Zálohování v Linuxu

Jak zálohovat svá oblíbená data v Arch Linuxu

- BorgBackup
  - BorgBackup
  - Vzdálené zálohování pomocí Borgbackup
  - Desktopový klient Vorta
- Jednoduché zálohování pomocí Rsync
- Zálohování pomocí Rdiffbackup

BorgBackup

Borg Backup je open-source nástroj pro zálohování souborů a složek, který je k dispozici pro operační systémy Linux, macOS. Tento program je navržen tak, aby byl jednoduchý k použití, ale zároveň poskytoval robustní a spolehlivou zálohovací funkci pro ukládání důležitých dat.

BorgBackup

Co je Borg

Borg je open-source nástroj pro zálohování souborů a složek, který je k dispozici pro operační systémy Linux, macOS. Tento program je navržen tak, aby byl jednoduchý k použití, ale zároveň poskytoval robustní a spolehlivou zálohovací funkci pro ukládání důležitých dat.

Borg používá jedinečnou technologii komprese dat, což umožňuje ukládat velké objemy dat v menších souborech, což zase znamená, že je možné ušetřit místo na disku. Další výhodou Borg Backupu je jeho rychlost – rychle zálohuje a obnovuje data, což je užitečné pro uživatele, kteří potřebují často měnit a ukládat data.

Borg poskytuje uživateli možnost zálohovat data do lokálního úložiště, jako jsou pevné disky, nebo do vzdáleného úložiště, jako jsou cloudové služby nebo FTP servery. Soubory a složky jsou zašifrovány pomocí AES-256, což zajišťuje bezpečnost a ochranu před neoprávněným přístupem.

V Borgu lze nastavit pravidelné zálohování, což znamená, že se data automaticky zálohují v určených intervalech. Tato funkce je užitečná pro uživatele, kteří chtějí mít neustále aktuální zálohu svých dat.

Další výhodou BorgBackup je jednoduchost. Pro zálohování stačí pouze několik příkazů v příkazovém řádku, což umožňuje uživatelům rychle a snadno zálohovat a obnovovat svá data.

Jak Borg funguje

Borg je to, čemu se říká „deduplikační zálohovací program“. Podobně jako u přírůstkových záloh jsou v následných zálohách archivována pouze data, která se skutečně změní na souborovém systému po provedení úplné zálohy, ale podobnosti jsou pouze koncepční. Borg funguje tak, že každý soubor rozdělí na části, které jsou identifikovány jejich hash součtem. Do „úložiště“ jsou přidány pouze bloky, které aplikace nerozpoznají. Tato technika deduplikace je skutečně účinná, protože mimo jiné nám umožňuje přesunout soubor nebo adresář, aniž by to bylo považováno za změnu, a proto vyžaduje další místo. Totéž platí pro časová razítka souborů. To, na čem skutečně záleží, jsou pouze části souborů, které se ukládají pouze jednou. V Linuxu Borg podporuje zachování všech standardních a rozšířených atributů souborového systému, jako jsou ACL a xattrs.

Instalace Borga

Nejprve si musíte nainstalovat Borg. To lze provést pomocí správce balíčků, nebo si můžete stáhnout instalační soubory z oficiálních stránek.

```
sudo pacman -S borgbackup
```

Vytvoření repozitáře

Předtím, než začnete zálohovat svá data, musíte vytvořit repozitář, kde se budou data ukládat. Můžete to udělat pomocí následujícího příkazu:

```
borg init -e repokey-blake2 /cesta/k/repozitari
```

Tento příkaz vytvoří nový Borg Backup repozitář v zadané cestě.

Vytvoření zálohy

Nyní můžete vytvořit zálohu svých dat pomocí následujícího příkazu:

```
borg create /cesta/k/repozitari::jmeno-zalohy /cesta/k/slozkam-k-zalohovani
```

Tento příkaz vytvoří novou zálohu s názvem “jmeno-zalohy” a uloží ji do vašeho Borg Backup repozitáře. Data, která chcete zálohovat, musí být umístěna v adresáři uvedeném po druhém lomítku.

Pravidelné zálohování

Pro pravidelné zálohování můžete použít nástroje jako Cron nebo Systemd timer. Pro příklad zálohování každý den v 2 ráno pomocí Cronu, můžete přidat následující řádek do svého Crontab souboru:

```
0 2 * * * /usr/bin/borg create /cesta/k/repozitari::`date +\%Y-\%m-\%d` /cesta/k/slozkam-k-zaloh
```

Tento příkaz vytvoří zálohu každý den v 2 ráno s aktuálním datem jako název zálohy.

Obnovení zálohy

Pokud potřebujete obnovit data ze zálohy, můžete to udělat pomocí následujícího příkazu:

```
borg extract /cesta/k/repozitari::jmeno-zalohy /cesta/k/kam-se-data-obnovi
```

Základní příkazy

1. `borg init`: Tento příkaz vytváří nový záložní repozitář.
2. `borg create`: Tento příkaz vytváří novou zálohu vašich souborů nebo složek.
3. `borg list`: Tento příkaz vypisuje seznam záloh v záložním repozitáři.
4. `borg info`: Tento příkaz poskytuje informace o záložním repozitáři, jako je například jeho velikost a počet záloh.
5. `borg extract`: Tento příkaz obnovuje soubory ze zálohy.
6. `borg delete`: Tento příkaz odstraňuje zálohy z repozitáře.
7. `borg mount`: Tento příkaz umožňuje přístup k obsahu záložního repozitáře jako ke složce, což umožňuje snadnější obnovu souborů.
8. `borg prune`: Tento příkaz odstraňuje staré zálohy z repozitáře, aby se uvolnilo místo na disku

To jsou základní kroky pro použití Borg Backup pro zálohování vašich souborů a složek. Samozřejmě existuje mnoho dalších pokročilých funkcí, které můžete použít, abyste co nejlépe využili tohoto nástroje.

