LAP-Prüfungsvorbereitung - Virtuelle Maschinen – LinuX, Windows

PC Konfiguration

- Konfigurieren Sie den PC wie folgt:
  - Computername: F205xx
  - IP-Adresse: 10.2.205.xx / 24
  - Gateway: 10.2.205.254
  - DNS: 10.96.28.35
- Installieren Sie eine Virtualisierungssoftware Ihrer Wahl, z.B. Hyper-V, VMware Player oder VirtualBox.

Virtuelle Maschinen erstellen

Was ist der Vorteil von Virtualisierung?

- Erstellen Sie folgende virtuelle Maschinen. Konfigurieren Sie die Netzwerkkarten der virtuellen Maschinen auf eine Netzwerkbrücke. Alle VMs sollen im Anschluss Zugang zum Internet besitzen.

Windows 11

- 4 CPU, 8192 MB RAM
- IP-Adresse: 10.2.205.1xx/24
- Gateway: 10.2.205.254
- DNS: 10.96.28.35

Ubuntu

- 4 CPU, 4096 MB RAM
- IP-Adresse: 10.2.205.2xx/24
- DNS: 10.96.28.35
- Gateway: 10.2.205.254

Freigaben

- Legen Sie auf der Windows-VM am Desktop eine Freigabe „data" mit Lese/Schreibrechte an. Die Ubuntu-VM soll auf diese Freigabe über den Ordner /mnt/netz zugreifen können.
- Legen Sie auf der Ubuntu VM am Desktop eine Freigabe „data" mit Lese/Schreibrechte an. Verwenden Sie dazu den Samba-Dienst. Die Windows VM soll auf dese Freigabe über das Laufwerk F:\ zugreifen können.

Fernwartung und Drucker

- Installieren bzw. aktivieren Sie auf beiden VMs einen Fernwartungsdienst (z.B. RDP oder VNC) und testen Sie den Zugriff von der jeweils anderen VM.
- Installieren Sie den Netzwerkdrucker in jeder VM. IP-Adresse des Druckers: 10.2.205.50

VNC- sudo apt install gnome-remote-desktop
reboot

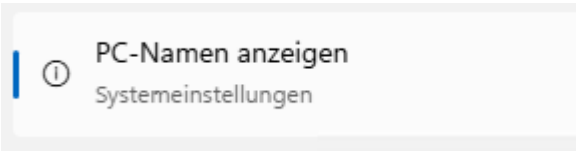
Wartungstools

- Installieren Sie auf der Ubuntu VM den Dienst „Webmin" und machen Sie sich damit vertraut.
  - Dashboard
  - Verfügbarer Speicher
  - Verfügbare Updates
  - Automatische Jobs

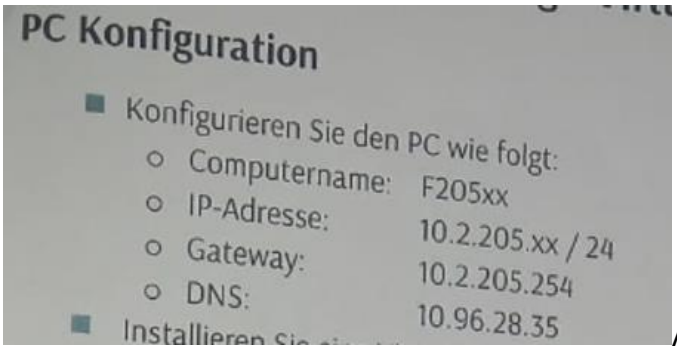
Cron Job erstellen

- Erstelle ein Shell-Skript „uptimelog.sh"mit folgenden Inhalt:
  echo "$(date) - VM ist online" >> /mnt/netz/uptimelog.txt
- Erstelle einen Cron Job, welcher das Skript jede Minute ausführt. Überprüfe die Funktion, indem du kontrollierst, ob die Datei auf der Windows VM im Ordner „data" erstellt wurde und aktualisiert wird.

PC-Namen wo die VMs unter Hyper V laufen den Computernamen unter



umbenennen.

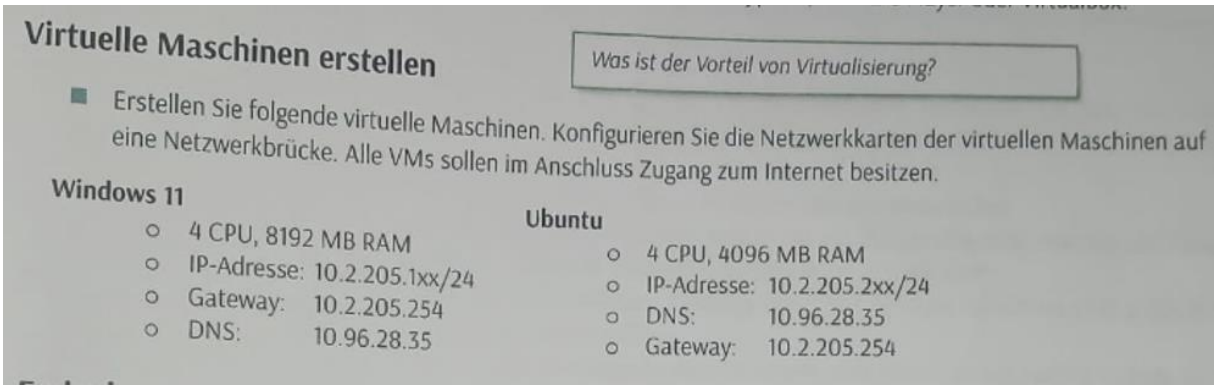


Adressen in der LAP so einstellen statt den xx wird die Nummer auf den Bildschirmen angewendet.

