[5.4 Volume Management](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4+Volume+Management)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified on [08 Apr 2020](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=521112282&selectedPageVersions=2&selectedPageVersions=3)

Table of contents:

[5.4 Volume Management](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4+Volume+Management?src=contextnavpagetreemode)

* [5.4.1 FS add-Extend on standalone physical server (XFS file system)](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=667006241&src=contextnavpagetreemode)
* [5.4.2 Linux LVM](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.2+Linux+LVM?src=contextnavpagetreemode)
  + [5.4.2.1 Create LVM VG+LVs](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=520963000&src=contextnavpagetreemode)
  + [5.4.2.2 Extending LVM Storage with AutoSAN Tools](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.2.2+Extending+LVM+Storage+with+AutoSAN+Tools?src=contextnavpagetreemode)
  + [5.4.2.3 Grow Existing VG/LVs](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=520964200&src=contextnavpagetreemode)
  + [5.4.2.4 LVM Problems and Solutions](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.2.4+LVM+Problems+and+Solutions?src=contextnavpagetreemode)
  + [5.4.2.5 XEN VM Grow LV](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.2.5+XEN+VM+Grow+LV?src=contextnavpagetreemode)
* [5.4.3 Return SAN VxVM + LVM](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=521112291&src=contextnavpagetreemode)
* [5.4.4 SAN Storage Request](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.4+SAN+Storage+Request?src=contextnavpagetreemode)
* [5.4.5 Veritas VxVM](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5+Veritas+VxVM?src=contextnavpagetreemode)
  + [5.4.5.1 Add Additional VxVM Resources](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.1+Add+Additional+VxVM+Resources?src=contextnavpagetreemode)
  + [5.4.5.2 Change type from gen to fsgen](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.2+Change+type+from+gen+to+fsgen?src=contextnavpagetreemode)
  + [5.4.5.3 Configure VxVM](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.3+Configure+VxVM?src=contextnavpagetreemode)
  + [5.4.5.4 Defragment VxFS](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.4+Defragment+VxFS?src=contextnavpagetreemode)
  + [5.4.5.5 Extend secfs](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.5+Extend+secfs?src=contextnavpagetreemode)
  + [5.4.5.6 Grow Existing VxVM DGs/LVs](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=521125891&src=contextnavpagetreemode)
  + [5.4.5.7 LINUX Kernel Bug Remove Disks](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.7+LINUX+Kernel+Bug+Remove+Disks?src=contextnavpagetreemode)
  + [5.4.5.8 Update UDID](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.8+Update+UDID?src=contextnavpagetreemode)
  + [5.4.5.9 Veritas Filesystem errors](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.9+Veritas+Filesystem+errors?src=contextnavpagetreemode)
  + [5.4.5.10 VxVM cheat sheet](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.10+VxVM+cheat+sheet?src=contextnavpagetreemode)
  + [5.4.5.11 VxVM HowTos](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.11+VxVM+HowTos?src=contextnavpagetreemode)
  + [5.4.5.12 VxVM Reduce](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.12+VxVM+Reduce?src=contextnavpagetreemode)
  + [5.4.5.13 VxVM Setting Read policy](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.13+VxVM+Setting+Read+policy?src=contextnavpagetreemode)
  + [5.4.5.14 VxVM volume manually](https://atc.bmwgroup.net/confluence/display/IAAS18/5.4.5.14+VxVM+volume+manually?src=contextnavpagetreemode)

Last edited by Liebl Markus, (Markus.Liebl@[partner.bmw.de](http://partner.bmw.de/)) , based on work by Kirchberger Marco, (Marco.Kirchberger@[partner.bmw.de](http://partner.bmw.de/)) and [system](https://bsswiki.muc/tiki-user_information.php?userId=-1) . Page last modified on Wednesday 16 of October, 2019 11:30:11 CEST. (Version 3)

* Created by [Sanjiv Soraganvi (ext.)](https://atc.bmwgroup.net/confluence/display/~qxw9324), last modified by [VijayKumarReddy Ayyaluri (ext.)](https://atc.bmwgroup.net/confluence/display/~qxy7563" \o ") on [10 Feb 2023](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=667006241&selectedPageVersions=12&selectedPageVersions=13)

#### **Disk addition threshold limit - Number of Luns per Xenfarm and Per Xen VMs** -Farm server:  Total number of LUNs - 1000  (#san\_info -ls  |grep -i "Disks seen"|sort -u) -Virtual Server : Total number of Luns assigned (#vmc info <vmname> or san\_shortinfo |grep -i <vmname>, other steps)                               Minumum Limit = 25 Luns                               Maximum Limit = 30 Luns                 If the Luns on Farm  is equal or beyond limit then no more Luns addition to farm is possible. If its needed then disk consolidation is required.                If the Luns on VMs is min limit  you may add more Luns till max limit but its a trigger to initiate the disk consolidation first and then proceed addition/extension.

**This document provides the steps to add new file system and Extend  file system on standalone physical server running with xfs file system.**  
Make sure the system is stand alone and running XFS file system  
#hastatus -summ => Server should not be a part of cluster  
#df -hPT => should have XFS file system

**1) Order the number of disks with required size using**[**autosan.pl**](http://autosan.pl/)**tool if its failed use STORM tool**  
in the below example ordering two 14GB disks(LUNs) and make a note of the LDEV numbers which you see in the command output .

[INSTSERV][qxw9324][lpinstiaas01] ~ $ [autosan.pl](http://autosan.pl/) order-disks-servers -o INC000025395072 -d 14:2 --server ltiaastest01 --server ltiaastest02  
Script started, file is /global/instserv/logs/full\_outputs/2020/02/05/[2020-02-05\_14-01-30\_autosan.pl](http://2020-02-05_14-01-30_autosan.pl/).qxw9324.log

AutoSAN version 20190320  
User qxw9324 called command: "/global/instserv/bin/[autosan.pl](http://autosan.pl/) order-disks-servers -o INC000025395072 -d 14:2 --server ltiaastest01 --server ltiaastest02"

Info[1]: 2020-02-05 14:01:31 Collecting storage data from server(s)  
Info[1]: 2020-02-05 14:01:36 Collecting SAN name, contract, APP-ID...  
Info[1]: 2020-02-05 14:02:31 Server names: ltiaastest01 ltiaastest02, SAN name is EDC305 (19.0 2.2/13.4 1.1), contract is SAN\_GS\_3, APP-ID is APP-108617  
Info[1]: 2020-02-05 14:02:31 Server: ltiaastest01, target ports are 7L/8L, WWNs are 10:00:00:10:9b:57:09:73/10:00:00:10:9b:57:0a:60  
Info[1]: 2020-02-05 14:02:31 Server: ltiaastest02, target ports are 7L/8L, WWNs are 10:00:00:10:9b:52:cc:5f/10:00:00:10:9b:52:cc:10  
Info[1]: 2020-02-05 14:02:31 Ordering storage through SPS, this may take a while...  
Info[1]: 2020-02-05 14:02:35 Storm ticket ID: 307530, OrderPOSid(s): 307531  
Info[1]: 2020-02-05 14:02:35 OrderPOSid: 307531: Status: Submitted - SPS automated ticket submitted, is\_automated\_ticket: Y  
Info[1]: 2020-02-05 14:04:06 OrderPOSid: 307531: Status: Work in Progress - work in progress, is\_automated\_ticket: Y  
Info[1]: 2020-02-05 14:16:44 OrderPOSid: 307531: Status: Closed - storage provisioned, is\_automated\_ticket: Y  
**Info[1]: 2020-02-05 14:17:03 Array: 10774, LUN: 00:0A:B5**  
**Info[1]: 2020-02-05 14:17:03 Array: 10774, LUN: 00:0A:B7**  
**Info[1]: 2020-02-05 14:17:03 Array: 10754, LUN: 00:0A:B7**  
**Info[1]: 2020-02-05 14:17:03 Array: 10754, LUN: 00:0A:B5**  
Info[1]: 2020-02-05 14:17:03 Rescanning SAN on server ltiaastest01 and waiting 20 seconds to settle multipath device nodes...  
Info[1]: 2020-02-05 14:17:33 SUCCESS: All new luns are available on server ltiaastest01.  
Info[1]: 2020-02-05 14:17:33 Rescanning SAN on server ltiaastest02 and waiting 20 seconds to settle multipath device nodes...  
Info[1]: 2020-02-05 14:18:02 SUCCESS: All new luns are available on server ltiaastest02.  
Script done, file is /global/instserv/logs/full\_outputs/2020/02/05/[2020-02-05\_14-01-30\_autosan.pl](http://2020-02-05_14-01-30_autosan.pl/).qxw9324.log

**2) Check the noted LDEV disks are deteched or not on the server**  
if not you may scan using #san\_rescan -o  
ltiaastest01:~ # san\_shortinfo |egrep -i "0A:B5|0A:B7"  
dm-29 | sdaz 0A:B5 7L 10754 14405 | sdat 0A:B5 8L 10754 14405 | - -  
dm-28 | sdba 0A:B7 7L 10754 14405 | sdau 0A:B7 8L 10754 14405 | - -  
dm-27 | sdax 0A:B5 7L 10774 14405 | sdav 0A:B5 8L 10774 14405 | - -  
dm-26 | sday 0A:B7 7L 10774 14405 | sdaw 0A:B7 8L 10774 14405 | - -

**3) Create a new volume group using the command dmdiskctl (using the LDEV 0ab5 as an example)**  
0ab5 is an LDEV number , 10754 and 10774 are the storage arrays. -a is to add an entry in mdadm.conf file

ltiaastest01:~ # dmdiskctl newvg test\_vg 10754,0ab5 10774,0ab5 -a  
14:40:30 Creating new MD device /dev/md/test\_vg\_1 with name SAN:test\_vg\_1 OK  
14:40:31 Setting writemostly flag for dm-29(10754) in /dev/md/test\_vg\_1 OK  
14:40:31 Adding /dev/md/test\_vg\_1 to /etc/mdadm.conf OK  
14:40:31 Creating PV data on /dev/md/test\_vg\_1 OK  
14:40:31 Creating Volume Group test\_vg OK  
ltiaastest01:~ #

ltiaastest01:~ # vgs test\_vg  
VG #PV #LV #SN Attr VSize VFree  
test\_vg 1 0 0 wz--n- 14.06g 14.06g  
ltiaastest01:~ #

**4)dmdiskctl will add an entry in the mdadm.conf file while creating the volume group. Ensure the required entries are added to mdadm.conf file**

ltiaastest01:~ # cat /etc/mdadm.conf  
DEVICE /dev/disk/by-id/dm-uuid-mpath-\* /dev/dm-\*  
ARRAY /dev/md/test\_vg\_1 metadata=1.1 UUID=7891f1f3:80117ea4:9f4da6b6:2cbfe96b name=SAN:test\_vg\_1  
ltiaastest01:~ #

**5)Create a logical volume using dmdiskctl**  
  
#dmdiskctl newlv test\_vg test\_lv max  
13:27:19 Creating LV test\_lv in VG test\_vg with size 518472M OK

ltiaastest01:~ # lvs /dev/test\_vg/test\_lv  
LV VG Attr LSize Pool Origin Data% Move Log Copy% Convert  
test\_lv test\_vg -wi-a---- 14.06g  
ltiaastest01:~ #

**6)Create a xfs file system on the created lv**

ltiaastest01:~ # mkfs.xfs /dev/test\_vg/test\_lv  
meta-data=/dev/test\_vg/test\_lv isize=256 agcount=4, agsize=921344 blks  
= sectsz=512 attr=2, projid32bit=0  
data = bsize=4096 blocks=3685376, imaxpct=25  
= sunit=0 swidth=0 blks  
naming =version 2 bsize=4096 ascii-ci=0  
log =internal log bsize=4096 blocks=2560, version=2  
= sectsz=512 sunit=0 blks, lazy-count=1  
realtime =none extsz=4096 blocks=0, rtextents=0  
ltiaastest01:~ #

**7) Create a mount point and mount the LV and add an entry in fstab file**

ltiaastest01:~ # mkdir /tmp/test\_mount /tmp/test\_mount

Add an entry in fstab example:  : /dev/mapper/test\_vg-test\_lv  defaults,nofail,x-systemd.device-timeout=20   0 0  
ltiaastest01:~ # mount -a   ==> Verify the file system is mounted   
ltiaastest01:~ # df -hP /tmp/test\_mount  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/test\_vg-test\_lv 15G 33M 15G 1% /tmp/test\_mount  
ltiaastest01:~ #  
======================================================================  
**Below steps to extend the existing xfs file system using dmdiskctl**  
**1)Order the number of disks with required size using**[**autosan.pl**](http://autosan.pl/)**tool if its failed use STORM tool**  
in the below example ordering two 14GB disks(LUNs) and make a note of the LDEV numbers which you see in the command output .

**2)Check the noted LDEV disks are deteched or not on the server**  
if not you may scan using #san\_rescan -o  
ltiaastest01:~ # san\_shortinfo |egrep -i "0A:B5|0A:B7"  
dm-29 | sdaz 0A:B5 7L 10754 14405 | sdat 0A:B5 8L 10754 14405 | - -  
dm-28 | sdba 0A:B7 7L 10754 14405 | sdau 0A:B7 8L 10754 14405 | - -  
dm-27 | sdax 0A:B5 7L 10774 14405 | sdav 0A:B5 8L 10774 14405 | - -  
dm-26 | sday 0A:B7 7L 10774 14405 | sdaw 0A:B7 8L 10774 14405 | - -

**3) Add a new disk to existing volume group**  
0ab7 is an LDEV number , 10754 and 10774 are the storage arrays. -a is to add an entry in mdadm.conf file  
  
ltiaastest01:~ # dmdiskctl adddisks test\_vg 10754,0ab7 10774,0ab7 -a  
14:51:16 Creating new MD device /dev/md/test\_vg\_2 with name SAN:test\_vg\_2 and devs /dev/dm-28 /dev/dm-26 OK  
14:51:16 Setting writemostly flag for dm-28(10754) in /dev/md/test\_vg\_2 OK  
14:51:16 Adding /dev/md/test\_vg\_2 to /etc/mdadm.conf OK  
14:51:16 Creating PV data on /dev/md/test\_vg\_2 OK  
14:51:17 Extending Volume Group test\_vg with new disks OK

**4) dmdiskctl will add an entry in the mdadm.conf file while creating the volume group. Ensure the required entries are added to mdadm.conf file**

ltiaastest01:~ # cat /etc/mdadm.conf  
DEVICE /dev/disk/by-id/dm-uuid-mpath-\* /dev/dm-\*  
ARRAY /dev/md/test\_vg\_1 metadata=1.1 UUID=7891f1f3:80117ea4:9f4da6b6:2cbfe96b name=SAN:test\_vg\_1  
ARRAY /dev/md/test\_vg\_2 metadata=1.1 UUID=540dcf53:a3c0b148:8034724a:09641669 name=SAN:test\_vg\_2  
ltiaastest01:~ #

**5) Verify new disk added to VG and side is increased**

ltiaastest01:~ # vgs test\_vg  
VG #PV #LV #SN Attr VSize VFree  
test\_vg 2 1 0 wz--n- 28.12g 14.06g  
ltiaastest01:~ #

**6) Increase the existing logical volume size using dmdiskctl and verify the size**

ltiaastest01:~ # dmdiskctl growlv test\_vg test\_lv max  
14:54:00 Growing LV test\_lv in VG test\_vg by 14.06G OK

ltiaastest01:~ # lvs /dev/test\_vg/test\_lv  
LV VG Attr LSize Pool Origin Data% Move Log Copy% Convert  
test\_lv test\_vg -wi-ao--- 28.12g  
ltiaastest01:~ #

**7) Grow the xfs file system on newly increased LV**

ltiaastest01:~ # xfs\_growfs /tmp/test\_mount  
meta-data=/dev/mapper/test\_vg-test\_lv isize=256 agcount=4, agsize=921344 blks  
= sectsz=512 attr=2  
data = bsize=4096 blocks=3685376, imaxpct=25  
= sunit=0 swidth=0 blks  
naming =version 2 bsize=4096 ascii-ci=0  
log =internal bsize=4096 blocks=2560, version=2  
= sectsz=512 sunit=0 blks, lazy-count=1  
realtime =none extsz=4096 blocks=0, rtextents=0  
data blocks changed from 3685376 to 7370752

ltiaastest01:~ # df -hP /tmp/test\_mount  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/test\_vg-test\_lv 29G 33M 29G 1% /tmp/test\_mount  
ltiaastest01:~ #

This document provides the steps to add new file system and Extend  file system on standalone physical server running with xfs file system.  
Make sure the system is stand alone and running XFS file system  
#hastatus -summ => Server should not be a part of cluster  
#df -hPT => should have XFS file system

**1) Order the number of disks with required size using**[**autosan.pl**](http://autosan.pl/)**tool if its failed use STORM tool**  
in the below example ordering two 14GB disks(LUNs) and make a note of the LDEV numbers which you see in the command output .

