

# Vchanger jako wirtualna biblioteka taśmowa (VTL)

16 listopad 2011 autor: **gani**

Vchanger jest wolnodostępnym narzędziem emulującym taśmowe urządzenie ze zmieniarką (autoloader, biblioteka taśmowa) używając do tego wyznaczonej przestrzeni dyskowej. Artykuł przedstawia opis instalacji i konfiguracji programu Vchanger wraz z Bacula.

## Wstęp

Od czasu pojawienia się w branży rozwiązań kopii zapasowych takiego narzędzia jak **wirtualna biblioteka taśmowa**, świat aplikacji do backupu zyskał możliwość wykonywania kopii zapasowych na emulowane urządzenia taśmowe z prędkością dysków twardej. Taka właściwość wirtualnych bibliotek taśmowych stanowi alternatywę dla typowego magazynu danych „dysk-jako-dysk” a za sprawą emulacji dysków jako magazynów taśmowych nazywane jest niejednokrotnie rozwiązaniem „dysk-jako-taśma”. Przyjęło się również nazywać wirtualne biblioteki taśmowe skrótem **VTL (Virtual Tape Library)**.

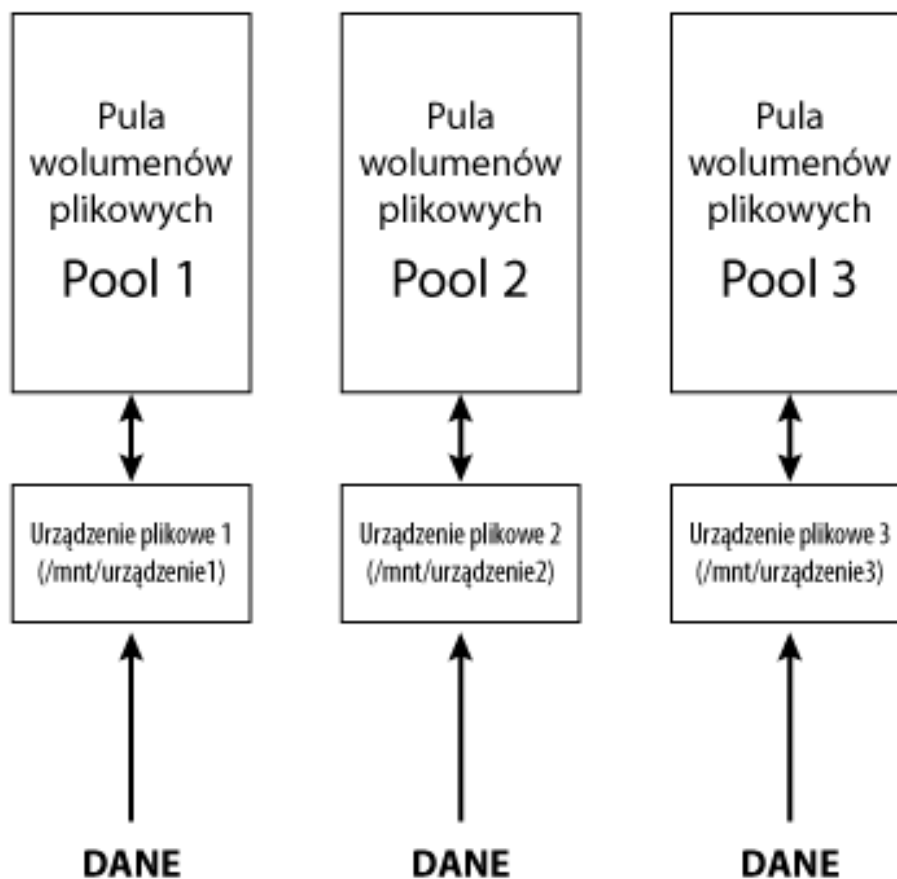
## Czym jest Vchanger

Udostępniony na zasadach Powszechnej Licencji Publicznej GNU (GPL) program **Vchanger** jest wolnodostępnym narzędziem emulującym taśmowe urządzenie ze zmieniarką (**autoloader, biblioteka taśmowa**) używając do tego wyznaczonej przestrzeni dyskowej z dowolnego źródła (np. katalogi, partycje, całe dyski, wymienne napędy dyskowe, macierze RAID). Doskonale wpasowuje się w środowisko Bacula współpracując razem z nią poprzez interfejs zgodny z interfejsem dostarczonym poprzez skrypt powłoki daemona magazynowania (ang. Storage Daemon) o nazwie „**mtx-changer**”. Można powiedzieć, że **Vchanger** wypełnia lukę w funkcjonalności Bacula jaką jest emulacja urządzeń taśmowych. Bacula co prawda dostarcza skrypt o nazwie „**disk-changer**”, który to ma pełnić rolę zbliżoną do funkcjonalności **Vchanger**, lecz nie jest on zalecany do używania w środowiskach produkcyjnych. Zdecydowanie bardziej do tego celu nadaje się **Vchanger**.

