Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'sg_format.8' command

$ man sg_format.8

SG_FORMAT(8)   SG3_UTILS   SG_FORMAT(8)

NAME

sg_format - format, format with preset, resize SCSI disk; format tape

SYNOPSIS

sg_format   [--cmplst={0|1}]   [--count=COUNT]   [--dcrt]   [--dry-run]

[--early] [--ffmt=FFMT] [--fmtmaxlba-fmtpinfo=FPI] [--format] [--help]

[--ip-def]   [--long] [--mode=MP] [--pfu=PFU] [--pie=PIE] [--pinfo]

[--poll=PT] [--preset=ID] [--quick] [--resize] [--rto_req] [--security]

[--six] [--size=LB_SZ] [--tape=FM] [--timeout=SECS] [--verbose] [--ver?

ify] [--version] [--wait] DEVICE

DESCRIPTION

Not all SCSI direct access devices need to be formatted and some have

vendor specific formatting procedures. SCSI disks with rotating media

are probably the largest group that do support a 'standard' format op?

eration. They are typically factory formatted to a block size of 512

bytes with the largest number of blocks that the manufacturer recom?

mends. The manufacturer's recommendation typically leaves aside a cer?

tain number of tracks, spread across the media, for reassignment of

blocks to logical block addresses during the life of the disk.

This utility issues one of three SCSI format commands: FORMAT UNIT,

FORMAT MEDIUM or FORMAT WITH PRESET. In the following description, un?

qualified sections will usually be referring to the SCSI FORMAT UNIT

command. Both FORMAT UNIT and FORMAT WITH PRESET apply to disks (or disk-like devices). The FORMAT MEDIUM command is for tapes.

This utility can format modern SCSI disks and potentially change their block size (if permitted) and the block count (i.e. number of accessible blocks on the media also known as "resizing"). Resizing a disk to less than the manufacturer's recommended block count is sometimes called "short stroking" (see NOTES section). Resizing the block count while not changing the block size may not require a format operation.

The SBC-2 standard (see www.t10.org) has obsoleted the "format device" mode page. Many of the low level details found in that mode page are now left up to the discretion of the manufacturer. There is a Format Status log page which reports on the previous successful format operation(s).

When this utility is used without options (i.e. it is only given a DEVICE argument) it prints out the existing block size and block count derived from two sources. These two sources are a block descriptor in the response to a MODE SENSE command and the response to a READ CAPACITY command. The reason for this double check is to detect a "format corrupt" state (see the NOTES section). This usage will not modify the disk.

When this utility is used with either --format, --preset=ID or --tape=FM, it will attempt to format the given DEVICE. In the absence of the --quick option there is a 15 second pause during which time the user is invited thrice (5 seconds apart) to abort sg_format. This occurs just prior the SCSI FORMAT UNIT, FORMAT WITH PRESET or FORMAT MEDIUM command being issued. See the NOTES section for more information.

Protection information (PI) is optional and is made up of one or more protection intervals, each made up of 8 bytes associated with a logical block. When PI is active each logical block will have 1, 2, 4, 8, etc protection intervals (i.e. a power of two), interleaved with (and following) the user data to which they refer. Four protection types are defined with protection type 0 being no protection intervals. See the PROTECTION INFORMATION section below for more information.

When the --tape=FM option is given then the SCSI FORMAT MEDIUM command is sent to the DEVICE. FORMAT MEDIUM is defined in the SSC documents at T10 and prepares a volume for use. That may include partitioning the medium. See the section below on TAPE for more information.

The FORMAT WITH PRESET was added in draft SBC-4 revision 18. A preset pattern, selected by the PRESET IDENTIFIER field (--id=FWPID), is writ? ten to the disk. See the FORMAT PRESETS VPD page (0xb8) for a list of available Format preset identifiers and their associated data.

OPTIONS

Arguments to long options are mandatory for short options as well. The options are arranged in alphabetical order based on the long option name.

