



*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 'strlen.3p' command***

***\$ man strlen.3p***

STRLEN(3P)                    POSIX Programmer's Manual                    STRLEN(3P)

### 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

strlen, strlen ? get length of fixed size string

### SYNOPSIS

```
#include <string.h>
```

```
size_t strlen(const char *s);
```

```
size_t strnlen(const char *s, size_t maxlen);
```

### DESCRIPTION

For `strlen()`: The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1?2017 defers to the ISO C standard.

The `strlen()` function shall compute the number of bytes in the string to which `s` points, not including the terminating NUL character.

The `strnlen()` function shall compute the smaller of the number of bytes in the array to which `s` points, not including any terminating NUL char?

acter, or the value of the `maxlen` argument. The `strnlen()` function shall never examine more than `maxlen` bytes of the array pointed to by

s.

## RETURN VALUE

The `strlen()` function shall return the length of `s`; no return value shall be reserved to indicate an error.

The `strnlen()` function shall return the number of bytes preceding the first null byte in the array to which `s` points, if `s` contains a null byte within the first `maxlen` bytes; otherwise, it shall return `maxlen`.

## ERRORS

No errors are defined.

The following sections are informative.

## EXAMPLES

### Getting String Lengths

The following example sets the maximum length of `key` and `data` by using `strlen()` to get the lengths of those strings.

```
#include <string.h>

...

struct element {
    char *key;
    char *data;
};

...

char *key, *data;
int len;

*keylength = *datalength = 0;

...

if ((len = strlen(key)) > *keylength)
    *keylength = len;

if ((len = strlen(data)) > *datalength)
    *datalength = len;

...
```

## APPLICATION USAGE

None.

## RATIONALE

None.

## FUTURE DIRECTIONS

None.

## SEE ALSO

wcslen()

The Base Definitions volume of POSIX.1-2017, <string.h>

## 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.kernel.org/doc/man-pages/reporting\\_bugs.html](https://www.kernel.org/doc/man-pages/reporting_bugs.html) .

IEEE/The Open Group

2017

STRLEN(3P)