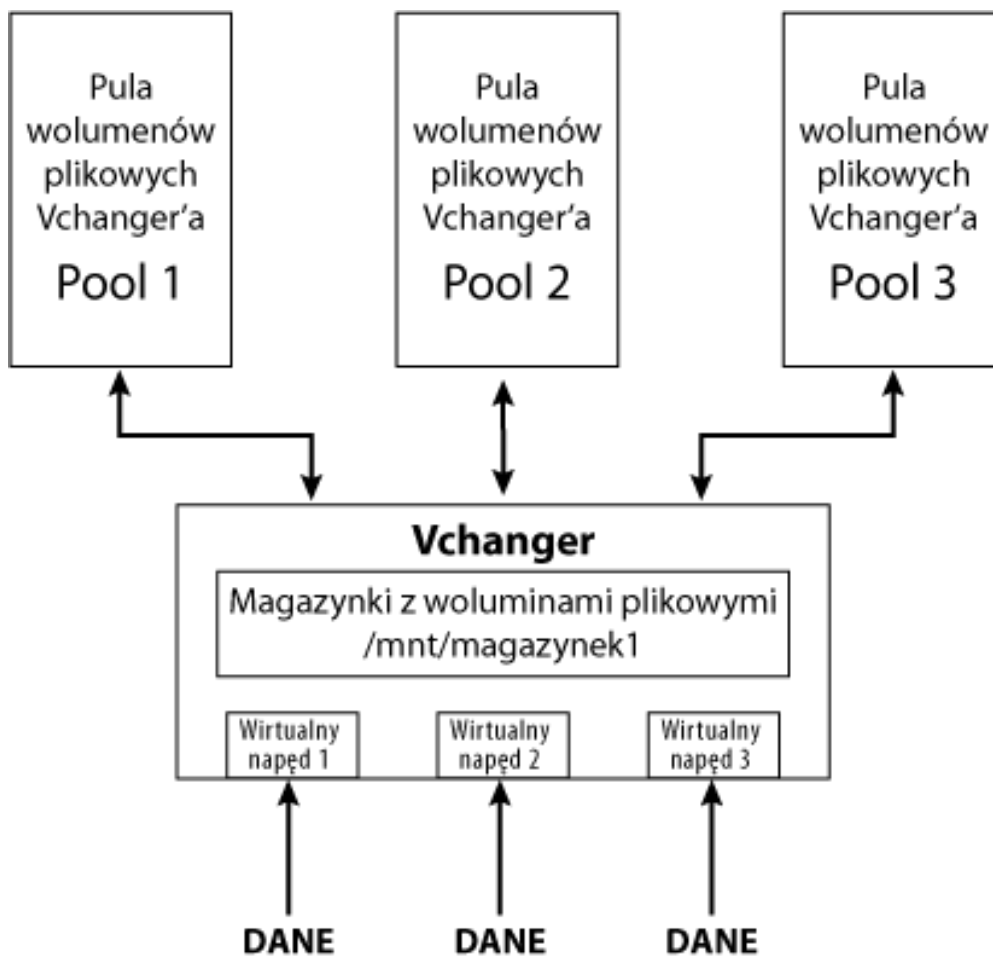
## Dlaczego właśnie Vchanger?

Jedną z właściwości Baculi w operacji na wolumenach plikowych (taśmowych również) jest to, że każde zdefiniowane urządzenie może jednocześnie mieć podmontowany jeden wolumen. Oznacza to tym samym, że jeśli użytkownik zachce wykonywać 7 backupów w tym samym czasie na urządzenia plikowe, to koniecznym jest, aby posiadał on zdefiniowane 7 urządzeń plikowych. Proszę sobie teraz wyobrazić sytuację, gdyby konieczne było wykonanie 77 równoczesnych backupów na urządzenia plikowe. Wiązałoby to się ze zdefiniowaniem co najmniej 77 urządzeń plikowych.

