

# Configurer la date et l'heure sous Linux

## Comment configurer la date et l'heure sous Linux ?

---

Originale: [Linux - date et timedatectl : configurer la date et l'heure \(it-connect.fr\)](https://it-connect.fr/linux-date-et-timedatectl-configurer-la-date-et-l-heure)

---

### I. Présentation

**Dans ce tutoriel, nous allons apprendre à configurer la date et l'heure sur une machine Linux, en utilisant la commande date.** Cette commande native est intégrée aux différentes distributions Linux : Ubuntu, Debian, etc... Elle sert principalement à afficher la date et l'heure actuelle, ainsi qu'à effectuer la configuration de sa machine pour modifier la date et l'heure. Nous verrons que l'on peut faire aussi d'autres choses. **Nous verrons également comment utiliser la commande timedatectl.**

### II. Syntaxe de la commande date

Si vous exécutez la commande "date" dans un terminal Linux sans préciser d'options, c'est la date et l'heure actuelle qui sera retournée, avec le fuseau horaire en complément. Par exemple :

```
date
jeu. 15 sept. 2022 11: 53: 56 CEST
```

C'est pratique lorsque l'on cherche à voir la date et l'heure locale d'une machine Linux.

Pour la suite, il va falloir utiliser plutôt ce format :

```
date <option> <format>
```

Avant de passer aux exemples, je souhaitais vous montrer comment récupérer la date selon le format "Année Mois Jour", sans espace. C'est pratique pour nommer un fichier ou une sauvegarde avec la date du jour, au format compact.

```
date +%Y%m%d
```

Cette commande retourne bien la date actuelle, dans un format réduit :

```
root@debian-11:~# date
jeu. 15 sept. 2022 14:08:50 CEST
root@debian-11:~# date +%Y%m%d
20220915
root@debian-11:~#
```

## III. Exemples d'utilisation de la commande date

### A. Modifier la date et l'heure du système

Commençons par l'exemple qui fait l'objet de cet article : **modifier la date et l'heure du système**. L'option "--set" que l'on peut raccourcir par "-s" doit être utilisée. Pour définir le 15 septembre 2022 à 12:00:00 comme date et heure, cela donne :

```
date --set "15 SEP 2022 12:00:00"
ou
date -s "15 SEP 2022 12:00:00"
```

On peut considérer la syntaxe suivante :

```
date --set "<jour> <mois> <année> <heure>:<minutes>:<secondes>"
```

**Pour le mois, il faut l'indiquer sur trois lettres, mais en se basant sur les noms anglais.**

Ainsi, si l'on prend l'exemple du mois de février, ce sera "FEB" pour "February", et non "FEV". En exécutant la commande "date", on peut voir que le système a bien pris en compte la modification.

Nous pouvons aussi modifier uniquement la date, ou uniquement l'heure, en précisant le format que l'on veut utiliser.

- Définir la date au 15 septembre (09) 2022, sans affecter l'heure

```
date +%Y%m%d -s "20220915"
```

- Définir l'heure à 12:30:00, sans affecter la date

```
date +%T -s "10:13:13"
```

## B. Obtenir une date précise

La commande `date` sert aussi à effectuer des calculs sur les dates, de manière à obtenir une date antérieure. Non pas pour altérer l'heure du système, mais pour récupérer une autre date, ce qui peut être utile dans un script.

On peut déterminer une date spécifique en précisant l'option `"-d"`, comme dans l'exemple ci-dessous avec le `"15/09/2022"` à minuit :

```
date -d "2022-09-15 00:00:00"  
jeu. 15 sept. 2022 00:00:00 CEST
```

- Les dates antérieures

Récupérer la date d'hier :

```
date --date="yesterday"
```

Bien sûr, nous pouvons stocker cette information dans une variable pour la réutiliser. L'exemple ci-dessous stocke l'information dans une variable nommée `"DateHier"`.

```
DateHier=$(date --date="yesterday")
```

