Visita nuestro sitio SoloLinux.es

MAGAZINE SOLO LINUX

N° 11

Tu revista, la revista de tod@s

DICIEMBRE 2019

IP dedicada vs **IP compartida**

Diferencias entre **HTTP** y **HTTPS**

Microsoft Team para Linux

Cómo instalar el modo **Kali Undercover** en cualquier Linux

Servidor dedicado vs **servidor cloud**

Instalar **Drupal** en **Ubuntu Server**

MANUALES, SCRIPTS, SOFTWARE, HARDWARE, DISTROS LINUX, SEGURIDAD, REDES Y MUCHO MAS EN LA WEB...

REDACCIÓN:
- Sergio G. B. (Administrador y redactor artículos SoloLinux)
  info@sololinux.es
- Henry G. R. (Redactor artículos SoloWordPress)
  info@solowordpress.es

MAQUETACIÓN Y EDICIÓN:
- Adrián A. A.
  adrian@sololinux.es

Síguenos en las Redes:

SoloLinux Magazine esta realizada con Libre Office Impress 6.2.8.

AGRADECIMIENTOS:
A **Dieguinchi** por la imagen de la publicidad de SOLOWORDPRESS

CONTACTO:
Para cualquier consulta sobre las revistas, publicidad o colaboraciones escribir un email a:
adrian@sololinux.es

Feliz Navidad linuxera y mucho OpenSource para el nuevo año.

Ya estamos en Navidad, ya se termina el año; un año lleno de emociones, sorpresas, y un gran trabajo por parte de los que hacemos que **sololinux.es** sea posible.

De momento ya está online el nuevo sitio **SoloWorpress**, pero no frenamos ahí, no, en el próximo 2020 ademas de las campanadas de fin de año, también sonaran otras para todos los lectores de sololinux.es.

La gran mayoría de proyectos previstos ya están muy avanzados (no todos), vamos lentos pero con paso firme. De momento solo puedo adelantar algunas pinceladas de lo que viene:

- Un nuevo sitio web verá la luz con una temática realmente interesante.
- Un nuevo sitio recopilatorio sobre linux.
- Alquiler de servidores dedicados a precios de risa (totalmente administrados).
- Direcciones de email con una extensión espectacular (para linuxeros).
- Sysadmins de alquiler.
- Proyecto de un nuevo sitio para centralizar revistas online.
- Se quitaran las descargas desde FileHorse (era por evitar abusos), y serán directas para los usuarios de SoloLinux.
- Creamos nuevos canales de Telegram; SoloLinux y SoloWordPress
- Proyecto de foro o chat de ayuda (en este punto nos falta ayuda).
- alguno más que me dejo en el tintero…

Sergio, Adrián y Henry os deseamos…

Feliz Navidad linuxera con mucho OpenSource, para estos días tan emotivos.

Gracias a todos por acompañarnos en está aventura, gracias.

PUBLICIDAD

Quieres poner publicidad en la revista, ahora puedes hacerlo de forma muy simple, llegando a todo el mundo con esta revista digital de software libre y GNU/Linux en ESPAÑOL

CON SOLOLINUX MULTIPLICARAS TUS CLIENTES

Para mayor información escribe un email a: adrian@sololinux.es

COLABORA

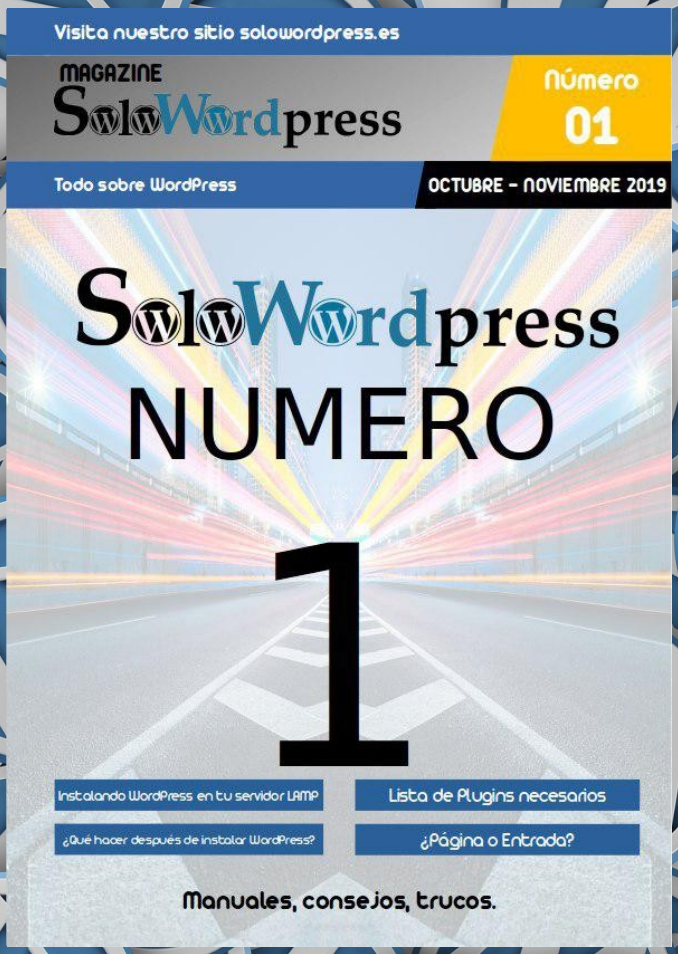
Quieres colaborar en la revista.
Para mayor información escribe un email a: adrian@sololinux.es

La **Revista SOLOLINUX**, se distribuye gratuitamente en forma digital para todo el mundo que quiere disfrutar de ella. Si quieres imprimirla es cosa tuya.

Esta revista es de **distribución gratuita**, si lo consideras oportuno puedes ponerle precio.
Tu también puedes ayudar, contamos con la posibilidad de hacer donaciones para la REVISTA, de manera muy simple a través de **PAYPAL**

AYUDANOS A SEGUIR CRECIENDO

PayPal™ Apoyo con una donación



- Revista bimestral
- Últimas novedades
- Trucos
- Consejos útiles
- Manuales paso a paso
- Debates abiertos
- Opiniones de expertos
- Artículos

¡Si crees que puedes ayudar contacta con nosotros!

SoloWordpress

Tu revista sobre Wordpress

Si el formato digital no te convence, también tenemos todo el contenido en una Página Web

¡VISITANOS!

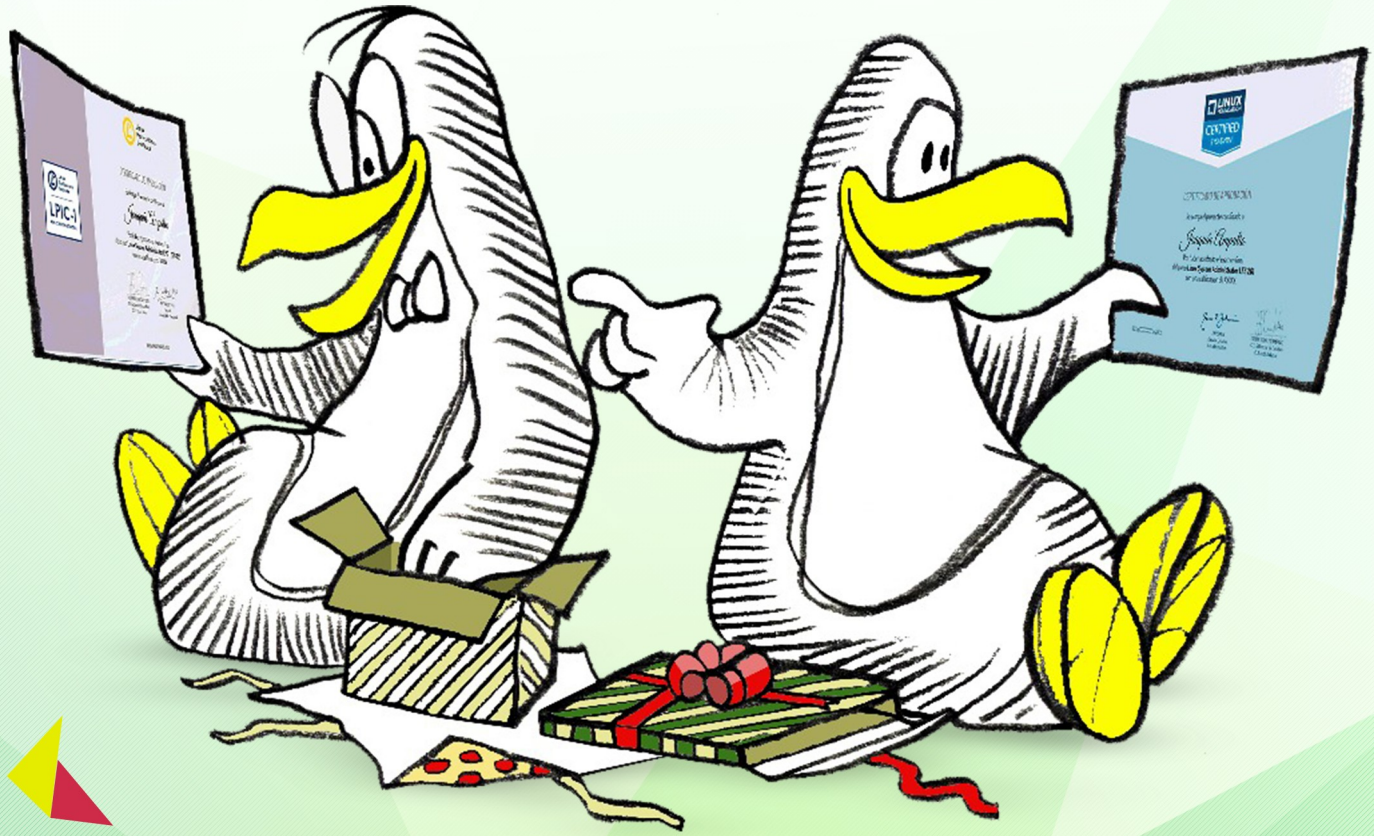
www.solowordpress.es

INSTITUTO LINUX

[UNIVERSIDAD TECNOLOGICA NACIONAL] FRD [FACULTAD REGIONAL DELTA]

Linux Professional Institute APPROVED TRAINING PARTNER

THE LINUX FOUNDATION AUTHORIZED TRAINING PARTNER



ACTUALIZA TU PERFIL LABORAL

Aprende y Certifica LINUX

www.institutolinux.com

+54 9 11 6969 9993

SEGUINOS EN Instagram @fabianampalio

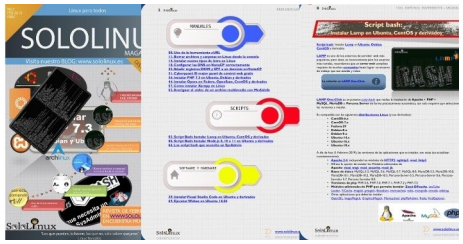
SEGUINOS en INSTAGRAM

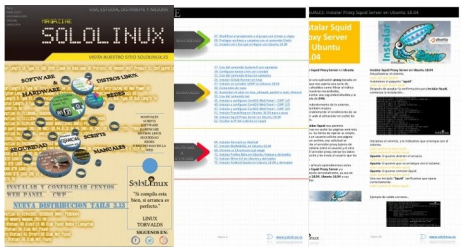
LIVES CON TIPS DE EXÁMENES

Todos los lunes y miércoles

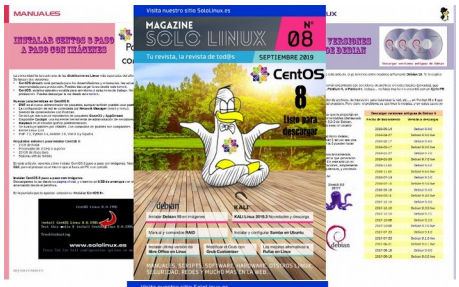
@exameneslinux @fabianampalio www.linkedin.com/in/Fabian-Ampalio

NUESTROS NÚMEROS SOLOLINUX Y SOLOWORDPRESS





En Sololinux.es seguimos creciendo gracias a nuestros lectores, puedes colaborar con el simple gesto de compartir nuestros artículos en tu sitio web, blog, foro o redes sociales.



Descarga la revista SOLOLINUX en PDF:

Download Revista digital – Magazine SoloLinux Nº1
Download Revista digital – Magazine SoloLinux Nº2
Download Revista digital – Magazine SoloLinux Nº3
Download Revista digital – Magazine SoloLinux Nº4
Download Revista digital – Magazine SoloLinux Nº5
Download Revista digital – Magazine SoloLinux Nº6
Download Revista digital – Magazine SoloLinux Nº7
Download Revista digital – Magazine SoloLinux Nº8
Download Revista digital – Magazine SoloLinux Nº9
Download Revista digital – Magazine SoloLinux Nº10

Descarga la revista SOLOWORDPRESS en PDF:

Download Revista digital – Magazine SoloWordpress Nº1

@vantpc vant.pc vantpc_es t.me/vantpc

como cada año, desde 2011,

2020

será

EL AÑO DE LINUX EN EL ESCRITORIO

para nuestros clientes

VANT

#somoslinuxeros

linux inside

la gama más completa de ordenadores linuxeros

descúbrenos en www.vantpc.es

IP dedicada vs IP compartida

Cada nombre de dominio enlaza a una dirección IP, esto quiere decir que en vez de tener que recordar un número (por ejemplo 192.168.155.87), insertamos en el navegador el nombre de dominio y con las **DNS** se redirecciona a la IP correspondiente.
Se nos ofrecen dos opciones: dirección **IP dedicada** y dirección **IP compartida**. En este articulo vemos los pros y los contras de cada una de las opciones.

IP dedicada vs IP compartida

IP Dedicada

La dirección IP dedicada es exclusiva de un dominio o servidor. Es posible que cada sitio web alojado en un servidor tenga su propia dirección IP, lo que quiere decir que dependiendo de la configuración del sistema, es posible acceder al sitio web directamente desde la IP (muy útil durante la propagación del dominio).

La mayoría de los grandes sitios web tienen su IP exclusiva, es normal, la respuesta es mucho más rápida incluso si soportan mucho trafico. También es recomendable en sitios que manejan su propio sistema de pagos, lo tienen más controlado al no depender de aplicaciones de terceros.

Tener una dirección IP dedicada es la mejor opción, debemos tener presente que el flujo de tráfico a una IP en particular siempre será menor que hacia una dirección IP compartida.

IP Compartida

La dirección IP compartida se usa para alojar múltiples sitios web en un solo servidor, es lo que se conoce como **hosting compartido** donde decenas, incluso cientos de sitios web diferentes comparten la misma dirección IP. Esta difundida practica puede tener resultados nefastos para nuestro sitio web, por ejemplo, si uno de los usuarios se dedica a enviar **spam** es evidente que la IP acabara en las **listas negras**, con el consecuente perjuicio causado al resto de usuarios.

Otro detalle importante que tal vez desconozcas, es que algunos países bloquean este tipo de direcciones IP, tu piensas que la web se visualiza a nivel mundial pero no es así. Esta claro que la IP compartida no es lo más recomendable, su único beneficio es el costo económico (mucho mas barato).

Beneficios de la IP Dedicada

Estabilidad:
Al usar una dirección dedicada es fácil vigilar la reputación de nuestro sitio web, y como comentamos anteriormente no se verá influenciada por una acción dañina de otro usuario. Una ip que se encuentra en listas negras, no solo tiene el problema de que los mails acabaran en la carpeta de span, los motores de búsqueda también las tienen en cuenta y pueden dañar gravemente tu reputación con el prejuicio que eso supone.

Correo mail:
Si una dirección IP Compartida está listada en las listas negras por el envío de correo no deseado, el servicio de mail asociado a la ip compartida se vera afectado de manera grave. La dirección IP dedicada utilizada para el servicio de correo electrónico evita que tu servicio aparezca en las listas negras, siempre y cuando no realices tareas oscuras.

Inactividad:
Al utilizar una IP dedicada en tu propio servidor o VPS, eres tu quien la controla. Te evitaras tiempos de inactividad por culpa de otros usuarios, recuerda que los sitios web consumen mucha memoria ram del servidor, cientos de usuarios, cuentas y sitios web, equivalen a tiempos de carga excesivos, mientras que la dirección dedicada ofrece una carga constante del sitio web.

Identidad:
Una dirección IP dedicada genera confianza, sobre todo en tiendas online. La impresión de un usuario al percatarse que haces uso de una IP dedicada es muy buena, ofreces confiabilidad, una identidad propia, y como comentamos anteriormente, confianza, mucha confianza.

A los clientes les gustaría tratar con dichos sitios web. Crea una impresión entre los visitantes y clientes y establece una identidad comercial separada. Si el sitio web no maneja servicios de pago de terceros como **PayPal, Authorize.net, Google Checkout, etc.**

SoloLinux

Conclusión final

La verdad, una IP compartida no se puede comparar con una dedicada, si es posible te recomiendo que siempre elijas la IP dedicada, ofrece grandes ventajas sobre la compartida. Nunca olvides que por ejemplo **Google**, cada día considera más importante la velocidad de un sitio web como parte fundamental para clasificar tu sitio web en los resultados de búsqueda.

Reducir el tamaño de un archivo PDF

Hoy vemos un **script bash** capaz de reducir el tamaño de cualquier archivo pdf, creado por **Alfred Klomp** utiliza **Ghostscript** para modificar los archivos.

Su uso es bastante simple y por defecto reduce los pdf a 72 ppp, de todas formas es posible modificar este valor por el que más nos interese.

Reducir el tamaño de un archivo PDF

Creamos el script.

```
nano redu-pdf.sh
```

Copia y pega lo siguiente:

```
#!/bin/sh

# http://www.alfredklomp.com/programming/shrinkpdf
# Licensed under the 3-clause BSD license:
#
# Copyright (c) 2014-2019, Alfred Klomp
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
# 1. Redistributions of source code must retain the above copyright notice,
#    this list of conditions and the following disclaimer.
# 2. Redistributions in binary form must reproduce the above copyright notice,
#    this list of conditions and the following disclaimer in the documentation
#    and/or other materials provided with the distribution.
# 3. Neither the name of the copyright holder nor the names of its contributors
#    may be used to endorse or promote products derived from this software
#    without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
# AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
# ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
# LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
# CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
# SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
# INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
# CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
# POSSIBILITY OF SUCH DAMAGE.


shrink ()
{
 gs \
   -q -dNOPAUSE -dBATCH -dSAFER \
   -sDEVICE=pdfwrite \
   -dCompatibilityLevel=1.3 \
   -dPDFSETTINGS=/screen \
   -dEmbedAllFonts=true \
   -dSubsetFonts=true \
   -dAutoRotatePages=/None \
   -dColorImageDownsampleType=/Bicubic \
   -dColorImageResolution=$3 \
   -dGrayImageDownsampleType=/Bicubic \
   -dGrayImageResolution=$3 \
   -dMonoImageDownsampleType=/Subsample \
   -dMonoImageResolution=$3 \
   -sOutputFile="$2" \
   "$1"
}
```

OJO!!!! CONTINUA...

```
check_smaller ()
{
        # If $1 and $2 are regular files, we can compare file sizes to
        # see if we succeeded in shrinking. If not, we copy $1 over $2:
        if [ ! -f "$1" -o ! -f "$2" ]; then
                return 0;
        fi
        ISIZE="$(echo $(wc -c "$1") | cut -f1 -d\ )"
        OSIZE="$(echo $(wc -c "$2") | cut -f1 -d\ )"
        if [ "$ISIZE" -lt "$OSIZE" ]; then
                echo "Input smaller than output, doing straight copy" >&2
                cp "$1" "$2"
        fi
}

usage ()
{
        echo "Reduces PDF filesize by lossy recompressing with Ghostscript."
        echo "Not guaranteed to succeed, but usually works."
        echo "  Usage: $1 infile [outfile] [resolution_in_dpi]"
}

IFILE="$1"

# Need an input file:
if [ -z "$IFILE" ]; then
        usage "$0"
        exit 1
fi

# Output filename defaults to "-" (stdout) unless given:
if [ ! -z "$2" ]; then
        OFILE="$2"
else
        OFILE="-"
fi

# Output resolution defaults to 72 unless given:
if [ ! -z "$3" ]; then
        res="$3"
else
        res="72"
fi

shrink "$IFILE" "$OFILE" "$res" || exit $?

check_smaller "$IFILE" "$OFILE"
```

Anteriormente comentamos, que por defecto el archivo de salida se genera en 72 ppp, si quieres modificar el valor debes indicarlo al final del comando.

Por ejemplo…
./redu-pdf.sh archivo1.pdf archivo2.pdf 85

./redu-pdf.sh archivo1.pdf archivo2.pdf 95

./redu-pdf.sh archivo1.pdf archivo2.pdf 105

Este script es publicado bajo la licencia BSD de 3 cláusulas.

Guarda el archivo y cierra el editor.

Le concedemos los permisos necesarios.

```
sudo chmod +x redu-pdf.sh
```

Ejecutar el script, reducir el tamaño de un PDF
Su uso es bastante simple, observa dos formas diferentes de ejecutarlo:

```
./redu-pdf.sh archivo1.pdf archivo2.pdf
```

también es valido….

```
bash redu-pdf.sh archivo1.pdf archivo2.pdf
```

Instalar Drupal en Ubuntu Server

Junto con **WordPress** y **Joomla**, **Drupal** es uno de los **CMS** más utilizados, lógico, es buenísimo, altamente seguro y de código abierto.

En este articulo instalaremos un servidor para Drupal desde cero en **Ubuntu Server 18.04 LTS** (es valido para sus derivados y otras versiones), con las siguientes características: **MySQL, PHP-FPM 7.2 y Nginx**. Solo debes seguir los pasos indicados uno por uno.

Instalar Drupal en Ubuntu Server

Antes de comenzar, actualizamos el sistema e instalamos el descompresor **unzip**.

```
sudo apt update && sudo apt upgrade
sudo apt install unzip
```

Instalar y configurar MySQL

Ahora procedemos a instalar MySQL o MariaDB, nuestro Ubuntu 18.04 viene por defecto con MySQL.

```
sudo apt-get install mysql-server
```

Aseguramos la instalacion:

```
mysql_secure_installation
```

Responde a las siguientes preguntas:

- Configurar la contraseña de root. **N**
- Eliminar usuarios anónimos. **Y**
- Deshabilitar el inicio de sesión remoto para el usuario root. **Y**
- Borrar la base de datos demo y los accesos. **Y**

Puedes configurar una **password** especifica o no, es tu decisión. En el resto de preguntas responde siempre «Y», esto eliminará los usuarios anónimos, la base de datos demo, deshabilitará los inicios de sesión remotos, y cargará las nuevas reglas de MySQL.

La base de datos ya esta configurada y lista para ser usada. Se supone que debería estar corriendo, por si acaso la iniciamos manualmente y habilitamos su arranque con el sistema.

```
systemctl start mysql
systemctl enable mysql
```

Bien… nos falta crear una base de datos para Drupal y su usuario. Nosotros insertaremos como usuario de MySQL el nombre drupal, no te olvides de insertar una contraseña segura.

Abrimos la consola de MySQL.

```
sudo mysql
```

Linea por linea ejecuta lo siguiente:

```
mysql> CREATE DATABASE drupal;

mysql> GRANT ALL PRIVILEGES ON drupal.* TO 'drupal'@'localhost' IDENTIFIED BY 'mi-password-seguro';

mysql> FLUSH PRIVILEGES;

mysql> \q
```

Instalar y configurar PHP

Vamos con el php, en este caso instalaremos la versión 7.2.

```
sudo apt install php7.2-cli php7.2-fpm php7.2-mysql php7.2-json php7.2-opcache php7.2-mbstring php7.2-xml php7.2-gd php7.2-curl
```

Al instalar Drupal necesitamos unos requisitos un poco superiores a WordPress, aplicamos los mínimos en el archivo php.ini desde nuestra terminal (puedes aumentarlos si es necesario)

```
sudo sed -i "s/memory_limit = .*/memory_limit = 256/" /etc/php/7.2/fpm/php.ini

sudo sed -i "s/upload_max_filesize = .*/upload_max_filesize = 128M/" /etc/php/7.2/fpm/php.ini

sudo sed -i "s/post_max_size = .*/post_max_size = 128M/" /etc/php/7.2/fpm/php.ini

sudo sed -i "s/max_execution_time = .*/max_execution_time = 3000/" /etc/php/7.2/fpm/php.ini
```

Instalar y configurar Nginx

Llego el momento de instalar Nginx, no perdamos más tiempo.

```
sudo apt install nginx
```

Creamos el archivo de configuración de nuestro sitio web Drupal.

```
sudo nano /etc/nginx/sites-available/tu-dominio.com
```

Copia y pega lo siguiente, OJO!!!, con tu nombre de dominio.

```
server {
    listen 80;
    server_name tu-dominio.com www.tu-dominio.com;
    root /var/www/tu-dominio.com;
    index index.html index.htm index.php;
    charset utf-8;
    access_log /var/log/nginx/tu-dominio.com.access.log;
    error_log /var/log/nginx/tu-dominio.com.error.log info;

    location ~ \..*/.*\.php {
        return 403;
    }
    location ~ ^/sites/.*/private/ {
        return 403;
    }
    location ~ ^/sites/[^/]+/files/.*\.php$ {
        deny all;
    }
    location ~ (^|/)\. {
        return 403;
    }

    location / {
        try_files $uri /index.php?$query_string;
    }
    location @rewrite {
        rewrite ^/(.*)$ /index.php?q=$1;
    }
    location ~ /vendor/.*\.php$ {
        deny all;
        return 404;
    }
    location = /favicon.ico { access_log off; log_not_found off;
}
    location = /robots.txt  { access_log off; log_not_found
off; }
    location ~ '\.php$|^/update.php' {
        fastcgi_pass unix:/var/run/php/php7.2-fpm.sock;
        fastcgi_index index.php;
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
        include /etc/nginx/fastcgi.conf;
    }
    location ~ ^/sites/.*/files/styles/ {
        try_files $uri @rewrite;
    }
    location ~ ^(/[a-z\-]+)?/system/files/ {
        try_files $uri /index.php?$query_string;
    }
    location ~* \.(js|css|png|jpg|jpeg|gif|ico|svg)$ {
        try_files $uri @rewrite;
        expires max;
        log_not_found off;
    }
    location ~ /.(?!well-known).* {
        deny all;
    }
}
```

Guarda el archivo y cierra el editor.

