

# Wstęp do systemu operacyjnego UNIX

## Laboratorium 5:

### LVM Zarządca przestrzeni logicznej

Historycznie rzecz ujmując, rozmiar partycji miał charakter statyczny. Instalując system operacyjny, dokonywaliśmy podziału dysku na partycje i już w tym momencie musieliśmy każdej z nich przydzielić funkcje (systemowa, konta użytkowników, scratch) oraz przewidzieć konieczny rozmiar. Ponieważ wolne miejsce na dysku jest po papierze w drukarce najczęściej wyczerpującym się zasobem – zawsze zdarzało się tak, że któraś z partycji okazywała się za mała. W tej sytuacji możliwe były dwa wyjścia:

1. dokonanie nowego podziału na partycje, jeśli pozwalała na to zajętość pozostałych partycji na dysku,
2. zakup nowego większego dysku, podział na partycje oraz skopiowanie danych ze starego dysku.

W obu przypadkach należało wykonać kopie zapasową, a czynności wymuszały długie przerwy w pracy systemu. Dodatkowo maksymalny rozmiar partycji ograniczony był pojemnością dysku, gdyż nie mogła się ona rozciągać na więcej niż jeden dysk, ponadto partycja musi być ciągłym obszarem dysku co utrudnia zmiany w układzie partycji. Jako pierwsza problemy te rozwiązała firma **IBM** wprowadzając do systemu AIX w wersji 3.2 **Logical Volume Manager** (LVM) - zarządcę przestrzeni logicznej. Pomysł ten wprowadziły następnie do swoich systemów operacyjnych inne firmy w tym Hewlett-Packard, skąd LVM trafił do systemu *Linux*.

Do wyjaśnienia zasady działania LVM konieczne będzie zdefiniowanie kilku pojęć, wspólnych dla każdego z rozwiązań.

- **medium fizyczne** - pod tym pojęciem należy rozumieć pojedynczy dysk sztywny lub inne urządzenie blokowe. Media są reprezentowane w katalogu `/dev` jako pliki np: `/dev/hda`, `/dev/hda6` czy `/dev/sda`.
- **wolumen fizyczny, PV - *Physical Volume*** jest on medium fizycznym do którego dodano opis jego charakterystyki w postaci identyfikatora i danych fizycznych jak np. pojemność.
- **fizyczny extend, PE - *Physical Extend*** jest blokiem na które podzielony został wolumen fizyczny.
- **grupa wolumenów, VG - *Volume Group*** - tworzą ją wolumeny fizyczne (jeden lub więcej). Grupę mogą tworzyć jedynie te wolumeny fizyczne, w których rozmiar fizycznego extendu jest taki sam.
- **partycja logiczna, LP - *Logical Partition*** - najmniejsza jednostka organizacyjna (porcja, blok danych) zarządzana w systemie plików.
- **wolumen logiczny, LV - *Logical Volume*** - jest tworem abstrakcyjnym zbudowanym z fizycznych extendów należących do dowolnego wolumenu fizycznego, ale z tej samej grupy wolumenów. Wolumen logiczny z punktu widzenia użytkownika stanowi ciągłą przestrzeń w postaci urządzenia blokowego umożliwiającą przechowywanie danych.
- **system plików, *Filesystem*** - ciągła przestrzeń adresowa o zorganizowanym zapisie. W zależności od sposobu organizacji zapisu mamy: ext2, ext3, ReiserFS, NTFS itd.

Prześledźmy zatem teoretycznie proces tworzenia systemu plików. Załóżmy, że mamy do dyspozycji wolumen fizyczne podzielone na extendy fizyczne o jednakowych rozmiarach, jak na rysunku 1:

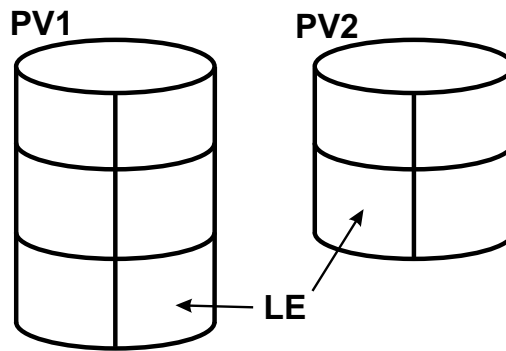
Oba wolumeny fizyczne łączymy w grupę wolumenów (rysunek 2):

Wykorzystując extendy fizyczne z obu wolumenów fizycznych budujemy wolumen logiczny jak na rysunku 3. Na powstałym w ten sposób wolumenie, będącym urządzeniem blokowym, możemy założyć system plików.

