alternatives --set php /usr/bin/php7.4
$ sudo update-alternatives --set phar /usr/bin/phar7.4
$ sudo systemctl restart apache2
```

In the above, we assume that php 8.1 was installed due to OS upgrade.

If you have other php version installed, you can replace 8.1 with other version numbers.

[Back to Table of Contents](#)

Binlog files in MySQL taking up space

Sometimes there would be space taken up by MySQL log files.

To get rid of them, access the file below.

```
$ vim /etc/mysql/mysql.conf.d/mysqld.cnf
```

Add the below line to the end of the file.

```
binlog_expire_logs_seconds=3600
```

Restart MySQL service

```
$ Systemctl restart MySQL
```

[Back to Table of Contents](#)