**Vchanger** ułatwia realizację zapisu i odczytu z i na nośniki danych z jednocześnie uruchomionych zadań (np. **backup, restore**) poprzez udostępnienie jednego urządzenia z wieloma napędami, które to urządzenie traktowane jest przez Baculę jako urządzenie biblioteki taśmowej. Za sprawą emulacji napędów oraz zmieniarki użytkownik ma do dyspozycji mocno elastyczne w konfiguracji urządzenia z tyloma napędami, magazynkami oraz slotami ile tylko będzie mu potrzebne. Chciałbym położyć tu specjalny nacisk na wielość napędów, gdyż to ona gwarantuje to, że jedno urządzenie Vchanger'a może realizować wiele zadań jednocześnie i tym samym mieć podmontowane wiele taśm plikowych (każda taśma w jednym napędzie). Poniżej zamieszczam rysunki ilustrujące tradycyjne podejście do urządzeń plikowych (Rysunek 1) oraz podejście realizowane przy użyciu Vchanger'a (Rysunek 2).



**Rysunek 1:** Jednoczesne trzy backupy na trzy urządzenia plikowe (podejście tradycyjne).



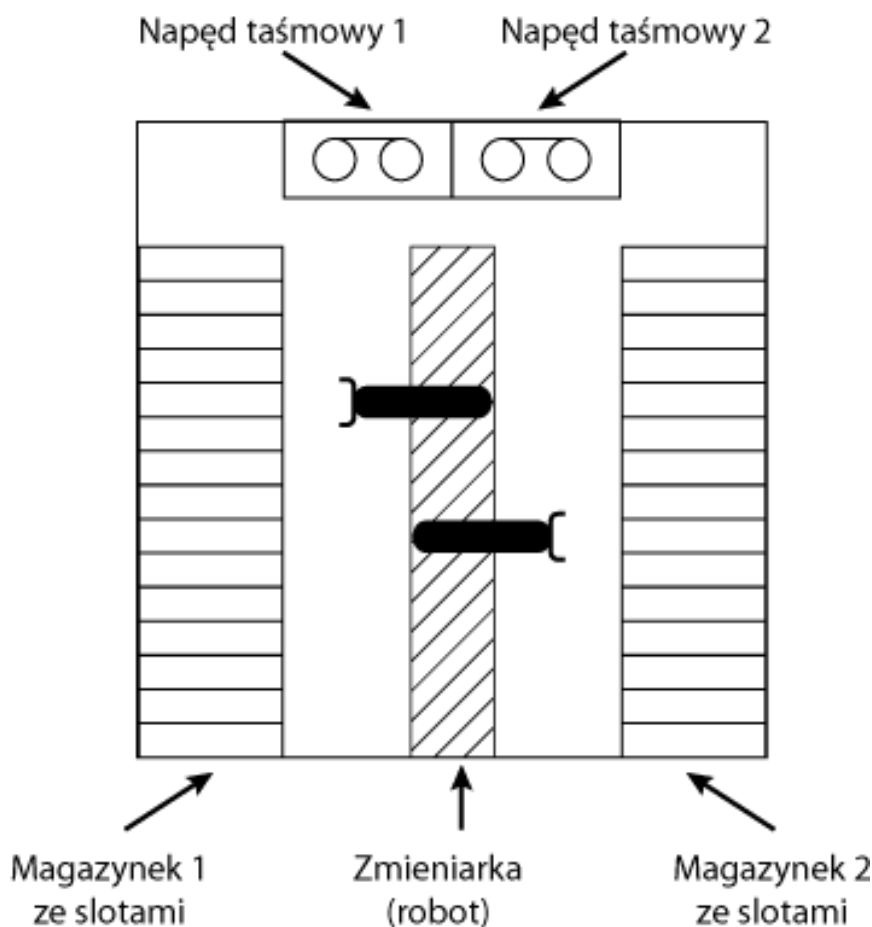
**Rysunek 2:** Jednoczesne trzy backupy na jedno urządzenie Vchanger'a z wykorzystaniem trzech wirtualnych napędów Vchanger'a.

Na powyższych rysunkach można zauważyć to, że przy wykorzystaniu tradycyjnego podejścia, gdzie każdy backup na wolumen realizowany jest poprzez jedno urządzenie plikowe, potrzeba aż trzech urządzeń. Przy użyciu Vchangera jest jedno urządzenie, gdzie **wirtualny changer** podaje taśmy z magazynka do odpowiednich wirtualnych napędów taśmowych. Jest to typowe zachowanie urządzenia taśmowego ze zmieniarką (**autoloader**, **biblioteka taśmowa**) i o to właśnie w **Vchanger** chodzi.

## Jak to działa

Przed przystąpieniem do skonfigurowania Vchangera warto zapoznać się z elementami, którymi on dysponuje oraz zależnością pomiędzy nimi.

Przede wszystkim trzeba rozgraniczyć cztery warstwy Vchangera: **sloty**, **magazynki**, **zmieniarka** oraz **napędy taśmowe**. Jako, że elementy te i ich zastosowanie są analogiczne do tych występujących w rzeczywistej bibliotece taśmowej, poniżej (Rysunek 3) zamieszczam bardzo uproszczony schemat **rzeczywistej biblioteki taśmowej**.