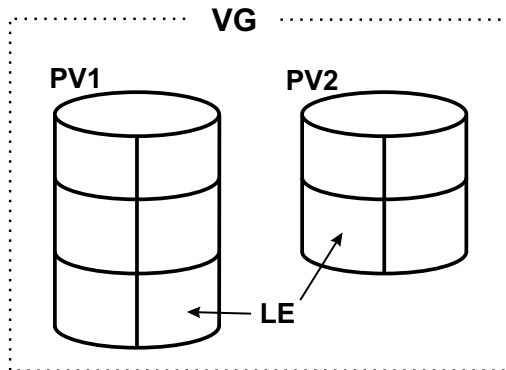
Posługując się analogią do dysków grupa wolumenów stanowi dysk zaś wolumen logiczny jest partycją w obrębie tego dysku. Analogicznie jak w przypadku dysków w systemie możemy skonfigurować wiele niezależnych od siebie grup wolumenów.

Po pewnym czasie wolumen logiczny okazał się za mały. Mamy jeszcze w grupie wolne extendy fizyczne, zatem dokładamy je do istniejącego wolumenu logicznego (rysunek 4). Po zwiększeniu rozmiaru urządzenia blokowego jakim jest wolumen logiczny możemy rozszerzyć znajdujący się na nim system plików. Nie jest konieczne tworzenie kopii systemu plików.

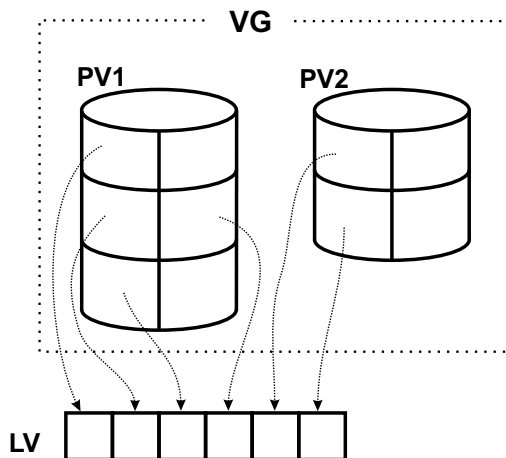
W grupie wolumenów zabrakło już wolnych extendów fizycznych. Do grupy wolumenów dokładamy kolejny wolumen fizyczny. Extendy wchodzące w skład dołączanego wolumenu stają się dostępne w grupie wolumenów i mogą zostać dowolnie wykorzystane (rysunek 5).



