

Alcapdemy:

Curso de personalización de kali linux con BSPWM

1. BSPWM

Bspwm (Binary Space Partitioning Window Manager) es un gestor de ventanas de código abierto para sistemas operativos basados en Linux

2. Instalación de BSPWM

Para instalar bspwm estaremos utilizando un script del usuario rlvs3c. Este script nos automatizará todo el proceso

-Accedemos al repositorio:

<https://github.com/rlvs3c/auto-bspwm>

-Abrimos la terminal e introducimos los siguientes comandos:

```
sudo apt update
```

```
sudo apt upgrade -y
```

```
git clone https://github.com/rlvs3c/auto-bspwm
```

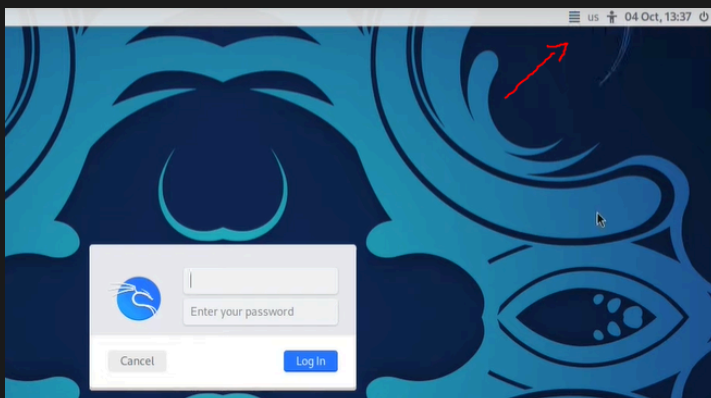
```
cd auto-bspwm
```

```
chmod +x setup.sh
```

```
bash setup.sh
```

-Esperamos: hasta que nos salga un mensaje en el que nos diga si queremos reiniciar, y le daremos a que si escribiendo "y" en la terminal

-Cuando termine de reiniciarse: tendremos que seleccionar el entorno de bspwm de la siguiente forma:



Hacemos click sobre el icono de las 4 líneas y seleccionamos bspwm

3. Lista de atajos de bspwm (básicos)

- Windows + Enter**: Abre la terminal
 - Windows + W**: Cierra la ventana actual
 - Windows + Alt + R**: Ajustar bspwm al tamaño de la ventana.
 - Windows + Alt + Q**: Cierra sesión
 - Windows + (←→)**: Navegar por las ventanas en el espacio actual
 - Windows + D**: Abre el menú de programas
 - Windows + (1,2,3,4,5,6,7,8,9,0)**: Cambia a un espacio de trabajo diferente
- Extra**: En el repositorio oficial (<https://github.com/r1vs3c/auto-bspwm>)

4. Cambiar fondo de pantalla

- Abrimos la terminal
- Accedemos a la configuración de bspwm:
`cd ~/.config/bspwm`
- Abrimos como root el archivo bspwmrc:
`sudo vim bspwmrc`
- Nos encontraremos con lo siguiente:

```
#!/bin/sh

pgrep -x sxhkd & /dev/null || sxhkd &

wmname LG3D &
vmware-user-suid-wrapper &

bspc monitor -d I II III IV V VI VII VIII IX X

bspc config border_width 0
bspc config window_gap 10
bspc config split_ratio 0.50
bspc config borderless_monocle true
bspc config gapless_monocle true
bspc config paddingless_monocle true
bspc config focus_follows_pointer true
bspc config pointer_modifier mod1
bspc config normal_border_color '#1f1f1f'
bspc config focused_border_color '#30302f'
bspc config presel_feedback_color '#512da8'

bspc rule -a Gimp desktop='^8' state=floating follow=on
bspc rule -a Chromium desktop='^2'
bspc rule -a mplayer2 state=floating
bspc rule -a Kupfer.py focus=on
bspc rule -a Screenkey manage=off

feh --bg-fill ~/Wallpapers/archkali.png &
~/.config/polybar/.launch.sh --shapes &
picom --config ~/.config/picom/picom.conf &
# wal -R
```

Tendremos que modificar la línea donde pone `feh -bg-fill ~/Wallpapers/archkali.png`

Dentro de esa línea cambiaremos la ruta que pone, por la de nuestro fondo

EJEMPLO: `feh -bg-fill ~/home/kali/Desktop/Fondo.png`

5. Funciones personalizadas

La `~/.zshrc` es un archivo de texto donde podremos crear funciones personalizadas, te preguntarás que son las funciones personalizadas, pues básicamente son comandos en la terminal que ejecutan lo que hayas definido en la función (scripts de python, bash, shell)

6. Como crear funciones personalizadas

-Abre la terminal y abre el archivo `~/.zshrc`:

```
sudo vim ~/.zshrc
```

-Dentro del archivo busca el comentario funciones:

Encontraras varias funciones como `extractports` o `mkt`, podrás añadir hay tus propias funciones

-Estructura:

toda función sigue esta estructura:

```
function nombre(){  
    Contenido  
}
```

-IMPORTANTE: Cuando termines de crear la función debes refrescar la terminal lanzando el siguiente comando:

```
source ~/.zshrc
```

EJEMPLO: vamos a crear una función que se encargue de cambiar el teclado al idioma español:

```
function español{  
    setxkbmap -layout es  
}
```

Ahora cada vez que pongamos la palabra `español` en la terminal se ejecutará esa función.