



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

Rocky Enterprise Linux 9.2 Manual Pages on command 'sg_ident.8'

\$ man sg_ident.8

SG_IDENT(8) SG3_UTILS SG_IDENT(8)

NAME

sg_ident - send SCSI REPORT/SET IDENTIFYING INFORMATION command

SYNOPSIS

sg_ident [--ascii] [--clear] [--help] [--itype=IT] [--raw] [--set] [--verbose] [--version]

DEVICE

DESCRIPTION

Send a SCSI REPORT IDENTIFYING INFORMATION or SET IDENTIFYING INFORMATION command to DE?

VICE. Prior to SPC-4 (revision 7) these commands were called REPORT DEVICE IDENTIFIER and

SET DEVICE IDENTIFIER respectively. SCSI devices that support these two commands allow users to write (set) identifying information and report it back at some later time. The information is persistent (i.e. stored on some non-volatile medium within the SCSI device that will survive a power outage).

Typically the space allocated for the information is limited: SPC-4 (revision 7) states that for information type 0, the minimum length is 64 bytes and the maximum is 512 bytes. For other information types (1 to 126 inclusive) the maximum length is 256 bytes. Also information types 1 to 126 (inclusive) should contain a null terminated UTF-8 string. The author has seen older disks that only support 16 bytes.

The default action when no options are given is to invoke the Report Identifying Information command with the information type defaulting to zero. Error reports are sent to stderr. By default the information is shown in ASCII-HEX (up to 16 bytes per line) with an ASCII representation to the right with dots replacing non printable characters.

OPTIONS

Arguments to long options are mandatory for short options as well.

-A, --ascii

invokes the Report Identifying Information command and if anything is found pretends it as ASCII (or UTF-8 which is locale dependent) and prints the information to stdout.

-C, --clear

invokes the Set Identifying Information command with an information length of zero. This has the effect of clearing the existing information.

-h, --help

output the usage message then exit.

-i, --itype=IT

where IT is the information type. Defaults to zero. The maximum value is 127 which is special and cannot be used with --set or --clear. The information type of 127 (if supported) causes the REPORT IDENTIFYING INFORMATION command to respond with a list of available information types and their maximum lengths in bytes. The odd numbered information types between 3 and 125 (inclusive) are not to be used (as they clash with the SCC-2 standard).

-r, --raw

invokes the Report Identifying information command and if anything is found sends the information (which may be binary) to stdout. Nothing else is sent to stdout however error reports, if any, are sent to stderr.

-S, --set

first reads stdin until an EOF is detected then invokes the Set Identifying Information command to set what has been fetched from stdin as the information. The amount of data read must be between 1 and 512 bytes length (inclusive).

-v, --verbose

increase the level of verbosity, (i.e. debug output).

-V, --version

print the version string and then exit.

This utility permits users to write their own identifying information to their SCSI devices. There are several other types of descriptors (or designators) that the user cannot change. These include the SCSI INQUIRY command with its standard vendor and product identification strings and the product revision level; plus the large amount of information

provided by the "Device Identification" VPD page (see `sg_vpd`). There is also the `READ ME? DIA SERIAL NUMBER` command (see `sg_rmsn`). The MMC-4 command set for CD and DVDs has a "me? dia serial number" feature (0x109) [and a "logical unit serial number" feature]. These can be viewed with the `sg_get_config` utility.

EXAMPLES

First, to see if there is an existing information whose format is unknown (for information type 0), use no options:

```
# sg_ident /dev/sdb
00 31 32 33 34 35 36 37 38 39 30      1234567890
```

If it is ASCII then it can be printed as such:

```
# sg_ident --ascii /dev/sdb
1234567890
```

The information can be copied to a file, cleared and then re-asserted with this sequence:

```
# sg_ident --raw /dev/sdb > t
# sg_ident --clear /dev/sdb
# cat t | sg_ident --set /dev/sdb
```

EXIT STATUS

The exit status of `sg_ident` is 0 when it is successful. Otherwise see the `sg3_utils(8)` man page.

AUTHORS

Written by Douglas Gilbert.

REPORTING BUGS

Report bugs to <dgilbert at interlog dot com>.

COPYRIGHT

Copyright ? 2005-2018 Douglas Gilbert

This software is distributed under a FreeBSD license. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

SEE ALSO

`sg_vpd(sg3_utils)`, `sg_rmsn(sg3_utils)`, `sg_get_config(sg3_utils)`