[INSTSERV][qxw9324][lpinstiaas01] ~ $ [autosan.pl](http://autosan.pl/) order-disks-servers -o INC000025395072 -d 14:2 --server ltiaastest01 --server ltiaastest02  
Script started, file is /global/instserv/logs/full\_outputs/2020/02/05/[2020-02-05\_14-01-30\_autosan.pl](http://2020-02-05_14-01-30_autosan.pl/).qxw9324.log

AutoSAN version 20190320  
User qxw9324 called command: "/global/instserv/bin/[autosan.pl](http://autosan.pl/) order-disks-servers -o INC000025395072 -d 14:2 --server ltiaastest01 --server ltiaastest02"

Info[1]: 2020-02-05 14:01:31 Collecting storage data from server(s)  
Info[1]: 2020-02-05 14:01:36 Collecting SAN name, contract, APP-ID...  
Info[1]: 2020-02-05 14:02:31 Server names: ltiaastest01 ltiaastest02, SAN name is EDC305 (19.0 2.2/13.4 1.1), contract is SAN\_GS\_3, APP-ID is APP-108617  
Info[1]: 2020-02-05 14:02:31 Server: ltiaastest01, target ports are 7L/8L, WWNs are 10:00:00:10:9b:57:09:73/10:00:00:10:9b:57:0a:60  
Info[1]: 2020-02-05 14:02:31 Server: ltiaastest02, target ports are 7L/8L, WWNs are 10:00:00:10:9b:52:cc:5f/10:00:00:10:9b:52:cc:10  
Info[1]: 2020-02-05 14:02:31 Ordering storage through SPS, this may take a while...  
Info[1]: 2020-02-05 14:02:35 Storm ticket ID: 307530, OrderPOSid(s): 307531  
Info[1]: 2020-02-05 14:02:35 OrderPOSid: 307531: Status: Submitted - SPS automated ticket submitted, is\_automated\_ticket: Y  
Info[1]: 2020-02-05 14:04:06 OrderPOSid: 307531: Status: Work in Progress - work in progress, is\_automated\_ticket: Y  
Info[1]: 2020-02-05 14:16:44 OrderPOSid: 307531: Status: Closed - storage provisioned, is\_automated\_ticket: Y  
**Info[1]: 2020-02-05 14:17:03 Array: 10774, LUN: 00:0A:B5**  
**Info[1]: 2020-02-05 14:17:03 Array: 10774, LUN: 00:0A:B7**  
**Info[1]: 2020-02-05 14:17:03 Array: 10754, LUN: 00:0A:B7**  
**Info[1]: 2020-02-05 14:17:03 Array: 10754, LUN: 00:0A:B5**  
Info[1]: 2020-02-05 14:17:03 Rescanning SAN on server ltiaastest01 and waiting 20 seconds to settle multipath device nodes...  
Info[1]: 2020-02-05 14:17:33 SUCCESS: All new luns are available on server ltiaastest01.  
Info[1]: 2020-02-05 14:17:33 Rescanning SAN on server ltiaastest02 and waiting 20 seconds to settle multipath device nodes...  
Info[1]: 2020-02-05 14:18:02 SUCCESS: All new luns are available on server ltiaastest02.  
Script done, file is /global/instserv/logs/full\_outputs/2020/02/05/[2020-02-05\_14-01-30\_autosan.pl](http://2020-02-05_14-01-30_autosan.pl/).qxw9324.log

**2) Check the noted LDEV disks are deteched or not on the server**  
if not you may scan using #san\_rescan -o  
ltiaastest01:~ # san\_shortinfo |egrep -i "0A:B5|0A:B7"  
dm-29 | sdaz 0A:B5 7L 10754 14405 | sdat 0A:B5 8L 10754 14405 | - -  
dm-28 | sdba 0A:B7 7L 10754 14405 | sdau 0A:B7 8L 10754 14405 | - -  
dm-27 | sdax 0A:B5 7L 10774 14405 | sdav 0A:B5 8L 10774 14405 | - -  
dm-26 | sday 0A:B7 7L 10774 14405 | sdaw 0A:B7 8L 10774 14405 | - -

**3) Create a new volume group using the command dmdiskctl (using the LDEV 0ab5 as an example)**  
0ab5 is an LDEV number , 10754 and 10774 are the storage arrays. -a is to add an entry in mdadm.conf file

ltiaastest01:~ # dmdiskctl newvg test\_vg 10754,0ab5 10774,0ab5 -a  
14:40:30 Creating new MD device /dev/md/test\_vg\_1 with name SAN:test\_vg\_1 OK  
14:40:31 Setting writemostly flag for dm-29(10754) in /dev/md/test\_vg\_1 OK  
14:40:31 Adding /dev/md/test\_vg\_1 to /etc/mdadm.conf OK  
14:40:31 Creating PV data on /dev/md/test\_vg\_1 OK  
14:40:31 Creating Volume Group test\_vg OK  
ltiaastest01:~ #

ltiaastest01:~ # vgs test\_vg  
VG #PV #LV #SN Attr VSize VFree  
test\_vg 1 0 0 wz--n- 14.06g 14.06g  
ltiaastest01:~ #

**4)dmdiskctl will add an entry in the mdadm.conf file while creating the volume group. Ensure the required entries are added to mdadm.conf file**

ltiaastest01:~ # cat /etc/mdadm.conf  
DEVICE /dev/disk/by-id/dm-uuid-mpath-\* /dev/dm-\*  
ARRAY /dev/md/test\_vg\_1 metadata=1.1 UUID=7891f1f3:80117ea4:9f4da6b6:2cbfe96b name=SAN:test\_vg\_1  
ltiaastest01:~ #

**5)Create a logical volume using dmdiskctl**  
  
#dmdiskctl newlv test\_vg test\_lv max  
13:27:19 Creating LV test\_lv in VG test\_vg with size 518472M OK

ltiaastest01:~ # lvs /dev/test\_vg/test\_lv  
LV VG Attr LSize Pool Origin Data% Move Log Copy% Convert  
test\_lv test\_vg -wi-a---- 14.06g  
ltiaastest01:~ #

**6)Create a xfs file system on the created lv**

ltiaastest01:~ # mkfs.xfs /dev/test\_vg/test\_lv  
meta-data=/dev/test\_vg/test\_lv isize=256 agcount=4, agsize=921344 blks  
= sectsz=512 attr=2, projid32bit=0  
data = bsize=4096 blocks=3685376, imaxpct=25  
= sunit=0 swidth=0 blks  
naming =version 2 bsize=4096 ascii-ci=0  
log =internal log bsize=4096 blocks=2560, version=2  
= sectsz=512 sunit=0 blks, lazy-count=1  
realtime =none extsz=4096 blocks=0, rtextents=0  
ltiaastest01:~ #

**7) Create a mount point and mount the LV and add an entry in fstab file**

ltiaastest01:~ # mkdir /tmp/test\_mount /tmp/test\_mount

Add an entry in fstab example:  : /dev/mapper/test\_vg-test\_lv  defaults,nofail,x-systemd.device-timeout=20   0 0  
ltiaastest01:~ # mount -a   ==> Verify the file system is mounted   
ltiaastest01:~ # df -hP /tmp/test\_mount  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/test\_vg-test\_lv 15G 33M 15G 1% /tmp/test\_mount  
ltiaastest01:~ #  
======================================================================  
**Below steps to extend the existing xfs file system using dmdiskctl**  
**1)Order the number of disks with required size using**[**autosan.pl**](http://autosan.pl/)**tool if its failed use STORM tool**  
in the below example ordering two 14GB disks(LUNs) and make a note of the LDEV numbers which you see in the command output .

**2)Check the noted LDEV disks are deteched or not on the server**  
if not you may scan using #san\_rescan -o  
ltiaastest01:~ # san\_shortinfo |egrep -i "0A:B5|0A:B7"  
dm-29 | sdaz 0A:B5 7L 10754 14405 | sdat 0A:B5 8L 10754 14405 | - -  
dm-28 | sdba 0A:B7 7L 10754 14405 | sdau 0A:B7 8L 10754 14405 | - -  
dm-27 | sdax 0A:B5 7L 10774 14405 | sdav 0A:B5 8L 10774 14405 | - -  
dm-26 | sday 0A:B7 7L 10774 14405 | sdaw 0A:B7 8L 10774 14405 | - -

**3) Add a new disk to existing volume group**  
0ab7 is an LDEV number , 10754 and 10774 are the storage arrays. -a is to add an entry in mdadm.conf file  
  
ltiaastest01:~ # dmdiskctl adddisks test\_vg 10754,0ab7 10774,0ab7 -a  
14:51:16 Creating new MD device /dev/md/test\_vg\_2 with name SAN:test\_vg\_2 and devs /dev/dm-28 /dev/dm-26 OK  
14:51:16 Setting writemostly flag for dm-28(10754) in /dev/md/test\_vg\_2 OK  
14:51:16 Adding /dev/md/test\_vg\_2 to /etc/mdadm.conf OK  
14:51:16 Creating PV data on /dev/md/test\_vg\_2 OK  
14:51:17 Extending Volume Group test\_vg with new disks OK

**4) dmdiskctl will add an entry in the mdadm.conf file while creating the volume group. Ensure the required entries are added to mdadm.conf file**

ltiaastest01:~ # cat /etc/mdadm.conf  
DEVICE /dev/disk/by-id/dm-uuid-mpath-\* /dev/dm-\*  
ARRAY /dev/md/test\_vg\_1 metadata=1.1 UUID=7891f1f3:80117ea4:9f4da6b6:2cbfe96b name=SAN:test\_vg\_1  
ARRAY /dev/md/test\_vg\_2 metadata=1.1 UUID=540dcf53:a3c0b148:8034724a:09641669 name=SAN:test\_vg\_2  
ltiaastest01:~ #

**5) Verify new disk added to VG and side is increased**

ltiaastest01:~ # vgs test\_vg  
VG #PV #LV #SN Attr VSize VFree  
test\_vg 2 1 0 wz--n- 28.12g 14.06g  
ltiaastest01:~ #

**6) Increase the existing logical volume size using dmdiskctl and verify the size**

ltiaastest01:~ # dmdiskctl growlv test\_vg test\_lv max  
14:54:00 Growing LV test\_lv in VG test\_vg by 14.06G OK

ltiaastest01:~ # lvs /dev/test\_vg/test\_lv  
LV VG Attr LSize Pool Origin Data% Move Log Copy% Convert  
test\_lv test\_vg -wi-ao--- 28.12g  
ltiaastest01:~ #

**7) Grow the xfs file system on newly increased LV**

ltiaastest01:~ # xfs\_growfs /tmp/test\_mount  
meta-data=/dev/mapper/test\_vg-test\_lv isize=256 agcount=4, agsize=921344 blks  
= sectsz=512 attr=2  
data = bsize=4096 blocks=3685376, imaxpct=25  
= sunit=0 swidth=0 blks  
naming =version 2 bsize=4096 ascii-ci=0  
log =internal bsize=4096 blocks=2560, version=2  
= sectsz=512 sunit=0 blks, lazy-count=1  
realtime =none extsz=4096 blocks=0, rtextents=0  
data blocks changed from 3685376 to 7370752

ltiaastest01:~ # df -hP /tmp/test\_mount  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/test\_vg-test\_lv 29G 33M 29G 1% /tmp/test\_mount  
ltiaastest01:~ #

====================================================

### To Migrate data from OLD SAN to NEW SAN

Prechecks :

→ Make sure you have the storage from NEW SAN

Use:

→ we can use dmdiskctl utility to perform the Migration , which will  Adds the given ldevs to the volume group, first creating md devices if necessary, then moves all the data off the original disks, removes them from the vg, and wipes them.

|  |
| --- |
| dmdiskctl migratevg <vgname> <serial,ldev[,ldev,...]> [serial,ldev[,ldev,...]]    Example : dmdiskctl migratevg dgkwom 60201,b0a9 60255,b0a9 |

use nohup if the filesystem size is large

      Ex : nohup dmsiskcrl migratevg dgkwom 60201,b0a9 60255,b0a9 &

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified by [Sanjiv Soraganvi (ext.)](https://atc.bmwgroup.net/confluence/display/~qxw9324) on [22 Dec 2021](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=520963000&selectedPageVersions=1&selectedPageVersions=2)

#### **Disk addition threshold limit - Number of Luns per Xenfarm and Per Xen VMs** -Farm server:  Total number of LUNs - 1000  (#san\_info -ls  |grep -i "Disks seen"|sort -u) -Virtual Server : Total number of Luns assigned (#vmc info <vmname> or san\_shortinfo |grep -i <vmname>, other steps)                               Minumum Limit = 25 Luns                               Maximum Limit = 30 Luns                 If the Luns on Farm  is equal or beyond limit then no more Luns addition to farm is possible. If its needed then disk consolidation is required.                If the Luns on VMs is min limit  you may add more Luns till max limit but its a trigger to initiate the disk consolidation first and then proceed addition/extension.

# Adding disks

## VMC

Add any application data disks if requested the "vm adddisk" MUST be run on the node where the VM is currently running! To add ADDITIONAL disks for the application data (when requestedl) AFTER the new VM is running:  
vm adddisk -d "<DISK3 DISK4>" <VMNAME>  
vm adddisk -d "<DISK5 DISK6>" <VMNAME>  
  
e.g. for a single LVM Volume Group in the VM with two 56GB mirrored pairs on the XEN hosts:  
ltxentest01 # vm adddisk -d "ST18655-59-1bd8 ST18686-59-226c" ltjri99  
ltxentest01 # vm adddisk -d "ST18686-59-2268 ST18655-59-1bd4" ltjri99  
  
ltxentest01 # vm storage | egrep "ST18655-59-1bd8|ST18686-59-226c"  
ST18655-59-1bd8, 18655, 56G, 360060e800548df00000048df00001bd8, /dev/dm-10, md/ltjri99\_1, SAN:ltjri99\_1  
ST18686-59-226c, 18686, 56G, 360060e800548fe00000048fe0000226c, /dev/dm-38, md/ltjri99\_1, SAN:ltjri99\_1  
  
Now get the device name, which is in use for this mirroed pair in the VM:  
ltxentest01 # vm info ltjri99 | egrep ltjri99\_1\|ltjri99\_2  
Block device xvdb : 'dmmd:md;/dev/md/ltjri99\_1(/lfs/xen/config/mdadm.conf)' (SAN:ltjri99\_1)  
Block device xvdc : 'dmmd:md;/dev/md/ltjri99\_2(/lfs/xen/config/mdadm.conf)' (SAN:ltjri99\_2)  
  
So on the VM ltjri99 the mirrored pair "ltjri99\_1" has been imported as /dev/xvdb and the ltjri99\_2 as /dev/xvdc

## LVM

Setup the LVM Volume Group in the VM  
Login to the VM and with sudo/as root prepare the both devices for LVM usage with "pvcreate"  
ltjri99 # pvcreate /dev/xvdb ; pvcreate /dev/xvdc  
  
Create/extend the Volume Group:  
The Volume Group (aka "VG" or Disk Group or "DG") Names are defined by BMW as follows.  
At BMW "Virtual IP Names" and Cluster "IP Resources" are called "Packages or Pakete" = PKG.  
  
Everything EXCEPT SAP:  
dg(PKG)(mnt-pointname):  
eg: /dev/dgtldb0018vmdb/lvdb for mountpoint /global/tldb0018vm/db  
  
For SAP systems the Volume group names (LVM and Veritas) are named in accordance with their function:  
  
For a CI dg(PKG)ci01  
For a CS dg(PKG)cs01  
For a LiveCache Filesystem dg(LC\_PKG)lc01  
For a DI dg(PKG)di01  
  
If we were setting up a SAP DI, and the virtual IP in DNS is "tjridi55" we would need one DG called dgtjridi55di01  
If we were settings up a CI + LiveCache System with a CI IP tjrici55 and a LiveCache IP ltjrilc55, we would need two DGs, a dgltjrici55ci01 for the CI and a LiveCache DG dgltjrilc55lc01.  
  
On this VM we have a SAP DI with two (md mirrored) disks:  
  
ltjri99 # vgcreate dgtjridi55di01 /dev/vxdb  
  
Now add the secon disk to the Volume Group  
  
ltjri99 # vgextend dgtjridi55di01 /dev/vxdc  
  
Check with "vgdisplay -v <Volume Group> | egrep "PV Name"  
  
  
For SAP systems, the Logical Volume (aka LV) configuration is done via the Override "BMW-CFG-SAP-VM-FS-LAYOUT" via install\_override.sh on the Install Server.  
  
For all other systems, the LVs are set up manually. On non SAP systems the LVs are usually just called lv01 lv02 etc..  
  
lvcreate -L <SIZE\_IN\_MB>M -n lv01 /dev/<Volume Grouyp Name>  
OR  
lvcreate /dev/dgtldb0018vmbkup -l 100%FREE -n lvbkup  
To Use ALL free extents in the Volume Group !! NOT FOR Multi LVOL Volumegroups !!  
  
To Create a 20GB lvol01 and a 15GB lvol02:  
lvcreate -L 20480 -n lvol01 /dev/dgtest01  
lvcreate -L 15360 -n lvol02 /dev/dgtest01  
  
Create a FXFS Filsystem on the LVOL  
  
  
mkfs.xfs /dev/dg<pkgname><mnt-name>/lv<mnt-name>

e.g.

mkfs.xfs /dev/dgtldb0023vmdb/lvdb

Create Mountpoint & add to /etc/fstab, mount & set Oracle Rights:

ltdb0023vm:/home/qxf5649 # mkdir -p /global/tldb0023vm/db  
ltdb0023vm:/home/qxf5649 # vi /etc/fstab  
ltdb0023vm:/home/qxf5649 # egrep /global/tldb0023vm/db /etc/fstab  
/dev/dgtldb0023vmdb/lvdb /global/tldb0023vm/db xfs defaults 0 0  
ltdb0023vm:/home/qxf5649 # mount /global/tldb0023vm/db  
ltdb0023vm:/home/qxf5649 # df -hP /global/tldb0023vm/db  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/dgtldb0023vmdb-lvdb 450G 33M 450G 1% /global/tldb0023vm/db  
ltdb0023vm:/home/qxf5649 # chown oracle:dba /global/ltdb0023vm/db ; chmod 750 /global/ltdb0023vm/db

For swap DG LVOLs instead of mk.xfs, use the following:

mkswap -f /dev/dgtldb0018vmswap1/lvswap1  
mkswap -f /dev/dgtldb0018vmswap2/lvswap2  
  
  
  
Add to /etc/fstab  
/dev/dgtldb0018vmswap1/lvswap1 swap swap pri=42 0 0  
/dev/dgtldb0018vmswap2/lvswap2 swap swap pri=42 0 0  
  
  
Turn on:  
swapon -a  
  
Check:  
swapon -s

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by [Kujau Christian, (Christian.CK.Kujau@partner.bmw.de)](https://bsswiki.muc/tiki-user_information.php?userId=55) , Engert Michael, (Michael.Engert@partner.bmw.de) , Knopp Ekkehard, (Ekkehard.Knopp@partner.bmw.de) , Noyes Geoffrey, (Geoffrey.Noyes@partner.bmw.de) and [system](https://bsswiki.muc/tiki-user_information.php?userId=-1) .  
Page last modified on Wednesday 16 of October, 2019 14:44:59 CEST. (Version 8)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified by [Sanjiv Soraganvi (ext.)](https://atc.bmwgroup.net/confluence/display/~qxw9324) on [22 Dec 2021](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=547156350&selectedPageVersions=2&selectedPageVersions=3)

#### **Disk addition threshold limit - Number of Luns per Xenfarm and Per Xen VMs** -Farm server:  Total number of LUNs - 1000  (#san\_info -ls  |grep -i "Disks seen"|sort -u) -Virtual Server : Total number of Luns assigned (#vmc info <vmname> or san\_shortinfo |grep -i <vmname>, other steps)                               Minumum Limit = 25 Luns                               Maximum Limit = 30 Luns                 If the Luns on Farm  is equal or beyond limit then no more Luns addition to farm is possible. If its needed then disk consolidation is required.                If the Luns on VMs is min limit  you may add more Luns till max limit but its a trigger to initiate the disk consolidation first and then proceed addition/extension.

| Extending LVM Storage with AutoSAN Tools |
| --- |

# General informations

This documentation handles filesystem extensions for VMs which have [AutoSAN](https://atc.bmwgroup.net/confluence/x/3IUPHw) enabled. The installation tasks do not mention AutoSAN at all. It is your responsibility to find out, if AutoSAN can be used or not. In short you have following things to do:  
- decide if AutoSAN is available,  
- fetch the data needed to do the task,  
- ordering disks and attaching them to the VM,  
- add the disks to the correct volume group and extend the logical volume, and  
- finally extend the filesystem.  
  
Should AutoSAN not be available for your virtual server, there is a document describing [the manual filesystem extension for a VM](https://atc.bmwgroup.net/confluence/x/eEgNHw).

# Decide if AutoSAN is available

The foremost thing you should do is to decide if you can use AutoSAN. Two commands give you the information you need to decide:

[INSTSERV][qx12345][lpinstiaas01] ~ $ cmdb ltdb0178vm | grep -e "virtual server" -e "farmservers" -e "service contract"

virtual server: Y

farmservers: lpxentdb031, lpxentdb032

service contract: BSS\_CPX\_LI-UX\_S2\_VIRT

[INSTSERV][qx12345][lpinstiaas01] ~ $ check4autosan.pl ltdb0178vm

ltdb0178vm is a VM and has AutoSAN: FIZ06

Checking availability of AutoSAN

In the example above, the first command was to find out, if this server belongs to us, if it is a virtual server and what the XEN farm servers are. The second command checks the availability of AutoSAN. You see, that we have a VM in SAN FIZ06 and AutoSAN is available here.

# Fetching data

The next step is to fetch all the data you need to do this change. We start with the mount point, the directory where the file system has been mounted to.

ltdb0178vm:/global/tldb0178vm/db # df -h .