**Rysunek 3:** Uproszczony schemat urządzenia najprostszej biblioteki taśmowej.

Na Rysunku 3 można zauważyć następujące elementy:

- **Magazynki** - urządzenie ze schematu wyposażone jest w dwa magazynki o nazwach „Magazynek 1” oraz „Magazynek 2”, a każdy z nich zawiera po 15 slotów z taśmami. Magazynki te są wymienne w znaczeniu, że użytkownik biblioteki taśmowej może wyciągać je i wymieniać taśmy w slotach na nowe np. gdy obecne zostaną zapełnione danymi.
- **Sloty** - określane jako elementy składowe magazynków. Na załączonym schemacie każdy magazynek zwiera po 15 slotów, co tym samym oznacza, że jeden magazynek może zawierać 15 taśm. Biblioteka ze schematu zawiera dwa magazynki, więc łącznie może ona pomieścić 30 taśm (2 magazynki \* 15 taśm = 30 taśm).

- **Napędy taśmowe** - zadaniem tych urządzeń jest odczyt i zapis danych taśm biblioteki taśmowej. Od ich ilości zależy to, ile zadań jednocześnie może realizować biblioteka taśmowa.
- **Zmieniarka taśm (robot)** - wyciąga taśmy ze slotów i załadowuje je do napędów taśmowych, a po operacji na taśmach wykonuje operację odwrotną, tzn. wysuwa taśmy z napędów taśmowych i z powrotem dostarcza taśmy do swoich magazynków.

## Wymagania instalacyjne Vchanger

Do kompilacji oraz poprawnej pracy Vchanger wymaga następujących bibliotek:

- libblkid
- libuuid

Podane nazwy są nazwami, które występują w dystrybucji GNU/Linux Debian. Użytkownicy innych dystrybucji GNU/Linux lub innych systemów operacyjnych będą potrzebowali rozejrzeć się w repozytoriach w poszukiwaniu tych bibliotek wraz z plikami nagłówkowymi.

Do zainstalowania potrzebnych bibliotek i ich plików nagłówkowych używam Debian'owej komendy:

```
apt-get install libuuid1 libblkid1 libblkid-dev uuid-dev
```

## Kompilacja i instalacja Vchanger

Na początek trzeba zaopatrzyć się w archiwum ze źródłami Vchanger. Można tego dokonać wprost z oficjalnej strony internetowej Vchanger, która znajduje się w serwisie SourceForge.net pod adresem:

<http://sourceforge.net/projects/vchanger/>

W chwili pisania artykułu stabilną wersją **Vchanger** jest wersja oznaczona numerem 0.8.6.

Po ściągnięciu archiwum ze źródłami, rozpakowuję je

```
tar zxvf vchanger-0.8.6.tar.gz
```

następnie wchodzę do katalogu źródeł

```
cd vchanger/
```

i przystępuję do konfiguracji źródeł:

```
./configure --prefix=/usr/local/bacula/vchanger
```

gdzie:

--**prefix** - oznacza lokalizację, w której zostaną zainstalowane pliki Vchanger'a.

Po poprawnie zakończonej konfiguracji źródeł wykonuję ich kompilację poprzez komendę:

```
make
```

oraz instalację poprzez komendę:

```
make install-strip
```

Po tych operacjach w lokalizacji

```
/usr/local/bacula/vchanger/
```

otrzymałem dwa katalogi:

```
/usr/local/bacula/vchanger/bin  
/usr/local/bacula/vchanger/share
```

gdzie:

**bin** - jest katalogiem z programem Vchanger

**share** - zawiera dokumentację programu Vchanger w formacie HTML oraz w plikach tekstowych.

## Konfiguracja Vchanger

Plik konfiguracyjny programu Vchanger trzeba stworzyć samodzielnie, np. w następujący sposób:

```
mkdir /usr/local/bacula/vchanger/etc  
cd /usr/local/bacula/vchanger/etc  
touch vchanger.conf
```

Plik ten można stworzyć w dowolnej lokalizacji. Będzie on wskazywany z absolutną ścieżką jako parametr do wywołania programu Vchanger przez Baculę. Lecz o tym opowiem za chwilę.