Récupérer la date 7 jours avant :

```
date --date="7 day ago"
```

Récupérer la date actuelle - 1 an :

```
date --date="1 year ago"
```

- Les dates futures

Récupérer la date de demain :

```
date --date="tomorrow"
```

Récupérer la date actuelle + 7 jours :

```
date --date="7 day"
```

Récupérer la date actuelle + 1 an :

```
date --date="1 year"
```

## C. Date de dernière modification d'un fichier

Il y a plusieurs manières d'obtenir la date de dernière modification d'un fichier, et la commande `date` fait partie des commandes qui permettent d'obtenir cette information. Prenons l'exemple du fichier système `/etc/hostname` qui contient le nom d'hôte :

```
date -r /etc/hostname
```

## IV. L'alternative : la commande `timedatectl`

La commande **`timedatectl`** permet aussi de modifier la date et l'heure du système, à l'instar de la commande `date`. Lorsque l'on exécute cette commande, elle affiche des informations complémentaires vis-à-vis de la commande `date` :

```
timedatectl
Local time: jeu. 2022-09-15 13:41:18 CEST
Universal time: jeu. 2022-09-15 11:41:18 UTC
RTC time: jeu. 2022-09-15 11:41:18
Time zone: Europe/Paris (CEST, +0200)
System clock synchronized: yes
NTP service: active
RTC in local TZ: no
```

Il y a l'heure locale, l'heure universelle, mais aussi le fuseau horaire (*Time zone*), l'état du service NTP pour la synchronisation de l'heure (*NTP service*), etc.

```
root@debian-11:~# timedatectl
      Local time: jeu. 2022-09-15 13:55:58 CEST
      Universal time: jeu. 2022-09-15 11:55:58 UTC
          RTC time: jeu. 2022-09-15 11:55:58
          Time zone: Europe/Paris (CEST, +0200)
System clock synchronized: yes
          NTP service: active
          RTC in local TZ: no
root@debian-11:~#
```

## A. Changer la date et l'heure avec `timedatectl`

Pour définir une nouvelle date, on utilise la syntaxe suivante :

```
timedatectl set-time YYYY-MM-DD
```

Où "YYYY" correspond à l'année, "MM" au mois et "DD" au jour. Pour le 15 septembre 2022, cela donne :

```
timedatectl set-time 2022-09-15
```

On peut aussi modifier l'heure en même temps :

```
timedatectl set-time YYYY-MM-DD HH:MM:SS
```

Où "HH" correspond à l'heure, "MM" aux minutes et "SS" aux secondes. Au moment de changer la date et l'heure, si vous obtenez le message "**Failed to set time: Automatic time synchronization is enabled**", cela signifie que la synchronisation NTP est activée. Pour gérer la synchronisation NTP :

```
# Désactiver la synchronisation NTP
timedatectl set-ntp 0
# Activer la synchronisation NTP
timedatectl set-ntp 1
```

Dans le cas où la synchronisation NTP est activée, il faut veiller à **configurer un serveur NTP que l'on va utiliser comme source de temps** pour se synchroniser. Sinon, cela n'a pas réellement d'intérêt... **La configuration s'effectue dans le fichier `"/etc/systemd/timesyncd.conf"`.**

## B. Changer de fuseau horaire avec `timedatectl`

Si le fuseau horaire de votre machine ne vous convient pas, vous pouvez le changer. Tout d'abord, vous pouvez lister les différents fuseaux horaires disponibles :

```
list-timezones
```

Ensuite, il faut repérer la valeur dans la - longue - liste. Une fois que c'est fait, on peut changer le fuseau horaire. Voici un exemple pour utiliser le fuseau horaire "Europe/Paris" :

```
timedatectl set-timezone "Europe/Paris"
```

## V. Conclusion

**Grâce à ce tutoriel, vous êtes capable de gérer la date et l'heure sous Linux, avec les commandes `date` et `timedatectl` !**

---

Revision #1

Created 6 October 2022 12:38:20 by edoppel

Updated 7 October 2022 08:20:08 by edoppel