-C, --cmplst={0|1}

sets the CMPLST ("complete list") bit in the FORMAT UNIT cdb to 0 or 1. If the value is 0 then the existing GLIST (grown list) is taken into account. If the value is 1 then the existing GLIST is ignored. CMPLST defaults to 1 apart from when the --ffmt=FFMT option's value is non-zero in which case CMPLST de? faults to 0. See the LISTS section below. In most cases this bit should be left at its default value.

-c, --count=COUNT

where COUNT is the number of blocks to be formatted or media to be resized to. Can be used with either --format or --resize. With --format this option need not be given in which case it is assumed to be zero.

With --format the interpretation of COUNT is:

(COUNT > 0) : only format the first COUNT blocks and READ CA? PACITY will report COUNT blocks after format

(COUNT = 0) and block size unchanged : use existing block count

(COUNT = 0) and block size changed : recommended maximum block count for new block size

(COUNT = -1) : use recommended maximum block count

(COUNT < -1) : illegal
With --resize this option must be given and COUNT has this in?
terpretation:
(COUNT > 0) : after resize READ CAPACITY will report COUNT
blocks
(COUNT = 0) : after resize READ CAPACITY will report 0 blocks
(COUNT = -1) : after resize READ CAPACITY will report its max?
imum number of blocks
(COUNT < -1) : illegal
In both cases if the given COUNT exceeds the maximum number of
blocks (for the block size) then the disk reports an error. See
NOTES section below.

-D, --dcrt

this option sets the DCRT bit in the FORMAT UNIT command's pa?
rameter list header. It will "disable certification". Certifica?
tion verifies that blocks are usable during the format process.
Using this option may speed the format but --ffmt=FFMT, if
available, would probably be better. The default action of this
utility (i.e. when this option is not given) is to clear the
DCRT bit thereby requesting "media certification" (also unless
another option needs it, the FOV bit will be cleared). When the
DCRT bit is set, the FOV bit must also be set hence sg_format
does that.
If this option is given twice then certification is enabled by
clearing the DCRT bit and setting the FOV bit. Both these bits
are found in the parameter list associated with the FORMAT UNIT
cdb.

-d, --dry-run

this option will parse the command line, do all the preparation
but bypass the actual FORMAT UNIT, FORMAT WITH PRESET or FORMAT
MEDIUM command. Also if the options would otherwise cause the
logical block size to change, then the MODE SELECT command that
would do that is also bypassed when the dry run option is given.

-e, --early

during a format operation, The default action of this utility is to poll the disk every 60 seconds (or every 10 seconds if FFMT is non-zero) to determine the progress of the format operation until it is finished. When this option is given this utility will exit "early", that is as soon as the format operation has commenced. Then the user can monitor the progress of the ongoing format operation with other utilities (e.g. sg_turs(8) or sg_re?quests(8)). This option and --wait are mutually exclusive.

-t, --ffmt=FFMT

FFMT (fast format) is placed in a field of the same name in the FORMAT UNIT cdb. The field was introduced in SBC-4 revision 10. The default value is 0 which implies the former action which is typically to overwrite all blocks on the DEVICE. That can take a long time (e.g. with hard disks over 10 TB in size that can be days). With FFMT set that time may be reduced to minutes or less. So it is worth trying if it is available.

FFMT has values 1 and 2 for fast format with 3 being reserved currently. These two values include this description: "The de?vice server initializes the medium ... without overwriting the medium (i.e. resources for managing medium access are initial?ized and the medium is not written)". The difference between 1 and 2 concerns read operations on LBAs to which no data has been written to, after the fast format. When FFMT is 1 the read oper?ation should return "unspecified logical block data" and com?plete without error. When FFMT is 2 the read operation may yield check condition status with a sense key set to hardware error, medium error or command aborted. See draft SBC-4 revision 16 section 4.34 for more details.

