NAME

lvchange – Change the attributes of logical volume(s)

SYNOPSIS

lvchange *option_args position_args*
[*option_args*]

–a|––activate y|n|ay
––activationmode partial|degraded|complete
––addtag *Tag*
––alloc contiguous|cling|cling_by_tags|normal|anywhere|inherit
–A|––autobackup y|n
––cachemode writethrough|writeback|passthrough
––cachepolicy *String*
––cachesettings *String*
––commandprofile *String*
––compression y|n
––config *String*
–C|––contiguous y|n
–d|––debug
––deduplication y|n
––deltag *Tag*
––detachprofile
––discards passdown|nopassdown|ignore
––driverloaded y|n
––errorwhenfull y|n
–f|––force
–h|––help
–K|––ignoreactivationskip
––ignorelockingfailure
––ignoremonitoring
––lockopt *String*
––longhelp
–j|––major *Number*
––[raid]maxrecoveryrate *Size*[k|UNIT]
––metadataprofile *String*
––minor *Number*
––[raid]minrecoveryrate *Size*[k|UNIT]
––monitor y|n
––nolocking
––noudevsync
–P|––partial
–p|––permission rw|r
–M|––persistent y|n
––poll y|n
––profile *String*
–q|––quiet
–r|––readahead auto|none|*Number*
––readonly
––rebuild *PV*
––refresh
––reportformat basic|json
––resync
–S|––select *String*
–k|––setactivationskip y|n
––[raid]syncaction check|repair

--sysinit
-t|--test
-v|--verbose
--version
--[raid]writebehind *Number*
--[raid]writemostly *PV*[**:t|n|y**]
-y|--yes
-Z|--zero y|n

DESCRIPTION

lvchange changes LV attributes in the VG, changes LV activation in the kernel, and includes other utilities for LV maintenance.

USAGE

Change a general LV attribute.
For options listed in parentheses, any one is
required, after which the others are optional.

lvchange
(**-C|--contiguous y|n**,
-p|--permission rw|r,
-r|--readahead auto|none|*Number*,
-k|--setactivationskip y|n,
-Z|--zero y|n,
-M|--persistent n,
--addtag *Tag*,
--deltag *Tag*,
--alloc contiguous|cling|cling_by_tags|normal|anywhere|inherit,
--compression y|n,
--deduplication y|n,
--detachprofile,
--metadataprofile *String*,
--profile *String*,
--errorwhenfull y|n,
--discards passdown|nopassdown|ignore,
--cachemode writethrough|writeback|passthrough,
--cachepolicy *String*,
--cachesettings *String*,
--[raid]minrecoveryrate *Size*[k|UNIT],
--[raid]maxrecoveryrate *Size*[k|UNIT],
--[raid]writebehind *Number*,
--[raid]writemostly *PV*[**:t|n|y**])
VG|LV|Tag|Select ...
[**-a|--activate y|n|ay**]
[**--poll y|n**]
[**--monitor y|n**]
[COMMON_OPTIONS]

-

Resyncronize a mirror or raid LV.
Use to reset 'R' attribute on a not initially synchronized LV.

lvchange --resync *VG|LV_mirror_raid|Tag|Select ...*
[**-a|--activate y|n|ay**]
[COMMON_OPTIONS]

-

Resynchronize or check a raid LV.

lvchange ––syncaction check|repair *VG|LV_raid|Tag|Select ...*
[COMMON_OPTIONS]
-

Reconstruct data on specific PVs of a raid LV.

lvchange ––rebuild *PV VG|LV_raid|Tag|Select ...*
[COMMON_OPTIONS]
-

Activate or deactivate an LV.

lvchange –a|––activate y|n|ay *VG|LV|Tag|Select ...*
[**–P|––partial**]
[**–K|––ignoreactivationskip**]
[**––activationmode partial|degraded|complete**]
[**––poll y|n**]
[**––monitor y|n**]
[**––ignorelockingfailure**]
[**––sysinit**]
[**––readonly**]
[COMMON_OPTIONS]
-

Reactivate an LV using the latest metadata.

lvchange ––refresh *VG|LV|Tag|Select ...*
[**–P|––partial**]
[**––activationmode partial|degraded|complete**]
[**––poll y|n**]
[**––monitor y|n**]
[COMMON_OPTIONS]
-