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/vgtldb0178vmdb-lvdb 900G 802G 99G 90% /global/tldb0178vm/db

Fetching volume group and logical volume name

We identify the volume group and the logical volume name in the above output in the column "Filesystem". The volume group is "vgtldb0178vmdb" and the logical volume name is "lvdb". As we know the volume group name now, we can look up the raw device(s) in the VM where the volume group lives on.

ltdb0178vm:/root # pvs

PV VG Fmt Attr PSize PFree

/dev/xvdb vgtldb0178vmdb lvm2 a- 1023.87g 124.03g

/dev/xvdf vgtldb0178vmredo1 lvm2 a- 14.06g 0

/dev/xvdg vgtldb0178vmredo2 lvm2 a- 14.06g 0

/dev/xvdh dgltdb0178vmaudit lvm2 a- 56.24g 0

/dev/xvdk vgtldb0178vmbkup lvm2 a- 506.32g 218.36g

Fetching the raw device name(s)

In our case the volume group "vgtldb0178vmdb" resides on the raw device "/dev/xvdb" and there are 124 GByte free. The free space can later be used for our extension.  
We worked in the VM. Now we switch to the XEN farm node where the VM is running at. "/dev/xvdb" is the name of the raw device in the VM, but we need the name of this device at the XEN farm node.

lpxentdb032:/root # vm info ltdb0178vm

Name : ltdb0178vm

Comment : N/A

Domain ID : 5

Status : Running on lpxentdb032

VM Type : PV

Mounts : None

Storage Type : SAN+MD

Image : sles11sp2\_x86\_64

Last Backup (UTC) : 2016-06-21 22:28:18

Cur. VCPUs : 6

Max. VCPUs : 6

Cur. Mem [MB] : 14336

Max. Mem [MB] : 14336

XML - Cur. VCPUs : 6

XML - Max. VCPUs : 6

XML - Cur. Mem [MB] : 14336

XML - Max. Mem [MB] : 14336

XML - Target State : 1

XML - Priority : 1

Network device 0 : 00:16:3e:2e:c3:e7 (br\_160046195128)

Network device 1 : 00:16:3e:64:7c:2d (br\_010100124000)

Block device xvda : /dev/md/ltdb0178vm

Block device xvdf : /dev/md/ltdb0178vm\_5

Block device xvdg : /dev/md/ltdb0178vm\_6

Block device xvdh : /dev/md/ltdb0178vm\_7

Block device xvdb : /dev/md/ltdb0178vm\_13

Block device xvdk : /dev/md/ltdb0178vm\_14

Converting from VM view to XEN farm node view

At the end of the vm info command we see a list of raw devices and their counterpart in XEN farm nodes view. The raw device "/dev/xvdb" is "/dev/md/ltdb0178vm\_13" on the XEN farm node. This is a symbolic link to the real MD device at the XEN farm node. So, we have to look up the MD device.

lpxentdb032:/root # l /dev/md/ltdb0178vm\_13

lrwxrwxrwx 1 root root 7 May 7 15:02 /dev/md/ltdb0178vm\_13 -> ../md67

Finally getting the MD device

At this point we have all information ready to find the LUN(s) that hold our filesystem. To do this, we look up the storage assigned to the VM.

lpxentdb032:/root # vm storage ltdb0178vm

17:22:52 Gathering storage data (please be patient)

17:22:52 Establishing libvirt connection to lpxentdb032 OK

Disk UUID Array LDEV Size Dev MDName MDDev VMName

360060e80164e390000014e39000000d7 85561 00d7 1024G dm-27 SAN:ltdb0178vm\_13 md/ltdb0178vm\_13 ltdb0178vm

360060e80164e390000014e39000000d8 85561 00d8 506G dm-28 SAN:ltdb0178vm\_14 md/ltdb0178vm\_14 ltdb0178vm

360060e80164e390000014e39000000d9 85561 00d9 14G dm-29 SAN:ltdb0178vm\_5 md/ltdb0178vm\_5 ltdb0178vm

360060e80164e390000014e39000000da 85561 00da 14G dm-30 SAN:ltdb0178vm\_6 md/ltdb0178vm\_6 ltdb0178vm

360060e80164e390000014e39000000db 85561 00db 225G dm-31 SAN:ltdb0178vm md/ltdb0178vm ltdb0178vm

360060e80164e390000014e39000000dc 85561 00dc 56G dm-32 SAN:ltdb0178vm\_7 md/ltdb0178vm\_7 ltdb0178vm

360060e80164e3f0000014e3f000000d7 85567 00d7 1024G dm-107 SAN:ltdb0178vm\_13 md/ltdb0178vm\_13 ltdb0178vm

360060e80164e3f0000014e3f000000d8 85567 00d8 506G dm-108 SAN:ltdb0178vm\_14 md/ltdb0178vm\_14 ltdb0178vm

360060e80164e3f0000014e3f000000d9 85567 00d9 14G dm-109 SAN:ltdb0178vm\_5 md/ltdb0178vm\_5 ltdb0178vm

360060e80164e3f0000014e3f000000da 85567 00da 14G dm-110 SAN:ltdb0178vm\_6 md/ltdb0178vm\_6 ltdb0178vm

360060e80164e3f0000014e3f000000db 85567 00db 225G dm-111 SAN:ltdb0178vm md/ltdb0178vm ltdb0178vm

360060e80164e3f0000014e3f000000dc 85567 00dc 56G dm-112 SAN:ltdb0178vm\_7 md/ltdb0178vm\_7 ltdb0178vm

Getting the LUN

We see in the above output, that our filesystem resides on the LUN "00:D7" on array "85561" and on the LUN "00:D7" on array "85567". The LUNs are 1024 GByte in size. As next step, we have to look up one of the LUNs to get the storage type and an order position. In the example below, the root filesystem (MDName: ltdb0178vm, LUN: 00:DB) has been choosen.

[INSTSERV][qx12345][lpinstiaas01] ~ $ check\_rid.sh lpxentdb032 | grep 00:DB

SFIZ06BV2 85561 00:00:DB 230490 000175122 B2\_GS2 LPXENTDB032 2016-04-28 00:00:00

SFIZ06BV1 85567 00:00:DB 230490 000175122 B2\_GS2 LPXENTDB032 2016-04-28 00:00:00

Getting an StorM order position

The important information is the order position, which is displayed in the fifth column. The StorM order position "175122" will be used in the next command, to get the list of hosts (the two XEN farm nodes in our case) where the new LUNs have to be assigned to.

[INSTSERV][qx12345][lpinstiaas01] ~ $ check\_rid.sh 175122

arrayname serial arrayldev size orderposid class secured\_server assigndate

SFIZ06BV2 85561 00:00:DB 230490 000175122 B2\_GS2 LPXENTDB031,LPXENTDB032 2016-04-28 00:00:00

SFIZ06BV1 85567 00:00:DB 230490 000175122 B2\_GS2 LPXENTDB031,LPXENTDB032 2016-04-28 00:00:00

Fetching host list

With the last two commands in the example above, you look up the servers to which the storage is currently assigned. If there is a VM migration to another farm in preparation, you would see more than two servers. In this case please look up the assignee of the appropriate migration or migration preparation task and coordinate your task with him.

# Ordering and attaching the new disks to the VM

When ordering disks with AutoSAN tools, there are two limits to keep in mind:

* you can't order more than 15 disks per ticket, and
* you can't order more than 2.5 TByte per ticket.

In our example the original disk with 1 TB storage and 125 GB free space on it will be replaced by three new disks with 506 GB storage each and 250 GB free space. Hence we use following command to order and attach three new disks to the VM:

[INSTSERV][qx12345][lpinstiaas01] ~ $ autosan.pl extend-vm -o TAS000001560865 -d 506:3 --vmname ltdb0178vm

20160622T180036 INFO: ARGV: -o TAS000001560865 -d 506:3 ltdb0178vm

Script started, file is /global/instserv/logs/full\_outputs/2016/06/22/2016-06-22\_18:00:36\_vm\_extend.ltdb0178vm.qxj5457.log

20160622T180036 INFO: ARGV: -o TAS000001560865 -d 506:3 ltdb0178vm

20160622T180046 INFO: Querying lpxentdb031, if it's enabled for AutoSAN...this may take a while

20160622T180046 INFO: Server lpxentdb031 is enabled to use AutoSAN

20160622T180047 INFO: Checking if ltdb0178vm has a /etc/.donotrescan file...

20160622T180047 INFO: Done. No /etc/.donotrescan file found.

20160622T180047 INFO: Querying lpxentdb032, if it's enabled for AutoSAN...this may take a while

20160622T180047 INFO: Server lpxentdb032 is enabled to use AutoSAN

20160622T180048 INFO: Checking if ltdb0178vm has a /etc/.donotrescan file...

20160622T180048 INFO: Done. No /etc/.donotrescan file found.

20160622T180048 INFO: lpxentdb031 lpxentdb032 are in module bu Xen-Farmserver Datenbank-BU, so using APP-ID APP-108617

20160622T180048 INFO: Querying WWNs for server lpxentdb031...

20160622T180049 INFO: More than 2 online ports were found on server lpxentdb031! Auto detecting WWNs

20160622T180059 INFO: Found following WWNs on server lpxentdb031: 10:00:00:90:fa:a9:15:6a 10:00:00:90:fa:a9:15:8e

20160622T180059 INFO: Querying WWNs for server lpxentdb032...

20160622T180100 INFO: More than 2 online ports were found on server lpxentdb032! Auto detecting WWNs

20160622T180112 INFO: Found following WWNs on server lpxentdb032: 10:00:00:90:fa:a0:f3:d2 10:00:00:90:fa:a0:f2:46

20160622T180112 INFO: Querying SAN names for servers lpxentdb031 lpxentdb032...this may take a while

20160622T180112 INFO: Done, SAN name is FIZ06

20160622T180112 INFO: Submitting storage order ticket...

20160622T180115 INFO: Ordered disks with ticket ID 178427

20160622T180116 INFO: 178427: Accepted - automatically accepted

20160622T180116 INFO: this may take a while ...

20160622T180157 INFO: 178427: WorkInProgress - work in progress

20160622T180958 INFO: 178427: Closed - SQL Exception in storageOrder.updateTicket(): LUN: (00:00:98/ArraySerial:85567) already exists in storm!

20160622T181001 INFO: Checking if LUNs are available on servers lpxentdb031 lpxentdb032..

20160622T181001 INFO: Rescanning farmserver lpxentdb031

20160622T181001 INFO: rootssh lpxentdb031 san\_rescan -o

stty: standard input: Invalid argument

stty: standard input: Invalid argument

\*\*

\*\* HBA Check...

Starting with PID '26707' and options '-o'

- Checking host1 (0x10000090faa9156a): 'Link Up - Ready Fabric'

- Checking host2 (0x10000090faa9156b): 'Link Up - Ready Fabric'

- Checking host3 (0x10000090faa9158e): 'Link Up - Ready Fabric'

- Checking host4 (0x10000090faa9158f): 'Link Up - Ready Fabric'

\*\* Cleaning up SCSI Devices...

- No unused SCSI devices found

\*\* Cleaning up zero sized SCSI Devices

- No zero sized SCSI devices found

\*\*

\*\* Starting rescan of FC hosts 'host1 host2 host3 host4'.

Rescanning host1

= Triggering Scan on host1

- Checking host1:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 6 new os-device(s) on host1

Rescanning host2

= Triggering Scan on host2

- Checking host2:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 0 new os-device(s) on host2

Rescanning host3

= Triggering Scan on host3

- Checking host3:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 6 new os-device(s) on host3

Rescanning host4

= Triggering Scan on host4

- Checking host4:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Device summary

Found 0 new os-device(s) on host4

- Found 12 new devices: sdap sday sdaz sdcd sdcy sdcz sdda sddb sdam sdao sdan sdax

\*\* Looking for new LVM devices

- New LVM device(s):

DM | RAW CU:LDEV PORT XP-SN SIZE | RAW CU:LDEV PORT XP-SN SIZE |

-------|---------------------------------|---------------------------------|

dm-54 | sdam 00:98 7C 85561 518602 | sdcz 00:98 8C 85561 518602 |

dm-63 | sdax 00:99 7C 85567 518602 | sdcd 00:99 8C 85567 518602 |

dm-61 | sdap 00:98 7C 85567 518602 | sdaz 00:98 8C 85567 518602 |

dm-56 | sdao 00:9A 7C 85561 518602 | sddb 00:9A 8C 85561 518602 |

dm-62 | sday 00:9A 7C 85567 518602 | sdcy 00:9A 8C 85567 518602 |

dm-55 | sdan 00:99 7C 85561 518602 | sdda 00:99 8C 85561 518602 |

-

\*\* Found 6 new LVM device(s)

Normal exit after 17 sec.

20160622T181018 INFO: Done rescanning host lpxentdb031

20160622T181018 INFO: Waiting 10 seconds to settle multipath device nodes on lpxentdb031

20160622T181028 INFO: Checking if LUNs are available on server lpxentdb031

20160622T181044 INFO: LUN 00:99, 85561 was found on lpxentdb031

20160622T181044 INFO: LUN 00:9A, 85561 was found on lpxentdb031

20160622T181044 INFO: LUN 00:98, 85561 was found on lpxentdb031

20160622T181044 INFO: LUN 00:98, 85567 was found on lpxentdb031

20160622T181044 INFO: LUN 00:99, 85567 was found on lpxentdb031

20160622T181044 INFO: LUN 00:9A, 85567 was found on lpxentdb031

20160622T181044 INFO: Found all new LUNs for VM ltdb0178vm on farmserver lpxentdb031

20160622T181044 INFO: Rescanning farmserver lpxentdb032

20160622T181044 INFO: rootssh lpxentdb032 san\_rescan -o

stty: standard input: Invalid argument

stty: standard input: Invalid argument

\*\* Starting with PID '10725' and options '-o'

\*\* HBA Check...

- Checking host1 (0x10000090faa0f246): 'Link Up - Ready Fabric'

- Checking host2 (0x10000090faa0f247): 'Link Up - Ready Fabric'

- Checking host3 (0x10000090faa0f3d2): 'Link Up - Ready Fabric'

- Checking host4 (0x10000090faa0f3d3): 'Link Up - Ready Fabric'

\*\* Cleaning up SCSI Devices...

- No unused SCSI devices found

\*\* Cleaning up zero sized SCSI Devices

- No zero sized SCSI devices found

\*\*

\*\* Starting rescan of FC hosts 'host1 host2 host3 host4'.

Rescanning host1

= Triggering Scan on host1

- Checking host1:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 6 new os-device(s) on host1

Rescanning host2

= Triggering Scan on host2

- Checking host2:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 0 new os-device(s) on host2

Rescanning host3

= Triggering Scan on host3

- Checking host3:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Found 6 new os-device(s) on host3

Rescanning host4

= Triggering Scan on host4

- Checking host4:

-> State: 'Link Up - Ready Fabric' up.

-> Udev: done.

=

\*\* Device summary

Found 0 new os-device(s) on host4

- Found 12 new devices: sday sdap sdaz sdcd sdcy sdda sdcz sddb sdam sdan sdao sdax

\*\* Looking for new LVM devices

- New LVM device(s):

DM | RAW CU:LDEV PORT XP-SN SIZE | RAW CU:LDEV PORT XP-SN SIZE |

-------|---------------------------------|---------------------------------|

dm-55 | sdam 00:98 7C 85561 518602 | sdaz 00:98 8C 85561 518602 |

dm-612 | sdib 00:83 7D 85561 518602 | sdajn 00:83 8D 85561 518602 |

dm-561 | sdgh 01:17 7D 85561 230490 | sdaht 01:17 8D 85561 230490 |

dm-62 | sdax 00:99 7C 85567 518602 | sdda 00:99 8C 85567 518602 |

dm-625 | sdin 00:8F 7D 85561 1048576 | sdajz 00:8F 8D 85561 1048576 |

dm-551 | sdfy 01:0E 7D 85561 115245 | sdahk 01:0E 8D 85561 115245 |

dm-563 | sdfi 00:FE 7D 85561 518602 | sdagu 00:FE 8D 85561 518602 |

dm-611 | sdia 00:82 7D 85561 518602 | sdajm 00:82 8D 85561 518602 |

dm-560 | sdgg 01:16 7D 85561 230490 | sdahs 01:16 8D 85561 230490 |

dm-61 | sdap 00:98 7C 85567 518602 | sdcz 00:98 8C 85567 518602 |

dm-624 | sdim 00:8E 7D 85561 57622 | sdajy 00:8E 8D 85561 57622 |

dm-550 | sdfx 01:0D 7D 85561 115245 | sdahj 01:0D 8D 85561 115245 |

dm-638 | sdjh 01:0E 7D 85567 115245 | sdakt 01:0E 8D 85567 115245 |

dm-552 | sdfh 00:FD 7D 85561 518602 | sdagt 00:FD 8D 85561 518602 |

dm-610 | sdhz 00:81 7D 85561 518602 | sdajl 00:81 8D 85561 518602 |

dm-559 | sdgf 01:15 7D 85561 230490 | sdahr 01:15 8D 85561 230490 |

dm-623 | sdil 00:8D 7D 85561 14405 | sdajx 00:8D 8D 85561 14405 |

dm-549 | sdfw 01:0C 7D 85561 115245 | sdahi 01:0C 8D 85561 115245 |

dm-637 | sdjg 01:0D 7D 85567 115245 | sdaks 01:0D 8D 85567 115245 |

dm-541 | sdfg 00:FC 7D 85561 518602 | sdags 00:FC 8D 85561 518602 |

dm-639 | sdiq 00:FD 7D 85567 518602 | sdakc 00:FD 8D 85567 518602 |

dm-558 | sdge 01:14 7D 85561 230490 | sdahq 01:14 8D 85561 230490 |

dm-622 | sdik 00:8C 7D 85561 14405 | sdajw 00:8C 8D 85561 14405 |

dm-548 | sdfv 01:0B 7D 85561 115245 | sdahh 01:0B 8D 85561 115245 |

dm-636 | sdjf 01:0C 7D 85567 115245 | sdakr 01:0C 8D 85567 115245 |

dm-628 | sdip 00:FC 7D 85567 518602 | sdakb 00:FC 8D 85567 518602 |

dm-557 | sdgd 01:13 7D 85561 230490 | sdahp 01:13 8D 85561 230490 |

dm-621 | sdij 00:8B 7D 85561 14405 | sdajv 00:8B 8D 85561 14405 |

dm-547 | sdfu 01:0A 7D 85561 115245 | sdahg 01:0A 8D 85561 115245 |

dm-635 | sdje 01:0B 7D 85567 115245 | sdakq 01:0B 8D 85567 115245 |

dm-556 | sdgc 01:12 7D 85561 230490 | sdaho 01:12 8D 85561 230490 |

dm-620 | sdii 00:8A 7D 85561 115245 | sdaju 00:8A 8D 85561 115245 |

dm-569 | sdgo 01:1E 7D 85561 14405 | sdaia 01:1E 8D 85561 14405 |

dm-634 | sdjd 01:0A 7D 85567 115245 | sdakp 01:0A 8D 85567 115245 |

dm-546 | sdft 01:09 7D 85561 115245 | sdahf 01:09 8D 85561 115245 |

dm-555 | sdgb 01:11 7D 85561 14405 | sdahn 01:11 8D 85561 14405 |

dm-568 | sdgn 01:1D 7D 85561 57622 | sdahz 01:1D 8D 85561 57622 |

dm-619 | sdih 00:89 7D 85561 230490 | sdajt 00:89 8D 85561 230490 |

dm-545 | sdfs 01:08 7D 85561 115245 | sdahe 01:08 8D 85561 115245 |

dm-633 | sdjc 01:09 7D 85567 115245 | sdako 01:09 8D 85567 115245 |

dm-554 | sdga 01:10 7D 85561 57622 | sdahm 01:10 8D 85561 57622 |

dm-567 | sdgm 01:1C 7D 85561 230490 | sdahy 01:1C 8D 85561 230490 |

dm-617 | sdig 00:88 7D 85561 230490 | sdajs 00:88 8D 85561 230490 |

dm-544 | sdfr 01:07 7D 85561 518602 | sdahd 01:07 8D 85561 518602 |

dm-632 | sdjb 01:08 7D 85567 115245 | sdakn 01:08 8D 85567 115245 |

dm-566 | sdgl 01:1B 7D 85561 230490 | sdahx 01:1B 8D 85561 230490 |

dm-616 | sdif 00:87 7D 85561 230490 | sdajr 00:87 8D 85561 230490 |

dm-543 | sdfq 01:06 7D 85561 518602 | sdahc 01:06 8D 85561 518602 |

dm-631 | sdja 01:07 7D 85567 518602 | sdakm 01:07 8D 85567 518602 |

dm-565 | sdgk 01:1A 7D 85561 230490 | sdahw 01:1A 8D 85561 230490 |

dm-615 | sdie 00:86 7D 85561 518602 | sdajq 00:86 8D 85561 518602 |

dm-542 | sdfp 01:05 7D 85561 518602 | sdahb 01:05 8D 85561 518602 |

dm-630 | sdiz 01:06 7D 85567 518602 | sdakl 01:06 8D 85567 518602 |

dm-56 | sdao 00:9A 7C 85561 518602 | sdcy 00:9A 8C 85561 518602 |

dm-626 | sdfo 01:04 7D 85561 518602 | sdaha 01:04 8D 85561 518602 |

dm-614 | sdid 00:85 7D 85561 518602 | sdajp 00:85 8D 85561 518602 |

dm-564 | sdgj 01:19 7D 85561 230490 | sdahv 01:19 8D 85561 230490 |

dm-629 | sdiy 01:05 7D 85567 518602 | sdakk 01:05 8D 85567 518602 |

dm-63 | sday 00:9A 7C 85567 518602 | sddb 00:9A 8C 85567 518602 |

dm-54 | sdan 00:99 7C 85561 518602 | sdcd 00:99 8C 85561 518602 |

dm-618 | sdfn 01:03 7D 85561 518602 | sdagz 01:03 8D 85561 518602 |

dm-613 | sdic 00:84 7D 85561 518602 | sdajo 00:84 8D 85561 518602 |

dm-562 | sdgi 01:18 7D 85561 230490 | sdahu 01:18 8D 85561 230490 |

dm-553 | sdfz 01:0F 7D 85561 115245 | sdahl 01:0F 8D 85561 115245 |

- Found 6 new LVM device(s)

\*\*

Normal exit after 19 sec.

