

How to convert between local time and UNIX timestamp in Perl

Author : Dan Nanni

Categories : [Perl](#)

Tagged as : [time](#)

Question: I need to convert readable date and time to a corresponding UNIX timestamp (i.e., Epoch time), or vice versa in a Perl program. Can you show me Perl code examples of converting date/time to UNIX timestamp, or conversely, converting UNIX timestamp to human-readable date and time?

When your Perl script needs to deal with time information, there are two different ways to represent and manipulate date and time in the script. One way is a human-readable string representation of time (e.g., "Sat Mar 14 10:14:05 EDT 2015"), and the other is via UNIX timestamp (also known as "Epoch time") which is the number of elapsed seconds since 00:00:00 UTC, 1 January 1970. Either method has its own pros and cons, and depending on your requirements you may want to convert one format to the other.

Convert Local Time to UNIX Timestamp in Perl

To obtain UNIX time from a date string, you can use `str2time()` in `Date::Parse` module. It can handle a variety of date formats, such as:

- Sat Mar 14 10:14:05 EDT 2015
- 3/14/2015 10:14:05 -0400
- 14/Mar/15 10:14:05
- 14 Mar 15 10:14:05

`Date::Parse` module supports multiple languages (English, French, German and Italian) and timezones. For example:

Convert UNIX Timestamp to Human-Readable Date and Time in Perl

If you want to show a UNIX timestamp in a human-readable format, you can use `localtime()` function which converts a UNIX timestamp into a 9-element list. You can use the returned list to construct any kind of human-readable date/time string as you want. Here is a code snippet.