



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

### ***Linux Ubuntu 26.04 Manual Pages on command 'lrand48\_r.3'***

***\$ man lrand48\_r.3***

drand48\_r(3)                      Library Functions Manual                      drand48\_r(3)

#### **NAME**

drand48\_r, erand48\_r, lrand48\_r, nrand48\_r, mrand48\_r, jrand48\_r, srand48\_r, seed48\_r,  
lcong48\_r - generate uniformly distributed pseudo-random numbers reentrantly

#### **LIBRARY**

Standard C library (libc, -lc)

#### **SYNOPSIS**

```
#include <stdlib.h>
```

```
int drand48_r(struct drand48_data *restrict buffer,  
             double *restrict result);
```

```
int erand48_r(unsigned short xsubi[3],  
             struct drand48_data *restrict buffer,  
             double *restrict result);
```

```
int lrand48_r(struct drand48_data *restrict buffer,  
             long *restrict result);
```

```
int nrand48_r(unsigned short xsubi[3],  
             struct drand48_data *restrict buffer,  
             long *restrict result);
```

```
int mrand48_r(struct drand48_data *restrict buffer,  
             long *restrict result);
```

```
int jrand48_r(unsigned short xsubi[3],  
             struct drand48_data *restrict buffer,  
             long *restrict result);
```

```
int srand48_r(long int seedval, struct drand48_data *buffer);

int seed48_r(unsigned short seed16v[3], struct drand48_data *buffer);

int lcong48_r(unsigned short param[7], struct drand48_data *buffer);
```

Feature Test Macro Requirements for glibc (see `feature_test_macros(7)`):

All functions shown above:

```
/* glibc >= 2.19: */ _DEFAULT_SOURCE

|| /* glibc <= 2.19: */ _SVID_SOURCE || _BSD_SOURCE
```

DESCRIPTION

These functions are the reentrant analogs of the functions described in `drand48(3)`. Instead of modifying the global random generator state, they use the supplied data buffer. Before the first use, this struct must be initialized, for example, by filling it with `zeroes`, or by calling one of the functions `srand48_r()`, `seed48_r()`, or `lcong48_r()`.

RETURN VALUE

The return value is 0.

ATTRIBUTES

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

|                                                                                      |                                         |         |   |
|--------------------------------------------------------------------------------------|-----------------------------------------|---------|---|
| ???????????????????????????????????????????????????????????????????????????????????? |                                         |         |   |
| ? Interface                                                                          | ? Attribute                             | ? Value | ? |
| ???????????????????????????????????????????????????????????????????????????????????? |                                         |         |   |
| ? <code>drand48_r()</code> , <code>erand48_r()</code> , <code>lrand48_r()</code> ,   | ? Thread safety ? MT-Safe race:buffer ? |         |   |
| ? <code>nrand48_r()</code> , <code>mrnd48_r()</code> , <code>jrand48_r()</code> ,    | ?                                       | ?       | ? |
| ? <code>srand48_r()</code> , <code>seed48_r()</code> , <code>lcong48_r()</code>      | ?                                       | ?       | ? |
| ???????????????????????????????????????????????????????????????????????????????????? |                                         |         |   |

STANDARDS

GNU.

SEE ALSO

`drand48(3)`, `rand(3)`, `random(3)`