



Raspberry Pi als Cloud-Lösung und Entwicklungsplattform

*Von der Einrichtung des Systems bis zur Datenanalyse in
der Cloud*

Erstellt von:

Frédéric C. Kurbel

Studiengang: **Computational and Data Science**

Modul: **Computer Science**

Professor: **Ana Petrus**

Abgabedatum: 10.12.2024

Fachhochschule Graubünden

Inhaltsverzeichnis

1 Einleitung

- 1.1 Überblick zum Projekt
- 1.2 Der Raspberry Pi
- 1.3 Projektressourcen

2 Methodik

3 Raspberry Pi – Einrichtung

- 3.1 Installation des Betriebssystems
- 3.2 Netzwerkverbindung und SSH-Zugriff

4 VNC-Viewer Einrichtung

- 4.1 Download und Installation des VNC-Viewers
- 4.2 Verbindung zum Raspberry Pi herstellen
- 4.3 Authentifizierung und Zugriff
- 4.4 Graphischer Zugriff auf den Raspberry Pi
- 4.5 Zusätzlicher Hinweise

5 Cloud-Setup

- 5.1 Installation des Webservers
- 5.2 Einrichtung der Datenbank
- 5.3 Installation von Nextcloud
 - 5.3.1 Schritt 1: Installation der Software
 - 5.3.2 Schritt 2: Konfiguration des Apache-Webservers
 - 5.3.3 Schritt 3: Zugriff auf Nextcloud über den Browser
 - 5.3.4 Hinweis

6 Python-Skripte in der Cloud

- 6.1 Einrichtung von Jupyter Notebook
 - 6.1.1 Schritt 1: Installation von Jupyter Notebook
 - 6.1.2 Schritt 2: Konfiguration von Jupyter Notebook
 - 6.1.3 Schritt 3: Starten von Jupyter Notebook

6.1.4	Schritt 4: Titanic-Dataset in der Cloud speichern und verar- beiten
6.1.5	Schritt 5: Titanic-Dataset in Jupyter Notebook analysieren . .
6.1.6	Erwarteter Output
6.1.7	Erwarteter Output

7 Ergebnisse

8 Diskussion und Ausblick

9 Fazit

10 Quellen

1 Einleitung

1.1 Überblick zum Projekt

Diese Dokumentation beschreibt den Prozess der Einrichtung eines Raspberry Pi als Cloud-Server, das Einrichten einer SQL-Datenbank und die Ausführung von Python-Skripten innerhalb der Cloud. Der Fokus liegt auf der Installation und Konfiguration der Hardware und Software sowie auf einem Beispiel zur Datenanalyse. Das Projekt bietet eine Grundlage für weiterführende Arbeiten im Bereich Data Science und Automatisierung.

1.2 Der Raspberry Pi

Der Raspberry Pi ist ein kostengünstiger und vielseitiger Einplatinencomputer, der häufig für Projekte in den Bereichen Cloud-Computing, Automatisierung und IoT verwendet wird. In diesem Projekt wurde ein Raspberry Pi 4 als Grundlage für die Implementierung einer Cloud-Lösung und die Ausführung von Python-Skripten eingesetzt.

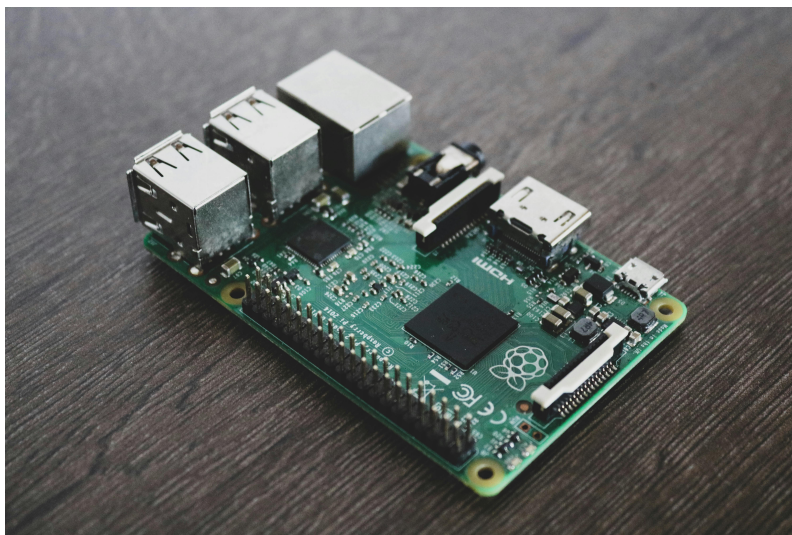


Figure 1: Ein Raspberry Pi 4, wie er in diesem Projekt verwendet wurde.