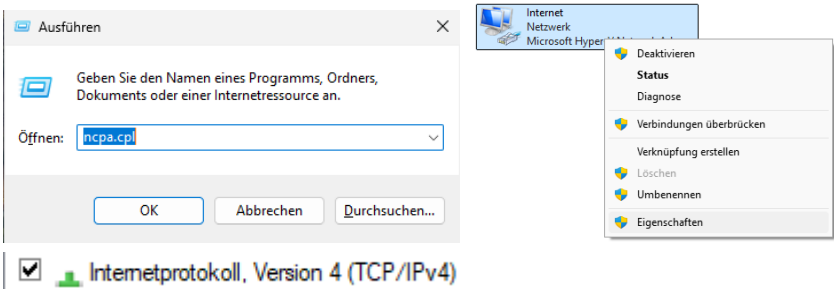


Privat einstellen

Dann diese VMs erstellen



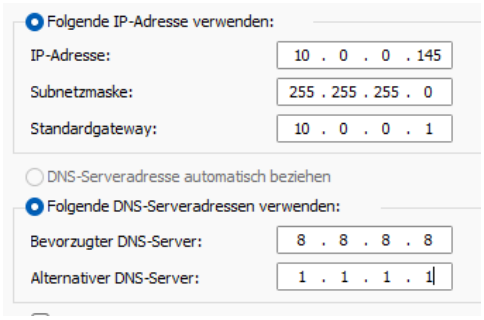
Mit diesem Befehl Netzwerkadapter einstellen



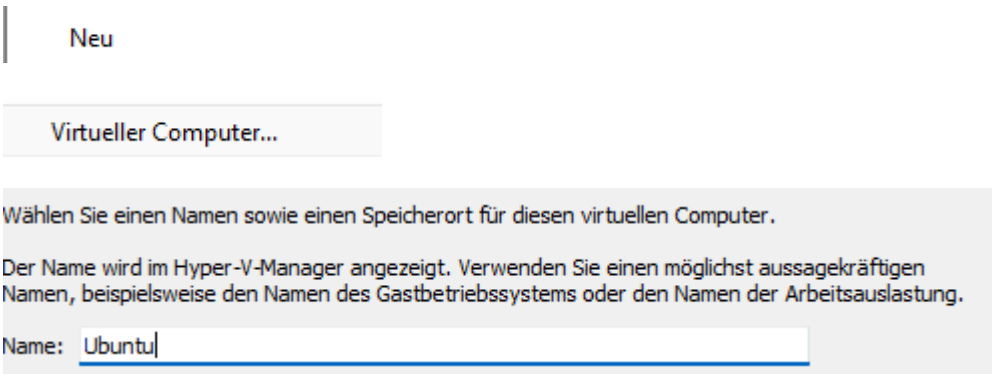

```
Ethernet-Adapter Internet:

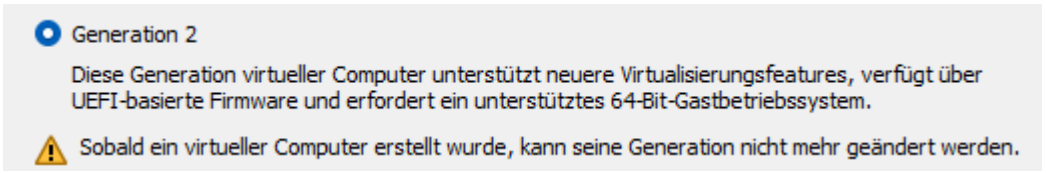
   Verbindungsspezifisches DNS-Suffix: fritz.box
   IPv6-Adresse. . . . . . . . . . . : fd2a:e6b1:5a9:0:d168:5721:2480:b578
   Temporäre IPv6-Adresse. . . . . . : fd2a:e6b1:5a9:0:790d:8b5f:352b:b2c1
   Verbindungslokale IPv6-Adresse  . : fe80::8d0:c723:18c4:643a%11
   IPv4-Adresse  . . . . . . . . . . : 10.0.0.145
   Subnetzmaske  . . . . . . . . . . : 255.255.255.0
   Standardgateway . . . . . . . . . : 10.0.0.1
```

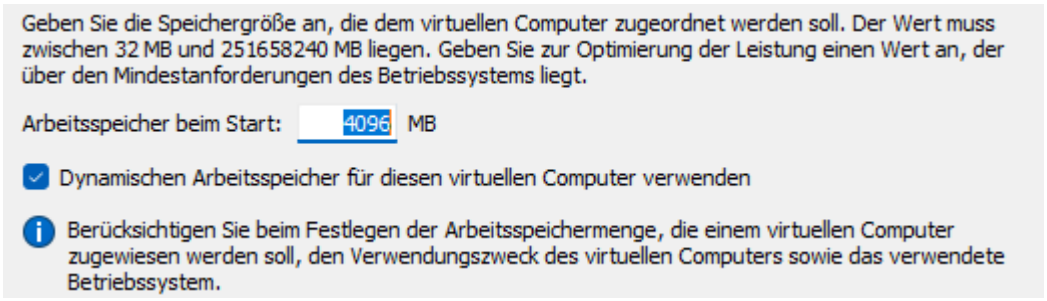
Es wird in der LAP die IP eingegeben, die laut dem Zettel vorgegeben ist und in meiner Labor Umgebung nehme ich die Adresse, die ich mit dem DHCP gezogen habe und stelle sie auf statisch. Wie hier.

