**Asset Management Application Full installation and configuration 0n a Ubuntu server:**

**1. Install Ubuntu Server 20.04**

You can do this either on a physical server or VM Server.

Open SSH or Putty and first update and upgrade the packages by running below command.

**Sudo apt update -y && sudo apt upgrade -y**

**2. Install LAMP on Ubuntu**

You need to make Ubuntu a Web server. In order to install the lamp, we will use the below steps.

* Install Apache2
* Install apache2 by using below command

**sudo apt install apache2 -y**

* Now Enable and Start apache2 service

**systemctl start apache2 && systemctl enable apache2**

* Now check the status of the service

**systemctl status apache2**

* Let us now check the version of apache2 to validate

**apache2 -v**

great if you see the version information you are done.

In case the firewall is enabled you need to open HTTP and HTTPS ports to this server. just run the below commands for the firewall

**sudo ufw allow 80/tcp**

**sudo ufw allow 443/tcp**

**sudo ufw reload**

Now you can open the browser and validate that the webserver is ready with apache2.

http://<serverip> in my case it is <http://localhost:80>

Snipe-IT needs various extensions, let us enable one which is mandatory

**sudo a2enmod rewrite**

**systemctl restart apache2**

Restart apache2 and move to the next step.

**Now let us install other packages**

**MySql / MariaDB**

This is the Database Server Installation. You need to first install the database Server and Client both will be installed using below command.

**sudo apt install mariadb-server mariadb-client -y**

Now let us follow the same step as mentioned in apache2. Enable and Start the service and then check the version and status of the service

**sudo systemctl start mariadb && sudo systemctl enable mariadb**

**sudo systemctl status mariadb**

Before we proceed further let us now secure the installation by running the below command

**sudo mysql\_secure\_installation**

You will need to enter when the password is promoted as there is no password. Then you have to set the root password. disallow remote and anonymous access and remove the test database.

**PHP Installation**

Installing PHP and its related extensions, i will install php7.4 as I have already tested this installation for Snipe-IT and it works fine

**sudo apt install php php-bcmath php-bz2 php-intl php-gd php-mbstring php-mysql php-zip php-opcache php-pdo php-calendar php-ctype php-exif php-ffi php-fileinfo php-ftp php-iconv php-intl php-json php-mysqli php-phar php-posix php-readline php-shmop php-sockets php-sysvmsg php-sysvsem php-sysvshm php-tokenizer php-curl php-ldap -y**

**Snipe-IT needs PHP Composer**

Snipe-IT Depends on some of the libraries that is why we need to install PHP Composer. Now follow the below steps and install PHP composer. The below command will download the PHP composer in your current directory.

**sudo curl -sS https://getcomposer.org/installer | php**

Now move the composer installer to the recommended directory

**sudo mv composer.phar /usr/local/bin/composer**

Now don’t do anything with the composer after the above command. We will use composer during installation of Snipe-IT.

**Create Database in MariaDB for Snipe-IT**

Before installing the snipe-IT Package we make sure our platform is ready. Therefore, let us create the database with the required privileges. Open MySQL using the root user

**mysql -u root -p**

The below command will create the DB, user set the password, and assign them privileges. You just need to modify the details by pasting in notepad first otherwise you can continue with what I mentioned below script. You can copy all or use commands one by one. I recommend using it one by one to ensure it is error-free.

**CREATE DATABASE snipeitdb;**

**CREATE USER snipeituser@localhost IDENTIFIED BY 'Password';**

**GRANT ALL PRIVILEGES ON snipeitdb.\* TO snipeituser@localhost;**

**FLUSH PRIVILEGES;**

**EXIT;**

**Installation of Snipe-IT**

Now system is ready for the installation of SnipeIT. We will first move to the Website Root Directory.

**cd /var/www/**

The latest repository of Snipe-IT can be cloned from GIT. Use the below command which will copy from the git and will make a folder in the www directory. You can change the folder name if you wish. The folder name of snipe-it is at end of the command.

**git clone https://github.com/snipe/snipe-it snipe-it**

Once clone is compelted it will copy entire applicaiotn into your snipe it folder.

Now let us open the folder.

**cd /var/www/snipe-it**

We will be working on this directory now to make few changes. The main configuration file is .env which is stored as example, we will make a copy of that as below;

**sudo cp /var/www/snipe-it/.env.example /var/www/snipe-it/.env**

Now ENV is the environment file which is created. In this configuration file we will need to mainly look at the Application and the Database connection details. You can use nano command to edit your ENV file.

**sudo nano /var/www/snipe-it/.env**

Change mainly the following;

**APP\_DEBUG=false**

**APP\_URL=snipe-it.syncbricks.com  
APP\_TIMEZONE=’Asia/Muscat’**

**DB\_DATABASE=snipeitdb  
DB\_USERNAME=snipeituser  
DB\_PASSWORD=Password**

In case you want to fix issues initially after installation, I will recommend changing the following as well

**APP\_DEBUG=true** it is the second row in the configuration file. Now you need to set permission to the directories within the Snipe-IT directories use below commands.

**sudo chown -R www-data:www-data /var/www/snipe-it**

**sudo chmod -R 755 /var/www/snipe-it**

We downloaded the composer installer initially, now using the same folder /var/www/snipe-it. We will install the composer using the below commands.

**sudo composer update --no-plugins --no-scripts**

**sudo composer install --no-dev --prefer-source --no-plugins --no-scripts**

Now we need to generate the APPKEY which will automatically update the.env file with APP\_KEY use below command while remaining in the same directory.

**php artisan key:generate**

Update Virtual Host. Finally you need to now set up the default web directory to snipe-it by updating the virtual host let us see how to do it.

**Disable Default virtual Host**

First of all, let us disable the default virtual host in our web server

**sudo a2dissite 000-default.conf**

**Create and configure Snipe-IT virtual Host**

Let us now create a Snipe-IT virtual host configuration

**sudo nano /etc/apache2/sites-available/snipe-it.conf**

use below line to create this configuration and modify to meet your requirements.

**<VirtualHost \*:80>  
ServerName localhost  
DocumentRoot /var/www/snipe-it/public  
<Directory /var/www/snipe-it/public>  
Options Indexes FollowSymLinks MultiViews  
AllowOverride All  
Order allow,deny  
allow from all  
</Directory>  
</VirtualHost>**

Enable New Configuration as virtual host

Enable Snipe-it configuration

**a2ensite snipe-it.conf**

**Folder Permissions for Upload and Storage**

Before you proceed to restart the apache server. just make sure to update the folder permission as follows;

**sudo chown -R www-data:www-data ./storage**

**sudo chmod -R 755 ./storage**

Restart Apache Server

**systemctl restart apache2**

Done! Now you can open the browser and run the setup wizard.