Wracając do konfiguracji Vchanger, edytuję stworzony przed chwilą plik konfiguracyjny o nazwie vchanger.conf. Możliwe w nim są do wpisania następujące opcje:

- **changer\_name** = nazwaWirtualnejBibliotekiTaśmowej - jest to dowolna nazwa, jaką użytkownik zachce nazwać swoje **VTL**.
- **work\_dir** = lokalizacja - w opcji tej trzeba wskazać lokalizację, w której Vchanger będzie składował metadane opisujące stan wirtualnych napędów oraz dowiązania symboliczne do aktualnie używanych magazynków (a dokładniej katalogów z magazynkami). Pliki z metadanymi wirtualnych napędów taśmowych są kilkubajtowymi plikami tekstowymi, więc wybrany w opcji work\_dir katalog nie musi znajdować się na nośniku o dużej przestrzeni dyskowej. Tak jak wspomniałem, to zaledwie kilka do kilkudziesięciu bajtów w zależności od ilości napędów i złożoności nazwy Vchangera.
- **log\_file** = lokalizacja - w tej opcji wskazuje się lokalizację pliku dziennika programu Vchanger. Może on okazać się użyteczny np. podczas diagnozowania problemów z Vchanger'em.
- **log\_level** = poziomLogowania - określa to, jak dokładne mają być wiadomości składowane w pliku dziennika Vchangera. Możliwe do wyboru są m.in. poziomy logowania o nazwach: LOG\_EMERG, LOG\_CRIT, LOG\_ERR. Jeśli nie ustawiono dyrektywy log\_level to domyślnym poziomem logowania będzie LOG\_ERR.
- **virtual\_drives** = liczba - określa liczbę wirtualnych napędów, która będzie dostępna w urządzeniu Vchanger'a. To jeden z ważniejszych punktów całej konfiguracji, gdyż od wybranej ilości wirtualnych napędów taśmowych uzależnione będzie to, ile zadań (np. **backupów**) będzie mogło być wykonywanych jednocześnie przy użyciu urządzenia Vchanger'a.

- **slots\_per\_magazine** = liczba - wyraża ilość slotów z taśmami na każdy magazynek. Vchanger operuje na magazynkach ze stałą ilością slotów z taśmami. To znaczy, że jeśli zdefiniuje się w opisywanej opcji slots\_per\_magazine ilość slotów jako osiem, to każdy z użytych magazynków będzie miał po osiem slotów z taśmami.
- **magazine\_bays** = liczba - ta opcja definiuje możliwą do użycia w tym samym czasie ilość magazynków. Np. jeśli ilość magazynków zdefiniowana w tej dyrektywie to 4, znaczy to, że jednocześnie Vchanger może operować na 4 magazynkach. Jeśli nie używa się wymiennych dysków jako kontenera na magazynki, lecz stałe dyski, to ilość wyrażona w tej dyrektywie będzie całkowitą liczbą wykorzystywanych magazynków. Jeśli natomiast użytkownik zdecyduje się np. na użycie osobnego wymiennego dysku na każdy magazynek, to wartość zdefiniowana w magazine\_bays będzie oznaczała ilość magazynków jakie jednocześnie może podmontować i tym samym obsłużyć Vchanger.
- **automount\_dir** = lokalizacja - opcja ta ma znaczenie w przypadku używania wymiennych dysków jako magazynków Vchangersa. Określa wspólny dla magazynków katalog, w którego podkatalogach będą automatycznie montowane dyski z magazynkami. W niniejszym artykule posłużę się lokalizacją magazynków na stałych dyskach, lecz uważam, że warto znać tą opcję, gdyby użytkownik potrzebował użyć wymiennych dysków z magazynkami.
- **magazine** = lokalizacja/UUID dysku - lokalizacja katalogów przeznaczonych na magazynki lub identyfikatory dysków w przypadku używania wymiennych dysków. Opcja może być definiowana wielokrotnie w ramach pliku konfiguracyjnego i powinna być zapisana tyle razy, ile magazynków będzie używał Vchanger. Od ilości użytych magazynków zależeć będzie również to, ile taśm będzie możliwych do użycia przez Vchanger. Odpowiada za to wzór:

ilość taśm na magazynek (slots\_per\_magazine) \* ilość magazynków (ilość opcji „magazine”) = całkowita ilość taśm Vchanger’a.

W przypadku podania w tej dyrektywie identyfikatorów dysków (**UUID**) będą one automatycznie montowane w systemie.

Tak jak wspomniałem, w mojej konfiguracji Vchanger’a posłużę się stale podmontowanym dyskiem (a dokładniej będzie to programowa macierz **RAID 1**). Na nim będę składował cztery magazynki (każdy po **15 slotów**) oraz wyposażę urządzenie Vchangersa w cztery wirtualne napędy taśmowe.

