Pull-Backup mit Bontmia

Mit bontmia lassen sich tägliche Snapshots von Dateien auf einem entfernten Server erstellen. Bontmia läuft hierbei auf dem Backupserver, so dass auf dem zu sichernden Server keinerlei Login-Daten vom Backupserver liegen. Um auf dem Backupserver Platz zu sparen werden nicht geänderte Dateien als Hardlinks abgelegt.

Installation

Backupserver

Benötigt wird das FeM-Gentoo-Overlay!

- ***emerge app-backup/bontmia***
- ***emerge net-misc/rsync***

Versionen:

- 0.14 - Original-Version des Authors
- 0.14-r1 - mit integriertem Patch , welcher die Angabe eines SSH-Schlüssels (--identity) und einer Exclude-Datei (--exclude-from) für rsync erlaubt

Alternativ ist das Skript auch unter http://folk.uio.no/johnen/bontmia verfügbar werden. Im FeM-Gentoo-Overlay liegt ein passender Patch, welcher die Angabe eines SSH-Schlüssels (--identity) und einer Exclude-Datei (--exclude-from) für rsync erlaubt.

zu sichernder Server

- ***emerge net-misc/rsync***
- ***emerge app-admin/sudo***

Konfiguration

zu sichernder Server

Benutzer und SSH-Key anlegen

Der Schlüssel muss **ohne** Passwort (Passphrase) erstellt werden, sonst macht das automatische Backup wenig Spaß.

- ***useradd -m backup***
- ***ssh-keygen -f /home/backup/.ssh/id_rsa***

Ausgabe

```
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/backup/.ssh/id_rsa.
Your public key has been saved in /home/backup/.ssh/id_rsa.pub.
The key fingerprint is:
3e:65:ec:fd:ba:92:15:94:18:da:74:26:d4:fc:b9:f0 root@server1
The key's randomart image is:
+--[ RSA 2048]----+
|         .==o.   |
|         +.+=    |
|        . .. . . |
|         .  o o  |
|        S +  + . |
|       . + .. E  |
|        o .o.    |
|         .o  .   |
|           .oo.  |
+-----------------+
```

- ***echo "no-port-forwarding,no-X11-forwarding $(cat /home/backup/.ssh/id_rsa.pub)" >> /home/backup/.ssh/authorized_keys***
- ***chown backup:backup /home/backup/.ssh/****

Der Private Schlüssel für den Backup-Nutzer (/home/backup/.ssh/id_rsa) muss nun sicher (z.B. per SCP) auf den Backup-Server übertragen werden. Anschließend kann er auf dem zu sichernden Server gelöscht werden.

- ***rm /home/backup/.ssh/id_rsa***

Sudo einrichten

Damit rsync auf alle Dateien zugreifen kann, muss es mit root-Rechten ausgeführt werden. Dies geschieht via sudo.

/etc/sudoers

```
# Allow rsync for backup-user
backup  ALL=(root)NOPASSWD:/usr/bin/rsync --server --sender -
vlHogDtprRze.iLsf . /
```

Hinweis: Der zu sichernde Pfad ist hier /, kann aber bei Bedarf auch eingeschränkt werden.

Backupserver

Annahmen:

- Server: server1.example.org (Port 1022)
- Backup-Verzeichnis: /mnt/backup/server1.example.org/backup/
- SSH-Key: /mnt/backup/server1.example.org/identity
- exclude-Datei: /mnt/backup/server1.example.org/exclude

SSH-Hostkey speichern

Zu allererst muss man sich einmal per SSH mit dem zu sichernden Server als der Benutzer auf dem Backupserver verbinden unter dem dann das Backup läuft.

- ***ssh backup@server1.example.org -i /mnt/backup/server1.example.org/identity***
  - Are you sure you want to continue connecting (yes/no)? ***yes***

Ausgabe

```
The authenticity of host 'server1.example.org (10.200.2.211)' can't be
established.
RSA key fingerprint is c9:08:84:ea:1c:2c:84:5a:0f:cd:6e:1f:a1:bf:fe:4a.
Are you sure you want to continue connecting (yes/no)?
```

Nun sollte man als Nutzer Backup auf dem genannten Server eingeloggt sein.

Sicherungsinhalt festlegen

Ausschließliche Exclude-Liste

Mit dieser Variante werden die genannten Verzeichnisse und/oder Dateien (Wildcards erlaubt) vom Backup **ausgeschlossen**. Alles andere wird gesichert.

In diesem Beispiel werden alle Verzeichnisse außer /dev, /mnt, /proc, /sys und /usr/portage und alle Dateien außer /opt/xen//.fs gesichert.

/mnt/backup/server1.example.org/exclude

```
/dev
/mnt
/opt/xen/*/*.fs
/proc
/sys
/usr/portage
```

Kombinierte Include-/Exclude-Liste

Mit dieser Variante legt man explizit die zu sichernden Verzeichnisse und/oder Dateien (Wildcards erlaubt) fest, die vom Backup **eingeschlossen** werden. Alles andere wird **nicht** gesichert.

In diesem Beispiel werden die Verzeichnisse /etc, /home, /root (ohne root/.cache), /usr/local, /var/lib/mysql, /var/lib/portage, /var/log und /var/www gesichert.

/mnt/backup/server1.example.org/exclude

```
+ etc
+ etc/**
+ home
+ home/**
- root/.cache
- root/.cache/**
+ root
+ root/**
+ usr
+ usr/local
+ usr/local/**
+ var
+ var/lib
+ var/lib/mysql
+ var/lib/mysql/**
+ var/lib/portage
+ var/lib/portage/**
+ var/log
+ var/log/**
+ var/www
+ var/www/**
- *
```

Backup

Auf dem Backupserver kann nun mit

- ***bontmia --dest /mnt/backup/server1.example.org/backup --rotation 0minutes0hours7days4weeks3month0years --port 1022 --identity /mnt/backup/server1.example.org/identity --exclude-from /mnt/backup/server1.example.org/exclude backup@server1.example.org:***

das Backup anstoßen.

Ausgabe

```
bdirmatch: ^(./server1.example.org:/)
```

```
No previous backup detected, will start with an empty replication

Backing up
  server1.example.org:

Deletes files that should not be in the latest snapshot

Moving the complete backup into the backup archive
  (/mnt/backup/server1.example.org/backup/unfinished_backup ->
/mnt/backup/server1.example.org/backup/)

Calculates which backups to save
(0 minutes, 0 hours, 7 days, 4 weeks, 3 months, 0 years)
  Saving /mnt/backup/server1.example.org/backup/2012/03/31/12:50 by filters:
days weeks month
```

Anschließend kann man dies auch per Cronjob erledigen lassen. Der Parameter „–rotation“ gibt an, welche Backups vorgehalten werden - im Beispiel sind das

- monatliches Backup für 3 Monate
- wöchentliches Backup für 4 Wochen
- tägliches Backup für 7 Tage

From:
https://wiki.fem.tu-ilmenau.de/ - **FeM-Wiki**

Permanent link:
https://wiki.fem.tu-ilmenau.de/public/technik/howto/bontmia?rev=1357393084

Last update: **2013/01/05 14:38**