

# Funktionen von rsync

**rsync** ist ein leistungsfähiges und vielseitiges Tool, das hauptsächlich für die Dateiübertragung und -synchronisation verwendet wird. Es bietet mehrere nützliche Funktionen:

## Hauptfunktionen

1. **Effiziente Synchronisation:** rsync überträgt nur die Differenzen zwischen Quell- und Zielfilen. Dadurch werden Bandbreite und Zeit gespart.
2. **Kopieren über Netzwerke:** Es unterstützt Übertragungen über SSH, was eine sichere Übertragung ermöglicht.
3. **Datei- und Verzeichnisrekursion:** rsync kann ganze Verzeichnisbäume rekursiv kopieren, einschließlich von Unterverzeichnissen.
4. **Optionale Dateiliste:** Mit der `--exclude`-Option können bestimmte Dateien oder Verzeichnisse von der Synchronisation ausgeschlossen werden.
5. **Delta-Übertragungen:** Statt ganze Dateien zu übertragen, werden nur die geänderten Bits innerhalb der Dateien übertragen.
6. **Hardlinks und symbolische Links:** rsync kann auch mit symbolischen Links und Hardlinks umgehen, was die Datei-Integrität bewahrt.
7. **Checksummen:** Es prüft die Integrität der Dateien während der Übertragung mithilfe von Prüfziffern.

## Verwendung für Backups

Um **rsync** effektiv für Backups auf einem Debian-System zu nutzen, sollten einige Tipps beachtet werden:

---

## Schritte zur Nutzung von rsync für Backups

### 1. Basis-Befehl

Ein einfacher Befehl zum Sichern eines Verzeichnisses könnte folgendermaßen aussehen:

```
bash
```

```
rsync -avz /Pfad/zum/Quellverzeichnis/ /Pfad/zum/Zielverzeichnis/
```

- **-a:** Archivmodus, behält Rechte, Zeitstempel und symbolische Links bei.
- **-v:** Verbose-Modus, zeigt den Fortschritt an.
- **-z:** Komprimiert die Daten während der Übertragung.

### 2. Exklusion von Dateien

Wenn du bestimmte Dateien oder Verzeichnisse vom Backup ausschließen möchtest, kannst du die `--exclude`-Option verwenden:

```
bash
```

```
rsync -avz --exclude='*.tmp' /Pfad/zum/Quellverzeichnis/ /Pfad/zum/Zielverzeichnis/
```

### 3. Nutzung über SSH

Um eine sichere Verbindung zu einem Remote-Server zu nutzen, füge die SSH-Option hinzu:

```
bash
```

```
rsync -avz -e ssh /Pfad/zum/Quellverzeichnis/ benutzer@remote-
server:/Pfad/zum/Zielverzeichnis/
```

### 4. Bestätigungen und Berichte

Nutze die `--dry-run`-Option, um zu prüfen, was geschehen würde, ohne tatsächlich Änderungen vorzunehmen:

```
bash
```

```
rsync -avz --dry-run /Pfad/zum/Quellverzeichnis/ /Pfad/zum/Zielverzeichnis/
```

### 5. Cron-Jobs für automatisierte Backups

Um regelmäßig Backups durchzuführen, kann ein Cron-Job eingerichtet werden. Editiere die Crontab mit:

```
bash
```

```
crontab -e
```

Füge einen Eintrag hinzu, der das Backup beispielsweise täglich um 2 Uhr morgens ausführt:

```
bash
```

```
0 2 * * * rsync -avz /Pfad/zum/Quellverzeichnis/ /Pfad/zum/Zielverzeichnis/
```

---

**Zusammenfassung:** rsync ist ein äußerst effektives Tool für die Dateisynchronisation und -sicherung. Durch seine flexiblen Optionen und die Möglichkeit, nur geänderte Daten zu übertragen, ist es ideal für regelmäßige Backups geeignet. Durch die Kombination mit Cron-Jobs kannst du die Ausführung von Backups automatisieren und sicherstellen, dass deine Daten immer geschützt sind.