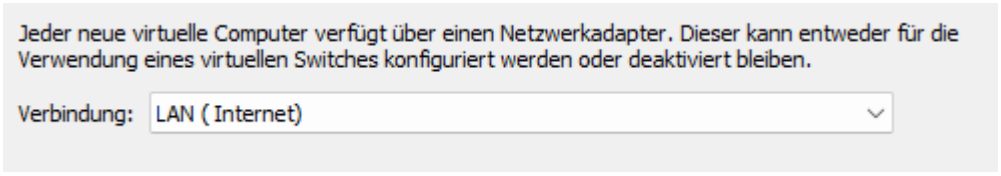


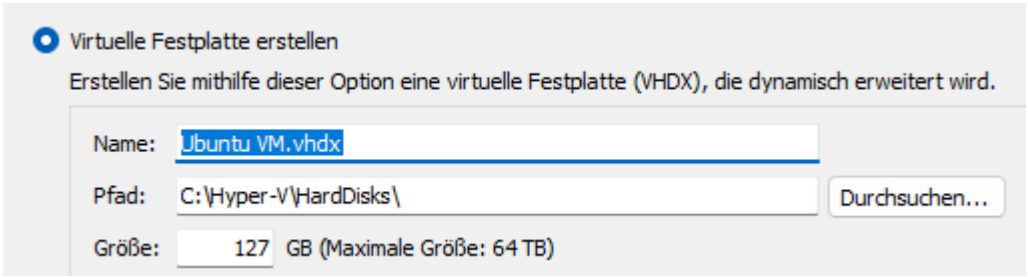
Ubuntu VM erstellen unter Hyper V

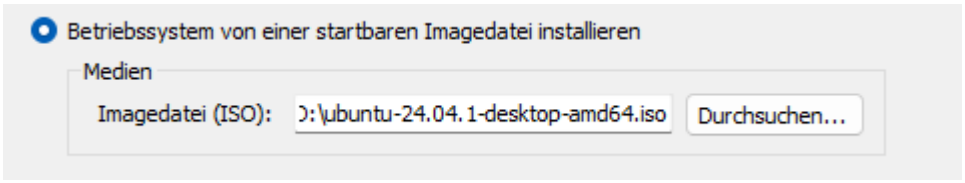


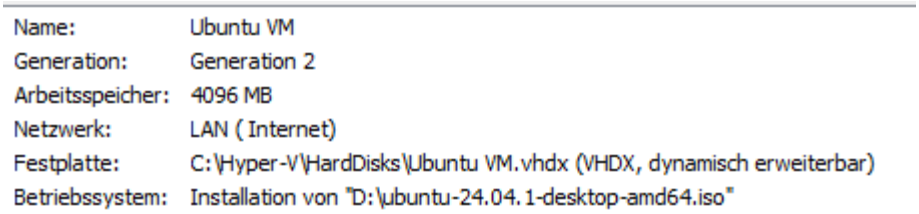

Generation 2
Diese Generation virtueller Computer unterstützt neuere Virtualisierungsfeatures, verfügt über UEFI-basierte Firmware und erfordert ein unterstütztes 64-Bit-Gastbetriebssystem.
Sobald ein virtueller Computer erstellt wurde, kann seine Generation nicht mehr geändert werden.

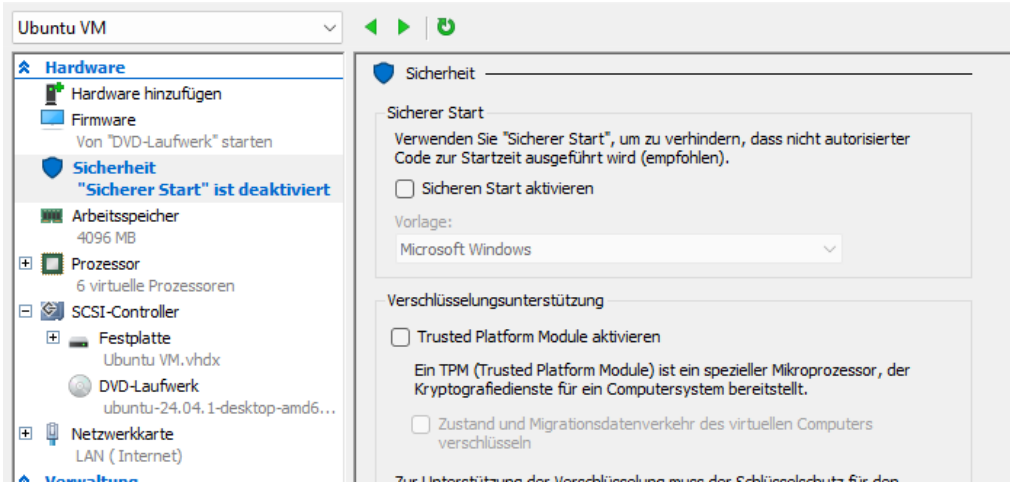

Geben Sie die Speichergröße an, die dem virtuellen Computer zugeordnet werden soll. Der Wert muss zwischen 32 MB und 251658240 MB liegen. Geben Sie zur Optimierung der Leistung einen Wert an, der über den Mindestanforderungen des Betriebssystems liegt.
Arbeitsspeicher beim Start: 4096 MB
Dynamischen Arbeitsspeicher für diesen virtuellen Computer verwenden
Berücksichtigen Sie beim Festlegen der Arbeitsspeichermenge, die einem virtuellen Computer zugewiesen werden soll, den Verwendungszweck des virtuellen Computers sowie das verwendete Betriebssystem.


