

NAME

`llvm-readobj` - LLVM Object Reader

SYNOPSIS

`llvm-readobj` [*options*] [*input...*]

DESCRIPTION

The `llvm-readobj` tool displays low-level format-specific information about one or more object files.

If **input** is "-", `llvm-readobj` reads from standard input. Otherwise, it will read from the specified **filenames**.

DIFFERENCES TO LLVM-READELF

`llvm-readelf` is an alias for the `llvm-readobj` tool with a slightly different command-line interface and output that is GNU compatible. Following is a list of differences between `llvm-readelf` and `llvm-readobj`:

- ⊕ `llvm-readelf` uses *GNU* for the `--elf-output-style` option by default. `llvm-readobj` uses *LLVM*.
- ⊕ `llvm-readelf` allows single-letter grouped flags (e.g. `llvm-readelf -SW` is the same as `llvm-readelf -S -W`). `llvm-readobj` does not allow grouping.
- ⊕ `llvm-readelf` provides `-s` as an alias for `--symbols`, for GNU `readelf` compatibility, whereas it is an alias for `--section-headers` in `llvm-readobj`.
- ⊕ `llvm-readobj` provides `-t` as an alias for `--symbols`. `llvm-readelf` does not.
- ⊕ `llvm-readobj` provides `--sr`, `--sd`, `--st` and `--dt` as aliases for `--section-relocations`, `--section-data`, `--section-symbols` and `--dyn-symbols` respectively. `llvm-readelf` does not provide these aliases, to avoid conflicting with grouped flags.

GENERAL AND MULTI-FORMAT OPTIONS

These options are applicable to more than one file format, or are unrelated to file formats.

--all

Equivalent to specifying all the main display options relevant to the file format.

--addrsig

Display the address-significance table.

--expand-relocs

When used with *--relocs*, display each relocation in an expanded multi-line format.

--file-header, -h

Display file headers.

--headers, -e

Equivalent to setting: *--file-header*, *--program-headers*, and *--sections*.

--help

Display a summary of command line options.

--hex-dump=<section[,section,...]>, -x

Display the specified section(s) as hexadecimal bytes. **section** may be a section index or section name.

--needed-libs

Display the needed libraries.

--relocations, --relocs, -r

Display the relocation entries in the file.

--sections, --section-headers, -S

Display all sections.

--section-data, --sd

When used with *--sections*, display section data for each section shown. This option has no effect for GNU style output.

--section-relocations, --sr

When used with *--sections*, display relocations for each section shown. This option has no effect for GNU style output.

--section-symbols, --st

When used with *--sections*, display symbols for each section shown. This option has no effect for GNU style output.

--sort-symbols=<sort_key[,sort_key]>

Specify the keys to sort symbols before displaying syntab. Valid values for *sort_key* are **name** and **type**.

--stackmap

Display contents of the stackmap section.

--string-dump=<section[,section,...]>, -p

Display the specified section(s) as a list of strings. **section** may be a section index or section name.

--string-table

Display contents of the string table.

--symbols, --syms, -s

Display the symbol table.

--unwind, -u

Display unwind information.

--version

Display the version of the **llvm-readobj** executable.

@<FILE>

Read command-line options from response file *<FILE>*.

ELF SPECIFIC OPTIONS

The following options are implemented only for the ELF file format.

--arch-specific, -A

Display architecture-specific information, e.g. the ARM attributes section on ARM.

--bb-addr-map

Display the contents of the basic block address map section(s), which contain the address of each function, along with the relative offset of each basic block.

--demangle, -C

Display demangled symbol names in the output.

--dependent-libraries

Display the dependent libraries section.

--dyn-relocations

Display the dynamic relocation entries.

--dyn-symbols, --dyn-syms, --dt

Display the dynamic symbol table.

--dynamic-table, --dynamic, -d

Display the dynamic table.

--cg-profile

Display the callgraph profile section.

--histogram, -I

Display a bucket list histogram for dynamic symbol hash tables.

--elf-linker-options

Display the linker options section.

--elf-output-style=<value>

Format ELF information in the specified style. Valid options are **LLVM**, **GNU**, and **JSON**. **LLVM** output (the default) is an expanded and structured format. **GNU** output mimics the equivalent GNU **readelf** output. **JSON** is JSON formatted output intended for machine consumption.

--section-groups, -g

Display section groups.

--gnu-hash-table

Display the GNU hash table for dynamic symbols.

--hash-symbols

Display the expanded hash table with dynamic symbol data.

--hash-table

Display the hash table for dynamic symbols.

--notes, -n

Display all notes.

--pretty-print

When used with *--elf-output-style*, JSON output will be formatted in a more readable format.

--program-headers, --segments, -l

Display the program headers.

--raw-relr

Do not decode relocations in RELR relocation sections when displaying them.

--section-mapping

Display the section to segment mapping.

--stack-sizes

Display the contents of the stack sizes section(s), i.e. pairs of function names and the size of their stack frames. Currently only implemented for GNU style output.

--version-info, -V

Display version sections.

MACH-O SPECIFIC OPTIONS

The following options are implemented only for the Mach-O file format.

--macho-data-in-code

Display the Data in Code command.

--macho-dsymtab

Display the Dsymtab command.

--macho-indirect-symbols

Display indirect symbols.

--macho-linker-options

Display the Mach-O-specific linker options.

--macho-segment

Display the Segment command.

--macho-version-min

Display the version min command.

PE/COFF SPECIFIC OPTIONS

The following options are implemented only for the PE/COFF file format.

--codeview

Display CodeView debug information.

--codeview-ghash

Enable global hashing for CodeView type stream de-duplication.

--codeview-merged-types

Display the merged CodeView type stream.

--codeview-subsection-bytes

Dump raw contents of CodeView debug sections and records.

--coff-basereloc

Display the .reloc section.

--coff-debug-directory

Display the debug directory.

--coff-tls-directory

Display the TLS directory.

--coff-directives

Display the .directve section.

--coff-exports

Display the export table.

--coff-imports

Display the import table.

--coff-load-config

Display the load config.

--coff-resources

Display the .rsrc section.

XCOFF SPECIFIC OPTIONS

The following options are implemented only for the XCOFF file format.

--auxiliary-header

Display XCOFF Auxiliary header.

--exception-section

Display XCOFF exception section entries.

--loader-section-header

Display XCOFF loader section header.

--loader-section-symbols

Display symbol table of loader section.

--loader-section-relocations

Display relocation entries of loader section.

EXIT STATUS

llvm-readobj returns 0 under normal operation. It returns a non-zero exit code if there were any errors.

SEE ALSO

llvm-nm(1), **llvm-objdump(1)**, **llvm-readelf(1)**

AUTHOR

Maintained by the LLVM Team (<https://llvm.org/>).

COPYRIGHT

2003-2023, LLVM Project