Creamos un enlace simbólico para no permitir el acceso al archivo original.

```
sudo ln -s /etc/nginx/sites-available/tu-dominio.com
/etc/nginx/sites-enabled/tu-dominio.com
```

Verificamos que la configuración es valida y reiniciamos el servicio.

```
sudo nginx -t
sudo systemctl restart nginx
```

Una vez concluido el ultimo paso, ya tenemos listo nuestro servidor web. Te recomiendo reiniciar el sistema antes de instalar Drupal.

```
reboot
```

Instalar Drupal

Descargamos la ultima versión estable de Drupal, en este caso la 8.8.0 (puedes revisar si existen nuevas versiones aquí).

```
wget https://ftp.drupal.org/files/projects/drupal-8.8.0.zip -O
drupal.zip
```

Descomprimimos el archivo descargado y lo movemos al directorio de nuestro sitio web.

```
sudo unzip drupal.zip
sudo mv drupal-8.8.0/ /var/www/tu-dominio.com
```

No olvides conceder los permisos necesarios, es importante.

```
sudo chown -R www-data: /var/www/tu-dominio.com
```

El ultimo paso es acceder desde tu navegador web a la url del dominio, y finalizar el proceso de instalación.

Te recomiendo que revises la documentación de Drupal, no te arrepentirás.

Instalar un servidor Redis en Ubuntu y derivados

Redis es un almacén de estructura de datos en memoria cache de código abierto, utilizado como servidor de base de datos, caché y agente de mensajes. Admite varias estructuras de datos como por ejemplo, cadenas, hashes, listas, conjuntos, y muchas más.

Implantar tu propio **servidor Redis** es saltar a un nivel superior, ademas de ofrecer un rendimiento excelente también nos proporciona una alta disponibilidad con Redis Sentinel, monitoreo, notificaciones failover, e incluso el **particionado** automático de varios nodos.

En este articulo no nos limitaremos a instalar Redis en un servidor que contiene nuestros sitios webs, lo que vamos a realizar es montar un servidor exclusivo para Redis, y que admita las conexiones entrantes de otras maquinas.

Instalar un servidor Redis en Ubuntu

Esta herramienta viene en los repositorios de la mayoría de distribuciones linux, así que directamente actualizamos e instalamos Redis.

```
sudo apt update
sudo apt install redis-server
```

Una vez instalado, lo iniciamos y habilitamos que arranque con el sistema.

```
sudo systemctl start redis-server
sudo systemctl enable redis-server
```

Verificamos su funcionamiento.

```
sudo systemctl status redis-server
```

Ejemplo de salida correcta…

```
redis-server.service – Advanced key-value store
   Loaded: loaded (/lib/systemd/system/redis-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2019-12-04 19:02:30 CST; 15s ago
   Docs: http://redis.io/documentation,
         man:redis-server(1)
Main PID: 26589 (redis-server)
   Tasks: 4 (limit: 4674)
   CGroup: /system.slice/redis-server.service
           └─26589 /usr/bin/redis-server 127.0.0.1:6379
```

Configurar un servidor Redis

Ahora que lo tenemos instalado y verificado, sigue los pasos que te indico para configurar el inicio de sesión remoto y el firewall de Ubuntu.

Como ya comentamos anteriormente, de forma predeterminada Redis no admite el acceso desde una ubicación remota (un servidor, cliente, etc…). El acceso está restringido para su uso exclusivo en el host local, donde instalaste la aplicación (127.0.0.1).

Si tus aplicaciones o sitios web están instalados en el mismo servidor que **Redis Server**, no necesitas acceso remoto para nada. Si tus requerimientos son superiores debes montar un servidor exclusivo para Redis, sigue los pasos que vemos a continuación para permitir el acceso remoto.

Abrimos el archivo de configuración.

```
sudo nano /etc/redis/redis.conf
```

Busca la linea siguiente:

```
# IF YOU ARE SURE YOU WANT YOUR INSTANCE TO LISTEN TO ALL THE INTERFACES
# JUST COMMENT THE FOLLOWING LINE.
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
bind 127.0.0.1 ::1
```

Ahora modifica la linea «bind 127.0.0.1 ::1», tal como te indico.

```
# IF YOU ARE SURE YOU WANT YOUR INSTANCE TO LISTEN TO ALL THE INTERFACES
# JUST COMMENT THE FOLLOWING LINE.
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
bind 0.0.0.0 ::1
```

Guarda el archivo y cierra el editor.

Reinicia Redis.

```
sudo systemctl restart redis-server
```

Por defecto Redis utiliza el puerto «6379», te recomiendo que verifiques si escucha o no.

```
ss -an | grep 6379
```

Ejemplo de salida…

```
tcp    LISTEN  0    128    0.0.0.0:6379    0.0.0.0:*
tcp    LISTEN  0    128    [::1]:6379    [::]:*
```

Si estas ejecutando el firewall UFW puedes conceder permisos a una ip.

```
sudo ufw allow proto tcp from 192.168.0.56 to any port 6379
```

Tal vez te interese conceder permisos a una subnet.

```
sudo ufw allow proto tcp from 192.168.0.0/24 to any port 6379
```

Una vez concluido todo el proceso te recomiendo que reinicies tu servidor.

```
reboot
```

Cómo desinstalar aplicaciones en Ubuntu correctamente

Ubuntu, Linux Mint, y otras distribuciones. Tienen la particularidad de ser muy fáciles de usar incluso para los recién llegados a linux.

Como no podía ser menos, desinstalar aplicaciones a través de su interfaz gráfica también es muy sencillo. Pero existe un problema con este método, al igual que en otros sistemas operativos no existe una herramienta capaz de borrar totalmente una aplicación.

Es prácticamente imposible borrar todo el rastro de una aplicación, siempre queda algún registro o una cadena perdida.

En este articulo vemos como borrar una aplicación en Ubuntu incluyendo sus bibliotecas y carpetas ocultas, siempre quedara más limpio que si lo haces con la herramienta GUI de Ubuntu o Linux Mint (se dejan demasiada basura inútil).

Cómo desinstalar aplicaciones en Ubuntu

Para lograr nuestro objetivo debemos operar desde la **terminal / consola**. En este ejemplo tenemos instalada la herramienta de diseño y dibujo, **Gimp 2.8**.

Independientemente del software a desinstalar, debes utilizar el siguiente comando (para otras herramientas sustituyes gimp por lo que corresponda).

```
sudo apt-get --purge remove gimp
```

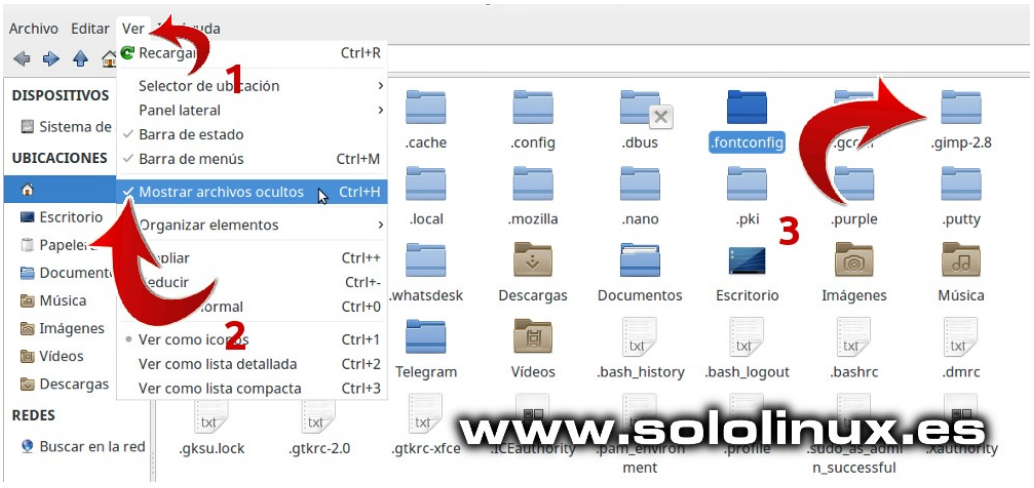
Siempre quedan librerías y dependencias innecesarias en el sistema. Copia, pega y ejecuta el siguiente comando:

```
sudo apt-get autoremove
```

Bien ya casi lo tenemos, solo nos falta borrar la carpeta oculta que tenemos en nuestra carpeta de usuario.

Por defecto este tipo de carpetas están ocultas, pero tranquilo… que sean visibles es tarea sencilla. Abre tu administrador de archivos y sigue los pasos indicados en la imagen.

1. Hacemos click en «VER» o similar.
2. En el desplegable marca la opción «Mostrar archivos ocultos».
3. Aparecen las carpetas ocultas, elimina la que te interesa.



Hemos desinstalado una aplicación correctamente.

Servidor Dedicado vs Servidor Cloud

Tu sitio web va creciendo, cada vez tienes más visitas, ya es insoportable, llego el momento de tomar una decisión drástica. Decides abandonar tu **hosting** o **VPS** y lanzarte al vació.

Cambiar de un hosting compartido a un VPS no supone casi ningún prejuicio, saltar a un servidor cloud y sobre todo a un dedicado ya es otra cosa; no solo en tema económico que también es importante. Cuando se te presenta el momento tienes dudas, muchas dudas.

En este articulo trataremos de disiparlas y que elijas la opción que más te convenga, siempre según mi criterio y experiencia personal.

Servidor Dedicado vs Servidor Cloud

Antes de comenzar el articulo, quiero advertirte que ninguna de las dos opciones es valida si tienes un pequeño sitio web o blog personal, sería tirar el dinero. Tampoco nos referimos a pequeñas instancias cloud de 8€ o servidores dedicados tipo Kimsufi, la comparativa la hacemos para sitios con necesidades medias, así que vamos a ello.

Servidor Cloud

Estoy seguro del potencial de los servidores cloud, es el futuro, pero de momento no. Este tipo de servicio está compuesto por muchos servidores físicos que se localizan en un uno o varios centros de datos (normalmente en uno), que alojan varios servidores virtuales.

Gracias a las nuevas tecnologías de computación en la nube, este sistema permite que un número ilimitado de máquinas se comporten como si fueran un único sistema. Permiten definir tus necesidades actuales, como núcleos del procesador, memoria ram, tamaño del disco, etc. Son fáciles de distinguir porque normalmente su precio lo establecen por horas, dependiendo de tu selección el precio será uno u otro.

Lo que me gusta de este sistema es que es escalable, puedes aumentar o disminuir tus requerimientos de hardware sin la intervención física de nadie, solo tu, desde tu panel de control. Lo que no me gusta… no tienes el control de la maquina, es como si tienes varios vps en uno, eso es realmente un servidor cloud. También tiene su lado positivo, si uno falla o esta lleno salta a otro, aun así no me acaba de convencer, ademas si necesitas muchos recursos el coste puede ser tremendo.

Tampoco me gusta como opera el manejo de la ram y cpu, yo mismo he verificado que al requerir más de 128GB de ram el sistema era inestable, supongo que la causa es que una maquina se quedo sin memoria y solicitaba recursos de hardware a otra, pero no estoy seguro (no me respondieron, ja).

Servidor Dedicado

Cuando hablamos de dedicamos, nos referimos a potencia y tradición. Hace muchos años que existen y el concepto no a cambiado, es tu maquina, solo para ti, sin intervenciones ni accesos ajenos. Tu manejas el **hardware** disponible, tu sistema y su seguridad. Personalmente es mi elección, pero claro, tiene sus pegas.

Los servidores dedicados son únicos, si falla el hardware se cae el sistema (puedes doblar servidores pero se encarece considerablemente). Otro punto importante a tener en cuenta es el trafico, normalmente el trafico de los dedicados es ilimitado o respetuoso (algunos datacenter denominan respetuoso a que tampoco abuses).

Este tipo de servidores no es recomendado para usuarios noveles (a no ser que contrates un **sysadmin**), ten en cuenta que el mínimo error puede romper el sistema. A prestaciones similares puedes encontrar buenos precios en dedicados y disculpa que insista… no comparte nada con nadie.

Si te decides por alquilar un servidor dedicado, analiza bien los consumos actuales y el predecible en el futuro, si más adelante necesitas más hardware te veras obligado a cambiar de maquina.

Conclusión final

Es difícil, muy difícil aconsejar sobre la elección ideal.

Si eres de los que estas dando el salto ahora, mi recomendación seria un servidor cloud (sobre todo por que es escalable y te permite valorar los recursos necesarios reales). Si tienes uno o varios sitios con miles de visitas y peticiones diarias, yo me decanto por el dedicado, si tanto creces lo puedes doblar o incluso alquilar tu propia jaula.

Recuerda que este articulo se basa en mi experiencia laboral diaria, y no tengo predilección por ningún vendedor, datacenter ni similar.

La decisión final te corresponde a ti.

Instalar Apache Maven en Debian 10

Apache Maven es una herramienta especializada en la creación y gestión de proyectos Maven es una herramienta de software para la gestión y construcción de proyectos **Java**. Su principal beneficio es que estandariza la configuración de nuestro proyecto en todas sus fases, desde la compilación y empaquetado hasta la instalación y administración de las librerías necesarias.

Maven fue creado en el 2002 por **Jason van Zyl**, y es la base de los compiladores **IDES** actuales, como **Eclipse**, **NetBeans**, etc. Antes de comenzar su instalación vemos sus principales características.

- Excelente sistema de gestión dependencias.
- Mecanismo de distribución de librerías desde el repositorio local de Maven, hacia los repositorios publicados en Internet o red local.
- Permite la creación de plugins customizables.
- Es multi-plataforma.
- **Open Source**, el código está disponible para que lo modifiques si es necesario.
- Los repositorios oficiales y públicos de software libre, ofrecen librerías que toda la comunidad de desarrolladores pueden utilizar.
- Es compatible con muchos IDEs.

Instalar Apache Maven en Debian 10
Actualizamos el sistema.

```
apt upgrade
apt upgrade -y
```

Necesitamos instalar el paquete de desarrollo de java (OpenJDK 11), la herramienta wget y GIT.

```
apt install -y default-jdk
apt install -y wget git
```

Apache Maven necesita la variable de entorno $JAVA_HOME, la incluimos en nuestro sistema.

```
echo "export JAVA_HOME=/lib/jvm/default-java" >>
/etc/profile
```

Bien, ya tenemos nuestro sistema listo para la instalación, comenzamos importando las keys publicas.

```
cd /tmp
wget https://www.apache.org/dist/maven/KEYS
gpg --import KEYS && rm KEYS
```

Ahora descargamos el archivo binario de la aplicación y el de la firma. Actualmente la ultima versión estable es la 3.6.3, puedes comprobar si existe alguna actualización en su pagina oficial.

```
wget -O maven.tgz https://www-eu.apache.org/dist/maven/
maven-3/3.6.3/binaries/apache-maven-3.6.3-bin.tar.gz

wget -O maven.tgz.asc https://www.apache.org/dist/maven/
maven-3/3.6.3/binaries/apache-maven-3.6.3-bin.tar.gz.asc
```

Verificamos el paquete con el archivo de firmas.

```
gpg --verify maven.tgz.asc maven.tgz
```

Descomprimimos el archivo y lo movemos a su directorio final.

```
tar -xzf maven.tgz
rm maven.tgz*
mv apache-maven* /opt/maven
```

Agregamos **/opt/maven/bin** a la variable de entorno.

```
echo "export PATH=$PATH:/opt/maven/bin" >> /etc/profile
```

Como punto final debes recargar las variables de entorno.

```
./etc/profile
```

Verifica que la instalación es correcta.

```
mvn -v
```

ejemplo de salida…

```
Apache Maven 3.6.3
(81f7586969310zs1dc0d5t7yc0dc55gtrd0p519; 2019-12-
05T19:17:21+01:00)
Maven home: /opt/maven
Java version: 11.0.5, vendor: Debian, runtime:
/usr/lib/jvm/java-11-openjdk-amd64
Default locale: en_US, platform encoding: UTF-8
OS name: «linux», version: «4.19.0-6-amd64», arch:
«amd64», family: «unix»
```

Ya puedes ejecutar Maven.

Si tienes dudas te recomiendo revisar su guía oficial, es muy buena.

Verificar la reputación de un dominio y email con Postmaster Google

Que nuestros correos vayan a la carpeta **spam** es algo que nos preocupa a todos, desde el usuario que envía dos o tres correos diarios, hasta el que envía 10.000 a la hora.

Hoy en día, uno de los correos gratuitos más populares es el de **Gmail** (Google), y precisamente suele ser el que mas problemas tiene a la hora de recibir mails. Al ser tan popular Google desconfía de todo el mundo, jeje, por que la mitad de ellos entra directamente a la carpeta de spam.

Bueno, tal vez no sabias que **Google** ofrece una herramienta donde puedes verificar tu dominio, emails enviados que se marcaron como **spam**, y un largo etcétera. Esta aplicación se llama **Google Postmaster Tools**, y nos ayudara a identificar el problema de nuestro **email**.

Postmaster Tools nos listara lo siguiente:

- Tasa de spam.
- Reputación de las IP.
- Reputación del dominio.
- Bucle de retroalimentación (FBL).
- Autentificación.
- Cifrado.
- Errores en la entrega.

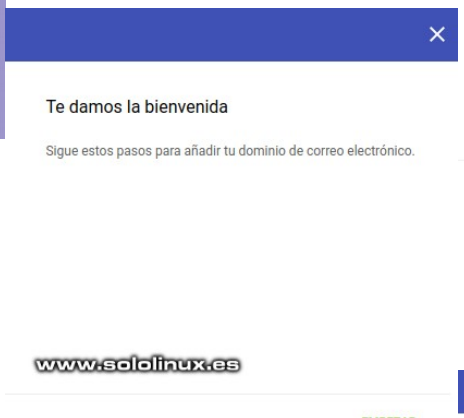
En este articulo vemos con imágenes como operar y los resultados de esta excelente herramienta que nos brinda Google.

Verificar la reputación de un dominio con Postmaster Tools

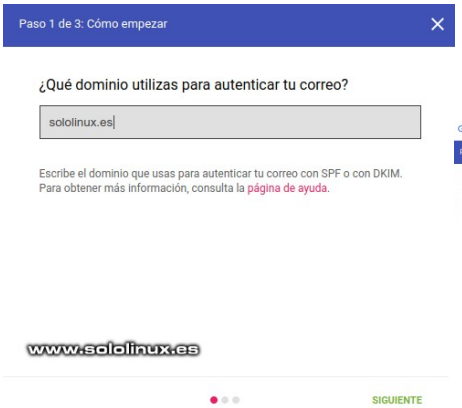
Desde tu navegador web favorito accedes a la siguiente dirección url (debes haber iniciado antes sesión en Google o Gmail).

https://postmaster.google.com/

Veras la pantalla de bienvenida, pulsa en empezar.



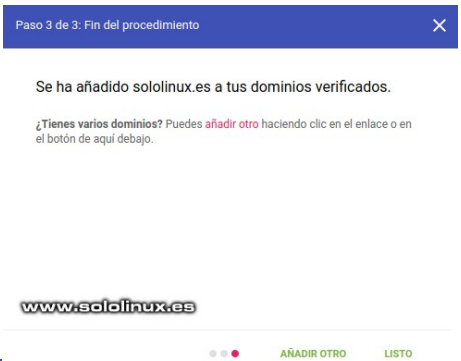
Nos solicita el dominio desde el que mandamos nuestros correos, en nuestro caso los boletines de noticias se envían desde nuestro dominio (sololinux.es), nos viene perfecto como ejemplo pues se envían miles de emails diariamente. Una vez introducido el dominio, pulsa en siguiente.



Ahora se requiere incluir un registro txt a la configuración DNS, también permite un CNAME pero es más sencillo el registro txt. Cuando termines con el registro pulsa en verificar.

Nota: Google ya estaba registrando datos antes de verificar el dominio

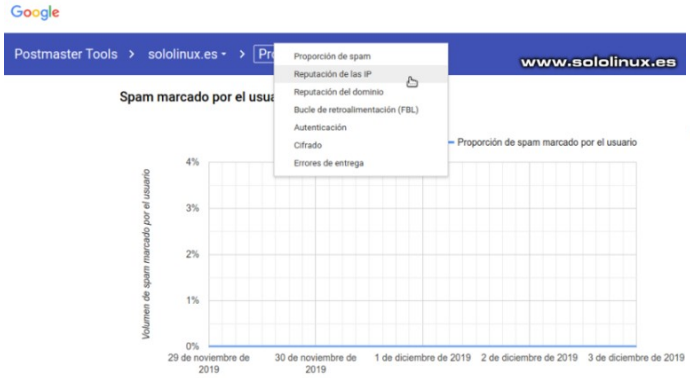
Bien, ya está verificado. Nos pregunta si queremos añadir otro dominio, en nuestro caso no nos interesa, pulsamos en el botón **LISTO**.



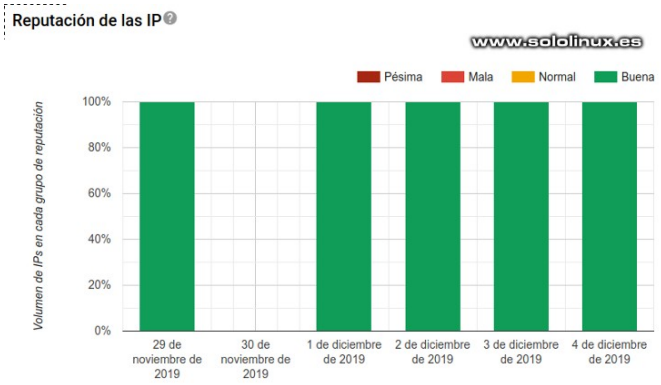
Nos aparece la tabla de dominios, hacemos clic sobre el que nos interese.

De manera predeterminada aparece la primera ventana, en ella nos aparece el porcentaje de **usuarios que marcaron tu email como spam**. Si observas donde la flecha roja veras un pequeño indicador, haz clic en el.

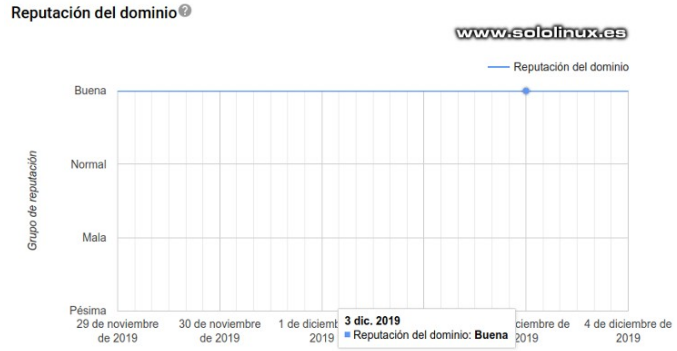
Nos aparece un desplegable con todos los tableros informativos, selecciona el que quieras. Nosotros los veremos todos.



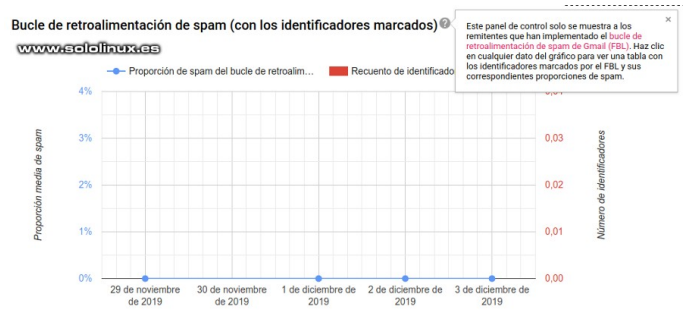
Aquí vemos el tablero de la reputación IP, vemos que es buena.