20160622T181103 INFO: Done rescanning host lpxentdb032

20160622T181103 INFO: Waiting 10 seconds to settle multipath device nodes on lpxentdb032

20160622T181113 INFO: Checking if LUNs are available on server lpxentdb032

20160622T181128 INFO: LUN 00:99, 85561 was found on lpxentdb032

20160622T181128 INFO: LUN 00:9A, 85561 was found on lpxentdb032

20160622T181128 INFO: LUN 00:98, 85561 was found on lpxentdb032

20160622T181128 INFO: LUN 00:98, 85567 was found on lpxentdb032

20160622T181128 INFO: LUN 00:99, 85567 was found on lpxentdb032

20160622T181128 INFO: LUN 00:9A, 85567 was found on lpxentdb032

20160622T181128 INFO: Found all new LUNs for VM ltdb0178vm on farmserver lpxentdb032

20160622T181128 INFO: Adding LUN 00:00:98 for VM ltdb0178vm on farmserver lpxentdb032 ...

20160622T181128 INFO: echo "vmc adddisk -d '85561-0098 85567-0098' ltdb0178vm" | rootssh lpxentdb032 bash

18:11:29 Checking farm status before running command 'adddisk'

18:11:32 Verifying free config space on lpxentdb031 lpxentdb032 OK

18:11:32 Creating new MD device for VM ltdb0178vm

18:11:48 Using disks 360060e80164e390000014e3900000098 360060e80164e3f0000014e3f00000098 to create array

18:11:48 Creating md device md/ltdb0178vm\_15 with mdname SAN:ltdb0178vm\_15 OK

18:11:48 Waiting for udev to settle OK

18:11:48 Stopping md device /dev/md/ltdb0178vm\_15 on lpxentdb032 OK

18:11:50 Starting md device /dev/md/ltdb0178vm\_15 on lpxentdb032 OK

18:11:55 Waiting for udev to settle OK

18:12:23 Attaching device /dev/md/ltdb0178vm\_15 to VM ltdb0178vm as xvdc OK

18:12:26 Updating definition of ltdb0178vm on lpxentdb031 OK

18:12:26 Checking if write-mostly should be set for this VM

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm to dm-111 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_13 to dm-107 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_14 to dm-108 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_15 to dm-61 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_5 to dm-109 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_6 to dm-110 OK

18:12:34 Setting write-mostly on lpxentdb032 for ltdb0178vm\_7 to dm-112 OK

20160622T181234 INFO: LUN 00:00:98 successfully added for VM ltdb0178vm

20160622T181234 INFO: Adding LUN 00:00:9A for VM ltdb0178vm on farmserver lpxentdb032 ...

20160622T181234 INFO: echo "vmc adddisk -d '85561-009a 85567-009a' ltdb0178vm" | rootssh lpxentdb032 bash

18:12:35 Checking farm status before running command 'adddisk'

18:12:38 Verifying free config space on lpxentdb031 lpxentdb032 OK

18:12:38 Creating new MD device for VM ltdb0178vm

18:12:54 Using disks 360060e80164e390000014e390000009a 360060e80164e3f0000014e3f0000009a to create array

18:12:54 Creating md device md/ltdb0178vm\_16 with mdname SAN:ltdb0178vm\_16 OK

18:12:55 Waiting for udev to settle OK

18:12:55 Stopping md device /dev/md/ltdb0178vm\_16 on lpxentdb032 OK

18:12:56 Starting md device /dev/md/ltdb0178vm\_16 on lpxentdb032 OK

18:13:02 Waiting for udev to settle OK

18:13:30 Attaching device /dev/md/ltdb0178vm\_16 to VM ltdb0178vm as xvdd OK

18:13:32 Updating definition of ltdb0178vm on lpxentdb031 OK

18:13:32 Checking if write-mostly should be set for this VM

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm to dm-111 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_13 to dm-107 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_14 to dm-108 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_15 to dm-61 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_16 to dm-63 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_5 to dm-109 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_6 to dm-110 OK

18:13:41 Setting write-mostly on lpxentdb032 for ltdb0178vm\_7 to dm-112 OK

20160622T181341 INFO: LUN 00:00:9A successfully added for VM ltdb0178vm

20160622T181341 INFO: Adding LUN 00:00:99 for VM ltdb0178vm on farmserver lpxentdb032 ...

20160622T181341 INFO: echo "vmc adddisk -d '85561-0099 85567-0099' ltdb0178vm" | rootssh lpxentdb032 bash

18:13:42 Checking farm status before running command 'adddisk'

18:13:44 Verifying free config space on lpxentdb031 lpxentdb032 OK

18:13:45 Creating new MD device for VM ltdb0178vm

18:14:02 Using disks 360060e80164e390000014e3900000099 360060e80164e3f0000014e3f00000099 to create array

18:14:02 Creating md device md/ltdb0178vm\_17 with mdname SAN:ltdb0178vm\_17 OK

18:14:02 Waiting for udev to settle OK

18:14:02 Stopping md device /dev/md/ltdb0178vm\_17 on lpxentdb032 OK

18:14:04 Starting md device /dev/md/ltdb0178vm\_17 on lpxentdb032 OK

18:14:09 Waiting for udev to settle OK

18:14:35 Attaching device /dev/md/ltdb0178vm\_17 to VM ltdb0178vm as xvde OK

18:14:37 Updating definition of ltdb0178vm on lpxentdb031 OK

18:14:37 Checking if write-mostly should be set for this VM

18:14:45 Setting write-mostly on lpxentdb032 for ltdb0178vm to dm-111 OK

18:14:45 Setting write-mostly on lpxentdb032 for ltdb0178vm\_13 to dm-107 OK

18:14:45 Setting write-mostly on lpxentdb032 for ltdb0178vm\_14 to dm-108 OK

18:14:45 Setting write-mostly on lpxentdb032 for ltdb0178vm\_15 to dm-61 OK

18:14:45 Setting write-mostly on lpxentdb032 for ltdb0178vm\_16 to dm-63 OK

18:14:46 Setting write-mostly on lpxentdb032 for ltdb0178vm\_17 to dm-62 OK

18:14:46 Setting write-mostly on lpxentdb032 for ltdb0178vm\_5 to dm-109 OK

18:14:46 Setting write-mostly on lpxentdb032 for ltdb0178vm\_6 to dm-110 OK

18:14:46 Setting write-mostly on lpxentdb032 for ltdb0178vm\_7 to dm-112 OK

20160622T181446 INFO: LUN 00:00:99 successfully added for VM ltdb0178vm

Script done, file is /global/instserv/logs/full\_outputs/2016/06/22/2016-06-22\_18:00:36\_vm\_extend.ltdb0178vm.qxj5457.log

Ordering and attaching new LUNs to a VM

Please look carefully at the output of the above command. You would realize, that three new disks have been added to the VM and made available as /dev/xvdc, /dev/xvdd and /dev/xvde in the VM.

# Configuring volume group and logical volume

Now we have three new raw disks, which have to be prepaired with physical volume headers:

ltdb0178vm:/global/tldb0178vm/db # pvcreate /dev/xvdc

Physical volume "/dev/xvdc" successfully created

ltdb0178vm:/global/tldb0178vm/db # pvcreate /dev/xvdd

Physical volume "/dev/xvdd" successfully created

ltdb0178vm:/global/tldb0178vm/db # pvcreate /dev/xvde

Physical volume "/dev/xvde" successfully created

Prepairing raw volumes

The prepaired raw disks may now be added to the already existing volume group "vgtldb0178vmdb":

ltdb0178vm:/global/tldb0178vm/db # vgextend vgtldb0178vmdb /dev/xvdc /dev/xvdd /dev/xvde

Volume group "vgtldb0178vmdb" successfully extended

Adding disks to the volume group

# Moving data from old to new disks

As we want replace one disk with three new disks, we have to move the physical extends from the old disk /dev/xvdb to the new disks. This step is not neccessary, if you just want to extend a filesystem. When this has been done, we can remove the old disk from the volume group and wipe the LVM headers from it.

ltdb0178vm:/global/tldb0178vm/db # pvmove -n lvdb /dev/xvdb /dev/xvdc /dev/xvdd /dev/xvde

ltdb0178vm:/global/tldb0178vm/db # vgreduce vgtldb0178vmdb /dev/xvdb

Removed "/dev/xvdb" from volume group "vgtldb0178vmdb"

ltdb0178vm:/global/tldb0178vm/db # pvremove /dev/xvdb

Labels on physical volume "/dev/xvdb" successfully wiped

Removing old disks from the volume group

The above steps are only neccessary, when you use the filesystem extension to clean the filesystem up. Normally you will just add some disks and need not to remove old disks.

# Extending logical volume and filesystem

As soon as the volume group has its desired size, the logical volume can be extended. When this is done, we can grow the XFS filesystem on the disk:

ltdb0178vm:/global/tldb0178vm/db # lvextend -l +100%FREE /dev/vgtldb0178vmdb/lvdb

Extending logical volume lvdb to 1.48 TiB

Logical volume lvdb successfully resized

ltdb0178vm:/global/tldb0178vm/db # xfs\_growfs /global/tldb0178vm/db

meta-data=/dev/mapper/vgtldb0178vmdb-lvdb isize=256 agcount=8, agsize=29486080 blks

= sectsz=512 attr=2

data = bsize=4096 blocks=235888640, imaxpct=25

= sunit=0 swidth=0 blks

naming =version 2 bsize=4096 ascii-ci=0

log =internal bsize=4096 blocks=57590, version=2

= sectsz=512 sunit=0 blks, lazy-count=1

realtime =none extsz=4096 blocks=0, rtextents=0

data blocks changed from 235888640 to 398186496

ltdb0178vm:/global/tldb0178vm/db # df -h .

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/vgtldb0178vmdb-lvdb 1.5T 622G 898G 41% /global/tldb0178vm/db

Extending logical volume and grow filesystem

# Clean up old disks

If old disks were replaced by new ones, we need to remove them from the VM and return them back to the storage pool. To keep the overview, we first display the currently attached devices and then we remove the old disks.

lpxentdb032:/root # vm info ltdb0178vm

Name : ltdb0178vm

Comment : N/A

Domain ID : 5

Status : Running on lpxentdb032

VM Type : PV

Mounts : None

Storage Type : SAN+MD

Image : sles11sp2\_x86\_64

Last Backup (UTC) : 2016-06-22 22:20:43

Cur. VCPUs : 6

Max. VCPUs : 6

Cur. Mem [MB] : 14336

Max. Mem [MB] : 14336

XML - Cur. VCPUs : 6

XML - Max. VCPUs : 6

XML - Cur. Mem [MB] : 14336

XML - Max. Mem [MB] : 14336

XML - Target State : 1

XML - Priority : 1

Network device 0 : 00:16:3e:2e:c3:e7 (br\_160046195128)

Network device 1 : 00:16:3e:64:7c:2d (br\_010100124000)

Block device xvda : /dev/md/ltdb0178vm

Block device xvdf : /dev/md/ltdb0178vm\_5

Block device xvdg : /dev/md/ltdb0178vm\_6

Block device xvdh : /dev/md/ltdb0178vm\_7

Block device xvdb : /dev/md/ltdb0178vm\_13

Block device xvdk : /dev/md/ltdb0178vm\_14

Block device xvdc : /dev/md/ltdb0178vm\_15

Block device xvdd : /dev/md/ltdb0178vm\_16

Block device xvde : /dev/md/ltdb0178vm\_17

lpxentdb032:/root # vm -wp rmdisk ltdb0178vm ltdb0178vm\_13

17:31:36 Checking farm status before running command 'rmdisk'

17:31:36 Establishing libvirt connection to lpxentdb031 OK

17:31:37 Establishing libvirt connection to lpxentdb032 OK

17:31:39 Verifying free config space on lpxentdb031 lpxentdb032 OK

17:31:47 Detaching block device /dev/md/ltdb0178vm\_13 from VM ltdb0178vm OK

17:31:49 Updating definition of ltdb0178vm on lpxentdb031 OK

17:31:49 Stopping md device /dev/md/ltdb0178vm\_13 on lpxentdb032 OK

17:31:50 Removing md device /dev/md/ltdb0178vm\_13 from ltdb0178vm's

mdadm.conf OK

17:31:51 /dev/md/ltdb0178vm\_13 removed from ltdb0178vm

17:31:51 Wiping header of device

/dev/disk/by-id/dm-uuid-mpath-360060e80164e390000014e39000000d7 OK

17:31:51 Wiping header of device

/dev/disk/by-id/dm-uuid-mpath-360060e80164e3f0000014e3f000000d7 OK

17:31:51 Preparing pvinfo output for storage return of removed disks...

--------------------------------------------------------------------------

/dev/dm-27 00 00 00 SAN:ltdb0178vm\_13 00:d7 NA 00085561

/dev/dm-107 00 00 00 SAN:ltdb0178vm\_13 00:d7 NA 00085567

--------------------------------------------------------------------------

17:31:51 Deleting SCSI devices for removed disk for VM ltdb0178vm on lpxentdb031

17:31:53 Deleting SCSI devices for removed disk for VM ltdb0178vm on lpxentdb032

Returning old disks

Now it is time to create a SAN return ticket to give the old storage back into the pool:

Ticket Details

SAN GS II: Return Ticket

Ticket ID: 178511

Location: FIZ Remedy CM Change ID: CRQ000000448391 Earliest Delivery Date: 23.06.2016

Target SAN: FIZ06 (19.0 2.1/13.4 1.2) Remedy CM Task ID: TAS000001560865 Latest Delivery Date: 24.06.2016

Remove Cables: No Approval required: No

Return Positions

Array Name Array Serial LDEV Servername

SFIZ06BV1 85567 00:00:D7 lpxentdb032,lpxentdb031

SFIZ06BV2 85561 00:00:D7 lpxentdb032,lpxentdb031

Ticket Worklog

Timestamp Status Comment Username

24.06.2016 08:38:40 Closed Task completed [LUNs successfully returned] Cordt, Heiko

24.06.2016 07:08:03 WorkInProgress Processing started by Provider Cordt, Heiko

24.06.2016 06:51:21 Accepted - Ticket assigned to user [Cordt, Heiko] Cordt, Heiko

23.06.2016 17:43:18 Submitted Ticket submitted by Customer Engert, Michael

It is always a good idea to edit the .donotrescan files on the both XEN farm nodes and add the task number and the StorM order ID to the automatically added entries.

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by [qxj5457](https://bsswiki.muc/tiki-user_information.php?userId=-1) and Klein Karin, (Karin.Klein@partner.bmw.de) .  
Page last modified on Monday 21 of October, 2019 18:51:54 CEST. (Version 26)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified by [Sanjiv Soraganvi (ext.)](https://atc.bmwgroup.net/confluence/display/~qxw9324) on [22 Dec 2021](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=520964200&selectedPageVersions=4&selectedPageVersions=5)

#### **Disk addition threshold limit - Number of Luns per Xenfarm and Per Xen VMs** -Farm server:  Total number of LUNs - 1000  (#san\_info -ls  |grep -i "Disks seen"|sort -u) -Virtual Server : Total number of Luns assigned (#vmc info <vmname> or san\_shortinfo |grep -i <vmname>, other steps)                               Minumum Limit = 25 Luns                               Maximum Limit = 30 Luns                 If the Luns on Farm  is equal or beyond limit then no more Luns addition to farm is possible. If its needed then disk consolidation is required.                If the Luns on VMs is min limit  you may add more Luns till max limit but its a trigger to initiate the disk consolidation first and then proceed addition/extension.

| Grow existing VG/LV of a Linux LVM |
| --- |

This page describes the neccessary steps to expand a XFS filesystem residing on a logical volume of LVM.  
  
**Important note:** Before you start any activities, please **always** do a full check and look for any special configuration regarding SAN LUN's or disk layouts in Base Server Details of [our SODB](https://dcodb.bmwgroup.net/dcodb/sodb/pages/baseServerDetails.jsf).

Our example filesystem

ltdb0007vm:/home/qxf6163 # df -hP /global/tldb0007vm/db

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/dgtldb0007vm\_ltdb0007vm\_db-lv01 169G 136G 34G 81% /global/tldb0007vm/db

* **Do a fullcheck**.

