



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'Internals.3pm' command

\$ man Internals.3pm

Internals(3pm) Perl Programmers Reference Guide Internals(3pm)

NAME

Internals - Reserved special namespace for internals related functions

SYNOPSIS

```
$is_ro= Internals::SvREADONLY($x)
```

```
$refcnt= Internals::SvREFCNT($x)
```

```
hv_clear_placeholders(%hash);
```

DESCRIPTION

The Internals namespace is used by the core Perl development team to expose certain low level internals routines for testing and other purposes.

In theory these routines were not and are not intended to be used outside of the perl core, and are subject to change and removal at any time.

In practice people have come to depend on these over the years, despite being historically undocumented, so we will provide some level of forward compatibility for some time. Nevertheless you can assume that any routine documented here is experimental or deprecated and you should find alternatives to their use.

FUNCTIONS

```
SvREFCNT(THING [, $value])
```

Historically Perl has been a refcounted language. This means that each variable tracks how many things reference it, and when the

variable is no longer referenced it will automatically free itself.

In theory Perl code should not have to care about this, and in a future version Perl might change to some other strategy, although in practice this is unlikely.

This function allows one to violate the abstraction of variables and get or set the refcount of a variable, and in generally is really only useful in code that is testing refcount behavior.

NOTE You are strongly discouraged from using this function in non-test code and especially discouraged from using the set form of this function. The results of doing so may result in segmentation faults or other undefined behavior.

`SvREADONLY(THING, [, $value])`

Set or get whether a variable is readonly or not. Exactly what the readonly flag means depend on the type of the variable affected and the version of perl used.

You are strongly discouraged from using this function directly. It is used by various core modules, like "Hash::Util", and the "constant" pragma to implement higher-level behavior which should be used instead.

See the core implementation for the exact meaning of the readonly flag for each internal variable type.

`hv_clear_placeholders(%hash)`

Clear any placeholders from a locked hash. Should not be used directly. You should use the wrapper functions provided by Hash::Util instead. As of 5.25 also available as "`Hash::Util::_clear_placeholders(%hash)`"

AUTHOR

Perl core development team.

SEE ALSO

`perlguts` `Hash::Util` `constant` `universal.c`