



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 'APR::Const.3pm'

\$ man APR::Const.3pm

libapache2-mod-perl2-2.0.12::doUserContrilibapache2-mod-perl2-2.0.12::docs::api::APR::Const(3pm)

NAME

APR::Const - Perl Interface for APR Constants

Synopsis

```
# make the constants available but don't import them
use APR::Const -compile => qw(constant names ...);

# w/o the => syntax sugar
use APR::Const ("-compile", qw(constant names ...));

# compile and import the constants
use APR::Const qw(constant names ...);
```

Description

This package contains constants specific to "APR" features.

Refer to "the Apache2::Const description section" for more information.

Constants

":common"

```
use APR::Const -compile => qw(:common);
```

The ":common" group is for XXX constants.

"APR::Const::SUCCESS"

since: 2.0.00

":error"

```
use APR::Const -compile => qw(:error);
```

The ":error" group is for XXX constants.

"APR::Const::EABOVEROOT"

since: 2.0.00

"APR::Const::EABSOLUTE"

since: 2.0.00

"APR::Const::EACCES"

since: 2.0.00

Due to possible variants in conditions matching "EACCES", for checking error codes against this you most likely want to use the "APR::Status::is_EACCES" function instead.

"APR::Const::EAGAIN"

since: 2.0.00

The error Resource temporarily unavailable, may be returned by many different system calls, especially IO calls. Most likely you want to use the "APR::Status::is_EAGAIN" function instead.

"APR::Const::EBADDATE"

since: 2.0.00

"APR::Const::EBADF"

since: 2.0.00

"APR::Const::EBADIP"

since: 2.0.00

"APR::Const::EBADMASK"

since: 2.0.00

"APR::Const::EBADPATH"

since: 2.0.00

"APR::Const::EBUSY"

since: 2.0.00

"APR::Const::ECONNABORTED"

since: 2.0.00

Due to possible variants in conditions matching "ECONNABORTED", for checking error codes against this you most likely want to use the "APR::Status::is_ECONNABORTED" function instead.

"APR::Const::ECONNREFUSED"

since: 2.0.00

"APR::Const::ECONNRESET"

since: 2.0.00

Due to possible variants in conditions matching "ECONNRESET", for checking error codes against this you most likely want to use the "APR::Status::is_ECONNRESET" function instead.

"APR::Const::EDSOOPEN"

since: 2.0.00

"APR::Const::EEXIST"

since: 2.0.00

"APR::Const::EFTYPE"

since: 2.0.00

"APR::Const::EGENERAL"

since: 2.0.00

"APR::Const::EHOSTUNREACH"

since: 2.0.00

"APR::Const::EINCOMPLETE"

since: 2.0.00

"APR::Const::EINIT"

since: 2.0.00

"APR::Const::EINPROGRESS"

since: 2.0.00

"APR::Const::EINTR"

since: 2.0.00

"APR::Const::EINVAL"

since: 2.0.00

"APR::Const::EINVALSOCK"

since: 2.0.00

"APR::Const::EMFILE"

since: 2.0.00

"APR::Const::EMISMATCH"

since: 2.0.00

"APR::Const::ENAMETOOLONG"

since: 2.0.00

"APR::Const::END"

since: 2.0.00

"APR::Const::ENETUNREACH"

since: 2.0.00

"APR::Const::ENFILE"

since: 2.0.00

"APR::Const::ENODIR"

since: 2.0.00

"APR::Const::ENOENT"

since: 2.0.00

Due to possible variants in conditions matching "ENOENT", for checking error codes against this you most likely want to use the "APR::Status::is_ENOENT" function instead.

"APR::Const::ENOLOCK"

since: 2.0.00

"APR::Const::ENOMEM"

since: 2.0.00

"APR::Const::ENOPOLL"

since: 2.0.00

"APR::Const::ENOPOOL"

since: 2.0.00

"APR::Const::ENOPROC"

since: 2.0.00

"APR::Const::ENOSHMAVAIL"

since: 2.0.00

"APR::Const::ENOSOCKET"

since: 2.0.00

"APR::Const::ENOSPC"

since: 2.0.00

"APR::Const::ENOSTAT"

since: 2.0.00

"APR::Const::ENOTDIR"

since: 2.0.00

"APR::Const::ENOTEMPTY"

since: 2.0.00

"APR::Const::ENOTHDKEY"

since: 2.0.00

"APR::Const::ENOTHREAD"

since: 2.0.00

"APR::Const::ENOTIME"

since: 2.0.00

"APR::Const::ENOTIMPL"

Something is not implemented

since: 2.0.00

"APR::Const::ENOTSOCK"

since: 2.0.00

"APR::Const::EOF"

since: 2.0.00

Due to possible variants in conditions matching "EOF", for checking error codes against this you most likely want to use the "APR::Status::is_EOF" function instead.

