NAME

strverscmp - compare two version strings

SYNOPSIS

#define _GNU_SOURCE /* See feature_test_macros(7)
*/"
#include <string.h>

int strverscmp(const char ****s1*, const char ****s2***);**

DESCRIPTION

Often one has files *jan1*, *jan2*, ..., *jan9*, *jan10*, ... and it feels wrong when ls(1) orders them *jan1*, *jan10*, ..., *jan2*, ..., *jan9*. In order to rectify this, GNU introduced the-*v* option to ls(1), which is implemented using versionsort(3), which again uses **strverscmp**().

Thus, the task of **strverscmp**() is to compare two strings and find the "right" order, while strcmp(3) finds only the lexicographic order. This function does not use the locale category **LC_COLLATE**, so is meant mostly for situations where the strings are expected to be in ASCII.

What this function does is the following. If both strings are equal, return 0. Otherwise, find the position between two bytes with the property that before it both strings are equal, while directly after it there is a difference. Find the largest consecutive digit strings containing (or starting at, or ending at) this position. If one or both of these is empty, then return what strcmp(3) would have returned (numerical ordering of byte values). Otherwise, compare both digit strings numerically, where digit strings with one or more leading zeros are interpreted as if they have a decimal point in front (so that in particular digit strings with more leading zeros come before digit strings with fewer leading zeros). Thus, the ordering is *000*, *00*, *01*, *010*, *09*, *0*, *1*, *9*, *10*.

RETURN VALUE

The **strverscmp**() function returns an integer less than, equal to, or greater than zero if *s1* is found, respectively, to be earlier than, equal to, or later than *s2*.

ATTRIBUTES

For an explanation of the terms used in this section, see attributes(7).

Interface	**Attribute**	**Value**
strverscmp()	Thread safety	MT-Safe

CONFORMING TO

This function is a GNU extension.

EXAMPLE

The program below can be used to demonstrate the behavior of **strverscmp**(). It uses**str verscmp**() to compare the two strings given as its command-line arguments. An example of its use is the following:

```
$ i./a.out jan1 jan10
jan1 < jan10
```

Program source

```
#define _GNU_SOURCE
#include <string.h>
#include <stdio.h>
#include <stdlib.h>

int
main(int argc, char *argv[])
{
int res;

if (argc != 3) {
fprintf(stderr, "Usage: %s <string1> <string2>\n", argv[0]);
```

```
        exit(EXIT_FAILURE);
    }

    res = strverscmp(argv[1], argv[2]);

    printf("%s %s %s\n", argv[1],
            (res < 0) ? "<" : (res == 0) ? "==" : ">", argv[2]);

    exit(EXIT_SUCCESS);
}
```

SEE ALSO

rename(1), strcasecmp(3), strcmp(3), strcoll(3)

COLOPHON

This page is part of release 4.10 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at https://www.kernel.org/doc/man-pages/.