

NAME

ctfdump – dumps CTF content

SYNOPSIS

ctfdump [-adfhlsStw] [-u *ctffile*] *file*...

DESCRIPTION

The **ctfdump** utility symbolically dumps the contents of the **CTF** (Compact C Type Format) data from the specified file(s). By default, the contents of the **CTF** header, label, data, function, and type information sections are shown. While the string table is not included by default, the strings it contains are seen as part of the output for the other sections. The **-a** option can be used to add the string table to the display of these defaults. Alternatively, options are provided to select the specific information to display.

OPTIONS

The following options are supported:

- | | |
|--------------------------|---|
| -a | Display the information from all sections of the CTF data. This consists of the CTF header, label, data, function, type information, and string table sections. |
| -d | Display CTF data definitions. |
| -f | Display CTF function definitions. |
| -h | Display the contents of the CTF header. |
| -l | Display CTF label definitions. |
| -s | Display the CTF string table. |
| -S | Display CTF statistics. This is a specialized option, of interest primarily to CTF developers. |
| -t | Display CTF type definitions. |
| -u <i>ctffile</i> | Copy the uncompressed CTF data to the file given by <i>ctffile</i> . |
| -w | Wide type definition output. When displaying type definitions, ctfdump sets a default width for the name column wide enough to accommodate most names while keeping the average line length under 80 columns. Longer names will disturb the alignment of the attribute column that follows. The -w option disables this limit, and makes the name column wide enough to fit the longest name. The result may be more readable when many long names are present. |

OPERANDS

The following operands are supported.

file The file for which the **CTF** data symbolically displayed.

EXIT STATUS

The following exit values are returned:

0 Successful completion.

> 0 An error occurred.

NOTES

Archive files, produced by **ar(1)**, can be inspected with **ctfdump**. In this case, each object within the archive is processed using the options supplied.

Objects that have associated ancillary objects can be inspected with the **ctfdump** utility. If *file* has associated ancillary objects, and those ancillary objects are available in the same directory as *file*, **ctfdump** will transparently read those ancillary objects to obtain any sections absent from the primary object which are needed to display the CTF. Support for reading ancillary objects is limited to plain objects. The **ctfdump** utility will not access ancillary objects for objects found in an archive. See **ld(1)**, and **ctf(7)**.

In addition to native Solaris **CTF**, the **ctfdump** utility is also able to dump the **GNU CTF** produced by the **-gctf** option to the **gcc** compilers. **GNU CTF** is a different variant that is not directly compatible with the native **CTF** used on Oracle Solaris. However, **ctfdump** can be used to inspect their contents, and the Solaris **CTF** implementation is able to translate them to native **CTF** as part of the merge process. See **ctf(7)**.

EXAMPLES

Example 1 Default Information

Dump the **CTF** header, label, data, function, and type information sections of an object.

```
% ctfdump filename
```

Example 2 Default Information + String Table

Add the string table section to the information displayed by the previous example.

```
% ctfdump -a filename
```

Example 3 Full Information

Add the **CTF** statistics to the information displayed by the previous example. This shows everything that **ctfdump** is able to display.

```
% ctfdump -a -S filename
```

Example 4 CTF header

Display the **CTF** header for an object.

% **ctfdump -h** *filename*

Example 5 CTF header + Functions

Display the **CTF** header and function section for an object.

% **ctfdump -h -f** *filename*

ATTRIBUTES

See **attributes(7)** for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	developer/base-developer-utilities
Interface Stability	Committed

HISTORY

Support for reading data from ancillary objects was added in the Oracle Solaris 11.4.97 release.

The **-a** and **-w** options were added, and the string table and statistics information dropped from the default output, in Oracle Solaris 11.4.90.

Support for dumping **GNU CTF** from relocatable objects created by the **gcc** compilers was added in the Oracle Solaris 11.4.84 release.

The **CTF** utilities, including **ctfdump**, have been used in the construction of Solaris since Sun Solaris 9. They were added as system utilities in Oracle Solaris 11.4.75.

SEE ALSO

ar(1), **ctfconvert(1)**, **ctfmerge(1)**, **ctf(7)**, **ld(1)**