"APR::Const::EPATHWILD"

since: 2.0.00

"APR::Const::EPIPE"

since: 2.0.00

"APR::Const::EPROC_UNKNOWN"

since: 2.0.00

"APR::Const::ERELATIVE"

since: 2.0.00

"APR::Const::ESPIPE"

since: 2.0.00

"APR::Const::ESYMNOTFOUND"

since: 2.0.00

"APR::Const::ETIMEDOUT"

since: 2.0.00

"APR::Const::EXDEV"

since: 2.0.00

":fopen"

use APR::Const -compile => qw(:fopen);

The ":fopen" group is for XXX constants.

"APR::Const::FOPEN_BINARY"

since: 2.0.00

"APR::Const::FOPEN_BUFFERED"

since: 2.0.00

"APR::Const::FOPEN_CREATE"

since: 2.0.00

"APR::Const::FOPEN_DELONCLOSE"

since: 2.0.00

"APR::Const::FOPEN_EXCL"

since: 2.0.00

"APR::Const::FOPEN_PEND"

since: 2.0.00

"APR::Const::FOPEN_READ"

since: 2.0.00

"APR::Const::FOPEN_TRUNCATE"

since: 2.0.00

"APR::Const::FOPEN_WRITE"

since: 2.0.00

":filepath"

use APR::Const -compile => qw(:filepath);

The ":filepath" group is for XXX constants.

"APR::Const::FILEPATH_ENCODING_LOCALE"

since: 2.0.00

"APR::Const::FILEPATH_ENCODING_UNKNOWN"

since: 2.0.00

"APR::Const::FILEPATH_ENCODING_UTF8"

since: 2.0.00

"APR::Const::FILEPATH_NATIVE"

since: 2.0.00

"APR::Const::FILEPATH_NOTABOVEROOT"

since: 2.0.00

"APR::Const::FILEPATH_NOTABSOLUTE"

since: 2.0.00

"APR::Const::FILEPATH_NOTRELATIVE"

since: 2.0.00

"APR::Const::FILEPATH_SECUREROOT"

since: 2.0.00

"APR::Const::FILEPATH_SECUREROOTTEST"

since: 2.0.00

"APR::Const::FILEPATH_TRUENAME"

since: 2.0.00

":fprot"

use APR::Const -compile => qw(:fprot);

The ":fprot" group is used by "\$finfo->protection".

"APR::Const::FPROT_GEXECUTE"

Execute by group

since: 2.0.00

"APR::Const::FPROT_GREAD"

Read by group

since: 2.0.00

"APR::Const::FPROT_GSETID"

Set group id

since: 2.0.00

"APR::Const::FPROT_GWRITE"

Write by group

since: 2.0.00

"APR::Const::FPROT_OS_DEFAULT"

use OS's default permissions

since: 2.0.00

"APR::Const::FPROT_UEXECUTE"

Execute by user

since: 2.0.00

"APR::Const::FPROT_UREAD"

Read by user

since: 2.0.00

"APR::Const::FPROT_USETID"

Set user id

since: 2.0.00

"APR::Const::FPROT_UWRITE"

Write by user

since: 2.0.00

"APR::Const::FPROT_WEXECUTE"

Execute by others

since: 2.0.00

"APR::Const::FPROT_WREAD"

Read by others

since: 2.0.00

"APR::Const::FPROT_WSTICKY"

Sticky bit

since: 2.0.00

"APR::Const::FPROT_WWRITE"

Write by others

since: 2.0.00

":filetype"

```
use APR::Const -compile => qw(:filetype);
```

The ":filetype" group is used by "\$finfo->filetype".

"APR::Const::FILETYPE_BLK"

a file is a block device

since: 2.0.00

"APR::Const::FILETYPE_CHR"

a file is a character device

since: 2.0.00

"APR::Const::FILETYPE_DIR"

a file is a directory

since: 2.0.00

"APR::Const::FILETYPE_LNK"

a file is a symbolic link

since: 2.0.00

"APR::Const::FILETYPE_NOFILE"

the file type is undetermined.

since: 2.0.00

"APR::Const::FILETYPE_PIPE"

a file is a FIFO or a pipe.

since: 2.0.00

"APR::Const::FILETYPE_REG"

a file is a regular file.

since: 2.0.00

"APR::Const::FILETYPE SOCK"

a file is a [unix domain] socket.

since: 2.0.00

"APR::Const::FILETYPE_UNKFILE"

a file is of some other unknown type or the type cannot be determined.

since: 2.0.00

":finfo"

use APR::Const -compile => qw(:finfo);

The ":finfo" group is used by "stat()" and "\$finfo->valid".