Im Folgenden wird detailliert beschrieben, wie der Raspberry Pi eingerichtet und konfiguriert wurde, um als Cloud-Server mit SQL-Unterstützung und Jupyter Notebook zu dienen.

1.3 Projektressourcen

Um den Quellcode sowie weiterführende Informationen zu diesem Projekt abzurufen, besuchen Sie das zugehörige GitHub-Repository: https://github.com/Fredeys/raspberry-pi_cloud_dev.

2 Methodik

Die Methodik gliedert sich in drei Hauptbereiche:

1. Einrichtung des Raspberry Pi (Hardware und Betriebssystem).
2. Konfiguration eines Cloud-Servers (Webserver und Datenbank).
3. Anwendung von Python-Skripten in der Cloud.

3 Raspberry Pi – Einrichtung

3.1 Installation des Betriebssystems

1. Setze die SD-Karte in einen PC oder Laptop ein und lade den *Raspberry Pi Imager* von <https://www.raspberrypi.com/software/> herunter.
2. Wähle das Betriebssystem *Raspberry Pi OS (64-BIT)*.
3. Konfiguriere Benutzernamen, Passwort, Landeseinstellungen und WLAN-Verbindung.
4. Installiere das Betriebssystem auf der SD-Karte und überprüfe die Installation.
5. Entnehme die SD-Karte und setze sie in den Raspberry Pi ein.

3.2 Netzwerkverbindung und SSH-Zugriff

1. Lade *Angry IP Scanner* von <https://angryip.org> herunter. Alternativ können auch andere Lösungen verwendet werden, die IP-Adressen von Geräten im Netzwerk auffinden.
2. Identifiziere die IP-Adresse des Raspberry Pi im lokalen Netzwerk.

3. Verbinde dich über SSH mit dem Raspberry Pi. Verwende dafür den zuvor festgelegten Benutzernamen und die IP-Adresse:

```
ssh Benutzername@IP-Adresse
```

4. Nach dem Ausführen des Befehls erscheint folgende Meldung, wenn es die erste Verbindung zu diesem Host ist:

```
The authenticity of host 'IP-Adresse' (IP-Adresse) can't be
established.
ED25519 key fingerprint is SHA256:
    QlDRk69v5IOZAut3CVMmCAduwqzXpGSymdHJB/FWars.
Are you sure you want to continue connecting (yes/no/[fingerprint
])?
```

5. Bestätige die Verbindung durch Eingabe von **yes**. Dadurch wird die IP-Adresse des Raspberry Pi zur Liste der bekannten Hosts hinzugefügt:

```
Warning: Permanently added 192.168.1.51 (ED25519) to the list of
known hosts.
```

6. Logge dich mit dem zuvor festgelegten Passwort ein. Beachte, dass das Passwort aus Sicherheitsgründen nicht im Terminal angezeigt wird.
7. Konfiguriere den Raspberry Pi mit folgendem Befehl:

```
sudo raspi-config
```

Durch den Befehl wird Zugriff zur graphischen Benutzeroberfläche erlangt, und die Einstellungen können angepasst werden:

- Wähle *Interface Options*.
- Wähle *VNC Enable/disable graphical remote desktop access* und bestätige mit **<Yes>**. Die darauf folgende Meldung *The VNC Server is enabled* kann mit **<Ok>** bestätigt werden.
- Wähle **<Finish>**, um wieder in das Raspberry Terminal zu gelangen.

4 VNC-Viewer Einrichtung

4.1 Download und Installation des VNC-Viewers

Der VNC-Viewer kann über den folgenden Link heruntergeladen werden: https://www.realvnc.com/de/connect/download/viewer/macos/?lai_vid=WKMJdXxy0HBkz&lai_sr=0-4&lai_sl=1&lai_p=1

4.2 Verbindung zum Raspberry Pi herstellen

Nach dem Öffnen der Applikation wird eine neue Verbindung angelegt:

- Für den Input *VNC-Server* wird die zuvor ermittelte IP-Adresse des Raspberry Pi verwendet.
- Ein Name für die Verbindung wird vergeben, um das Projekt leichter identifizieren zu können.

