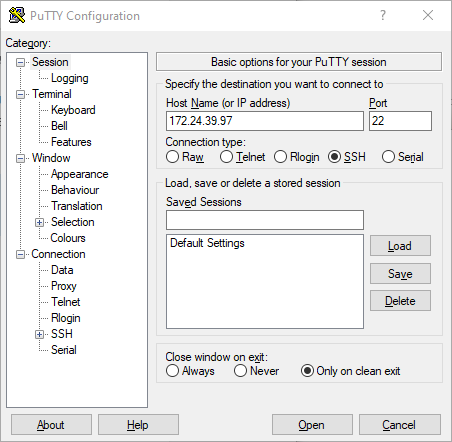
Source https://blog.masteringmdm.com/snipe-it-installation-on-ubuntu-server-nginx/

**1. Connect to Ubuntu server**

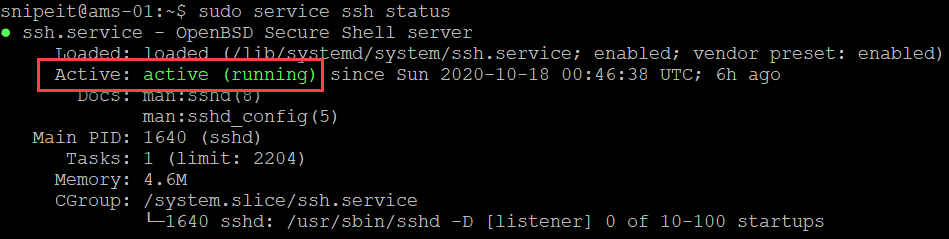
Highly recommend to use a SSH client for connectivity and it will make easy to copy/paste commands as compare to directly interacting with your Ubuntu server console.  
I will be using Putty as SSH client. Here is the direct link to [download Putty](https://www.putty.org/) or you could use your preferred SSH client.



**OpenSSH server status/installation**

* OpenSSH is already installed. You can check SSH server status

sudo service ssh status



* If Open SSH server is not installed, you can install it by using the following commands:

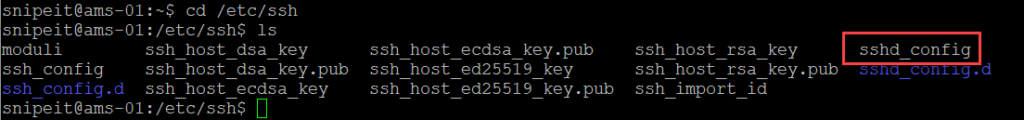
sudo apt install OpenSSH-Server

* Verify Open SSH server status

sudo service ssh status

* Open SSH server’s configuration file is located here, file name is sshd\_config. Open SSH should work out of the box, however, you can edit it if needed with your preferred choice of editor

cd /etc/ssh



**2. Update/Upgrade installed packages**

Update and upgrade installed packages on Ubuntu server. You will need to run this command as sudo and enter ‘y’ when prompted to confirm.

sudo apt update && sudo apt upgrade



**3. Install NGINX, MySQL, PHP and PHP FPM**

1. Install NGINX

sudo apt install nginx

1. Install MySQL

sudo apt install mysql-server

1. Install PHP and PHP-FPM

sudo add-apt-repository universe

sudo apt install php-fpm php-mysql

**Check versions**

* MySQL server version

sudo mysql -V

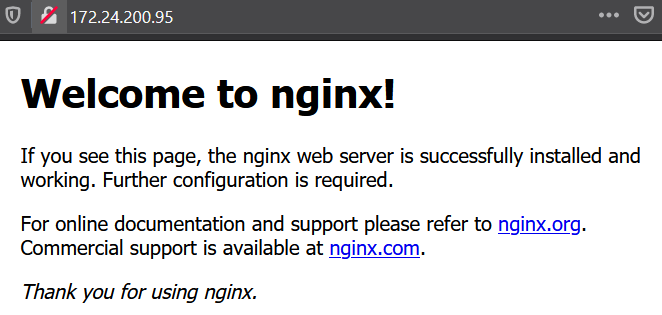
* NGINX version

sudo nginx -V

* PHP Version

sudo php --version

**Test Installation**

* NGINX: Browse to IP address of your server and you should see NGINX generic web page  
  
* MySQL: Run the following command to see the status of MySQL Server

sudo service mysql status

**4. Create a new site and test PHP**

In this step, we will create a new site for Snipe-IT, disable default site and test php. Also, we will restart the relevant services. We will come back to this later and update the root folder once SNIPE-IT is downloaded.

