



## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'sys\_un.h.0p' command***

**\$ man sys\_un.h.0p**

sys\_un.h(0P)          POSIX Programmer's Manual          sys\_un.h(0P)

### 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

sys/un.h ? definitions for UNIX domain sockets

### SYNOPSIS

```
#include <sys/un.h>
```

### DESCRIPTION

The <sys/un.h> header shall define the `sockaddr_un` structure, which shall include at least the following members:

`sa_family_t` `sun_family` Address family.

`char` `sun_path[]` Socket pathname.

The `sockaddr_un` structure is used to store addresses for UNIX domain sockets. Pointers to this type shall be cast by applications to `struct sockaddr *` for use with socket functions.

The <sys/un.h> header shall define the `sa_family_t` type as described in <sys/socket.h>.

The following sections are informative.

### APPLICATION USAGE

The size of `sun_path` has intentionally been left undefined. This is be?

cause different implementations use different sizes. For example, 4.3 BSD uses a size of 108, and 4.4 BSD uses a size of 104. Since most implementations originate from BSD versions, the size is typically in the range 92 to 108.

Applications should not assume a particular length for `sun_path` or assume that it can hold `{_POSIX_PATH_MAX}` bytes (256).

Although applications are required to initialize all members (including any non-standard ones) of a `sockaddr_in6` structure (see `<netinet_in.h>`), the same is not required for the `sockaddr_un` structure, since historically many applications only initialized the standard members. Despite this, applications are encouraged to initialize `sockaddr_un` structures in a manner similar to the required initialization of `sockaddr_in6` structures.

#### RATIONALE

None.

#### FUTURE DIRECTIONS

None.

#### SEE ALSO

`<netinet_in.h>`, `<sys_socket.h>`

The System Interfaces volume of POSIX.1-2017, `bind()`, `socket()`, `socketpair()`

#### 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

sys\_un.h(0P)