

Wie man rsync für Backups verwendet

Erläuterung eines leistungsstarken Dateiübertragungstools

Willem L. Middelkoop

Oct. 30, 2020

```
willem@WillemWindowsSurfaceLaptopGo:~$ rsync -arvz /mnt/c/Users/mail/Desktop/Sample-files/ willem@store.willem.com:./Backup
The authenticity of host 'store.willem.com (82.94.218.172)' can't be established.
ECDSA key fingerprint is SHA256:Oy8iUpwd0V3xy9LLXNdFSPNjizk0B9wA1a6BRwGQRiA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'store.willem.com,82.94.218.172' (ECDSA) to the list of known hosts.
willem@store.willem.com's password:
sending incremental file list
./
Tudor-Black-Bay-36-on-the-wrist_960px.jpg
omata-one.jpg
riding-your-bike.jpg
zwemdiploma-a.jpg

sent 425,822 bytes  received 95 bytes  40,563.52 bytes/sec
total size is 427,915  speedup is 1.00
willem@WillemWindowsSurfaceLaptopGo:~$
```

omata-one riding-your... zwemdiplom Tudor-Blac...

Es ist sehr wichtig, Backups Ihrer Daten zu erstellen, da man nie weiß, wann eine Katastrophe eintritt! Ein leistungsstarkes, plattformübergreifendes Tool, das Ihnen dabei hilft, ist „rsync“. In diesem Beitrag erkläre ich, warum rsync nützlich ist und wie Sie damit Ihre eigenen Backups einrichten können.

Rsync

Seit seiner Entstehung im Jahr 1996 (durch Andrew Tridgell und Paul Mackerras) hat sich rsync zu einem Standard-Dienstprogramm entwickelt, das auf gängigen GNU/Linux-Distributionen, Apple macOS, FreeBSD, NetBSD, OpenBSD und (über Cygwin oder WSL) auf Windows verfügbar ist.

Hauptmerkmale

Rsync wird typischerweise zur Synchronisierung von Dateien und Verzeichnissen zwischen zwei verschiedenen Computern verwendet. Es kann SSH verwenden, um sich mit einem anderen Computer zu verbinden und Dateien zu übertragen. Zusätzlich zu entfernten Computern können Sie rsync verwenden, um mit externen Festplatten und Netzwerkspeichern (NAS) zu synchronisieren. Rsync ist beliebt, weil:

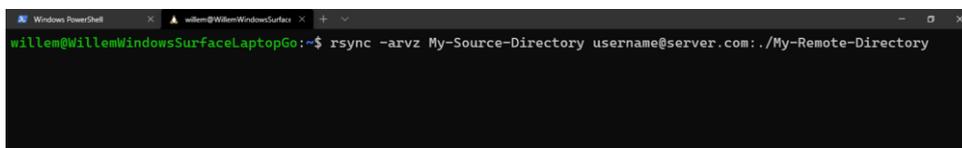
- Es ist unter der [GNU General Public License](#) lizenziert, was es zu **freier Software** macht; sowohl frei wie in "freie Rede" als auch "freies Bier"
- Der Algorithmus von Rsync ist eine Art [Delta-Kodierung](#), was bedeutet, dass er den Daten-/Netzwerkverkehr minimiert, da er **nur die Teile der Dateien kopiert, die sich geändert haben**. Darüber hinaus können Daten komprimiert werden, um den Dateiübertragungsverkehr noch weiter zu optimieren.
- Es kann **unterbrochene Übertragungen mithilfe der Prüfsummenprüfung fortsetzen**. Dies ermöglicht es rsync, Fehler beim Kopieren einer Datei von einem Ort zu einem anderen zu erkennen. Wenn sich zwei Dateien nur um ein Bit unterscheiden, werden ihre Prüfsummen unterschiedlich. Die Prüfsummenprüfung ist eine effiziente Möglichkeit, um zu überprüfen, ob jedes Byte ordnungsgemäß an sein Ziel übertragen wurde.

