

Shell Scripting: How to easily convert UNIX timestamp into date format

When writing shell scripts (bash, sh, etc) maybe you had to work with POSIX/UNIX timestamps from time to time. While the serialized nature of the timestamp is great to work with for scripting, it's easier for human beings to have them printed in date format.

Before you start digging around using some fancy conversion in Perl, check out the 'date' command first.

Here's a little snippet on how to do it on Linux:

```
date -u --date="1970-01-01 1285250916 sec GMT"
```

This will convert your timestamp 1285250916 into it's date representation of **Thu Sep 23 14:08:36 UTC 2010**.

As always, there's slight variations with Linux and BSD userland. To achieve the same on OS X and FreeBSD (maybe other BSDs as well), here's the appropriate command:

```
date -j -f %s 1285250916
```

And for Perl lovers anyway, here is the Perl command ;-)

```
perl -e "print scalar localtime (1285250916)"
```

As such it's very easy to capture the output using backticks and use it further on in your scripts. The date command comes in as a last resort especially if you're working in a restricted environment where higher level languages such as Perl may be unavailable. Though one notices Perl is more versatile than the date command - the later having slight syntactical variations between distributions and vendors probably causing some headache. This opts for the use of Perl if you want to be more platform independent with your script.