

# Log de instalación PorisToolkit Web Editor en Fedora 16

A continuación se detallan los pasos seguidos para instalar el PorisToolkit Web Editor en la máquina virtual de Fedora 16.

Bajamos la versión de Fedora de la página adecuada.

Nombre: Fedora-16-i686-Live-Desktop.iso

Podríamos intentarlo con la última versión:

<http://fedoraproject.org/es/get-fedora>

Aunque esta instalación se ha hecho con la 16.

<http://download.fedoraproject.org/pub/fedora/linux/releases/16/Live/i686/Fedora-16-i686-Live-Desktop.iso>

Creamos la máquina de Virtualbox con las preferencias:

- S.O.: Linux.
- Distribución: Fedora
- RAM : 768 Mb.
- HDD : 8 Gb, espacio incremental.

Iniciamos la máquina virtual con el ISO en la unidad de CD virtual, arranca desde el propio ISO.

A continuación, escogemos la aplicación “Applications → System Tools → Install to Hard Drive”.

Vamos escogiendo:

- US.English
- Basic Storage Devices
- Yes discard any data
- porisfedora.localdomain
- canary atlantic
- como root password será poristoolkit
- Use All Space (dejamos resto de opciones como está).
- Go writting changes to disk.
- Reboot

Al reiniciar, es posible que nos intente arrancar desde la imagen ISO, hemos de garantizar, para seguir, que arrancamos desde el disco duro (en Virtualbox, extraer virtualmente el disco livedisc, si es necesario).

Cuando arranque pulsamos:

- Forward
- Forward
- Forward

- En Create User:
  - Full Name: Poris Toolkit
  - username: poris
  - Add to administrator group (YES)
  - Contraseña : poris
  - Pulsamos Forward
- Finish.
- No, do not send.
- Entraremos como el usuario Poris Toolkit.
- Instalamos las “Guest Additions de VirtualBox”. Para ello seguimos las indicaciones de esta página: <http://www.sysprobs.com/install-fedora-16-virtualbox-guest-additions-get-working-gnome-shell-inside-virtual-machine>
  - `sudo yum -y update kernel`
  - `sudo yum -y install kernel-devel kernel-headers dkms gcc gcc-c++`
  - Ejecutamos el autorun de la imagen iso de las Guest Additions de VirtualBox.

## Instalando Ruby y Rubygems

Abrimos el terminal el Applications → System Tools → Terminal.

Si tecleamos `ruby -v` no conseguimos nada, así que determinamos que hay que instalarlo.

Applications → System Tools → Add/Remove Software.

Instalamos el paquete “ruby-1.8.7.....” titulado “An interpreter of object. .... language”.

Instalamos el paquete “rubygems-1.8.11...” titulado “The Ruby standard for packaging ruby libraries”.

**NO instalamos rails** desde aquí, ya que nos instalaría la versión 3, y la que necesitamos es la 2.3.14.

Veremos que al pulsar “Apply” nos pedirá comprobación para instalar todas las dependencias necesarias. Confirmamos todo.

Si ahora en la consola tecleamos `ruby -v` veremos que nos informa de la versión 1.8.7 del intérprete de Ruby.

Comprobamos que funciona mediante el comando `gem -v`

## Instalando Hobo 1.0.2 (y rails v.2.3.14)

PorisWebEditor está desarrollado usando Hobo v 1.0.2. Al instalar esa gem, nos intentará instalar Rails v2.3.14. Sin embargo, antes hemos de añadir `gems.github.com` como una de las fuentes válidas para el sistema de gems.

`sudo gem sources --add http://gems.github.com`

`sudo gem install hobo -v=1.0.2`

Probaremos que Rails funciona correctamente

```
cd
mkdir test
cd test
rails test_rails
cd test_rails
script/server
```

Abrimos un navegador (Applications → Internet → Firefox) y abriremos la URL <http://localhost:3000>. Si todo va bien veremos una página de comprobación de Rails.

De vuelta en la consola, pulsaremos Ctrl+C para abortar el servidor Rails.

Probaremos que Hobo funciona

```
cd ..
hobo test_hobo
```

Si al final de un largo proceso vemos que nos muestra un sumario de ficheros creados y vemos que existe el directorio test\_hobo, es que todo ha ido bien.

```
cd test_hobo
rake
```

Veremos que responde "no such file to load -- sqlite3", ya que no tiene el conector de sqlite3, sin embargo la aplicación si debe existir, podemos comprobarlo escribiendo en la consola sqlite3 --version.

Instalamos la gema sqlite3:

Applications → System Tools → Add/Remove Software

Instalamos el paquete rubygem-sqlite3 y rubygem-sqlite3-doc

(de momento no tenemos la gem sqlite3-ruby instalada, hara falta? Continuamos a ver...)

```
script/generate hobo_migration
```

Cuando nos pregunte pulsamos 'm' y luego aceptamos el nombre que propone

```
script/server
```

Volvemos a abrir el navegador en <http://localhost:3000> para ver que se muestra una aplicación Hobo pidiéndonos un registro.