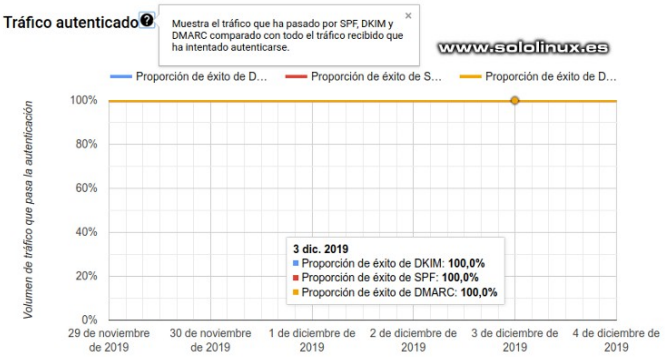
En el siguiente tablero vemos la reputación del dominio, también es bueno.



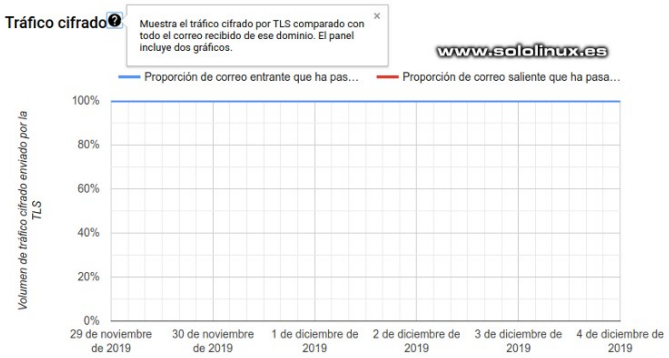
Llegamos al **Bucle de retroalimentación de spam** que paso a explicar porque tiene su miga. El Feedback Loop de Google no funciona como uno tradicional; no existe ninguna dirección FBL que lo ofrezca, por tanto tu no recibes copias de los correos que generaron las quejas de los usuarios. Sin embargo, al insertar un encabezado del tipo Feedback-ID en nuestros mails salientes, si podríamos rastrear los datos de FBL en nuestro tablero.



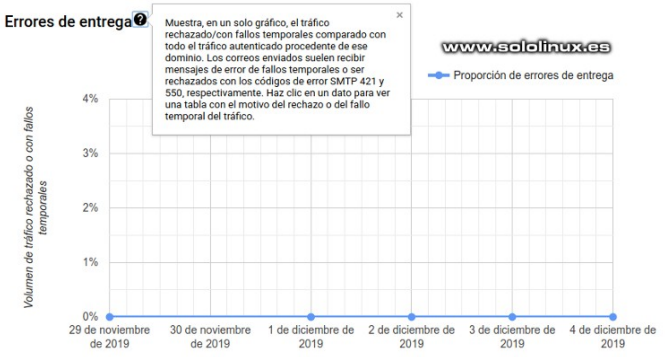
Llegamos a la opción **Autenticación**: en esta tabla se mide si se superaron las verificaciones de autentificación, SPF, DKIM y DMARC.



En el **Trafico cifrado** se muestra el porcentaje del correo enviado con **cifrado TLS** entre nuestro servidor y los de Google o Gmail.



La ultima opción es **Errores de entrega**, en ella se nos muestra el porcentaje de tráfico (mail) rechazado comparado con todo el tráfico autenticado procedente del dominio. Los correos enviados suelen recibir mensajes de error con los fallos temporales, o su rechazado indicando los códigos de error SMTP 421 y 550.



En próximos artículos veremos otras herramientas como por ejemplo **mailtester**, pero si tu problema está con **Gmail** tu herramienta es **Google Postmaster Tools**.

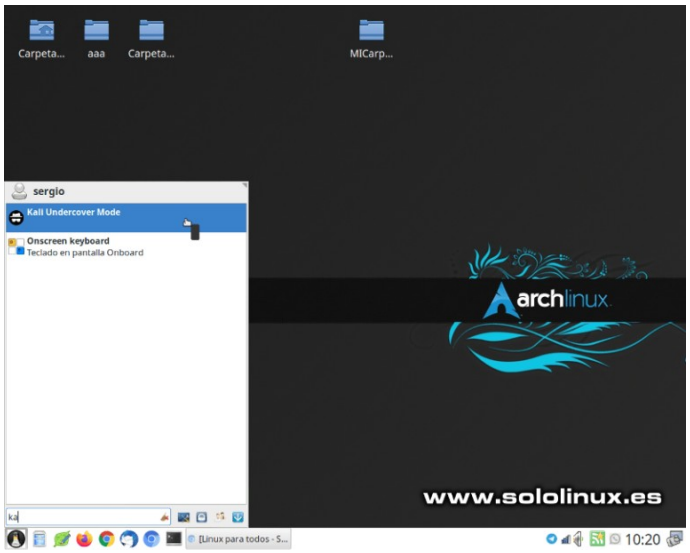
Cómo instalar el modo Kali Undercover en cualquier linux

Si recuerdas el articulo donde presentamos la ultima versión de Kali, vimos la novedosa opción anti-miradas **Kali Undercover**. Tal vez te resulte gracioso lo de anti-miradas, pero la verdad es que es muy útil.

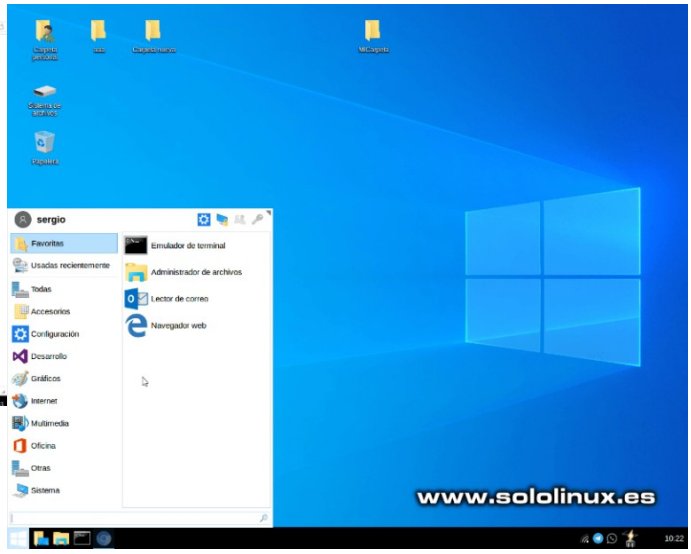
Estas a la espera de un vuelo, en el tren, en un restaurante, y tu con **Kali linux**. Lo bueno de esto, es, que solo tu sabes el sistema operativo que utilizas, las miradas indiscretas pensaras que eres otro de tantos con su **windows 10**. Comprobado personalmente con resultados óptimos, jajaj.

Con esta herramienta, obtienes una apariencia prácticamente igual a windows (modo encubierto de Kali), y tan solo haciendo click en un icono que encontraras en tu menú de aplicaciones. Lo bueno del **modo Kali Undercover** es que lo puedes instalar en cualquier **distribución linux** siempre que utilices el entorno de escritorio **XFCE**. Observa el antes y el después en las siguientes imágenes.

Kali Undercover sin habilitar:



Kali Undercover habilitado:



Impresionante verdad?. En este articulo vemos como instalar **Kali Undercover** en cualquier distribución linux que haga uso del entorno de escritorio XFCE.

Cómo instalar el modo Kali Undercover

Antes de comenzar la instalación conocemos sus principales características.

- **Fabuloso tema imitando Windows 10 creado en GTK por B00merang-Project.**
- **Iconos como los de Windows 10.**
- **Fondo de pantalla de Windows 10.**
- **Opera con un script Bash y dos Python que modifican temporalmente algunas configuraciones del escritorio**
- **Perfil personalizado de Xfce.**
- **Configuración del menú personalizada como en Windows.**

Instalar Kali Undercover

Si eres usuario de Ubuntu, Debian, Linux Mint y derivados, estás de suerte. Solo debes abrir está pagina y descargar la ultima versión, en nuestro ejemplo kali-undercover_2020.1.0_all.deb.

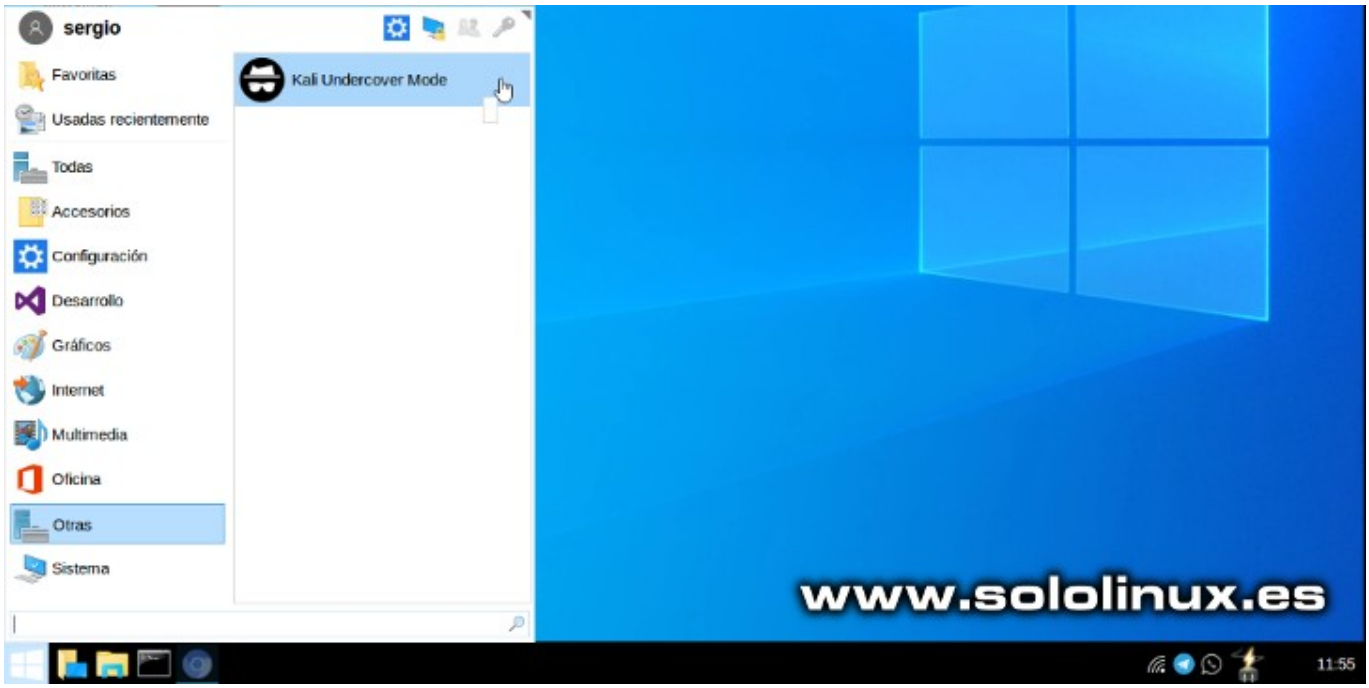
Una vez descargado el archivo haces clic para instalar. y listo.

Instalar el modo Kali Undercover en otras distribuciones linux, también es tarea sencilla, lo único es asegurarte de que tienes instaladas las siguientes dependencias (verifico que en Manjaro vienen instaladas por defecto).
fonts-liberation, gir1.2-glib -2.0, libnotify-bin, procps, psmisc, xfce4, xfce4-power-manager-plugins, xfce4-pulseaudio-plugin y xfce4-whiskermenu-plugin

Ademas necesitas tener instalado GIT, si no recuerdas como revisa esta anterior articulo. Una vez instalado GIT, clonamos el repositorio oficial.
git clone https://gitlab.com/kalilinux/packages/kali-undercover

Desde la consola /terminal abre la carpeta donde clonaste el repositorio, y ejecutas los siguientes comandos.
sudo cp -r share /usr/
sudo cp bin/kali-undercover /usr/bin/

Ya lo tienes instalado, ahora tan solo debes ejecutar el icono desde tu menú de aplicaciones, para salir del **modo Kali Undercover** debes operar de la misma forma. Desde el menú de aplicaciones de tu flamante **Windows 10**, jajajaj, pulsas en **otras**, y **Kali Undercover mode**.



Apúntate a nuestros canales de Telegram: SoloLinux y SoloWordpress.

SoloWordpress

Microsoft Team para linux

Con alevosía y nocturnidad, aun recordando sus tiempos oscuros contra linux… esta vez es al revés, Microsoft nos da la mano.

Ya comentamos en un articulo anterior, que **Microsoft** tiene previsto lanzar su **navegador Edge para linux**, ahora sorprende a propios y extraños ofreciendo para su descarga la primera herramienta del paquete **Office 360**. Es evidente, que los de **Bill Gates** no pueden frenar el movimiento linux y ahora quieren parte del pastel.

Para aquellos que no conozcan **Microsoft Teams**, debéis saber que hablamos de una plataforma de comunicación que incluye chat, mensajería de vídeo, almacenamiento colaborativo de archivos y otras características. Es muy similar a **Slack**, de hecho es su principal competidor.

Pero ojo, la diferencia entre los dos es evidente. **Slack** apostó por linux hace unos años, y ahora como ve que le come la tostada se lanza de cabeza a por ella.

La relación de Microsoft con Linux y el código abierto, está llena de ataques, denuncias, plagios, y otros abusos demasiado oscuros que no quiero recordar. Es evidente que yo, jamas instalare nada que provenga de ellos, me da igual que sea free, open, etc, no quiero saber nada.

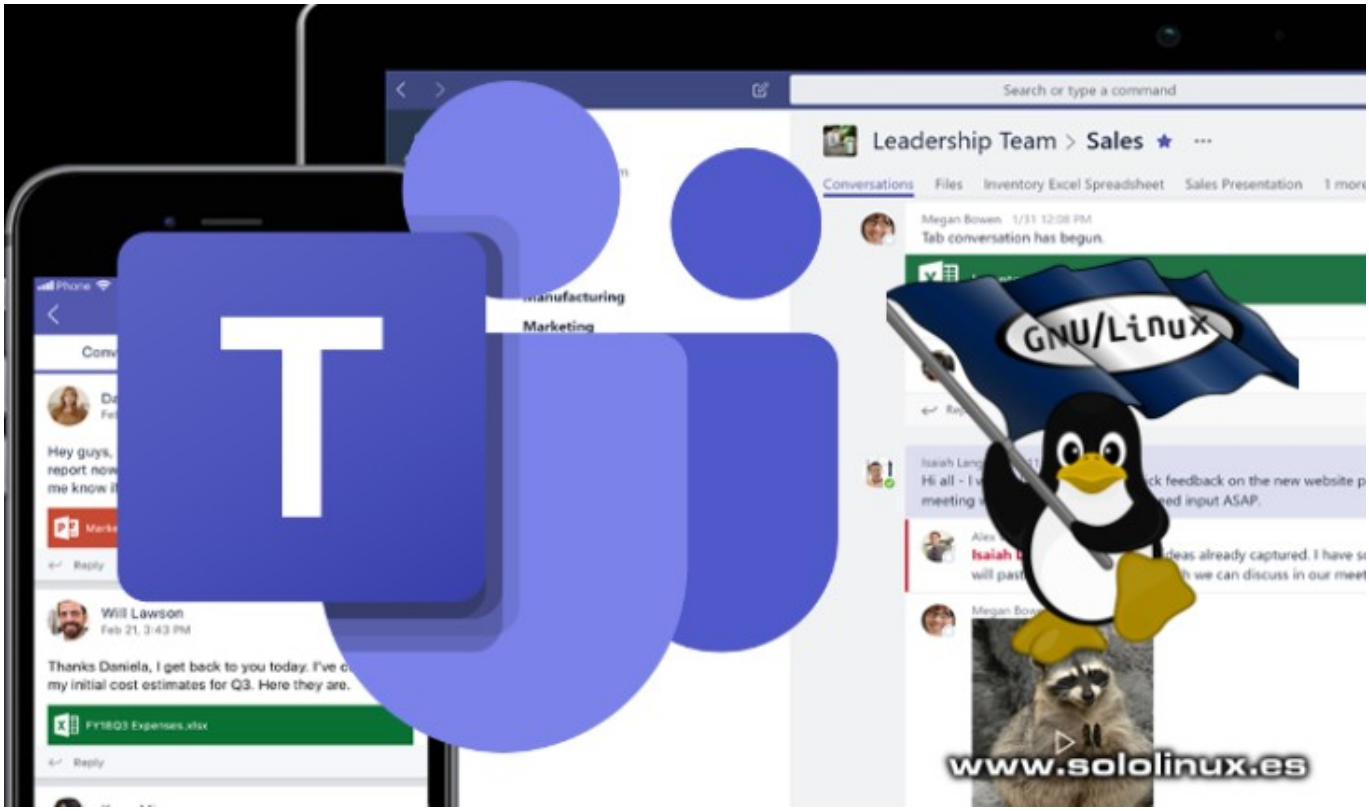
Aun así, debo reconocer que bajo la batuta de **Satya Nadella** (que parece tener dos dedos de frente), han adoptado y favorecido el código abierto (un poco tarde). Esto nos indica que Microsoft está dispuesto a poner sus herramientas donde los clientes las necesiten, independientemente de la plataforma o el sistema operativo que utilicen. Las vueltas que da la vida.

Microsoft Team para linux, sale al mercado como primera aplicación de **Office 365 en Linux**, si ellos son bien recibidos en nuestro sistema abrirán la puerta a más aplicaciones. En nuestras manos está, pero una cosa si te digo, **Microsoft llega a linux**, y **viene para quedarse**, no lo olvides.

Si quieres descargar la aplicación, accedes a su pagina oficial y seleccionas tu versión, deb o rpm.

Gracias a Miguel M. por el aviso a **través de nuestro canal de Telegram**.

Canales de Telegram: Canal SoloLinux – Canal SoloWordpress



Diferencias entre HTTP y HTTPS

A pesar de que hace años (2014) **Google recomendó que todos los sitios web mudaran a HTTPS**, muchos siguen utilizando el **protocolo HTTP**. Hasta esa fecha (2014) pocas webs utilizaban **HTTPS**, parecía un coto exclusivo de sitios con comercio electrónico (tiendas virtuales) y grandes corporaciones; de repente todo cambió.

Aun recuerdo aquellas fechas, vaya la que se armó. Programadores, Webmasters, clientes, todo el mundo preocupados porque en el comunicado de **Google** se anunciaba que los sitios con **protocolo HTTPS** se posicionarían mejor en su buscador.

La verdad es que no fue para tanto, pues **Google** fue implantado las reglas progresivamente y no dio un portazo a nadie en aquel momento, aun así, hoy en día siguen existiendo miles y miles de paginas que utilizan HTTP, tal vez por desconocimiento, o porque son sitios para un grupo de usuarios reducido.

Probablemente te preguntes: ¿Por qué es tan importante cambiar a HTTPS?, ¿realmente vale la pena?, ¿existen diferencias entre HTTP y HTTPS?, ¿afecta al SEO?. En este articulo obtendrás la respuesta a todas tus dudas.

Diferencias entre HTTP y HTTPS
Lo primero que debemos conocer son los conceptos básicos de cada protocolo, se asemejan pero no son iguales. Comenzamos con HTTP y continuaremos con HTTPS.

- **HTTP:** HTTP (protocolo de transferencia de hipertexto) permite la comunicación entre diferentes sistemas. Normalmente se utiliza para transferir datos desde un servidor web a un navegador para que los usuarios puedan visitar sus páginas web. Es el protocolo básico de cualquier sitio web o servidor.
- **HTTPS:** HTTPS (protocolo de transferencia de hipertexto seguro). El principal problema del protocolo HTTP es que la información que circula desde el servidor al navegador no está encriptada, lo que repercute negativamente en que puede ser robada sin tener grandes conocimientos técnicos. El protocolo HTTPS soluciona este problema con el uso de certificados SSL (capa de conexión segura), pues crea una conexión segura entre el servidor y el navegador protegiendo la información independientemente si es sensible o no.

Principales diferencias entre HTTP y HTTPS

La principal **diferencia entre HTTP y HTTPS** es el certificado SSL. De hecho, HTTPS es el protocolo HTTP con seguridades adicionales añadidas. OJO!!!, esta seguridad puede resultar importantísima especialmente en sitios web donde aportas datos extremadamente sensibles, por ejemplo tu tarjeta bancaria.

¿Cómo funciona HTTPS?: El **certificado SSL** se encarga de encriptar la información que los usuarios proporcionan al sitio web, básicamente convierte los datos en un código. Si alguien llegara a conseguir esos datos no podría entenderlos debido al cifrado (no hay nada imposible, pero si difícil).

La cosa no queda ahí, además de agregar la capa de seguridad, HTTPS también incorpora una seguridad adicional conocida como **TLS (Transport Layer Security)**. TLS nos aporta integridad en los datos, lo que evita que durante la transferencia de los mismos se modifiquen o corrompan, ademas de la autentificación que verifica que nos comunicamos con el sitio correcto.

Identificar si un sitio utiliza el protocolo HTTPS es tan fácil como mirar la url del sitio en tu navegador web. Antes de la propia dirección web veras si el sitio utiliza protocolos HTTP o HTTPS, en algún navegador no aparece la indicación pero aparece un candado cerrado.

Ventajas sobre SEO
La principal ventaja es la seguridad, pero no es el único beneficio de hacer uso de HTTPS. Cambiar a HTTPS también nos ayuda en nuestro trabajo SEO.

Vemos algunas formas en las que nos ayuda:

- Dejando a un lado el hecho de que Google anunciara que los sitios con HTTPS recibirán un pequeño aumento en sus clasificaciones, es evidente que los sitios con protocolos seguros se posicionan mucho mejor. Pocas paginas web con HTTP están en las 30 primeras posiciones de una búsqueda.
- Si utilizas **Google Analytics** debes saber que en los sitios HTTP las fuentes de referencia aparecen como «tráfico directo», esto ya es una gran ventaja del protocolo https.
- Debido a que un sitio HTTPS encripta todas las comunicaciones, los visitantes tendrán protección no solo en su información confidencial, sino también en su historial de navegación, esto genera una de las cosas más importantes en internet, la confianza del usuario. La confianza no es un tema baladí, es vital para que los usuarios sigan visitando un sitio.
- El protocolo HTTPS protege nuestro sitio de violaciones de seguridad, que tarde o temprano acabarían dañando la reputación del sitio.

Crear sitios web para móviles
AMP (páginas móviles aceleradas) fue creado por **Google** para cargar contenido en dispositivos móviles muy rápido. Si quieres utilizar esta tecnología, el **HTTPS** es obligatorio.

Podríamos decir que es como un HTML simplificado. El contenido de AMP aparece destacado en las **SERP de Google** para una mejor experiencia de los usuarios de teléfonos inteligentes y tabletas. Por lo dicho si quieres un sitio para móviles con AMP, el HTTPS es indispensable.

Conclusión final
Existen muchas razones para que tu sitio web sea https, y no es solo la de proteger información confidencial, sino que también debes asegurarte de que tus visitantes se sientan cómodos al navegar por el sitio. Lo dicho ya es una buena razón por sí sola, y si tienes en cuenta el efecto sobre el SEO los motivos son obvios.

Si eres de los que aun utiliza HTTP, cambia, pero cambia ya. Hoy en día no requiere un gran esfuerzo y la recompensa es grande.

Supongo que con este articulo ya no tengas dudas respecto a las **diferencias entre HTTP vs HTTPS**, sus beneficios y mejoras.

Cómo enviar correos electrónicos desde la terminal linux

Tal vez pienses que es una tontería innecesaria, que ya tienes tu **cliente de correo electrónico** para eso. Pues yo te digo que te equivocas, te puede sacar de más de un apuro aprender los comandos que hoy veremos.

Existen muchas herramientas con sus comandos capaces de enviar un **mail** desde la terminal, nosotros tratamos solo los más conocidos.

- **mailx / mail**
- **mutt**
- **mpack**
- **sendmail**
- **ssmtp**

Enviar correos electrónicos desde la terminal linux

Todos los comandos tienen la misma función, redactar correos electrónicos y enviarlos a un agente de transferencia de correo local (MTA), sendmail, postfix, etc. A continuación vemos cómo instalar las herramientas y enviar correos electrónicos, ya veras que fácil (es posible que algunas ya estén instaladas por defecto en tu sistema).

Mailx / Mail

Mailx es la versión mejorada del comando mail. Está basado en Berkeley Mail 8.1, y proporciona la funcionalidad del comando POSIX mailx, ademas ofrece extensiones para MIME, IMAP, POP3, SMTP y S / MIME.

Instalamos mailx.

```
# Debian, Ubuntu, Linux Mint, etc...
sudo apt-get install mailutils

# CentOS, RHEL, etc...
sudo yum install mailx

# Fedora, RHEL8, CentOS8, etc...
sudo dnf install mailx
```

Vemos como enviar un correo sin archivo adjunto, y con el.

```
# Normal
echo "mensaje" | mail -s "asunto" mimail@mail.com

# Con archivo adjunto
echo "mensaje" | mail -a adjunto.txt -s "asunto" mimail@mail.com
```

Mutt

Mutt es una herramienta pequeña en tamaño pero poderosa. Incluye soporte para MIME, OpenPGP, y un modo de clasificación por hilos.