Start or stop monitoring an LV from dmeventd.

lvchange ––monitor y|n *VG|LV|Tag|Select ...*
[COMMON_OPTIONS]
-

Start or stop processing an LV conversion.

lvchange ––poll y|n *VG|LV|Tag|Select ...*
[**––monitor y|n**]
[COMMON_OPTIONS]
-

Make the minor device number persistent for an LV.

lvchange –M|––persistent y ––minor *Number LV*
[**–j|––major** *Number*]
[**–a|––activate y|n|ay**]
[**––poll y|n**]
[**––monitor y|n**]
[COMMON_OPTIONS]
-

Common options for command:
[**–A|––autobackup y|n**]
[**–f|––force**]
[**–S|––select** *String*]
[**––ignoremonitoring**]
[**––noudevsync**]
[**––reportformat basic|json**]

Common options for lvm:
[**–d|––debug**]
[**–h|––help**]
[**–q|––quiet**]
[**–t|––test**]
[**–v|––verbose**]
[**–y|––yes**]
[**––commandprofile** *String*]
[**––config** *String*]
[**––driverloaded y|n**]
[**––lockopt** *String*]
[**––longhelp**]
[**––nolocking**]
[**––profile** *String*]
[**––version**]

OPTIONS

–a|––activate y|n|ay

Change the active state of LVs. An active LV can be used through a block device, allowing data on the LV to be accessed. **y** makes LVs active, or available. **n** makes LVs inactive, or unavailable. The block device for the LV is added or removed from the system using device-mapper in the kernel. A symbolic link /dev/VGName/LVName pointing to the device node is also added/removed. All software and scripts should access the device through the symbolic link and present this as the name of the device. The location and name of the underlying device node may depend on the distribution, configuration (e.g. udev), or release version. **ay** specifies autoactivation, in which case an LV is activated only if it matches an item in lvm.conf activation/auto_activation_volume_list. If the list is not set, all LVs are considered to match, and if if the list is set but empty, no LVs match. Autoactivation should be used during system boot to make it possible to select which LVs should be automatically activated by the system. See **lvmlockd**(8) for more information about activation options **ey** and **sy** for shared VGs.

––activationmode partial|degraded|complete

Determines if LV activation is allowed when PVs are missing, e.g. because of a device failure. **complete** only allows LVs with no missing PVs to be activated, and is the most restrictive mode. **degraded** allows RAID LVs with missing PVs to be activated. (This does not include the "mirror" type, see "raid1" instead.) **partial** allows any LV with missing PVs to be activated, and should only be used for recovery or repair. For default, see lvm.conf/activation_mode. See **lvmraid**(7) for more information.

––addtag *Tag*

Adds a tag to a PV, VG or LV. This option can be repeated to add multiple tags at once. See **lvm**(8) for information about tags.

––alloc contiguous|cling|cling_by_tags|normal|anywhere|inherit

Determines the allocation policy when a command needs to allocate Physical Extents (PEs) from the VG. Each VG and LV has an allocation policy which can be changed with vgchange/lvchange, or overriden on the command line. **normal** applies common sense rules such as not placing parallel stripes on the same PV. **inherit** applies the VG policy to an LV. **contiguous** requires new PEs be placed adjacent to existing PEs. **cling** places new PEs on the same PV as existing PEs in the

same stripe of the LV. If there are sufficient PEs for an allocation, but normal does not use them, **anywhere** will use them even if it reduces performance, e.g. by placing two stripes on the same PV. Optional positional PV args on the command line can also be used to limit which PVs the command will use for allocation. See **lvm**(8) for more information about allocation.

−A|−−autobackup y|n
Specifies if metadata should be backed up automatically after a change. Enabling this is strongly advised! See **vgcfgbackup**(8) for more information.

−−cachemode writethrough|writeback|passthrough
Specifies when writes to a cache LV should be considered complete. **writeback** considers a write complete as soon as it is stored in the cache pool. **writethough** considers a write complete only when it has been stored in both the cache pool and on the origin LV. While writethrough may be slower for writes, it is more resilient if something should happen to a device associated with the cache pool LV. With **passthrough**, all reads are served from the origin LV (all reads miss the cache) and all writes are forwarded to the origin LV; additionally, write hits cause cache block invalidates. See **lvmcache**(7) for more information.