Jeder neue virtuelle Computer verfügt über einen Netzwerkadapter. Dieser kann entweder für die Verwendung eines virtuellen Switches konfiguriert werden oder deaktiviert bleiben.
Verbindung: LAN (Internet)

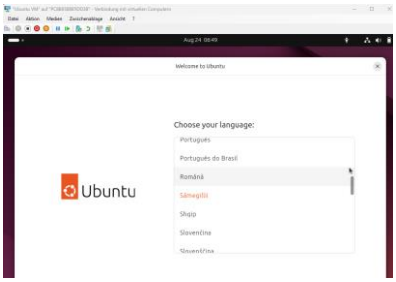

Virtuelle Festplatte erstellen
Erstellen Sie mithilfe dieser Option eine virtuelle Festplatte (VHDX), die dynamisch erweitert wird.
Name: Ubuntu VM.vhdx
Pfad: C:\Hyper-V\HardDisks\
Durchsuchen...
Größe: 127 GB (Maximale Größe: 64 TB)


Betriebssystem von einer startbaren Imagedatei installieren
Medien
Imagedatei (ISO): D:\ubuntu-24.04.1-desktop-amd64.iso
Durchsuchen...

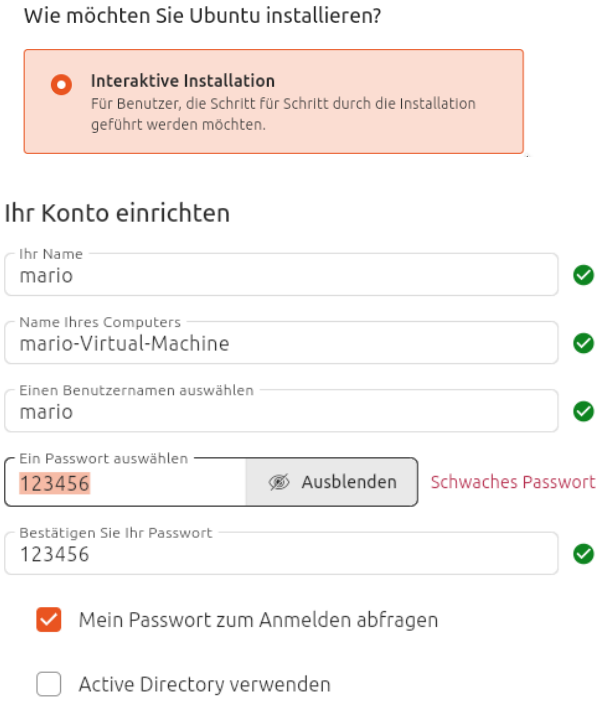

Name: Ubuntu VM
Generation: Generation 2
Arbeitsspeicher: 4096 MB
Netzwerk: LAN (Internet)
Festplatte: C:\Hyper-V\HardDisks\Ubuntu VM.vhdx (VHDX, dynamisch erweiterbar)
Betriebssystem: Installation von "D:\ubuntu-24.04.1-desktop-amd64.iso"



hier Sicherer Start Haken entfernen, sonst startet VM nicht!!!



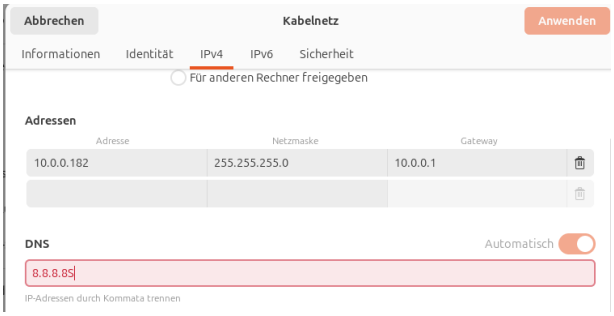
gesamte Abfolge hier bestätigen





diesen Wert übernehmen und statisch stellen

Hier in meiner Umgebung wieder die vom DHCP übernehme (statisch)