[INSTSERV][qx12345][lpinstiaas01] ~ $ fullcheck ltdb0007vm

* **Check for space** in the volume group (this is dgtldb0007vm\_ltdb0007vm in our example). As we see in the output below, there are still 56GB storage free and we can jump to the point "Expand the volume group". In most cases, storage will be short and we will have to go through the following steps.
* ltdb0007vm # vgdisplay dgtldb0007vm\_ltdb0007vm\_db | egrep Free

Free PE / Size 14484 / 56.58 GiB

* **Order neccessary storage** in [StorM-Tool](https://apex-p.bmwgroup.net:4950/pls/htmldb/f?p=215:1:).  
  Have a look at [XEN VM Grow LV](https://atc.bmwgroup.net/confluence/x/eEgNHw) to see how you can figure out what disks you need.
* **Make the storage visible.**  
  You have to do that on both farmservers.
* lpxentdb017:~ # san\_rescan -o

lpxentdb018:~ # san\_rescan -o

* **Add storage to the VM.** Add each ordered pair of disks one by one and don't mix up the pairs of B-storage. In our example, there is only one pair of disks to add.

lpxentdb017:~ # vmc adddisk -d "360060e80167b810000017b81000000e0 360060e80167bd40000017bd4000000e0" ltdb0026vm

* **Create physical volume in VM.**

ltdb0007vm:~ # pvcreate /dev/xvdo

* **Extend the volume group.**

ltdb0007vm:~ # vgextend dgtldb0026vmdb /dev/xvdo

* **Expand the volume group** by 56 GB with the following command:

ltdb0007vm:~ # lvextend -L +56G /dev/dgtldb0007vm\_ltdb0007vm\_db/lv01

Extending logical volume lv01 to 224.50 GiB

Logical volume lv01 successfully resized

* **Expand the XFS Filesystem** using all the new space (56GB in our example). Please always use the whole disk for the extension!
* ltdb0007vm # xfs\_growfs /global/tldb0007vm/db
* meta-data=/dev/mapper/dgtldb0007vm\_ltdb0007vm\_db-lv01 isize=256 agcount=6, agsize=7372800 blks
* = sectsz=512 attr=2
* data = bsize=4096 blocks=44171264, imaxpct=25
* = sunit=0 swidth=0 blks
* naming =version 2 bsize=4096 ascii-ci=0
* log =internal bsize=4096 blocks=14400, version=2
* = sectsz=512 sunit=0 blks, lazy-count=1
* realtime =none extsz=4096 blocks=0, rtextents=0

data blocks changed from 44171264 to 58851328

* **Check new filesystem size**
* ltdb0007vm:/home/qxf6163 # df -hP /global/tldb0007vm/db
* Filesystem Size Used Avail Use% Mounted on

/dev/mapper/dgtldb0007vm\_ltdb0007vm\_db-lv01 225G 136G 90G 61% /global/tldb0007vm/db

**BUGs**  
  
If xfs\_growfs fails with "Killed" and messages like

" \_xfs\_buf\_find: Block out of range:

BUG: unable to handle kernel NULL pointer dereference "

upgrade the kernel.  
  
On 29.11.2013 12:54, Poeschl Andreas, FG-940DG wrote:  
... Der 3.0.93er Kernel hat aber ein großes Problem: XFS-Filesysteme lassen sich nicht erweitern. Dies trifft uns insbesondere in VMs hart. Dafür gibt es nun den 3.0.101er Kernel, der auch dieses Problem behebt ...

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by Puskas Andras, (Andras.AP.Puskas@partner.bmw.de) , [qxj5457](https://bsswiki.muc/tiki-user_information.php?userId=-1) , Kemmerer Walter, (Walter.Kemmerer@partner.bmw.de) , Linden Sylvia, (Sylvia.LA.Linden@partner.bmw.de) , Urban Richard, (Richard.Urban@partner.bmw.de) , Noyes Geoffrey, (Geoffrey.Noyes@partner.bmw.de) and [system](https://bsswiki.muc/tiki-user_information.php?userId=-1) .  
Page last modified on Wednesday 16 of October, 2019 14:48:12 CEST. (Version 23)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified on [08 Apr 2020](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=520964241&selectedPageVersions=1&selectedPageVersions=2)

| Linux LVM Problems and Solutions |
| --- |

LVM: read failed after 0 of 4096

**Problem**  
  
**NOTE!** This only applies to "Logical Volume" errors, **NOT** block devices such as /dev/xvdb /dev/sda5 etc. If uncertain, then please **contact technical expert**!

When running LVM commands such as pvs, vgs, lvs:

lpmodthk05:~ # pvs

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286413824: Input/output error

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286471168: Input/output error

PV VG Fmt Attr PSize PFree

/dev/xvdb dgpmodthkapps lvm2 a- 506.32g 0

/dev/xvdc dgpmodthkapps lvm2 a- 506.32g 0

..

lpmodthk05:~ # vgs

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286413824: Input/output error

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286471168: Input/output error

VG #PV #LV #SN Attr VSize VFree

dgpmodthkapps 5 1 0 wz--n- 2.47t 0

dgswap 1 1 0 wz--n- 14.06g 13.06g

Here, the LV /dev/dgpmodthkapps/lv01 has problems - most probably due to LUNs being removed without a clean shutdown or mount, or detaching disks from logical volumes.

**Solution**  
  
The steps involved in repairing:  
**1)** Find the affected VG: run vgscan  
**2)** Find the LV's attached to the VG  
**3)** Deactivate the LVs (lvchange -an LV)

lpmodthk05:~ # lvchange -an /dev/dgpmodthkapps/lv01

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286413824: Input/output error

/dev/dgpmodthkapps/lv01: read failed after 0 of 4096 at 2718286471168: Input/output error

**4)** Deactivate the VG (vgchange -an VG)

lpmodthk05:~ # vgchange -an dgpmodthkapps

0 logical volume(s) in volume group "dgpmodthkapps" now active

**5)** Activate the VG (vgchange -ay VG)

lpmodthk05:~ # vgchange -ay dgpmodthkapps

1 logical volume(s) in volume group "dgpmodthkapps" now active

**6)** Scan for LVs

lpmodthk05:~ # lvscan

ACTIVE '/dev/dgswap/lvswap01' [1.00 GiB] inherit

ACTIVE '/dev/dgpmodthkapps/lv01' [2.47 TiB] inherit

**7)** Activate the LVs (lvchange -ay LV)

lpmodthk05:~ # lvchange -ay /dev/dgpmodthkapps/lv01

**8)** A quick check shows all is now OK. **NOTE:** if this **fails**, then please **contact a technical expert**

lpmodthk05:~ # pvs

PV VG Fmt Attr PSize PFree

/dev/xvdb dgpmodthkapps lvm2 a- 506.32g 0

/dev/xvdc dgpmodthkapps lvm2 a- 506.32g 0

..

lpmodthk05:~ # vgs

VG #PV #LV #SN Attr VSize VFree

dgpmodthkapps 5 1 0 wz--n- 2.47t 0

dgswap 1 1 0 wz--n- 14.06g 13.06g

lpmodthk05:~ # lvs

LV VG Attr LSize Origin Snap% Move Log Copy% Convert

lv01 dgpmodthkapps -wi-ao 2.47t

lvswap01 dgswap -wi-ao 1.00g

**9)** Now you can go ahead and re-mount the volume.

vgdisplay command hangs

**Solution**

$ vgssan -v --- /sbin/vgscan ...?

xfs\_buf\_find: Block out of range

If xfs\_growfs fails with "Killed" and messages like

\_xfs\_buf\_find: Block out of range:

BUG: unable to handle kernel NULL pointer dereference "

Upgrade the kernel:

> On 29.11.2013 12:54, Poeschl Andreas, FG-940DG wrote:

> Der 3.0.93er Kernel hat aber ein großes Problem: XFS-Filesysteme lassen sich nicht erweitern.

> Dies trifft uns insbesondere in VMs hart. Dafür gibt es nun den 3.0.101er Kernel, der

> auch dieses Problem behebt ...

XFS NOT full but "No space left on device"

**PROBLEM:**after a filesystem extension the cusomer reports that the message "No space left on device" is received when trying to write to the filesystem

lpspluind3b:/global/cold-index # df -hP .

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/dgspluind3ci-lv01 30T 22T 8.5T 72% /global/cold-index

lpspluind3b:/global/cold-index # touch bla

touch: cannot touch `bla': No space left on device

**SOLUTION:** modify /etc/fstab: change options from default to inode64, and remount the filesystem

lpspluind3b:~ # egrep cold-index /etc/fstab

/dev/dgspluind3ci/lv01 /global/cold-index xfs inode64 0 0

lpspluind3b:~ # umount /global/cold-index

lpspluind3b:~ # mount /global/cold-index

lpspluind3b:~ # touch /global/cold-index/bla

lpspluind3b:~ # rm /global/cold-index/bla

lpspluind3b:~ #

**REASON:** see [No space left on device after xfs\_growfs](http://xfs.org/index.php/XFS_FAQ#Q:_Why_do_I_receive_No_space_left_on_device_after_xfs_growfs.3F)

vgdisplay shows LVM Header problem

The error pattern is as follows:

lp34xen05# vgdisplay -v dgstblnet

Using volume group(s) on command line

Finding volume group "dgstblnet"

Wiping cache of LVM-capable devices

Couldn't find device with uuid 'KtKVFt-0m0W-QLaU-HtbT-UmrN-kFNb-cfqCmz'.

Couldn't find all physical volumes for volume group dgstblnet.

PV Name /dev/md66

PV UUID KtKVFt-0m0W-QLaU-HtbT-UmrN-kFNb-cfqCmz

PV Status allocatable

Total PE / Free PE 28811 / 0

**Attention only with aproval of a senior**

The solution (on XEN farms where VM is running or at least one VG running/active). The UUID is taken form the config file you restore from, here /etc/lvm/backup/dgstblnet:

lp34xen05# pvcreate --restorefile /etc/lvm/backup/dgstblnet --uuid 'KtKVFt-0m0W-QLaU-HtbT-UmrN-kFNb-cfqCmz' /dev/md66

Couldn't find device with uuid 'KtKVFt-0m0W-QLaU-HtbT-UmrN-kFNb-cfqCmz'.

Failed to read existing physical volume '/dev/md66'

Physical volume "/dev/md66" successfully created

It is also possible to run that inside a VM (/etc/lvm/backup/dgXXX also exists there), but there the device is named /dev/xvdY - keep that in mind!  
  
Running the above command again should show no error:

lp34xen05# vgdisplay -v dgstblnet

Using volume group(s) on command line

Finding volume group "dgstblnet"

Wiping cache of LVM-capable devices

PV Name /dev/md66

PV UUID KtKVFt-0m0W-QLaU-HtbT-UmrN-kFNb-cfqCmz

PV Status allocatable

Total PE / Free PE 28811 / 0

Volume group vanished/destroyed

We'll see here how a volume group is restored again in case the LVM metadata has been destroyed.

Starting point

For this little tutorial I have created a vg named test\_vg, with a lv named test\_lv, mounted to /test with some files in it.

root@lt1001:/etc/lvm/backup #pvs

PV VG Fmt Attr PSize PFree

/dev/dm-2 test\_vg lvm2 a- 14.07g 0

root@lt1001:/etc/lvm/backup #export PS1='#'

# pvs

PV VG Fmt Attr PSize PFree

/dev/dm-2 test\_vg lvm2 a- 14.07g 0

# vgs

VG #PV #LV #SN Attr VSize VFree

test\_vg 1 1 0 wz--n- 14.07g 0

# lvs

LV VG Attr LSize Origin Snap% Move Log Copy% Convert

test\_lv test\_vg -wi-ao 14.07g

# df -hP /test; ll /test

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/test\_vg-test\_lv 15G 1000M 14G 7% /test

total 0

drwxr-xr-x 8 root root 83 May 31 14:55 var

Destroy metadata

As I do currently not have a destroyed vg, we'll go ahead and do it on our own, just the metadata which makes up the fist 1MB; if you go beyond that point, you also destroy the XFS header and then a restore is most likely quicker:

# dd if=/dev/zero of=/dev/dm-2 bs=1K count=1024

1024+0 records in

1024+0 records out

1048576 bytes (1.0 MB) copied, 0.0547759 s, 19.1 MB/s

# pvs

# vgs

No volume groups found

# lvs

No volume groups found

As now there isn't any metadata any more, all information about the pv, vg and lv is apparently gone. The next point is the one where you'll normally start from:

Get the PV back

In order to recover metadata we need the configuration stored in /etc/lvm/backup/"volumegroup" which is updated every time one changes the lv, vg or pv.  
  
So let's try the first step and get the physical volume back. We need the UUID the physical volume had been created with and write it back to the disk:

# grep -B1 id /etc/lvm/backup/test\_vg

test\_vg {

id = "le26B4-4t20-3k5w-wVoq-L693-wEgV-vuwltn"

--

pv0 {

id = "2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk"

--

test\_lv {

id = "MR2dbo-oucF-dbUK-dpDR-Zp3s-kIWb-cUGmEu"

Here we need the id for pv0 which we write back to the disk:

# pvcreate --restorefile /etc/lvm/backup/test\_vg -u "2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk" /dev/dm-2

Couldn't find device with uuid 2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk.

Can't open /dev/dm-2 exclusively. Mounted filesystem?

OK, interesting, I'm pretty sure that there is nothing mounted, but let's see and try it forcibly again:

# mount |grep test

# pvcreate --restorefile /etc/lvm/backup/test\_vg -u "2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk" /dev/dm-2 -ff

Couldn't find device with uuid 2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk.

Can't open /dev/dm-2 exclusively. Mounted filesystem?

Still no success (this is on physical hosts). The problem is, that device mapper has still its hand on the VG and we have to remove that first:

# dmsetup ls

[...]

360060e8016535e000001535e0000002a (253, 17)

360060e8016535c000001535c0000002a (253, 6)

test\_vg-test\_lv (253, 31) <<<<<

360060e8016535e000001535e00000029 (253, 7)

360060e8016535c000001535c00000029 (253, 14)

[...]

# dmsetup remove test\_vg-test\_lv

# dmsetup ls

=> Now device should be gone!

Once done, we should be able to recover:

# pvcreate --restorefile /etc/lvm/backup/test\_vg -u "2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk" /dev/dm-2 -ff

Couldn't find device with uuid 2iQKIt-kMKH-pEi4-fph2-TE7d-n6fl-TCCGLk.

Physical volume "/dev/dm-2" successfully created

# pvs

PV VG Fmt Attr PSize PFree

/dev/dm-2 lvm2 a- 14.07g 14.07g

OK, so now we have or physical volume back.

Volume group and logical volume

Once we have back the physical volume, let's see if we get the rest by restoring the volume group configuration:

# vgcfgrestore -f /etc/lvm/backup/test\_vg test\_vg

Restored volume group test\_vg

# vgs

VG #PV #LV #SN Attr VSize VFree

test\_vg 1 1 0 wz--n- 14.07g 0

# lvs

LV VG Attr LSize Origin Snap% Move Log Copy% Convert

test\_lv test\_vg -wi--- 14.07g

As you can see, the logical volume is not active so far which means that we can not mount it, so let's do that first:

# vgchange -ay test\_vg

1 logical volume(s) in volume group "test\_vg" now active

# lvs

LV VG Attr LSize Origin Snap% Move Log Copy% Convert

test\_lv test\_vg -wi-a- 14.07g

Now we should be able to mount the file system again.

Bring back file system

As we had some trouble with the volume, it is not unlikely that we do have trouble with the file system, so before mounting, check it:

# xfs\_check -s /dev/mapper/test\_vg-test\_lv

That looks good - in case it does not, try a **xfs\_repair** and then mount it again:

# mount -t xfs /dev/mapper/test\_vg-test\_lv /test

And we're done :-)

Online LUN replacement using LVM Mirror

Sometimes it's required to replace LUNs without downtimes. This howto describes how to do that.  
  
Work Plan:

* add disks see at xen lvm grow
* use one or more luns to mirror existing luns
* detach pvs/ luns & san return

Pro / Cons:  
+ mirroring speed can be controlled  
+ mirroring can be halted or stopped  
+ progress can be monitored  
- requires at least LVM version > v2.02.54 for stablity (according an EMC document)

>> Sample from lt34xendb002, lt34db0008vm - TAS000000380190 / TAS000000412164

# dmesg

[...]

[612704.164007] blkfront: xvdg: flush diskcache: enabled

[612704.167421] xvdg: unknown partition table

[612801.317234] blkfront: xvdh: flush diskcache: enabled

[612801.325368] xvdh: unknown partition table

[613021.129910] blkfront: xvdi: flush diskcache: enabled

[613021.132883] xvdi: unknown partition table

[613203.065475] blkfront: xvdj: flush diskcache: enabled

[613203.068925] xvdj: unknown partition table

[613386.295576] blkfront: xvdk: flush diskcache: enabled

[613386.300956] xvdk: unknown partition table

[613536.703682] BIOS EDD facility v0.16 2004-Jun-25, 0 devices found

[613536.703688] EDD information not available.

root@lt34dbhk0006vm:~# pvcreate /dev/xvdg /dev/xvdh /dev/xvdi /dev/xvdj /dev/xvdk

Physical volume "/dev/xvdg" successfully created

Physical volume "/dev/xvdh" successfully created

Physical volume "/dev/xvdi" successfully created

Physical volume "/dev/xvdj" successfully created

Physical volume "/dev/xvdk" successfully created

root@lt34dbhk0006vm:~# pvs

PV VG Fmt Attr PSize PFree

/dev/xvdb dgtl34dbhk0006vmswap lvm2 a- 14.06g 0

/dev/xvdc dgtl34dbhk0006vmdb lvm2 a- 56.24g 0

/dev/xvdd dgtl34dbhk0006vmdb lvm2 a- 56.24g 0

/dev/xvde dgtl34dbhk0006vmdb lvm2 a- 56.24g 18.71g

/dev/xvdf dgtl34dbhk0006vmbkup lvm2 a- 56.24g 0

/dev/xvdg lvm2 a- 506.32g 506.32g

/dev/xvdh lvm2 a- 506.32g 506.32g

/dev/xvdi lvm2 a- 506.32g 506.32g

/dev/xvdj lvm2 a- 506.32g 506.32g

/dev/xvdk lvm2 a- 506.32g 506.32g

>> Extend the VG:

root@lt34dbhk0006vm:~# vgextend dgtl34dbhk0006vmdb /dev/xvdg

Volume group "dgtl34dbhk0006vmdb" successfully extended

>> View LVs:

root@lt34dbhk0006vm:~# lvs -a -o+devices

LV VG Attr LSize Origin Snap% Move Log Copy% Convert Devices

lvbkup dgtl34dbhk0006vmbkup -wi-ao 56.24g /dev/xvdf(0)

lvdb dgtl34dbhk0006vmdb -wi-ao 150.00g /dev/xvdc(0)

lvdb dgtl34dbhk0006vmdb -wi-ao 150.00g /dev/xvdd(0)

lvdb dgtl34dbhk0006vmdb -wi-ao 150.00g /dev/xvde(0)

lvswap dgtl34dbhk0006vmswap -wi-ao 14.06g /dev/xvdb(0)

>> Mirror LVs:

root@lt34dbhk0006vm:~# lvconvert -b -m1 --corelog dgtl34dbhk0006vmdb/lvdb

Logical volume lvdb converted.

>> Monitor LVs:

root@lt34dbhk0006vm:~# lvs -a -o+devices

LV VG Attr LSize Origin Snap% Move Log Copy% Convert Devices

lvbkup dgtl34dbhk0006vmbkup -wi-ao 56.24g /dev/xvdf(0)

lvdb dgtl34dbhk0006vmdb mwi-ao 150.00g 1.07 lvdb\_mimage\_0(0),lvdb\_mimage\_1(0 )

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdc(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdd(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvde(0)

[lvdb\_mimage\_1] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdg(0)

lvswap dgtl34dbhk0006vmswap -wi-ao 14.06g

/dev/xvdb(0)

root@lt34dbhk0006vm:~# lvs -a -o+devices

LV VG Attr LSize Origin Snap% Move Log Copy% Convert Devices

lvbkup dgtl34dbhk0006vmbkup -wi-ao 56.24g /dev/xvdf(0)

lvdb dgtl34dbhk0006vmdb mwi-ao 150.00g 5.58 lvdb\_mimage\_0(0),lvdb\_mimage\_1(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdc(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdd(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvde(0)

[lvdb\_mimage\_1] dgtl34dbhk0006vmdb Iwi-ao 150.00g /dev/xvdg(0)

lvswap dgtl34dbhk0006vmswap -wi-ao 14.06g /dev/xvdb(0)

>> When the mirror is at 100%, we sync amd detach old LUNs/storage:

root@lt34dbhk0006vm:~# lvs -a -o+devices

LV VG Attr LSize Origin Snap% Move Log Copy% Convert Devices

lvbkup dgtl34dbhk0006vmbkup -wi-ao 56.24g /dev/xvdf(0)

lvdb dgtl34dbhk0006vmdb mwi-ao 150.00g 100.00

lvdb\_mimage\_0(0),lvdb\_mimage\_1(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb iwi-ao 150.00g /dev/xvdc(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb iwi-ao 150.00g /dev/xvdd(0)

[lvdb\_mimage\_0] dgtl34dbhk0006vmdb iwi-ao 150.00g /dev/xvde(0)

[lvdb\_mimage\_1] dgtl34dbhk0006vmdb iwi-ao 150.00g /dev/xvdg(0)

lvswap dgtl34dbhk0006vmswap -wi-ao 14.06g /dev/xvdb(0)

root@lt34dbhk0006vm:~# sync

>> Remove old devices, break the mirror:

root@lt34dbhk0006vm:~# lvconvert -m0 dgtl34dbhk0006vmdb/lvdb /dev/xvdc /dev/xvdd /dev/xvde

Logical volume lvdb converted.