−−cachepolicy *String*
Specifies the cache policy for a cache LV. See **lvmcache**(7) for more information.

−−cachesettings *String*
Specifies tunable values for a cache LV in "Key = Value" form. Repeat this option to specify multiple values. (The default values should usually be adequate.) The special string value **default** switches settings back to their default kernel values and removes them from the list of settings stored in LVM metadata. See **lvmcache**(7) for more information.

−−commandprofile *String*
The command profile to use for command configuration. See **lvm.conf**(5) for more information about profiles.

−−compression y|n
Controls whether compression is enabled or disable for VDO volume. See **lvmvdo**(7) for more information about VDO usage.

−−config *String*
Config settings for the command. These override lvm.conf settings. The String arg uses the same format as lvm.conf, or may use section/field syntax. See **lvm.conf**(5) for more information about config.

−C|−−contiguous y|n
Sets or resets the contiguous allocation policy for LVs. Default is no contiguous allocation based on a next free principle. It is only possible to change a non-contiguous allocation policy to contiguous if all of the allocated physical extents in the LV are already contiguous.

−d|−−debug ...
Set debug level. Repeat from 1 to 6 times to increase the detail of messages sent to the log file and/or syslog (if configured).

−−deduplication y|n
Controls whether deduplication is enabled or disable for VDO volume. See **lvmvdo**(7) for more information about VDO usage.

−−deltag *Tag*
Deletes a tag from a PV, VG or LV. This option can be repeated to delete multiple tags at once. See **lvm**(8) for information about tags.

−−detachprofile
Detaches a metadata profile from a VG or LV. See **lvm.conf**(5) for more information about profiles.

−−discards passdown|nopassdown|ignore
Specifies how the device-mapper thin pool layer in the kernel should handle discards. **ignore** causes the thin pool to ignore discards. **nopassdown** causes the thin pool to process discards itself to allow reuse of unneeded extents in the thin pool. **passdown** causes the thin pool to process discards itself (like nopassdown) and pass the discards to the underlying device. See **lvmthin**(7) for more information.

−−driverloaded y|n
If set to no, the command will not attempt to use device-mapper. For testing and debugging.

−−errorwhenfull y|n
Specifies thin pool behavior when data space is exhausted. When yes, device-mapper will immediately return an error when a thin pool is full and an I/O request requires space. When no, device-mapper will queue these I/O requests for a period of time to allow the thin pool to be extended. Errors are returned if no space is available after the timeout. (Also see dm-thin−pool kernel module option no_space_timeout.) See **lvmthin**(7) for more information.

−f|−−force ...
Override various checks, confirmations and protections. Use with extreme caution.

−h|−−help
Display help text.

−K|−−ignoreactivationskip
Ignore the "activation skip" LV flag during activation to allow LVs with the flag set to be activated.

−−ignorelockingfailure
Allows a command to continue with read-only metadata operations after locking failures.

−−ignoremonitoring
Do not interact with dmeventd unless −−monitor is specified. Do not use this if dmeventd is already monitoring a device.

−−lockopt *String*
Used to pass options for special cases to lvmlockd. See **lvmlockd**(8) for more information.

−−longhelp
Display long help text.

−j|−−major *Number*
Sets the major number of an LV block device.

−−[raid]maxrecoveryrate *Size*[k|UNIT]
Sets the maximum recovery rate for a RAID LV. The rate value is an amount of data per second for each device in the array. Setting the rate to 0 means it will be unbounded. See **lvmraid**(7) for more information.

−−metadataprofile *String*
The metadata profile to use for command configuration. See **lvm.conf**(5) for more information about profiles.

−−minor *Number*
Sets the minor number of an LV block device.

−−[raid]minrecoveryrate *Size*[k|UNIT]
Sets the minimum recovery rate for a RAID LV. The rate value is an amount of data per second for each device in the array. Setting the rate to 0 means it will be unbounded. See **lvmraid**(7) for more information.

−−monitor y|n
Start (yes) or stop (no) monitoring an LV with dmeventd. dmeventd monitors kernel events for an LV, and performs automated maintenance for the LV in reponse to specific events. See **dmeventd**(8) for more information.

−−nolocking
Disable locking.

−−noudevsync
Disables udev synchronisation. The process will not wait for notification from udev. It will continue irrespective of any possible udev processing in the background. Only use this if udev is not running or has rules that ignore the devices LVM creates.

