

Hacking tint2 panel: weather, CPU temperature and volume executors

Adding weather, volume and CPU status a hacker friendly panel: tint2.

I tried to switch over to XFCE4^[1] numerous times, nearly always put down by something I missed; finally I decided I'm going to force it. I want my speedy, snappy computer back from 2009 and I'm done with the fancy aesthetics; just make it work and make it stable.

Soon I replaced the built-in panel with tint2^[2], mostly for it's unique little thing: executors. This makes tint2 similarly hackable as conky^[3] but due to the mouse handling actions, it can do even more.

Display and change volume

The script below started to use an excessive amount of CPU due to the high amount of scans it performed so I moved on to use `pnmixer`^[4]. It was also useful, since on Debian 9 with XFCE4, I don't have anything that handles the X86 audio keys with notifications correctly - `pnmixer` does the job and it's good enough, so I recommend it over `volumeicon` to display volume with `tint2`.

When you launch `tint2` some of the indicators, like network and bluetooth tend to show up, but not the volume indicator. The usually mentioned fix is to install `volumeicon`^[5] - this is neat, but there is another way.

Since the script below uses the emoji range (also known as astral plane characters) from unicode, installing "ttf-ancient-fonts" might be required. I have no idea why it's called like that.

```
apt-get install ttf-ancient-fonts
```

```
bash
```

Mouse interactions

left click

toggles Master (mute/unmute) output in pulseaudio

right click

toggles Mic (mute/unmute) input in pulseaudio

middle click

opens pulseaudio controls

scroll up

volume up by 2%

scroll down

volume down by 2%

tin2rc snippet

```
#-----
# Executor 3
execp = new
execp_command = ~/scripts/tint2_vol.sh
execp_interval = 1
execp_has_icon = 0
execp_cache_icon = 0
execp_continuous = 0
execp_markup = 1
execp_lclick_command = pactl list sinks | grep -qi 'Mute: yes'
&& pactl set-sink-mute 0 0 || pactl set-sink-mute 0 1
execp_rclick_command = pactl list sources | grep -A12 'Source
#1' | grep -qi 'Mute: yes' && pactl set-source-mute 1 0 ||
pactl set-source-mute 1 1
execp_mclick_command = pavucontrol
execp_uwheel_command = amixer set Capture 2%+; amixer set
Master 2%+
execp_dwheel_command = amixer set Capture 2%-; amixer set
Master 2%-
execp_font = Liberation Mono 8
execp_font_color = #000000 100
execp_padding = 0 0
execp_background_id = 0
execp_centered = 1
execp_icon_w = 0
execp_icon_h = 0
```

apache

tint2_vol.sh

```
#!/usr/bin/env bash

if [ "$1" == "up" ]; then
    /usr/bin/amixer set Master 5%+ >/dev/null 2>&1
elif [ "$1" == "down" ]; then
    /usr/bin/amixer set Master 5%- >/dev/null 2>&1
elif [ "$1" == "mute" ]; then
    a=$(amixer set Master 1+ toggle);
elif [ "$1" == "mic" ]; then
```

bash

```

    pactl list sources | grep -A12 'Source #1' | grep -qi
'Mute: yes' && pactl set-source-mute 1 0 || pactl set-source-
mute 1 1
fi

vpattern=".*\([([0-9]+)%\).*"
spattern=".*\[off\].*"

amixer="amixer -c1"

master=$(($amixer sget 'Master')
mic=$(($amixer sget 'Capture')

mavol=$(echo $master | grep '%' | sed -r "s/$vpattern/\1/")
mivol=$(echo $mic | grep '%' | sed -r "s/$vpattern/\1/")
mivol=0

jackdev=$(($amixer contents | grep -i "'headphone jack'" | cut -
d", " -f1,2)

THEME="/usr/share/icons/$(/usr/bin/gsettings get
org.gnome.desktop.interface icon-theme | tr -d "'")"

if grep -qi $spattern <<< $master; then
    icon="🔇"
    ipath="$(find "$THEME" -name *audio*mute* | grep 24 | head
-n1)"
elif grep -qi 'values=on' <<< $(($amixer cget "$jackdev"); then
    icon="🎧"
    ipath="$(find "$THEME" -name *headphone* | grep 24 | head -
n1)"
elif [ $mavol -gt 0 ] && [ $mavol -lt 31 ]; then
    icon="🔊"
    ipath="$(find "$THEME" -name *audio*low* | grep 24 | head -
n1)"
elif [ $mavol -gt 30 ] && [ $mavol -lt 60 ]; then
    icon="🔊"
    ipath="$(find "$THEME" -name *audio*medium* | grep 24 |
head -n1)"
else

```

```
    icon="🎵"  
    ipath="$(find "$THEME" -name *audio*high* | grep 24 | head  
-n1)"  
fi  
  
if [ -z $1 ]; then  
    printf '%3s\n%3s%' "$icon" "$mavol"  
else  
    notify-send -i $ipath "$mavol %"  
fi
```