Rysunek 1: Wolumeny fizyczne



Rysunek 2: Grupa wolumenów

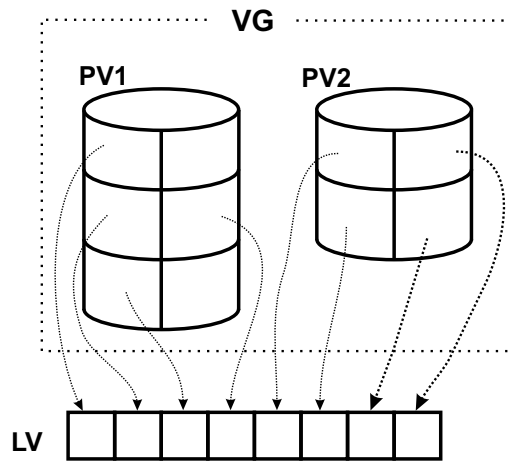


Rysunek 3: Wolumen logiczny

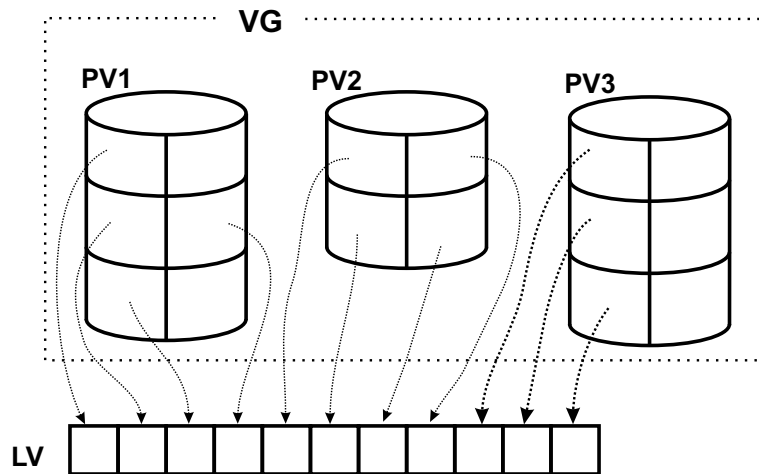
Poniżej przedstawimy przykład zastosowania LVM w systemie Linux. Zakładamy, że w systemie istnieją partycje nieużywane, widziane w systemie jako urządzenia `/dev/hda3` oraz `/dev/hdb2`. Dodatkowo zakładamy, że jądro systemu posiada możliwość obsługi LVM oraz systemu plików ext2.

W pierwszej kolejności, w tablicy partycji musimy zmienić identyfikator urządzenia. „Normalne”, nie zarządzane przez LVM urządzenie posiada identyfikator 0x83, zaś gotowe do obsługi przez LVM identyfikator 0x8e – Linux LVM. Zmianę wprowadzamy wykorzystując program `fdisk`.

1. Przy pomocy programu `fdisk` sprawdź, jakie urządzenie zostały zdefiniowane w systemie i jakie są ich identyfikatory. Wytypuj dwa do ewentualnego wprowadzenia LVM. Zmień ich identyfikator na 0x8e – Linux LVM. Jeśli w Twoim systemie nie ma nieużywanych partycji, utwórz je (trzy partycje). Zwracamy uwagę, że w systemie mogą być podłączone inne dyski niż `/dev/hda` i można



Rysunek 4: Zwiększenie rozmiaru wolumenu logicznego



Rysunek 5: Nowy wolumen fizyczny w grupie wolumenów

je wykorzystać.

Wracając do naszego przykładu, dokładnie to samo robimy dla urządzenia `/dev/hdb2`. Niektóre systemy wymagają w tym momencie restartu. Dla bezpieczeństwa wykonamy go.

**2. Dokonaj restartu systemu. Podłącz się ponownie jako użytkownik root.**

Ponieważ polecenia zarządzające podsystemem LVM zapisują konfigurację w pliku `/etc/lvmtab` oraz katalogu `/etc/lvmtab.d` należy zadbać o utworzenie wspomnianych plików. Można to zrobić wydając polecenie `vgscan`.

W tym momencie przykładowe urządzenia `/dev/hda3` oraz `/dev/hdb2` stanowią dla nas medium fizyczne. Są urządzeniami surowymi, których typ nie jest dla systemu znany. Musimy je zatem przekształcić w wolumeny fizyczne. Służy do tego komenda `pvcreate`:

```

1 # pvcreate /dev/hda3
2 pvcreate -- physical volume "/dev/hda3" successfully created
3 # pvcreate /dev/hdb2
4 pvcreate -- physical volume "/dev/hdb2" successfully created

```

**3. Zapoznaj się z manuałem komendy `pvcreate`. Wybrane przez siebie urządzenia przekształć w wolumeny fizyczne.**

Kolejny krok, to stworzenie pierwszej w systemie grupy wolumenów, do której wprowadzimy stworzone

przez nas wolumeny fizyczne. Służy do tego komenda `vgcreate` :

```
1 # vgcreate test /dev/hdb2 /dev/hda3
2 vgcreate -- INFO: using default physical extent size 4 MB
3 vgcreate -- INFO: maximum logical volume size is 255.99 Gigabyte
4 vgcreate -- doing automatic backup of volume group "test"
5 vgcreate -- volume group "test" successfully created and activated
```

