



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 'umask.1p' command

\$ man umask.1p

UMASK(1P) POSIX Programmer's Manual UMASK(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

umask ? get or set the file mode creation mask

SYNOPSIS

umask [-S] [mask]

DESCRIPTION

The umask utility shall set the file mode creation mask of the current shell execution environment (see Section 2.12, Shell Execution Environment) to the value specified by the mask operand. This mask shall affect the initial value of the file permission bits of subsequently created files. If umask is called in a subshell or separate utility execution environment, such as one of the following:

```
(umask 002)
```

```
nohup umask ...
```

```
find . -exec umask ... \;
```

it shall not affect the file mode creation mask of the caller's environment.

If the mask operand is not specified, the umask utility shall write to

standard output the value of the file mode creation mask of the invoking process.

OPTIONS

The umask utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following option shall be supported:

-S Produce symbolic output.

The default output style is unspecified, but shall be recognized on a subsequent invocation of umask on the same system as a mask operand to restore the previous file mode creation mask.

OPERANDS

The following operand shall be supported:

mask A string specifying the new file mode creation mask. The string is treated in the same way as the mode operand described in the EXTENDED DESCRIPTION section for chmod. For a symbolic_mode value, the new value of the file mode creation mask shall be the logical complement of the file permission bits portion of the file mode specified by the symbolic_mode string.

In a symbolic_mode value, the permissions op characters '+' and '-' shall be interpreted relative to the current file mode creation mask; '+' shall cause the bits for the indicated permissions to be cleared in the mask; '-' shall cause the bits for the indicated permissions to be set in the mask.

The interpretation of mode values that specify file mode bits other than the file permission bits is unspecified.

In the octal integer form of mode, the specified bits are set in the file mode creation mask.

The file mode creation mask shall be set to the resulting numeric value.

The default output of a prior invocation of umask on the same system with no operand also shall be recognized as a mask operand.

STDIN

Not used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of umask:

LANG Provide a default value for the internationalization variables that are unset or null. (See the Base Definitions volume of POSIX.1?2017, Section 8.2, Internationalization Variables for the precedence of internationalization variables used to determine the values of locale categories.)

LC_ALL If set to a non-empty string value, override the values of all the other internationalization variables.

LC_CTYPE Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error.

NLSPATH Determine the location of message catalogs for the processing of LC_MESSAGES.

ASYNCHRONOUS EVENTS

Default.

STDOUT

When the mask operand is not specified, the umask utility shall write a message to standard output that can later be used as a umask mask operand.

If -S is specified, the message shall be in the following format:

```
"u=%s,g=%s,o=%s\n", <owner permissions>, <group permissions>,  
<other permissions>
```

where the three values shall be combinations of letters from the set

{r, w, x}; the presence of a letter shall indicate that the corresponding bit is clear in the file mode creation mask.

If a mask operand is specified, there shall be no output written to standard output.

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

- 0 The file mode creation mask was successfully changed, or no mask operand was supplied.
- >0 An error occurred.

CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

APPLICATION USAGE

Since `umask` affects the current shell execution environment, it is generally provided as a shell regular built-in.

In contrast to the negative permission logic provided by the file mode creation mask and the octal number form of the mask argument, the symbolic form of the mask argument specifies those permissions that are left alone.

EXAMPLES

Either of the commands:

```
umask a=rx,ug+w
umask 002
```

sets the mode mask so that subsequently created files have their `S_IWOTH` bit cleared.

After setting the mode mask with either of the above commands, the `umask` command can be used to write out the current value of the mode

mask:

```
$ umask
```

```
0002
```

(The output format is unspecified, but historical implementations use the octal integer mode format.)

```
$ umask -S
```

```
u=rwx,g=rwx,o=rx
```

Either of these outputs can be used as the mask operand to a subsequent invocation of the umask utility.

Assuming the mode mask is set as above, the command:

```
umask g-w
```

sets the mode mask so that subsequently created files have their S_IW?

GRP and S_IWOTH bits cleared.

The command:

```
umask -- -w
```

sets the mode mask so that subsequently created files have all their write bits cleared. Note that mask operands -r, -w, -x or anything beginning with a <hyphen-minus>, must be preceded by "--" to keep it from being interpreted as an option.

RATIONALE

Since umask affects the current shell execution environment, it is generally provided as a shell regular built-in. If it is called in a subshell or separate utility execution environment, such as one of the following:

```
(umask 002)
```

```
nohup umask ...
```

```
find . -exec umask ... \;
```

it does not affect the file mode creation mask of the environment of the caller.

The description of the historical utility was modified to allow it to use the symbolic modes of chmod. The -s option used in early proposals was changed to -S because -s could be confused with a symbolic_mode form of mask referring to the S_ISUID and S_ISGID bits.

The default output style is unspecified to permit implementors to provide migration to the new symbolic style at the time most appropriate to their users. A -o flag to force octal mode output was omitted because the octal mode may not be sufficient to specify all of the information that may be present in the file mode creation mask when more secure file access permission checks are implemented.

It has been suggested that trusted systems developers might appreciate ameliorating the requirement that the mode mask "affects" the file access permissions, since it seems access control lists might replace the mode mask to some degree. The wording has been changed to say that it affects the file permission bits, and it leaves the details of the behavior of how they affect the file access permissions to the description in the System Interfaces volume of POSIX.1-2017.

FUTURE DIRECTIONS

None.

SEE ALSO

Chapter 2, Shell Command Language, chmod

The Base Definitions volume of POSIX.1-2017, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guidelines

The System Interfaces volume of POSIX.1-2017, umask()

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see <https://www.ker>

