

Vzdálené zálohování pomocí Borgbackup

Instalace Borg Backup

Instalaci Borga

```
sudo pacman -S borgbackup
```

Příprava cílového server

Na cílovém serveru je potřeba nainstalovat Borg Backup a vytvořit uživatele, který bude mít práva pro ukládání záloh. Pokud máte server s Ubuntu/Debianem, můžete nainstalovat Borg Backup pomocí příkazu:

```
sudo apt install borgbackup
```

Dále vytvořte uživatele s názvem “borg”, který bude mít práva pro ukládání záloh. To lze provést pomocí následujícího příkazu:

```
sudo adduser borg
```

Poté je třeba nastavit heslo pro uživatele borg a vytvořit adresář pro ukládání záloh. Vytvořte například adresář “/home/borg/backup”.

```
mkdir /home/borg/backup
```

Vytvoření klíče pro přihlášení

Pro přístup k cílovému serveru potřebujeme klíč pro přihlášení. Můžeme použít buď heslo nebo SSH klíč. V tomto tutoriálu se budeme zaměřovat na SSH klíč.

Na stroji, který bude provádět zálohy, vygenerujeme SSH klíč pro uživatele, který bude zálohu provádět. To můžete provést pomocí následujícího příkazu:

```
ssh-keygen -t rsa
```

Tento příkaz vygeneruje ve vašem domovském adresáři klíče pro SSH připojení – soubor `id_rsa` a soubor `id_rsa.pub`.

Nyní nahrajem SSH klíče na server.

```
cat ~/.ssh/id_rsa.pub | ssh borg@ip-adresa "mkdir -p ~/.ssh && touch ~/.ssh/authorized_keys && chmod -R go=
```

Může se zobrazit následující zpráva:

Output

The authenticity of host 'ip_dadresa' can't be established.

ECDSA key fingerprint is fd:fd:d4:f9:77:fe:73:84:e1:55:00:ad:d6:6d:22:fe.

Are you sure you want to continue connecting (yes/no)? yes

To znamená, že váš místní počítač nerozpozná vzdáleného hostitele. K tomu dojde při prvním připojení k novému hostiteli. Typ `yes` a stiskněte `ENTER` pokračovat.

Poté byste měli být vyzváni k zadání hesla vzdáleného uživatelského účtu:

Output

borg@ip_adresa's password:

Otestujte přihlášení

- Odhlaste se ze serveru příkazem `exit`
- Zkuste se přihlásit na server pomocí SSH opět příkazem `ssh borg@ip_adresa`
- Pokud se úspěšně přihlásíte bez zadávání hesla, pak jste správně nahráli klíče SSH na server.

Pokud jste se mohli přihlásit ke svému účtu pomocí SSH bez hesla, úspěšně jste nakonfigurovali ověřování na základě klíče SSH pro váš účet. Váš mechanismus ověřování na základě hesla je však stále aktivní, což znamená, že váš server je stále vystaven útokům hrubou silou.

Na tomto serveru nakonfigurováno ověřování na základě klíče SSH. **Před dokončením kroků v této části se ujistěte, že máte pro účet root**, nebo pokud možno, že máte na tomto serveru nakonfigurované ověřování na základě klíče SSH pro účet bez oprávnění root. server s `sudo` privilegia. Tento krok uzamkne přihlášení založená na heslech, takže je zásadní zajistit, že stále budete moci získat přístup pro správce.

Jakmile potvrdíte, že váš vzdálený účet má oprávnění správce, přihlaste se ke vzdálenému serveru pomocí klíčů SSH, buď jako **root** nebo pomocí účtu s `sudo` privilegia. Poté otevřete konfigurační soubor démona SSH:

```
sudo nano /etc/ssh/sshd_config
```

Odkomentujte řádek odstraněním `#` a nastavte hodnotu na `no`. Tím zakážete možnost přihlášení přes SSH pomocí hesel účtů:

```
...  
PasswordAuthentication no  
...
```

Po dokončení uložte a zavřete soubor stisknutím tlačítka `CTRL+X`, pak `Y` pro potvrzení uložení souboru a nakonec `ENTER` pro ukončení nano. Abychom skutečně aktivovali tyto změny, musíme restartovat `sshd` servis:

```
sudo systemctl restart ssh
```

Vytvořte repozitář

Předtím, než můžete vytvářet zálohy, musíte vytvořit repozitář, což je úložiště pro vaše zálohy. To můžete udělat pomocí následujícího příkazu:

```
borg init -e repokey-blake2 ssh borg@ip_adresa:/home/borg/backup
```

Vytváření záloh

3. Poté, co jste vytvořili repozitář, můžete začít vytvářet zálohy. Borg Backup umožňuje vytvářet inkrementální zálohy, což znamená, že při každém zálohování se zálohuje pouze změna oproti poslední záloze. To výrazně snižuje velikost záloh a rychlost zálohování.