"APR::Const::FINFO_ETIME"

Access Time

since: 2.0.00

"APR::Const::FINFO_CSIZE"

Storage size consumed by the file

since: 2.0.00

"APR::Const::FINFO_CTIME"

Creation Time

since: 2.0.00

"APR::Const::FINFO_DEV"

Device

since: 2.0.00

"APR::Const::FINFO_DIRENT"

an atomic unix apr_dir_read()

since: 2.0.00

"APR::Const::FINFO_GPROT"

Group protection bits

since: 2.0.00

"APR::Const::FINFO_GROUP"

Group id

since: 2.0.00

"APR::Const::FINFO_ICASE"

whether device is case insensitive

since: 2.0.00

"APR::Const::FINFO_IDENT"

device and inode

since: 2.0.00

"APR::Const::FINFO_INODE"

Inode

since: 2.0.00

"APR::Const::FINFO_LINK"

Stat the link not the file itself if it is a link

since: 2.0.00

"APR::Const::FINFO_MIN"

type, mtime, ctime, atime, size

since: 2.0.00

"APR::Const::FINFO_MTIME"

Modification Time

since: 2.0.00

"APR::Const::FINFO_NAME"

name in proper case

since: 2.0.00

"APR::Const::FINFO_NLINK"

Number of links

since: 2.0.00

"APR::Const::FINFO_NORM"

All fields provided by an atomic unix apr_stat()

since: 2.0.00

"APR::Const::FINFO_OWNER"

user and group

since: 2.0.00

"APR::Const::FINFO_PROT"

all protections

since: 2.0.00

"APR::Const::FINFO_SIZE"

Size of the file

since: 2.0.00

"APR::Const::FINFO_TYPE"

Type

since: 2.0.00

"APR::Const::FINFO_UPROT"

User protection bits

since: 2.0.00

"APR::Const::FINFO_USER"

User id

since: 2.0.00

"APR::Const::FINFO_WPROT"

World protection bits

since: 2.0.00

":flock"

```
use APR::Const -compile => qw(:flock);
```

The ":flock" group is for XXX constants.

"APR::Const::FLOCK_EXCLUSIVE"

since: 2.0.00

"APR::Const::FLOCK_NONBLOCK"

since: 2.0.00

"APR::Const::FLOCK_SHARED"

since: 2.0.00

"APR::Const::FLOCK_TYPEMASK"

since: 2.0.00

":hook"

```
use APR::Const -compile => qw(:hook);
```

The ":hook" group is for XXX constants.

"APR::Const::HOOK_FIRST"

since: 2.0.00

"APR::Const::HOOK_LAST"

since: 2.0.00

"APR::Const::HOOK_MIDDLE"

since: 2.0.00

"APR::Const::HOOK_REALLY_FIRST"

since: 2.0.00

"APR::Const::HOOK_REALLY_LAST"

since: 2.0.00

":limit"

```
use APR::Const -compile => qw(:limit);
```

The ":limit" group is for XXX constants.

"APR::Const::LIMIT_CPU"

since: 2.0.00

"APR::Const::LIMIT_MEM"

since: 2.0.00

"APR::Const::LIMIT_NOFILE"

since: 2.0.00

"APR::Const::LIMIT_NPROC"

since: 2.0.00

":lockmech"

```
use APR::Const -compile => qw(:lockmech);
```

The ":lockmech" group is for XXX constants.

"APR::Const::LOCK_DEFAULT"

since: 2.0.00

"APR::Const::LOCK_FCNTL"

since: 2.0.00

"APR::Const::LOCK_FLOCK"

since: 2.0.00

"APR::Const::LOCK_POSIXSEM"

since: 2.0.00

"APR::Const::LOCK_PROC_PTHREAD"

since: 2.0.00

"APR::Const::LOCK_SYSVSEM"

since: 2.0.00

":poll"

use APR::Const -compile => qw(:poll);

The ":poll" group is used by "poll".

"APR::Const::POLLERR"

since: 2.0.00

Pending error

"APR::Const::POLLHUP"

since: 2.0.00

Hangup occurred

"APR::Const::POLLIN"

since: 2.0.00

Can read without blocking

"APR::Const::POLLNVAL"

since: 2.0.00

Descriptor invalid

"APR::Const::POLLOUT"

since: 2.0.00

Can write without blocking

"APR::Const::POLLPRI"

since: 2.0.00

Priority data available

":read_type"

use APR::Const -compile => qw(:read_type);

The ":read_type" group is for IO constants.

"APR::Const::BLOCK_READ"

since: 2.0.00

the read function blocks

"APR::Const::NONBLOCK_READ"

since: 2.0.00

the read function does not block

":shutdown_how"

```
use APR::Const -compile => qw(:shutdown_how);
```

The ":shutdown_how" group is for XXX constants.

