

**NAME**

**efivar** - UEFI environment variable interaction

**SYNOPSIS**

```
efivar [-abDDHILNpqRtuw] [-n name] [-f file] [--append] [--ascii] [--attributes] [--binary] [--delete]
  [--device-path] [--fromfile file] [--guid] [--hex] [--list-guids] [--list] [--load-option] [--name name]
  [--no-name] [--print] [--print-decimal] [--quiet] [--raw-guid] [--utf8] [--write]
```

**DESCRIPTION**

This program manages "Unified Extensible Firmware Interface" (UEFI) environment variables. UEFI variables have three parts: A namespace, a name and a value. The namespace is a GUID that is self assigned by the group defining the variables. The name is a Unicode name for the variable. The value is binary data. All Unicode data is presented to the user as UTF-8.

The following options are available:

- |  |  |
|--|--|
| <b>-n <i>name</i> --name <i>name</i></b>       | Specify the name of the variable to operate on. The <i>name</i> argument is the GUID of the variable, followed by a dash, followed by the UEFI variable name. The GUID may be in numeric format, or may be one of the well known symbolic names (see <b>--list-guids</b> for a complete list).   |
| <b>-f <i>file</i> --fromfile <i>file</i></b>   | When writing or appending to a variable, take the data for the variable's value from <i>file</i> instead of from the command line. This flag implies <b>--write</b> unless the <b>--append</b> or <b>--print</b> flags are given. This behavior is not well understood and is currently unimplemented for writes. When <b>--print</b> is specified, the contents of the file are used as the value to print using any other specified flags. This is used primarily for testing purposes for more complicated variable decoding. |
| <b>-a --append</b>                             | Append the specified value to the UEFI variable rather than replacing it.  |
| <b>-t <i>attr</i> --attributes <i>attr</i></b> | Specify, in hexadecimal, the attributes for this variable. See section 7.2 (GetVariable subsection, Related Definitions) of the UEFI Specification for hex values to use.  |
| <b>-A --ascii</b>                              | Display the variable data as modified ASCII: All printable characters are printed, while unprintable characters are rendered   |

as a two-digit hexadecimal number preceded by a % character.

- b --binary** Display the variable data as binary data. Usually will be used with the **-N** or **--no-name** flag. Useful in scripts.
- D --delete** Delete the specified variable. May not be used with either the **--write** or the **--append** flags. No *value* may be specified.
- d --device --device-path** Interpret the variables printed as UEFI device paths and print the UEFI standard string representation.
- g --guid** Convert GUIDs to names if they are known (and show them in **--list-guids**).
- H --hex** List variable data as a hex dump.
- L --list-guids** Lists the well known GUIDs. The names listed here may be used in place of the numeric GUID values. These names will replace the numeric GUID values unless the **--raw-guid** flag is specified.
- l --list** List all the variables. If the **--print** flag is also listed, their values will be displayed.
- load-option** Decode the variable as if it were a UEFI Boot Option, including information about what device and/or paths the UEFI DevicePaths decode to.
- N --no-name** Do not display the variable name.
- p --print** Print the value of the variable.
- q --quiet** When an error occurs, exit with a non-zero value without outputting any error messages. Otherwise, produce the normal output and exit with a zero status.
- R --raw-guid** Do not substitute well known names for GUID numeric values in output.
- u --utf8** Treat the value of the variable as UCS2 and convert it to UTF8

and print the result.

**-w --write**

Write (replace) the variable specified with the value specified from standard input. No command line option to do this is available since UEFI variables are binary structures rather than strings. `echo(1)` **-n** can be used to specify simple strings.

*name*

Display the *name* environment variable.

## COMPATIBILITY

The **efivar** program is intended to be compatible (strict superset) with a program of the same name included in the Red Hat libefivar package, but the **-d** and **--print-decimal** flags are not implemented and never will be.

The **-d** flag is short for **--device-path**.

## SEE ALSO

Appendix A of the UEFI specification has the format for GUIDs. All GUIDs "Globally Unique Identifiers" have the format described in RFC 4122.

## HISTORY

The **efivar** utility first appeared in FreeBSD 11.1.