-b, --fmtmaxlba

This option is only active if it is given together with the --preset=ID option. If so it sets the FMTMAXLBA field in the FORMAT WITH PRESET command.

-f, --fmtpinfo=FPI

sets the FMTPINFO field in the FORMAT UNIT cdb to a value be?
tween 0 and 3. The default value is 0. The FMTPINFO field from
SBC-3 revision 16 is a 2 bit field (bits 7 and 6 of byte 1 in
the cdb). Prior to that revision it was a single bit field (bit
7 of byte 1 in the cdb) and there was an accompanying bit called
RTO_REQ (bit 6 of byte 1 in the cdb). The deprecated options
"--pinfo" and "--rto-req" represent the older usage. This option
should be used in their place. See the PROTECTION INFORMATION
section below for more information.

-F, --format

issue one of the three SCSI "format" commands. In the absence of
the --preset=ID and --tape=FM options, the SCSI FORMAT UNIT com?
mand is issued. These commands will destroy all the data held
on the media. This option is required to change the block size
of a disk. In the absence of the --quick option, the user is
given a 15 second count down to ponder the wisdom of doing this,
during which time control-C (amongst other Unix commands) can be
used to kill this process before it does any damage.

When used three times (or more) the preliminary MODE SENSE and
SELECT commands are bypassed, leaving only the initial INQUIRY
and FORMAT UNIT commands. This is for emergency use (e.g. when
the MODE SENSE/SELECT commands are not working) and cannot
change the logical block size.

See NOTES section for implementation details and EXAMPLES sec?
tion for typical use.

-h, --help

print out the usage information then exit.

-I, --ip-def

sets the default Initialization Pattern. Some disks (SSDs) use
this to flag that a format should fully provision (i.e. asso?
ciate a physical block with every logical block). The same disks
(SSDs) might thin provision if this option is not given. If this

option is given then the --security option cannot be given. Also accepts --ip_def for this option.

-l, --long

the default action of this utility is to assume 32 bit logical block addresses. With 512 byte block size this permits more than 2 terabytes (almost 2 ** 41 bytes) on a single disk. This option selects commands and parameters that allow for 64 bit logical block addresses. Specifically this option sets the "longlba" flag in the MODE SENSE (10) command and uses READ CAPACITY (16) rather than READ CAPACITY (10). If this option is not given and READ CAPACITY (10) or MODE SELECT detects a disk the needs more than 32 bits to represent its logical blocks then it is set in? ternally. This option does not set the LONGLIST bit in the FOR? MAT UNIT command. The LONGLIST bit is set as required depending other parameters (e.g. when '--pie=PIE' is greater than zero).

-M, --mode=MP

MP is a mode page number (0 to 62 inclusive) that will be used for reading and perhaps changing the device logical block size. The default is 1 which is the Read-Write Error Recovery mode page.

Preferably the chosen (or default) mode page should be saveable (i.e. accept the SP bit set in the MODE SELECT command used when the logical block size is being changed). Recent version of this utility will retry a MODE SELECT if the SP=1 variant fails with a sense key of ILLEGAL REQUEST. That retry will use the same MODE SELECT command but with SP=0 .

-P, --pfu=PFU

sets the "Protection Field Usage" field in the parameter block associated with a FORMAT UNIT command to PFU. The default value is 0, the only other defined value currently is 1. See the PRO? TECTION INFORMATION section below for more information.

-q, --pie=PIE

sets the "Protection Interval Exponent" field in the parameter block associated with a FORMAT UNIT command to PIE. The default value is 0. PIE can only be non-zero with protection types 2 and 3. The value of 0 is typical for 512 byte blocks; with 4096 byte blocks a value of 3 may be appropriate (i.e. 8 protection intervals interleaved with 4096 bytes of user data). A device may not support any non-zero values. This field first appeared in SBC-3 revision 18.

