# Running the Secure Search Server as a Linux Daemon

## 1 Overview

This document provides step-by-step instructions to run the Secure Search Server as a Linux daemon using `systemd`. It covers creating a service file, reloading `systemd`, starting/enabling the service, and monitoring/managing the service.

## 2 Creating the Systemd Service File

Create a new service file (e.g., `/etc/systemd/system/secure-search.service`) with the following content. Adjust the paths and user according to your environment.

```
[Unit]
Description=Secure Search Service
After=network.target

[Service]
Type=simple
User=your_username
WorkingDirectory=/path/to/your/server/directory
ExecStart=/usr/bin/python3 /path/to/your/server.py
Restart=on-failure
Environment="PYTHONUNBUFFERED=1"

[Install]
WantedBy=multi-user.target
```

**Notes:**

- Replace `your_username` with the Linux user under which the server should run.
- Replace `/path/to/your/server/directory` with the directory where your server files reside.
- Replace `/path/to/your/server.py` with the correct path to your `server.py` file.

## 3 Reloading Systemd

After creating or modifying the service file, reload the `systemd` manager configuration:

```
sudo systemctl daemon-reload
```

## 4 Starting and Enabling the Service

### 4.1 Start the Service

To start the service immediately, execute:

```
sudo systemctl start secure-search
```

### 4.2 Enable the Service at Boot

To ensure the service starts automatically at boot time, run:

```
sudo systemctl enable secure-search
```

## 5 Monitoring and Managing the Service

### 5.1 Checking Service Status

To check the status of the service:

```
sudo systemctl status secure-search
```

### 5.2 Viewing Logs

To view logs for the service, use `journalctl`:

```
sudo journalctl -u secure-search
```

### 5.3 Stopping or Restarting the Service

To stop the service:

```
sudo systemctl stop secure-search
```

To restart the service:

```
sudo systemctl restart secure-search
```

## 6 Conclusion

By following these instructions, you can run your Secure Search Server as a background daemon on a Linux system using `systemd`. For further customization (e.g., adding additional environment variables or resource limits), modify the service file as needed.