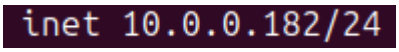


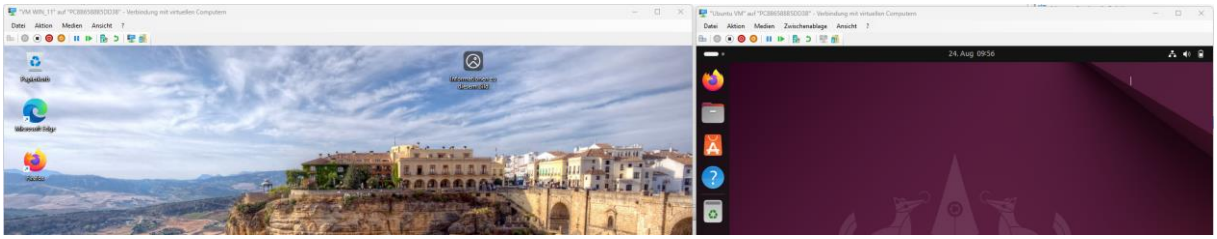
Beide VMs gegenseitig pingen testen

WIN11



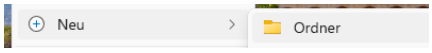
auf Ubuntu VM





Beide VMs Internet und pingen erfolgreich

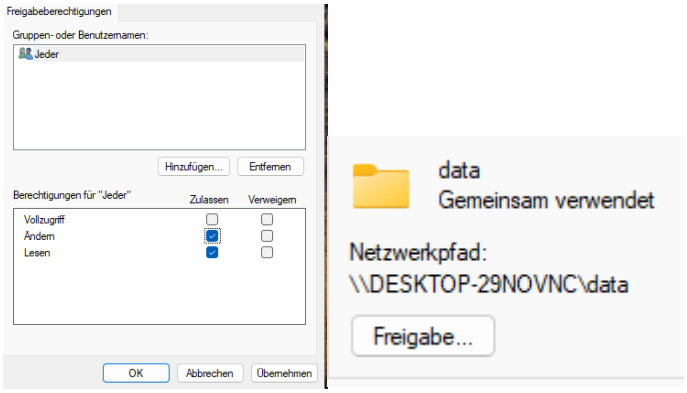
Ordner erstellen auf WIN11





Ordner freigeben

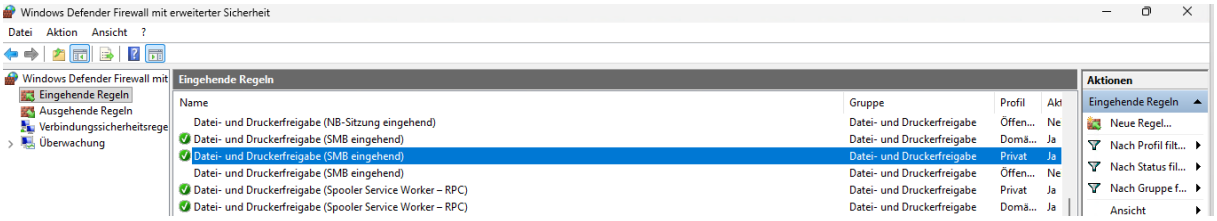




\\DESKTOP-29NOVNC\data

Windows-Firewall für Dateifreigabe

Muss aktiviert sein

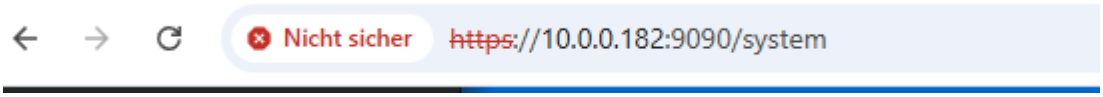


Ubuntu: Samba/CIFS installieren

sudo apt update

sudo apt install cifs-utils mit Yes bestätigen

Möglich mit Cockpit zuzugreifen: sudo apt install cockpit



Ubuntu-Verzeichnis /mnt/netz erstellen

sudo mkdir -p /mnt/netz

Jetzt versuchen wir zuerst, die Windows-Freigabe einzubinden

IP: WIn 11 = 10.0.0.145

sudo mount -t cifs //10.0.0.145/data /mnt/netz -o username=CLIENT01

dann Passwort Abfrage 123456

Testdatei auf Ubuntu erstellen

sudo touch /mnt/netz/test.txt

ls -l /mnt/netz jetzt ist auf Windows eine Textdatei ersichtlich

Ubuntu → Windows: Samba installieren

sudo apt install samba + Y bestätigen

Ubuntu-Ordner data erstellen

Auf dem Desktop direkt oder mit:

mkdir -p ~/Desktop/data

Samba-Benutzer erstellen

whoami
sudo smbpasswd -a mario Add
Danach:
sudo smbpasswd -e mario enable

Samba-Freigabe konfigurieren

sudo nano /etc/samba/smb.conf

```
[data]
  path = /home/mario/Desktop/data
  browseable = yes
  read only = no
  guest ok = no
```

STG +O + Enter + STG +X

Samba-Konfiguration prüfen

Testparm

Dann Samba neu starten:

sudo systemctl restart smbd

sudo systemctl enable smbd

Ubuntu-Firewall

Falls Ubuntu ufw verwendet:

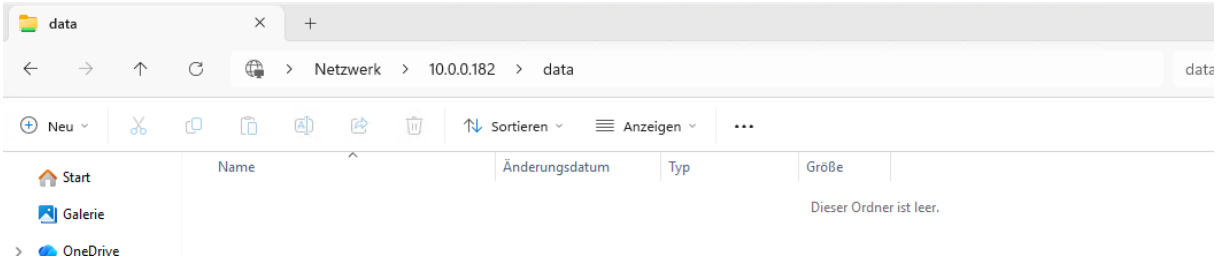
sudo ufw allow samba

Ubuntu-IP herausfinden

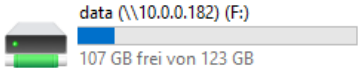
```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:09:9e:1e brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.182/24 brd 10.0.0.255 scope global noprefixroute eth0
```

Windows: Ubuntu-Samba-Freigabe testen:

\\10.0.0.182\data Ordner wird geöffnet



Ubuntu-Freigabe als Laufwerk F: verbinden



Das verlangt die Aufgabe ausdrücklich.