Instalamos la herramienta.

```
# Debian, Ubuntu, Linux Mint, etc...
sudo apt-get install mutt

# CentOS, RHEL, etc...
sudo yum install mutt

# Fedora, RHEL8, CentOS8, etc...
sudo dnf install mutt
```

Vemos como enviar un correo sin archivo adjunto, y con el.

```
# Normal
echo "mensaje" | mutt -s "asunto" mimail@mail.com

# Con archivo adjunto
echo "mensaje" | mutt -s "asunto" mimail@mail.com -a adjunto.txt
```

Mpack

El comando mpack es un tanto particular, ya que codifica el archivo en MIME. El resultado se envía a su destino.

Instalar Mpack.

```
# Debian, Ubuntu, Linux Mint, etc...
sudo apt-get install mpack

# CentOS, RHEL, etc...
sudo yum install mpack

# Fedora, RHEL8, CentOS8, etc...
sudo dnf install mpack
```

Ahora aprendemos a enviar un correo sin archivo adjunto, y con el.

```
# Normal
echo "mensaje" | mpack -s "asunto" mimail@mail.com

# Con archivo adjunto
echo "mensaje" | mpack -s "asunto" mimail@mail.com -a adjunto.txt
```

Sendmail

Sendmail es uno de los servidores SMTP más conocidos, pero tal vez no sabias que también permite enviar correos electrónicos desde linea de comandos.

Vemos como instalar Sendmail.

```
# Debian, Ubuntu, Linux Mint, etc...
sudo apt-get install sendmail

# CentOS, RHEL, etc...
sudo yum install sendmail

# Fedora, RHEL8, CentOS8, etc...
sudo dnf install sendmail
```

Enviar correos con sendmail es un poco diferente al resto de opciones, primero se crea el archivo del correo y después se envía.

```
# Crear archivo
echo -e "Subject: asuntol\nMensaje" > /tmp/send-mail.txt

# Enviar
sendmail mimail@mail.com < send-mail.txt
```

Ssmtp
Realmente, ssmtp es un emulador de sendmail que nos permite enviar correos electrónicos a través de un servidor SMTP desde la línea de comandos de Linux. La forma de operar es la misma que sendmail.

Instalar el emulador ssmtp.

```
# Debian, Ubuntu, Linux Mint, etc...
sudo apt-get install ssmtp

# CentOS, RHEL, etc...
sudo yum install ssmtp

# Fedora, RHEL8, CentOS8, etc...
sudo dnf install ssmtp
```

Al igual que sendmail, también debemos crear un archivo y después enviarlo.

```
# Crear archivo
echo -e "Subject: asuntol\nMensaje" > /tmp/send-mail.txt

# Enviar
ssmtp mimail@mail.com < send-mail.tx
```

Instalar Firefox (no ESR) en Debian 10 Buster

Firefox ESR (extended Support Release), es la versión empresarial de Firefox. Sus propiedades se centran en la seguridad y estabilidad. Durante su ciclo de vida se van agregando diversas funciones y correcciones en temas de seguridad y estabilidad, pero no las nuevas funciones de la ultima versión estable del Firefox que todos conocemos.

Si tu sistema es **Debian 10 Buster** seguro que te diste cuenta, el aspecto de **Firefox** parece un poco obsoleto, jaja, eso es porque la interfaz gráfica de **Firefox ESR** apenas cambia entre versión y versión.

El motivo de que tengas instalada la versión **ESR**, es porque **Debian Stable** y **Debian Testing** solo tienen **Firefox-ESR** en sus repositorios. En este artículo vemos cómo instalar la ultima versión de Firefox normal (estable pero no ESR) en Debian Stable (Buster) o Testing (Bullseye), ademas, si eres un usuario normal es recomendable, piensa que con la ESR puedes tener problemas con archivos multimedia y alguna cosa más.

Instalar Firefox (no ESR) en Debian 10 Buster
Como ya comentamos anteriormente, Debian 10 Buster (Estable) y Debian Testing (Bullseye) solo tienen Firefox ESR en sus repositorios, pero ojo al detalle… Debian Unstable si que tiene la última versión normal de Firefox (no ESR), ademas de la ESR.

Ja, entonces lo tenemos claro, vamos a usar el repositorio Debian «unstable» (Sid), para instalar la **ultima versión de Firefox** de forma muy sencilla. **Ten mucho cuidado y sigue los pasos tal como te los indico**, si no es así, puedes convertir tu Debian 10 Buster en un Debian Unstable.

Agregaremos manualmente el repositorio Debian Unstable en Debian Stable, después le aplicaremos una prioridad baja, para que por si solo no pueda instalar paquetes automáticamente a menos que nosotros se lo indiquemos.

Bien, vamos a por ello. Lo primero que debes hacer es abrir el archivo de los repositorios y agregar el unstable.
```
sudo nano /etc/apt/sources.list
```

Al final del archivo, copia y pega lo siguiente…
```
deb http://deb.debian.org/debian/ unstable main contrib
non-free
```

Guarda el archivo y cierra el editor.

ATENCIÓN: NO ACTUALICES EL SISTEMA BAJO NINGÚN CONCEPTO.

Bien… ahora establecemos una prioridad baja al repositorio unstable de Debian, para lograr nuestro objetivo debes crear un nuevo archivo como te indico a continuación.
```
sudo nano /etc/apt/preferences.d/99pin-unstable
```

Copia y pega lo siguiente.
```
Package: *
Pin: release a=stable
Pin-Priority: 900

Package: *
Pin release a=unstable
Pin-Priority: 10
```

Guarda el archivo y cierra el editor.

El proceso de instalación te extrañara, no es común. Por las reglas que hemos aplicado no funciona «apt install firefox», pero si lo haces como te indico no tendrás ningún problema, vamos.
```
sudo apt update
sudo apt install -t unstable firefox
```

Como puedes ver le decimos a Debian que instale firefox desde el repositorio indicado, y se olvide de los demás.

Bueno, ya lo tenemos instalado. Ahora te recomiendo que lo pruebes bien, recuerda que también tienes instalado Firefox ESR y es posible que tengas algún problema o notes cierta inestabilidad, si es tu caso borras el viejo Firefox y arreglado.
```
sudo apt purge firefox-esr
```

Desinstalar Firefox en Debian 10
Si por cualquier motivo, no te convence y quieres degradar de nuevo el sistema lo puedes hacer. Editamos el archivo creado anteriormente.
sudo nano /etc/apt/preferences.d/99pin-unstable

Borra el contenido y pega lo siguiente.
Package: *
Pin: release a=stable
Pin-Priority: 1001

Package: *
Pin release a=unstable
Pin-Priority: -1

Guarda el archivo y cierra el editor.

Con los siguientes comandos se eliminara todo lo que se instalo desde el repositorio unstable, y volverá a su estado original
sudo apt update
sudo apt full-upgrade

rm /etc/apt/preferences.d/99pin-unstable

Abres el archivo de los repositorios y eliminas o comentas el repositorio unstable.
sudo nano /etc/apt/sources.list

.....
.....
deb http://deb.debian.org/debian/ unstable main contrib non-free

Damos por concluido el articulo **instalar Firefox en Debian 10.**

Mutt, cliente de email en terminal

Mutt es cliente de **email en terminal** para sistemas basados en **Unix**, que nos permite enviar y recibir correos en linea de **comandos**.

Destacamos que **nos permite acceder** por completo a la bandeja de entrada de nuestra cuenta de e-mail preferida, así como el no tener que preocuparnos por el **MTA (agente de transporte mail)**, Mutt instala todo lo necesario para que disfrutemos de un completo cliente de correo en nuestra terminal.

En este articulo vemos como instalar y configurar Mutt en nuestro linux preferido.

Mutt, cliente de email en terminal
Vamos a instalar Mutt.

Debian, Ubuntu, Linux Mint, y derivados:
sudo apt install mutt

opcional
sudo apt install offlineimap msmtp
sudo apt install getmail procmail

CentOS, RHEL, y derivados:
sudo yum install mutt

opcional
sudo yum install offlineimap msmtp
sudo yum install getmail procmail

Fedora, CentOS8, RHEL8, y derivados:
sudo dnf install mutt

opcional
sudo dnf install offlineimap msmtp
sudo dnf install getmail procmail

Arch Linux, Manjaro, y derivados:
sudo pacman -S mutt

opcional
sudo pacman -S offlineimap msmtp
sudo pacman -S getmail procmail

Recordemos que no nos debemos preocupar por nada, Mutt instala todo lo necesario como vemos en el ejemplo.

```
sololinux # apt-get install mutt
Leyendo lista de paquetes… Hecho
Creando árbol de dependencias
Leyendo la información de estado… Hecho
Se instalarán los siguientes paquetes adicionales:
libtokyocabinet9
Paquetes sugeridos:
urlview mixmaster
Paquetes recomendados:
default-mta | mail-transport-agent
Se instalarán los siguientes paquetes NUEVOS:
libtokyocabinet9 mutt
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1.219 kB de archivos.
Se utilizarán 4.884 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

Configurar Mutt
El cliente viene sin archivo de configuración de usuario, lo creamos con los datos reales de nuestra cuenta de correo, en el ejemplo usamos Gmail.
nano ~/.muttrc

Copia y pega insertando tus datos reales.

```
set from = "usuario@gmail.com"
set realname = "nombre de la cuenta gmail"
set imap_user = "usuario@gmail.com"
set imap_pass = "password"
set folder = "imaps://imap.gmail.com:993"
set spoolfile = "+INBOX"
set postponed ="+[Gmail]/Drafts"
set header_cache =~/.mutt/cache/headers
set message_cachedir =~/.mutt/cache/bodies
set certificate_file =~/.mutt/certificates
set smtp_url = "smtp://usuario@smtp.gmail.com:587/"
set smtp_pass = "password"
set move = no
set imap_keepalive = 900
```

Guarda el archivo y cierra el editor.

Antes de poder utilizar el cliente necesitas crear una carpeta cache, la creamos.
mkdir -p ~/.mutt/cache

Listo, ya podemos lanzar nuestro cliente de email en terminal.
mutt

Se abre la aplicación Mutt que como puedes ver en la siguiente imagen, es sencilla de usar.

Pulsando la tecla ? veras los comandos de ejecución.

Ya está aquí VirtualBox 6.1 con importantes mejoras

Se ha lanzado una nueva versión principal de **VirtualBox**, y viene con importantes mejoras como los controladores gráficos **VBoxSVGA** y **VMSVGA**, soporte experimental para transferencias de archivos desde el portapapeles compartido, y como no podía ser de otra manera, también soporte para **Linux 5.4**.

VirtualBox es un software de virtualización x86 y AMD64 / Intel64 que se ejecuta en Windows, Linux, macOS y Solaris.

VirtualBox 6.1
El nuevo **VirtualBox 6.1** ofrece la posibilidad de importar una máquina virtual desde **Oracle Cloud Infrastructure**, también permite exportar una máquina virtual de **VirtualBox 6.1 a Oracle Cloud Infrastructurel**. Es una opción interesante, pues nos permite crear varias máquinas virtuales sin tener que volver a cargarlas.

Otra mejora importante es el soporte mejorado de la virtualización de hardware anidado (antes solo se permitía sobre **CPU AMD**), por tanto ahora podemos instalar un hipervisor como VirtualBox o **KVM** en un invitado VirtualBox, y que pueda crear y ejecutar sus propias máquinas virtuales en la máquina virtual invitada. También se agrega soporte para la quinta generación de las CPU de Intel (Broadwell).

El **soporte 3D** (VBoxSVGA y VMSVGA) se ha mejorado muchísimo. VBoxSVGA y VMSVGA admiten YUV2 y otros formatos de textura relacionados con sistemas que usan **OpenGL** (macOS y Linux), esto acelera y mejora la reproducción de vídeo cuando el 3D está habilitado.

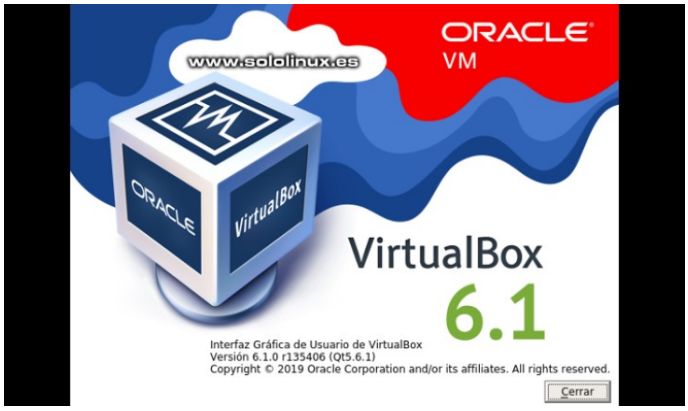
Otra mejora interesante es que ahora se soporta la ultima versión estable del kernel linux, la 5.4.

Otras modificaciones de VirtualBox 6.1

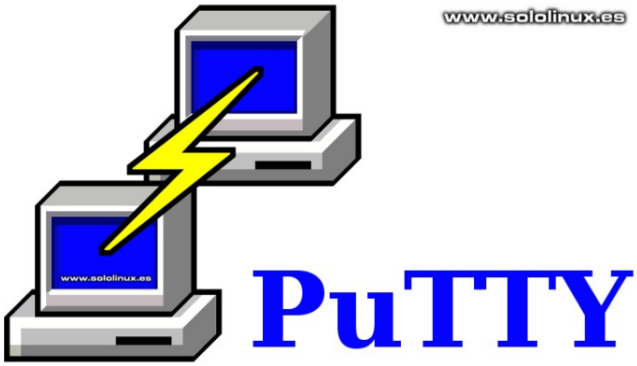
- **GUI**:
  - El panel de detalles de la máquina virtual se extiende con editores integrados para los atributos de la VM seleccionada.
  - Mejoras en el puntero del ratón.
  - Crear grupos de maquinas ahora es mas fácil.
  - El usuario puede cambiar el tipo de bus del controlador de almacenamiento y mover archivos adjuntos con arrastrar y soltar
  - Los marcos de diálogo del administrador de archivos se han mejorado.
  - Ahora tenemos un nuevo teclado virtual que permite entradas de usuarios invitados (se incluyen teclas multimedia).
  - Indicador de la carga de CPU del VM.
- **EFI**: firmware actualizado, soporte para NVRAM, para sistema de archivos APFS, y dispositivos SATA / NVMe no estándar.
- **Núcleo de virtualización**: Drop recompiler (VM ahora necesita una CPU que soporte la virtualización por hardware).
- **Tiempo de ejecución**: trabaja en hosts con multiples CPU (el límite es 1024).
- **vboximg-mount**: Soporte experimental para la lectura de archivos (en sistemas NTFS, FAT y ext2/3/4), que estén en una imagen de disco.

Descargar VirtualBox 6.1
Puedes descargas la ultima versión de VirtualBox desde su pagina oficial.

- Pagina oficial de descargas VirtualBox



Instalar PuTTY en linux

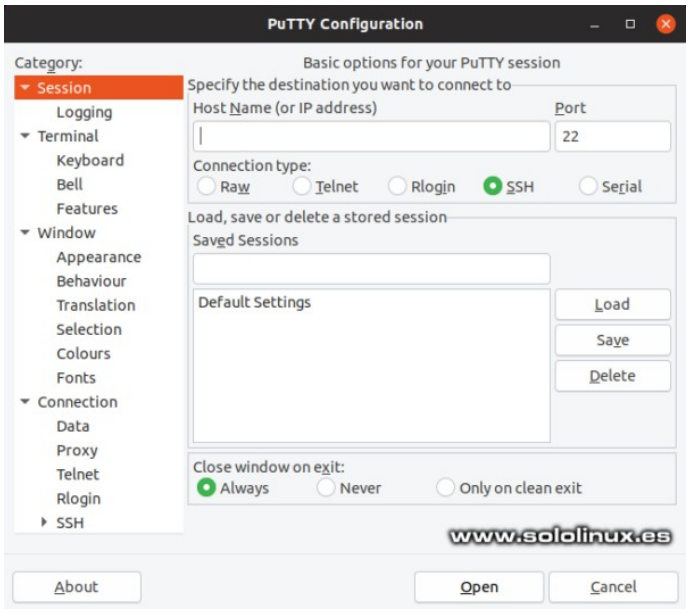


PuTTY es un emulador de terminal para **Windows** creado por **Simon Tatham** en el año 2000. Su diseño primario era desarrollar un emulador que permitiera la transferencia de archivos a través de la red.

El propio **Simon Tathan** no podía creer el rotundo éxito de su herramienta entre la comunidad. Todo comenzó por la comunicación por puerto serie, pero al agregar los protocolos de red **SCP, SSH, Telnet**, y más, se abrió un mundo de posibilidades. Ahora, los usuarios de Windows podían conectar con terminales **Unix** de forma muy sencilla.

Hoy en día, los usuarios que llevamos muchos años en el mundillo estamos acostumbrados a trabajar en cualquier emulador de terminal, pero los recién llegados a linux, quien más, quien menos a tocado alguna vez **PuTTY**, es normal que al estar familiarizados con el se sientan más cómodos.

Lamentablemente **PuTTY** no ofrece versión para linux, pero si su código fuente. En este articulo veremos como **instalar PuTTY** en cualquier **distribución linux**.



Instalar PuTTY en linux

Muchas distribuciones linux han incluido PuTTY en sus repositorios oficiales, pero no todas. Vemos como instalar el emulador en cualquier linux.

En Debian, Ubuntu, Linux Mint, y derivados:

```
sudo apt install putty
```

Si no encuentra el paquete, añadimos el repositorio Universe y continuamos con la instalación.

```
sudo add-apt-repository universe

sudo apt update

sudo apt install putty
```

En CentOS, RHEL, y derivados:

```
sudo yum install putty
```

En Fedora, CentOS 8, RHEL 8, y derivados:

```
sudo dnf install putty
```

En Arch Linux, Manjaro, y derivados:

```
sudo pacman -S putty
```

Otras distribuciones linux:

También puedes instalar la aplicación desde el código fuente.

Al publicar este articulo la ultima versión disponible es **PuTTY 0.73**, antes de seguir el ejemplo que te propongo asegúrate en esta pagina que no existe otra más nueva. Descargamos e instalamos PuTTY.

```
wget https://the.earth.li/~sgtatham/putty/latest/putty-0.73.tar.gz

tar -xvf putty-0.73.tar.gz

cd putty-0.73

./configure

sudo make

sudo make install
```

Una vez termine la instalación correctamente, ya puedes empezar a utilizar PuTTY.

Google bloquea el servicio en algunos navegadores web

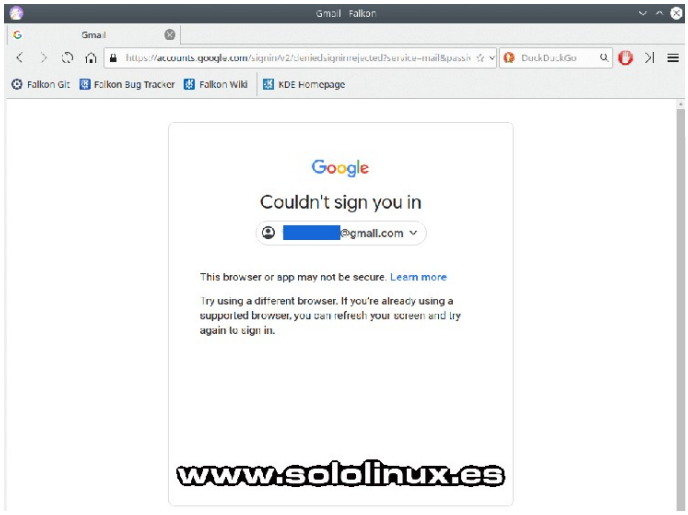
Parece mentira, pero navegadores tan conocidos en Linux como **Konqueror**, **Falkon** y **Qutebrowser**, están siendo bloqueados al intentar iniciar sesión en los servicios de sus cuentas de **Google** o **Gmail**.

No está muy claro a que es debido, ni cuándo el gigante Google comenzó a bloquear estos navegadores, todo salio a la luz cuando un usuario de **Reddit** escribió un aviso indicando el problema.

Blogs informativos tan conocidos como **BleepingComputer** han confirmado el problema, incluso en **sololinux** hemos realizado las pruebas pertinentes, y si, efectivamente bloquea el inicio de sesión.

Existen una serie de teorías sobre por qué estos navegadores no pueden iniciar sesión en los servicios de Google bajo linux. Unos dicen que son pruebas de Google, otros que es por temas de cookies, javascript y varios. Incluso alguno acusa a Google de estar quitando los baches del camino para la próxima incorporación al mundo linux del **navegador de Microsoft Edge**.

Imagen con el problema al iniciar sesión.



Google bloquea el servicio de inicio de sesión
Como dije antes, en SoloLinux.es también hemos realizado pruebas que seguro llamaran la atención. Nosotros intentamos iniciar sesión en los tres navegadores, y ademas en tres idiomas diferentes, Ingles, Ruso, y Español.

El Resultado es sorprendente, mira la tabla.

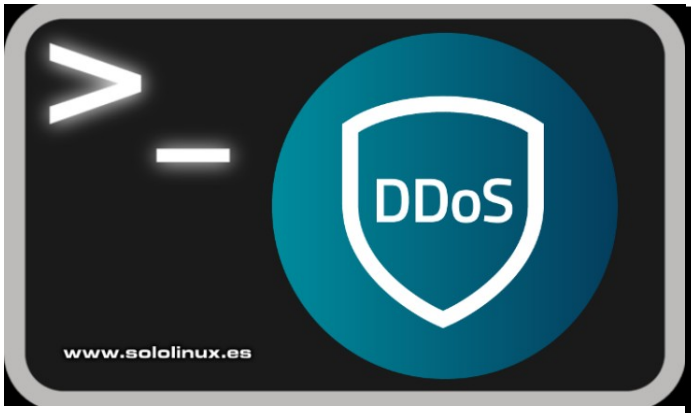
	INGLES	RUSO	ESPAÑOL
Konqueror	OK	FAIL	OK
Falkon	FAIL	FAIL	FAIL
QuteBrowser	OK	FAIL	FAIL

En la pantalla de error al iniciar sesión, aparece un mensaje que nos envía a un artículo del soporte de Google, en el articulo nos indica las posibles razones que causan el bloqueo de los servicios de Google en los navegadores Linux:

- No es compatible con JavaScript o lo tienes desactivado.
- Existen extensiones no seguras o no compatibles instaladas.
- Usas marcos de pruebas de automatizados.
- Navegador incrustado en otra aplicación.

A titulo personal lo que me mosquea es lo del idioma, tal vez están probando algún identificador que refiera tu idioma y localización como verificador, no lo se. De todas formas pese a las muchas quejas **Google** aún no ha publicado una respuesta oficial sobre que está pasando con estos navegadores en linux. Seguiremos esperando.

Anti DDos – Bash Script



Uno de los mayores quebraderos de cabeza de los **sysadmin**, son los ataques DDOS. Toda precaución es poca, y no hay nada seguro. Es prácticamente imposible detener un DDos a gran escala, aun así, como mínimo debes intentar protegerte de ataques menores.

Hoy presentamos un excelente script **Anti-DDOS** creado por Ismail Tasleden. Escrito en **bash** y de código abierto, este script que hoy vemos es diferente a otros que ya tratamos en **SoloLinux**; como norma general estos ejecutables rastrean los registros buscando errores para bloquear las IP que los provocan.

Este proyecto es diferente, lo que hace es habilitar las reglas preventivas que nos aporta el kernel linux y que suelen venir con normas excesivamente permisivas. También aplica reglas **anti DDos** en las **iptables o nftables**, y otras configuraciones necesarias como medidas alternativas de defensa.

Este **script bash** es 100% compatible con todos los sistemas Linux. Recordemos que frente a un **DDos** no hay nada infalible, pero por lo menos tomamos medidas preventivas ante **lamers**.

Anti DDos – Bash Script
Creamos el script.
nano anti-ddos.sh

Copia y pega lo siguiente.