**4. Zapoznaj się z manuałem komendy `vgcreate` . Utwórz grupę wolumenów fizycznych o nazwie `test` ze stworzonych w poprzednim kroku wolumenów fizycznych.**

Korzystając z komendy `vgdisplay` możemy sprawdzić jak wygląda charakterystyka utworzonej przez nas grupy wolumenów:

```
1 # vgdisplay -v test
2 --- Volume group ---
3 VG Name                test
4 VG Access               read/write
5 VG Status               available/resizable
6 VG #                   0
7 MAX LV                 256
8 Cur LV                 0
9 Open LV                0
10 MAX LV Size            255.99 GB
11 Max PV                 256
12 Cur PV                 2
13 Act PV                 2
14 VG Size                184 MB
15 PE Size                4 MB
16 Total PE               46
17 Alloc PE / Size        0 / 0
18 Free PE / Size         46 / 184 MB
19
20 --- No logical volumes defined in test ---
21
22
23 --- Physical volumes ---
24 PV Name (#)            /dev/hda3 (2)
25 PV Status               available / allocatable
26 Total PE / Free PE     13 / 13
27
28 PV Name (#)            /dev/hdb2 (1)
29 PV Status               available / allocatable
30 Total PE / Free PE     33 / 33
```

**5. Zapoznaj się z manuałem komendy `vgdisplay` . Sprawdź charakterystykę utworzonej grupy wolumenów. Wyniki działania komendy porównaj z zawartością plików występujących w katalogu `/proc/lvm/VGs/test/` . Sprawdź ile fizycznych extendów występuje w grupie i ile z nich jest wolnych. Jaki jest rozmiar pojedynczego fizycznego extendu.**

Dysponujemy już grupą wolumenów, a więc do naszej dyspozycji są fizyczne extendy, z których możemy utworzyć wolumen logiczny. Służy do tego komenda `lvcreate` . Poniższy przykład pokazuje jej postać dla stworzenia wolumenu logicznego o rozmiarze 50MB o nazwie `lv007`.

```
1 # lvcreate -L 50M -n lv007 test
2 lvcreate -- rounding up size to physical extent boundary "52 MB"
3 lvcreate -- doing automatic backup of "test"
4 lvcreate -- logical volume "/dev/test/lv007" successfully created
```

W wyniku działania komendy `lvcreate` zostanie utworzony wolumen logiczny oraz plik specjalny reprezentujący go jako urządzenie blokowe. W naszym przypadku będzie to `/dev/test/lv007`. Urządzenie to stanowi odpowiednik partycji i jest przez system traktowane w podobny sposób. Występuje również w liście partycji dostępnej w pliku `/proc/partitions`.

**6. Zapoznaj się z manuałem komendy `lvcreate`. W grupie wolumenów `test` utwórz wolumen fizyczny o nazwie `lv007`, o rozmiarze 100MB. Wyświetl zawartość pliku `/proc/partitions`, odszukaj nowoutworzone urządzenie.**

Celem wyświetlenia informacji dotyczącej wolumenu logicznego można posłużyć się komendą `lvdisplay`:

```
1 # lvdisplay /dev/test/lv007
2 --- Logical volume ---
3 LV Name                /dev/test/lv007
4 VG Name                test
5 LV Write Access        read/write
6 LV Status              available
7 LV #                  1
8 # open                 1
9 LV Size                52 MB
10 Current LE            13
11 Allocated LE          13
12 Allocation             next free
13 Read ahead sectors    1024
14 Block device          58:0
```

Możliwe jest również uzyskanie informacji szczegółowych zawierających między innymi powiązanie logicznych extendów z fizycznymi. Wymaga to użycia opcji `-v`:

```
1 # lvdisplay -v /dev/test/lv007
2 ...
3 Block device          58:0
4
5 --- Distribution of logical volume on 1 physical volume ---
6 PV Name              PE on PV    reads    writes
7 /dev/hda7            13          834671   1066177
8
9 --- logical volume i/o statistic ---
10 834671 reads 1066177 writes
11
12 --- Logical extents ---
13 LE    PV              PE    reads    writes
14 00000 /dev/hda3       00000 7413     414
15 00001 /dev/hda3       00001 57        0
16 00002 /dev/hda3       00002 0         0
17 ...
18 00012 /dev/hda3       00012 57        6553
19
```