-p, --pinfo

this option is deprecated, use the --fmtpinfo=FPI option in? stead. If used, then it sets bit 7 of byte 1 in the FORMAT UNIT cdb and that is equivalent to setting --fmtpinfo=2. [So if --pinfo is used (plus --fmtpinfo=FPI and --pfu=PFU are not given or their arguments are 0) then protection type 1 is selected.]

-x, --poll=PT

where PT is the type of poll used. If PT is 0 then a TEST UNIT READY command is used, otherwise a REQUEST SENSE command is used. The default is currently 0 but this will change to 1 in the near future. See the NOTES sections below.

-E, --preset=ID

this option instructs this utility to issue a SCSI FORMAT WITH PRESET command. The PRESET IDENTIFIER field in that cdb is set to ID. The IMMED field in that cdb is also set unless the --wait option is also given, in which case it is cleared.

-Q, --quick

the default action (i.e. when the option is not given) is to give the user 15 seconds to reconsider doing a format operation on the DEVICE. When this option is given that step (i.e. the 15 second warning period) is skipped.

-r, --resize

rather than format the disk, it can be resized. This means changing the number of blocks on the device reported by the READ CAPACITY command. This option should be used with the --count=COUNT option. The contents of all logical blocks on the media remain unchanged when this option is used. This means that any resize operation can be reversed. This option cannot be used together with either --format or a --size=LB_SZ whose argument is different to the existing block size.

-R, --rto_req

The option is deprecated, use the --fmtpinfo=FPI option instead. If used, then it sets bit 6 of byte 1 in the FORMAT UNIT cdb.

-S, --security

sets the "Security Initialization" (SI) bit in the FORMAT UNIT command's initialization pattern descriptor within the parameter list. According to SBC-3 the default initialization pattern "shall be written using a security erasure write technique". See the NOTES section on the SCSI SANITIZE command. If this option is given then the --ip_def option cannot be given.

-6, --six

Use 6 byte variants of MODE SENSE and MODE SELECT. The default action is to use the 10 byte variants. Some MO drives need this option set when doing a format.

-s, --size=LB_SZ

where LB_SZ is the logical block size (i.e. number of user bytes in each block) to format the device to. The default value is whatever is currently reported by the block descriptor in a MODE SENSE command. If the block size given by this option is differ? ent from the current value then a MODE SELECT command is used to change it prior to the FORMAT UNIT command being started (as recommended in the SBC standards). Some SCSI disks have 512 byte logical blocks by default and allow an alternate logical block size of 4096 bytes. If the given size in unacceptable to the disk, most likely an "Invalid field in parameter list" message will appear in sense data (requires the use of '-v' to decode sense data).

Note that formatting a disk to add or remove protection informa? tion is not regarded as a change to its logical block size so this option should not be used.

-T, --tape=FM

will send a FORMAT MEDIUM command to the DEVICE with its FORMAT field set to FM. This option is used to prepare a tape (i.e. the "medium") in a tape drive for use. Values for FM include 0 to do the "default" format; 1 to partition a volume and 2 to do a de? fault format then partition.

-m, --timeout=SECS

where SECS is the FORMAT UNIT, FORMAT WITH PRESET or FORMAT MEDIUM command timeout in seconds. SECS will only be used if it exceeds the internal timeout which is 20 seconds if the IMMED bit is set and 72000 seconds (20 hours) or higher if the IMMED bit is not set. If the disk size exceeds 4 TB then the timeout value is increased to 144000 seconds (40 hours). And if it is greater than 8 TB then the timeout value is increased to 288000 seconds (80 hours). If the timeout is exceeded then the operat? ing system will typically abort the command. Aborting a command may escalate to a LUN reset (or worse). A timeout may also leave the disk or tape format operation incomplete. And that may re? sult in the disk or tape being in a "format corrupt" state re? quiring another format to remedy the situation. So for various reasons command timeouts are best avoided.

-v, --verbose

increase the level of verbosity, (i.e. debug output). "-vvv" gives a lot more debug output.