Skript na automatickou zálohu i s prořezáváním

Následující skript můžete použít pro vytvoření automatických záloh pomocí Borg Backup:

```
#!/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='XYZl0ngandsecurepa_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}
```

Důležité odkazy:

- Borgova dokumentace

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
```

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='XYZl0ngandsecurepa_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}
```

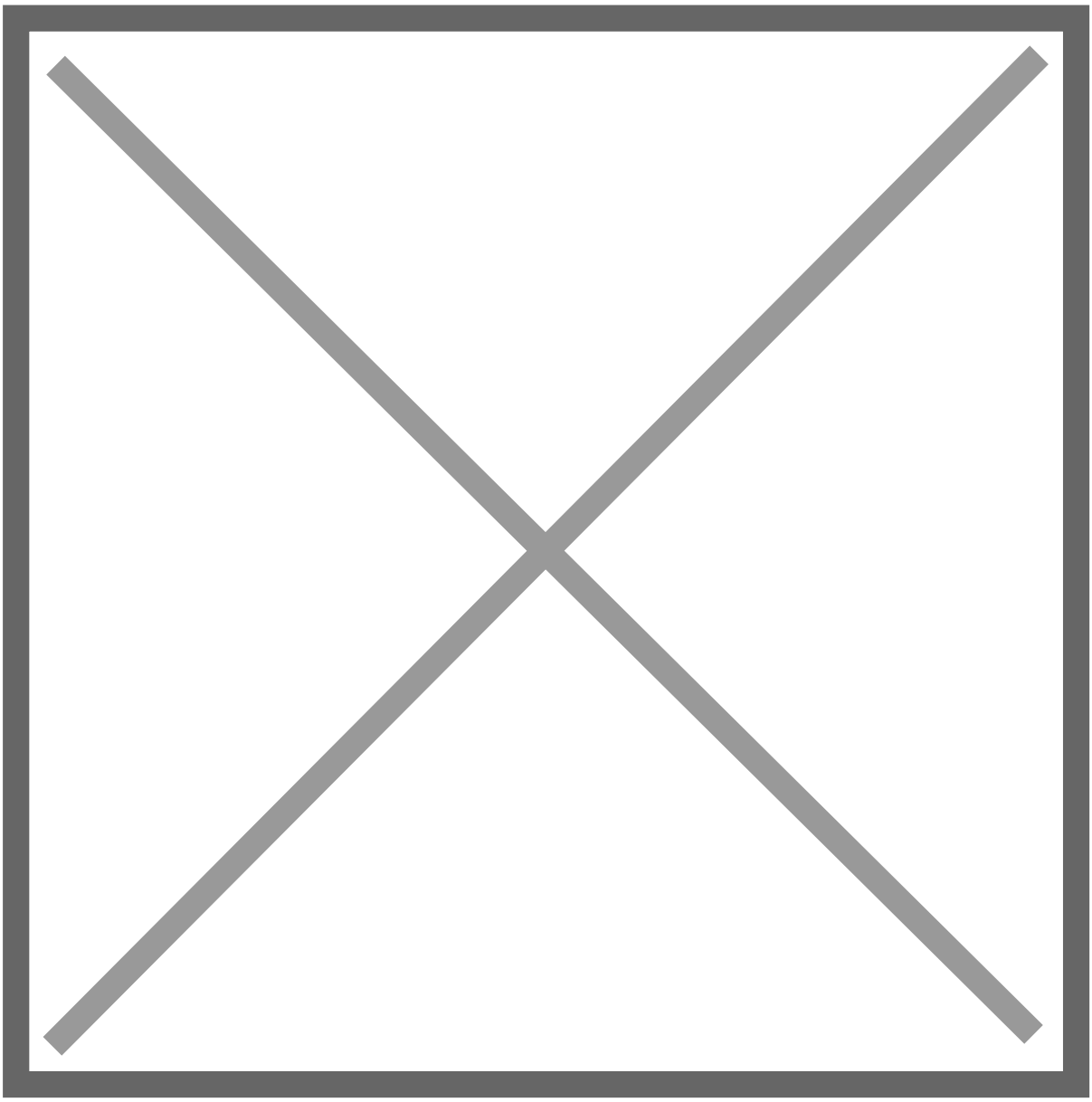
Desktopový klient Vorta

Co je Vorta

Vorta je open-source aplikace pro zálohování dat v Linuxu. Jedním z hlavních důvodů, proč se Vorta stala oblíbenou mezi uživateli Linuxu, je její snadná instalace a použití. Vorta je k dispozici v mnoha distribucích Linuxu, jako je například Ubuntu, Fedora, Debian, Arch Linux a další. Po instalaci aplikace se uživatelé mohou snadno připojit k existujícímu účtu Borg Backup nebo si vytvořit nový účet a začít vytvářet zálohy.