Auf Windows:

Dieser PC

→ oben:

… → **Netzlaufwerk verbinden**

Laufwerk:

```
F:
```

Ordner:

```
\\10.2.205.2xx\data
```

Haken setzen bei:

Verbindung bei Anmeldung wiederherstellen

und:

Verbindung mit anderen Anmeldeinformationen herstellen

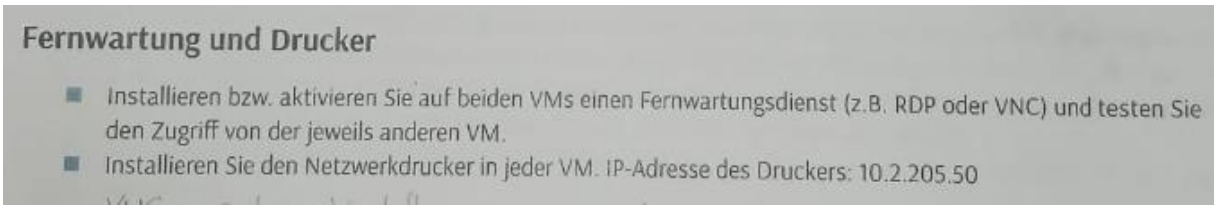
Dann **Fertig stellen.**

wird angezeigt unter dieser PC

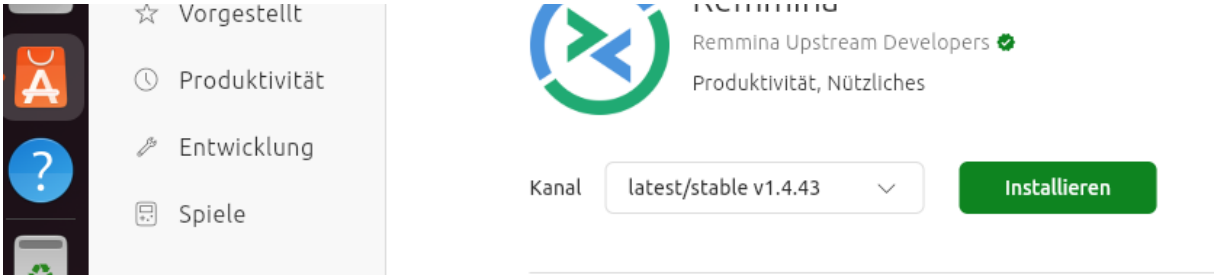
dann eine Test Text Datei erstellen und auf Vm Ubuntu kontrollieren



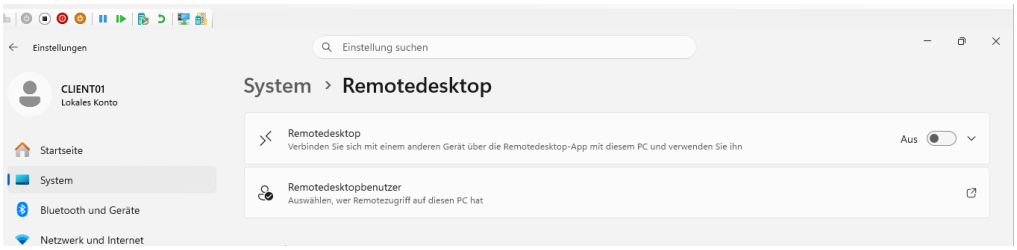

```
mario@mario-Virtual-Machine:~$ ls ~/Desktop/data
Windows-Test.txt.txt
```



Unter Anwendungzentrum Remmina installieren



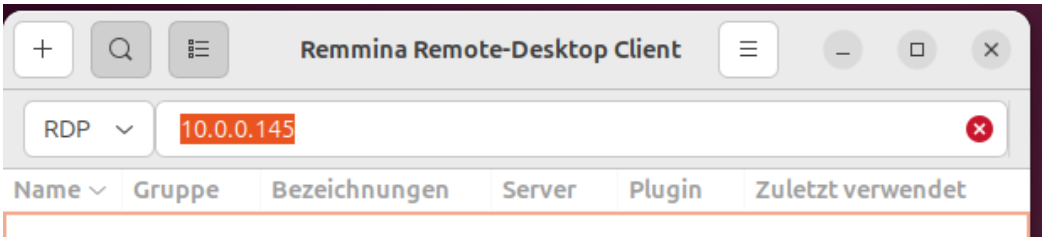
Unter Ubuntu



Unter WIN 11 hier aktivieren

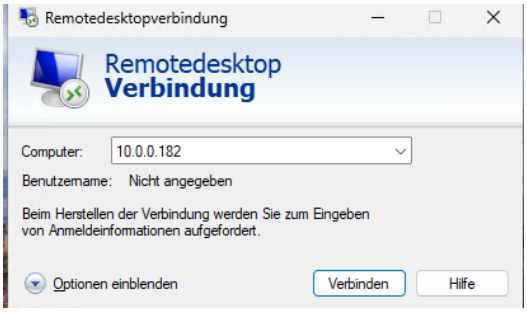
Zugriff testen

- **Von Ubuntu auf Windows**: Die App *Remmina* auf Ubuntu öffnen, Protokoll „RDP“ wählen und die IP der Windows-VM (`10.2.205.1xx`) eingeben.

