xinput

Ca fait un bail de **xinput** se doit d'être privilégié par rapport à **xorg**.

C'est toujours le zouk car, afin de ne pas perturber les habitudes des utilisateurs, nombre de distribution utilisent les deux pour configurer le serveur X.

Un petit mémo pour voir comment manipuler **xinput**. Très light j'essaierai de le compléter progressivement... en fonction de mon temps libre... donc n'hésitez par à participer

Donc **xinput** est un utilitaire capable de lister les périphériques d'entrée, de les interroger et de les configurer (tel qu'un clavier, touchpad, souris, etc...)

Il n'est pas installé par défaut sur toutes les distributions, donc pensez à l'installer si nécessaire. Plutôt que de recopier le man, voilà un exemple précis pour lister les devices, en questionner un particulier et modifier un de ses paramètres. A savoir dans ce cas précis le tapping d'un touchpad.

Lister les périphériques

```
ragnarok@Fenrir:~$ xinput --list
⎡ Virtual core pointer                    id=2    [master pointer  (3)]
⎜   ↳ Virtual core XTEST pointer              id=4    [slave  pointer
(2)]
⎜   ↳ AlpsPS/2 ALPS DualPoint TouchPad        id=13    [slave  pointer
(2)]
⎜   ↳ AlpsPS/2 ALPS DualPoint Stick           id=14    [slave  pointer
(2)]
⎜   ↳ PixArt USB Optical Mouse                id=15    [slave  pointer
(2)]
⎣ Virtual core keyboard                   id=3    [master keyboard (2)]
    ↳ Virtual core XTEST keyboard             id=5    [slave  keyboard
(3)]
    ↳ Power Button                            id=6    [slave  keyboard
(3)]
    ↳ Video Bus                               id=7    [slave  keyboard
(3)]
    ↳ Power Button                            id=8    [slave  keyboard
(3)]
    ↳ Sleep Button                            id=9    [slave  keyboard
(3)]
    ↳ Laptop_Integrated_Webcam_HD: In         id=10    [slave  keyboard
(3)]
    ↳ Dell WMI hotkeys                        id=11    [slave  keyboard
(3)]
    ↳ AT Translated Set 2 keyboard           id=12    [slave  keyboard
(3)]
```

Chaque périphérique est associé à un **identifiant unique** (ID), c'est lui qu'il faut noter.

Questionner un périphérique particulier

Par exemple, sur un laptop, le touchpad. On voit ici qu'il est associé à l'id 13. Pour avoir sa configuration détaillée, nous utiliserons l'option –list-props suivie du numéro d'identification du device à détailler

```
ragnarok@Fenrir:~$ xinput --list-props 13
Device 'AlpsPS/2 ALPS DualPoint TouchPad':
    Device Enabled (154):    1
    Coordinate Transformation Matrix (156):    1.000000, 0.000000, 0.000000,
0.000000, 1.000000, 0.000000, 0.000000, 0.000000, 1.000000
    libinput Tapping Enabled (292):    0
    libinput Tapping Enabled Default (293):    0
    libinput Tapping Drag Enabled (294):    1
    libinput Tapping Drag Enabled Default (295):    1
    libinput Tapping Drag Lock Enabled (296):    0
    libinput Tapping Drag Lock Enabled Default (297):    0
    libinput Tapping Button Mapping Enabled (298):    1, 0
    libinput Tapping Button Mapping Default (299):    1, 0
    libinput Natural Scrolling Enabled (300):    0
    libinput Natural Scrolling Enabled Default (301):    0
    libinput Disable While Typing Enabled (302):    1
    libinput Disable While Typing Enabled Default (303):    1
    libinput Scroll Methods Available (304):    1, 1, 0
    libinput Scroll Method Enabled (305):    1, 0, 0
    libinput Scroll Method Enabled Default (306):    1, 0, 0
    libinput Middle Emulation Enabled (307):    1
    libinput Middle Emulation Enabled Default (308):    1
    libinput Accel Speed (309):    0.000000
    libinput Accel Speed Default (310):    0.000000
    libinput Left Handed Enabled (311):    0
    libinput Left Handed Enabled Default (312):    0
    libinput Send Events Modes Available (277):    1, 1
    libinput Send Events Mode Enabled (278):    0, 0
    libinput Send Events Mode Enabled Default (279):    0, 0
    Device Node (280):    "/dev/input/event2"
    Device Product ID (281):    2, 8
    libinput Drag Lock Buttons (313):    <no items>
    libinput Horizontal Scroll Enabled (314):    1
```

Nous avons alors toutes les options de configurations supportées. 0 pas activé et 1 activé.