Edytuję stworzony wcześniej plik konfiguracyjny w lokalizacji

/usr/local/bacula/vchanger/etc/vchanger.conf

i wpisuję do niego następującą zawartość:

```
changer_name = VTL
work_dir = /media/backup/VTL
logfile = /var/log/vchanger.log
log_level = LOG_ERR
virtual_drives = 4
slots_per_magazine = 15
magazine_bays = 4
magazine = /media/backup/VTL/magazine1
magazine = /media/backup/VTL/magazine2
```

```
magazine = /media/backup/VTL/magazine3
magazine = /media/backup/VTL/magazine4
```

gdzie punktem montowania stale podmontowanego dysku jest lokalizacja:

```
/media/backup/
```

Po skonfigurowaniu Vchanger'a w pliku konfiguracyjnym, przygotowuję katalogi na magazynki poprzez wydanie komend:

```
mkdir /media/backup/VTL
mkdir /media/backup/VTL/magazine1
mkdir /media/backup/VTL/magazine2
mkdir /media/backup/VTL/magazine3
mkdir /media/backup/VTL/magazine4
```

Wszystkie te komendy wykonuję z konta super-użytkownika root, gdyż mój Daemon magazynowania (Storage Daemon) działa właśnie z tego użytkownika systemowego. Gdyby Storage Daemon działał z innego użytkownika niż root, wtedy potrzeba by nadać odpowiedniego użytkownika i grupę w/w katalogom np. poprzez wydanie komendy

```
chown -R jakis _user:jakas _grupa /media/backup/VTL
```

Przed podłączeniem urządzenia Vchanger do Bacula potrzeba jeszcze zainicjalizować magazynki. Nie jest to nic innego jak utworzenie struktury plików, które posłużą jako woluminy plikowe. Inicjalizacji dokonuje się poprzez program vchanger w następujący sposób:

```
/lokalizacja/programu/vchanger -u uzytkownik _storage _daemon -g
grupa _storage _daemon /lokalizacja/pliku/konfiguracyjnego/vchanger.
conf INITMAG numer _magazynka
```

Wykonuję inicjalizację czterech magazynków zdefiniowanych w pliku konfiguracyjnym vchanger.conf.

```
/usr/local/bacula/vchanger/bin/vchanger -u root -g root /usr/local/
bacula/vchanger/etc/vchanger.conf INITMAG 1
created magazine 1 in bay 1 [/media/backup/VTL/magazine1]
/usr/local/bacula/vchanger/bin/vchanger -u root -g root /usr/local/
bacula/vchanger/etc/vchanger.conf INITMAG 2
created magazine 2 in bay 2 [/media/backup/VTL/magazine2]
/usr/local/bacula/vchanger/bin/vchanger -u root -g root /usr/local/
bacula/vchanger/etc/vchanger.conf INITMAG 3
created magazine 3 in bay 3 [/media/backup/VTL/magazine3]
/usr/local/bacula/vchanger/bin/vchanger -u root -g root /usr/local/
bacula/vchanger/etc/vchanger.conf INITMAG 4
created magazine 4 in bay 4 [/media/backup/VTL/magazine4]
```

Sprawdzam czy struktura została utworzona poprawnie poprzez wylistowanie zawartości katalogów z magazynkami (poniżej listing z magazynku numer 4)

```

ls -l /media/backup/VTL/magazine4/
razem 20
-rw-r--r-- 1 root root 8 2011-11-05 21:52 index
-rw-r--r-- 1 root root 1 2011-11-05 21:52 loaded0
-rw-r--r-- 1 root root 1 2011-11-05 21:52 loaded1
-rw-r--r-- 1 root root 1 2011-11-05 21:52 loaded2
-rw-r--r-- 1 root root 1 2011-11-05 21:52 loaded3
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0001
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0002
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0003
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0004
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0005
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0006
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0007
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0008
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0009
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0010
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0011
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0012
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0013
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0014
-rw-r----- 1 root root 0 2011-11-05 21:52 VTL_0004_0015

```

Struktura woluminów plikowych urządzenia Vchanger została stworzona pomyślnie. Teraz pozostaje już tylko zdefiniować urządzenie Vchanger w pliku konfiguracyjnym Storage Daemon'a oraz Director'a. Moja konfiguracja wygląda jak poniżej.

Fragment pliku konfiguracyjnego bacula-sd.conf

```

Autochanger {
    Name = „Virtual Tape Library”
    Device = Drive-1
    Device = Drive-2
    Device = Drive-3
    Device = Drive-4
    Changer Command = „/usr/local/bacula/vchanger/bin/vchanger -u root -g
root %c %o %S %a %d”
    Changer Device = „/usr/local/bacula/vchanger/etc/vchanger.conf”
}
Device {
    Name = Drive-1
    Device Type = File
    Drive Index = 0
    Media Type = Plik
    Archive Device = /media/backup/VTL/0/drive0
    RemovableMedia = no
    RandomAccess = yes
    Autochanger = yes
}
Device {
    Name = Drive-2