-y, --verify

set the VERIFY bit in the FORMAT MEDIUM cdb. The default is that the VERIFY bit is clear. This option is only appropriate for tapes.

-V, --version

print the version string and then exit.

-w, --wait

the default format action is to set the "IMMED" bit in the FOR?

MAT UNIT command's (short) parameter header. If this option (i.e. --wait) is given then the "IMMED" bit is not set. If --wait is given then the FORMAT UNIT, FORMAT WITH PRESET or FOR? MAT MEDIUM command waits until the format operation completes before returning its response. This can be many hours on large disks. See the --timeout=SECS option.

Alternatively this option may be useful when used together with --ffmt=FFMT (and FFMT greater than 0) since the fast format may only be a matter of seconds.

LISTS

The SBC-3 draft (revision 20) defines PLIST, CLIST, DLIST and GLIST in section 4.10 on "Medium defects". Briefly, the PLIST is the "primary" list of manufacturer detected defects, the CLIST ("certification" list) contains those detected during the format operation, the DLIST is a list of defects that can be given to the format operation. The GLIST is the grown list which starts in the format process as CLIST+DLIST and can "grow" later due to automatic reallocation (see the ARRE and AWRE bits in the Read-Write Error Recovery mode page (see sdparm(8))) and use of the SCSI REASSIGN BLOCKS command (see sg_reassign(8)).

By the SBC-3 standard (following draft revision 36) the CLIST and DLIST had been removed, leaving PLIST and GLIST. Only PLIST and GLIST are found in the SBC-4 drafts.

The CMPLST bit (controlled by the --cmplst=0|1 option) determines whether the existing GLIST, when the format operation is invoked, is taken into account. The sg_format utility sets the FOV bit to zero which causes DPRY=0, so the PLIST is taken into account, and DCRT=0, so the CLIST is generated and used during the format process.

The sg_format utility does not permit a user to provide a defect list (i.e. DLIST).

PROTECTION INFORMATION

Protection Information (PI) is additional information held with logical blocks so that an application and/or host bus adapter can check the correctness of those logical blocks. PI is placed in one or more pro?

tection intervals interleaved in each logical block. Each protection interval follows the user data to which it refers. A protection inter? val contains 8 bytes made up of a 2 byte "logical block guard" (CRC), a 2 byte "logical block application guard", and a 4 byte "logical block reference tag". Devices with 512 byte logical block size typically have one protection interval appended, making its logical block data 520 bytes long. Devices with 4096 byte logical block size often have 8 pro? tection intervals spread across its logical block data for a total size of 4160 bytes. Note that for all other purposes the logical block size is considered to be 512 and 4096 bytes respectively.

The SBC-3 standard have added several "protection types" to the PI in? troduced in the SBC-2 standard. SBC-3 defines 4 protection types (types 0 to 3) with protection type 0 meaning no PI is maintained. While a de? vice may support one or more protection types, it can only be formatted with 1 of the 4. To change a device's protection type, it must be re-formatted. For more information see the Protection Information in section 4.21 of draft SBC-4 revision 16.

A device that supports PI information (i.e. supports one or more pro? tection types 1, 2 and 3) sets the "PROTECT" bit in its standard IN? QUIRY response. It also sets the SPT field in the EXTENDED INQUIRY VPD page response to indicate which protection types it supports. Given PROTECT=1 then SPT=0 implies the device supports PI type 1 only, SPT=1 implies the device supports PI types 1 and 2, and various other non-ob? vious mappings up to SPT=7 which implies protection types 1, 2 and 3 are supported. The current protection type of a disk can be found in the "P_TYPE" and "PROT_EN" fields in the response of a READ CAPACITY (16) command (e.g. with the 'sg_readcap --long' utility).