4.3 Authentifizierung und Zugriff

Um die Verbindung zum Raspberry Pi herzustellen, sind folgende Schritte erforderlich:

1. Beim ersten Verbindungsaufbau erscheint die folgende Meldung:

Identity Check - VNC Viewer has no record of connecting to this VNC Server,so its identity cannot be checked.

Diese Warnung weist darauf hin, dass die Verbindung zum VNC-Server nicht überprüft werden kann. Bestätige die Meldung durch Klicken auf *Continue*.

2. Nach der Bestätigung fordert der VNC-Viewer eine Authentifizierung an. Gib hierfür die folgenden Zugangsdaten ein:

- **Benutzername:** Der Benutzername, der während der Einrichtung des Raspberry Pi festgelegt wurde.
- **Passwort:** Das ebenfalls während der Einrichtung festgelegte Passwort.

3. Nach erfolgreicher Authentifizierung erhältst du Zugriff auf die graphische Oberfläche des Raspberry Pi.

4.4 Graphischer Zugriff auf den Raspberry Pi

Nach der erfolgreichen Authentifizierung kann der VNC-Viewer auf die graphische Oberfläche des Raspberry Pi zugreifen. Um den VNC-Server zu starten, verwende den folgenden Befehl im Raspberry Terminal:

```
vncserver-x11
```

4.5 Zusätzlicher Hinweise

Falls ein VNC-Account erstellt und mit dem VNC-Viewer verknüpft wird, ist der Zugriff auf den Raspberry Pi auch ortsunabhängig möglich.

5 Cloud-Setup

5.1 Installation des Webservers

Zur Bereitstellung der Cloud wird der *Apache*-Webserver verwendet. Die Installation erfolgt mit folgenden Befehlen:

```
sudo apt update  
sudo apt install apache2 -y  
sudo apt install php php-gd php-curl php-zip php-xml php-mysql -y
```

5.2 Einrichtung der Datenbank

MariaDB wird als Datenbank-Management-System installiert und konfiguriert. Die Installation erfolgt mit folgendem Befehl:

```
sudo apt install mariadb-server -y  
sudo mysql_secure_installation
```


Nach der Installation erfolgt die Konfiguration. Es werden folgende Schritte durchgeführt:

1. Führe den Befehl `sudo mysql_secure_installation` aus und beantworte die folgenden Eingabeaufforderungen:

- **Enter current password for root:** Bestätige mit der **Enter**-Taste (kein Passwort erforderlich).
- **Switch to unix_socket_authentication?** **[Y/n]**: Antworte mit **n**, um diese Option abzulehnen.
- **Change the root password?** **[Y/n]**: Antworte mit **n**, um kein neues Passwort zu setzen.
- **Remove anonymous users?** **[Y/n]**: Antworte mit **Y**, um anonyme Benutzer zu entfernen.
- **Disallow root login remotely?** **[Y/n]**: Antworte mit **Y**, um Root-Logins über das Netzwerk zu verbieten.
- **Remove test database and access to it?** **[Y/n]**: Antworte mit **Y**, um die Test-Datenbank zu löschen.
- **Reload privilege tables now?** **[Y/n]**: Antworte mit **Y**, um die Privilegientabellen neu zu laden.

2. Nach Abschluss dieser Schritte melde dich bei MariaDB an:

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

Wenn du zur Eingabe eines Passworts aufgefordert wirst (*Enter password:*), kannst du entweder ein neues Passwort setzen oder mit der **Enter**-Taste fortfahren.

3. Nachdem du Zugriff auf MariaDB erhalten hast, lege eine neue Datenbank an und konfiguriere die Benutzerrechte:

```
CREATE DATABASE nextcloud;
```

```
GRANT ALL ON nextcloud.* TO 'nextclouduser'@'localhost' IDENTIFIED BY  
    'deinPasswort';  
FLUSH PRIVILEGES;  
EXIT;
```

Durch diese Schritte ist die Datenbank erfolgreich eingerichtet und bereit für die Verwendung durch Nextcloud.

