

# Storage Box – SMB Share versleutelen met gocryptfs

Met deze configuratie wordt een **onversleutelde SMB-share** van een Storage Box gebruikt als opslag voor een **versleutelde gocryptfs-container**.

De structuur wordt:

```
Storage Box
└─ SMB Share
    └─ encrypted/          ← Versleutelde bestanden
        └─ ...             ← Niet rechtstreeks openen/bewerken
            └─ ...
/mnt/decrypted/          ← Normale, leesbare bestanden
```

Je werkt dus altijd met:

```
/mnt/decrypted
```

De bestanden die daadwerkelijk op de Storage Box staan, bevinden zich in:

```
/mnt/smb/encrypted
```

en zijn daar **versleuteld en onleesbaar zonder de gocryptfs-sleutel**.

## 1. Benodigde pakketten installeren

Installeer `cifs-utils` voor SMB en `gocryptfs` voor de encryptie:

```
sudo apt update  
sudo apt install cifs-utils gocryptfs -y
```

## 2. Mountpoints aanmaken

Maak de directories aan waarop de SMB-share en de gedecrypteerde bestanden worden gemount:

```
sudo mkdir -p /mnt/smb  
sudo mkdir -p /mnt/decrypted
```

## 3. SMB-inloggegevens opslaan

Maak een apart bestand voor de inloggegevens van de Storage Box:

```
sudo nano /etc/backup-credentials.txt
```

Zet hierin:

```
username=u474777  
password=PASSWORD
```

Vervang `PASSWORD` uiteraard door het daadwerkelijke wachtwoord.

Sla het bestand op:

```
CTRL + X  
Y  
Enter
```

Zorg vervolgens dat alleen `root` het bestand kan lezen:

```
sudo chmod 600 /etc/backup-credentials.txt
```



**Belangrijk:** dit bestand bevat het wachtwoord van de Storage Box. Deel het niet en zorg dat de permissies `600` blijven.

## 4. SMB-share automatisch mounten

Maak een systemd mount-unit:

```
sudo nano /etc/systemd/system/mnt-smb.mount
```

Plaats hierin:

```
[Unit]
Description=Mount SMB share
Wants=network-online.target
After=network-online.target

[Mount]
What=//u474777.your-storagebox.de/backup
Where=/mnt/smb
Type=cifs
Options=vers=3.1.1,credentials=/etc/backup-
credentials.txt,uid=0,gid=0,file_mode=0755,dir_mode=0755,nofail
TimeoutSec=30

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

Sla het bestand op:

```
CTRL + X
Y
Enter
```

# systemd opnieuw laden

```
sudo systemctl daemon-reload
```

Zorg dat de netwerk-wachtservice actief is:

```
sudo systemctl enable systemd-networkd-wait-online.service  
sudo systemctl start systemd-networkd-wait-online.service
```

Activeer vervolgens de SMB-mount:

```
sudo systemctl enable mnt-smb.mount  
sudo systemctl start mnt-smb.mount
```

Controleer de status:

```
systemctl status mnt-smb.mount
```

Als alles goed staat, moet de SMB-share beschikbaar zijn via:

```
/mnt/smb
```

Je kunt dit eventueel controleren met:

```
ls -la /mnt/smb
```

---

## 5. Versleutelde directory aanmaken

Maak binnen de SMB-share een directory voor de versleutelde opslag:

```
sudo mkdir -p /mnt/smb/encrypted
```



**Let op:** de directory `encrypted` bevat straks de versleutelde data. Bewerk de bestanden hierin **niet rechtstreeks**.

## 6. gocryptfs initialiseren

Initialiseer gocryptfs:

```
sudo gocryptfs -init /mnt/smb/encrypted
```

gocryptfs vraagt nu om een wachtwoord.

Dit wachtwoord is belangrijk: **zonder dit wachtwoord kan de versleutelde data niet worden ontsleuteld.**

Bewaar het wachtwoord daarom veilig.

## 7. gocryptfs-wachtwoord opslaan

Om de decrypted mount automatisch bij het opstarten te kunnen mounten, slaan we het wachtwoord op in een beveiligd bestand.

Maak het bestand:

```
sudo nano /root/.gocryptfs-key
```

Zet daarin het gocryptfs-wachtwoord:

```
gocryptfs-PASSWORD
```

Sla op:

```
CTRL + X  
Y
```

Enter

Beperk de toegang tot het bestand:

```
sudo chmod 600 /root/.gocryptfs-key
```

“ **Belangrijk:** dit bestand bevat de sleutel/wachtwoord waarmee de versleutelde Storage Box kan worden geopend. Zorg dat hiervan een veilige back-up bestaat. Als zowel de data als deze sleutel verloren gaan, kan de data niet meer worden geopend.

## 8. Decrypted mount aanmaken

Nu maken we een tweede systemd mount-unit.

Deze mount zorgt ervoor dat de versleutelde directory:

```
/mnt/smb/encrypted
```

zichtbaar wordt als normale, leesbare bestanden via:

```
/mnt/decrypted
```

Maak de unit aan:

```
sudo nano /etc/systemd/system/mnt-decrypted.mount
```

Plaats hierin:

```
[Unit]
Description=Mount gocryptfs decrypted folder
Requires=mnt-smb.mount
After=mnt-smb.mount network-online.target local-fs.target
```

```
[Mount]
What=/mnt/smb/encrypted
Where=/mnt/decrypted
Type=fuse.gocryptfs
Options=passfile=/root/.gocryptfs-key,allow_other,nonempty
TimeoutSec=30

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

Sla op:

```
CTRL + X
Y
Enter
```

## 9. Decrypted mount activeren

Laad systemd opnieuw:

```
sudo systemctl daemon-reload
```

Activeer de mount bij het opstarten:

```
sudo systemctl enable mnt-decrypted.mount
```

Start de mount:

```
sudo systemctl start mnt-decrypted.mount
```

Controleer de status:

```
systemctl status mnt-decrypted.mount
```

Als alles goed werkt, zijn de bestanden beschikbaar via:

```
/mnt/decrypted
```

---

## 10. Controleren of alles werkt

Controleer eerst de SMB-mount:

```
sudo systemctl status mnt-smb.mount
```

Controleer daarna de gocryptfs-mount:

```
sudo systemctl status mnt-decrypted.mount
```

Controleer of de decrypted directory toegankelijk is:

```
ls -la /mnt/decrypted
```

Maak eventueel een testbestand:

```
sudo touch /mnt/decrypted/test.txt
```

Controleer vervolgens of het bestand zichtbaar is:

```
ls -la /mnt/decrypted
```

Het bestand moet ook in de versleutelde opslag aanwezig zijn, maar **niet als leesbare** `test.txt`. Op de SMB-share zal gocryptfs hiervoor een versleutelde bestandsnaam en versleutelde inhoud gebruiken.

Verwijder het testbestand daarna:

```
sudo rm /mnt/decrypted/test.txt
```

---

## 11. Test na een reboot

Een reboot is belangrijk om te controleren of beide mounts automatisch correct worden opgebouwd:

```
sudo reboot
```

Na het opnieuw opstarten:

## Netwerk

```
sudo systemctl status systemd-networkd-wait-online.service
```

## SMB

```
sudo systemctl status mnt-smb.mount
```

## gocryptfs

```
sudo systemctl status mnt-decrypted.mount
```

## Bestanden controleren

```
ls -la /mnt/decrypted
```

---

# 12. Logs bekijken bij problemen

Als de SMB-mount niet werkt:

```
journalctl -u mnt-smb.mount
```

Als de gocryptfs-mount niet werkt:

```
journalctl -u mnt-decryptd.mount
```

Voor uitgebreidere informatie kun je ook gebruiken:

```
journalctl -u mnt-smb.mount -b
```

en:

```
journalctl -u mnt-decryptd.mount -b
```

De optie `-b` toont alleen de meldingen van de huidige boot.

## 13. Belangrijke structuur

Na de configuratie zijn er twee belangrijke locaties:

| Locatie                         | Functie                       |
|---------------------------------|-------------------------------|
| <code>/mnt/smb</code>           | De originele SMB-share        |
| <code>/mnt/smb/encrypted</code> | Versleutelde gocryptfs-opslag |
| <code>/mnt/decrypted</code>     | Normale leesbare bestanden    |

**Gebruik voor normale bestanden altijd:**

```
/mnt/decrypted
```

**Gebruik `/mnt/smb/encrypted` niet om bestanden handmatig te wijzigen.**

De encryptie wordt door gocryptfs geregeld.