>> View LVs, PVs:

root@lt34dbhk0006vm:~# lvs -a -o+devices

LV VG Attr LSize Origin Snap% Move Log Copy% Convert Devices

lvbkup dgtl34dbhk0006vmbkup -wi-ao 56.24g /dev/xvdf(0)

lvdb dgtl34dbhk0006vmdb -wi-ao 150.00g /dev/xvdg(0)

lvswap dgtl34dbhk0006vmswap -wi-ao 14.06g /dev/xvdb(0)

root@lt34dbhk0006vm:~# pvs

PV VG Fmt Attr PSize PFree

/dev/xvdb dgtl34dbhk0006vmswap lvm2 a- 14.06g 0

/dev/xvdc dgtl34dbhk0006vmdb lvm2 a- 56.24g 56.24g

/dev/xvdd dgtl34dbhk0006vmdb lvm2 a- 56.24g 56.24g

/dev/xvde dgtl34dbhk0006vmdb lvm2 a- 56.24g 56.24g

/dev/xvdf dgtl34dbhk0006vmbkup lvm2 a- 56.24g 0

/dev/xvdg dgtl34dbhk0006vmdb lvm2 a- 506.32g 356.32g

/dev/xvdh lvm2 a- 506.32g 506.32g

/dev/xvdi lvm2 a- 506.32g 506.32g

/dev/xvdj lvm2 a- 506.32g 506.32g

/dev/xvdk lvm2 a- 506.32g 506.32g

>> Remove old PVs from VG and wipe them:

root@lt34dbhk0006vm:~# vgreduce dgtl34dbhk0006vmdb /dev/xvdc /dev/xvdd /dev/xvde

Removed "/dev/xvdc" from volume group "dgtl34dbhk0006vmdb"

Removed "/dev/xvdd" from volume group "dgtl34dbhk0006vmdb"

Removed "/dev/xvde" from volume group "dgtl34dbhk0006vmdb"

root@lt34dbhk0006vm:~# pvremove /dev/xvdc

Labels on physical volume "/dev/xvdc" successfully wiped

root@lt34dbhk0006vm:~# pvremove /dev/xvdd

Labels on physical volume "/dev/xvdd" successfully wiped

root@lt34dbhk0006vm:~# pvremove /dev/xvde

Labels on physical volume "/dev/xvde" successfully wiped

>> On the farmservers, remove and cleanup the old PVs:

root@lt34xendb001:~# vm info lt34dbhk0006vm | egrep '(xvdc|xvdd|xvde)'

Block device xvdc : /dev/md/lt34dbhk0006vm\_2

Block device xvdd : /dev/md/lt34dbhk0006vm\_3

Block device xvde : /dev/md/lt34dbhk0006vm\_4

root@lt34xendb001:~# mdadm --detail /dev/md/lt34dbhk0006vm\_2

/dev/md/lt34dbhk0006vm\_2:

Version : 1.1

Creation Time : Thu Jul 10 13:44:23 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Thu Oct 2 20:06:23 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34dbhk0006vm\_2

UUID : b4c0a673:a05cc538:dc0decf1:a4fa60aa

Events : 2

Number Major Minor RaidDevice State

0 253 126 0 active sync /dev/dm-126

1 253 137 1 active sync /dev/dm-137

root@lt34xendb001:~# mdadm --detail /dev/md/lt34dbhk0006vm\_3

/dev/md/lt34dbhk0006vm\_3:

Version : 1.1

Creation Time : Thu Jul 10 13:45:23 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Thu Oct 2 20:06:24 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34dbhk0006vm\_3

UUID : f5e182ed:6c1d30f4:dc7e7632:a68fbd90

Events : 4

Number Major Minor RaidDevice State

0 253 123 0 active sync /dev/dm-123

1 253 136 1 active sync /dev/dm-136

root@lt34xendb001:~# mdadm --detail /dev/md/lt34dbhk0006vm\_4

/dev/md/lt34dbhk0006vm\_4:

Version : 1.1

Creation Time : Thu Jul 10 13:47:07 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Thu Oct 2 20:06:26 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34dbhk0006vm\_4

UUID : 3ca238a2:e971c971:c83b7a71:5e1e450d

Events : 2

Number Major Minor RaidDevice State

0 253 42 0 active sync /dev/dm-42

1 253 135 1 active sync /dev/dm-135

>> View LUNs for SAN return (must be done before DMs are removed)

root@lt34xendb001:~# san\_shortinfo | egrep -i '(dm-42|dm-135|dm-123|dm-136|dm-126|dm-137)'

dm-135 | sdip 01:67 3A 97255 57622 | sdjn 01:67 4A 97255 57622 |

dm-137 | sdio 01:66 3A 97255 57622 | sdjm 01:66 4A 97255 57622 |

dm-136 | sdim 01:64 3A 97255 57622 | sdjk 01:64 4A 97255 57622 |

dm-42 | sdid 01:67 3A 97231 57622 | sdjb 01:67 4A 97231 57622 |

dm-126 | sdec 01:66 3A 97231 57622 | sdja 01:66 4A 97231 57622 |

dm-123 | sddd 01:64 3A 97231 57622 | sdiy 01:64 4A 97231 57622 |

root@lt34xendb001:~# vm rmdisk lt34dbhk0006vm lt34dbhk0006vm\_2

20:25:11 Checking farm status before running command 'rmdisk'

20:25:11 Establishing libvirt connection to lt34xendb001 OK

20:25:14 Establishing libvirt connection to lt34xendb002 OK

20:25:17 Verifying free config space on lt34xendb001 lt34xendb002 OK

20:25:23 Establishing libvirt connection to lt34xendb001 OK

20:25:26 Detaching md device /dev/md/lt34dbhk0006vm\_2 from VM lt34dbhk0006vm OK

20:25:30 Establishing libvirt connection to lt34xendb002 OK

20:25:32 Updating definition of lt34dbhk0006vm on lt34xendb002 OK

20:25:32 Stopping md device /dev/md/lt34dbhk0006vm\_2 OK

20:25:32 Removing md device /dev/md/lt34dbhk0006vm\_2 from mdadm.conf OK

20:25:34 /dev/md/lt34dbhk0006vm\_2 removed from lt34dbhk0006vm

root@lt34xendb001:~# vm rmdisk lt34dbhk0006vm lt34dbhk0006vm\_3

20:25:45 Checking farm status before running command 'rmdisk'

20:25:45 Establishing libvirt connection to lt34xendb001 OK

20:25:47 Establishing libvirt connection to lt34xendb002 OK

20:25:50 Verifying free config space on lt34xendb001 lt34xendb002 OK

20:25:56 Establishing libvirt connection to lt34xendb001 OK

20:25:59 Detaching md device /dev/md/lt34dbhk0006vm\_3 from VM lt34dbhk0006vm OK

20:26:04 Establishing libvirt connection to lt34xendb002 OK

20:26:05 Updating definition of lt34dbhk0006vm on lt34xendb002 OK

20:26:05 Stopping md device /dev/md/lt34dbhk0006vm\_3 OK

20:26:06 Removing md device /dev/md/lt34dbhk0006vm\_3 from mdadm.conf OK

20:26:07 /dev/md/lt34dbhk0006vm\_3 removed from lt34dbhk0006vm

root@lt34xendb001:~# vm rmdisk lt34dbhk0006vm lt34dbhk0006vm\_4

20:26:14 Checking farm status before running command 'rmdisk'

20:26:14 Establishing libvirt connection to lt34xendb001 OK

20:26:16 Establishing libvirt connection to lt34xendb002 OK

20:26:19 Verifying free config space on lt34xendb001 lt34xendb002 OK

20:26:25 Establishing libvirt connection to lt34xendb001 OK

20:26:28 Detaching md device /dev/md/lt34dbhk0006vm\_4 from VM lt34dbhk0006vm OK

20:26:32 Establishing libvirt connection to lt34xendb002 OK

20:26:34 Updating definition of lt34dbhk0006vm on lt34xendb002 OK

20:26:34 Stopping md device /dev/md/lt34dbhk0006vm\_4 OK

20:26:34 Removing md device /dev/md/lt34dbhk0006vm\_4 from mdadm.conf OK

20:26:36 /dev/md/lt34dbhk0006vm\_4 removed from lt34dbhk0006vm

Done. Now the replaced LUNs can be returned to the SAN (see TAS000000412164).

SAN return I

**NOTE:** The san\_return command does **NOT** cover this. We have to manually generate a clean pvinfo output to return the LUNs in STORM. Make sure you're using the correct SAN boxes and LUNs.

root@lt34xendb001:~# vm info lt34db0008vm

Name : lt34db0008vm

Domain ID : NR

Status : Running on lt34xendb002

VM Type : PV

Mounts : None

Storage Type : SAN+MD

Image : N/A

Last Backup (UTC) : 2014-10-27 22:03:43

Cur. VCPUs : 6

Max. VCPUs : 12

Cur. Mem [MB] : 10240

Max. Mem [MB] : 20480

XML - Cur. VCPUs : 6

XML - Max. VCPUs : 12

XML - Cur. Mem [MB] : 10240

XML - Max. Mem [MB] : 20480

XML - Target State : 1

XML - Priority : 1

Network device 0 : 00:16:3e:19:c3:da (br\_172025195000)

Network device 1 : 00:16:3e:19:f8:5d (br\_172025248000)

Block device xvda : /dev/md/lt34db0008vm

Block device xvdb : /dev/md/lt34db0008vm\_1

Block device xvdc : /dev/md/lt34db0008vm\_2

Block device xvdd : /dev/md/lt34db0008vm\_3

Block device xvde : /dev/md/lt34db0008vm\_4

Block device xvdf : /dev/md/lt34db0008vm\_5

Block device xvdg : /dev/md/lt34db0008vm\_6

Block device xvdh : /dev/md/lt34db0008vm\_7

Block device xvdi : /dev/md/lt34db0008vm\_8

Block device xvdj : /dev/md/lt34db0008vm\_9

Block device xvdk : /dev/md/lt34db0008vm\_10

Block device xvdl : /dev/md/lt34db0008vm\_11

Block device xvdm : /dev/md/lt34db0008vm\_12

Block device xvdn : /dev/md/lt34db0008vm\_13

Block device xvdo : /dev/md/lt34db0008vm\_14

Block device xvdp : /dev/md/lt34db0008vm\_15

Block device xvdq : /dev/md/lt34db0008vm\_16

Now return the replaced LUNs, in our case xvdc|xvdd|xvde|xvdg|xvdh:

root@lt34xendb001:~# vm info lt34db0008vm | egrep '(xvdc|xvdd|xvde|xvdg|xvdh)'

Block device xvdc : /dev/md/lt34db0008vm\_2

Block device xvdd : /dev/md/lt34db0008vm\_3

Block device xvde : /dev/md/lt34db0008vm\_4

Block device xvdg : /dev/md/lt34db0008vm\_6

Block device xvdh : /dev/md/lt34db0008vm\_7

root@lt34xendb002:~# mdadm --detail /dev/md/lt34db0008vm\_2

/dev/md/lt34db0008vm\_2:

Version : 1.1

Creation Time : Fri Jul 11 11:46:17 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Oct 28 14:19:40 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34db0008vm\_2

UUID : c2049538:a3a8b3e2:aabdda14:338e95a3

Events : 4

Number Major Minor RaidDevice State

0 253 127 0 active sync /dev/dm-127

1 253 139 1 active sync /dev/dm-139

root@lt34xendb002:~# mdadm --detail /dev/md/lt34db0008vm\_3

/dev/md/lt34db0008vm\_3:

Version : 1.1

Creation Time : Fri Jul 11 11:49:24 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Oct 28 14:19:43 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34db0008vm\_3

UUID : 5126160d:be095b17:a71c0fc1:96496802

Events : 7

Number Major Minor RaidDevice State

0 253 130 0 active sync /dev/dm-130

1 253 144 1 active sync /dev/dm-144

root@lt34xendb002:~# mdadm --detail /dev/md/lt34db0008vm\_4

/dev/md/lt34db0008vm\_4:

Version : 1.1

Creation Time : Fri Jul 11 12:07:59 2014

Raid Level : raid1

Array Size : 58972604 (56.24 GiB 60.39 GB)

Used Dev Size : 58972604 (56.24 GiB 60.39 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Oct 28 14:19:50 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34db0008vm\_4

UUID : 8b6a2828:f773b377:0a97b870:990238fd

Events : 2

Number Major Minor RaidDevice State

0 253 132 0 active sync /dev/dm-132

1 253 141 1 active sync /dev/dm-141

root@lt34xendb002:~# mdadm --detail /dev/md/lt34db0008vm\_6

/dev/md/lt34db0008vm\_6:

Version : 1.1

Creation Time : Thu Aug 21 15:26:26 2014

Raid Level : raid1

Array Size : 235890556 (224.96 GiB 241.55 GB)

Used Dev Size : 235890556 (224.96 GiB 241.55 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Oct 28 14:19:54 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34db0008vm\_6

UUID : 939c7a34:2f65d126:72cabd2b:d47c23e1

Events : 2

Number Major Minor RaidDevice State

0 253 151 0 active sync /dev/dm-151

1 253 152 1 active sync /dev/dm-152

root@lt34xendb002:~# mdadm --detail /dev/md/lt34db0008vm\_7

/dev/md/lt34db0008vm\_7:

Version : 1.1

Creation Time : Thu Aug 21 15:28:44 2014

Raid Level : raid1

Array Size : 235890556 (224.96 GiB 241.55 GB)

Used Dev Size : 235890556 (224.96 GiB 241.55 GB)

Raid Devices : 2

Total Devices : 2

Persistence : Superblock is persistent

Intent Bitmap : Internal

Update Time : Tue Oct 28 14:19:56 2014

State : active

Active Devices : 2

Working Devices : 2

Failed Devices : 0

Spare Devices : 0

Name : SAN:lt34db0008vm\_7

UUID : 4c6e6b10:3932b0f5:82fd53eb:fb8c4ded

Events : 3

Number Major Minor RaidDevice State

0 253 154 0 active sync /dev/dm-154

1 253 150 1 active sync /dev/dm-150

Now that we know the configuration of eatch DM, we can remove them:

root@lt34xendb002:~# san\_shortinfo | egrep -iw '(dm-151|dm-152|dm-136|dm-142|dm-132|dm-141|dm-130|dm-144|dm127|dm139|dm-154|dm-150)'

dm-136 | sdij 01:6D 3A 97231 57622 | sdjh 01:6D 4A 97231 57622 |

dm-132 | sdii 01:6C 3A 97231 57622 | sdjg 01:6C 4A 97231 57622 |

dm-154 | sdkh 00:B1 3B 97231 230490 | sdkn 00:B1 4B 97231 230490 |

dm-130 | sdih 01:6B 3A 97231 57622 | sdjf 01:6B 4A 97231 57622 |

dm-151 | sdkg 00:B0 3B 97231 230490 | sdkm 00:B0 4B 97231 230490 |

dm-150 | sdke 00:B1 3B 97255 230490 | sdkk 00:B1 4B 97255 230490 |

dm-142 | sdiv 01:6D 3A 97255 57622 | sdjt 01:6D 4A 97255 57622 |

dm-152 | sdkd 00:B0 3B 97255 230490 | sdkj 00:B0 4B 97255 230490 |

dm-141 | sdiu 01:6C 3A 97255 57622 | sdjs 01:6C 4A 97255 57622 |

dm-144 | sdit 01:6B 3A 97255 57622 | sdjr 01:6B 4A 97255 57622 |

root@lt34xendb002:~# san\_shortinfo | egrep -iw '(01:6D|01:6C|00:B1|01:6B|00:B0)'

dm-136 | sdij 01:6D 3A 97231 57622 | sdjh 01:6D 4A 97231 57622 |

dm-132 | sdii 01:6C 3A 97231 57622 | sdjg 01:6C 4A 97231 57622 |

dm-154 | sdkh 00:B1 3B 97231 230490 | sdkn 00:B1 4B 97231 230490 |

dm-130 | sdih 01:6B 3A 97231 57622 | sdjf 01:6B 4A 97231 57622 |

dm-151 | sdkg 00:B0 3B 97231 230490 | sdkm 00:B0 4B 97231 230490 |

dm-150 | sdke 00:B1 3B 97255 230490 | sdkk 00:B1 4B 97255 230490 |

dm-142 | sdiv 01:6D 3A 97255 57622 | sdjt 01:6D 4A 97255 57622 |

dm-152 | sdkd 00:B0 3B 97255 230490 | sdkj 00:B0 4B 97255 230490 |

dm-141 | sdiu 01:6C 3A 97255 57622 | sdjs 01:6C 4A 97255 57622 |

dm-144 | sdit 01:6B 3A 97255 57622 | sdjr 01:6B 4A 97255 57622 |

Remove the disks from the VM:

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_2

15:02:46 Checking farm status before running command 'rmdisk'

15:02:46 Establishing libvirt connection to lt34xendb001 OK

15:02:49 Establishing libvirt connection to lt34xendb002 OK

15:02:51 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:02:58 Establishing libvirt connection to lt34xendb002 OK

15:02:59 Detaching md device /dev/md/lt34db0008vm\_2 from VM lt34db0008vm OK

15:03:06 Establishing libvirt connection to lt34xendb001 OK

15:03:13 Updating definition of lt34db0008vm on lt34xendb001 OK

15:03:14 Stopping md device /dev/md/lt34db0008vm\_2 OK

15:03:16 Removing md device /dev/md/lt34db0008vm\_2 from mdadm.conf OK

15:03:24 /dev/md/lt34db0008vm\_2 removed from lt34db0008vm

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_3

15:03:58 Checking farm status before running command 'rmdisk'

15:03:58 Establishing libvirt connection to lt34xendb001 OK

15:04:01 Establishing libvirt connection to lt34xendb002 OK

15:04:04 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:04:10 Establishing libvirt connection to lt34xendb002 OK

15:04:12 Detaching md device /dev/md/lt34db0008vm\_3 from VM lt34db0008vm OK

15:04:18 Establishing libvirt connection to lt34xendb001 OK

15:04:21 Updating definition of lt34db0008vm on lt34xendb001 OK

15:04:21 Stopping md device /dev/md/lt34db0008vm\_3 OK

15:04:22 Removing md device /dev/md/lt34db0008vm\_3 from mdadm.conf OK

15:04:41 /dev/md/lt34db0008vm\_3 removed from lt34db0008vm

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_4

15:04:53 Checking farm status before running command 'rmdisk'

15:04:53 Establishing libvirt connection to lt34xendb001 OK

15:04:56 Establishing libvirt connection to lt34xendb002 OK

15:04:58 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:05:04 Establishing libvirt connection to lt34xendb002 OK