**7. Zapoznaj się z manuałem komendy `lvdisplay`, wyświetl informacje o utworzonym uprzednio wolumenie logicznym. Wykorzystaj opcję `-v`.**

W tym momencie dysponujemy już wolumenem logicznym, który stanowi podstawę do stworzenia systemu plików. Jądro systemu Linux może obsługiwać różne systemy plików, o ile odpowiednie funkcje ich obsługi zostały do niego dołączone. Poniższy przykład tworzy system plików typu `ext2` na uprzednio stworzonym wolumenie logicznym:

```
1 # mke2fs /dev/test/lv007
2 mke2fs 1.18, 11-Nov-1999 for EXT2 FS 0.5b, 95/08/09
3 Filesystem label=
```

```

4 S type: Linux
5 Block size=1024 (log=0)
6 Fragment size=1024 (log=0)
7 13328 inodes, 53248 blocks
8 2662 blocks (5.00%) reserved for the super user
9 First data block=1
10 7 block groups
11 8192 blocks per group, 8192 fragments per group
12 1904 inodes per group
13 Superblock backups stored on blocks:
14     8193, 24577, 40961
15
16 Writing inode tables: done
17 Writing superblocks and filesystem accounting information: done

```

Dodatkowo stworzony system plików został zamontowany w katalogu `/mnt`. Operacja ta „udostępnia” system plików do użytkowania. Wejście do systemu plików stanowi punkt montowania - czyli katalog. Montowanie systemu plików w katalogu nie niszczy zawartości katalogu lecz jedynie ją przykrywa. Po odmontowaniu systemu plików jest ona ponownie dostępna. Podobnie zawartość systemu plików - po odmontowaniu nie jest ona tracona, a staje się dostępna po zamontowaniu systemu plików.

Komenda `mount` użyta bez opcji i argumentów listuje wszystkie aktualnie zamontowane systemy plików. Do zamontowania potrzebne są dwa argumenty: ścieżka dostępu do urządzenia reprezentującego wolumen logiczny, na którym system plików został założony oraz punkt montowania, czyli ścieżka dostępu do katalogu. Do odmontowania służy komenda `umount`, która wymaga podania jako argumentu nazwy wolumenu logicznego z systemem plików, który chcemy odmontować.

```

1 # mount /dev/test/lv007 /mnt
2 # ls /mnt
3 lost+found

```

**7. Zapoznaj się z manuałem komendy `mke2fs`. Na stworzonym uprzednio wolumenie logicznym załóż system plików `ext2`.**

**8. W katalogu `/tmp` utwórz katalog `mnt`. Zamontuj w nim utworzony system plików.**

**9. Zapoznaj się z manuałem komendy `df`. Wylistuj wszystkie zdefiniowane w systemie systemy plików. Sprawdź ile i-węzłów jest zajętych w zamontowanym właśnie systemie plików.**

Przypominamy, że pełną charakterystykę wykorzystania grupy wolumenów otrzymamy dzięki komendzie `vgdisplay`. Wymaga ona jako argumentu podania nazwy grupy wolumenów, zaś opcja `-v` przełączy ją w tryb gadatliwy.

**10. Sprawdź ile wolnych fizycznych extendów pozostało jeszcze w grupie wolumenów `test`.**

Jeżeli zamierzamy zwiększyć rozmiar systemu plików w pierwszej kolejności musimy zadbać o zwiększeniu rozmiaru wolumenu logicznego na którym system plików został założony. Używamy w tym celu komendy `lvextend` podając po opcji `-L` wielkość jaką chcemy osiągnąć oraz podobnie jak poprzednio, ścieżkę dostępu do urządzenia blokowego obsługującego ten wolumen logiczny:

```

1 # lvextend -L+12M /dev/test/lv007
2 lvextend -- rounding size to physical extent boundary
3 lvextend -- extending logical volume "/dev/test/lv007" to 116 MB
4 lvextend -- doing automatic backup of volume group "test"
5 lvextend -- logical volume "/dev/test/lv007" successfully extended