Installation von rsync

Wenn Sie GNU/Linux, *BSD oder Apple macOS verwenden, ist rsync wahrscheinlich bereits installiert. Wenn Sie Windows verwenden, können Sie sich [das Windows-Subsystem für Linux oder WSL](#) ansehen, um Zugriff auf alle Standard-GNU/Linux-Tools zu erhalten. Alternativ zu WSL können Sie [Cygwin](#) verwenden, um die große Sammlung von GNU- und Open-Source-Tools unter Windows zu installieren.

Typische Verwendung von rsync

Die Verwendung von rsync erfolgt über die Befehlszeile oder das Terminal. Sie führen den Befehl 'rsync' gefolgt von (optionalen) Flags, einem Quellpfad und einem Zielpfad aus.



```
willem@willemWindowsSurfaceLaptopGo:~$ rsync -arvz My-Source-Directory username@server.com:./My-Remote-Directory
```

Typischer rsync-Befehl mit den Flags -arvz, einem Mein-Quellverzeichnis und Mein-Zielverzeichnis auf einem Server

Mit den Flags **-arvz** konfigurieren Sie rsync so:

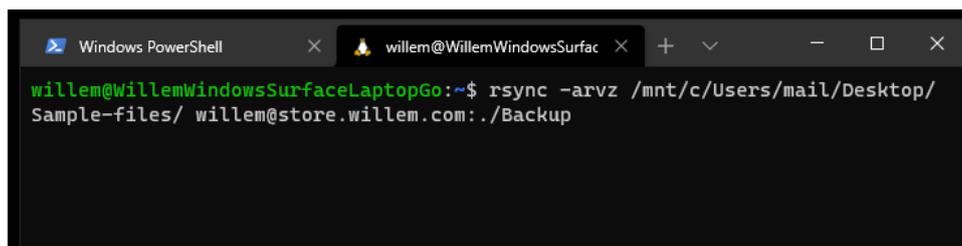
- **a**: "Archiv"-Modus, der Berechtigungen, Erstellungs- und Aktualisierungszeiten beibehält
- **r**: "rekursiver" Modus ermöglicht rsync die rekursive Synchronisierung von Unterordnern innerhalb der Quell-/Zielverzeichnisse

- **v**: "verbose" aktiviert eine ausführlichere Ausgabe auf der Konsole, sodass Sie den Fortschritt Zeile für Zeile verfolgen können
- **z**: "compress" aktiviert die Komprimierung von Daten, während sie an den Zielrechner gesendet werden, was die Menge der übertragenen Daten reduziert, nützlich bei langsamen oder getakteten Verbindungen
- **--delete**: ermöglicht das Entfernen von Dateien auf der entfernten Seite, wenn sie aus dem Quellverzeichnis entfernt werden. Sie sollten mit dieser Option vorsichtig sein!

Es gibt viele weitere Flags und Optionen, mit denen Sie das Verhalten von rsync an Ihre Bedürfnisse anpassen können. Sie sollten sich die [Manpage für rsync](#) ansehen, um eine vollständige Liste aller Optionen zu erhalten.

Wie man rsync zum Sichern von Dateien verwendet

Im folgenden Beispiel verwende ich rsync, um ein Verzeichnis ("Sample-Files") auf meinem Windows 10-Desktop auf einen Remote-Server zu kopieren. Ich verwende [WSL unter Windows](#), um eine funktionierende Debian_GNU/Linux-Umgebung in meinem Windows-Terminal zu haben.



```

Windows PowerShell
willem@WillemWindowsSurfac
willem@WillemWindowsSurfaceLaptopGo:~$ rsync -arvz /mnt/c/Users/mail/Desktop/Sample-files/ willem@store.willem.com:./Backup