Given that a device supports a particular protection type, a user can then choose to format that disk with that protection type by setting the "FMTPINFO" and "Protection Field Usage" fields in the FORMAT UNIT command. Those fields correspond to the --fmtpinfo=FPI and the --pfu=PFU options in this utility. The list below shows the four pro? tection types followed by the options of this utility needed to select them:

```
  0 : --fmtpinfo=0 --pfu=0
  1 : --fmtpinfo=2 --pfu=0
  2 : --fmtpinfo=3 --pfu=0
  3 : --fmtpinfo=3 --pfu=1
```

The default value of FPI (in --fmtpinfo=FPI) is 0 and the default value of PFU (in --pfu=PFU) is 0. So if neither --fmtpinfo=FPI nor --pfu=PFU are given then protection type 0 (i.e. no protection information) is chosen.

NOTES

After a format that changes the logical block size or the number of logical blocks on a disk, the operating system may need to be told to re-initialize its setting for that disk. In Linux that can be done with:

```
echo 1 > /sys/block/sd{letter(s)}/device/rescan
```

where "letter(s)" will be between 'a' and 'zzz'. The lsscsi utility in Linux can be used to check the various namings of a disk.

The SBC-2 standard states that the REQUEST SENSE command should be used for obtaining progress indication when the format command is underway. However, tests on a selection of disks shows that TEST UNIT READY com? mands yield progress indications (but not REQUEST SENSE commands). So the current version of this utility defaults to using TEST UNIT READY commands to poll the disk to find out the progress of the format. The --poll=PT option has been added to control this.

When the --format, --preset=ID or --tape=FM option is given without the --wait option then the corresponding SCSI command is issued with the IMMED bit set which causes the SCSI command to return after it has started the format operation. The --early option will cause sg_format to exit at that point. Otherwise the DEVICE is polled every 60 seconds or every 10 seconds if FFMT is non-zero. The poll is with TEST UNIT READY or REQUEST SENSE commands until one reports an "all clear" (i.e. the format operation has completed). Normally these polling commands will result in a progress indicator (expressed as a percentage) being output to the screen. If the user gets bored watching the progress report then sg_format process can be terminated (e.g. with control-C) without affecting the format operation which continues. However a target or device reset (or a power cycle) will probably cause the format to cease and the DEVICE to become "format corrupt".

When the --format (--preset=ID or --tape) and --wait options are both given then this utility may take a long time to return. In this case care should be taken not to send any other SCSI commands to the disk as it may not respond leaving those commands queued behind the active format command. This may cause a timeout in the OS driver (in a lot shorter period than 20 hours applicable to some format operations). This may result in the OS resetting the disk leaving the format operation incomplete. This may leave the disk in a "format corrupt" state requiring another format to remedy the situation. Modern SCSI devices should yield a "not ready" sense key with an additional sense indicating a format is in progress. With older devices the user should take precautions that nothing attempts to access a device while it is being formatted. Unmounting in mounted file systems on a DEVICE prior to calling this utility is strongly advised.

When the block size (i.e. the number of bytes in each block) is changed on a disk two SCSI commands must be sent: a MODE SELECT to change the block size followed by a FORMAT command. If the MODE SELECT command succeeds and the FORMAT fails then the disk may be in a state that the standard calls "format corrupt". A block descriptor in a subsequent MODE SENSE will report the requested new block size while a READ CAPACITY command will report the existing (i.e. previous) block size. Alternatively the READ CAPACITY command may fail, reporting the device is not ready, potentially requiring a format. The solution to this situation is to do a format again (and this time the new block size does not have to be given) or change the block size back to the original size.

The SBC-2 standard states that the block count can be set back to the manufacturer's maximum recommended value in a format or resize operation. This can be done by placing an address of 0xffffffff (or the 64

bit equivalent) in the appropriate block descriptor field to a MODE SE?
LECT command. In signed (two's complement) arithmetic that value corre?
sponds to '-1'. So a --count=-1 causes the block count to be set back
to the manufacturer's maximum recommended value. To see exactly which
SCSI commands are being executed and parameters passed add the "-vvv"
option to the sg_format command line.

