

# Snipe-IT setup in almalinux 9

---

## Snipe-IT Installation Guide on AlmaLinux 9 (PHP 8.3 + Nginx + MariaDB)

---

This guide covers installing **Snipe-IT** on **AlmaLinux 9** with **PHP 8.3**, **Nginx**, and **MariaDB**, including **EPEL installation** for additional packages.

---

### Step 1: Install EPEL Repository

EPEL (Extra Packages for Enterprise Linux) provides additional packages not available in the default repo.

```
sudo dnf install epel-release -y
sudo dnf update -y
```

---

### Step 2: Install Required Repositories for PHP 8.3

Add **Remi repository** for PHP 8.3:

```
sudo dnf install -y https://rpms.remirepo.net/enterprise/remi-release-9.rpm
```

Enable **Remi PHP 8.3** module:

```
sudo dnf module reset php -y
sudo dnf module enable php:remi-8.3 -y
```

---

### Step 3: Install PHP 8.3 and Required Extensions

```
sudo dnf install -y php php-fpm php-mysqlnd php-curl php-xml php-zip php-mbstring php-gd php-cli php-soap php-intl php-ldap php-bcmath unzip git curl
```

---

### Step 4: Install Composer

Download and install **Composer**:

```
wget -O composer-setup.php https://getcomposer.org/installer
php composer-setup.php --install-dir=/usr/local/bin --filename=composer
composer -V # Verify installation
```

---

### Step 5: Install and Configure MariaDB

## 5.1 Install MariaDB

```
sudo dnf install mariadb-server -y
sudo systemctl start mariadb
sudo systemctl enable mariadb
```

## 5.2 Secure MySQL Installation

```
sudo mysql_secure_installation
```

- Set root password
- Remove anonymous users
- Disable remote root login
- Remove test database
- Reload privilege tables

## 5.3 Create Snipe-IT Database and User

Log in to MySQL:

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

Run:

```
CREATE DATABASE snipeit;
CREATE USER 'snipeit'@'127.0.0.1' IDENTIFIED BY 'StrongPassword';
CREATE USER 'snipeit'@'localhost' IDENTIFIED BY 'StrongPassword';
GRANT ALL PRIVILEGES ON snipeit.* TO 'snipeit'@'127.0.0.1';
GRANT ALL PRIVILEGES ON snipeit.* TO 'snipeit'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

✓ Replace `StrongPassword` with a secure password.

---

## Step 6: Download and Configure Snipe-IT

### 6.1 Clone Snipe-IT Repository

```
cd /var/www
sudo git clone https://github.com/snipe/snipe-it snipe-it
```

### 6.2 Set Up `.env` Configuration

```
cd /var/www/snipe-it
cp .env.example .env
sudo nano .env
```

Modify:

```
APP_URL=http://localhost:8095
APP_TIMEZONE='Europe/Athens'

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=snipeit
DB_USERNAME=snipeit
DB_PASSWORD=StrongPassword
```

Save and exit ( `CTRL+X`, `Y`, `Enter` ).

## 6.3 Set Permissions

```
sudo chown -R apache:apache /var/www/snipe-it
sudo chmod -R 755 /var/www/snipe-it
sudo chmod -R 775 /var/www/snipe-it/storage /var/www/snipe-
it/bootstrap/cache
```

---

## Step 7: Install Dependencies and Generate App Key

### 7.1 Update Dependencies

```
composer update --no-plugins --no-scripts
composer install --no-dev --prefer-source --no-plugins --no-scripts
```

### 7.2 Generate Laravel App Key

```
php artisan key:generate
```

---

## Step 8: Install and Configure Nginx

### 8.1 Install Nginx

```
sudo dnf install nginx -y
```

### 8.2 Configure Nginx for Snipe-IT

Create a new configuration file:

```
sudo nano /etc/nginx/conf.d/snipeit.conf
```

Paste:

```
server {
    listen 8095;
    server_name 127.0.0.1;

    root /var/www/snipe-it/public;
    index index.php index.html;

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location ~ \.php$ {
        include fastcgi.conf;
        include fastcgi_params;
        fastcgi_pass unix:/run/php-fpm/www.sock;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    }

    error_log /var/log/nginx/snipeit_error.log;
    access_log /var/log/nginx/snipeit_access.log;
}
```

Save and exit (`CTRL+X`, `Y`, `Enter`).

---

## 8.3 Modify Nginx Main Configuration

```
sudo nano /etc/nginx/nginx.conf
```

Find:

```
include /etc/nginx/conf.d/*.conf;
```

Ensure it exists.

Save and exit.

---

## Step 9: Start and Enable Services

```
sudo systemctl start nginx php-fpm mariadb
sudo systemctl enable nginx php-fpm mariadb
```

---

## Step 10: Clear Laravel Cache

```
php artisan config:clear
php artisan cache:clear
php artisan route:clear
php artisan view:clear
```

---

## Step 11: Restart Services

```
sudo systemctl restart nginx php-fpm mariadb
```

---

## Step 12: Test Snipe-IT

Open your browser and go to:

```
http://127.0.0.1:8095/setup
```

✓ Follow the setup process in the browser.

---

## Troubleshooting

---

### 1. 502 Bad Gateway

If you get **502 Bad Gateway**, check:

```
sudo systemctl status php-fpm
```

Check PHP-FPM socket:

```
ls /run/php-fpm/www.sock
```

---

### 2. Database Connection Errors

If you see **"Could not find driver"**:

```
sudo dnf install php-mysqldb -y
sudo systemctl restart php-fpm
```

If **"Access Denied"**:

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

Run:

```
SET PASSWORD FOR 'snipeit'@'127.0.0.1' = PASSWORD('StrongPassword');
SET PASSWORD FOR 'snipeit'@'localhost' = PASSWORD('StrongPassword');
FLUSH PRIVILEGES;
EXIT;
```

Restart MySQL:

```
sudo systemctl restart mariadb
```

---

### 3. Error 500 (Internal Server Error)

Check Laravel logs:

```
sudo tail -f /var/www/snipe-it/storage/logs/laravel.log
```

Try:

```
php artisan config:clear  
php artisan cache:clear  
sudo systemctl restart nginx php-fpm
```

---

## Conclusion

---

✅ Snipe-IT is now running on AlmaLinux 9 with PHP 8.3, Nginx, and MariaDB.