```

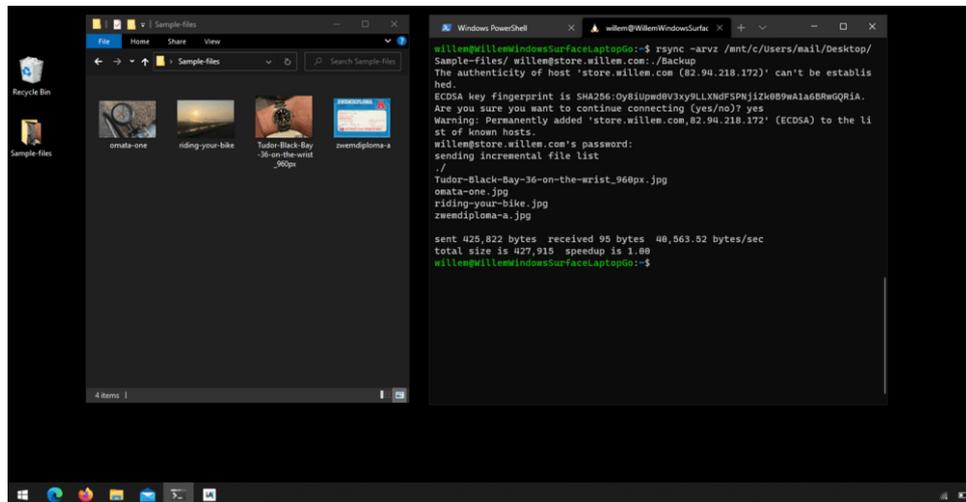
Verwendung von rsync unter Windows 10 mit WSL

Lassen Sie mich den Befehl erklären, den ich verwende:

- **rsync** ist der Name des Programms, das ich verwende
- **-arvz** sind die Flags, die rsync so konfigurieren, dass ein archiviertes Backup erstellt wird, einschließlich Unterverzeichnissen und der Übertragung von Daten mit Komprimierung
- **/mnt/c/Users/mail/Desktop/Sample-files/** ist der Pfad zum Quellverzeichnis, wie er in der WSL-Verzeichnisstruktur angezeigt wird. Er verweist auf meinen Windows-Desktop (der sich auf dem Laufwerk C: befindet)
- **willem @ store.willem.com:./Backup** ist das Ziel, das Sie als "Benutzer willem auf dem Server 'store.willem.com'" und "auf diesem Server" den Ordner "Backup" im **./**-Homeverzeichnis des "willem-Benutzers" lesen sollten.

Wenn Sie diesen Befehl zum ersten Mal ausführen, wird der kryptografische Fingerabdruck des Remote-Servers angezeigt. Dies dient dazu, sicherzustellen, dass Sie mit *Ihrem* Server sprechen. Dies verhindert, dass Hacker *Ihren* Server imitieren, ohne dass Sie es bemerken (da der Fingerabdruck für Ihren Server eindeutig ist). Wenn rsync eine Änderung dieses Fingerabdrucks feststellt, verweigert es die Übertragung von Dateien. Dies ist entscheidend, da das Kopieren Ihrer Backups an ein zufälliges entferntes Ziel ein potenzielles Risiko für Datenlecks darstellen würde!

Sie werden nach Ihrem Passwort gefragt, wenn Sie sich über SSH ohne schlüsselbasierte Authentifizierung verbinden. Dies ist in Ordnung, wenn Sie Ihre Dateiübertragung manuell durchführen möchten, aber wenn Sie die Übertragungen automatisieren möchten, sollten Sie eine [schlüsselbasierte Authentifizierung](#) einrichten (mit dem praktischen Befehl "ssh-copy-id"). Ihr Computer kann sich dann sicher beim Remote-Rechner anmelden, ohne dass Sie ein Passwort manuell eingeben müssen.