```



```
Device Type = File
Drive Index = 1
Media Type = Plik
Archive Device = /media/backup/VTL/1/drive1
RemovableMedia = no
RandomAccess = yes
Autochanger = yes
}
Device {
Name = Drive-3
Device Type = File
Drive Index = 2
Media Type = Plik
Archive Device = /media/backup/VTL/2/drive2
RemovableMedia = no
RandomAccess = yes
Autochanger = yes
}
Device {
Name = Drive-4
Device Type = File
Drive Index = 3
Media Type = Plik
Archive Device = /media/backup/VTL/3/drive3
RemovableMedia = no
RandomAccess = yes
Autochanger = yes
}
```

### Fragment pliku konfiguracyjnego bacula-dir.conf

```
Storage {
Name = VTL
SD Port = 9103
Address = 10.0.0.3
Password = „ZZZ”
Device = „Virtual Tape Library”
Autochanger = yes
Media Type = Plik
Maximum Concurrent Jobs = 4
}
```

## Vchanger a etykietowanie taśm

Dostarczone przez Vchanger magazynki mogą być etykietowane w taki sam sposób, co taśmy z rzeczywistej biblioteki taśmowej obsługujące etykiety zapisane kodem kreskowym na taśmie. Funkcjonalność ta w Bacula nazywana jest etykietowaniem z „barcode”, gdzie samo „barcode” to właśnie nazwa taśmy zapisana kodem kreskowym.

Posiadam cztery puste pule woluminów o nazwach „Full-VTL1”, „Full-VTL2”, „Full-VTL3” oraz „Full-VTL4”. Dla każdej z nich mogę zaetykietować taśmy plikowe Vchanger’a w dowolny

sposób. To, jak zagospodarowane mogą być taśmy urządzeń taśmowych nieco wykracza poza temat konfiguracji wirtualnej biblioteki taśmowej i nie zostanie tu omówiony. Zakładam więc wspomniane wyżej cztery puli woluminów i etykietuję do nich taśmy w taki sposób, aby taśmy z magazynka 1 znalazły się w puli woluminów „Full-VTL1”, taśmy z magazynka 2 znalazły się w puli woluminów „Full-VTL2”, taśmy z magazynka 3 znalazły się w puli woluminów „Full-VTL3” a taśmy z magazynka 4 znalazły się w puli woluminów „Full-VTL4”. W tym celu wydaję komendy jak poniżej.

```
label barcodes storage=VTL pool=Full-VTL1 drive=0 slots=1-15
label barcodes storage=VTL pool=Full-VTL2 drive=0 slots=16-30
label barcodes storage=VTL pool=Full-VTL3 drive=0 slots=31-45
label barcodes storage=VTL pool=Full-VTL4 drive=0 slots=46-60
```

## Przykładowe użycie Vchanger

Uruchomiłem 4 backupy o nazwie „src” z wykorzystaniem czterech różnych puli woluminów o nazwach „Full-VTL1”, „Full-VTL2”, „Full-VTL3” oraz „Full-VTL4”.

```
*run job=src storage=VTL pool=Full-VTL1 yes
Using Catalog „MyCatalog”
Job queued. JobId=38
*run job=src storage=VTL pool=Full-VTL2 yes
Job queued. JobId=39
*run job=src storage=VTL pool=Full-VTL3 yes
Job queued. JobId=40
*run job=src storage=VTL pool=Full-VTL4 yes
Job queued. JobId=41
```