"APR::Const::SHUTDOWN_READ"

since: 2.0.00

"APR::Const::SHUTDOWN_READWRITE"

since: 2.0.00

"APR::Const::SHUTDOWN_WRITE"

since: 2.0.00

":socket"

```
use APR::Const -compile => qw(:socket);
```

The ":socket" group is for the "APR::Socket" object constants, in methods "opt_get" and "opt_set".

The following section discusses in detail each of the ":socket" constants.

"APR::Const::SO_DEBUG"

Possible values:

XXX

since: 2.0.00

Turns on debugging information

"APR::Const::SO_DISCONNECTED"

Queries the disconnected state of the socket. (Currently only used on Windows)

Possible values:

XXX

since: 2.0.00

"APR::Const::SO_KEEPALIVE"

Keeps connections active

Possible values:

XXX

since: 2.0.00

"APR::Const::SO_LINGER"

Lingers on close if data is present

since: 2.0.00

"APR::Const::SO_NONBLOCK"

Turns blocking IO mode on/off for socket.

Possible values:

1 nonblocking

0 blocking

For example, to set a socket to a blocking IO mode:

```
use APR::Socket ();  
use APR::Const -compile => qw(SO_NONBLOCK);  
...  
if ($socket->opt_get(APR::Const::SO_NONBLOCK)) {  
    $socket->opt_set(APR::Const::SO_NONBLOCK => 0);  
}
```

You don't have to query for this option, before setting it. It was done for the demonstration purpose.

since: 2.0.00

"APR::Const::SO_RCVBUF"

Controls the "ReceiveBufferSize" setting

Possible values:

XXX

since: 2.0.00

"APR::Const::SO_REUSEADDR"

The rules used in validating addresses supplied to bind should allow reuse of local addresses.

Possible values:

XXX

since: 2.0.00

"APR::Const::SO_SNDBUF"

Controls the "SendBufferSize" setting

Possible values:

XXX

since: 2.0.00

":status"

```
use APR::Const -compile => qw(:status);
```

The `":status"` group is for the API that return status code, or set the error variable XXXXXX.

The following section discusses in detail each of the available `":status"` constants.

`"APR::Const::TIMEUP"`

The operation did not finish before the timeout.

since: 2.0.00

Due to possible variants in conditions matching `"TIMEUP"`, for checking error codes against this you most likely want to use the `"APR::Status::is_TIMEUP"` function instead.

`":table"`

```
use APR::Const -compile => qw(:table);
```

The `":table"` group is for `"overlap()"` and `"compress()"` constants. See `"APR::Table"` for details.

`"APR::Const::OVERLAP_TABLES_MERGE"`

since: 2.0.00

See `"APR::Table::compress"` and `"APR::Table::overlap"`.

`"APR::Const::OVERLAP_TABLES_SET"`

since: 2.0.00

See `"APR::Table::compress"` and `"APR::Table::overlap"`.

`":uri"`

```
use APR::Const -compile => qw(:uri);
```

The `":uri"` group of constants is for manipulating URIs.

`"APR::Const::URI_ACAP_DEFAULT_PORT"`

since: 2.0.00

`"APR::Const::URI_FTP_DEFAULT_PORT"`

since: 2.0.00

`"APR::Const::URI_GOPHER_DEFAULT_PORT"`

since: 2.0.00

`"APR::Const::URI_HTTPS_DEFAULT_PORT"`

since: 2.0.00

`"APR::Const::URI_HTTP_DEFAULT_PORT"`

since: 2.0.00

`"APR::Const::URI_IMAP_DEFAULT_PORT"`

since: 2.0.00

"APR::Const::URI_LDAP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_NFS_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_NNTP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_POP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_PROSPERO_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_RTSP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_SIP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_SNEWS_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_SSH_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_TELNET_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_TIP_DEFAULT_PORT"

since: 2.0.00

"APR::Const::URI_UNP_OMITPASSWORD"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_UNP_OMITPATHINFO"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_UNP_OMITQUERY"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_UNP_OMITSITEPART"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_UNP_OMITUSER"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_UNP_OMITUSERINFO"

since: 2.0.00

"APR::Const::URI_UNP_REVEALPASSWORD"

since: 2.0.00

See "APR::URI::unparse".

"APR::Const::URI_WAIS_DEFAULT_PORT"

since: 2.0.00

Other Constants

"APR::PerlIO::PERLIO_LAYERS_ARE_ENABLED"

since: 2.0.00

See "APR::PerlIO::Constants")

See Also

mod_perl 2.0 documentation.

Copyright

mod_perl 2.0 and its core modules are copyrighted under The Apache Software License,

Version 2.0.

Authors

The mod_perl development team and numerous contributors.

perl v5.34.0

libapache2-mod-perl2-2.0.12::docs::api::APR::Const(3pm)