15:05:06 Detaching md device /dev/md/lt34db0008vm\_4 from VM lt34db0008vm OK

15:05:12 Establishing libvirt connection to lt34xendb001 OK

15:05:15 Updating definition of lt34db0008vm on lt34xendb001 OK

15:05:15 Stopping md device /dev/md/lt34db0008vm\_4 OK

15:05:16 Removing md device /dev/md/lt34db0008vm\_4 from mdadm.conf OK

15:05:18 /dev/md/lt34db0008vm\_4 removed from lt34db0008vm

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_5

15:05:23 Checking farm status before running command 'rmdisk'

15:05:23 Establishing libvirt connection to lt34xendb001 OK

15:05:26 Establishing libvirt connection to lt34xendb002 OK

15:05:29 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:05:35 Establishing libvirt connection to lt34xendb002 OK

15:05:37 Detaching md device /dev/md/lt34db0008vm\_5 from VM lt34db0008vm OK

15:07:20 Establishing libvirt connection to lt34xendb001 OK

15:07:24 Updating definition of lt34db0008vm on lt34xendb001 OK

15:07:24 Stopping md device /dev/md/lt34db0008vm\_5 FAILED

15:07:24 ERROR: Could not stop md device /dev/md/lt34db0008vm\_5, aborting

15:07:24 ERROR: Command rmdisk did not complete successfully

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_6

15:08:02 Checking farm status before running command 'rmdisk'

15:08:02 Establishing libvirt connection to lt34xendb001 OK

15:08:05 Establishing libvirt connection to lt34xendb002 OK

15:08:08 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:08:14 Establishing libvirt connection to lt34xendb002 OK

15:08:16 Detaching md device /dev/md/lt34db0008vm\_6 from VM lt34db0008vm OK

15:08:33 Establishing libvirt connection to lt34xendb001 OK

15:08:36 Updating definition of lt34db0008vm on lt34xendb001 OK

15:08:37 Stopping md device /dev/md/lt34db0008vm\_6 OK

15:08:38 Removing md device /dev/md/lt34db0008vm\_6 from mdadm.conf OK

15:08:39 /dev/md/lt34db0008vm\_6 removed from lt34db0008vm

root@lt34xendb002:~# vm rmdisk lt34db0008vm lt34db0008vm\_7

15:08:43 Checking farm status before running command 'rmdisk'

15:08:43 Establishing libvirt connection to lt34xendb001 OK

15:08:47 Establishing libvirt connection to lt34xendb002 OK

15:08:49 Verifying free config space on lt34xendb001 lt34xendb002 OK

15:08:56 Establishing libvirt connection to lt34xendb002 OK

15:08:57 Detaching md device /dev/md/lt34db0008vm\_7 from VM lt34db0008vm OK

15:09:03 Establishing libvirt connection to lt34xendb001 OK

15:09:06 Updating definition of lt34db0008vm on lt34xendb001 OK

15:09:06 Stopping md device /dev/md/lt34db0008vm\_7 OK

15:09:07 Removing md device /dev/md/lt34db0008vm\_7 from mdadm.conf OK

15:09:09 /dev/md/lt34db0008vm\_7 removed from lt34db0008vm

As we do know the LUN IDs, verify the DM devices and remove them:

root@lt34xendb001:~# pvinfo -i | egrep -iw '(00:b0|00:b1|01:6a|01:6b|01:6c)' | egrep '(97231|97255)' | grep -i "dm-"

/dev/dm-155 00 00 00 CL4-F-1 00:b1 OPEN-V 00097255

/dev/dm-153 00 00 00 CL4-F-1 00:b1 OPEN-V 00097231

/dev/dm-152 00 00 00 CL4-F-1 00:b0 OPEN-V 00097255

/dev/dm-150 00 00 00 CL4-F-1 00:b0 OPEN-V 00097231

/dev/dm-142 00 00 00 CL4-E-1 01:6c OPEN-V 00097255

/dev/dm-141 00 00 00 CL4-E-1 01:6b OPEN-V 00097255

/dev/dm-132 00 00 00 CL4-E-1 01:6c OPEN-V 00097231

/dev/dm-128 00 00 00 CL4-E-1 01:6b OPEN-V 00097231

/dev/dm-140 00 00 00 CL4-E-1 01:6a OPEN-V 00097255

/dev/dm-129 00 00 00 CL3-E-1 01:6a OPEN-V 00097231

Remove the DMs:

root@lt34xendb001:~# dmsetup info /dev/dm-155

Name: 360060e80167be70000017be7000000b1

State: ACTIVE

Read Ahead: 1024

Tables present: LIVE

Open count: 0

Event number: 0

Major, minor: 253, 155

Number of targets: 1

UUID: mpath-360060e80167be70000017be7000000b1

Note: "dmsetup" works on the WWN or /dev/dm-??? block device file:

root@lt34xendb001:~# dmsetup remove 360060e80167be70000017be7000000b1

root@lt34xendb001:~# dmsetup info /dev/dm-155

Device /dev/dm-155 not found

Command failed

root@lt34xendb001:~# dmsetup info /dev/dm-153

Name: 360060e80167bcf0000017bcf000000b1

State: ACTIVE

Read Ahead: 1024

Tables present: LIVE

Open count: 0

Event number: 0

Major, minor: 253, 153

Number of targets: 1

UUID: mpath-360060e80167bcf0000017bcf000000b1

root@lt34xendb001:~# dmsetup info /dev/dm-153

Device /dev/dm-153 not found

Command failed

As the 2nd command fails, we know that it's really removed from the system. Remove the remaining devices too:

root@lt34xendb001:~# dmsetup remove /dev/dm-152

root@lt34xendb001:~# dmsetup remove /dev/dm-150

root@lt34xendb001:~# dmsetup remove /dev/dm-142

root@lt34xendb001:~# dmsetup remove /dev/dm-141

root@lt34xendb001:~# dmsetup remove /dev/dm-132

root@lt34xendb001:~# dmsetup remove /dev/dm-128

root@lt34xendb001:~# dmsetup remove /dev/dm-140

root@lt34xendb001:~# dmsetup remove /dev/dm-129

SAN return II

**NOTE:** The san\_return command does **NOT** cover this. We have to manually generate a clean pvinfo output to return the LUNs in STORM. Make sure you're using the correct SAN boxes and LUNs.

root@lt34xendb001:~# pvinfo -i | egrep -w -i '(00:b0|00:b1|01:6a|01:6b|01:6c)' | egrep '(97231|97255)'

No supported HBA found!

Unable to get information.

Possible reasons:

HBAAPI not registered / loaded

driver with different capabilities

a.s.o

/dev/sdkn 00 00 00 CL4-F-1 00:b1 OPEN-V 00097255

/dev/sdkm 00 00 00 CL4-F-1 00:b0 OPEN-V 00097255

/dev/sdkj 00 00 00 CL4-F-1 00:b0 OPEN-V 00097231

/dev/sdkk 00 00 00 CL4-F-1 00:b1 OPEN-V 00097231

/dev/sdke 00 00 00 CL4-E-1 01:6b OPEN-V 00097255

/dev/sdkg 00 00 00 CL4-E-1 01:6c OPEN-V 00097255

/dev/sdjt 00 00 00 CL4-E-1 01:6c OPEN-V 00097231

/dev/sdjs 00 00 00 CL4-E-1 01:6b OPEN-V 00097231

/dev/sdjr 00 00 00 CL3-F-1 00:b1 OPEN-V 00097255

/dev/sdjg 00 00 00 CL3-F-1 00:b1 OPEN-V 00097231

/dev/sdjh 00 00 00 CL3-F-1 00:b0 OPEN-V 00097255

/dev/sdjf 00 00 00 CL3-F-1 00:b0 OPEN-V 00097231

/dev/sdiu 00 00 00 CL3-E-1 01:6c OPEN-V 00097255

/dev/sdit 00 00 00 CL3-E-1 01:6b OPEN-V 00097255

/dev/sdii 00 00 00 CL3-E-1 01:6c OPEN-V 00097231

/dev/sdih 00 00 00 CL3-E-1 01:6b OPEN-V 00097231

/dev/sdjq 00 00 00 CL4-E-1 01:6a OPEN-V 00097255

/dev/sdje 00 00 00 CL4-E-1 01:6a OPEN-V 00097231

/dev/sdis 00 00 00 CL3-E-1 01:6a OPEN-V 00097255

/dev/sdig 00 00 00 CL3-E-1 01:6a OPEN-V 00097231

Now clean up the devices and return them in STORM:

root@lt34xendb002:~# for i in sdkn sdkm sdkk sdkj sdkg sdke sdjt sdjs sdjr sdjh sdjg sdjf sdiu sdit sdii sdih sdjq sdje sdis sdig; do echo 1 > /sys/block/$i/device/delete; done

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by Kemmerer Walter, (Walter.Kemmerer@partner.bmw.de) , [Kujau Christian, (Christian.CK.Kujau@partner.bmw.de)](https://bsswiki.muc/tiki-user_information.php?userId=55) , Venzke Horst, (Horst.Venzke@partner.bmw.de) , [qxc0474](https://bsswiki.muc/tiki-user_information.php?userId=-1) and Urban Richard, (Richard.Urban@partner.bmw.de) .  
Page last modified on Wednesday 16 of October, 2019 16:09:43 CEST. (Version 25)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified by [Sanjiv Soraganvi (ext.)](https://atc.bmwgroup.net/confluence/display/~qxw9324) on [22 Dec 2021](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=520964216&selectedPageVersions=4&selectedPageVersions=5)

#### **Disk addition threshold limit - Number of Luns per Xenfarm and Per Xen VMs** -Farm server:  Total number of LUNs - 1000  (#san\_info -ls  |grep -i "Disks seen"|sort -u) -Virtual Server : Total number of Luns assigned (#vmc info <vmname> or san\_shortinfo |grep -i <vmname>, other steps)                               Minumum Limit = 25 Luns                               Maximum Limit = 30 Luns                 If the Luns on Farm  is equal or beyond limit then no more Luns addition to farm is possible. If its needed then disk consolidation is required.                If the Luns on VMs is min limit  you may add more Luns till max limit but its a trigger to initiate the disk consolidation first and then proceed addition/extension.

| Grow XEN Guest (VM) FileSystem |
| --- |
| Scope |

Describes ALL of the steps involved in extending an LVM Volume on a XEN Virtual Machine (VM)  
**NOTE!! First Check if you need to Order Storage or not: See**[**Grow Existing VG/LVs**](https://atc.bmwgroup.net/confluence/pages/viewpage.action?pageId=520964200&src=contextnavpagetreemode)

# Storm Order Infos

## 1) Change Task Infos

Our Example

Please increase the file system on the server ltdb0007vm.muc  
/global/tldb0007vm/db  
extend at least 100 GB.  
Please allocate whole disk  
FS occupancy > 81%  
APP-ID: APP-103027

The following steps are performed to get the Storage Type associated with the Volume we want to extend.

## 2) Find XEN Farm Servers

[INSTSERV][qx12345][lpinstiaas01] ~ $ cmdb ltdb0007vm | egrep 'farmservers: '

farmservers: lpxentdb003, lpxentdb004

## 3) Where is my VM running?

[INSTSERV][qx12345][lpinstiaas01] ~ $ ssh ltdb0007vm "sudo xenstore-read /tool/hostname"

lpxentdb003

## 4) Get LV / VG Names

Need to find the Logical Volume and Volume Group names. So, on the VM server run:

ltdb0007vm:~ # df -hP /global/tldb0007vm/db/

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/dgtldb0007vm\_ltdb0007vm\_db-lv01 169G 136G 34G 81% /global/tldb0007vm/db

## 5) Get md Devices

LV / VG Names are related to dgtldb0007vm\_ltdb0007vm\_db. Next get the md devices from pvdisplay. So on the active Xen farm server for this VM run:

lpxentdb003: ~ # pvdisplay | grep -B 1 dgtldb0007vm\_ltdb0007vm\_db

PV Name /dev/md4

VG Name dgtldb0007vm\_ltdb0007vm\_db

PV Name /dev/md47

VG Name dgtldb0007vm\_ltdb0007vm\_db

## 6) Relate: SAN LUN <=> md

So our md devices are **md4** and **md47** . We now can find the SAN LUN's associated with these md devices.

llpxentdb003:/home/qxf6163 # cat /proc/mdstat | egrep -w 'md4|md47'

md47 : active raid1 dm-103[0] dm-104[1]

md4 : active raid1 dm-4[0] dm-35[1]

NEW: it is better to grep vm storage for the md devices.

## 7) Get dm-XXX Devices

Use san\_shortinfo on the farmserver to get the dm-XX devices:

llpxentdb003:/home/qxf6163 # san\_shortinfo | egrep -w -e 'dm-103|dm-104|dm-4|dm-35'

dm-103 | sdgl 00:90 1C 97153 115245 | sdgo 00:90 2C 97153 115245 |

dm-35 | sdaf 00:26 1C 97236 115245 | sdbe 00:26 2C 97236 115245 |

dm-4 | sdg 00:26 1C 97153 115245 | sdcd 00:26 2C 97153 115245 |

dm-104 | sdgm 00:90 1C 97236 115245 | sdgn 00:90 2C 97236 115245 |

## 8) Get Storage Type

On the install server use check\_rid.sh to get the Storage Type:

[INSTSERV][qx12345][lpinstiaas01] ~ $ check\_rid.sh lpxentdb003 | grep 97153| egrep -i '00:90|00:26'

SFIZ03AV1 97153 00:00:26 115245 000120694 B2\_GS2 LPXENTDB003 2013-05-14 00:00:00

SFIZ03AV1 97153 00:00:90 115245 000122238 B2\_GS2 LPXENTDB003 2013-06-20 00:00:00

[INSTSERV][qx12345][lpinstiaas01] ~ $ check\_rid.sh lpxentdb003 | grep 97236 | egrep -i '00:26|00:90'

SFIZ03AV2 97236 00:00:26 115245 000120694 B2\_GS2 LPXENTDB003 2013-05-14 00:00:00

SFIZ03AV2 97236 00:00:90 115245 000122238 B2\_GS2 LPXENTDB003 2013-06-20 00:00:00

## 9) Order Storage

Go ahead and order your Storage via the [StorM Tool](https://apex-p.bmwgroup.net:4950/pls/htmldb/f?p=215:1:) using the information collected. In our example:  
Visibility for Servers: lpxentdb003, lpxentdb004  
Storage Type: B2\_GS2  
LUN Size: 112G  
Number of LUN's: 1 (B2\_GS2 Storage comes as a Pair)  
  
For C Class Storage you will need to order 2 LUNs

# Grow the Filesystem

## 10) Storm Order Infos

Your Completed Storm Order will provide the following information:

Extension Order

Ticket ID: 123711 Desired Delivery Date:

Location: FIZ Latest Delivery Date: 20.07.2013

Target SAN: FIZ03 (18.4 1.2/19.0 1.1) Priority: NORMAL

Affected Servers: lpxentdb003,lpxentdb004 Project: ORACLE

Remedy CM Change ID: CRQ000000042355 Remedy CM Task ID: TAS000000077786

..

Order Positions

OrderPosID Storage Class LUN Size LUN Amount Total Size [GB] Usage Consumer Name Special Layout

123712 [B2\_GS2] B2 GS2 112 GB 1 225.088 DB-Data APP-108617 No

Assigned LUNs

OrderPosID Array Name Array Serial LDEV LUN Size [GB] Secured Server Assign Date Remove Date

123712 SFIZ03AV1 97153 00:01:41 112.544 lpxentdb003,lpxentdb004 19.07.2013 -

123712 SFIZ03AV2 97236 00:01:41 112.544 lpxentdb003,lpxentdb004 19.07.2013 -

The "check\_rid" script should provide the same information:

[INSTSERV][qx12345][lpinstiaas01] ~ $ check\_rid.sh 123712

arrayname serial arrayldev size orderposid class secured\_server assigndate

SFIZ03AV1 97153 00:01:41 112544 000123712 B2\_GS2 lpxentdb003,lpxentdb004 2016-02-13 00:00:00

SFIZ03AV2 97236 00:01:41 112544 000123712 B2\_GS2 lpxentdb003,lpxentdb004 2016-02-13 00:00:00

## 11) LUN rescan, and rename

Re-Scan for the new LUN's on your XEN Faremservers and the rename the new LUNs

[INSTSERV][qx12345][lpinstiaas01] ~ $ ssh lpxentdb003 "sudo /sbin/san\_rescan -o"

[INSTSERV][qx12345][lpinstiaas01] ~ $ ssh lpxentdb004 "sudo /sbin/san\_rescan -o"

# the commands below are obsolete!!! DO not use!

#qxf6163@lpinstbss1 ~ $ ssh lpxentdb003 "sudo /sbin/san\_rename"

#qxf6163@lpinstbss1 ~ $ ssh lpxentdb004 "sudo /sbin/san\_rename"

## 12) Find dm-XXX devices

NOTE!!: check where the VM is running again:

[INSTSERV][qx12345][lpinstiaas01] ~ $ ssh ltdb0007vm "sudo xenstore-read /tool/hostname"

On the active Farm Server, to find the new dm-XXX devices run:

lpxentdb003:/home/qxf6163 # san\_shortinfo | grep 97153 | egrep -i '01:41'

dm-151 | sdik 01:41 1C 97153 115245 | sdnp 01:41 2C 97153 115245 |

lpxentdb003:/home/qxf6163 # san\_shortinfo | grep 97236 | egrep -i '01:41'

dm-195 | sdkd 01:41 1C 97236 115245 | sdlw 01:41 2C 97236 115245 |

## 13) SAN Device Names

So our new LUNs have the device names dm-151 and dm-195. Using this information we can now get the SAN device names:

lpxentdb003:/home/qxf6163 # vmc storage | egrep 'dm-151|dm-195'

VSP97153-118-0141, 97153, 113G, 360060e80167b810000017b8100000141, /dev/dm-151, n/a , no md header

VSP97236-118-0141, 97236, 113G, 360060e80167bd40000017bd400000141, /dev/dm-195, n/a , no md header

## 14) Add new devices to VM

lpxentdb003:/home/qxf6163 # vmc -d "VSP97153-118-0141 VSP97236-118-0141" adddisk ltdb0007vm

2013-07-23 10:47:34 INFO All nodes have at least 10 MB of free disk space on /lfs/xen

2013-07-23 10:47:36 INFO Creating new md device

2013-07-23 10:47:36 INFO Creating md device /dev/md49 ... OK

2013-07-23 10:47:37 INFO Writing information into /lfs/xen/config/mdadm.conf ... OK

2013-07-23 10:47:37 INFO Copying /lfs/xen/config/mdadm.conf to lpxentdb004 .... OK

2013-07-23 10:47:38 INFO MDNAME: SAN:ltdb0007vm\_6, VMNAME: ltdb0007vm

2013-07-23 10:47:38 INFO Deactivating md device /dev/md49 ... OK

2013-07-23 10:47:38 INFO Creating new md device

2013-07-23 10:47:39 INFO Activating /dev/md49 on lpxentdb003 OK

2013-07-23 10:47:39 INFO Attaching /dev/md49 to ltdb0007vm as xvdg OK

## 15) Extend Volume Group

Our new disk in the VM is /dev/xvdg. Log onto the VM, create the PV and extend the Volume Group

ltdb0007vm ~ # pvcreate /dev/xvdg

Physical volume "/dev/xvdg" successfully created

ltdb0007vm ~ # vgextend /dev/dgtldb0007vm\_ltdb0007vm\_db /dev/xvdg

## 16) Extend Logical Volume

lvextend -L +112G /dev/dgtldb0007vm\_ltdb0007vm\_db/lv01

or, if all free extents are to be used for the LVOL:

lvextend -l +100%FREE /dev/dgtldb0007vm\_ltdb0007vm\_db/lv01

or to extend to the maximum VG size:

lvextend -l 100%VG /dev/dgtldb0007vm\_ltdb0007vm\_db/lv01

## 17) Grow XFS Filesystem

lltdb0007vm # xfs\_growfs /global/tldb0007vm/db

## 18) Check new FS size

ltdb0007vm:~ # df -hP /global/tldb0007vm/db/

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/dgtldb0007vm\_ltdb0007vm\_db-lv01 281G 136G 145G 48% /global/tldb0007vm/db

| Notes |
| --- |

**BUGs**  
If xfs\_growfs fails with "Killed" and messages like

" \_xfs\_buf\_find: Block out of range:

BUG: unable to handle kernel NULL pointer dereference "

upgrade the kernel.  
  
