



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 'mktime.3p' command

\$ man mktime.3p

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

mktime ? convert broken-down time into time since the Epoch

SYNOPSIS

```
#include <time.h>

time_t mktime(struct tm *timeptr);
```

DESCRIPTION

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 mktime() function shall convert the broken-down time, expressed as local time, in the structure pointed to by timeptr, into a time since the Epoch value with the same encoding as that of the values returned by time(). The original values of the tm_wday and tm_yday components of the structure shall be ignored, and the original values of the other components shall not be restricted to the ranges described in <time.h>.

A positive or 0 value for tm_isdst shall cause mktime() to presume ini?

tially that Daylight Savings Time, respectively, is or is not in effect for the specified time. A negative value for `tm_isdst` shall cause `mktime()` to attempt to determine whether Daylight Savings Time is in effect for the specified time.

Local timezone information shall be set as though `mktime()` called `tzset()`.

The relationship between the `tm` structure (defined in the `<time.h>` header) and the time in seconds since the Epoch is that the result shall be as specified in the expression given in the definition of seconds since the Epoch (see the Base Definitions volume of POSIX.1-2017, Section 4.16, Seconds Since the Epoch) corrected for timezone and any seasonal time adjustments, where the names other than `tm_yday` in the structure and in the expression correspond, and the `tm_yday` value used in the expression is the day of the year from 0 to 365 inclusive, calculated from the other `tm` structure members specified in `<time.h>` (excluding `tm_wday`).

Upon successful completion, the values of the `tm_wday` and `tm_yday` components of the structure shall be set appropriately, and the other components shall be set to represent the specified time since the Epoch, but with their values forced to the ranges indicated in the `<time.h>` entry; the final value of `tm_mday` shall not be set until `tm_mon` and `tm_year` are determined.

RETURN VALUE

The `mktime()` function shall return the specified time since the Epoch encoded as a value of type `time_t`. If the time since the Epoch cannot be represented, the function shall return the value $(\text{time_t})-1$ and set `errno` to indicate the error.

ERRORS

The `mktime()` function shall fail if:

E_OVERFLOW

The result cannot be represented.

The following sections are informative.

EXAMPLES

What day of the week is July 4, 2001?

```
#include <stdio.h>

#include <time.h>

struct tm time_str;

char daybuf[20];

int main(void)
{
    time_str.tm_year = 2001 - 1900;

    time_str.tm_mon = 7 - 1;

    time_str.tm_mday = 4;

    time_str.tm_hour = 0;

    time_str.tm_min = 0;

    time_str.tm_sec = 1;

    time_str.tm_isdst = -1;

    if (mktime(&time_str) == -1)
        (void)puts("-unknown-");
    else {
        (void)strftime(daybuf, sizeof(daybuf), "%A", &time_str);

        (void)puts(daybuf);
    }

    return 0;
}
```

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

asctime(), clock(), ctime(), difftime(), gmtime(), localtime(), strf?
time(), strftime(), time(), tzset(), utime()

The Base Definitions volume of POSIX.1?2017, Section 4.16, Seconds

Since the Epoch, <time.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 .

IEEE/The Open Group

2017

MKTIME(3P)