Pro vytvoření zálohy použijte následující příkaz:

```
borg create username@remote-server:/path/to/repo::backup-name /path/to/directory
```

Kde `username` je vaše uživatelské jméno na vzdáleném serveru, `remote-server` je adresa vzdáleného serveru, `/path/to/repo` je cesta k repozitáři na vzdáleném serveru, `backup-name` je jméno zálohy a `/path/to/directory` je cesta ke složce, kterou chcete zálohovat.

```
borg list username@remote-server:/path/to/repo
```

Tento příkaz vypíše seznam všech záloh uložených v repozitáři.

Obnova zálohy

```
borg extract user@remote:/path/to/repo::backup-name /path/to/restore/folder
```

Vytvoření skriptu pro zálohování

Nyní musíme vytvořit skript pro zálohování, který bude spouštět BorgBackup a ukládat zálohy do záložního repozitáře.

Vytvořte soubor s názvem “backup.sh” a vložte do něj následující kód:

```
#!/bin/bash

# Nastavení proměnných
SOURCE="/cesta/k/souborum/k/zalohovani"
REPO="/cesta/k/repozitari"
DATE=`date +%Y-%m-%d`

# Spuštění zálohování
borg create $REPO::$DATE $SOURCE
```

Tento skript spustí BorgBackup a uloží zálohu do repozitáře s aktuálním datem jako název zálohy.

Otevřete terminál a spusťte příkaz:

Tento příkaz otevře cronovou tabulku

```
crontab -e
```

, kde můžete nastavit automatické spouštění skriptu.

Skript na automatické zálohy i sprořezáváním

```
#!/bin/sh

# Setting this, so the repo does not need to be given on the commandline:
export BORG_REPO=ssh://username@example.com:2022/~/backup/main

# See the section "Passphrase notes" for more infos.
export BORG_PASSPHRASE='XYZI0ngandsecurepa_55_phrasea&&123'

# some helpers and error handling:
info() { printf "\n%s %s\n\n" "$( date )" "$*" >&2; }
trap 'echo $( date ) Backup interrupted >&2; exit 2' INT TERM

info "Starting backup"
```

```
# Backup the most important directories into an archive named after
# the machine this script is currently running on:
```

```
borg create                \
  --verbose                 \
  --filter AME              \
  --list                    \
  --stats                   \
  --show-rc                 \
  --compression lz4         \
  --exclude-caches          \
  --exclude 'home/*/.cache/*' \
  --exclude 'var/tmp/*'     \
                             \
  ::'{hostname}-{now}'      \
  /etc                      \
  /home                     \
  /root                     \
  /var
```

```
backup_exit=$?
```

```
info "Pruning repository"
```

```
# Use the `prune` subcommand to maintain 7 daily, 4 weekly and 6 monthly
# archives of THIS machine. The '{hostname}-*' matching is very important to
# limit prune's operation to this machine's archives and not apply to
# other machines' archives also:
```

```
borg prune                 \
  --list                    \
  --glob-archives '{hostname}-*' \
  --show-rc                 \
  --keep-daily 7            \
  --keep-weekly 4           \
  --keep-monthly 6
```

```
prune_exit=$?
```

```
# actually free repo disk space by compacting segments
```

```
info "Compacting repository"
```

```
borg compact
```

```
compact_exit=$?
```

```
# use highest exit code as global exit code
```

```
global_exit=$(( backup_exit > prune_exit ? backup_exit : prune_exit ))
```

```
global_exit=$(( compact_exit > global_exit ? compact_exit : global_exit ))
```

```
if [ ${global_exit} -eq 0 ]; then
```

```
    info "Backup, Prune, and Compact finished successfully"
```

```
elif [ ${global_exit} -eq 1 ]; then
```

```
    info "Backup, Prune, and/or Compact finished with warnings"
```

```
else
```

```
    info "Backup, Prune, and/or Compact finished with errors"
```

```
fi
```

```
exit ${global_exit}
```

Revision #4

Created 8 April 2023 12:25:24 by archos

Updated 8 April 2023 12:36:58 by archos