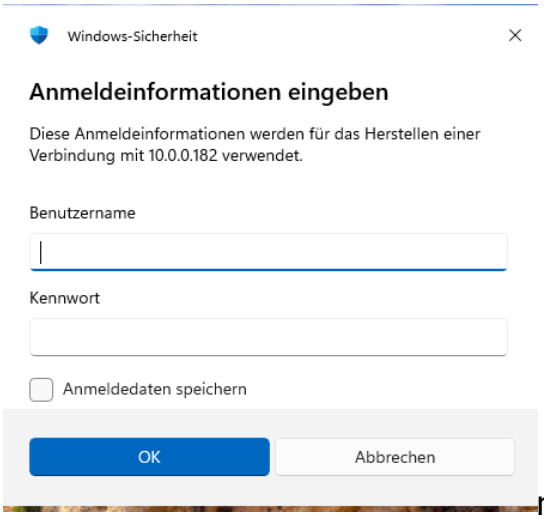


Zugriff funktioniert von Ubuntu auf WIN VM

Umgekehrt

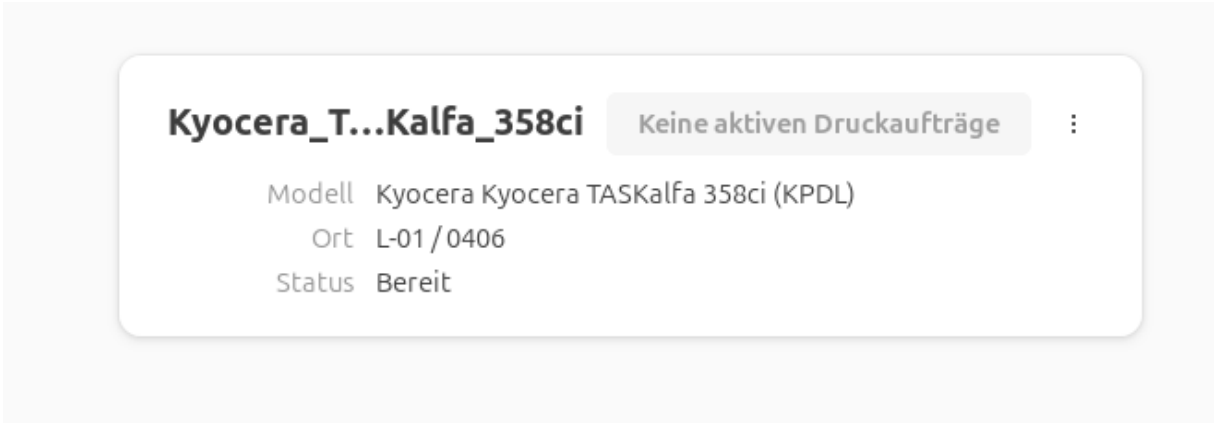


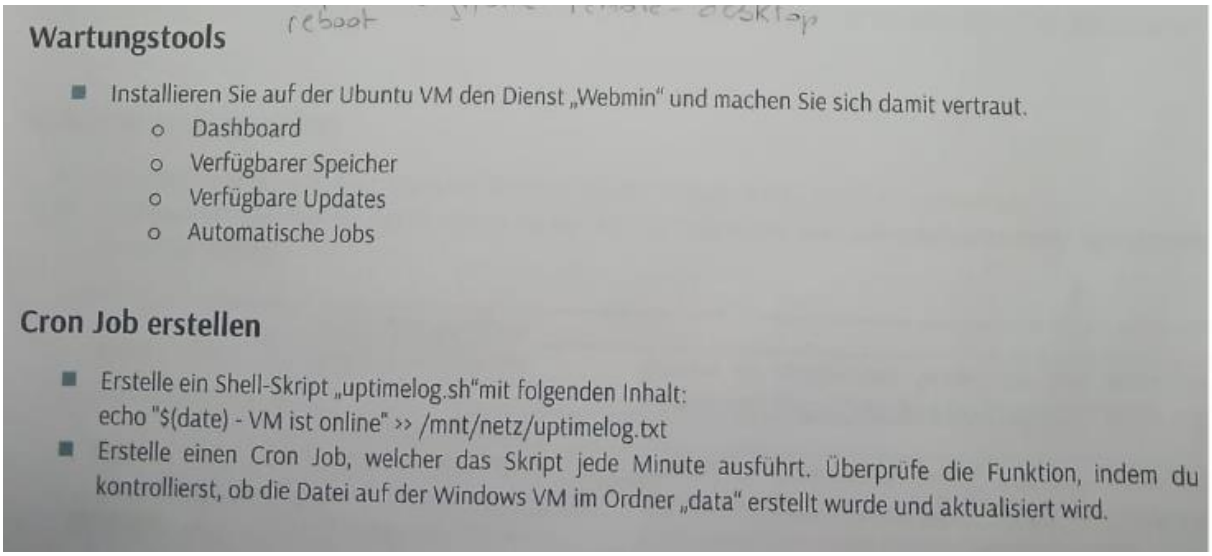
Ubuntu auf abmelden gehen



mario + 123456

Drucker hinzugefügt





sudo apt update

sudo apt install curl -y

curl -o webmin-setup-repo.sh https://raw.githubusercontent.com/webmin/webmin/master/webmin-setup-repo.sh