```
#!/bin/sh

##############################################################
#               ANTI-DDOS BASH SCRIPT                        #
##############################################################

# For debugging use iptables -v.
IPTABLES="/sbin/iptables"
IP6TABLES="/sbin/ip6tables"
MODPROBE="/sbin/modprobe"
RMMOD="/sbin/rmmod"
ARP="/usr/sbin/arp"

# Logging options.
#--------------------------------------------------------------------------
LOG="LOG --log-level debug --log-tcp-sequence --log-tcp-options"
LOG="$LOG --log-ip-options"

# Defaults for rate limiting
#--------------------------------------------------------------------------
RLIMIT="-m limit --limit 3/s --limit-burst 8"

# Unprivileged ports.
#--------------------------------------------------------------------------
PHIGH="1024:65535"
PSSH="1000:1023"

# Load required kernel modules
#--------------------------------------------------------------------------
$MODPROBE ip_conntrack_ftp
$MODPROBE ip_conntrack_irc

# Mitigate ARP spoofing/poisoning and similar attacks.
#--------------------------------------------------------------------------
# Hardcode static ARP cache entries here
# $ARP -s IP-ADDRESS MAC-ADDRESS

# Kernel configuration.
#--------------------------------------------------------------------------

# Disable IP forwarding.
# On => Off = (reset)
echo 1 > /proc/sys/net/ipv4/ip_forward
echo 0 > /proc/sys/net/ipv4/ip_forward

# Enable IP spoofing protection
for i in /proc/sys/net/ipv4/conf/*/rp_filter; do echo 1 > $i; done

# Protect against SYN flood attacks
echo 1 > /proc/sys/net/ipv4/tcp_syncookies

# Ignore all incoming ICMP echo requests
echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_all

# Ignore ICMP echo requests to broadcast
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

# Log packets with impossible addresses.
for i in /proc/sys/net/ipv4/conf/*/log_martians; do echo 1 > $i; done

# Don't log invalid responses to broadcast
echo 1 >
/proc/sys/net/ipv4/icmp_ignore_bogus_error_responses

# Don't accept or send ICMP redirects.
for i in /proc/sys/net/ipv4/conf/*/accept_redirects; do echo 0 > $i; done
for i in /proc/sys/net/ipv4/conf/*/send_redirects; do echo 0 > $i; done

# Don't accept source routed packets.
for i in /proc/sys/net/ipv4/conf/*/accept_source_route; do echo 0 > $i; done

# Disable multicast routing
for i in /proc/sys/net/ipv4/conf/*/mc_forwarding; do echo 0 > $i; done
```

```
# Disable proxy_arp.
for i in /proc/sys/net/ipv4/conf/*/proxy_arp; do echo 0 > $i;
done

# Enable secure redirects, i.e. only accept ICMP redirects
for gateways
# Helps against MITM attacks.
for i in /proc/sys/net/ipv4/conf/*/secure_redirects; do echo 1
> $i; done

# Disable bootp_relay
for i in /proc/sys/net/ipv4/conf/*/bootp_relay; do echo 0 > $i;
done

# Default policies.
#------------------------------------------------------------------------

# Drop everything by default.
$IPTABLES -P INPUT DROP
$IPTABLES -P FORWARD DROP
$IPTABLES -P OUTPUT DROP

# Set the nat/mangle/raw tables' chains to ACCEPT
$IPTABLES -t nat -P PREROUTING ACCEPT
$IPTABLES -t nat -P OUTPUT ACCEPT
$IPTABLES -t nat -P POSTROUTING ACCEPT

$IPTABLES -t mangle -P PREROUTING ACCEPT
$IPTABLES -t mangle -P INPUT ACCEPT
$IPTABLES -t mangle -P FORWARD ACCEPT
$IPTABLES -t mangle -P OUTPUT ACCEPT
$IPTABLES -t mangle -P POSTROUTING ACCEPT

# Cleanup.
#------------------------------------------------------------------------

# Delete all
$IPTABLES -F
$IPTABLES -t nat -F
$IPTABLES -t mangle -F

# Delete all
$IPTABLES -X
$IPTABLES -t nat -X
$IPTABLES -t mangle -X

# Zero all packets and counters.
$IPTABLES -Z
$IPTABLES -t nat -Z
$IPTABLES -t mangle -Z

# Completely disable IPv6.
#------------------------------------------------------------------------

# Block all IPv6 traffic
# If the ip6tables command is available, try to block all IPv6
traffic.
if test -x $IP6TABLES; then
# Set the default policies
# drop everything
$IP6TABLES -P INPUT DROP 2>/dev/null
$IP6TABLES -P FORWARD DROP 2>/dev/null
$IP6TABLES -P OUTPUT DROP 2>/dev/null

# The mangle table can pass everything
$IP6TABLES -t mangle -P PREROUTING ACCEPT 2>/dev/
null
$IP6TABLES -t mangle -P INPUT ACCEPT 2>/dev/null
$IP6TABLES -t mangle -P FORWARD ACCEPT 2>/dev/null
$IP6TABLES -t mangle -P OUTPUT ACCEPT 2>/dev/null
$IP6TABLES -t mangle -P POSTROUTING ACCEPT
2>/dev/null

# Delete all rules.
$IP6TABLES -F 2>/dev/null
$IP6TABLES -t mangle -F 2>/dev/null

# Delete all chains.
$IP6TABLES -X 2>/dev/null
$IP6TABLES -t mangle -X 2>/dev/null

# Zero all packets and counters.
$IP6TABLES -Z 2>/dev/null
$IP6TABLES -t mangle -Z 2>/dev/null
Fi

# Custom user-defined chains.
#------------------------------------------------------------------------

# LOG packets, then ACCEPT.
$IPTABLES -N ACCEPTLOG
$IPTABLES -A ACCEPTLOG -j $LOG $RLIMIT --log-prefix
"ACCEPT "
$IPTABLES -A ACCEPTLOG -j ACCEPT

# LOG packets, then DROP.
$IPTABLES -N DROPLOG
$IPTABLES -A DROPLOG -j $LOG $RLIMIT --log-prefix
"DROP "
$IPTABLES -A DROPLOG -j DROP

# LOG packets, then REJECT.
# TCP packets are rejected with a TCP reset.
$IPTABLES -N REJECTLOG
$IPTABLES -A REJECTLOG -j $LOG $RLIMIT --log-prefix
"REJECT "
$IPTABLES -A REJECTLOG -p tcp -j REJECT --reject-with
tcp-reset
$IPTABLES -A REJECTLOG -j REJECT

# Only allows RELATED ICMP types
# (destination-unreachable, time-exceeded, and parameter-
problem).
# TODO: Rate-limit this traffic?
# TODO: Allow fragmentation-needed?
# TODO: Test.
$IPTABLES -N RELATED_ICMP
$IPTABLES -A RELATED_ICMP -p icmp --icmp-type
destination-unreachable -j ACCEPT
$IPTABLES -A RELATED_ICMP -p icmp --icmp-type time-
exceeded -j ACCEPT
$IPTABLES -A RELATED_ICMP -p icmp --icmp-type
parameter-problem -j ACCEPT
$IPTABLES -A RELATED_ICMP -j DROPLOG

# Make It Even Harder To Multi-PING
$IPTABLES  -A INPUT -p icmp -m limit --limit 1/s --limit-
burst 2 -j ACCEPT
$IPTABLES  -A INPUT -p icmp -m limit --limit 1/s --limit-
burst 2 -j LOG --log-prefix PING-DROP:
$IPTABLES  -A INPUT -p icmp -j DROP
$IPTABLES  -A OUTPUT -p icmp -j ACCEPT

# Only allow the minimally required/recommended parts of
ICMP. Block the rest.
#------------------------------------------------------------------------
```

```
# TODO: This section needs a lot of testing!

# First, drop all fragmented ICMP packets (almost always malicious).
$IPTABLES -A INPUT -p icmp --fragment -j DROPLOG
$IPTABLES -A OUTPUT -p icmp --fragment -j DROPLOG
$IPTABLES -A FORWARD -p icmp --fragment -j DROPLOG

# Allow all ESTABLISHED ICMP traffic.
$IPTABLES -A INPUT -p icmp -m state --state ESTABLISHED -j ACCEPT $RLIMIT
$IPTABLES -A OUTPUT -p icmp -m state --state ESTABLISHED -j ACCEPT $RLIMIT

# Allow some parts of the RELATED ICMP traffic, block the rest.
$IPTABLES -A INPUT -p icmp -m state --state RELATED -j RELATED_ICMP $RLIMIT
$IPTABLES -A OUTPUT -p icmp -m state --state RELATED -j RELATED_ICMP $RLIMIT

# Allow incoming ICMP echo requests (ping), but only rate-limited.
$IPTABLES -A INPUT -p icmp --icmp-type echo-request -j ACCEPT $RLIMIT

# Allow outgoing ICMP echo requests (ping), but only rate-limited.
$IPTABLES -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT $RLIMIT

# Drop any other ICMP traffic.
$IPTABLES -A INPUT -p icmp -j DROPLOG
$IPTABLES -A OUTPUT -p icmp -j DROPLOG
$IPTABLES -A FORWARD -p icmp -j DROPLOG

# Selectively allow certain special types of traffic.

#-----------------------------------------------------------------------

# Allow loopback interface to do anything.
$IPTABLES -A INPUT -i lo -j ACCEPT
$IPTABLES -A OUTPUT -o lo -j ACCEPT

# Allow incoming connections related to existing allowed connections.
$IPTABLES -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Allow outgoing connections EXCEPT invalid
$IPTABLES -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT

# Miscellaneous.
#-----------------------------------------------------------------------

# We don't care about Milkosoft, Drop SMB/CIFS/etc..
$IPTABLES -A INPUT -p tcp -m multiport --dports 135,137,138,139,445,1433,1434 -j DROP
$IPTABLES -A INPUT -p udp -m multiport --dports 135,137,138,139,445,1433,1434 -j DROP

# Explicitly drop invalid incoming traffic
$IPTABLES -A INPUT -m state --state INVALID -j DROP

# Drop invalid outgoing traffic, too.
$IPTABLES -A OUTPUT -m state --state INVALID -j DROP

# If we would use NAT, INVALID packets would pass - BLOCK them anyways
$IPTABLES -A FORWARD -m state --state INVALID -j DROP

# PORT Scanners (stealth also)
$IPTABLES -A INPUT -m state --state NEW -p tcp --tcp-flags ALL ALL -j DROP
$IPTABLES -A INPUT -m state --state NEW -p tcp --tcp-flags ALL NONE -j DROP

# TODO: Some more anti-spoofing rules? For example:
# $IPTABLES -A INPUT -p tcp --tcp-flags ALL FIN,URG,PSH -j DROP
# $IPTABLES -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
# $IPTABLES -A INPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP
$IPTABLES -N SYN_FLOOD
$IPTABLES -A INPUT -p tcp --syn -j SYN_FLOOD
$IPTABLES -A SYN_FLOOD -m limit --limit 2/s --limit-burst 6 -j RETURN
$IPTABLES -A SYN_FLOOD -j DROP

# TODO: Block known-bad IPs (see http://www.dshield.org/top10.php).
# $IPTABLES -A INPUT -s INSERT-BAD-IP-HERE -j DROPLOG

# Drop any traffic from IANA-reserved IPs.
#-----------------------------------------------------------------------

$IPTABLES -A INPUT -s 0.0.0.0/7 -j DROP
$IPTABLES -A INPUT -s 2.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 5.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 7.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 10.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 23.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 27.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 31.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 36.0.0.0/7 -j DROP
$IPTABLES -A INPUT -s 39.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 42.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 49.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 50.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 77.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 78.0.0.0/7 -j DROP
$IPTABLES -A INPUT -s 92.0.0.0/6 -j DROP
$IPTABLES -A INPUT -s 96.0.0.0/4 -j DROP
$IPTABLES -A INPUT -s 112.0.0.0/5 -j DROP
$IPTABLES -A INPUT -s 120.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 169.254.0.0/16 -j DROP
$IPTABLES -A INPUT -s 172.16.0.0/12 -j DROP
$IPTABLES -A INPUT -s 173.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 174.0.0.0/7 -j DROP
$IPTABLES -A INPUT -s 176.0.0.0/5 -j DROP
$IPTABLES -A INPUT -s 184.0.0.0/6 -j DROP
$IPTABLES -A INPUT -s 192.0.2.0/24 -j DROP
$IPTABLES -A INPUT -s 197.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 198.18.0.0/15 -j DROP
$IPTABLES -A INPUT -s 223.0.0.0/8 -j DROP
$IPTABLES -A INPUT -s 224.0.0.0/3 -j DROP

# Selectively allow certain outbound connections, block the rest.
#-----------------------------------------------------------------------
```

```
# Allow outgoing DNS requests. Few things will work without this.
$IPTABLES -A OUTPUT -m state --state NEW -p udp --dport 53 -j ACCEPT
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 53 -j ACCEPT

# Allow outgoing HTTP requests. Unencrypted, use with care.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 80 -j ACCEPT

# Allow outgoing HTTPS requests.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 443 -j ACCEPT

# Allow outgoing SMTPS requests. Do NOT allow unencrypted SMTP!
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 465 -j ACCEPT

# Allow outgoing "submission" (RFC 2476) requests.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 587 -j ACCEPT

# Allow outgoing POP3S requests.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 995 -j ACCEPT

# Allow outgoing SSH requests.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 22 -j ACCEPT

# Allow outgoing FTP requests. Unencrypted, use with care.
$IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 21 -j ACCEPT

# Allow outgoing NNTP requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 119 -j ACCEPT

# Allow outgoing NTP requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p udp --dport 123 -j ACCEPT

# Allow outgoing IRC requests. Unencrypted, use with care.
# Note: This usually needs the ip_conntrack_irc kernel module.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 6667 -j ACCEPT

# Allow outgoing requests to various proxies. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 8080 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 8090 -j ACCEPT

# Allow outgoing DHCP requests. Unencrypted, use with care.
# TODO: This is completely untested, I have no idea whether it works!
# TODO: I think this can be tightened a bit more.
$IPTABLES -A OUTPUT -m state --state NEW -p udp --sport 67:68 --dport 67:68 -j ACCEPT

# Allow outgoing CVS requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 2401 -j ACCEPT

# Allow outgoing MySQL requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 3306 -j ACCEPT

# Allow outgoing SVN requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 3690 -j ACCEPT

# Allow outgoing PLESK requests. Unencrypted, use with care.
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 8443 -j ACCEPT

# Allow outgoing Tor (http://tor.eff.org) requests.
# Note: Do _not_ use unencrypted protocols over Tor (sniffing is possible)!
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9001 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9002 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9030 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9031 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9090 -j ACCEPT
# $IPTABLES -A OUTPUT -m state --state NEW -p tcp --dport 9091 -j ACCEPT

# Allow outgoing OpenVPN requests.
$IPTABLES -A OUTPUT -m state --state NEW -p udp --dport 1194 -j ACCEPT

# TODO: ICQ, MSN, GTalk, Skype, Yahoo, etc...

# Selectively allow certain inbound connections, block the rest.

#-----------------------------------------------------------------------

# Allow incoming DNS requests.
$IPTABLES -A INPUT -m state --state NEW -p udp --dport 53 -j ACCEPT
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport 53 -j ACCEPT

# Allow incoming HTTP requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport 80 -j ACCEPT

# Allow incoming HTTPS requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport 443 -j ACCEPT

# Allow incoming POP3 requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport 110 -j ACCEPT
```

```
# Allow incoming IMAP4 requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport
143 -j ACCEPT

# Allow incoming POP3S requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport
995 -j ACCEPT

# Allow incoming SMTP requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport
25 -j ACCEPT

# Allow incoming SSH requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport
22 -j ACCEPT

# Allow incoming FTP requests.
$IPTABLES -A INPUT -m state --state NEW -p tcp --dport
21 -j ACCEPT

# Allow incoming NNTP requests.
# $IPTABLES -A INPUT -m state --state NEW -p tcp --dport
119 -j ACCEPT

# Allow incoming MySQL requests.
# $IPTABLES -A INPUT -m state --state NEW -p tcp --dport
3306 -j ACCEPT

# Allow incoming PLESK requests.
# $IPTABLES -A INPUT -m state --state NEW -p tcp --dport
8843 -j ACCEPT

# Allow incoming BitTorrent requests.
# TODO: Are these already handled by ACCEPTing
established/related traffic?
# $IPTABLES -A INPUT -m state --state NEW -p tcp --dport
6881 -j ACCEPT
# $IPTABLES -A INPUT -m state --state NEW -p udp --dport
6881 -j ACCEPT

# Allow incoming nc requests.
# $IPTABLES -A INPUT -m state --state NEW -p tcp --dport
2030 -j ACCEPT
# $IPTABLES -A INPUT -m state --state NEW -p udp --dport
2030 -j ACCEPT

# Explicitly log and reject everything else.
#------------------------------------------------------------------------

# Use REJECT instead of REJECTLOG if you don't
need/want logging.
$IPTABLES -A INPUT -j REJECTLOG
$IPTABLES -A OUTPUT -j REJECTLOG
$IPTABLES -A FORWARD -j REJECTLOG

#------------------------------------------------------------------------
# Testing the firewall.
#------------------------------------------------------------------------

# You should check/test that the firewall really works, using
# iptables -vnL, nmap, ping, telnet, ...

# Appending rules : Let's add some more IPv6 rules to our
firewall.

sudo ip6tables -A INPUT -p tcp --dport ssh -s
HOST_IPV6_IP -j ACCEPT
sudo ip6tables -A INPUT -p tcp --dport 80 -j ACCEPT
sudo ip6tables -A INPUT -p tcp --dport 21 -j ACCEPT
sudo ip6tables -A INPUT -p tcp --dport 25 -j ACCEPT

# To see the IPv6 rules with line numbers, type the
following command:

sudo ip6tables -L -n --line-numbers

# Deleting rules

sudo ip6tables -D INPUT -p tcp --dport 21 -j ACCEPT

# Exit gracefully.
#------------------------------------------------------------------------

exit 0
```

Puedes modificar las reglas que sean necesarias, una vez editado, guarda el archivo y cierra el editor.

Este script esta creado para servidores y vps, por tanto para crearlo debes ser root o usuario con permisos necesarios. En este caso no es necesario conceder permisos al script, ya los tenemos, tan solo ejecuta el script…

```
bash anti-ddos.sh
```

Qué son y cómo ejecutar archivos bin y run

Antes de aprender a ejecutar archivos **.bin y .run** en linux, vamos a ver qué son exactamente estas extensiones de archivos (ejecutables).

- **Qué son los archivos bin:** Un archivo bin, es un archivo binario que contiene paquetes de instalación (normalmente) ejecutables y autoextraíbles, que instalan o ejecutan una aplicación en nuestro sistema. No todo el software está disponible en tu Administrador de paquetes, sobre todo si hablamos de ultimas versiones (beta) o aplicaciones de pago.

- **Qué son los archivos run:** Los archivos run también son ejecutables, y normalmente se usan para lanzar los instaladores de software en Linux. Este tipo de archivos contienen los datos del programa a instalar y las instrucciones de la misma; es el formato más común para distribuir controladores de dispositivos (drivers) y aplicaciones.

Cómo ejecutar archivos bin y run
El proceso es muy similar en los dos tipos de archivo, por no decir iguales.

Ejecutar archivos bin
Desde la carpeta donde descargaste el archivo bin, abres la terminal de tu **distribución linux** y le concedes los permisos necesarios al **archivo bin** para que sea ejecutable.
sudo chmod +x archivo.bin

Ahora es tan sencillo como….
./archivo.bin

Ejecutar archivos run
El proceso de los archivos run es el mismo, primero lo hacemos ejecutable.
sudo chmod +x archivo.run

Solo falta lanzar la aplicación.
./archivo.run

Como ves…. más fácil imposible.

Razones para cambiar a linux

Prefieres **Windows** o **Linux**?. Si visitas **mi sitio web** sobre Linux, creo que ya se la respuesta, pero ¿cuáles son las razones para cambiar a Linux?.

Linux es un sistema operativo 100% libre, eso está claro; pero existen otros motivos de peso para que mudes a linux, si es que no lo has echo ya, en el articulo de hoy vemos las mas importantes.

Razones para cambiar a linux

Linux es gratis

Una de los motivos por los que debes cambiar a linux es su precio, independientemente de la distribución que elijas, Debian con Ubuntu y Linux MInt, derivados de Red Hat como Fedora o Centos, el propio Arch con Manjaro, y otras tantas que me dejo en el tintero; son gratis.

Salvo alguna excepción o distribución especifica, la practica totalidad de distros linux no solo son gratuitas, también son de **código abierto**. Te parece poco?, pues tranquilo que aun tenemos más, porque de la misma forma que se distribuye linux, igualmente se concede la inmensa mayoría de su software.

Soy consciente que «los de las ventanas» saldrán con lo de siempre… que si **photoshop**, que si no se que, bla,bla,bla, palabrería nada más. Vamos a ver, tu piensa, muchos son los que vienen a linux, muy pocos los que retornan a **Windows**.

Linux es estable

Tras más de 20 años usando linux y anteriormente otros sistemas **Unix**, puedo contar con los dedos de mis manos los bloqueos que me han ocurrido por culpa del sistema, algo común en otros sistemas con sus típicas pantallas azules más conocidas como **pantallazo de la muerte**, jaja. **Linux** es capaz casi al 100% sin reiniciarlo y sin peligro de cosas extrañas.

Si por algún caso, una aplicación o software se bloquea en tu Linux, puede matarlo con un solo comando de forma sencilla y confiable desde la terminal. En Windows, cruza los dedos y reza a San Pantuflo para que el administrador de tareas sea capaz de cerrar el proceso que genera inestabilidad.

Privacidad y Seguridad

Por su diseño, linux es un S.O. mucho más seguro que otros con coste. Nunca he visto ningún problema relacionado con la seguridad de un sistema que estuviera actualizado.

Cuando vemos que las grandes empresas del sector, como **Google**, **Amazon**, incluso el propio **Microsoft**, apenas sufren violaciones de seguridad en sus sistemas… por algo será, no pienses que es casualidad, todos usan linux en sus servidores críticos (incluso Microsoft).

Es evidente que nadie puede afirmar que linux es 100% seguro, pero que es el más seguro, si.

Soporte y compatibilidad

Soporte – Cambiar a linux

El soporte para Linux no tiene comparación con ningún otro sistema operativo. No importa qué problema tienes y que distribución uses, tutoriales, manuales, usuarios altruistas, etc, seguro que encontraras la solución muy rápido.

También tenemos sitios web como este, listas de correo, grupos y foros con el único propósito de ayudarte a resolver tus dudas y a que continúes aprendiendo. Ademas de la comunidad, cada distribución linux cuenta con sus propios manuales específicos y sistemas de foros, un buen ejemplo lo tenemos con la wiki de Arch Linux que es de lo mejor que puedes encontrar.

Compatibilidad – Cambiar a linux

En otros sistemas operativos, cada vez que lanzan una nueva versión

parece que tienes que actualizar tu hardware. Esto nos hace sospechar que tanto los desarrolladores de los sistemas operativos, como los del hardware están de acuerdo, si no es así que alguien me lo explique.

Tienen una política de fabricación, que ante los frecuentes cambios en los requisitos de diseño y especificaciones, todo se queda obsoleto de una forma casi insultante para el usuario. Es comercio abusivo, y pondría la mano en el fuego porque es una practica totalmente planificada por **Windows**, **Apple**, y los grandes fabricantes de **hardware**.

Y requieren ser reemplazados debido a . Hay, se llama obsolescencia planificada y Windows, Apple y los fabricantes de hardware son expertos en eso.

Afortunadamente, esta practica abusiva no existe en el movimiento linux, y ademas es uno de los motivos por los que miles de usuarios de **Windows 7** (termina su soporte) buscan refugio en distribuciones linux que sean similares estéticamente y fáciles de usar, te recomiendo revisar este articulo.

Conclusión final
Hemos visto las principales razones por las que deberías cambiar a Linux inmediatamente. Cada una de ellas es motivo suficiente para migrar, y eso que omitimos otras como la facilidad de uso, su excelente administración de paquetes, flexibilidad, actualizaciones constantes, velocidad, rendimiento, el peso del sistema operativo, y muchas más.

¡Linux te espera!

Este articulo no es una critica, es una realidad; soy consciente que Windows tiene derecho a existir al igual que todo bicho viviente, incluyendo partes orgánicas pestilentes. Aun así lo tengo claro, de mi plato no comerá, que le invite otro.

Canales de Telegram: Canal SoloLinux – Canal SoloWordpress

Espero que este articulo te sea de utilidad, puedes ayudarnos a mantener el servidor con una donación (paypal), o también colaborar con el simple gesto de compartir nuestros artículos en tu sitio web, blog, foro o redes sociales.

Sudo nos insulta por contraseña incorrecta

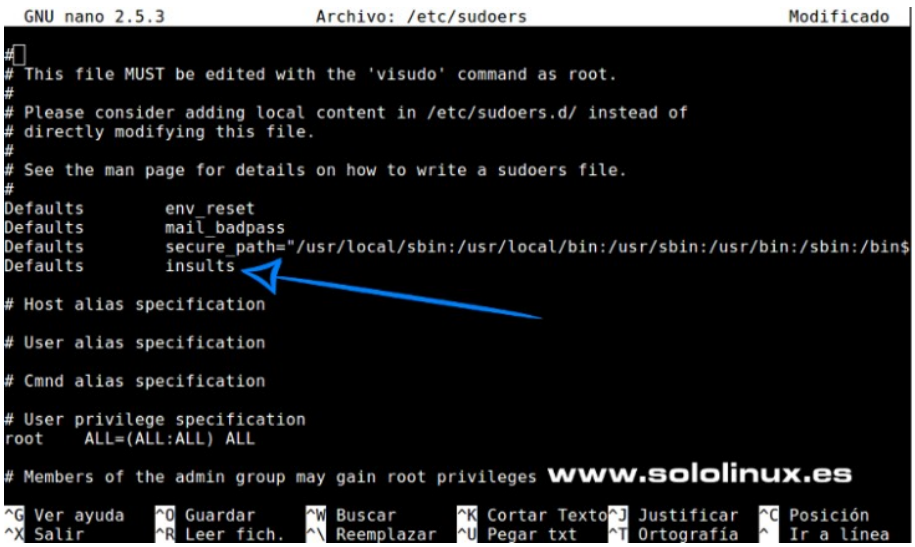
Existen cientos de tips para la **terminal linux**, uno de ellos bastante gracioso se dedica a insultarnos cuando nos equivocamos al introducir nuestra contraseña.

Este efecto tiene ya sus años, pero seguro que muchos usuarios no lo conocen. Su aplicación es muy simple, y su efecto por lo menos curioso, jaja; dependiendo de tu **distribución linux** los insultos e improperios pueden variar, así que no existe un patrón definido. Vemos como habilitar el efecto que por lo menos nos sacara una sonrisa.

Sudo nos insulta por contraseña incorrecta
Para aplicar sudo insulte tenemos que editar el archivo etc/sudoers.
sudo nano /etc/sudoers

Donde los «Defaults» agregamos otro «Defaults» con la orden «insults».
Defaults insults

Observa la imagen de ejemplo…



Guarda el archivo y cierra el editor.

Limpiamos la sesión sudo.
sudo -k

A partir de ahora cada vez que cometas un error con la **password**, sudo te insultara, jaja.

Instalar GameMode de Feral Interactive en linux

GameMode es una combinación de bibliotecas y demonios desarrollada por **Feral Interactive**, que mejora considerablemente el rendimiento de los **juegos en sistemas Linux**. No debes tener miedo a realizar estos cambios, los demonios y bibliotecas solo se aplican cuando juegas (temporal).

Fue creado de manera provisional para solucionar los errores con el ahorro de energía que provocaba **CPU governor**, el desarrollo funciono tan bien que ampliaron el **proyecto** con una gama de características y configuraciones de optimización realmente impresionantes.

- CPU governor
- I/O priority
- Process niceness
- Kernel scheduler (SCHED_ISO)
- Screensaver inhibiting
- GPU performance mode (**NVIDIA** and **AMD**), GPU overclocking (NVIDIA)
- Custom scripts

La herramienta es compatible con la gran mayoría de distribuciones linux, así que en este articulo vemos como **instalar GameMode en linux**.

Instalar GameMode en linux
La instalación de GameMode exige unas dependencias y herramientas que debemos cumplir, vemos como hacerlo en tu distribución linux.

Debian, Ubuntu, Linux Mint, y derivados:
sudo apt-get install meson libsystemd-dev pkg-config ninja-build git

CentOS, RHEL, y derivados:
sudo yum install meson systemd-devel pkg-config git

Fedora y derivados:
sudo dnf install meson systemd-devel pkg-config git

OpenSuse, Suse, y derivados:
sudo zypper install meson systemd-devel pkg-config git

Arch Linux, Manjaro, y derivados:
sudo pacman -S meson systemd ninja git

Una vez termines de cumplimentar los requisitos necesarios, clonamos el repositorio GIT oficial.
git clone https://github.com/FeralInteractive/gamemode.git

ejemplo…
sololinux # git clone https://github.com/FeralInteractive/gamemode.git
Clonar en «gamemode»…
remote: Enumerating objects: 96, done.
remote: Counting objects: 100% (96/96), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 2448 (delta 46), reused 59 (delta 28), pack-reused 2352
Receiving objects: 100% (2448/2448), 611.61 KiB | 0 bytes/s, done.
Resolving deltas: 100% (1751/1751), done.
Comprobando la conectividad… hecho.

Abrimos la carpeta gamemode.
cd gamemode

Creamos la herramienta.
./bootstrap.sh

Te hará unas preguntas:

- Confirmar la instalación: Y.
- Selecciona la ubicación; Y (recomendada la ubicación por defecto).
- Introduce tu password sudo: tu-pass.

Usar el GameMode
Bien, ya lo tienes instalado. Otro detalle importante es que de forma nativa poco a poco se van incorporando nuevos juegos a la herramienta (solo tienes que ejecutar el juego), en estos momentos (18-Dic-2019) son los siguientes:

- DiRT 4
- Shadow of the Tomb Raider – Definitive Edition
- Medieval II: Total War™ Collection
- Life is Strange 2
- Total War: THREE KINGDOMS
- Total War: WARHAMMER II
- Life is Strange: Before the Storm Deluxe Edition
- A Total War Saga: THRONES OF BRITANNIA
- Rise of the Tomb Raider: 20º Aniversario
- HITMAN – Game Of The Year Edition
- F1™ 2017
- A Total War Saga: FALL OF THE SAMURAI
- Warhammer 40,000: Dawn of War III
- DiRT Rally
- Mad Max
- Warhammer® 40,000®: Dawn of War® II – Master Collection
- Deus Ex: Mankind Divided™ – Digital Deluxe Edition
- Life Is Strange™
- Total War: WARHAMMER
- F1™ 2015
- GRID™ Autosport
- Alien: Isolation™
- Company of Heroes 2
- La Tierra Media™: Sombras de Mordor™ GOTY
- XCOM 2
- Empire: Total War Collection
- Tomb Raider

También funciona en otros juegos que tengas instalados en tu linux, ponemos como ejemplo SuperTuxCart:
gamemoderun supertuxkart

Cuanto tiempo tarda un script bash en ejecutarse

Tal vez pienses que este articulo es a modo de curiosidad, ni mucho menos querido lector, conocer el tiempo de ejecución es algo indispensable en entornos donde se lanzan muchos script y **tareas cron** para evitar posibles encolamientos y saturación del sistema.

Con el **comando time** podemos saber (entre otras cosas) cuanto tiempo tarda un **script bash** desde que comienza hasta que termina, ademas nos aporta el tiempo del usuario y del sistema. Vamos a ver como conseguir estos resultados.

Cuanto tiempo tarda un script bash en ejecutarse
Como ejemplo usamos un script de mantenimiento que uso en mis servidores.

Sintaxis:
time [script]

Comando time con script de ejemplo.
time bash libera.sh

El script se ejecuta, cuando termina imprime en pantalla algo similar a lo siguiente…

