



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 'File::MimeInfo::Magic.3pm'

\$ man File::MimeInfo::Magic.3pm

File::MimeInfo::Magic(3pm) User Contributed Perl Documentation File::MimeInfo::Magic(3pm)

NAME

File::MimeInfo::Magic - Determine file type with magic

SYNOPSIS

```
use File::MimeInfo::Magic;

my $mime_type = mimetype($file);
```

DESCRIPTION

This module inherits from File::MimeInfo, it is transparent to its functions but adds support for the freedesktop magic file.

Magic data is hashed when you need it for the first time. If you want to force hashing earlier use the "rehash()" function.

EXPORT

The method "mimetype" is exported by default. The methods "magic", "inodetype", "globs" and "describe" can be exported on demand.

METHODS

See also File::MimeInfo for methods that are inherited.

"mimetype(\$file)"

Returns a mime-type string for \$file, returns undef on failure.

This method bundles "inodetype()", "globs()" and "magic()".

Magic rules with an priority of 80 and higher are checked before "globs()" is called, all other magic rules afterwards.

If this doesn't work the file is read and the mime-type defaults to 'text/plain' or to 'application/octet-stream' when the first ten chars of the file match ascii control

chars (white spaces excluded). If the file doesn't exist or isn't readable "undef" is returned.

If \$file is an object reference only "magic" and the default method are used. See below for details.

"magic(\$file)"

Returns a mime-type string for \$file based on the magic rules, returns undef on failure.

\$file can be an object reference, in that case it is supposed to have a "seek()" and a "read()" method. This allows you for example to determine the mimetype of data in memory by using IO::Scalar.

Be aware that when using a filehandle or an "IO::" object you need to set the ":utf8" binmode yourself if appropriate.

"rehash()"

Rehash the data files. Glob and magic information is preparsed when this method is called.

If you want to by-pass the XDG basedir system you can specify your database directories by setting @File::MimeInfo::DIRS. But normally it is better to change the XDG basedir environment variables.

"default"

"describe"

"extensions"

"globs"

"inodetype"

These routines are imported from File::MimeInfo.

SEE ALSO

File::MimeInfo

LIMITATIONS

Only word sizes of 1, 2 or 4 are supported. Any other word size is ignored and will cause a warning.

AUTHOR

Jaap Karssenber <pardus@cpan.org> Maintained by Michiel Beijen <mb@x14.nl>

COPYRIGHT

Copyright (c) 2003, 2012 Jaap G Karssenber. All rights reserved. This program is free

software; you can redistribute it and/or modify it under the same terms as Perl itself.

perl v5.32.1

2022-01-10

File::MimeInfo::Magic(3pm)