The FMTDATA field shown in the FORMAT UNIT cdb does not have a corre?
sponding option in this utility. When set in the cdb it indicates an
additional parameter list will be sent to the DEVICE along with the
cdb. It is set as required, basically when any field in the parameter
list header is set.

Short stroking is a technique to trade off capacity for performance on
hard disks. "Hard" disk is often used to mean a storage device with
spinning platters which contain the user data. Solid State Disk (SSD)
is the newer form of storage device that contains no moving parts. Hard
disk performance is usually highest on the outer tracks (usually the
lower logical block addresses) so by resizing or reformatting a disk to
a smaller capacity, average performance will usually be increased.

Other utilities may be useful in finding information associated with
formatting. These include sg_inq(8) to fetch standard INQUIRY informa?
tion (e.g. the PROTECT bit) and to fetch the EXTENDED INQUIRY VPD page
(e.g. RTO and GRD_CHK bits). The sdparm(8) utility can be used to ac?
cess and potentially change the now obsolete format mode page.

scsiformat is another utility available for formatting SCSI disks with
Linux. It dates from 1997 (most recent update) and may be useful for
disks whose firmware is of that vintage.

The COUNT numeric argument may include a multiplicative suffix or be
given in hexadecimal. See the "NUMERIC ARGUMENTS" section in the
sg3_utils(8) man page.

The SCSI SANITIZE command was introduced in SBC-3 revision 27. It is
closely related to the ATA sanitize disk feature set and can be used to
remove all existing data from a disk. Sanitize is more likely to be im?
plemented on modern disks (including SSDs) than FORMAT UNIT's security

initialization feature (see the --security option) and in some cases much faster.

SSDs that support thin provisioning will typically unmap all logical blocks during a format. The reason is to improve the SSD's endurance. Also thin provisioned formats typically complete faster than fully pro? visioned ones on the same disk (see the --ip_def option). In either case format operations on SSDs tend to be a lot faster than they are on hard disks with spinning media.

TAPE

Tape system use a variant of the FORMAT UNIT command used on disks. Tape systems use the FORMAT MEDIUM command which is simpler with only three fields in the cdb typically used. Apart from sharing the same op? code the cdbs of FORMAT UNIT and FORMAT MEDIUM are quite different. FORMAT MEDIUM's fields are VERIFY, IMMED and FORMAT (with TRANSFER LENGTH always set to 0). The VERIFY bit field is set with the --verify option. The IMMED bit is manipulated by the --wait option in the same way it is for disks; one difference is that if the --poll=PT option is not given then it defaults to PT of 1 which means the poll is done with REQUEST SENSE commands.

The argument given to the --tape=FM option is used to set the FORMAT field. FM can take values from "-1" to "15" where "-1" (the default) means don't do a tape format; value "8" to "15" are for vendor specific formats. The --early option may also be used to set the IMMED bit and then exit this utility (rather than poll periodically until it is fin? ished). In this case the tape drive will still be busy doing the format for some time but, according to T10, should still respond in full to the INQUIRY and REPORT LUNS commands. Other commands (including REQUEST SENSE) should yield a "not ready" sense key with an additional sense code of "Logical unit not ready, format in progress". Additionally RE? QUEST SENSE should contain a progress indication in its sense data.

When FM is 1 or 2 then the settings in the Medium partition mode page control the partitioning. That mode page can be viewed and modified with the sdparm utility.

Prior to invoking this utility the tape may need to be positioned to the beginning of partition 0. In Linux that can typically be done with the mt utility (e.g. 'mt -f /dev/st0 rewind').

EXAMPLES

These examples use Linux device names. For suitable device names in other supported Operating Systems see the sg3_utils(8) man page.