```
# datos de ejemplo
real    0m27.529s
user    0m0.074s
sys     0m1.365s
```

Como puedes ver no ofrece los resultados con microsegundos, los explicamos:

- **real:** Explicado de forma brusca, es el tiempo que tarda desde que pulsas la tecla ENTER, hasta que termina todo el proceso.
- **user:** Es el tiempo que el usuario (nosotros) a utilizado la CPU.
- **sys:** Es el tiempo que la CPU trabaja con el kernel.

También puedes medir el tiempo de cualquier comando, por ejemplo de update, observa…
time yum update

ejemplo de salida…

```
[root@demo ~]# time yum update
Complementos cargados:fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: centos.crazyfrogs.org
 * epel: mirror.freethought-internet.co.uk
 * extras: centos.crazyfrogs.org
 * updates: centos.crazyfrogs.org
No packages marked for update

real    0m3.886s
user    0m0.488s
sys     0m0.063s
```

Como hemos visto, time es más útil de lo que parece en un principio.

Scipt bash libera.sh
Bueno, dejando a un lado el articulo >>Cuanto tiempo tarda un script bash en ejecutarse<<, seguro que más de uno me manda un correo preguntando por el script que utilice en el ejemplo. Es un script bastante simple pero muy útil en el mantenimiento diario de un servidor o vps. Viene de premio con este post, jajaj.
nano libera.sh

Copia y pega lo siguiente.

```
#libera.sh
#!/bin/bash
#*****Created by SergioG.B.*****
#*****https://www.sololinux.es*****
echo "<-------------------------------------------------------->"
echo "<-------------------------------------------------------->"
# hostnamectl set-hostname sololinux
#
echo "Comprobando estado de memoria"
free
echo "OK - Verificacion completada"
echo "<-------------------------------------------------------->"

sleep 2s; echo “Limpieza de memoria cache y swap“;
echo "<-------------------------------------------------------->"

sleep 2s; echo "Deshabilitando HTTP"
killall -KILL httpd
echo "OK - Http deshabilitado"
echo "<-------------------------------------------------------->"

sleep 1s; echo "Deshabilitando Swap"
swapoff -a
echo "OK - Swap deshabilitado"
echo "<-------------------------------------------------------->"

sleep 2s; echo "Liberando pagecaches, dentries e inodes"
sync;sysctl -w vm.drop_caches=3;sync
echo "OK - Server liberado"
echo "<-------------------------------------------------------->"

sleep 3s; echo "Habilitando la Swap"
swapon -a
echo "OK - Swap habilitado"
echo "<-------------------------------------------------------->"

sleep 3s; echo "Habilitar HTTP y reinciar Nginx/MariaDB"
#systemctl restart mariadb
sleep 1s;
service httpd start
sleep 1s;
service nginx restart
echo "OK - Todo habilitado"
echo "<-------------------------------------------------------->"

sleep 3s; echo "Reiniciar-habilitar Memcached"
chkconfig memcached on
service memcached restart
service memcached start
echo "OK - Memcached habilitado"
echo "<-------------------------------------------------------->"
#service ddos restart
sleep 2s; echo "Verificacion rutinario"
echo "Script libera sh"
echo "Created by SergioG.B."
echo "https://www.sololinux.es"
echo "<-------------------------------------------------------->"

sleep 2s; free
echo ".....TODO CORRECTO....."
echo "<-------------------------------------------------------->"
echo "<-------------------------------------------------------->"
```

Lo modificas según tus necesidades.

Guarda el script, y cierra el editor. Lo ejecutas y listo.

Linux Mint 19.3 Tricia – Novedades y descarga

Como es habitual, todos los años para estas fechas **Linux Mint** nos regala una nueva versión. Como no podía ser menos, este año también. En este articulo veremos las novedades que nos presenta (alguna importante), y sus respectivas descargas.

Linux Mint sigue con su tradición de llamar a sus lanzamientos con nombres femeninos, esta ultima denominada **Linux Mint Tricia**, que es la continuación de **Linux Mint Tina** y de **Linux Mint Tessa**.

Esta nueva versión se basa en **Ubuntu 18.04.3 LTS** y su Kernel 5.0, pero no es la única novedad. Por fin se elimina **Tomboy** que la verdad es que me parecía una aplicación desfasada, su sustituta es **Gnote** que tiene otro aspecto ademas de un excelente soporte con pantallas HiDPI.

Gimp ya no viene instalado por defecto (se mantiene en sus repositorios), cosa que agradecerán los usuarios que no lo necesitan, las ultimas versiones son excesivamente pesadas; tranquilo, ahora tenemos Drawing que no tiene el nivel de **Gimp**, pero es perfecto para necesidades básicas de edición de imágenes, como recortar, cambiar el tamaño, agregar texto y más.

Linux Mint 19.3 Tricia
Aquí llega lo bueno, los reproductores de vídeo **VLC** y **Xplayer** se sustituyen por **Celluloid** (antes GNOME MPV) que es extremadamente ligero; los usuarios de equipos no actualizados lo agradecerán, sin duda un buen reproductor multimedia. Como ultimo añadido importante, **Linux Mint 19.3 Tricia** integra una herramienta que detecta posibles problemas en tu sistema y te dice como solucionarlos.

Se estrena el nuevo **logo de Linux Mint**, que no pierde su idea original pero si está mucho más trabajado y refinado.

Todos los escritorios han sido actualizados, y por fin veremos un Linux Mint XFCE con su ultima versión estable, **XFCE 4.14**.

Resumen de las nuevas características:

- Herramienta de errores del sistema.
- Ahora se permite establecer la zona horaria desde la configuración del idioma.
- Soporte mejorado de pantallas HiDPI.
- Celuloide sustituye a VLC.
- GNote sustituye a Tomboy.
- Drawing sustituye a GIMP
- Mejoras para establecer el tamaño del texto.
- Se permite configurar el menú contextual de Nemo.
- Se actualiza la GUI de bluetooth Blueberry.
- Mejoras notables en el rendimiento.
- Opción que deshabilita automáticamente el panel táctil al conectar un mouse.
- El logo de Linux Mint a sido actualizado.
- La pantalla gráfica del GRUB es más agradable visualmente.

Descargar Linux Mint 19.3 Tricia
Insertamos los enlaces oficiales para que puedas descargar **Linux Mint 19.3**, es recomendable usar los **enlaces torrent** porque ademas de compartir las descargas son más rápidas.

XFCE
XFCE 64bits: Descarga torrent – Descarga directa
XFCE 32bits: Descarga torrent – Descarga directa

MATE
MATE 64bits: Descarga torrent – Descarga directa
MATE 32bits: Descarga torrent – Descarga directa

Cinnamon
Cinnamon 64bits: Descarga torrent – Descarga directa
Cinnamon 32bits: Descarga torrent – Descarga directa

Eliminar dependencias de paquetes eliminados en CentOS

El **administrador de paquetes yum** elimina las dependencias al eliminar un paquete, pero no todas, siempre quedan librerías perdidas que con el paso del tiempo solo ocupan espacio. Esto tiene solución.

Para lograr que nuestro sistema quede lo más limpio posible al desinstalar una aplicación, debemos aplicar la directiva **clean_requirements_on_remove** (alguna versión ya la tiene integrada, no todas), y eso es lo que vemos en este articulo, no te preocupes que es algo muy simple.

Eliminar dependencias de paquetes eliminados
Debemos agregar la nueva directiva al archivo de configuración de yum, sigue los pasos que te indico que es algo muy simple. Editamos el archivo.
nano /etc/yum.conf

Veras algo similar a…
```
[main]
cachedir=/var/cache/yum/$basearch/$releasever
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
gpgcheck=1
plugins=1
installonly_limit=5
bugtracker_url=http://bugs.centos.org/set_project.php?project_id=23&ref=http://$
distroverpkg=centos-release

# This is the default, if you make this bigger yum won't see if the metadata
```

Bien, después de la ultima linea (antes de las que están comentadas), en este caso distroverpkg=centos-release, insertamos lo siguiente:
clean_requirements_on_remove=1

ejemplo…
```
installonly_limit=5
bugtracker_url=http://bugs.centos.org/set_project.php?project_id=23&ref=http://$
distroverpkg=centos-release
clean_requirements_on_remove=1
```

Guarda el archivo y cierra el editor.

Para limpiar todas las dependencias y librerías de anteriores aplicaciones, ejecutamos lo siguiente…
yum autoremove

Veremos en la pantalla la basura de nuestro CentOS, revisa y elimina.

```
================================================================================
 Package         Arquitectura
                       Versión                       Repositorio         Tamaño
================================================================================
Eliminando:
 cyrus-sasl-md5  x86_64 2.1.26-23.el7                 @base                80 k
 gd              x86_64 2.0.35-26.el7                 @base               542 k
 libopendkim     x86_64 2.11.0-0.1.el7                @epel               141 k
 perl-Date-Calc  noarch 6.3-14.el7                    @base               689 k
 tokyocabinet    x86_64 1.4.48-3.el7                  @base               1.3 M
Eliminando para las dependencias:
 libbsd          x86_64 0.8.3-1.el7                   @epel               254 k
 perl-Bit-Vector x86_64 7.3-3.el7                     @base               481 k
 perl-Carp-Clan  noarch 6.04-10.el7                   @base                46 k

Resumen de la transacción
================================================================================
Eliminar  5 Paquetes (+3 Paquetes dependientes)                www.sololinux.es
```

Listo, ya tenemos el sistema libre de librerías y dependencias innecesarias.

Instalar Soundconverter en Ubuntu y derivados

Instalar Soundconverter en Ubuntu y derivados

Soundconverter es una herramienta para convertir formatos de audio, es exclusiva de linux ademas de ser gratuita y de **código abierto**.

Esta aplicación es muy simple y fácil de usar; es capaz de convertir cualquier archivo que sea compatible con **GStreamer**, como por ejemplo: AAC, MP3, FLAC, Ogg Vorbis, MOV, M4A, AC3, WAV, AVI, MPEG, DTS, ALAC, MPC, Shorten, APE, SID , MOD, XM y S3M a los formatos Opus, MP3 AAC, Ogg Vorbis, FLAC y WAV.

Te aviso!!!, para acelerar la conversión utiliza varios núcleos de la CPU.

Realmente es una herramienta muy buena para ser gratis y su pequeño tamaño, y más si tenemos en cuenta que también nos permite extraer los archivos de audio de **vídeos**. Otra característica importante es que puedes cambiar los nombres de los archivos que conviertes y moverlos a nuevas carpetas de manera automática. Vemos como instalar **Soundconverter** en **Ubuntu**, **Linux Mint** y derivados.

Instalar Soundconverter en Ubuntu

Soundconverter es una aplicación veterana, la podemos encontrar en los repositorios de nuestro Ubuntu o Linux Mint incluso si son versiones anteriores.

Para instalar el **conversor de audio**, actualizamos e instalamos, así de fácil.

```
sudo apt update
```

```
sudo apt install soundconverter
```

Puedes ejecutarla desde tu menú de aplicaciones, o desde la terminal con el siguiente comando.

```
soundconverter
```

No tendrás ningún problema, tampoco es necesario un manual de uso ya que es muy intuitiva y simple.

Una compañía canadiense paga a los hackers para recuperar sus registros

La principal compañía canadiense de análisis clínicos, **LifeLabs**, a sufrido el mayor robo de datos que se recuerda en ese país.

El incidente ocurrido el pasado mes de octubre a salido ahora a la luz, y ojo.. no es una filtración mínima pues afecta a más de 15.000.000 millones de ciudadanos canadienses. Los datos robados son:

- Nombres y apellidos
- Direcciones
- Fechas de nacimiento
- Credenciales de acceso

Al parecer, los hackers solicitaron a la compañía el pago de una cantidad importante (no se conocen cifras), a cambio de no revelar públicamente los datos de sus clientes. Expusieron algunos como forma de presión.

El final de esta historia es evidente, LifeLabs paga a los hackers para recuperar la información.

Una compañía canadiense paga a los hackers

El equipo informático de LifeLabs decidido pagar el rescate solicitado por los hackers para mantener sus datos seguros, también se solicitó la colaboración de expertos en ciberseguridad (un poco tarde). La propia empresa lanzo este comunicado:

Carta abierta a los clientes de LifeLabs

A nuestros clientes:
A través de la vigilancia proactiva, LifeLabs identificó recientemente un ataque cibernético cuyo resultado fue el acceso no autorizado a nuestros sistemas informáticos, el contenido era información del cliente que incluía nombre, dirección, correo electrónico, inicio de sesión, contraseñas, fecha de nacimiento, número de tarjeta de salud y resultados de pruebas de laboratorio.
Personalmente, quiero decir que lamento que esto haya sucedido. A medida que intentamos resolver este problema, mi equipo y yo seguimos enfocados en los mejores intereses de nuestros clientes. Usted nos confía información importante sobre su salud, y es una responsabilidad grande.

Hemos tomado varias medidas para proteger la información de nuestros clientes, que incluyen:

- Involucrarse inmediatamente con expertos en seguridad cibernética de clase mundial para aislar y asegurar los sistemas afectados y determinar el alcance de la violación.
- Fortalecer aún más nuestros sistemas para evitar futuros incidentes.
- Recuperando los datos haciendo un pago. Lo hicimos en colaboración con expertos familiarizados con ciberataques y negociaciones con ciberdelincuentes.
- Comprometerse con las fuerzas policiales, que actualmente están investigando el asunto.
- Ofrecer servicios de protección de seguridad cibernética a nuestros clientes, como robo de identidad y seguro de protección contra fraudes.

Quiero enfatizar que en este momento, las empresas contratadas de seguridad cibernética han informado que el riesgo para nuestros clientes en relación con este ataque es mínimo y que no hay más filtraciones.

Hemos solucionado los problemas del sistema relacionados con la actividad delictiva y trabajamos durante todo el día para establecer medidas adicionales que protejan su información. En aras de la transparencia y según lo exigen las normas de privacidad, hacemos este anuncio para notificar a todos los clientes. Existe información relacionada con aproximadamente 15 millones de clientes en los sistemas informáticos a los que se accedió. La gran mayoría de estos clientes están en BC y Ontario, con relativamente pocos clientes en otros lugares. En el caso de los resultados de las pruebas de laboratorio, nuestras investigaciones indican que hay 85,000 clientes afectados desde 2016 o antes ubicados en Ontario; seguimos trabajando para identificar y contactar con estos clientes directamente.
Si bien tiene derecho a presentar una queja ante los comisionados de privacidad, ya les hemos notificado esta violación y están investigando el asunto. También hemos notificado a nuestros socios gubernamentales.

Si bien hemos estado tomando medidas en los últimos años para fortalecer nuestras defensas cibernéticas, este ataque nos ha servido como recordatorio de que debemos adelantarnos al cibercrimen, un problema generalizado en todo el mundo y en todos los sectores.

Cualquier cliente que esté preocupado por este incidente puede solicitar un año de protección gratuita, se incluye monitoreo de la web oscura y seguro contra robo de identidad.

Charles Brown
Presidente y CEO
LifeLabs

Los comisionados de Ontario y la Columbia Británica, dos de las provincias más afectadas, están investigando este robo de datos, sin embargo, sus propio expertos confirman que aún no hay suficiente información disponible para saber si ha sido culpa de la empresa, o es fallo del proveedor externo.

El creciente número de incidentes en **Canadá** sobre **ciberseguridad** preocupa a las autoridades. Las empresas parecen no aprender de los errores del pasado. Fuente: hispasec.com

Alpine Linux 3.11.0 – ese gran desconocido

A pesar de llevar casi 10 años entre nosotros, pocos usuarios conocen está **distribución linux** y aun menos los que se atreven a instalarla.

Esta distribución basada en **uClibcy** y **BusyBox** fue creada para equipos con pocos recursos, pero también para usuarios expertos en linux. Su instalación es compleja, pero si te atreves con ella dispone de una estupenda wiki que seguro te ayudara.

Fue creada por la comunidad para enrutadores, cortafuegos, redes VPN, y servidores VoIP; aunque és para trabajar en consola también permite la instalación de entornos de escritorio como **XFCE** o **Gnome**. Su principal virtud es la solidez y cuenta con características proactivas de seguridad como **PaX** y **SSP** (*System Security Plan*).

Dicho esto… **Alpine** ha lanzado su ultima versión, la 3.11.0. En este articulo vemos sus principales novedades y los enlaces de descarga.

Alpine Linux 3.11.0 – ese gran desconocido
Alpine es una de las mejores distribuciones linux que puedes encontrar (en su gama). Vemos sus principales novedades.

- Kernel Linux 5.4 (linux-lts)
- Ofrece soporte para Raspberry Pi 4 (aarch64 y armv7)
- Instalación fácil de XFCE, GNOME y KDE
- Permite agregar Vulkan
- Soporte MinGW-w64 y DXVK
- Se dispone de Rust en todas las arquitecturas excepto en s390x

Las actualizaciones que más nos llaman la atención:

- Linux 5.4.5
- GCC 9.2.0
- Busybox 1.31.1
- musl libc 1.1.24
- LLVM 9.0.0
- Go 1.13.4
- Node.js 12.14.0
- Python 3.8.0
- Perl 5.30.1
- Postgresql 12.1
- Ruby 2.6.5
- Rust 1.39.0
- Crystal 0.31.1
- Erlang 22.1
- Zabbix 4.4.3
- Nextcloud 17.0.2
- Git 2.24.1
- Xen 4.13.0
- Qemu 4.2.0

Nota: Se elimina linux-vanilla y Python 2; el contenido de /var/spool/mail ahora lo tenemos en /var/mail; la dependencia clamav-libunrar ya no viene instalada por defecto, si la necesitas debes instalarla tu mismo.

Descargar Alpine Linux 3.11.0
Alpine publica ocho versiones diferentes, aquí insertamos los enlaces de descarga de las versiones x86_64 (64bits):

- STANDART
- EXTENDED
- NETBOOT
- MINI ROOT FILESYSTEM
- VIRTUAL
- XEN
- RASPBERRY P
- GENERIC ARM

Para otras arquitecturas incluyendo x86 (32bits) visita su pagina oficial de descargas.

Acelerar mi web con dns-prefetch

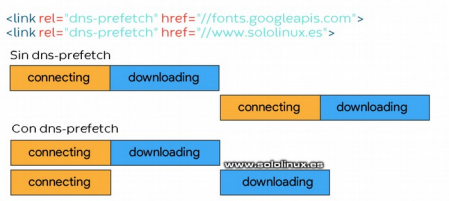
La técnica **dns-prefetch** o captación previa de **dns**, es una operación en la que el servidor de nombres de dominio se resuelve en segundo plano para una URL determinada, para que me entiendas mejor… carga la url del dominio antes de que hagas click en el.

La **captación previa de DNS** mejora el rendimiento del front-end de una web de forma considerable. Podemos decirle al navegar qué debe precargar y qué no debe precargar, sin ninguna intervención del usuario incluso antes de que el cliente visualice el recurso.

Una página web contiene recursos que se cargan desde varios dominios, la captación previa de DNS comunica al navegador web que hay activos web, como un archivo de datos, una imagen o un archivo de audio, que se van a necesitar más adelante; el navegador escucha la propuesta y resuelve el servidor de nombres de dominio en segundo plano, por ejemplo las **fuentes de google**.

Pero claro, como todo en está vida no todo es de color de rosa. Y es que **dns-prefretch** no siempre es compatible con todos los navegadores, solo con los modernos. Por ejemplo, si eres de los que aun utilizan el vetusto Internet explorer debes saber que IE 11 y anteriores no son compatibles. OJO!!!, eso no quiere decir que no veras la web, no, simplemente no puedes aprovecharte de esta aceleración extra.

En la siguiente imagen puedes ver como trabaja.



Acelerar mi web con dns-prefetch
Implantar la **captación previa de DNS** (dns-prefetch) es un proceso sencillo. Debes agregar una etiqueta que contenga **rel=»dns-prefetch»** entre los <head> y </head> de tu sitio web. Vemos un ejemplo:

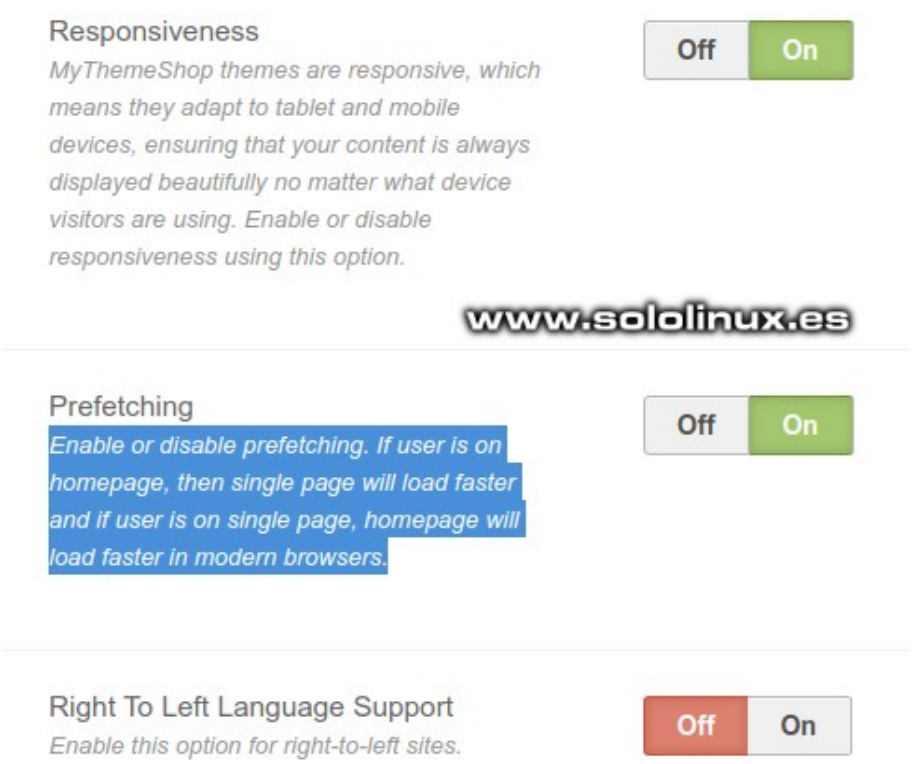
```
<link rel="dns-prefetch" href="//fonts.googleapis.com">
<link rel="dns-prefetch" href="//www.sololinux.es">
```

Ahora completo, tal como debería verse en tu sitio web.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <link rel="dns-prefetch" href="//fonts.googleapis.com">
    <link rel="dns-prefetch" href="//www.sololinux.es">
  </head>
  <body>
```

Como puedes ver **implantar dns-prefetch** es sencillo, ademas cuenta con la particularidad de que es un sistema muy extendido.

