

Einrichtungsanleitung: Automatischer POS-Drucker für WooCommerce

Einleitung

Herzlichen Glückwunsch zum Erwerb des POS-Drucker Plugins! Dieses System ermöglicht es Ihnen, eingehende Bestellungen aus Ihrem WooCommerce-Onlineshop automatisch auf einem lokalen Thermodrucker auszudrucken. Die Bestellungen werden vom WordPress-Plugin per E-Mail an einen Print-Server ([AutoPrint Server für WooCommerce](#)) gesendet, der dann den eigentlichen Druckauftrag an den angeschlossenen Drucker weiterleitet.

Diese Anleitung führt Sie Schritt für Schritt durch die Einrichtung des Print-Servers und die Konfiguration des WordPress-Plugins.

1. Hardware-Setup des Print-Servers

Der Print-Server ist ein kleiner Computer (empfohlen: [AutoPrint Server für WooCommerce](#)), der dauerhaft in Ihrem Restaurant läuft und die Druckaufträge empfängt und verarbeitet.

****Benötigte Hardware:****

- * Print-Server (z. B. [AutoPrint Server für WooCommerce](#)) mit Netzteil
- * SD-Karte mit installiertem Betriebssystem (wird bei der Bestellung des Print-Servers vorinstalliert mitgeliefert)
- * USB-Thermodrucker ([GastroPrint Pro 80](#)(ESC/POS-kompatibel)) mit Netzteil und USB-Kabel
- * LAN-Kabel (Ethernet-Kabel)

****Verbindungen:****

1. ****Drucker verbinden:**** Schließen Sie den USB-Thermodrucker mit dem USB-Kabel an einen freien USB-Port des Print-Servers an. Verbinden Sie den Drucker auch mit dem Stromnetz.
2. ****Netzwerk verbinden:**** Verbinden Sie den Print-Server mit einem LAN-Kabel direkt mit Ihrem Internet-Router. Eine stabile Kabelverbindung ist für den zuverlässigen Empfang der Druckaufträge wichtig.
3. ****Stromversorgung:**** Schließen Sie den Print-Server mit seinem Netzteil an das Stromnetz an und starten Sie ihn.

2. Software-Setup des Print-Servers

Auf dem Print-Server müssen zwei kleine Programme (Python-Skripte) laufen: Eines, das die E-Mails abruft (`email_print_poller.py`), und eines, das den Drucker ansteuert (`druck_api.py`).

****Vorbereitung:****

1. ****SSH-Zugang:**** Stellen Sie sicher, dass Sie per SSH auf Ihren Print-Server zugreifen können. Die folgenden Befehle werden im Terminal des Print-Servers ausgeführt.
2. ****Verzeichnis erstellen (optional):**** Sie können die Skripte direkt im Home-Verzeichnis (`/home/admin/` oder `/home/pi/`) speichern oder ein eigenes Verzeichnis erstellen:

```
``bash

mkdir ~/printserver

cd ~/printserver

``
```

3. ****Skripte kopieren:**** Kopieren Sie die beiden Python-Dateien (`email_print_poller.py` und `druck_api.py`) in das gewählte Verzeichnis auf dem Print-Server (z. B. `/home/admin/printserver/`).

****Einrichtung der E-Mail-Abfrage:****

1. ****Dedizierte E-Mail-Adresse:**** Richten Sie eine ****neue, separate E-Mail-Adresse**** ein, die ****ausschließlich**** für den Empfang der Druckaufträge verwendet wird (z. B. bei GMX, Gmail,

Web.de oder Ihrem eigenen Hoster). ****Wichtig:**** Dieses Postfach darf für nichts anderes verwendet werden, da das Skript alle ungelesenen E-Mails mit passendem Betreff verarbeitet. Notieren Sie sich:

- * Die vollständige E-Mail-Adresse
- * Das Passwort des E-Mail-Kontos
- * Den IMAP-Servernamen (z. B. ``imap.gmx.net``)
- * Den IMAP-Port (meist ``993`` für SSL/TLS)