5.3 Installation von Nextcloud

Die Installation und Konfiguration von Nextcloud erfolgt in mehreren Schritten. Alle verfügbaren Versionen können auf der offiziellen Webseite eingesehen werden: <https://download.nextcloud.com/server/releases/>

5.3.1 Schritt 1: Installation der Software

Führe die folgenden Befehle im Raspberry Terminal aus, um die Software herunterzuladen und zu entpacken:

```
wget https://download.nextcloud.com/server/releases/nextcloud-28.0.0.  
tar.bz2  
tar -xvf nextcloud-28.0.0.tar.bz2  
sudo mv nextcloud /var/www/html/  
sudo chown -R www-data:www-data /var/www/html/nextcloud
```

5.3.2 Schritt 2: Konfiguration des Apache-Webservers

Nach der Installation von Nextcloud muss der Apache-Webserver konfiguriert werden. Öffne die Konfigurationsdatei mit folgendem Befehl:

```
sudo nano /etc/apache2/sites-available/nextcloud.conf
```

Füge den folgenden Inhalt in die Datei ein, speichere die Änderungen und beende den Editor:

```
<VirtualHost *:80>
```

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html/nextcloud

<Directory /var/www/html/nextcloud/>
    Require all granted
    AllowOverride All
    Options FollowSymLinks MultiViews

    <IfModule mod_dav.c>
        Dav off
    </IfModule>
</Directory>

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Aktiviere die neue Konfiguration und die benötigten Apache-Module, und starte den Apache-Webserver neu:

```
sudo a2ensite nextcloud
sudo a2enmod rewrite headers env dir mime
sudo systemctl restart apache2
```

5.3.3 Schritt 3: Zugriff auf Nextcloud über den Browser

Nach der erfolgreichen Konfiguration kannst du über einen Webbrowser auf Nextcloud zugreifen. Gib dazu folgende URL ein, wobei **IP-Adresse** durch die IP deines Raspberry Pi ersetzt wird:

```
http://IP-Adresse/nextcloud
```

Im Browser erscheint die Installationsseite von Nextcloud. Gib folgende Informationen ein:

- **Benutzername und Passwort:** Lege einen neuen Administrator-Benutzer an.
- **Datenbank-Konfiguration:**
 - Datenbank-Benutzer: `nextclouduser`
 - Datenbank-Name: `nextcloud`
 - Passwort: Das während der Einrichtung festgelegte Passwort.

Bestätige die Installation. Nach Abschluss wirst du auf die Hauptseite weitergeleitet, auf der verschiedene empfohlene Apps zur Installation angezeigt werden. Installiere die vorgeschlagenen Apps, um die Funktionalität von Nextcloud zu erweitern.

5.3.4 Hinweis

Nach der Installation steht Nextcloud zur Verfügung und kann für Dateifreigabe, Synchronisation und andere Cloud-Dienste verwendet werden.

6 Python-Skripte in der Cloud

6.1 Einrichtung von Jupyter Notebook

Die Installation und Konfiguration von Jupyter Notebook erfolgt in mehreren Schritten:

6.1.1 Schritt 1: Installation von Jupyter Notebook

Führe die folgenden Befehle aus, um Jupyter Notebook zu installieren:

```
sudo apt update
sudo apt install python3-pip -y
pip3 install notebook
```

6.1.2 Schritt 2: Konfiguration von Jupyter Notebook

Generiere die Konfigurationsdatei für Jupyter Notebook:

```
jupyter notebook --generate-config
```

Die Konfigurationsdatei wird unter dem Pfad `~/.jupyter/jupyter_notebook_config.py` erstellt. Öffne sie mit folgendem Befehl:

```
nano ~/.jupyter/jupyter_notebook_config.py
```

Füge den folgenden Code in die Datei ein, speichere die Änderungen und kehre zum Terminal zurück:

```
c.NotebookApp.port = 8889
c.NotebookApp.ip = '0.0.0.0'
c.NotebookApp.open_browser = False
```

6.1.3 Schritt 3: Starten von Jupyter Notebook

Starte Jupyter Notebook mit folgendem Befehl:

```
jupyter notebook
```

Hinweis: Das Terminal gibt einen Link aus, der in den Browser eingefügt werden muss, da sich das Notebook nicht automatisch öffnet.

6.1.4 Schritt 4: Titanic-Dataset in der Cloud speichern und verarbeiten

Installiere das Python-Paket `seaborn` und überprüfe die Installation:

```
sudo apt update
sudo apt install python3-seaborn -y
python3 -c "import seaborn as sns; print('Seaborn installiert ')"
```

Speichere das Titanic-Dataset als CSV-Datei:

1. Erstelle ein Python-Skript mit folgendem Inhalt:

```
import seaborn as sns
import pandas as pd

# Titanic-Dataset laden
```

```
titanic = sns.load_dataset('titanic')

# Dataset als CSV speichern
titanic.to_csv('/home/pi/titanic.csv', index=False)
print("Titanic-Dataset wurde als 'titanic.csv' gespeichert.")
```