## Instalando Poris

Extraemos poris.tar.gz en un directorio y arrancamos su servidor:

```
tar xvf poris.tar.gz
cd poris
ruby script/server -p 3001 -e production
```

Abrimos el navegador en el puerto 3001

Listo!!!

**Es un buen momento para hacer un “snapshot” a la máquina virtual antes de continuar.**

## Configurando el reenvío de puertos

En el menú Dispositivos de la máquina virtual podemos escoger Adaptadores de Red. Y en las propiedades encontramos "Reenvío de puertos". De esa manera podemos conectar el puerto 3001 del anfitrión con el 3001 de la máquina en Fedora.

## Configurando el autoarranque

De momento no hay un autoarranque configurado. Hay que, manualmente, abrir un terminal, ir al directorio correspondiente, y ejecutar

```
ruby script/server -p 3001 -e production
```

## Instalación de AstroSystems

Astrosystems incluye PorisPlayer. Lo descargamos de la web de descargas de PorisToolkit:

<https://sites.google.com/a/elporis.com/poris-toolkit/poris/downloads>

Una vez descargado lo descomprimos:

```
> tar xvf AstroSystems\ 0.1.tar.gz
```

```
> cd AstroSystems
```

```
> java -jar AstroSystems.jar
```

Vemos que Java no está instalado, así que instalaremos el paquete Java utilizando Applications → System Tools → Add/Remove Software.

Hemos comprobado que es la versión 1.6.0\_22 mediante el comando `java -version`.

Abrimos un explorador en `localhost:3001` y buscamos un modelo para generar un archivo `instrument.xml`. Nos situamos en el nodo en cuestión (en nuestro ejemplo `TxTestInstrument`) y pulsamos sobre “Generate Tree Code for xxx” y con el botón derecho lo guardamos con el nombre `instrument.xml` en el mismo directorio que el fichero `AstroSystems.jar`.

## Borrando espacio

Utilizando la herramienta Applications → System tools → Add/Remove Software, eliminamos:

- Other:
  - evolution.
  - Shotwell.
- Office:
  - evince.
- Internet:
  - transmission-gtk.
  - Empathy.
- GNOME desktop:
  - aisleriot.
  - Brasero-nautilus.
  - Cheese.

- Evince-nautilus.
- Gnome-bluetooth.
- Gnome-contacts.
- Gnome-games.
- Gnome-icon-theme-extras.
- Simple-scan.
- Other Desktops.
  - Matchbox-window-manager.
- Graphics.
  - Libsane-hpaio
  - sane-backends-drivers-scanners.
- Multimedia.
  - Brasero,
  - cdparanoia
  - icedax.
  - Pulseaudio-module-bluetooth.
  - Pulseaudio-utils
  - rhythmbox
  - sound-juicer.
  - Sox
  - totem.
  - Totem-mozplugin
  - totem-nautilus
  - wodim

Aplicamos los cambios, en una posterior criba eliminaremos más aplicaciones (en especial firmwares de la carpeta Systems). Antes comprobaremos que reinicia y que funcionan las aplicaciones del Poris Toolkit.

- Kernel-devel
- Kernel-headers.
- System:
  - Aic94xx-firmware
  - ar0170-firmware
  - atmel-firmware
  - ipw2100-firmware

- ipw2200-firmware
- irda-utils
- ivtv-firmware
- iwl1000-firmware
- iwl100-firmware
- iwl3945-firmware
- iwl4565-firmware
- iwl5000-firmware
- iwl5150-firmware
- iwl6000-firmware
- iwl6000g2a-firmware
- iwl6000g2b-firmware
- iwl6050-firmware
- libertas-usb8388
- rt61pci-firmware
- r673usb-firmware

Tras comprobarlo, tomamos una nueva instantánea de la máquina virtual.

Por fin, vamos a eliminar soporte a lenguajes “extra” para aligerar el directorio de “locales” del disco duro. También eliminamos otros soportes no necesarios.

- Package Collections:
  - arabic-support.
  - Armenian-support.
  - Assamese-support.
  - Bengali-support.
  - Bhutanese-support.
  - Burmese-support.
  - Ethiopic-support.
  - Georgian-support.
  - Gujarati-support.
  - Hebrew-support.
  - Hindi-support.
  - Inuktitut-support.
  - Japanese-support.

- Kannada-support.
- Kashmiri-support.
- Khmer-support.
- Konkani-support.
- Korean-support.
- Lao-support.
- Legacy-fonts
- maithili-support.
- Malayalam-support.
- Marathi-support.
- Oriya-support.
- Punjabi-support.
- Russian-support.
- Sanskrit-support.
- Simplified-chinese-support.
- Sindhi-support.
- Sinhala-support.
- Tajik-support.
- Tamil-support.
- Telugu-support.
- Thai-support.
- Traditional-chinese-support.
- Urdu-support.
- Venda-support.
- Yiddish-support.

Tras comprobarlo, tomamos una nueva instantánea de la máquina virtual.

Añadimos las siguientes páginas al arranque de Firefox:

- <http://localhost:3001>
- <http://www.elporis.com>
- <http://hobocentral.net>
- <http://rubyonrails.org>