Vorta má také mnoho užitečných funkcí, jako je například možnost plánovat pravidelné zálohování, možnost vytvářet zálohy z různých zdrojů dat, jako jsou například složky, soubory, nebo celé disky, a možnost šifrování záloh pro zajištění bezpečnosti dat.

Dále je také možné vytvářet zálohy na vzdálených serverech.



Výhody

- **Šifrované, deduplikované a komprimované zálohy** využívající Borg jako backend.
- **Žádné uzamčení dodavatel**é – zálohujte na místní disky, svůj vlastní server nebo BorgBase , hostingovou službu pro Borgské zálohy.
- **Open source** – zdarma k použití, úpravě, vylepšení a auditu.
- **Flexibilní profily** pro seskupování zdrojových složek, cílů zálohování a plánů.
- **Jedno místo** pro zobrazení všech archivů k určitému okamžiku a obnovení jednotlivých souborů.

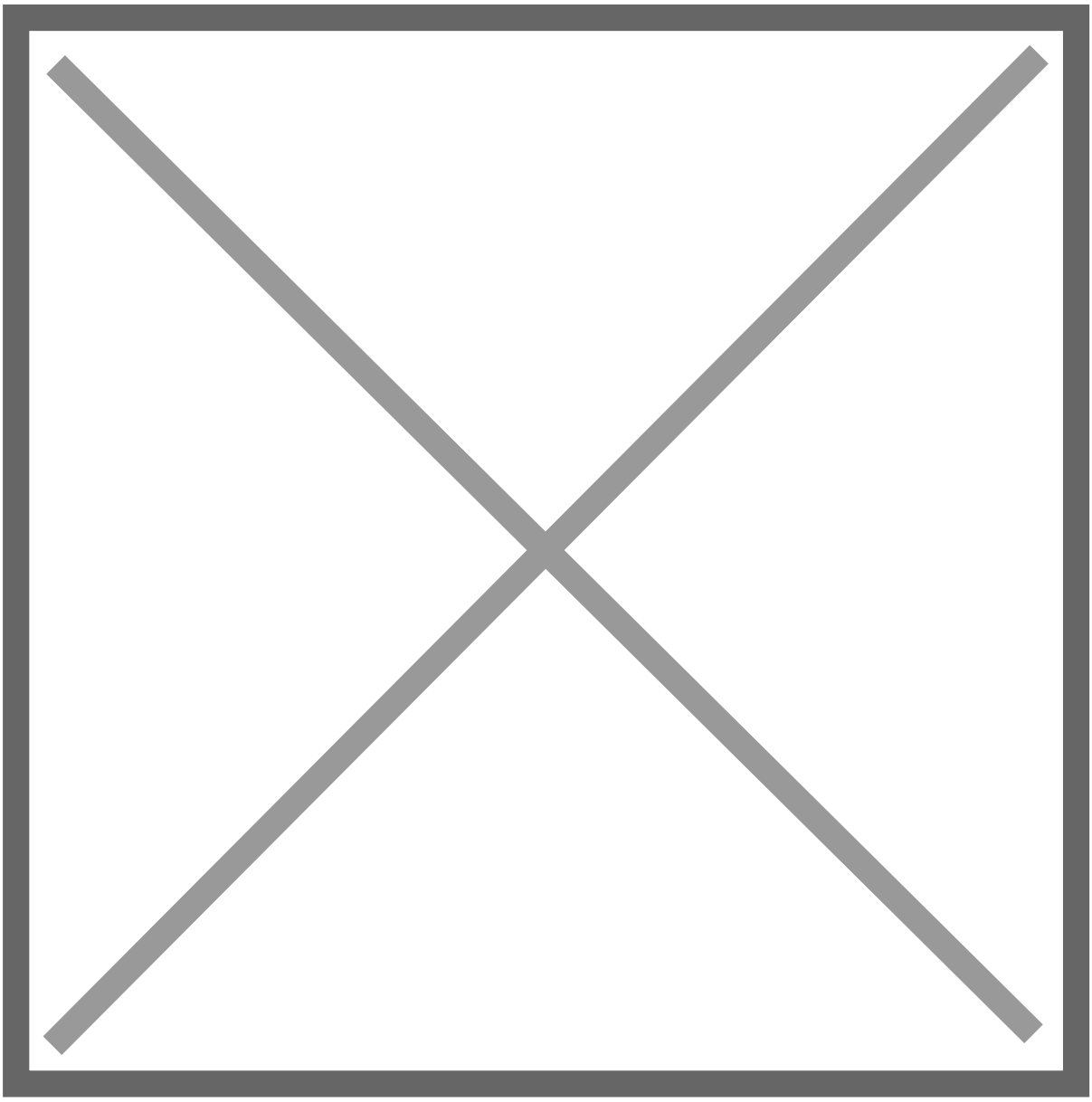
Instalace

K instalaci použijte balíček AUR

```
yay -S vorta
```

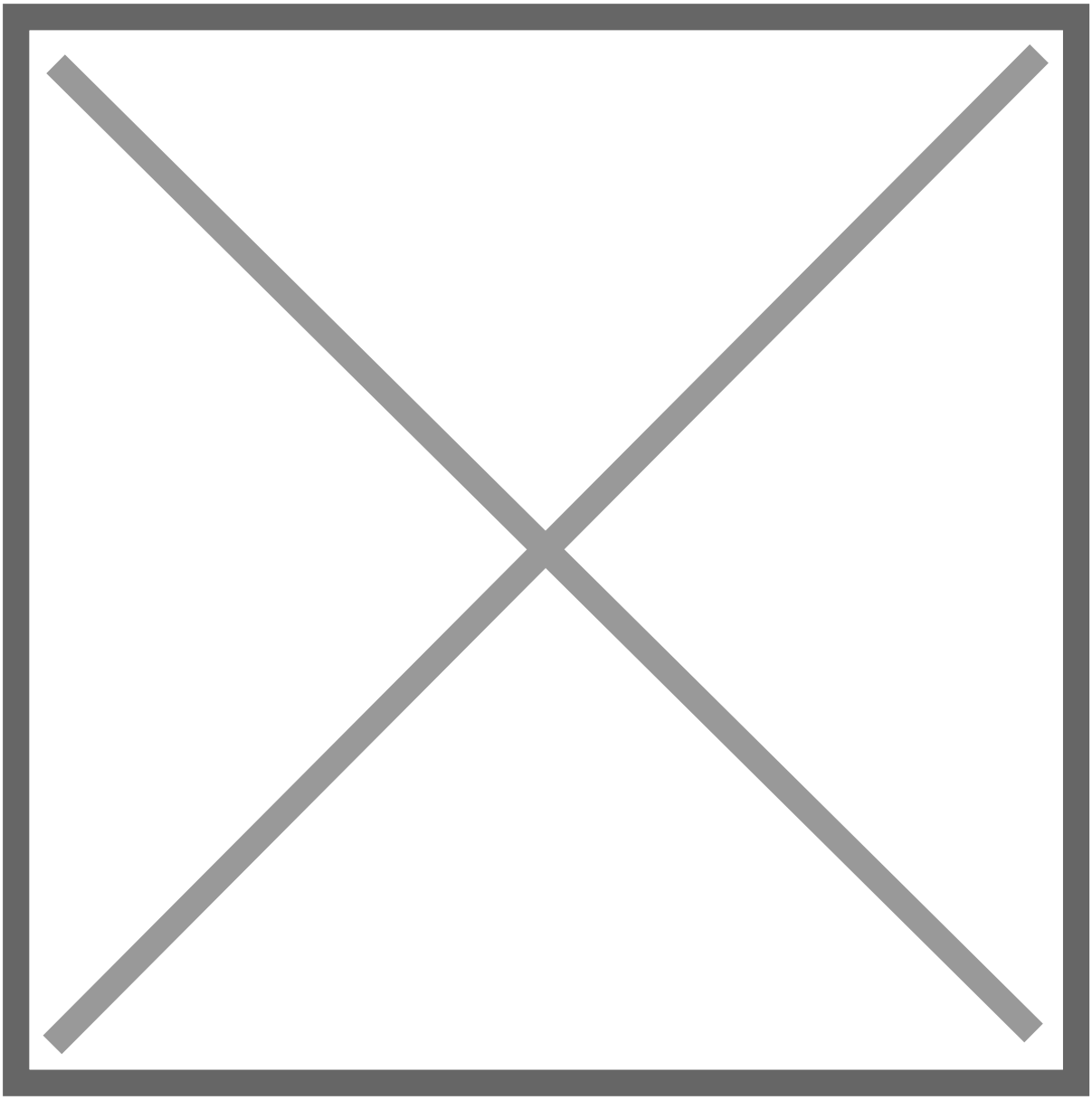
Nastavení Vorty pro místní zálohy

Po spuštění Vorty byste měli vidět nové okno nastavení.