−P|−−partial
Commands will do their best to activate LVs with missing PV extents. Missing extents may be replaced with error or zero segments according to the lvm.conf missing_stripe_filler setting. Metadata may not be changed with this option.

−p|−−permission rw|r
Set access permission to read only **r** or read and write **rw**.

−M|−−persistent y|n
When yes, makes the specified minor number persistent.

−−poll y|n
When yes, start the background transformation of an LV. An incomplete transformation, e.g. pvmove or lvconvert interrupted by reboot or crash, can be restarted from the last checkpoint with −−poll y. When no, background transformation of an LV will not occur, and the transformation will not complete. It may not be appropriate to immediately poll an LV after activation, in which case −−poll n can be used to defer polling until a later −−poll y command.

−−profile *String*
An alias for −−commandprofile or −−metadataprofile, depending on the command.

−q|−−quiet ...
Suppress output and log messages. Overrides −−debug and −−verbose. Repeat once to also suppress any prompts with answer 'no'.

−r|−−readahead auto|none|*Number*
Sets read ahead sector count of an LV. **auto** is the default which allows the kernel to choose a suitable value automatically. **none** is equivalent to zero.

−−readonly
Run the command in a special read-only mode which will read on-disk metadata without needing to take any locks. This can be used to peek inside metadata used by a virtual machine image while the virtual machine is running. No attempt will be made to communicate with the device-mapper kernel driver, so this option is unable to report whether or not LVs are actually in use.

−−rebuild *PV*
Selects a PV to rebuild in a raid LV. Multiple PVs can be rebuilt by repeating this option. Use this option in place of −−resync or −−syncaction repair when the PVs with corrupted data are known, and their data should be reconstructed rather than reconstructing default (rotating) data. See **lvmraid**(7) for more information.

−−refresh
If the LV is active, reload its metadata. This is not necessary in normal operation, but may be useful if something has gone wrong, or if some form of manual LV sharing is being used.

−−reportformat basic|json
Overrides current output format for reports which is defined globally by the report/output_format setting in lvm.conf. **basic** is the original format with columns and rows. If there is more than one report per command, each report is prefixed with the report name for identification. **json** produces report output in JSON format. See **lvmreport**(7) for more information.

−−resync
Initiates mirror synchronization. Synchronization generally happens automatically, but this option forces it to run. Also see −−rebuild to synchronize a specific PV. During synchronization, data is

read from the primary mirror device and copied to the others. This can take considerable time, during which the LV is without a complete redundant copy of the data. See **lvmraid**(7) for more information.

–S|––select *String*
Select objects for processing and reporting based on specified criteria. The criteria syntax is described by **––select help** and **lvmreport**(7). For reporting commands, one row is displayed for each object matching the criteria. See **––options help** for selectable object fields. Rows can be displayed with an additional "selected" field (–o selected) showing 1 if the row matches the selection and 0 otherwise. For non-reporting commands which process LVM entities, the selection is used to choose items to process.

–k|––setactivationskip y|n
Persistently sets (yes) or clears (no) the "activation skip" flag on an LV. An LV with this flag set is not activated unless the -–ignoreactivationskip option is used by the activation command. This flag is set by default on new thin snapshot LVs. The flag is not applied to deactivation. The current value of the flag is indicated in the lvs lv_attr bits.

––[raid]syncaction check|repair
Initiate different types of RAID synchronization. This causes the RAID LV to read all data and parity blocks in the array and check for discrepancies (mismatches between mirrors or incorrect parity values). **check** will count but not correct discrepancies. **repair** will correct discrepancies. See lvs for reporting discrepancies found or repaired.

––sysinit
Indicates that vgchange/lvchange is being invoked from early system initialisation scripts (e.g. rc.sysinit or an initrd), before writable filesystems are available. As such, some functionality needs to be disabled and this option acts as a shortcut which selects an appropriate set of options. Currently, this is equivalent to using ––ignorelockingfailure, ––ignoremonitoring, -–poll n, and setting env var LVM_SUPPRESS_LOCKING_FAILURE_MESSAGES. vgchange/lvchange skip autoactivation, and defer to pvscan autoactivation.