CPU temperature, fan speed and governor setting

Mouse interactions

left click

sets CPU governor to `ondemand` if governor is available

right click

sets CPU governor to `powersave` if governor is available

middle click

sets CPU governor to `performance` if governor is available

tin2rc snippet

```
#-----
# Executor 2
execp = new
execp_command = ~/scripts/tint2_cputemp.sh
execp_interval = 30
execp_has_icon = 0
execp_cache_icon = 0
execp_continuous = 0
execp_markup = 0
execp_lclick_command = ~/scripts/tint2_cpufreq.sh ondemand
execp_rclick_command = ~/scripts/tint2_cpufreq.sh powersave
execp_mclick_command = ~/scripts/tint2_cpufreq.sh performance
execp_uwheel_command =
execp_dwheel_command =
execp_font = Liberation Mono 8
execp_font_color = #000000 100
execp_padding = 0 0
execp_background_id = 0
execp_centered = 1
execp_icon_w = 0
execp_icon_h = 0
```

apache

tint2_cputemp.sh

```
#!/bin/bash

temp=$(sensors | grep -i temp1 | head -n1 | sed -r 's/.*:\s+[\+]\?(*C)\s+.*\/\1/')
rpm=$(sensors | grep -i fan | head -n1 | sed -r 's/.*?:\s+(*?)\s+RPM\/\1/')

printf '%8s\n%8s ' "$temp" "$rpm/m"
```

bash

tint2_cpufreq.sh

```
#!/usr/bin/env bash

setto="$1"

if [ "$1" == '' ]; then
    exit 0;
fi

if [ ! -x "$(which cpufreq-selector)" ]; then
    exit 0;
fi

governors="$(cpufreq-info | grep 'available.*governors' |
head -n1)";

if ! grep -q "$setto" <<< "$governors"; then
    echo "this governor is not available"
    exit 0;
fi

declare out;
for proc in $(cat /proc/cpuinfo | grep processor | sed -r 's/
^processor\s+:\s+(.*)$/\1/'); do
    out+="CPU#$proc governor is to '$setto';"
    cpufreq-selector -c $proc -g $setto
done

if [ -x "$(which notify-send)" ]; then
    THEME=$(gsettings get org.gnome.desktop.interface icon-
theme | tr -d "'")
    ICON="/usr/share/icons/${THEME}/rest/of/path/to/icon.svg"

    notify-send -a "cpufreq-selector" "$(tr ';' '\n' <<< $out)"
fi

exit 0;
```

bash

Weather

Mouse interactions

left click

opens ascii art weather forecast from wttr.in Cambridge in browser

tin2rc snippet

```
#-----  
# Executor 1  
execp = new  
execp_command = ~/scripts/tint2_weather.sh  
execp_interval = 300  
execp_has_icon = 0  
execp_cache_icon = 0  
execp_continuous = 0  
execp_markup = 0  
execp_lclick_command = firefox http://wttr.in/cambridge  
execp_rclick_command = firefox http://wttr.in/amsterdam  
execp_mclick_command = firefox http://wttr.in/budapest  
execp_uwheel_command =  
execp_dwheel_command =  
execp_font = Liberation Mono 8  
execp_font_color = #000000 100  
execp_padding = 0 0  
execp_background_id = 0  
execp_centered = 1  
execp_icon_w = 0  
execp_icon_h = 0
```

apache

tint2_weather.sh

```
#!/bin/bash
```

bash

```

geo="$(wget -O- -q http://geoip.ubuntu.com/lookup)"
if grep -qi '88.96.115.94' <<< $geo; then
    lat="52.218011"
    lon="0.140549"
else
    lat="$(sed -r 's/.*<Latitude>(.*?)</Latitude>.*\/\1/g' <<<
$geo)"
    lon="$(sed -r 's/.*<Longitude>(.*?)</Longitude>.*\/\1/g'
<<< $geo)"
fi

weather="$(wget -q -O- http://api.wunderground.com/auto/wui/
geo/WXCurrentObXML/index.xml?query=$lat,$lon)"

kw="weather"
condition="$(sed -r "s/.*<$kw>(.*?)</$kw>.*\/\1/g" <<<
$weather)"

kw="temp_c"
temperature="$(sed -r "s/.*<$kw>(.*?)</$kw>.*\/\1/g" <<<
$weather)"

kw="relative_humidity"
humidity="$(sed -r "s/.*<$kw>(.*?)</$kw>.*\/\1/g" <<<
$weather)"

kw="wind_mph"
wind="$(sed -r "s/.*<$kw>(.*?)</$kw>.*\/\1/g" <<< $weather)"

kw="windchill_c"
feelslike="$(sed -r "s/.*<$kw>(.*?)</$kw>.*\/\1/g" <<<
$weather)"

if grep -qi 'rain' <<< $condition; then
    icon="☔"
elif grep -qi 'storm' <<< $condition; then
    icon="⚡"
elif grep -qi 'cloud' <<< $condition; then
    icon="☁"
elif grep -qi 'clear' <<< $condition; then
    icon="☀"

```

```
elif grep -qi 'snow' <<< $condition; then
    icon="❄️"
else
    icon=$condition
fi

firstline="${icon}  ${temperature}°C (${feelslike}°C) "
secondline="${wind:-0}km/h ${humidity}"

echo "${firstline:0:12}"
echo "${secondline:0:12}"
```

Links

1. <http://xfce.org/>
2. <https://gitlab.com/o9000/tint2>
3. <https://github.com/brndnmtthws/conky>
4. <https://github.com/nicklan/pnmxer>
5. <http://nullwise.com/volumeicon.html>

Created by Peter Molnar <mail@petermolnar.net>, published at 2016-11-04 19:00 UTC, last modified at 2021-05-11 11:49 UTC , to canonical URL <https://petermolnar.net/article/hacking-tint2-panel-weather-cpu-temperature-and-volume-executors/> , licensed under CC-BY-4.0 .