On 29.11.2013 12:54, Poeschl Andreas, FG-940DG wrote:  
... Der 3.0.93er Kernel hat aber ein großes Problem: XFS-Filesysteme lassen sich nicht erweitern. Dies trifft uns insbesondere in VMs hart. Dafür gibt es nun den 3.0.101er Kernel, der auch dieses Problem behebt ...

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by Miko Andras, (Andras.Miko@partner.bmw.de) , Kujau Christian, (Christian.CK.Kujau@partner.bmw.de) , Urban Richard, (Richard.Urban@partner.bmw.de) , Leeb Reinhold, (Reinhold.RL.Leeb@partner.bmw.de) , Noyes Geoffrey, (Geoffrey.Noyes@partner.bmw.de) and Kemmerer Walter, (Walter.Kemmerer@partner.bmw.de) .  
Page last modified on Wednesday 16 of October, 2019 15:37:58 CEST. (Version 44)

* Created by [Kishore Kumar (ext.)](https://atc.bmwgroup.net/confluence/display/~qxz0crd), last modified on [08 Apr 2020](https://atc.bmwgroup.net/confluence/pages/diffpagesbyversion.action?pageId=521112291&selectedPageVersions=1&selectedPageVersions=2)

| Return SAN Luns or Remove Visiblity VXVM/XEN |
| --- |
| Scope |

NOTE: This HOWTO applies only to SLES11.  
  
NOTE: Only works well with Kernel > 2.6.32.49 SEE: [LINUX Kernel Bug Remove Disks](https://atc.bmwgroup.net/confluence/display/IAAS18/LINUX+Kernel+Bug+Remove+Disks)  
  
NEVER (regardless if VxVM or md/LVM) EVER delete Diskgroup/VolumeGroup/Disks which have been migrated to and are visible on other servers outside of the Standalone-Server/Cluster/Xen-Farm)!  
(Caution: check\_rid.sh <server-name> does \*NOT\* show all servers for which the SAN is visible, check either via the Request-ID or via storm tool!!!)  
  
If you need to unconfigure LUNs from a running system and you have no chance to reboot, then follow this howto.  
First you need to identify the LUNs. ALWAYS run san\_shortinfo on ALL Nodes in the cluster for the Diskgroup/Disks concerned and copy the output into your task, so you have a copy of all disk devices to be removed on the nodes!  
  
NOTE: If you do this in a clustered environment, you need to do the steps 3-8 on the other cluster node as well. NOTE: The other cluster node dosen't have the same device names, so you MUST identify the corresponding device names using SN/LDEV (the san\_shortinfo on that server which you copied into your task above) search.

| Content |
| --- |

# HOWTO Return LUNs without reboot

## Veritas VxVM Linux

### scriptbased (fast way)

#### Quick procedure overview

1. check filesystem on server and search LUNs
2. umount filesystems
3. cleanup fstab
4. destroy DG
5. return luns with san\_return (lpinstbss1)
6. create storm remove order
7. add work info and provision accouting to task and close

#### Detailed Step By Step procedure

1. Locally on Server

* Check FS on server (example task: TAS000000082590) and find disks to return

sudo vxdisk list |egrep -w "dgtlsfava1bkup|dgtlsfava1db" >> tlsfava1.dgs

sudo pvinfo -i |egrep -w "sdac|sdaq|sdd|sdr|sdz" >> pvinfo\_tlsfava1.txt

sudo san\_shortinfo | grep dgtlsfava1 >> san\_shortinfo\_tlsfava1.txt

* Umount FS to return

[qx83246@ltbkdb02:~ ] $ sudo umount /global/tlsfava1/bkup

[qx83246@ltbkdb02:~ ] $ sudo umount /global/tlsfava1/db

* Cleanup fstab

Edit the /etc/fstab file and remove the corresponding line(s) from the files. Please double check every change to ensure that everything is correct.

* Destroy DG from umounted FS

[qx83246@ltbkdb02:~ ] $ sudo vxdg destroy dgtlsfava1bkup

[qx83246@ltbkdb02:~ ] $ sudo vxdg destroy dgtlsfava1db

1. On installserver

* Create a file with pvinfo -i output of return Disks

eg:

[INSTSERV][qx12345][lpinstiaas01] ~ $ cat pvinfo\_return\_tlsfava1.txt

/dev/sdaq 00 00 00 CL6-A-3 12:68 OPEN-V 00078173

/dev/sdac 00 00 00 CL6-A-3 12:66 OPEN-V\*2 00078173

/dev/sdz 00 00 00 CL6-A-3 12:64 OPEN-V\*2 00078173

/dev/sdr 00 00 00 CL6-A-3 12:62 OPEN-V\*2 00078173

/dev/sdd 00 00 00 CL6-A-3 12:5e OPEN-V\*2 00078173

* run san\_return

[INSTSERV][qx12345][lpinstiaas01] ~ $ san\_return -f 'pvinfo\_return\_tlsfava1.txt' -s ltbkdb02 -t 'TAS000000082590'

# --- For a detailed log, please see: '/global/instserv/logs/san\_return/ltbkdb02\_TAS000000082590\_2013\_08\_08\_082033.log'

# --- User 'qx83246' fired command: /global/instserv/bin/san\_return -f pvinfo\_return\_tlsfava1.txt -s ltbkdb02 -t TAS000000082590

# --- Remedy ticketnumber: TAS000000082590

# --- Displaying input file 'pvinfo\_return\_tlsfava1.txt' ...:

/dev/sdaq 00 00 00 CL6-A-3 12:68 OPEN-V 00078173

/dev/sdac 00 00 00 CL6-A-3 12:66 OPEN-V\*2 00078173

/dev/sdz 00 00 00 CL6-A-3 12:64 OPEN-V\*2 00078173

/dev/sdr 00 00 00 CL6-A-3 12:62 OPEN-V\*2 00078173

/dev/sdd 00 00 00 CL6-A-3 12:5e OPEN-V\*2 00078173

... done

ltbkdb02: # - Fetching cluster partner list ...

ltbkdb02: ... done

ltbkdb02: # - Checking SLES version ...

ltbkdb02: ... done

ltbkdb02: # - Checking Veritas version ...

ltbkdb02: ... done

ltbkdb02: # - Checking availability of needed commands ...

ltbkdb02: ... done

ltbkdb02: # - Getting 'pvinfo' data ...

ltbkdb02: ... done

ltbkdb02: # - Building LUN records ...

ltbkdb02: ... done

ltbkdb02: # - Getting 'pvinfo -i' data ...

ltbkdb02: ... done

ltbkdb02: # - Getting Veritas device information ...

ltbkdb02: ... done

ltbkdb02: # - Listing LUNs before being unconfigured ...:

ltbkdb02: ENCLR-SN CU:LDEV VxDMP PHYSICAL DEVICES (PATHS) SIZE

ltbkdb02: ======== ======= ===== ======================== ========

ltbkdb02: 78173 12:5E sdd sday,sdd 28811

ltbkdb02: 78173 12:62 sdr sdaz,sdr 28811

ltbkdb02: 78173 12:64 sdz sdba,sdz 28811

ltbkdb02: 78173 12:66 sdac sdbb,sdac 28811

ltbkdb02: 78173 12:68 sdaq sdbc,sdaq 14405

ltbkdb02: ... done

ltbkdb02: # - Starting LUN removal, for a detailed log see /global/instserv/logs/san\_return/ltbkdb02\_TAS000000082590\_2013\_08\_08\_082033.log on server lpinstbss1

ltbkdb02: # - Placing '/etc/.donotrescan' file to avoid redetection of unconfigured LUNs ...

ltbkdb02: ... done

ltbkdb02: # - Wiping LUN headers ...

ltbkdb02: \*STDERR: 65536+0 records in

65536+0 records out

33554432 bytes (34 MB) copied, 1.38688 seconds, 24.2 MB/s

ltbkdb02: \*STDERR: 65536+0 records in

65536+0 records out

33554432 bytes (34 MB) copied, 1.45351 seconds, 23.1 MB/s

ltbkdb02: \*STDERR: 65536+0 records in

65536+0 records out

33554432 bytes (34 MB) copied, 1.56597 seconds, 21.4 MB/s

ltbkdb02: \*STDERR: 65536+0 records in

65536+0 records out

33554432 bytes (34 MB) copied, 1.51793 seconds, 22.1 MB/s

ltbkdb02: \*STDERR: 65536+0 records in

65536+0 records out

33554432 bytes (34 MB) copied, 1.67007 seconds, 20.1 MB/s

ltbkdb02: ... done

ltbkdb02: # - Removing selected LUNs from Veritas control ...

ltbkdb02: ... done

ltbkdb02: # - Unconfiguring LUN paths ...

ltbkdb02: ... done

ltbkdb02: # - Verifying removal of paths ...

ltbkdb02: ... done

ltbkdb02: # - Scanning devices and refreshing VxDMP device name database ...

ltbkdb02: ... done

ltbkdb02: # - Verifying removal of VxDMP devices ...

ltbkdb02: ... done

######### 'ltbkdb02' finished ##################################################################

* open a Return Ticket on storm.muc

### manual way

1. Save the information about disks to get removed

> san\_shortinfo dgtlsfdwh8db$ | tee san\_shortinfo.<task\_number>

VM | RAW CU:LDEV PORT XP-SN SIZE | RAW CU:LDEV PORT XP-SN SIZE | DG

------|---------------------------------|---------------------------------|------

sdw | sdw 25:4C 4C 78135 57622 | sdbi 25:4C 3C 78135 57622 | dgtlsfdwh8db

sdx | sdx 25:50 4C 78135 57622 | sdbj 25:50 3C 78135 57622 | dgtlsfdwh8db

sdy | sdy 25:54 4C 78135 57622 | sdbk 25:54 3C 78135 57622 | dgtlsfdwh8db

sdz | sdz 25:58 4C 78135 57622 | sdbl 25:58 3C 78135 57622 | dgtlsfdwh8db

2. Remove them from Diskgroup using vxdg -g <diskgroup> rmdisk <dmnames>  
e.g:

vxdg -g dgtlsfdwh8db rmdisk 13\_78135\_07\_254C 13\_78135\_08\_2550 13\_78135\_09\_2554 13\_78135\_10\_2558

3. Remove the header using vxdiskunsetup <veritas device name>  
e.g:

for i in sdw sdx sdy sdz;do vxdiskunsetup $i;done

4. Offline them:  
e.g:

for i in sdw sdx sdy sdz;do vxdisk offline $i;done

5. Delete them:  
e.g:

for i in sdw sdx sdy sdz;do vxdisk rm $i;done

6. Unconfigure the sub pathes from DMP using vxdmpadm -f disable path=sdxx  
e.g:

for i in sdw sdx sdy sdz sdbi sdbj sdbk sdbl;do vxdmpadm -f disable path=$i;done

7. Unconfigure them from the OS using echo 1 > /sys/block/sdxx/device/delete  
e.g:

for i in sdw sdx sdy sdz sdbi sdbj sdbk sdbl;do echo 1 > /sys/block/$i/device/delete;done

8. Use "vxdctl enable" and "vxdisk scandisks" to completely remove them from DMP  
e.g:

vxdctl enable

vxdisk scandisks

Then check the DMP configuration, if there are NO DISABLED DMP nodes  
e.g:

vxdmpadm getsubpaths | grep DISABLED

If you find any DISABLED nodes, then you forgot either to remove them from OS Level, or to execute the above 2 commands(vxdctl enable, vxdisk scandisks)  
  
9. Use "vxddladm assign names" to recreate the missing DMP node names under /dev/vx/dsk/dmp

vxddladm assign names

10. On the Server resp on all Nodes of the Cluster create a file /etc/.donotrescan so the disks are not reconfigured before they have been removed from the system. Always add the change/task number e.g.

lp34d20:/home/qxf5649 # cat /etc/.donotrescan  
SAN Rueckgabe in Progress CRQ000000039230/TAS000000077642 geoff  
lp34d20:/home/qxf5649 #

IF there is already an entry, add the additional entry (and perhaps check if the Task has been completed and someone forgot to remove the entry)  
  
NOTE: If you do this in a clustered environment, you need to do the steps 1 and 3-8 on the other cluster node as well.  
NOTE: The other cluster node dosen't have the same device names, so you MUST identify the corresponding device names  
using SN/LDEV (the san\_shortinfo on that server which you copied into your task above) search.  
  
11. Now you can return the LUNs with StorM (http://storm.muc/). Use the saved san\_shortinfo.<task\_number> file to select the LUNs in StorM. DO NOT close the task until the STORM Ticket has been completed. When completed check the Servers, run "san\_rescan -o" and check if the disks (SN + LDEV) have been removed. If OK, remove your entry from the file /etc/.donotrescan, if the file is then empty, delete it.  
  
Done

## 2. XEN LVM

At a first step, save the information about the disks to be returned on both XEN farm nodes. Be sure to remove the disks from your output, that should be kept. Check the result - twice! This info can be used later to return the disks:

vm storage <vmname>

ls -l /dev/md/<vmname>\*

pvinfo -i | grep $(vm storage <vmname> | awk 'NR>2 {print "-e " substr($3,1,2) ":" substr($3,3) ".\*" $2 "$"}') | tee pvinfo.<taskno>

The second step is to gather information about multipath devices:

multipath -ll | grep $(vm storage <vmname> | awk '/dm-/ {print "-e " $5}') | tee multipath.<taskno>

* On DomU:

umount <LV(s)>  
cleanup fstab  
vgchange -a n /dev/<VG>  
lvremove <LV(s)>  
vgremove -f<VG>  
pvremove /dev/DEV (you get from vm info VM)

* On Dom0s

Before mulipath handling below first remove disk abstractions:  
vm rmdisk <vmname> /dev/DEV  
  
Step-by-Step:

#In case you are unsure, which devices belong together

mdadm --query --detail /dev/md4

#Remove old disks from configuration

mdadm --fail /dev/md4 /dev/dm-30

mdadm --fail /dev/md4 /dev/dm-24

mdadm --remove /dev/md4 /dev/dm-30

mdadm --remove /dev/md4 /dev/dm-24

#Clear Superblock to prepare return ticket

mdadm --zero-superblock /dev/dm-30

mdadm --zero-superblock /dev/dm-24

#Show infos of raw devices

#(this information is also available in pvinfo.<taskno>)

lpxens30:/home/qqlinux # san\_shortinfo | grep -w -e dm-30 -e dm-24

dm-30 | sdk 29:56 3A 29755 57622 | sdae 29:56 4A 29755 57622 |

dm-24 | sdb 2B:BC 3A 45312 57622 | sdv 2B:BC 4A 45312 57622 |

#Delete Device from Multipathing

dmsetup remove ST29755-59-2956

dmsetup remove ST45312-59-2bbc

#Delete Device on OS Level

echo 1 > /sys/block/sdk/device/delete

echo 1 > /sys/block/sdae/device/delete

echo 1 > /sys/block/sdb/device/delete

echo 1 > /sys/block/sdv/device/delete

# HOWTO Remove Visibility LUNs without reboot (does not remove DATA)

## Veritas VxVM Linux

### scriptbased (fast way)

1. Locally on Server

* Check FS on Server (example Task: TAS000000082590) and find Disks to return

sudo vxdisk -o alldgs list |egrep "dgtlsfava1bkup|dgtlsfava1db" >> tlsfava1.dgs

sudo pvinfo -i |egrep -w "sdac|sdaq|sdd|sdr|sdz" >> pvinfo\_tlsfava1.txt

sudo san\_shortinfo | grep dgtlsfava1 >> san\_shortinfo\_tlsfava1.txt

* Verify that DGS and FS not active on Server

If you should only remove visibilty on Server it is important that the affected DG is not imported on requested Server/s!  
vxdisk -o alldgs list must show the DG in "()", please be carefully on Cluster Systems, if it is requested to remove visibility of a full Cluster, both nodes must be checked for the inactive DG!  
e.g.  
sdv auto:cdsdisk - (dgTEST) online thinrclm

1. On installserver

* Create a file with pvinfo -s output of return Disks

eg:

[INSTSERV][qx12345][lpinstiaas01] ~ $ cat pvinfo\_return\_tlsfava1.txt

/dev/sdaq 00 00 00 CL6-A-3 12:68 OPEN-V 00078173

/dev/sdac 00 00 00 CL6-A-3 12:66 OPEN-V\*2 00078173

/dev/sdz 00 00 00 CL6-A-3 12:64 OPEN-V\*2 00078173

/dev/sdr 00 00 00 CL6-A-3 12:62 OPEN-V\*2 00078173

/dev/sdd 00 00 00 CL6-A-3 12:5e OPEN-V\*2 00078173

* run san\_return

[INSTSERV][qx12345][lpinstiaas01] ~ $ san\_return \_\_-m\_\_ -f 'pvinfo\_return\_tlsfava1.txt' -s ltbkdb02 -t 'TAS000000082590'

# --- For a detailed log, please see: '/global/instserv/logs/san\_return/ltbkdb02\_TAS000000082590\_2013\_08\_08\_082033.log'

# --- User 'qx83246' fired command: /global/instserv/bin/san\_return -f pvinfo\_return\_tlsfava1.txt -s ltbkdb02 -t TAS000000082590

# --- Remedy ticketnumber: TAS000000082590

# --- Displaying input file 'pvinfo\_return\_tlsfava1.txt' ...:

/dev/sdaq 00 00 00 CL6-A-3 12:68 OPEN-V 00078173

/dev/sdac 00 00 00 CL6-A-3 12:66 OPEN-V\*2 00078173

/dev/sdz 00 00 00 CL6-A-3 12:64 OPEN-V\*2 00078173

/dev/sdr 00 00 00 CL6-A-3 12:62 OPEN-V\*2 00078173

/dev/sdd 00 00 00 CL6-A-3 12:5e OPEN-V\*2 00078173

* open a Remove Visibilty Ticket on storm.muc

Last edited by Liebl Markus, (Markus.Liebl@partner.bmw.de) , based on work by [qxj5457](https://bsswiki.muc/tiki-user_information.php?userId=-1) , [qxc0474](https://bsswiki.muc/tiki-user_information.php?userId=-1) , Linden Sylvia, (Sylvia.LA.Linden@partner.bmw.de) , Lianas Zisis, (Zisis.Lianas@partner.bmw.de) , Urban Richard, (Richard.Urban@partner.bmw.de) , Kirchberger Marco, (Marco.Kirchberger@partner.bmw.de) , Noyes Geoffrey, (Geoffrey.Noyes@partner.bmw.de) , Hoferichter Philipp, (Philipp.Hoferichter@partner.bmw.de) and [system](https://bsswiki.muc/tiki-user_information.php?userId=-1) .  
Page last modified on Tuesday 15 of October, 2019 16:55:45 CEST. (Version 49)