

# Projekt-Dokumentation: Ubuntu-Server mit Windows-Integration

---

## Zielsetzung

Ein Ubuntu-Server (Raspberry Pi) wurde eingerichtet, um typische Serverdienste in einem lokalen Netzwerk bereitzustellen. Zusätzlich wurde ein Windows-System eingebunden, um die Zusammenarbeit beider Systeme zu demonstrieren.

## Ubuntu-Server (192.168.1.33)

### Systemkonfiguration

- System mit 'sudo apt update && sudo apt upgrade' aktualisiert
- Benutzer angelegt: Anton, Anna, Berta, Bert
- Gruppen angelegt: agruppe, bgruppe
- Benutzer den Gruppen zugewiesen mit 'usermod -aG'

### Samba-Freigaben

Samba wurde installiert und in der Konfigurationsdatei '/etc/samba/smb.conf' wurden folgende Freigaben eingerichtet:

- /shareall: Für alle Benutzer offen, Gastzugriff erlaubt
- /sharea: Zugriff nur für Benutzer in 'agruppe'
- /shareb: Zugriff nur für Benutzer in 'bgruppe'

Die Freigaben wurden mit entsprechenden Rechten und Gruppenbesitz versehen.

### Webdienste

- Apache installiert mit 'sudo apt install apache2'
- Standard-Webseite unter '/var/www/html/index.html' angepasst
- Webmin als Admin-Oberfläche installiert (<https://192.168.1.33:10000>)
- Cockpit installiert für Systemübersicht (<https://192.168.1.33:9090>)

### Docker & Portainer

- Docker installiert mit 'sudo apt install docker.io'
- Portainer installiert per Docker-Container auf Port 9443 (<https://192.168.1.33:9443>)
- Portainer ermöglicht einfache Verwaltung von Docker-Containern über Weboberfläche

### Dokumentation und Verlauf

- Der komplette Verlauf der Arbeitsschritte wurde gespeichert mit: 'history > ~/setup\_history.txt'

## Windows-VM (192.168.1.43)

### Freigaben

- Auf C:\ wurden die Ordner 'share1' und 'share2' erstellt
- Über 'Erweiterte Freigabe' wurden sie im Netzwerk freigegeben
- Die Firewall wurde konfiguriert, um Datei- und Druckerfreigabe zuzulassen
- Netzwerkprofil wurde auf 'Privat' gestellt, um Freigaben zu ermöglichen

### Einbindung unter Ubuntu

- 'cifs-utils' wurde installiert mit 'sudo apt install cifs-utils'
- /mnt/share1 wurde manuell mit mount -t cifs verbunden
- /mnt/share2 wurde dauerhaft über /etc/fstab eingetragen und gemountet
- Zugangsdaten wurden sicher in /etc/samba/cred\_share2 gespeichert
- Rechte auf 600 gesetzt, um sensible Daten zu schützen

### Zweck und Nutzen

Dieses Setup ermöglicht eine realitätsnahe Lern- und Testumgebung. Es deckt die Bereiche Benutzerverwaltung, Netzwerktechnologien, Datei- und Druckdienste sowie Containerisierung ab. Darüber hinaus dient es als solide Basis für weitere Dienste wie Nextcloud, VPN, Webserver oder Automatisierungslösungen.

Durch die Kombination aus Linux-Serverdiensten und Windows-Integration eignet sich dieses Projekt ideal für IT-Ausbildung, HomeLab-Nutzung, Testumgebungen oder kleine Unternehmensnetzwerke.

## Verwendete Befehle (Command History)

Nachfolgend sind alle relevanten Befehle aufgeführt, die im Rahmen des Projekts verwendet wurden.

- `sudo apt update && sudo apt upgrade -y`
- `sudo adduser anton`
- `sudo groupadd agruppe`
- `sudo usermod -aG agruppe anton`
- `sudo apt install samba -y`
- `sudo nano /etc/samba/smb.conf`
- `sudo mkdir /shareall /sharea /shareb`
- `sudo apt install apache2 -y`
- `sudo nano /var/www/html/index.html`
- `sudo apt install docker.io -y`
- `sudo docker run hello-world`
- `sudo apt install cifs-utils -y`
- `sudo mount -t cifs //192.168.1.43/share1 /mnt/share1 -o username=Installation,password=MeinPasswort`
- `sudo nano /etc/fstab`
- `sudo apt install ./webmin_2.105_all.deb`

## History:



setup\_history.txt

## Troubleshooting

### 1. Raspberry Pi kann Windows-VM nicht erreichen

- **Symptom:**
  - Ping von 192.168.1.33 zu 192.168.1.43 schlägt fehl
  - Mount-Versuche hängen oder schlagen fehl
- **Ursachen & Lösung:**
  - Windows-VM befindet sich im NAT- oder Host-Only-Modus → Umstellen auf 'Extern' (Bridged) in Hyper-V
  - Windows-Netzwerkprofil auf 'Öffentlich' gesetzt → Umstellen auf 'Privat'
  - Firewall blockiert Ports → Datei- und Druckerfreigabe (Port 445) freigeben

### 2. CIFS Mount schlägt fehl mit Fehler 115

- **Fehlermeldung:** mount error(115): Operation now in progress  
Dies tritt auf, wenn das Ziel nicht erreichbar ist (Netzwerkfehler).
- **Lösung:**
  - Prüfen, ob Windows-IP korrekt ist (z. B. mit ipconfig)
  - Firewall & Netzwerk prüfen wie in Punkt 1
  - Testen mit: ping 192.168.1.43 oder nc -zv 192.168.1.43 445

### 3. Webmin oder Cockpit im Browser nicht erreichbar

- **Lösung:**
  - Zugriff über https://IP:10000 (Webmin) oder https://IP:9090 (Cockpit)
  - Firewall mit sudo ufw allow 10000 bzw. 9090 freigeben
  - Port mit ss -tln prüfen, ob Dienst lauscht
  - Browser-Warnung ignorieren (selbsterstelltes Zertifikat)

### 4. Zugriff auf Samba-Freigabe wird verweigert

- **Lösung:**
  - Benutzer ist nicht Mitglied der entsprechenden Gruppe (agruppe, bgruppe)
  - Rechte auf Ordner falsch gesetzt → chown & chmod verwenden
  - In smb.conf prüfen, ob 'valid users' korrekt gesetzt ist

### 5. Webmin Login schlägt fehl

- **Lösung:**
  - Systembenutzer benötigt Passwort: sudo passwd BENUTZERNAME
  - Benutzer muss sudo-Rechte besitzen
  - Zugang nur über HTTPS im Browser möglich