SI je regarde la ligne suivante

```
    libinput Tapping Enabled (292):    0
```

Je me doute qu'il s'agit tu “tapping” sur la zone sensible de mon touchpad et je vois la valeur 0. Donc il n'est pas activé. Je peux le vérifier en faisant un test sur n'importe quel icone

Configurer les périphériques

Pour configurer ce touchpad et activer le tapping, il me suffit de passer la valeur 0 à 1 Tres simple, nous utiliserons l'option –set-prop suivie de l'identifiant du device, de son numéro d'option et de la valeur de ce choix. Exemple

```
 xinput --set-prop 13 292 1
```

Les modifications sont instantanées, le tapping est activé.

Activer / désactiver le TouchPad

On remarque que son ID est le 13 et on jette un oeil sur l'option 154 de ce device

```
    Device Enabled (154):    1
```

Pour le desactiver, simplement passer la valeur de ce choix à 0

```
xinput --set-prop 13 154 0
```

Votre touchpad est desactivé. Pour le réactiver, faire l'opération inverse.

```
xinput --set-prop 13 154 1
```

Et il est réactivé de suite

Vous êtes allergique à la lecture des sorties et chercher les numéros des périphériques d'entrée vous gonfle ?

Recherche de périphérique : alternative

```
 xinput --set-prop `xinput | sed -nr 's/.*TouchPad.*id=([0-9]*).*/\1/p'`
"Device Enabled" 0
```

Désactivera le touchpad

```
xinput --set-prop `xinput | sed -nr 's/.*TouchPad.*id=([0-9]*).*/\1/p'`
"Device Enabled" 0
```

Réactivera le touchpad

Idem si pour le détail des périphériques, vous ne souhaitez pas passer des ID mais des noms de périphériques

```
xinput --list-props `xinput | sed -nr 's/.*TouchPad.*id=([0-9]*).*/\1/p'`
```

Le clavier : autres exemples

Très en vitesse : le clavier

```
serveur@serveurProliant:~$ xinput --list
⎡ Virtual core pointer                    id=2    [master pointer  (3)]
⎜   ↳ Virtual core XTEST pointer              id=4    [slave  pointer
(2)]
⎜   ↳ USB Optical Mouse                       id=8    [slave  pointer
(2)]
⎣ Virtual core keyboard                   id=3    [master keyboard (2)]
    ↳ Virtual core XTEST keyboard             id=5    [slave  keyboard
(3)]
    ↳ Power Button                            id=6    [slave  keyboard
(3)]
    ↳ CHICONY HP Basic USB Keyboard           id=7    [slave  keyboard
(3)]
```

Détailler le clavier

```
serveur@serveurProliant:~$ xinput --list-props 7
Device 'CHICONY HP Basic USB Keyboard':
    Device Enabled (118):    1
    Coordinate Transformation Matrix (120):    1.000000, 0.000000, 0.000000,
0.000000, 1.000000, 0.000000, 0.000000, 0.000000, 1.000000
    libinput Send Events Modes Available (241):    1, 0
    libinput Send Events Mode Enabled (242):    0, 0
    libinput Send Events Mode Enabled Default (243):    0, 0
    Device Node (244):    "/dev/input/event0"
    Device Product ID (245):    1008, 36
```

Tester les touches et récupérer leur ID (un peu comme xbindkeys -k)

```
serveur@serveurProliant:~$ xinput test 7
key release 36
key press   25
zkey release 25
```

From:
https://cbiot.fr/dokuwiki/ - **Cyrille BIOT**

Permanent link:
https://cbiot.fr/dokuwiki/xinput

Last update: **2020/04/04 09:48**