Nastavení místního úložiště

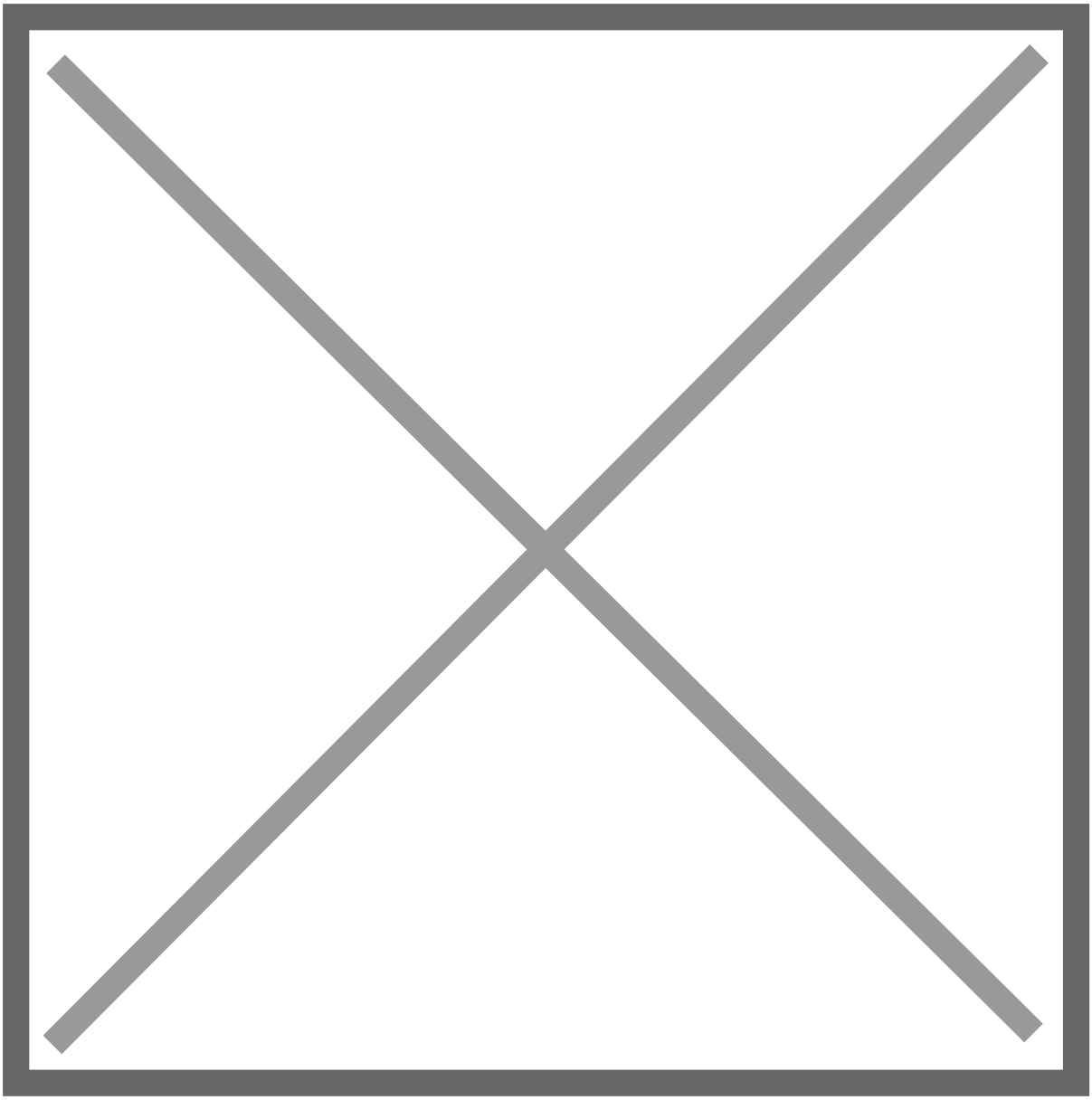
Klikněte na **Úložiště** rozevírací nabídku **a vyberte možnost Inicializovat nové úložiště** . Po kliknutí by se mělo objevit nové okno.



Klikněte na local3.png?w=810&ssl=1ikonu a vyberte adresář, ve kterém chcete inicializovat své nové úložiště Borg. Zadejte vámi vybranou přístupovou frázi pro úložiště. **Ujistěte se a uložte své přístupové heslo na bezpečném místě!** Když jste si jisti, že jsou všechny informace, které jste zadali, správné, stiskněte **Přidat** .

Přidejte složky k zálohování

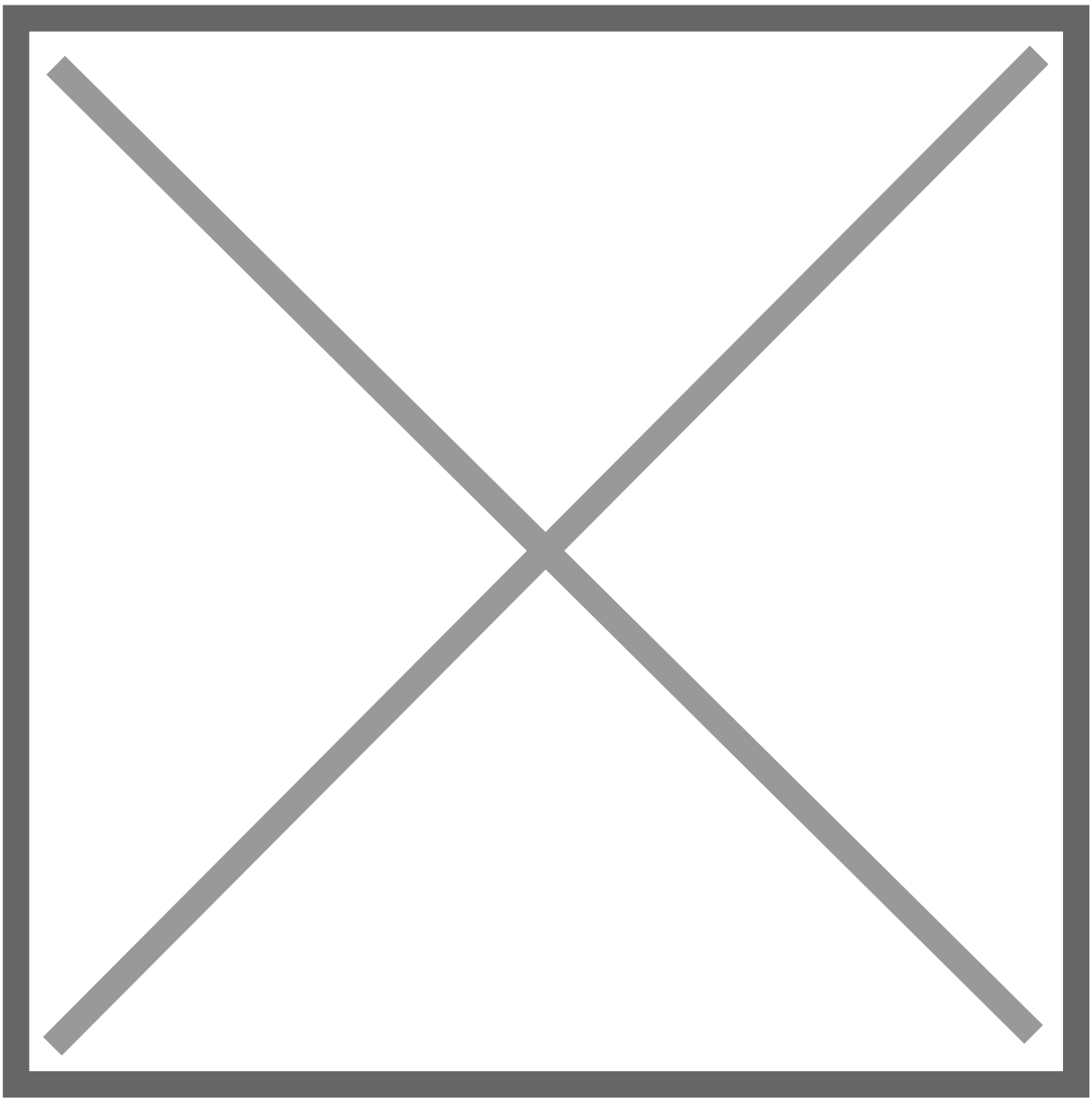
S nastavením úložiště můžete nyní přidat nějaké záložní složky a vytvořit svou první zálohu. Přejděte na kartu Zdroje a přidejte nějaké složky nebo výjimky.