Si usas algún **CMS** por ejemplo **WordPress**, debes saber que es posible acelerar WordPress con esta técnica (ya que estamos te invito a visitar SoloWordpress.es que la tiene activada). Muchos de los plugins de cache la llevan incorporada, incluso algunas templates la incluyen para su uso interno. Mira la imagen siguiente:



Que DNS debo pre-cargar?
Bueno, pues esa es la gran pregunta jajaj. Todo depende de tu sitio web y lo que necesites, pero revisando las métricas de GTmetrix te puedes hacer una idea.

Aun dicho lo anterior te pongo un listado de las más comunes, muchas de ellas implantadas en nuestros sitios web.

```
XHTML
<!-- Google CDN -->
<link rel="dns-prefetch" href="//ajax.googleapis.com">
<!-- Google API -->
<link rel="dns-prefetch" href="//apis.google.com">
<!-- Google Fonts -->
<link rel="dns-prefetch" href="//fonts.googleapis.com">
<link rel="dns-prefetch" href="//fonts.gstatic.com">
<!-- Google Analytics -->
<link rel="dns-prefetch" href="//www.google-analytics.com">
<!-- Google Tag Manager -->
<link rel="dns-prefetch" href="//www.googletagmanager.com">
<!-- Google Publisher Tag -->
<link rel="dns-prefetch" href="//www.googletagservices.com">
<!-- Google AdSense -->
<link rel="dns-prefetch" href="//adservice.google.com">
<link rel="dns-prefetch" href="//pagead2.googlesyndication.com">
<link rel="dns-prefetch" href="//tpc.googlesyndication.com">
<!-- Google Blogger -->
<link rel="dns-prefetch" href="//bp.blogspot.com">
<link rel="dns-prefetch" href="//1.bp.blogspot.com">
<link rel="dns-prefetch" href="//2.bp.blogspot.com">
<link rel="dns-prefetch" href="//3.bp.blogspot.com">
<link rel="dns-prefetch" href="//4.bp.blogspot.com">
<!-- Microsoft CDN -->
<link rel="dns-prefetch" href="//ajax.microsoft.com">
<link rel="dns-prefetch" href="//ajax.aspnetcdn.com">
<!-- Amazon S3 -->
<link rel="dns-prefetch" href="//s3.amazonaws.com">
<!-- Cloudflare CDN -->
<link rel="dns-prefetch" href="//cdnjs.cloudflare.com">
<!-- jQuery CDN -->
<link rel="dns-prefetch" href="//code.jquery.com">
<!-- Bootstrap CDN -->
<link rel="dns-prefetch" href="//stackpath.bootstrapcdn.com">
<!-- Font Awesome CDN -->
<link rel="dns-prefetch" href="//use.fontawesome.com">
<!-- Facebook -->
<link rel="dns-prefetch" href="//connect.facebook.net">
<!-- Twitter -->
<link rel="dns-prefetch" href="//platform.twitter.com">
<!-- Linkedin -->
<link rel="dns-prefetch" href="//platform.linkedin.com">
<!-- Vimeo -->
<link rel="dns-prefetch" href="//player.vimeo.com">
<!-- GitHub -->
<link rel="dns-prefetch" href="//github.githubassets.com">
<!-- Disqus -->
<link rel="dns-prefetch" href="//referrer.disqus.com">
<link rel="dns-prefetch" href="//c.disquscdn.com">
<!-- Gravatar -->
<link rel="dns-prefetch" href="//0.gravatar.com">
<link rel="dns-prefetch" href="//2.gravatar.com">
<link rel="dns-prefetch" href="//1.gravatar.com">
<!-- BuySellads -->
<link rel="dns-prefetch" href="//stats.buysellads.com">
<link rel="dns-prefetch" href="//s3.buysellads.com">
<!-- DoubleClick -->
<link rel="dns-prefetch" href="//ad.doubleclick.net">
<link rel="dns-prefetch" href="//googleads.g.doubleclick.net">
<link rel="dns-prefetch" href="//stats.g.doubleclick.net">
<link rel="dns-prefetch" href="//cm.g.doubleclick.net">
```

Notas finales y apuntes varios
Recuerda que los navegadores web buscan el encabezado HTTP de **X-DNS-Prefetch-Control** (no distingue entre mayúsculas y minúsculas) con el valor ON / OFF, dependiendo de su respuesta cambia el comportamiento.

Si una página está excluida de la pre-carga se ignorarán los sucesivos reintentos.

La captación previa de DNS es compatible con los principales navegadores (**Safari 5.0+**, **IE 9.0**+ (no 100%, mejor usa el 11), **Firefox 3.5**+, **Google Chrome**). Lamentablemente parece que no interesa esta fabulosa técnica con dispositivos móviles (nos meten **AMP** hasta por las orejas), no es efectiva con **iOS Safari, Opera Mini y Android Browser**.

Actualizar el Grub en linux

Saber como **actualizar el Grub** correctamente no es una tontería; cada vez que actualizamos o hacemos una modificación en el **Grub** es necesario que la actualicemos si queremos que todo funcione como debería.

En este articulo tomamos como ejemplo **Linux Mint 18.3 Sylvia**, pero es valido para cualquier **distribución linux**.

Actualizar el Grub en linux
Modificar el grub, reparar el Grub, muchos artículos hemos publicado en **sololinux.es** sobre el tema. Para actualizar el Grup tan solo debes ejecutar el siguiente comando:

```
sudo update-grub
```

Si no te reconoce el comando puedes ejecutar…

```
grub2-mkconfig –o /boot/grub2/grub.cfg
```

Como ya comentamos antes, en nuestro ejemplo usamos **Linux Mint 18.3 Sylvia**.

```
sololinux ~ # sudo update-grub
Generando archivo de configuración grub...
Aviso: Ya no se permite establecer GRUB_TIMEOUT a un valor distinto de cero cuando GRUB
_HIDDEN_TIMEOUT está activado.
Encontrada imagen de linux: /boot/vmlinuz-4.15.0-72-generic
Encontrada imagen de memoria inicial: /boot/initrd.img-4.15.0-72-generic
Encontrada imagen de linux: /boot/vmlinuz-4.15.0-70-generic
Encontrada imagen de memoria inicial: /boot/initrd.img-4.15.0-70-generic
Encontrada imagen de linux: /boot/vmlinuz-4.10.0-38-generic
Encontrada imagen de memoria inicial: /boot/initrd.img-4.10.0-38-generic
Found memtest86+ image: /boot/memtest86+.elf
Found memtest86+ image: /boot/memtest86+.bin
hecho
sololinux ~ #
www.sololinux.es
```

Si observas la imagen anterior nos imprime en pantalla el siguiente error:

Aviso: Ya no se permite establecer GRUB_TIMEOUT a un valor distinto de cero cuando GRUB_HIDDEN_TIMEOUT está activado.

Este error se produce por la configuración del tiempo en el **Grub**, ejemplo:

```
GRUB_DEFAULT=0
GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=lsb_release -i -s 2> /dev/null ||
echo Debian
GRUB_CMDLINE_LINUX_DEFAULT=“”
GRUB_CMDLINE_LINUX=“”
```

La solución es simple, solo debes agregar un valor numérico (segundos), o comentar la linea “GRUB_HIDDEN_TIMEOUT=0”. Vemos un ejemplo…

```
GRUB_DEFAULT=0
#GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
GRUB_DISTRIBUTOR=lsb_release -i -s 2> /dev/null ||
echo Debian
GRUB_CMDLINE_LINUX_DEFAULT=“”
GRUB_CMDLINE_LINUX=“”
```

Ahora te forzara a que selecciones una opción en el **Grub**, y no lanzara ningún error.

Life in Strange 2 en linux

Por fin se ha lanzado el popular juego **Life in Strange 2** para sistemas basados en Linux. Una gran noticia para los seguidores de este juego, ya que hasta hace pocos días solo estaba disponible para Windows.

La trama del juego es emocionante, dos hermanos Sean y Daniel Díaz, se fugan de su casa después de un terrible accidente por miedo a que la policía los detenga. La historia está muy elaborada, es todo una sucesión de aventuras en el largo camino que tienen que recorrer hasta la ciudad natal de su padre, en México.

El drama de dos hermanos junto con mucha acción.

Life in Strange 2 en linux

En la versión especifica para linux nos encontramos con todas las DC, Episodio 1, Episodio 2, Episodio 3, Episodio 4, y Episodio 5 (así podemos comenzar la historia desde el principio).

Destacamos que de forma nativa es compatible con el GameMode de Feral, y es la misma Feral interactive la que lo distribuye en su tienda oficial.

Requisitos:

	Req. Mínimos	Req. Recomendados
Sistema operativo	Ubuntu 18.04 (64bit)	
Procesador	Intel Core i3-4130 3,4 GHz	Intel Core i5-6500 3,2 GHz
Memoria Ram	4GB	8GB
Espacio en disco	42GB	
Tarjeta gráfica	Nvidia GTX 680 / 2GB o sup.	Nvidia GTX 970 / 4GB o sup. AMD RX 470 / 4GB o sup.

Otros detalles sobre los requisitos mínimos:

- Necesita **Vulkan**.
- En las tarjetas gráficas Nvidia el driver mínimo es la versión 430.14.
- En las tarjetas gráficas AMD se requiere Mesa 19.1.2.
- Otras tarjetas gráficas, como por ejemplo las Intel, no son compatibles.

Nota final:
Ten en cuenta que Life is Strange 2 se activa con una clave a través de Steam, por tanto necesitas estar conectado a Internet y tener una cuenta gratuita en Steam.

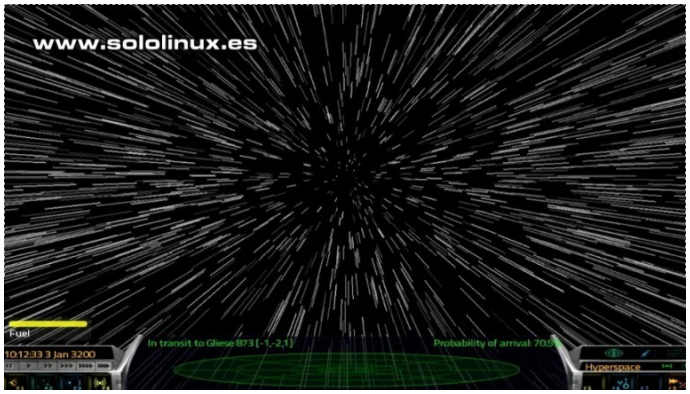
Instalar el juego Pioneer Space Trading en linux

Pioneer Space Trading es un **juego de combate espacial** totalmente gratis y de código abierto. Está disponible para Linux, Mac OS X y Microsoft Windows.

Si te gustan estas temáticas Pioneer Space es una excelente alternativa, ya que no te obliga a cumplir ninguna misión pre-establecida, puedes explorar todos los planetas de la galaxia sin ningún objetivo definido; tu juegas, tu decides.

Inspirado en el vetusto **Frontier Elite 2** y bajo licencia pública GNU de código abierto, este juego de aventura espacial mono-jugador es para uso exclusivo offline. Es similar a otros juegos populares de combate espacial como, Oolite o Endless Sky.

A diferencia de otros, el juego Pionner Space Trading recibe abundantes actualizaciones (ver su changelog) y eso es fundamental. En este articulo vemos como instalar el juego en nuestro linux.



Instalar el juego Pioneer Space Trading en linux

Desde su pagina oficial puedes descargar el paquete tar.gz e instalarlo manualmente; como alternativa y mucho más fácil… desde su paquete Flatpak.

Puedes verificar si tienes Flatpak en tu sistema, con:

```
flatpak --version
```

Ejemplo de salida…

```
sololinux ~ # flatpak –version
Flatpak 1.0.3
sololinux ~ #
```

Otra opción para saber si la tienes instalada, es buscar un paquete en su almacén de software (en el ejemplo buscamos Gimp).

```
flatpak search gimp
```

Ejemplo de salida…

```
sololinux ~ # flatpak search gimp
Application ID Version Remotes Description
org.gimp 2.10.14 flathub Cree imágenes y edite fotografías
org.glimpse_editor 0.1.0 flathub Cree imágenes y edite
fotografías
com.github.unrud 0.1.0 flathub Create small, searchable
PDFs from scanned documents
Sololinux ~ #
```

Muchas distribuciones ya vienen con Flatpak preinstalado, si no es tu caso, visita esta pagina, selecciona tu distribución linux, y sigue los pasos indicados.

Una vez tengas Flatpak en tu sistema, instala el juego con el siguiente comando.

```
flatpak install flathub net.pioneerspacesim.Pioneer
```

La instalación puede demorar un rato, el juego son 500MG así que se paciente. Cuando termine la instalación lo ejecutas directamente desde la terminal.

```
flatpak run net.pioneerspacesim.Pioneer
```

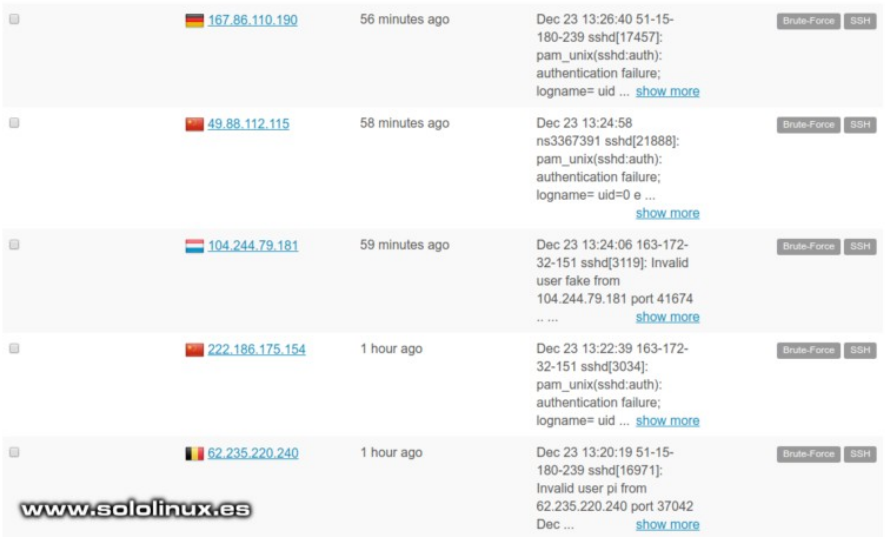
Te recomiendo que revises el manual oficial del juego.

Bloquear ataques de fuerza bruta al puerto SSH

Los ataques de fuerza bruta conocidos como brute force attack, son la pesadilla de servidores y sitios web. Todos sufrimos en mayor o menor medida estos intentos de intrusión, y si hay algo goloso para un lamer es obtener acceso a un servidor vía SSH.

OJO!!!, no debemos confundir los ataques de fuerza bruta con los que intentan una denegación de servicio (DDos),son diferentes, el DDos nos inunda a peticiones hasta que cae el servicio, y el de fuerza bruta intenta averiguar nuestro password a base de probar y probar. Es recomendable usar contraseñas seguras, en un anterior articulo vimos un script bash generador de passwords.

Tal vez no seas consciente de la cantidad de intentos de intrusión que se producen al cabo del día, yo te aseguro que son muchos. Nosotros reportamos automáticamente cada intento de ataque o intrusión a la base de datos de Abuseipdb, no solo de sololinux, de todos nuestros servidores.



Bloquear ataques de fuerza bruta al puerto SSH

La tarea de bloquear ataques de fuerza bruta al puerto SSH es sencilla si hacemos uso de las reglas de iptables. Vamos a implantar unas reglas que obligaran a iptableso nftables a bloquear cualquier ip, que haya intentado acceder a nuestro sistema vía ssh en un periodo de tiempo definido; a nosotros nos gusta 3 intentos en un tiempo máximo de 30 segundos, pero tu puedes poner el que quieras, por ejemplo 5 intentos en un minuto (60 segundos).

Supongo que tienes el puerto ssh abierto?, si no es el 22 modifica las reglas.

```
iptables -A INPUT  -p tcp --dport 22 -m state --state NEW -j ACCEPT
```

Insertamos las tres reglas en nuestra terminal linux (una por una):

```
iptables -A INPUT  -p tcp --dport 22 -m state --state NEW -m recent --set --name SSH
```

```
iptables -A INPUT  -p tcp --dport 22 -m state --state NEW -m recent --update --seconds 30 --hitcount 3 --rttl --name SSH -j LOG --log-prefix 'SSH-HIT-RATE: '
'
```

```
iptables -A INPUT  -p tcp --dport 22 -m state --state NEW -m recent --update --seconds 30 --hitcount 3 --rttl --name SSH -j DROP
```

Opciones que puedes modificar:

- –dport: por defecto 22, puedes modificarlo.
- –seconds: tiempo definido en segundos.
- –hitcount: intentos de acceso máximos permitidos en el tiempo definido.

Si te quieres asegurar que las reglas se implantaron correctamente, puedes ejecutar el siguiente comando y revisar iptables o nftables.

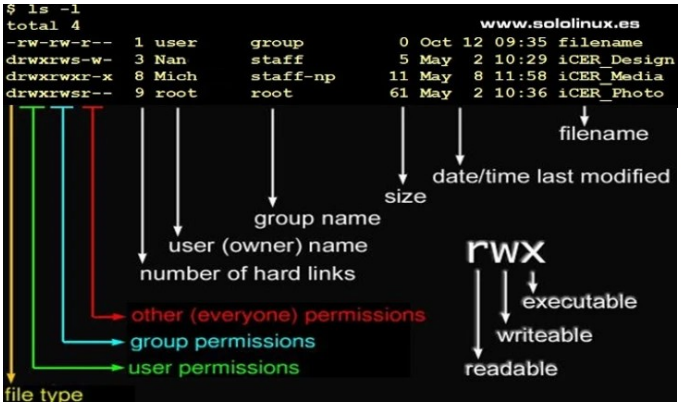
```
iptables -L
```

Las nuevas reglas generaran registros fácilmente identificables, todos tienen el prefijo <SSH-HIT-RATE>.

Los podemos visualizar con el comando less:

```
less +F /var/log/messages
```

Qué es el valor umask en linux



Umask (**user mask** o **máscara de usuario**), es un comando para entornos **POSIX** que determina los permisos predeterminados que se establecen cuando creamos un archivo o directorio (carpeta).

También hace referencia a la función que establece la máscara (**mask**), y a la máscara en sí; normalmente se conoce como la máscara de creación del modo de archivo.

En este articulo aprenderemos **qué es el valor umask en linux**, y como se desarrolla para lograr su fin.

Qué es el valor umask en linux

Ahora que ya sabemos lo que es realmente el valor **umask**, vamos a profundizar un poco más en el tema.

Valores predeterminados de umask

Los valores predeterminados pueden variar dependiendo del **administrador de sistemas** que lo explique, cada uno tiene sus preferencias, o simplemente están acostumbrados a definir sus propios valores dependiendo del sistema que manejen.

Una cosa si está clara, el valor **umask en linux** se establece según el propósito, como archivos, directorios, directorio de inicio para cualquier usuario, etc. En la siguiente tabla vemos los de uso general (común).

VALOR MASK	ARCHIVOS	DIRECTORIOS
000	666 (rw-rw-rw-)	777 (rwxrwxrwx)
002	664 (rw-rw-r--)	775 (rwxrwxr-x)
007	660 (rw-rw----)	770 (rwxrwx---)
022	644 (rw-r--r--)	755 (rwxr-xr-x)
27	640 (rw-r-----)	750 (rwxr-x---)
77	600 (rw-------)	700 (rwx------)
277	400 (r--------)	500 (r-x------)

Calcular el valor umask

En la siguiente tabla vemos la equivalencia de la **notación octal** con los permisos linux.

Valor octal	Permisos linux
0	Sin permiso
1	Solo ejecutar
2	Solo escribir
3	Ejecutar y escribir
4	Solo lectura
5	Leer y ejecutar
6	Leer y escribir
7	Permiso total

Tranquilo, no necesitas memorizar la tabla anterior, ya que los permisos se calculan con dígitos binarios. En la siguiente podemos ver como obtener los valores del 0 al 7.

Binarios	Octal	Permisos
2^0	1	Solo permite ejecutar
2^1	2	Solo permite escribir
2^2	4	Solo permite leer

Valor de umask

Según las tabla anterior, el permiso total para directorios o carpetas es 777, si tratamos archivos el permiso máximo sería 666.

Según lo visto hasta ahora, para conceder permisos predeterminados 755 se crea lo siguiente: **777 – 755 = 022**. Esto significa que el **valor umask** para tener **permisos 755, es 022**.

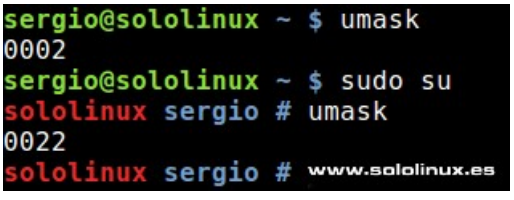
Si por ejemplo queremos que el permiso predefinido sea 700, entonces sería: **777 – 700 = 077** con un **valor umask de 077**.

Como ver el valor umask

Desde la terminal podemos conocer el valor umask predeterminado de nuestro usuario, es evidente que si eres root el valor es otro (como podemos ver en las siguientes imágenes). Podemos visualizar los valores de dos formas diferentes, en modo octal y en modo simbólico.

