NAME

ndisasm − the Netwide Disassembler, an 80x86 binary file disassembler

SYNOPSIS

ndisasm [**−o** origin] [**−s** sync−point [...]] [**−a** | **−i**] [**−b** bits] [**−u**] [**−e** hdrlen] [**−p** vendor] [**−k** offset,length [...]] infile

DESCRIPTION

The **ndisasm** command generates a disassembly listing of the binary file infile and directs it to stdout.

OPTIONS

−h

Causes **ndisasm** to exit immediately, after giving a summary of its invocation options.

−r|**−v**

Causes **ndisasm** to exit immediately, after displaying its version number.

−o *origin*

Specifies the notional load address for the file. This option causes **ndisasm** to get the addresses it lists down the left hand margin, and the target addresses of PC−relative jumps and calls, right.

−s *sync−point*

Manually specifies a synchronisation address, such that **ndisasm** will not output any machine instruction which encompasses bytes on both sides of the address. Hence the instruction which starts at that address will be correctly disassembled.

−e *hdrlen*

Specifies a number of bytes to discard from the beginning of the file before starting disassembly. This does not count towards the calculation of the disassembly offset: the first *disassembled* instruction will be shown starting at the given load address.

−k *offset,length*

Specifies that *length* bytes, starting from disassembly offset *offset*, should be skipped over without generating any output. The skipped bytes still count towards the calculation of the disassembly offset.

−a|**−i**

Enables automatic (or intelligent) sync mode, in which **ndisasm** will attempt to guess where synchronisation should be performed, by means of examining the target addresses of the relative jumps and calls it disassembles.

−b *bits*

Specifies 16−, 32− or 64−bit mode. The default is 16−bit mode.

−u

Specifies 32−bit mode, more compactly than using ‘−b 32’.

−p *vendor*

Prefers instructions as defined by *vendor* in case of a conflict. Known *vendor* names include **intel**, **amd**, **cyrix**, and **idt**. The default is **intel**.

RESTRICTIONS

ndisasm only disassembles binary files: it has no understanding of the header information present in object or executable files. If you want to disassemble an object file, you should probably be using **objdump**(1).

Auto−sync mode won’t necessarily cure all your synchronisation problems: a sync marker can only be placed automatically if a jump or call instruction is found to refer to it *before* **ndisasm** actually disassembles that part of the code. Also, if spurious jumps or calls result from disassembling non−machine−code data, sync markers may get placed in strange places. Feel free to turn auto−sync off and go back to doing it manually if necessary.

SEE ALSO

objdump(1)