```

**11. Korzystając z komendy `lvextend` zwiększ rozmiar wolumenu logicznego `lv007` o 60 MB. Podobnie jak poprzednio, przy użyciu komend `df` oraz `vgdisplay` sprawdź poprawność zmian**

**12. Zbadaj zmiany które nastąpiły w pliku `/proc/partitions` i porównaj rozmiar wolumenu logicznego z rozmiarem systemu plików. O ile procent rozmiar systemu plików jest mniejszy od rozmiaru urządzenia na którym występuje.**

Ponieważ zmiana rozmiaru urządzenia blokowego (wolumenu logicznego) nie pociąga za sobą automatycznej zmiany rozmiaru systemu plików musimy tę operację wykonać samodzielnie. Sposób przeprowadzenia tej operacji jest zależny od używanego przez nas systemu plików, możliwe jest nawet , że system plików nie posiada możliwości zmiany rozmiaru.

Dla systemu plików ext2 oraz ext3 służy do tego polecenie `resize2fs` jego wymaganym argumentem jest urządzenie blokowe zawierające system plików. Dodatkowo możemy podać nowy rozmiar systemu plików, domyślnie pobierany jest rozmiar urządzenia czyli system plików jest „naciągany” do rozmiaru urządzenia. Jawne podanie rozmiaru jest przydatne w chwili gdy zamierzamy zmniejszyć rozmiar systemu plików.

Ponieważ operacja zmiany rozmiaru jest ryzykowna wymaga się aby przed jej wykonaniem sprawdzić spójność systemu plików nawet jeśli system plików wydaje się być poprawny. Do sprawdzania systemu plików służy polecenie `fsck` lub dla systemu plików ext2/ext3 `e2fsck`. Operacja sprawdzania systemu plików jest możliwa wyłącznie wówczas gdy system plików **nie jest używany**, tak więc przed przystąpieniem do sprawdzania systemu plików konieczne musimy go odmonotować. Sprowadza się to do wywołania poleceń:

```
1 # umount /dev/test/lv007
2 # e2fsck -f /dev/test/lv007
```

Podana opcja `-f` dla polecenia `e2fsck` przekonuje polecenie do sprawdzenia systemu plików pomimo iż wydaje się być poprawny (wg. stanu zapisanego w systemie plików).

Ostatecznym krokiem zwiększania rozmiaru jest wywołanie polecenia:

```
1 # resize2fs /dev/test/lv007
```

Przypominamy, że tę operację wykonujemy również na odmontowanym systemie plików. Po zwiększeniu systemu plików można rozpocząć jego normalną eksploatację (zamonotować go).

Opisany powyżej sposób zmiany rozmiaru dotyczy systemu plików ext2/ext3. W przypadku innych systemów plików mogą istnieć inne narzędzia zmieniające rozmiar. Istnieją systemy plików które umożliwiają zmianę (zwiększenie) rozmiaru systemu plików bez konieczności odmontowywania (pochodzący z firmy *IBM* system plików JFS).

**13. Przeprowadź operację zwiększenia rozmiaru systemu plików rezydującego na urządzeniu blokowym `/dev/test/lv007` do wielkości tego urządzenia. Sprawdź rozmiar systemu plików po zwiększeniu i porównaj go z wielkością urządzenia zapisaną w pliku `/proc/partitions`.**

W przypadku systemu plików ext2/ext3 możliwe jest również zmniejszenie rozmiaru. Uwaga przy zmniejszaniu rozmiaru należy wykonać najpierw zmniejszenie rozmiaru systemu plików a następnie zmniejszyć rozmiar wolumenu logicznego.

Istnieje również polecenie `e2fsadm`, które umożliwia zmianę rozmiaru systemu plików wykonując krok po kroku opisane powyżej czynności. W istocie wykonywane są dokładnie te same programy które zostały opisane powyżej.