Poniższy status urządzenia pokazuje, że wszystkie cztery backupy korzystają z wszystkich wirtualnych napędów plikowych (każdy backup z osobnego wirtualnego napędu plikowego)

```
*status storage=VTL
Connecting to Storage daemon VTL at 10.0.0.3:9103
darkstar-sd Version: 5.2.1 (30 October 2011) x86_64-unknown-linux-gnu
ubuntu 11.10
Daemon started 15-Nov-11 10:48. Jobs: run=1, running=3.
  Heap: heap=864,256 smbbytes=612,249 max_bytes=640,368 bufs=249 max_
bufs=251
  Sizes: boffset_t=8 size_t=8 int32_t=4 int64_t=8
  Running Jobs:
Writing: Full Backup job src JobId=38 Volume="VTL_0001_0009"
  pool="Full-VTL1" device="Drive-1" (/media/backup/VTL/0/drive0)
  spooling=0 despooling=0 despool_wait=0
  Files=89 Bytes=437,506,040 Bytes/sec=13,257,758
  FDReadSeqNo=7,427 in_msg=7174 out_msg=5 fd=5
Writing: Full Backup job src JobId=39 Volume="VTL_0001_0010"
  pool="Full-VTL2" device="Drive-2" (/media/backup/VTL/1/drive1)
  spooling=0 despooling=0 despool_wait=0
  Files=89 Bytes=403,558,392 Bytes/sec=13,915,806
  FDReadSeqNo=6,909 in_msg=6656 out_msg=5 fd=8
Writing: Full Backup job src JobId=40 Volume="VTL_0001_0011"
  pool="Full-VTL3" device="Drive-3" (/media/backup/VTL/2/drive2)
```

```
spooling=0 despooling=0 despool_wait=0
Files=89 Bytes=379,310,072 Bytes/sec=15,172,402
FDReadSeqNo=6,539 in _msg=6286 out _msg=5 fd=11
Writing: Full Backup job src JobId=41 Volume="VTL_0001_0007"
pool="Full-VTL4" device="Drive-4" (/media/backup/VTL/3/drive3)
spooling=0 despooling=0 despool_wait=0
Files=89 Bytes=266,653,688 Bytes/sec=20,511,822
FDReadSeqNo=4,820 in _msg=4567 out _msg=5 fd=14
====
Jobs waiting to reserve a drive:
====
[--WYCIECIE--]
Device status:
Autochanger „Virtual Tape Library” with devices:
„Drive-1” (/media/backup/VTL/0/drive0)
„Drive-2” (/media/backup/VTL/1/drive1)
„Drive-3” (/media/backup/VTL/2/drive2)
„Drive-4” (/media/backup/VTL/3/drive3)
Device „Drive-1” (/media/backup/VTL/0/drive0) is mounted with:
Volume:      VTL_0001_0009
Pool:        Full-VTL1
Media type:  Plik
Slot 9 is loaded in drive 0.
Total Bytes=437,778,627 Blocks=6,786 Bytes/block=64,512
Positioned at File=0 Block=437,778,626
Device „Drive-2” (/media/backup/VTL/1/drive1) is mounted with:
Volume:      VTL_0001_0010
Pool:        Full-VTL2
Media type:  Plik
Slot 10 is loaded in drive 1.
Total Bytes=403,845,315 Blocks=6,260 Bytes/block=64,512
Positioned at File=0 Block=403,845,314
Device „Drive-3” (/media/backup/VTL/2/drive2) is mounted with:
Volume:      VTL_0001_0011
Pool:        Full-VTL3
Media type:  Plik
Slot 11 is loaded in drive 2.
Total Bytes=379,588,803 Blocks=5,884 Bytes/block=64,512
Positioned at File=0 Block=379,588,802
Device „Drive-4” (/media/backup/VTL/3/drive3) is mounted with:
Volume:      VTL_0001_0007
Pool:        Full-VTL4
Media type:  Plik
Slot 7 is loaded in drive 3.
Total Bytes=266,821,827 Blocks=4,136 Bytes/block=64,512
Positioned at File=0 Block=266,821,826
====
Used Volume status:
VTL_0001_0007 on device „Drive-4” (/media/backup/VTL/3/drive3)
Reader=0 writers=1 devres=0 volinuse=1
VTL_0001_0009 on device „Drive-1” (/media/backup/VTL/0/drive0)
```

```
Reader=0 writers=1 devres=0 volinuse=1
VTL_0001_0010 on device „Drive-2” (/media/backup/VTL/1/drive1)
Reader=0 writers=1 devres=0 volinuse=1
VTL_0001_0011 on device „Drive-3” (/media/backup/VTL/2/drive2)
Reader=0 writers=1 devres=0 volinuse=1
====
```

## Podsumowanie

Pisząc powyższy artykuł wykorzystałem najprostszą konfigurację wirtualnego changer'a czyli jedną stale podmontowaną jednostkę, jaką jest programowa macierz **RAID1** złożona z dwóch jedno terabajtowych **dysków**. Użycie wraz z Vchanger wymowalnych dysków w większości nie różni się od opisanej tutaj instalacji i konfiguracji, a niniejszy artykuł może stanowić podstawę do własnych eksperymentów z vchanger.

Na koniec chciałbym zauważyć, że w kilku miejscach użyłem sformułowania „**VTL**” czy też „**wirtualna biblioteka taśmowa**” w odniesieniu do **Vchanger**. W dokumentacji Vchanger'a jego twórcy udowadniają, że Vchanger nie jest wirtualną biblioteką taśmową w ścisłym tego słowa znaczeniu. W moich wypowiedziach odnośnie **VTL** mam na myśli to mniej ścisłe znaczenie, czyli generalne pojęcie magazynu danych „*dysk-jako-taśma*”. Ot tyle.

Życzę sukcesów w konfiguracji Vchanger'a oraz wytrwałości w jej realizacji :-)