2. ****Konfigurationsdatei erstellen (`.env`):****

- * Wechseln Sie in das Verzeichnis, in dem die Skripte liegen (z. B. ``cd ~/printserver``).
- * Erstellen Sie die Konfigurationsdatei: ``nano .env``
- * Fügen Sie folgenden Inhalt ein und ersetzen Sie die Platzhalter durch Ihre ****eigenen E-Mail-Zugangsdaten****:

```
``dotenv
```

```
# .env - Konfiguration für den E-Mail-Druckauftrag-Abruf
```

```
# --- IMAP Zugangsdaten ---
```

```
# Ersetzen Sie die Werte mit den Daten Ihres DEDIZIERTEN E-Mail-Kontos!
```

```
IMAP_SERVER=imap.gmx.net
```

```
IMAP_PORT=993
```

```
EMAIL_ADDRESS=IhreDruckerEmail@provider.de
```

```
EMAIL_PASSWORD=IhrEmailPasswort
```

```
# --- Lokaler Druckserver ---
```

```
# (Normalerweise nicht ändern)
```

```
PRINT_SERVER_URL=http://127.0.0.1:5000/print-order
```

```
# --- Abrufintervall ---
```

```
# (Wie oft soll auf neue E-Mails geprüft werden? In Sekunden)
```

```
POLL_INTERVAL_SECONDS=15
```

```
...
```

- * Speichern Sie die Datei (in `nano`: Strg+O, Enter) und schließen Sie den Editor (Strg+X).

- * ****Wichtig:**** Sichern Sie diese Datei gut ab, damit nur der ausführende Benutzer Lesezugriff hat: `chmod 600 .env`

****Installation der Abhängigkeiten (Python-Bibliotheken):****

Moderne Linux-Systeme (wie Raspberry Pi OS) schützen die System-Python-Installation. Daher installieren wir die benötigten Pakete in einer isolierten "virtuellen Umgebung":

1. **Virtuelle Umgebung erstellen:**

```
```bash
```

```
cd ~/printserver # Wechseln Sie in Ihr Skript-Verzeichnis
```

```
python3 -m venv venv
```

```
...
```

(Dies erstellt einen Ordner `venv` in Ihrem Verzeichnis.)

### **2. \*\*Virtuelle Umgebung aktivieren:\*\***

```
```bash
```

```
source venv/bin/activate
```

```
...
```

(Ihre Terminal-Eingabezeile sollte sich ändern und `(venv)` anzeigen.)

3. **Pakete installieren:**

```
```bash
```

```
pip install python-dotenv requests Flask python-escpos Pillow
```

```
...
```

### **4. \*\*Virtuelle Umgebung deaktivieren (optional für jetzt):\*\***

```
```bash
```

deactivate

...

****Systemd Services einrichten (Automatischer Start):****

Damit die Skripte automatisch starten und im Hintergrund laufen, richten wir `systemd` Services ein.

1. **Service für den Druckserver (`druck_api.py`):**

- * Datei erstellen: `sudo nano /etc/systemd/system/printserver.service`
- * Inhalt einfügen (passen Sie `User`, `Group`, `WorkingDirectory` und Pfade bei Bedarf an):

```
``ini
```

```
[Unit]
```

```
Description=Lokaler Flask Druckserver (druck_api.py)
```

```
After=network.target
```

```
[Service]
```

```
User=admin # Benutzer, dem die Skripte gehören (oft 'pi' oder 'admin')
```

```
Group=admin # Gruppe des Benutzers
```

```
WorkingDirectory=/home/admin/printserver # Pfad zu Ihren Skripten
```

```
ExecStart=/home/admin/printserver/venv/bin/python
```

```
/home/admin/printserver/druck_api.py # Python aus venv nutzen!
```

```
Restart=always
```

```
RestartSec=10
```

```
StandardOutput=journal
```

```
StandardError=journal
```

```
[Install]
```

```
WantedBy=multi-user.target
```

...

- * Speichern und schließen (Strg+O, Enter, Strg+X).

2. ****Service für den E-Mail-Poller (`email\_print\_poller.py`):****

- * Datei erstellen: `sudo nano /etc/systemd/system/emailpoller.service`
- * Inhalt einfügen (passen Sie `User`, `Group`, `WorkingDirectory` und Pfade bei Bedarf an):

```
``ini
```

```
[Unit]
```

```
Description=E-Mail Print Poller Service (email_print_poller.py)
```

```
After=network.target printserver.service # Startet nach Netzwerk und Druckserver
```

```
[Service]
```

```
User=admin # Benutzer, dem die Skripte gehören
```

```
Group=admin # Gruppe des Benutzers
```

```
WorkingDirectory=/home/admin/printserver # Pfad zu Ihren Skripten
```

```
ExecStart=/home/admin/printserver/venv/bin/python  
/home/admin/printserver/email_print_poller.py # Python aus venv nutzen!
```

```
Restart=always
```

```
RestartSec=15 # Etwas längere Pause bei Neustart
```

```
StandardOutput=journal
```

```
StandardError=journal
```

```
[Install]
```

```
WantedBy=multi-user.target
```

...

- * Speichern und schließen (Strg+O, Enter, Strg+X).