Poté stiskněte **Spustit zálohování** a proveďte první zálohu. Po každé úspěšné záloze (nebo snímku) bude na **Snímek** záložku přidán nový řádek. Tam můžete také připojit snímek a obnovit soubory.

Obnovte soubory ze zálohy

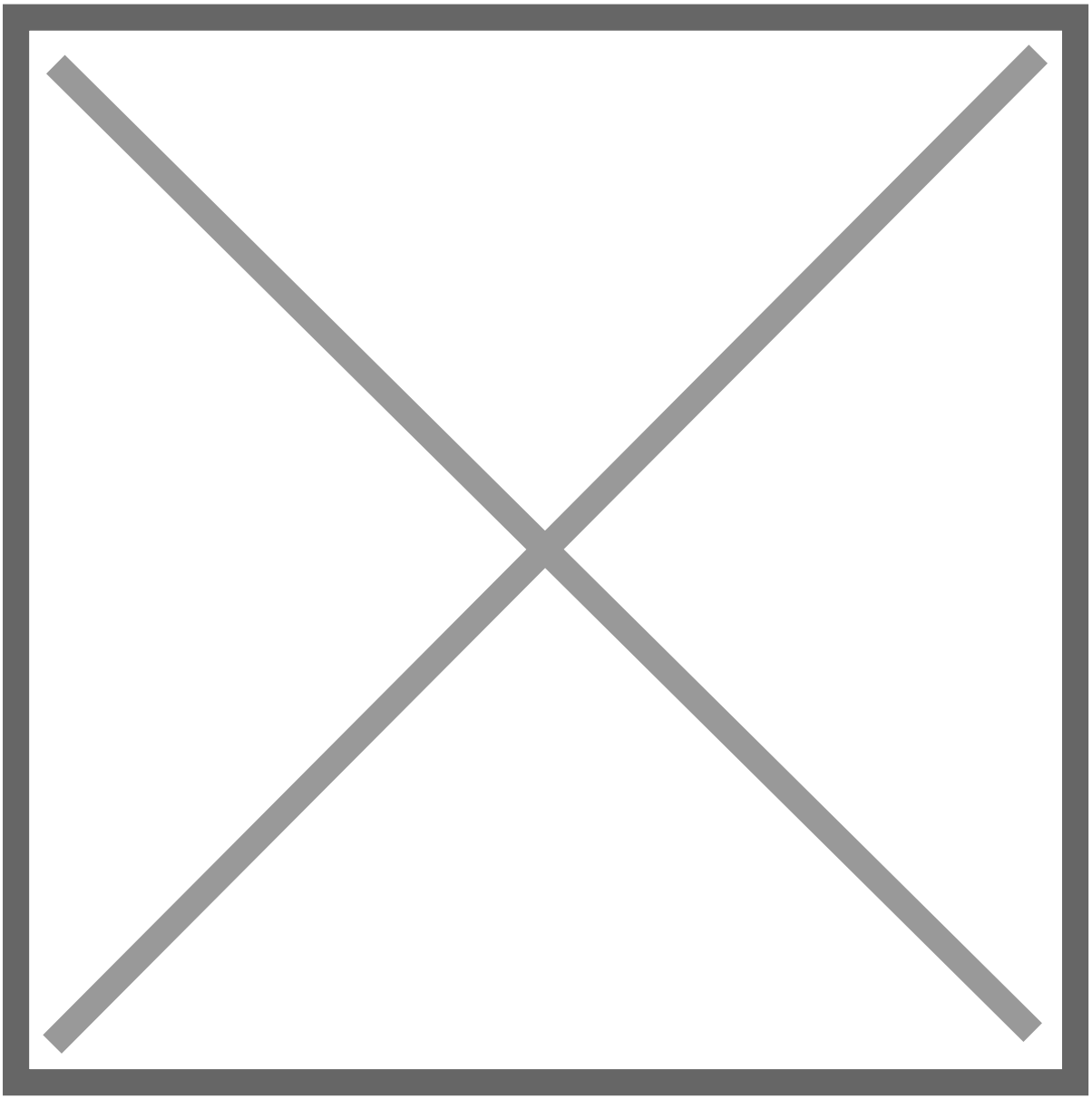
Chcete-li začít s obnovou snímku souborů, spusťte Vorta a klikněte na **kartu Archivy** .