2. Speichere das Skript als `save_titanic.py` und führe es aus:

```
python save_titanic.py
```

3. Verschiebe die erzeugte CSV-Datei in das Nextcloud-Verzeichnis:

```
sudo mv /home/pi/titanic.csv /var/www/html/nextcloud/
```

4. Setze die Zugriffsrechte für die Datei:

```
sudo chown www-data:www-data /var/www/html/nextcloud/titanic.csv
sudo chmod 644 /var/www/html/nextcloud/titanic.csv
```

6.1.5 Schritt 5: Titanic-Dataset in Jupyter Notebook analysieren

Öffne Jupyter Notebook mit dem folgenden Befehl:

```
jupyter notebook --port=8889 --no-browser
```

Erstelle ein neues Notebook und führe den folgenden Python-Code aus, um das Titanic-Dataset zu laden und die ersten Zeilen auszugeben:

```
import pandas as pd

# Titanic-Dataset laden
df = pd.read_csv('/var/www/html/nextcloud/titanic.csv')
print(df.head())
```

6.1.6 Erwarteter Output

Die Ausgabe des Titanic-Datasets wird in drei Teilen dargestellt, um die Übersichtlichkeit zu erhöhen:

6.1.7 Erwarteter Output

Die Ausgabe des Titanic-Datasets wird in drei Teilen dargestellt, um die Übersichtlichkeit zu erhöhen:

	survived	pclass	sex	age	sibsp	parch
0	0	3	male	22.0	1	0
1	1	1	female	38.0	1	0
2	1	3	female	26.0	0	0
3	1	1	female	35.0	1	0
4	0	3	male	35.0	0	0

	fare	embarked	class	who	adult_male
0	7.2500	S	Third	man	True
1	71.2833	C	First	woman	False
2	7.9250	S	Third	woman	False
3	53.1000	S	First	woman	False
4	8.0500	S	Third	man	True

	deck	embark_town	alive	alone
0	NaN	Southampton	no	False
1	C	Cherbourg	yes	False
2	NaN	Southampton	yes	True
3	C	Southampton	yes	False
4	NaN	Southampton	no	True

Durch diese Schritte wird das Titanic-Dataset erfolgreich in der Cloud gespeichert und mit Jupyter Notebook analysiert.

7 Ergebnisse

Durch die beschriebenen Schritte wurde eine voll funktionsfähige Cloud auf dem Raspberry Pi eingerichtet. Python-Skripte wie Datenanalysen können in dieser Umgebung effizient ausgeführt werden. Zudem wurde eine Datenbank eingerichtet,

die bislang jedoch nicht weiter verwendet wurde und für zukünftige Erweiterungen bereitsteht.

8 Diskussion und Ausblick

Dieses Projekt zeigt die Vielseitigkeit eines Raspberry Pi als kostengünstige und flexible Cloud-Lösung. Zukünftige Arbeiten könnten Sicherheitsaspekte wie SSL-Zertifikate oder Firewall-Konfigurationen integrieren, um die Cloud noch sicherer zu gestalten.

9 Fazit

Die Einrichtung eines Raspberry Pi für Cloud-Computing und Datenanalysen bietet eine exzellente Grundlage für weiterführende Projekte im Bereich Data Science und Automatisierung.

10 Quellen

1. Raspberry Pi Foundation. *Raspberry Pi Imager*. Zugriff unter: <https://www.raspberrypi.com/software/>
2. RealVNC. *VNC Viewer Download*. Zugriff unter: <https://www.realvnc.com/connect/download/viewer/>
3. Nextcloud. *Nextcloud Releases*. Zugriff unter: <https://download.nextcloud.com/server/releases/>
4. Debian. *MariaDB Secure Installation Guide*. Zugriff unter: https://mariadb.com/kb/en/mysql_secure_installation/
5. Seaborn Library. *Load Titanic Dataset*. Zugriff unter: https://seaborn.pydata.org/generated/seaborn.load_dataset.html
6. Raspberry Pi Beginner Tutorial. Zugriff unter: <https://www.youtube.com/watch?v=6vJeDlffCAQ>

7. Nextcloud Installation Tutorial. Zugriff unter: <https://www.youtube.com/watch?v=QU15ID4YKfw>
8. Foto von Alessandro Oliverio von Pexels: <https://www.pexels.com/de-de/foto/silber-und-grun-leiterplatte-1472443/>