* Create new site with name ams.teknex.com.au, you can update this name if you like to something else

sudo nano /etc/nginx/sites-available/ams.teknex.com.au

* Add site content as follows, update Directory if you download Snipe-IT in a different folder and Server Name to your Ubuntu Server’s IP address

server **{**

listen 80;

listen **[**::**]**:80;

server\_name 172.24.200.95;

root /var/www/html;

index index.php index.html index.htm;

access\_log /var/log/nginx/ams.teknex.com.au.access.log;

error\_log /var/log/nginx/ams.teknex.com.au.error.log;

location / **{**

try\_files $uri $uri/ /index.php$is\_args$args;

**}**

location ~ \.php$ **{**

include snippets/fastcgi-php.conf;

fastcgi\_pass unix:/var/run/php/php7.4-fpm.sock;

include fastcgi\_params;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

fastcgi\_intercept\_errors on;

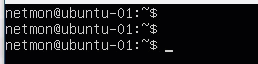
**}**

**}**

|  |  |  |
| --- | --- | --- |
| Line number | Config | Update |
| 5 | server\_name | Enter your server’s IP Address/Host Name |
| 9 | access\_log | Update name of your website |
| 10 | error\_log | Update name of your website |
| 20 | location ~ \.php$ | Update PHP-FPM version in fastcgi\_pass |

Easiest way to check PHP-FPM version is by typing the following in the terminal and hit tab. This should populate the value as shown below. For instance if your PHP-FPM version is 7.4 then enter the value as php7.4-fpm.

php-fpm



* Enable new site

sudo ln -s /etc/nginx/sites-available/ams.teknex.com.au /etc/nginx/sites-enabled/

* Disable default site

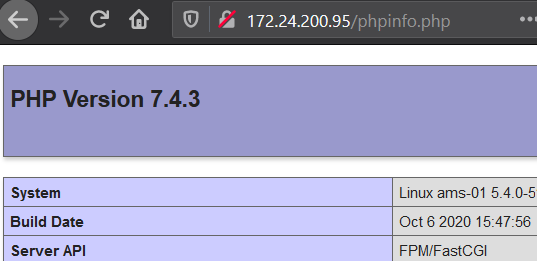
sudo unlink /etc/nginx/sites-enabled/default

* Restart Nginx

sudo service nginx restart

* PHP: Create a simple PHP page and browse to IP\_Address/phpinfo.php

sudo bash -c "echo -e '<?php\nphpinfo();\n?>' > /var/www/html/phpinfo.php"



***Do not proceed further if PHP is not working***

**5. Download Snipe-IT**

Clone recent version of Snipe-IT from GIT into a local folder **snipeit.**

* Working directory is /var/www

cd /var/www/

* Clone Snipe-IT to local folder snipeit

sudo git clone https://github.com/NaveenSutar/comtech-asset-mgmt-tool.git snipeit

* Make a copy of .env environmental file

cd /var/www/snipeit

sudo cp .env.example .env

* If you like, you can have a quick look inside the .env file with nano editor. We will come back to this file later on. Don’t make any changes to the file and use **Ctrl + X** to exit

sudo nano .env

**6. Configure MySQL Server**

Run MySQL secure installation command line wizard and secure the installation.

sudo mysql\_secure\_installation

* **yes or no**, if you would like to install Validate Password Plugin (**I selected no** for this in my lab environment)
* The next part offers you to change the password for the root of MySQL and enter password twice
* The next option to remove anonymous users, I will choose **Yes**
* The next option to stop root login remotely, I would go for **no**
* Reload privileges table: **Yes**
* You will see, “All done”, once it is all done

**7. Create an initial database and a user**

In this step, we will create database, user and grant specific privileges to new user.

|  |  |
| --- | --- |
| Database | snipeit\_db |
| User | snipeit\_user |
| Password | password |

* Login to MySQL server, provide password for MySQL root user when it prompts

sudo mysql -u root -p

* Create new database **snipeit\_db**

CREATE DATABASE snipeit\_db;

* Create new user **snipeit\_user**

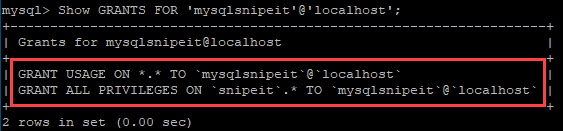
CREATE USER 'snipeit\_user'@'localhost' IDENTIFIED BY 'password';