Kolejnym rozwiązaniem, które stanie się konieczne, gdy zabraknie extendów fizycznych w grupie wolumenów polega na dodaniu nowego wolumenu fizycznego do rozpatrywanej grupy wolumenów. Proces dodawania nowego wolumenu fizycznego do grupy wolumenów składa się z dwóch kroków. W pierwszym nowo zainstalowane w systemie medium fizyczne (dysk) przekształcamy do wolumenu fizycznego przy pomocy komendy `pvccreate`. W drugim rozszerzamy istniejącą i aktywną grupę wolumenów o utworzony właśnie wolumen fizyczny przy pomocy komendy `vgextend`. Komenda `vgextend` wymaga podania dwóch argumentów: nazwy grupy wolumenów do której ma zostać dołączony nowy wolumen oraz nazwę urządzenia blokowego będącego nowym wolumenem fizycznym:

```
1 # pvccreate /dev/sda1
2 pvccreate -- physical volume "/dev/sda1" successfully created
3 # vgextend test /dev/sda1
4 vgextend -- INFO: maximum logical volume size is 255.99 Gigabyte
5 vgextend -- doing automatic backup of volume group "test"
6 vgextend -- volume group "test" successfully extended
```

Po dołączeniu nowego wolumenu fizycznego możemy wykorzystać znajdujące się na nim extendy fizyczne do zwiększenia dowolnego wolumenu logicznego w obrębie rozpatrywanej grupy wolumenów.

14. Zapoznaj się z manuałem komendy `vgextend`. Przekształć dostępne w systemie medium fizyczne w wolumen fizyczny.

15. Włącz do grupy wolumenów `test` utworzony właśnie wolumen fizyczny.

16. Zwiększ rozmiar dowolnego wolumenu logicznego zdefiniowanego w grupie wolumenów `test` o 40 bloków (fizycznych extendów).

17. Sprawdź wykorzystanie extendów fizycznych w grupie wolumenów `test`. Jaką komendę należy użyć?

Do tej pory zajmowaliśmy się powiększaniem liczby dostępnych fizycznych extendów w grupie wolumenów, rozmiaru grupy wolumenów oraz pojemności systemu plików. Zdarza się jednak, że zachodzi konieczność odwrotna, tzn. z grupy wolumenów musimy z pewnych przyczyn usunąć wolumen fizyczny. Dzięki filozofii zastosowanej w LVM operacja ta jest również prosta.

Pierwszą rzeczą o której należy pamiętać jest przeniesienie danych z extendów fizycznych znajdujących się na usuwanym wolumenie. Extendy te przenoszone są na inne wolumeny fizyczne znajdujące się w tej samej grupie wolumenów. Warunkiem powodzenia jest istnienie odpowiedniej liczby wolnych fizycznych extendów. Przeniesienie danych z extendów fizycznych dokonuje się z wykorzystaniem komendy `pvmove`.

Załóżmy, że chcemy z grupy wolumenów fizycznych usunąć urządzenie `/dev/hda1`, a na jego miejsce zainstalować przykładowo `/dev/sdb3`. W pierwszym kroku urządzenie `/dev/sdb3` dodajemy do grupy wolumenów, w której znajduje się urządzenie `/dev/hda1`. Spowoduje to, że w grupie wolumenów pojawiają się dodatkowe wolne extendy fizyczne, na które może zostać przesunięta informacja z wolumenu który zamierzamy usunąć. W drugim kroku przesuamy informacje z usuwanego wolumenu fizycznego. Jedynym koniecznym argumentem komendy `pvmove` jest w tym przypadku nazwa wolumenu fizycznego który zamierzamy zwolnić:

```
1 # pvmove /dev/hda1
2 pvmove -- moving physical extents in active volume group "test"
3 pvmove -- WARNING: moving of active logical volumes may cause data loss!
4 pvmove -- do you want to continue? [y/n] y
5 pvmove -- doing automatic backup of volume group "test"
6 pvmove -- 51 extents of physical volume "/dev/hda1" successfully moved
```

Po przeniesieniu informacji, wolumen fizyczny może zostać usunięty z grupy. Służy do tego komenda `vgreduce`. Wymaga ona podania dwóch argumentów: nazwy grupy wolumenów z której urządzenie ma zostać usunięte oraz nazwy urządzenia:

```
1 # vgreduce test /dev/hda1
2 vgreduce -- doing automatic backup of volume group "test"
3 vgreduce -- volume group "test1" successfully reduced by physical volume:
4 vgreduce -- /dev/hda1
```

Operacje przenoszenia danych pomiędzy wolumenami fizycznymi wykonywana może być na działającym systemie. Urządzenie blokowe jakim jest wolumen logiczny jest aktualne w każdej chwili, również podczas operacji przenoszenia danych.