In the first example below simply find out the existing block count and size derived from two sources: a block descriptor in a MODE SELECT com? mand response and from the response of a READ CAPACITY commands. No changes are made:

```
# sg_format /dev/sdm
```

Now a simple format, leaving the block count and size as they were pre? viously. The FORMAT UNIT command is executed in IMMED mode and the de? vice is polled every 60 seconds to print out a progress indication:

```
# sg_format --format /dev/sdm
```

Now the same format, but waiting (passively) until the format operation is complete:

```
# sg_format --format --wait /dev/sdm
```

Next is a format in which the block size is changed to 520 bytes and the block count is set to the manufacturer's maximum value (for that block size). Note, not all disks support changing the block size:

```
# sg_format --format --size=520 /dev/sdm
```

Now a resize operation so that only the first 0x10000 (65536) blocks on a disk are accessible. The remaining blocks remain unaltered.

```
# sg_format --resize --count=0x10000 /dev/sdm
```

Now resize the disk back to its normal (maximum) block count:

```
# sg_format --resize --count=-1 /dev/sdm
```

One reason to format a SCSI disk is to add protection information. First check which protection types are supported by a disk (by checking the SPT field in the Extended inquiry VPD page together with the Pro? tect bit in the standard inquiry response):

```
# sg_vpd -p ei -l /dev/sdb
extended INQUIRY data VPD page:
     ACTIVATE_MICROCODE=0
     SPT=1 [protection types 1 and 2 supported]
     ....
```

Format with type 1 protection:

```
   # sg_format --format --fmtpinfo=2 /dev/sdm
```

After a successful format with type 1 protection, READ CAPACITY(16) should show something like this:

```
   # sg_readcap -l /dev/sdm
   Read Capacity results:
      Protection:  prot_en=1,  p_type=0, p_i_exponent=0 [type 1 protec?
tion]
      Logical block provisioning: lbpme=0, lbprz=0
      ....
```

To format with type 3 protection:

```
   # sg_format --format --fmtpinfo=3 --pfu=1 /dev/sdm
```

For the disk shown above this will probably fail because the Extended inquiry VPD page showed only types 1 and 2 protection are supported.

Here are examples of using fast format (FFMT field in FORMAT UNIT cdb) to quickly switch between 512 and 4096 byte logical block size. Assume disk starts with 4096 byte logical block size and all important data has been backed up.

```
   # sg_format --format --ffmt=1 --size=512 /dev/sdd
```

Now /dev/sdd should have 512 byte logical block size. And to switch it back:

```
   # sg_format --format --ffmt=1 --size=4096 /dev/sdd
```

Since fast formats can be very quick (a matter of seconds) using the --wait option may be appropriate.

And to use the Format with preset command this invocation could be used:

```
   # sg_format --preset=1 --fmtmaxlba /dev/sdd
```

The FORMAT PRESETS VPD page (0xb8) should be consulted to check that Preset identifier 0x1 is there and has the expected format (i.e. "de?fault host aware zoned block device model with 512 bytes of user data in each logical block"). That VPD page can be viewed with the sg_vpd utility.

EXIT STATUS

The exit status of sg_format is 0 when it is successful. Otherwise see the sg3_utils(8) man page. Unless the --wait option is given, the exit status may not reflect the success of otherwise of the format. Using sg_turs(8) and sg_readcap(8) after the format operation may be wise. The Unix convention is that "no news is good news" but that can be a bit unnerving after an operation like format, especially if it finishes quickly (i.e. before the first progress poll is sent). Giving the --verbose option once should supply enough additional output to settle those nerves.

AUTHORS

Written by Grant Grundler, James Bottomley and Douglas Gilbert.

REPORTING BUGS

Report bugs to <dgilbert at interlog dot com>.

COPYRIGHT

SEE ALSO

sg_turs(8), sg_requests(8), sg_inq(8), sg_modes(8), sg_vpd(8), sg_reas?sign(8), sg_readcap(8), sg3_utils(8), sg_sanitize(8) [all in sg3_utils], lsscsi(8), mt(mt-st), sdparm(8), scsiformat (old), hd?parm(8)

sg3_utils-1.45 January 2020 SG_FORMAT(8)