Datensicherung und -wiederherstellung mit BorgBackup

Geborgen

Michael Plura

BorgBackup ist eine einfach zu bedienende Backup-Software, die wenig Ressourcen benötigt. Es füllt die Lücke zwischen einfachen Tools wie *rsync* und Backup-Frameworks wie Bacula oder Bareos.

BorgBackup, kurz Borg, ist ein in Python 3 entwickeltes Backup-Programm, dessen zeitkritische Teile der Software in C/Cython geschrieben und kompiliert wurden. Es basiert ursprünglich auf Attic und wird von einem Team rund um den Schweden Jonas Borgström entwickelt – daher der Name. Zu sichernde Daten dedupliziert und komprimiert es, was die benötigte Bandbreite zum und den Platz im Backup-Storage gering hält. Dank der Authentifizierung per HMAC-SHA256 und einer optionalen, clientseitigen Verschlüsselung mit 256-Bit-AES kann man Backups ohne allzu große Bedenken auf nicht vertrauenswürdigen Zielen wie der Cloud anlegen.

Das quelloffene Borg unterliegt der BSD-Lizenz und ist in der Regel über das Paketmanagement von Linux, FreeBSD, OpenBSD, NetBSD oder Mac OS X (brew) als *borg* oder *borgbackup* zu installieren. Für WSL (Windows Subsystem for Linux) oder Cygwin gibt es eine experimentelle Variante. Alternativ lässt sich die Software über den Python-Installer *pip* einrichten:

```
pip install borgbackup
```

Zusätzlich stehen auf der GitHub-Seite des Projekts Binaries für Linux, FreeBSD und Mac OS X zur Verfügung. Benutzer rufen diese, falls sie im *$PATH* liegen, per *borg* auf. Eine Übersicht über alle Befehle und zusätzliche Infos zu einem bestimmten Befehl erhält man per

```
borg --help
borg create --help
```

Die Entwickler stellen eine gute Onlinedokumentation unter borgbackup.readthedocs.io bereit.

```
mipl@z600-1: ~
Datei  Bearbeiten  Ansicht  Suchen  Terminal  Hilfe
mipl@openbsd63:~$ borg create --stats /home/mipl/backup/mipl/::Mittwoch ~/doc>
Enter passphrase for key /home/mipl/backup/mipl:
------------------------------------------------------------------------------
Archive name: Mittwoch
Archive fingerprint: 460ff98aa6c37d3f14453e4553e1f1b5de6dbd79d286bffd63c1184b4a
1e57d6
Time (start): Fri, 2018-04-06 12:17:06
Time (end):   Fri, 2018-04-06 12:17:07
Duration: 0.23 seconds
Number of files: 300
Utilization of max. archive size: 0%
------------------------------------------------------------------------------
                       Original size      Compressed size    Deduplicated size
This archive:               20.29 MB             18.59 MB             18.22 kB
All archives:               40.41 MB             37.09 MB             18.20 MB

                       Unique chunks         Total chunks
Chunk index:                     192                  570
------------------------------------------------------------------------------
mipl@openbsd63:~$
```

Das freie BorgBackup sichert Daten schnell, einfach und platzsparend.

Borg organisiert das Backup-Storage in Form eines Repository (ein Verzeichnis), in dem die einzelnen Backups als Info- und Datenblöcke abgelegt werden.

Im Beispiel sichert und überprüft der Nutzer *mipl* seine Dokumente und Sourcen verschlüsselt auf einem per NFS eingebundenen Server:

```
borg init --encryption=repokey /<path>/mipl
```

Diese Befehle legt das Repository unter */<path>p/mipl* an und fragt den Benutzer nach einem Passwort, das für jeden weiteren Zugriff erforderlich ist. Das Passwort kann als Umgebungsvariable *export BORG_PASSPHRASE="KeineGuteIdee.666"* für einen automatischen Betrieb gesetzt werden.

Ein Backup erzeugt man mit dem Argument *create* und einem Namen, im Beispiel „Montag“. Gesichert werden die Verzeichnisse *doc* und *code*:

```
borg create --stats /<path>/mipl::Montag ↵
                                        ~/doc ~/code
borg check -v --verify-data /<path>/ ↵
                                        mipl::Montag
```

Derselbe Befehl mit anderen Labels legt weitere, deduplizierte Backups an. Eine Übersicht liefert *borg list*. Nicht mehr benötigte Sicherungen löscht der Befehl

```
borg delete /<path>/mipl::Montag
```

Dabei berücksichtigt Borg die Deduplizierung und hält immer einen kompletten Backup-Satz vor. Eine vollständige Datensicherung stellt man mit

```
borg extract --list /<path>/mipl::Freitag
```

wieder her. Ein Repository oder einzelnes Backup lässt sich über FUSE in das Dateisystem einbinden, um so mit einem Dateimanager einzelne Dateien zu restaurieren:

```
mkdir tmp
borg mount /<path>/mipl::Freitag ~/tmp
[...]
borg unmount ~/tmp
```

Normalerweise komprimiert Borg die Daten mit dem schnellen *lz4*. Besser und dabei anpassbar ist *zstd,N*, wobei *N* zwischen N=1 (schnell) und N=22 (hochkomprimiert) gewählt wird. Zum Testen (und Lernen) lassen sich die meisten Borg-Befehle mit den Parametern *-v --list --dry-run* starten. Sie zeigen dann ausführlich, was passiert, nehmen aber keinerlei Änderungen vor. (jab@ix.de)

Michael Plura

ist freier Autor.

Alle Links: ix.de/ix1805142

© Copyright by Heise Medien
Persönliches PDF für alle Leser der iX aus 30625 Hannover