* Grant permissions to **sipeit\_user** on database **snipeit\_db**

GRANT ALL PRIVILEGES ON snipeit\_db.\* TO 'snipeit\_user'@'localhost';

* Verify permissions

Show GRANTS FOR 'snipeit\_user'@'localhost';



* Flush privileges

flush privileges;

* Quit

quit;

* Restart MySQL service

sudo service mysql restart

* Check MySQL service status

sudo service mysql status

**8. Update .env file**

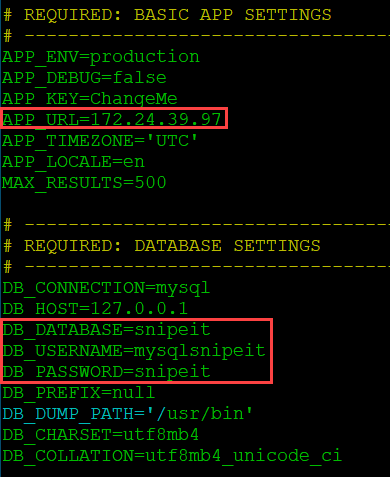
Let’s update .env file with the required information.

cd /var/www/snipeit

sudo nano .env

Update the following:

* App\_URL
* DB\_DATABASE
* DB\_USERNAME
* DB\_PASSWORD



**9. Update folder permissions**

Update folder permissions in the Snipe-IT root directory

* Working directory is /var/www/snipeit

cd /var/www/snipeit

* Update permissions

sudo chown -R www-data:www-data /var/www/snipeit/

sudo chmod -R 755 storage

sudo chmod -R 755 public/uploads

**10. Download Composer**

**Working directory is logged in user’s home directory. In my case it is /home/snipeit**

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

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

**11. Install PHP dependencies**

There are a number of PHP dependencies, ensure these all are successfully installed.   
Working directory: Home directory of logged in user

sudo add-apt-repository universe

sudo apt-get install -y git unzip php php-curl php-mysql php-gd php-ldap php-zip php-mbstring php-xml php-bcmath php-tokenizer

**12. Install Composer**

Working directory: /var/www/snipeit

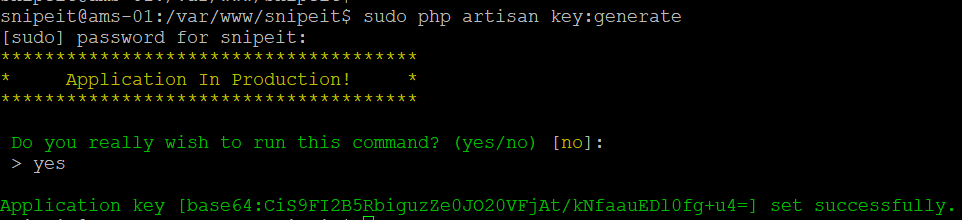
cd /var/www/snipeit

sudo composer install --no-dev --prefer-source

**13. Generate App Key for .env file**

**Run the following command and enter yes when prompted**

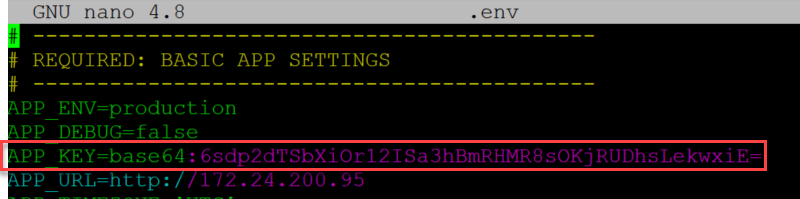
sudo php artisan key:generate

****

**Verify the key generation**

Open .env file and check APP\_KEY value

sudo nano .env

****

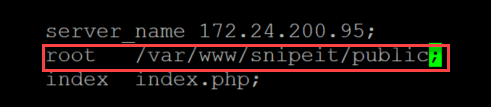
**14. Update Snipe-IT site**

We created a site earlier and in this step we will update the root folder.

sudo nano /etc/nginx/sites-available/ams.teknex.com.au

Update the root to the following:

/var/www/snipeit/public



Restart the web server

sudo service nginx restart

**15. Snipe-IT pre-flight setup**

Browse to IP address of your Ubuntu server and you should see the pre-flight page. Follow the prompts and you are done!!