Existují dva způsoby, jak obnovit soubory pomocí Vorta, *Extrahování* a *Připojení* . Oba dosahují stejného cíle, ale jeden může být výhodnější než druhý v závislosti na situaci.

Extrahování z archivu

Po otevření **karty Archivy** ve Vortě a výběru snímku, ze kterého chcete obnovit, klikněte na **tlačítko Rozbalit** v boční části a zobrazí se následující okno.



V dalším okně, které se zobrazí, se zobrazí dotaz na *Bod extrahování* nebo umístění, do kterého chcete zálohu extrahovat. Vyberte umístění a pokračujte.

Připojení zálohy

Po otevření **karty Archivy** ve Vortě a výběru snímku, ze kterého chcete obnovit, klikněte na **tlačítko Připojit** ve spodní části. Vyberte adresář pro připojení, bude to adresář, který Vorta použije, aby vám umožnil přístup k této záloze, jako by byla stále ve vašem souborovém systému. Ujistěte se, že se jedná o prázdný adresář.

Po dokončení obnovy souborů zavřete všechna aktivní okna a stiskněte **tlačítko Odpojit** .

Jednoduché zálohování pomocí Rsync

Co je rsync

Rsync je aplikace pro unixové systémy, která dokáže synchronizovat soubory a složky z jednoho umístění do jiného. Přitom, pokud je to možné, minimalizuje objem přenášených dat pomocí delta encodingu.

V Arch Linuxu můžete použít rsync k zálohování souborů na externí disk nebo na vzdálený server.

Může být použit k synchronizaci dat mezi dvěma umístěními, jako je například zálohování dat z disku na externí úložiště.



Instalace Rsync

Nejprve musíte nainstalovat balíček rsync, pokud již není nainstalován. Pro instalaci použijte příkaz:

```
sudo pacman -S rsync
```

- Poté připojte externí disk nebo přistupte k vzdálenému serveru, ke kterému chcete zálohovat data.
- Spusťte příkaz rsync s následujícími parametry:

```
rsync -avz --delete /home/user/Documents/ /mnt/backup/Documents/
```

Tento příkaz zkopíruje obsah adresáře "/home/user/Documents" do složky "/mnt/backup/Documents" s možností synchronizace souborů a adresářů, které se změnily od posledního spuštění. Volba "-**a**" zajistí, že se zachovají všechny atributy souborů, jako jsou oprávnění, časy a vlastníci. Volba "**-v**" zobrazí informace o průběhu zálohování a volba "**-z**" komprimuje přenos dat, což může urychlit přenos, pokud zálohujete na vzdálené zařízení. Volba "**--delete**" zajistí, že soubory a adresáře, které byly odstraněny ze zdrojového adresáře, budou také odstraněny z cílového umístění.

```
rsync -av --delete -e ssh /cesta/k/souborum/
jmenouzivatele@adresa:/cesta/k/zalohovacimu/ulozisti
```

Pokud potřebujete přihlašovací údaje k vzdálenému serveru, použijte tento příkaz:

- `-e ssh` - používá ssh pro připojení k vzdálenému serveru
- `jmenouzivatele` - vaše uživatelské jméno na vzdáleném serveru
- `adresa` - adresa vzdáleného serveru
- `/cesta/k/zalohovacimu/ulozisti/` - cesta k složce na vzdáleném serveru, kam chcete data zálohovat.

Automatické zálohy

Zde je příklad skriptu pro automatické zálohy **Rsync** v Arch Linux:

Otevřete editor textu (např. nano) a vytvořte nový soubor:

```
sudo nano /usr/local/bin/rsync-backup.sh
```

Do souboru vložte následující skript:

```
#!/bin/bash

# Nastavte zdrojovou a cílovou složku
SOURCE="/home/user/Documents/"
DESTINATION="/mnt/backup/Documents/"

# Spusťte příkaz rsync s nastavením pro zálohování
rsync -avz --delete "$SOURCE" "$DESTINATION"
```

Uložte soubor a uzavřete textový editor

Nastavte soubor jako spustitelný:

```
sudo chmod +x /usr/local/bin/rsync-backup.sh
```

Nakonfigurujte plánovač úloh **cron**, aby spouštěl skript s určitou četností (např. každý den v 2:00 ráno):

```
sudo crontab -e
```

Vložte následující řádek do konfiguračního souboru **cron** (změňte čas spouštění podle potřeby):

```
0 2 * * * /usr/local/bin/rsync-backup.sh
```

Uložte soubor a uzavřete editor textu.

Skript by nyní měl automaticky spouštět zálohu každý den v 2:00 ráno. Pokud chcete změnit nastavení zdrojové a cílové složky, jednoduše upravte proměnné `SOURCE` a `DESTINATION` ve skriptu.