–t|––test
Run in test mode. Commands will not update metadata. This is implemented by disabling all metadata writing but nevertheless returning success to the calling function. This may lead to unusual error messages in multi-stage operations if a tool relies on reading back metadata it believes has changed but hasn't.

–v|––verbose ...
Set verbose level. Repeat from 1 to 4 times to increase the detail of messages sent to stdout and stderr.

––version
Display version information.

––[raid]writebehind *Number*
The maximum number of outstanding writes that are allowed to devices in a RAID1 LV that is marked write-mostly. Once this value is exceeded, writes become synchronous (i.e. all writes to the constituent devices must complete before the array signals the write has completed). Setting the value to zero clears the preference and allows the system to choose the value arbitrarily.

––[raid]writemostly *PV*[**:t|n|y**]
Mark a device in a RAID1 LV as write-mostly. All reads to these drives will be avoided unless absolutely necessary. This keeps the number of I/Os to the drive to a minimum. The default behavior is to set the write-mostly attribute for the specified PV. It is also possible to remove the write-mostly flag by adding the suffix **:n** at the end of the PV name, or to toggle the value with the suffix **:t**. Repeat this option to change the attribute on multiple PVs.

–y|––yes
Do not prompt for confirmation interactively but always assume the answer yes. Use with extreme

caution. (For automatic no, see −qq.)

−Z|−−zero y|n

Set zeroing mode for thin pool. Note: already provisioned blocks from pool in non-zero mode are not cleared in unwritten parts when setting −−zero y.

VARIABLES

VG

Volume Group name. See **lvm**(8) for valid names.

LV

Logical Volume name. See **lvm**(8) for valid names. An LV positional arg generally includes the VG name and LV name, e.g. VG/LV. LV followed by _<type> indicates that an LV of the given type is required. (raid represents raid<N> type)

Tag

Tag name. See **lvm**(8) for information about tag names and using tags in place of a VG, LV or PV.

Select

Select indicates that a required positional parameter can be omitted if the **−−select** option is used. No arg appears in this position.

String

See the option description for information about the string content.

Size[UNIT]

Size is an input number that accepts an optional unit. Input units are always treated as base two values, regardless of capitalization, e.g. 'k' and 'K' both refer to 1024. The default input unit is specified by letter, followed by |UNIT. UNIT represents other possible input units: **bBsSkKmMgGtTpPeE**. b|B is bytes, s|S is sectors of 512 bytes, k|K is kilobytes, m|M is megabytes, g|G is gigabytes, t|T is terabytes, p|P is petabytes, e|E is exabytes. (This should not be confused with the output control −−units, where capital letters mean multiple of 1000.)

ENVIRONMENT VARIABLES

See **lvm**(8) for information about environment variables used by lvm. For example, LVM_VG_NAME can generally be substituted for a required VG parameter.

EXAMPLES

Change LV permission to read-only:

lvchange −pr vg00/lvol1

SEE ALSO

lvm(8) **lvm.conf**(5) **lvmconfig**(8)

pvchange(8) **pvck**(8) **pvcreate**(8) **pvdisplay**(8) **pvmove**(8) **pvremove**(8) **pvresize**(8) **pvs**(8) **pvscan**(8)

vgcfgbackup(8) **vgcfgrestore**(8) **vgchange**(8) **vgck**(8) **vgcreate**(8) **vgconvert**(8) **vgdisplay**(8) **vgexport**(8) **vgextend**(8) **vgimport**(8) **vgimportclone**(8) **vgmerge**(8) **vgmknodes**(8) **vgreduce**(8) **vgremove**(8) **vgrename**(8) **vgs**(8) **vgscan**(8) **vgsplit**(8)

lvcreate(8) **lvchange**(8) **lvconvert**(8) **lvdisplay**(8) **lvextend**(8) **lvreduce**(8) **lvremove**(8) **lvrename**(8) **lvresize**(8) **lvs**(8) **lvscan**(8)

lvm-fullreport(8) **lvm-lvpoll**(8) **lvm2−activation−generator**(8) **blkdeactivate**(8) **lvmdump**(8)

dmeventd(8) **lvmpolld**(8) **lvmlockd**(8) **lvmlockctl**(8) **cmirrord**(8) **lvmdbusd**(8)

lvmsystemid(7) **lvmreport**(7) **lvmraid**(7) **lvmthin**(7) **lvmcache**(7)