## 14. Hoe de encryptie werkt

De Storage Box zelf weet niets van gocryptfs en ziet alleen bestanden in:

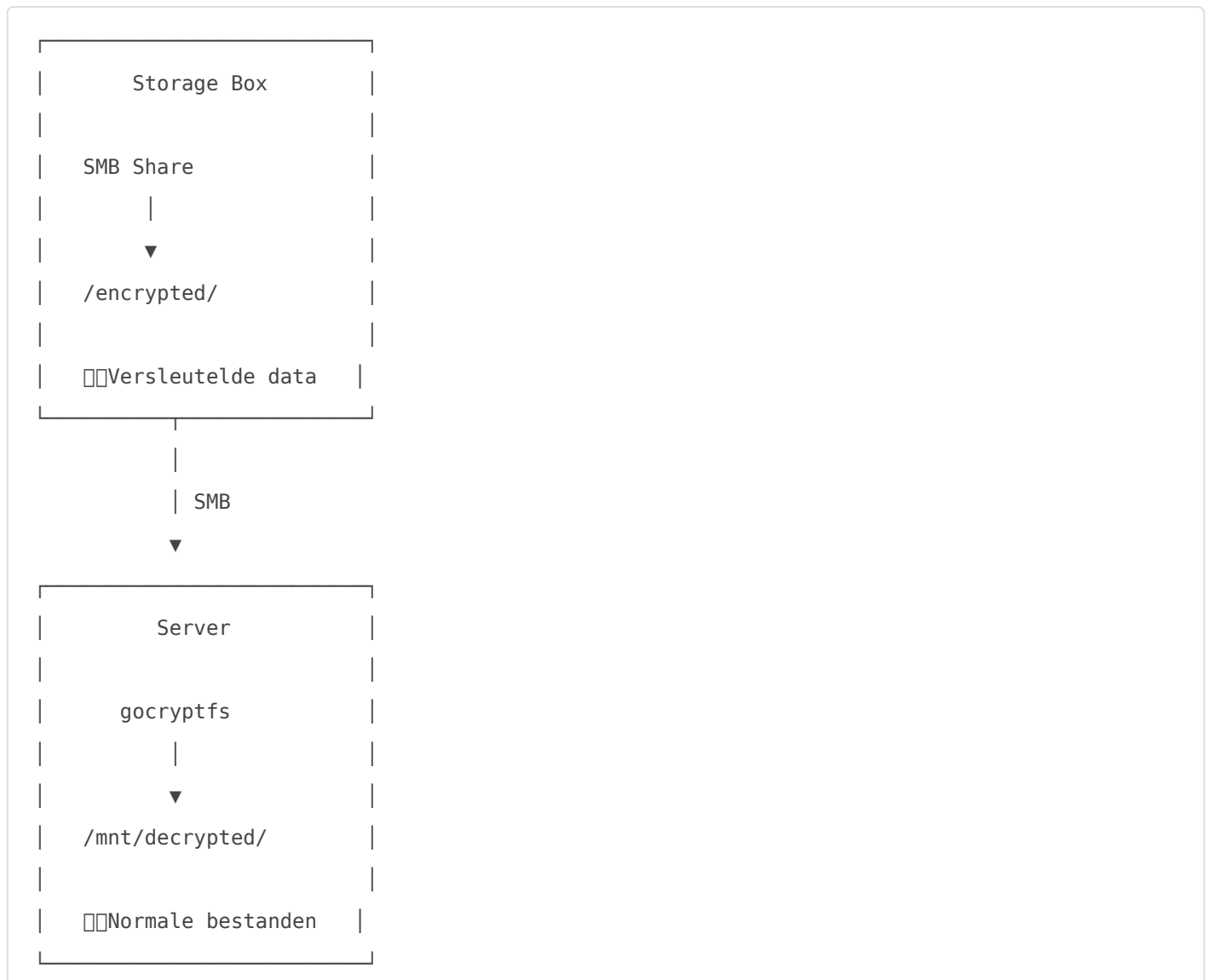
```
/mnt/smb/encrypted
```

De server waarop gocryptfs draait, ontsleutelt de bestanden wanneer ze via:

/mnt/decrypted

worden benaderd.

Het resultaat is:



Hierdoor staat de data **versleuteld op de Storage Box**, terwijl applicaties op de server gewoon met normale bestanden kunnen werken.

# 15. Wat je absoluut moet bewaren

Bewaar veilig:

1. Het **Storage Box-wachtwoord**
2. Het **gocryptfs-wachtwoord**
3. Bij voorkeur ook de volledige gocryptfs-configuratie/back-up

Het belangrijkste is het **gocryptfs-wachtwoord**. Zonder dit wachtwoord kan de versleutelde data niet worden ontsleuteld.

---

Created 2026-09-04 18:21:44 UTC by Robbert

Updated 2026-09-04 18:22:33 UTC by Robbert