Zálohování pomocí Rdiffbackup

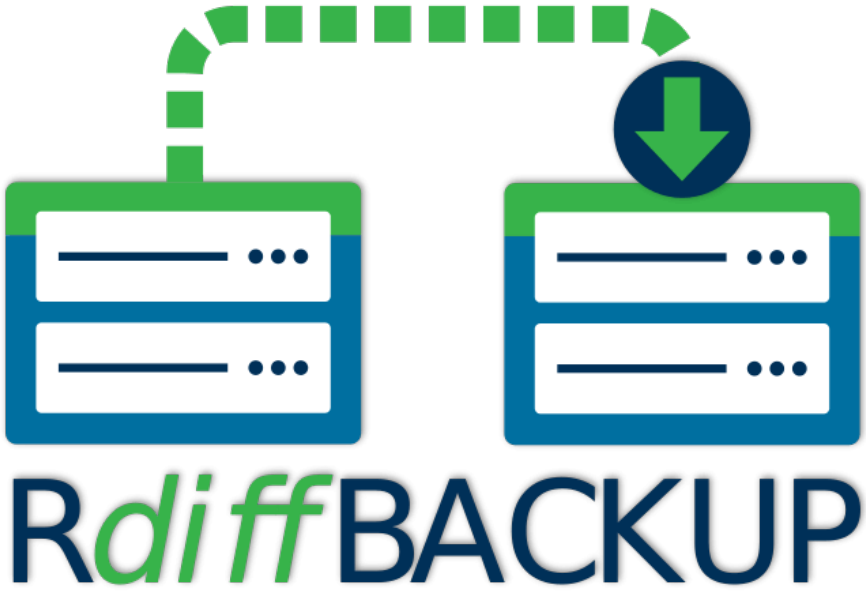
O Rdiff Backup

Rdiff Backup je open source software pro zálohování dat, který je navržen tak, aby byl spolehlivý a efektivní. Tato aplikace poskytuje uživatelům možnost vytvářet zálohy souborů a adresářů na různých úložištích, včetně externích pevných disků a síťových úložišť.

Hlavní výhodou Rdiff Backupu je, že umožňuje uživatelům vytvářet inkrementální zálohy, což znamená, že pouze změny provedené na souborech a adresářích od poslední zálohy jsou uloženy. Tento přístup šetří čas a místo na úložišti, což znamená, že uživatelé mohou udržovat svá data v bezpečí bez nutnosti vytvářet kompletní zálohy, které by mohly zabírat výrazné množství místa na disku.

Další výhodou Rdiff Backupu je jeho schopnost obnovit data z libovolného bodu v čase. To znamená, že pokud uživatel zjistí, že některé soubory nebo adresáře byly smazány nebo poškozeny, může použít Rdiff Backup k obnovení dat ze zálohy. Uživatelé mohou také snadno porovnávat různé verze souborů a adresářů, aby zjistili, co se od jedné zálohy změnilo.

Vývojáři Rdiff Backupu dbají na bezpečnost uživatelů, což znamená, že aplikace je navržena tak, aby minimalizovala riziko ztráty dat nebo úniku citlivých informací. Aplikace je také vysoce konfigurovatelná, což umožňuje uživatelům upravovat nastavení aplikace tak, aby vyhovovalo jejich specifickým potřebám.



Instalace Rdiff Backup

Nejprve musíte nainstalovat Rdiff Backup na svůj počítač. Pokud používáte Arch Linux, můžete Rdiff Backup nainstalovat z oficiálních repozitářů pomocí následujícího příkazu v příkazovém řádku:

```
sudo pacman -S rdiff-backup
```

Vytvoření zálohy

2. Nyní můžete začít vytvářet zálohy. Použijte následující příkaz v příkazovém řádku pro vytvoření zálohy

```
rdiff-backup /cesta/k/slozce /cesta/k/ulozisti
```

Místo "/cesta/k/slozce" nahraďte cestou ke složce, kterou chcete zálohovat, a místo "/cesta/k/ulozisti" nahraďte cestou k místu, kde chcete uložit zálohu. Tento příkaz vytvoří plnou zálohu složky.

Obnovení zálohy

Pokud chcete obnovit zálohu, použijte následující příkaz:

```
rdiff-backup --restore-as-of "datum a čas" /cesta/k/ulozisti /cesta/k/nova_slozka
```

Místo "datum a čas" nahraďte datum a čas, kdy jste provedli zálohu, "/cesta/k/ulozisti" nahraďte cestou k uložišti a "/cesta/k/nova_slozka" nahraďte cestou k nové složce, do které chcete obnovit zálohu.

Některé příkazy v Rdiffbackup

1. **rdiff-backup --backup-direcory source-dir backup-dir**
   Tento příkaz spustí zálohování zdrojového adresáře "source-dir" do cílového adresáře "backup-dir".
2. r**diff-backup --restore-as-of date backup-dir restored-dir** Tento příkaz obnoví adresář "restored-dir" do stavu, který odpovídá zálohovací verzi z "backup-dir" z datumu "date".
3. rd**iff-backup --list-increments backup-di**r Tento příkaz vypíše seznam všech zálohovacích inkrementálních souborů v adresáři "backup-dir".
4. rd**iff-backup --remove-older-than date backup-dir** Tento příkaz odstraní všechny zálohovací soubory starší než "date" z adresáře "backup-dir".
5. **rdiff-backup --exclude pattern source-dir backup-dir** Tento příkaz vyloučí soubory, které odpovídají "pattern" z adresáře "source-dir" během zálohování do "backup-dir".

Důležité odkazy

- Dokumentace
- GitHub