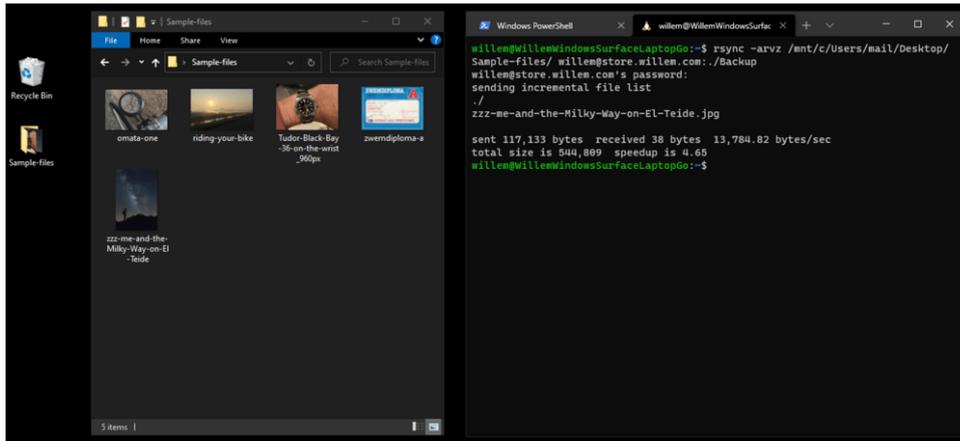


Verwendung von rsync zum Kopieren von Dateien von meinem Windows 10-Desktop auf einen Debian GNU/Linux-Server über SSH (mit WSL)

Wenn rsync erfolgreich eine Verbindung herstellen kann, fährt es fort, zu bestimmen, welche Dateien übertragen werden müssen. Wenn Sie das Flag "-v" gesetzt haben, sehen Sie den Fortschritt Zeile für Zeile. Nachdem die Übertragung abgeschlossen ist, beendet rsync mit einer Zusammenfassung.

Inkrementelle Übertragungen

Der Clou von Rsync ist, dass es inkrementelle Datenübertragungen standardmäßig vollständig unterstützt. Wenn Sie einige neue Dateien haben und dann denselben rsync-Befehl erneut ausführen, werden nur die Dateien übertragen, die übertragen werden müssen.



Nach dem Hinzufügen des Milchstraßenfotos zu meinen Beispieldateien führe ich rsync erneut aus, damit es meine Dateien inkrementell überträgt und nur die neue Datei kopiert

Sie können im Screenshot sehen, dass rsync die neue Datei automatisch erkannt hat. Es hat nur diese Datei auf den Remote-Server übertragen. Dies spart Bandbreite, Zeit und Rechenleistung. Es funktioniert mit neuen Dateien und Dateien, die aktualisiert wurden.

Optional können Sie das Flag "--delete" verwenden, um lokale Löschungen an das Zielverzeichnis weiterzugeben und so ein perfektes Spiegelbild der lokalen Dateistruktur zu erstellen. Für Sicherheitszwecke sollten Sie dies möglicherweise überspringen (da Sie Dateien unbeabsichtigt entfernen könnten).

Automatisieren von Übertragungen und Backups

Da rsync "nur ein Terminalbefehl" ist, können Sie es einfach in Ihren eigenen Skripten verwenden. Stellen Sie sich ein Skript als eine Sammlung von rsync-Befehlen vor, die eine Reihe von "Quell"- und "Ziel"-Links zwischen Ihren Dateien und ihren jeweiligen Backup-Speicherorten umfassen.

Sie können planen, dass Ihr Skript automatisch an einem bestimmten Wochentag und zu einer bestimmten Tageszeit ausgeführt wird. Unter GNU/Linux, *BSD und macOS können Sie dazu **cron** verwenden. Unter Windows können Sie die App "Aufgabenplanung" verwenden.

Schlussfolgerung

Mit rsync können Sie Dateien von Ihrem Computer effizient, sicher und inkrementell auf ein anderes System übertragen. Es kann ein entscheidendes Werkzeug in Ihrem Backup-Plan sein.

Es ist kostenlos und weit verbreitet verfügbar. Seine Leistung und Zuverlässigkeit sind weithin anerkannt. Es gibt wirklich keine Entschuldigung dafür, warum Sie Ihre Backups nicht richtig einrichten sollten!