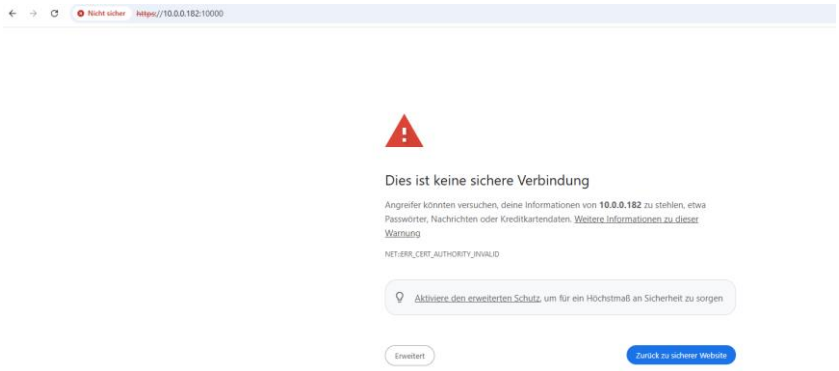
sudo apt install webmin --install-recommends -y

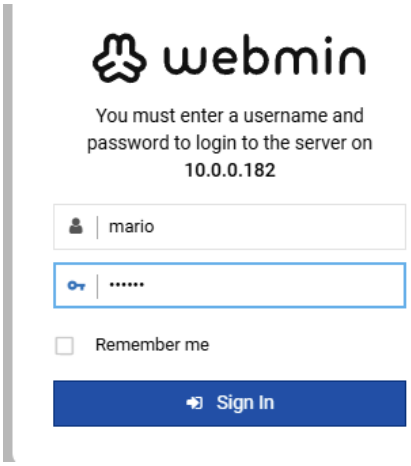
Prüfen

sudo systemctl status Webmin

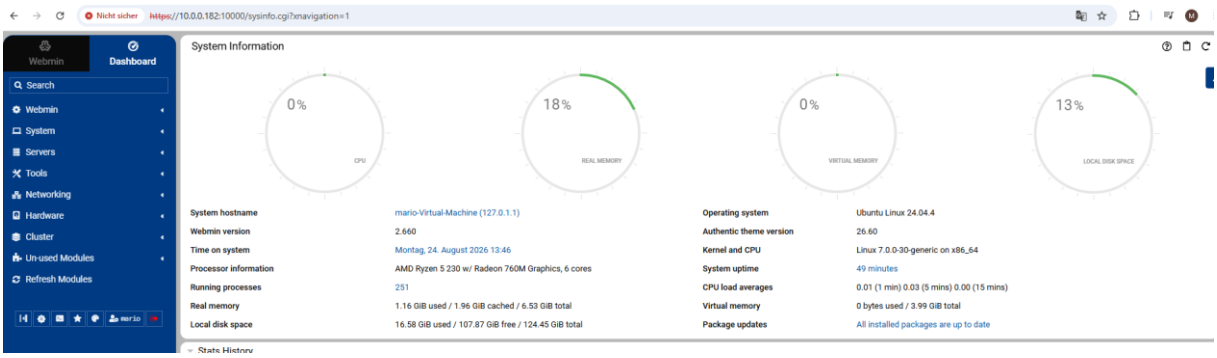
hostname -I : 10.0.0.182

https://10.0.0.182:10000/





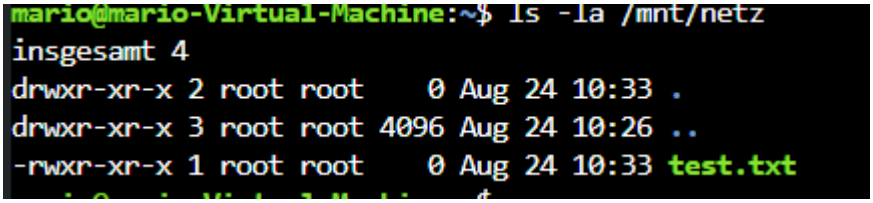
mario + 123456



Windows-Freigabe einhängen (Mounten)

sudo mount -t cifs -o username=CLIENT01,password=123456 //10.0.0.145/data /mnt/netz

ls -la /mnt/netz Abfrage



richtig gemountet

Cron Job erstellen

sudo nano /usr/local/bin/uptimelog.sh

Füge im Texteditor **exakt** diesen Inhalt ein:

#!/bin/bash

echo "$(date) - VM ist online" >> /mnt/netz/uptimelog.txt

Skript ausführbar machen

sudo chmod +x /usr/local/bin/uptimelog.sh

Cron Job einrichten

sudo crontab -e (Tabelle ganz nach unten und einfügen)

* * * * * /usr/local/bin/uptimelog.sh




```
text
*      *      *      *      *     /usr/local/bin/uptimelog.sh
|      |      |      |      |
|      |      |      |      └─ Wochentag (0-7) -> * bedeutet: jeden Tag
|      |      |      └─ Monat (1-12)     -> * bedeutet: jeden Monat
|      |      └─ Tag (1-31)       -> * bedeutet: jeden Tag
|      └─ Stunde (0-23)    -> * bedeutet: jede Stunde
└─ Minute (0-59)    -> * bedeutet: jede Minute
```