3. ****Systemd neu laden und Services aktivieren/starten:****

```
```bash

sudo systemctl daemon-reload

sudo systemctl enable printserver.service

sudo systemctl enable emailpoller.service

sudo systemctl start printserver.service

sudo systemctl start emailpoller.service

```
```

4. ****Status prüfen:****

```
```bash

sudo systemctl status printserver.service

sudo systemctl status emailpoller.service

```
```

Beide sollten als "active (running)" angezeigt werden. Bei Problemen prüfen Sie die Logs mit ``sudo journalctl -u printserver.service`` bzw. ``sudo journalctl -u emailpoller.service``.

Der Print-Server ist nun bereit, E-Mails zu empfangen und Druckaufträge zu verarbeiten.

3. WordPress Plugin Installation & Konfiguration

1. ****Installation:****

- * Laden Sie die Plugin-Datei (``woo-drucker-email.php.zip`` oder den entpackten Ordner ``woo-drucker-email``) herunter.
- * Gehen Sie in Ihrem WordPress-Adminbereich zu "Plugins" -> "Installieren".
- * Klicken Sie auf "Plugin hochladen", wählen Sie die ZIP-Datei aus und klicken Sie auf "Jetzt installieren".
- * Alternativ: Entpacken Sie die ZIP-Datei und laden Sie den Ordner ``woo-drucker-email`` per FTP in das Verzeichnis ``/wp-content/plugins/`` Ihrer WordPress-Installation hoch.

2. ****Aktivierung:****

- * Gehen Sie zu "Plugins" -> "Installierte Plugins".
- * Suchen Sie nach "Woo-Drucker (POS-Drucker)" und klicken Sie auf "Aktivieren".

3. **Konfiguration:**

- * Nach der Aktivierung finden Sie im WordPress-Adminmenü einen neuen Punkt namens "POS-Drucker". Klicken Sie darauf.
- * **Empfänger E-Mail-Adresse (Print-Server):** Tragen Sie hier **exakt die dedizierte E-Mail-Adresse** ein, die Sie in Schritt 2.1 eingerichtet und in der `.env`-Datei auf dem Print-Server konfiguriert haben. Das Plugin sendet die Druckaufträge an diese Adresse.
- * **Bon-Typen:** Wählen Sie aus, welche Bons bei einer Bestellung gedruckt werden sollen (Küche, Fahrer, Theke). Sie können mehrere auswählen.
- * **Logo:** Laden Sie optional ein Logo hoch, das auf den Fahrer- und Thekenbons gedruckt wird.
- * **Footer Text:** Geben Sie einen Text ein, der am Ende jedes Bons erscheinen soll (z. B. "Danke für Ihre Bestellung!").
- * Klicken Sie auf "Änderungen speichern".

4. Testen des Systems

1. **Testdruck senden:**

- * Gehen Sie zur "POS-Drucker"-Einstellungsseite im WordPress-Admin.
- * Klicken Sie auf den Button "Testdruck Senden".
- * WordPress sollte eine Erfolgsmeldung anzeigen, dass die Test-E-Mail gesendet wurde.
- * **Warten Sie ca. 15-30 Sekunden.** Der Print-Server prüft das Postfach nur alle paar Sekunden.
- * Der angeschlossene Thermodrucker sollte nun einen Testbon ausdrucken.
- * **Fehlersuche:** Wenn nichts gedruckt wird:
 - * Prüfen Sie das E-Mail-Postfach (Ihre Drucker-E-Mail): Ist die Test-E-Mail angekommen? Ist sie noch ungelesen?
 - * Prüfen Sie die Logs auf dem Print-Server: `sudo journalctl -f -u emailpoller.service -u printserver.service`. Suchen Sie nach Fehlermeldungen (z. B. falsches Passwort, IMAP-Verbindungsfehler, JSON-Parsing-Fehler, Druckerfehler).

- * Stellen Sie sicher, dass Ihr WordPress-System korrekt E-Mails versenden kann (ggf. SMTP-Plugin in WordPress konfigurieren).

2. ****Testbestellung:****

- * Wenn der Testdruck funktioniert, führen Sie eine normale Testbestellung in Ihrem WooCommerce-Shop durch.

- * Schließen Sie die Bestellung ab.

- * Nach kurzer Zeit sollte(n) der/die entsprechende(n) Bon(s) am Drucker ausgegeben werden.

Herzlichen Glückwunsch! Ihr automatisches Drucksystem ist nun eingerichtet.

Benötigen Sie weitere Hilfe bei der Einrichtung Ihres Restaurant/Lieferdienst Bestellungen-Drucksystem? Kontaktieren Sie uns einfach, wir helfen Ihnen gerne weiter!

[Kontakt](#)