18. Zapoznaj się z manuałem komend `pvmove` oraz `vgreduce`. Z grupy wolumenów `test` usuń dowolny wolumen fizyczny, przenieś jego zawartość na inny wolumen fizyczny. Czy możliwe jest usunięcie komendą `vgreduce` z grupy wolumenów ostatniego wolumenu. Co dzieje się wówczas z grupą wolumenów?

19. Zbadaj jak wygląda przyporządkowanie extendów logicznych do fizycznych po operacji usunięcia fizycznego wolumenu.

Niektóre systemy komputerowe wymagają stosowania przenośnych dysków, na których mogą być zapisywane np. dane analizowane w różnych systemach. Z różnych powodów przesyłanie danych z wykorzystaniem sieci lub kopiowanie ich przy pomocy nośnika może nie być wykonalne. Wówczas najprostszym sposobem przesłania danych jest fizyczne przeniesienie dysków między systemami. W przypadku podsystemu LVM operacja przenoszenia dyków jest znacznie bardziej skomplikowana niż w przypadku dysków podzielonych na



partycje. Jest to spowodowane skomplikowaną strukturą LVM. Na taką okoliczność LVM wyposażony został w odpowiednie narzędzia. Operacja przenoszenia dysków współpracujących z podsystemem LVM wymaga przeniesienia wszystkich dysków (wolumenów fizycznych) wchodzących w skład przenoszonej grupy wolumenów. Sama operacja przeniesienia grupy wolumenów fizycznych jest bardzo prosta i logiczna. Wymaga ona wyeksportowania grupy wolumenów z jednego systemu i dokonania operacji jej importu w drugim systemie.

Struktura danych LVM opisująca konfigurację grupy wolumenów zapisana jest w pliku `/etc/lvmconf` oraz na każdym wolumenie fizycznym grupy. Obszar na wolumenie fizycznym zajmowany przez strukturę opisującą nazywany jest VGDA (Volume Group Description Area). Operacja eksportu dokonuje zmian w pliku `/etc/lvmconf` systemu, z którego grupa wolumenów jest exportowana – komenda `vgexport` po prostu usuwa odpowiedni rekord z pliku `/etc/lvmconf`. Podczas wykonywania operacji importu, komenda `vgimport` odczytuje informacje o importowanej grupie z jej obszarów VGDA i na tej podstawie dokonuje wpisu do pliku `/etc/lvmconf`. Operacja przeniesienia grupy wolumenów między systemami komputerowymi może zostać wykonana np. w następujący sposób:

Na komputerze #1 (zabieramy):

```
1 # vgchange -a n test
2 # vgexport test
```

Na komputerze #2 (dokładamy):

```
1 # vgimport vg01 /dev/sdd1 /dev/sde1
2 # vgchange -a y vg01
```

**Uwaga !** Nazwa grupy wolumenów, do której importowane są wolumeny fizyczne nie musi być taka sama jak ta, z której wolumeny pochodzą. Podobnie nazwy wolumenów fizycznych po podłączeniu do nowego systemu komputerowego mogą być inne, co jest zależne od jego konfiguracji..

Operacja eksportowania grupy wolumenów wymaga jej wyłączenia co oznacza, że nie możemy używać wolumenów logicznych wchodzących w skład tej grupy. Podobnie po zaimportowaniu grupy wolumenów, przed rozpoczęciem jej eksploatacji konieczne jest jej włączenie.

Aktywację i deaktywację grupy wolumenów przeprowadzić można wykorzystując komendę `vgchange`. Przykładowo, aktywację grupy wolumenów `test` może przeprowadzić użytkownik root pisząc: `vgchange -a y test`, zaś deaktywację wydając polecenie: `vgchange -a n test`.