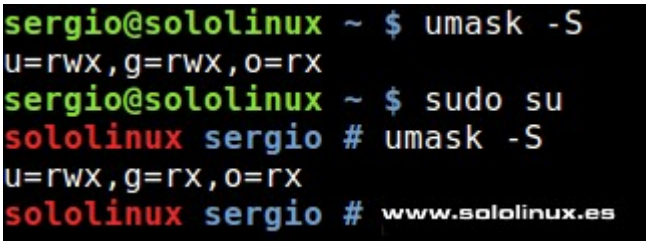
Modo octal:
Umask

Ejemplo de usuario normal y de usuario con permisos…



Modo simbólico:
umask -S

Ejemplo de usuario normal y de usuario con permisos…



Por qué el valor predeterminado tiene 4 dígitos
El primer numero normalmente es 0, y se reserva para otorgar permisos especiales. En un próximo articulo trataremos los permisos especiales, son varios y necesitan una buena explicación.

Cómo cambiar el valor predeterminado de umask
Puede cambiar el valor predeterminado de umask de dos formas, temporal o permanente; te recomiendo que antes de hacer una modificación que pueda ser drástica para el sistema, hagas la prueba en temporal.

La sintaxis para cambiar el valor de forma temporal es la siguiente:
umask new_valor_umask

ejemplo…
umask 0077

Si quieres que los cambios sean permanentes (no recomendado), agregas el valor de umask que quieras en «~/.bashrc» o «~/.bash_profile». Ahora cada vez que inicies sesión se actualizara el valor predeterminado de umask.

Ejemplos del comando curl

Curl es una herramienta en **línea de comandos** esencial para los usuarios de Linux, especialmente útil si tienes que transferir datos hacia, o desde un servidor independientemente del protocolo utilizado.

Admite una gran cantidad de protocolos, por ejemplo: HTTP/s, FTP/s, IMAP, POP3, SCP, SFTP, SMTP, TFTP, Telnet, LDAP, POP3S, RTMP, RTSP, DICT, FILE, PUT, Gopher, y muchos más. También soporta cookies, **password** de usuario, proxy tunelizado, etc…

Destacamos que la **utilidad curl** cuenta con una serie de opciones que nos permiten ampliar sus funciones, ademas permite transferir varios archivos a la vez.

Ejemplos del comando curl

La mejor forma de aprender es con ejemplos, por tanto así lo hacemos.

Descargar un archivo con curl

Para descargar un archivo de Internet puedes usar curl con la opción «-o», esta opción descargará y guardará el archivo en la carpeta actual con el mismo nombre que tiene en remoto.

```
curl -O https://sitioweb.com/archivo.tar.gz
```

Descargar varios archivos con curl

Si quieres descargar varios archivos de igual manera que la formula anterior, utilizando «-o», observa el siguiente ejemplo.

```
curl -O https://sitioweb.com/archivo.tar.gz
https://otraweb.com/archivo1.rar
```

Limitar la velocidad de las descargas

Si no tienes una conexión a Internet potente, puedes limitar el ancho de banda de las descargas (en el ejemplo a 500K).

```
curl --limit-rate 500K https://misitio.com/archivo.tar.gz -O
```

Descargar varios archivos desde una lista

Si nos ayudamos de Xargs, podemos crear un archivo de texto con el listado de los recursos a descargar (esta opción es especialmente útil cuando tenemos una gran cantidad de archivos a descargar).

```
xargs -n 1 curl -O < listadearchivos.txt
```

Reanudar descargas incompletas

La opción «-c» nos permite reanudar descargas que han sido interrumpidas.

```
curl -C - -O https://miweb.com/archivo.tar.gz
```

Descargar archivos a través de un proxy

El comando curl también nos permite descargar archivos en una red con proxy.

```
curl -x proxy.misitio.com:8080 -U user:usuario -O
http://misitio.com/miarchivo.tar.gz
```

Consultar el encabezado HTTP con curl

Con el encabezado HTTP obtenemos información adicional del servidor remoto.

```
curl -I www.sitiodemo.com
```

Pasar parámetros con curl

Curl nos permite publicar solicitudes a una url y pasar parámetros. En el ejemplo pasamos muestro nombre a un archivo.

```
curl --data "firstName=Name2&lastName=Name1"
https://miweb.com/info.php
```

Descargar archivos de un servidor FTP

Curl lo descarga todo, incluso archivos en un servidor ftp.

```
curl -u usuario:password -O ftp://misitio.com/archivo.tar.gz
```

Nota: si es un servidor ftp anónimo elimina «usuario:password».

Subir archivos a un servidor FTP con curl

La herramienta curl no solo permite descargar archivos de un servidor ftp remoto, también subirlos.

```
curl -u usuario:password -T archivoasubir.zip
ftp://misitio.com
```

Nota: si es un servidor ftp anónimo elimina «usuario:password».

Especificar el agente de usuario con curl

Con el comando curl, también podemos especificar el agente de usuario que se enviará a través de la solicitud https.

```
curl -I https://misitio.com --user-agent "AGENTE-DE-
USUARIO"
```

Guardar en un archivo las cookies de un sitio web

Esta opción nos permite guardar un archivo de texto en nuestro equipo de las cookies de un sitio web. En el ejemplo las descargamos de la CNN.

```
curl --cookie-jar cookiescnn.txt
https://edition.cnn.com/index.html -O
```

Enviar un archivo de cookies a un sitio web

Lo mismo que en el paso anterior pero a la inversa, ahora se las enviamos nosotros a ellos.

```
curl --cookie cookiescnn.txt https://edition.cnn.com
```

Verificar la versión de curl instalada

```
curl --version
```

Ejemplo de salida…

```
sololinux ~ # curl –version
curl 7.47.0 (x86_64-pc-linux-gnu) libcurl/7.47.0 GnuTLS/3.4.10 zlib/
1.2.8 libidn/1.32 librtmp/2.3
Protocols: dict file ftp ftps gopher http https imap imaps ldap ldaps
pop3 pop3s rtmp rtsp smb smbs smtp smtps telnet tftp
Features: AsynchDNS IDN IPv6 Largefile GSS-API Kerberos
SPNEGO NTLM NTLM_WB SSL libz TLS-SRP UnixSockets
```

Ejemplos del comando Tar

Copias de seguridad, compresión y archivado de datos, si eres usuario de linux seguro que conoces **Tar**, y si no es así… también (es muy habitual). En este articulo veremos los ejemplos de uso más comunes del comando.

Tar es un comando de los más utilizados, yo mismo lo uso a diario decenas de veces. Un detalle importante que muchos usuarios pasan por alto es, que para sacar el máximo provecho a la herramienta es importante que agrupes todos los archivos y carpetas en un solo archivo, se obtiene más ganancia.

Ejemplos del comando Tar
El nombre de **Tar** proviene de **Tape Archiver** (archivador en cinta), piensa que la herramienta data de 1979 y las copias de seguridad se guardaban en cintas (hasta no hace mucho aun era habitual encontrarlas).

El detalle anterior no es una simple anécdota, pues para saber como funciona **tar** debes conocer su origen; con esto quería explicaros que los **datos tar** se procesan y leen de forma lineal, eso quiere decir que para extraer un solo archivo debe recorrer todo hasta que lo encuentra.

Una vez explicada su manera de procesar vamos con los ejemplos de uso.

Crear un archivo tar
Para crear un tar es interesante usar las opciones «-cvf», las explicamos:

- **c** – crea un nuevo tar.
- **v** – indica la ruta donde se guarda el archivo tar.
- **f** – aplica un nombre al tar.

```
tar -cvf mibackup.tar /home/sololinux/backs/
```

Extraer un archivo tar
```
tar -xvf mibackup.tar
```

Crear un archivo tar.gz
Aplicando la opción «z» el archivo será un gzip.
```
tar cvzf mibackup.tar.gz /home/sololinux/backs/
o
tar cvzf mibackup.tgz /home/sololinux/backs/
```

Extraer un archivo tar.gz
```
tar -xvf mibackup.tar.gz
```

Crear un archivo tar.bz2
Tar.bz2 es una excelente opción a la hora de comprimir, pero la compresión y descompresión de archivos bz2 tarda más tiempo que con gzip.
```
tar cvfj mibackup.tar.bz2 /home/sololinux/backs/
```

Extraer un archivo tar.bz2
```
tar -xvf mibackup.tar.bz2
```

Listar el contenido de un archivo tar
```
tar -tvf mibackup.tar
```

Listar el contenido de un archivo tar.gz
```
tar -tvf mibackup.tar.gz
```

Listar el contenido de un archivo tar.bz2
```
tar -tvf mibackup.tar.bz2
```

Extraer un solo archivo de un archivo tar
```
tar -xvf mibackup.tar script.sh
```

Extraer un solo archivo de un archivo tar.gz
```
tar -zxvf mibackup.tar.gz script.sh
```

Extraer un solo archivo de un archivo tar.bz2
```
tar -jxvf mibackup.tar.bz2 script.sh
```

Extraer varios archivos del archivo tar, tar.gz y tar.bz2
```
tar -xvf mibackup.tar "script.sh" "otro.txt"

tar –zxvf mibackup.tar.gz "script.sh" "otro.txt"

tar -jxvf mibackup.tar.bz2 "script.sh" "otro.txt"
```

Agregar archivos o carpetas al archivo tar, tar.gz, tar.bz2
```
tar -rvf mibackup.tar archivo.txt

tar -rvf mibackup.tar.gz archivo.txt

tar -rvf mibackup.tar.bz2 archivo.txt
```

Verificar el tamaño del tar, tar.gz y tar.bz2
```
tar -czf - mibackup.tar | wc -c

tar -czf - mibackup.tar.gz | wc -c

tar -czf - mibackup.tar.bz2 | wc -c
```

Registrar la ip real del cliente en Apache

Para lograr más velocidad y seguridad, es una practica común instalar nuestros **servidores web apache** detrás de un proxy de equilibrio de carga como **Nginx** o **Haproxy** (Haproxy Load Balancer a resucitado).

También es frecuente que el **servidor Apache** este instalado detrás de un proxy de **almacenamiento caché**, como **Squid** o un **proxy** del tipo **BlueCoat** (actualmente de Symantec),

Las practicas mencionadas anteriormente son perfectamente validas, incluso recomendadas. Pero tienen un problema, en los registros de Apache la **dirección IP real del cliente** se reemplaza por la dirección IP del proxy.

Para evitar este problema, **Squid** desarrolló un encabezado de solicitudes HTTP personalizado, se conoce como **X-Forwards-For**. Actualmente es el encabezado estándar que identifica el origen real de la IP de un cliente conectado a través de un proxy HTTP, o un balanceador de carga.

Esta cabecera depura las estadísticas, y genera contenido dependiente de la ubicación. La información generada se ofrece de forma publica al **registro de Apache** para ser publicada (en este tema existe una gran controversia en la comunidad, recuerda que hablamos de datos sensibles del usuario).

Nada es perfecto, X-Forwards-For, tampoco.

Bueno, parece que con este encabezado si registramos las IP de los clientes, pero ojo!!!, si haces peticiones directamente a tu Apache sin pasar por el proxy, no se registran. En este articulo vemos como registrar todas las ip reales de los clientes en Apache.

Registrar la ip real del cliente en Apache

Para lograr nuestro objetivo debemos editar el archivo httpd.conf. Dependiendo del linux que tengas instalado en el servidor su localización puede variar, vemos las dos rutas posibles.

```
sudo nano/etc/httpd/conf/httpd.conf
```

o

```
sudo nano/usr/local/apache2/httpd.conf
```

Ahora busca la siguiente linea:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
```

Ahora debes cambiar <%h>, por:

```
%{X-Forwarded-For}i
```

```
# LogLevel: Control the number of messages logged to the error_log.
# Possible values include: debug, info, notice, warn, error, crit,        www.sololinux.es
# alert, emerg.
#
LogLevel warn

<IfModule log_config_module>
    #
    # The following directives define some format nicknames for use with
    # a CustomLog directive (see below).
    #
    LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
    LogFormat "%h %l %u %t \"%r\" %>s %b" common

    <IfModule logio_module>
      # You need to enable mod_logio.c to use %I and %O
      LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I %O" combinedio
    </IfModule>
```

La cadena quedara de la siguiente forma:

```
LogFormat “%{X-Forwarded-For}i %l %u %t \”%r\” %>s %b \”%{Referer}i\” \”%{User-Agent}i\”” combined”
```

Guarda el archivo y cierra el editor.

Como ultimo paso reiniciamos Apache.

```
sudo systemctl restart httpd
```

o

```
sudo systemctl restart apache2
```

Modificar la solicitud de inicio de sesión en shell bash

Al iniciar la **shell bash** en linux (también conocida como terminal o consola), y después de introducir nuestro nombre de usuario y contraseña, nos aparece el indicador de shell.

sergio@sololinux ~ $
sololinux sergio #

Cuando lo ejecutamos de forma interactiva, y cuando está preparado para leer un comando, **bash** nos muestra el indicador primario **PS1**. El indicador secundario **PS2** aparece cuando necesita más información para completar un comando.

Bash nos permite personalizar las cadenas de mensajes, insertando barra invertida y caracteres especiales. En la siguiente tabla vemos como se decodifican los caracteres.

Carácter	Que mensaje insertan
\a	an ASCII bell character (07)
\d	the date in "Weekday Month Date" format (e.g., "Tue May 26")
\D	the format is passed to strftime(3) and the result is inserted into the prompt string; an empty format results in a locale-specific time representation. The braces are required
\e	an ASCII escape character (033)
\h	the hostname up to the first `.'
\H	the hostname
\j	the number of jobs currently managed by the shell
\l	the basename of the shell's terminal device name
\n	newline
\r	carriage return
\s	the name of the shell, the basename of $0 (the portion following the final slash)
\t	the current time in 24-hour HH:MM:SS format
\T	the current time in 12-hour HH:MM:SS format
\@	the current time in 12-hour am/pm format
\A	the current time in 24-hour HH:MM format
\u	the username of the current user
\v	the version of bash (e.g., 4.3)
\V	the release of bash, version + patch level (e.g., 4.3.48)
\w	the current working directory, with $HOME abbreviated with a tilde (uses the value of the PROMPT_DIRTRIM variable)
\W	the basename of the current working directory, with $HOME abbreviated with a tilde
\!	the history number of this command
\#	the command number of this command
\$	if the effective UID is 0, a #, otherwise a $
\nnn	the character corresponding to the octal number nnn
\\	a backslash
\[	begin a sequence of non-printing characters, which could be used to embed a terminal control sequence into the prompt

Modificar la solicitud de inicio de sesión en shell bash

En este articulo vemos varias cadenas con ejemplos, que te ayudaran a personalizar el inicio de sesión de tu terminal.

Mostrar el nombre de host

```
export PS1='[\h :~]# '
```

```
[root@51-15-180-239 ~]# export PS1='[\h :~]# '
[51-15-180-239 :~]#
```

Mostrar el directorio de trabajo actual

```
export PS1='[\w :~]# '
```

```
[root@demo ~]# export PS1='[\w :~]# '
[~ :~]#
```

Mostrar el host y el directorio de trabajo actual con la ruta completa

```
export PS1='[\h:\w :~]# '
```

```
sergio@sololinux ~ $ export PS1='[\h:\w :~]# '
[sololinux:~ :~]#
```

Mostrar el host y el directorio de trabajo actual

```
export PS1='[\h \W :~]# '
```

```
sergio@sololinux ~ $ export PS1='[\h \W :~]# '
[sololinux ~ :~]#
```

Mostrar el usuario, el host y el directorio de trabajo actual con ruta

```
export PS1='[\u@\h \w :~]# '
```

```
sergio@sololinux ~ $ export PS1='[\u@\h \w :~]# '
[sergio@sololinux ~ :~]#
```

Mostrar el usuario, el host y el directorio de trabajo actual

```
export PS1='[\u@\h \W :~]# '
```

```
sergio@sololinux ~ $ export PS1='[\u@\h \W :~]# '
[sergio@sololinux ~ :~]#
```

Mostrar el usuario, el FQDN y el directorio de trabajo actual con ruta

```
export PS1='[\u@\H \w]# '
```

```
sergio@sololinux ~ $ export PS1='[\u@\H \w]# '
[sergio@sololinux ~]#
```

Mostrar la fecha, el host y el nombre de usuario

```
export PS1='[\u@\h \d]# '
```

```
sergio@sololinux ~ $ export PS1='[\u@\h \d]# '
[sergio@sololinux sáb dic 28]#
```

Mostrar la hora, el host y el nombre de usuario

```
export PS1='[\u@\h \A ~]# '
```

```
sergio@sololinux ~ $ export PS1='[\u@\h \A ~]# '
[sergio@sololinux 09:10 ~]#
```

Mostrar el host, el nombre de usuario y el shell (en este caso bash)

```
export PS1='[\u@\h \s ~]# '
```

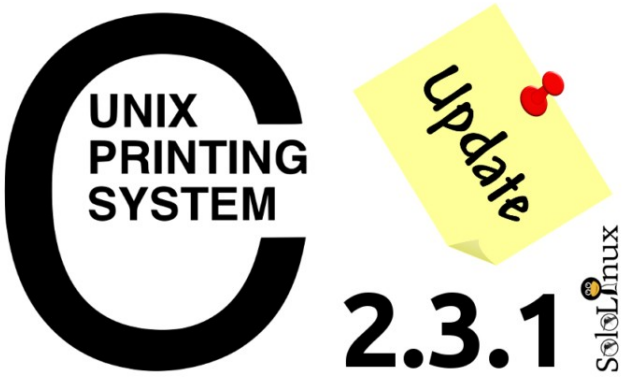
```
sergio@sololinux ~ $ export PS1='[\u@\h \s ~]# '
[sergio@sololinux bash ~]#
```

Cambiar la shell a sh

```
sh
```

```
sergio@sololinux ~ $ sh
$ sudo su
sololinux sergio # sh
#
```

Actualizar Cups en linux



Common Unix Printing System, más conocido como **Cups** (Sistema de impresión común de Unix), es un servidor de impresión modular para sistemas basados en **Unix**.

Cuando instalamos la **herramienta Cups**, esta actúa como un servidor de impresión que acepta trabajos locales o en red.

Cups es standard, y eso puede darnos algún dolor de cabeza a la hora de instalar una impresora en nuestro sistema. La mayoría de los entornos de escritorio tienen su propia mascara de cups (para instalar una impresora), pero a mi humilde parecer… dan más problemas que otra cosa.

Si quieres **instalar una impresora**, lo ideal es abrir tu navegador web favorito y acceder a la siguiente url:

http://localhost:631

Te aparecerá en pantalla la aplicación nativa de Cups, en ella podrás agregar y configurar tu impresora incluyendo parámetros adicionales. Es realmente buena y potente, mucho mejor que la que viene con los entornos de escritorio.

Después de unos años anclados en la versión **2.2.x,** en 2019 por fin se dio el salto a la nueva **Cups 2.3.0**. La gran mayoría de **distribuciones linux** siguen montando **Cups 2.2.x** o incluso **Cups 2.1.x**, y hoy en dia eso supone un atraso. En este articulo vemos como actualizar a la ultima versión de Cups, pero ojo!!!, es un proceso delicado, así que bajo tu responsabilidad.

Actualizar Cups en linux

Si accedemos a la url que mencionamos anteriormente veremos la versión actual.

Hoy 28 de diciembre del 2019 la ultima versión estable es la 2.3.1, por tanto instalamos esa. Antes de continuar, quiero decirte que este método a sido probado en **Ubuntu 16.04** con un resultado satisfactorio.

Descargamos la ultima versión.

```
wget
https://github.com/apple/cups/releases/download/v2.3.1/cups-2.3.1-source.tar.gz
```

Descomprimimos el paquete tar, como vimos en el articulo anterior.

```
tar xzvf cups-2.3.1-source.tar.gz
```

Accedemos al directorio.

```
cd cups-2.3.1
```

Compilamos e instalamos Cups 2.3.1.

```
./configure

make

make check

sudo make install
```

He leído en algún articulo por ahí, que reiniciando Cups inicia automáticamente la ultima versión.

```
sudo service cups restart
```

Probando en varios sistemas observo que no inicia correctamente, lo mejor es que reinicies tu linux.

```
reboot
```

Una vez arranque linux, inicia de manera manual Cups.

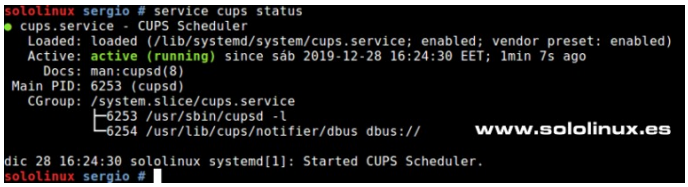
```
service cups start
```

Ahora recargamos demonios.

```
sudo systemctl daemon-reload
```

Verificamos que Cups este en marcha.

```
service cups status
```



Para concluir accedemos a la url del administrador Cups, en el veremos que está corriendo la ultima versión estable.

http://localhost:631

Si has llegado a este punto es porque todo ha salido bien, felicidades.

Habilitar Gzip en Apache

GNU ZIP, más conocido como **gzip**. Es el compresor por excelencia de cualquier sitio web que se precie.

Seguro que te suena la regla **mod_deflate** de tu **htaccess**, también es posible que cuando analizas tu sitio web en las herramientas **Gtmetrix** o **pagespeed de google** te indiquen que tienes recursos sin comprimir.

En ese momento tu piensas… ¿como puede ser?, tengo el **mod_deflate** en mi archivo **htaccess**. Tranquilo, lo que pasa es que no tienes configurada la regla en tu servidor. De todas formas antes de comenzar quiero decir que existen otras alternativas a **Gzip**, pero no vale la pena ni que las pruebes (de momento), hoy en día Gzip es el rey de la **compresión web**.

La verdad es que es un tema bastante simple, no quiero perder más tiempo con esto. Vamos directamente a la solución.

Habilitar Gzip en Apache
Lo primero que debes hacer es asegurarte que el modulo deflate esta activado en Apache.

```
httpd -M | grep deflate
```

Ejemplo de salida con el modulo habilitado…

```
[root@host ~]# httpd -M | grep deflate
deflate_module (shared)
```

Creamos el archivo deflate.conf en /etc/httpd/conf.d.

```
nano /etc/httpd/conf.d/deflate.conf
```

Copia y pega lo siguiente:

```
<IfModule mod_deflate.c>
# Compress HTML, CSS, JavaScript, Text, XML and fonts
AddOutputFilterByType DEFLATE application/javascript
AddOutputFilterByType DEFLATE application/rss+xml
AddOutputFilterByType DEFLATE application/vnd.ms-fontobject
AddOutputFilterByType DEFLATE application/x-font
AddOutputFilterByType DEFLATE application/x-font-opentype
AddOutputFilterByType DEFLATE application/x-font-otf
AddOutputFilterByType DEFLATE application/x-font-truetype
AddOutputFilterByType DEFLATE application/x-font-ttf
AddOutputFilterByType DEFLATE application/x-javascript
AddOutputFilterByType DEFLATE application/xhtml+xml
AddOutputFilterByType DEFLATE application/xml
AddOutputFilterByType DEFLATE font/opentype
AddOutputFilterByType DEFLATE font/otf
AddOutputFilterByType DEFLATE font/ttf
AddOutputFilterByType DEFLATE image/svg+xml
AddOutputFilterByType DEFLATE image/x-icon
AddOutputFilterByType DEFLATE text/css
AddOutputFilterByType DEFLATE text/html
AddOutputFilterByType DEFLATE text/javascript
AddOutputFilterByType DEFLATE text/plain
AddOutputFilterByType DEFLATE text/xml
# Remove browser bugs (only needed for really old browsers)
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4.0[678] no-gzip
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
Header append Vary User-Agent
</IfModule>
```

Guarda el archivo y cierra el editor.

Reinicia el servicio.

```
service httpd restart
```

Ya lo tienes habilitado. En el próximo articulo veremos como hacer lo mismo en **Nginx**.

Habilitar Gzip en Nginx

Aplicaciones de compresión como Gzip reducen el tamaño de un archivo web de forma considerable, esto repercute de forma directa en como se acelera la transferencia de archivos web, con el consecuente ahorro de ancho de banda.

En el articulo anterior, vimos como habilitar y aplicar la compresión Gzip en un servidor **Apache** para todos los sitios alojados en el mismo. Como no podía ser menos, le llega el turno a Nginx (importante si tienes un **vps** o **servidor** con **Apache httpd** y **Nginx** como **proxy inverso**).

Habilitar Gzip en Nginx
Antes de ver como **habilitar Nginx** quiero darte un consejo, Gzip puede estar habilitado tanto en Apache como en Nginx, no genera ningún problema de incompatibilidad. Si utilizas un **panel de control web** como **VestaCP** (por defecto es Apache + Nginx inverso), te recomiendo que habilites Gzip en Apache y en Nginx.

Bueno, lo primero que debemos hacer es crear el archivo de configuración.

```
touch /etc/nginx/conf.d/gzip.conf
```

Editamos el archivo de configuración.

```
nano /etc/nginx/conf.d/gzip.conf
```

Ahora, copia y pega lo siguiente.

```
gzip on;
gzip_disable "MSIE [1-6]\\.(?!.*SV1)";
gzip_proxied any;
gzip_comp_level 5;
gzip_types text/plain text/css application/javascript application/x-javascript text/xml application/xml application/rss+xml text/javascript image/x-icon image/bmp image/svg+xml;
gzip_vary on;
```

Guarda el archivo y cierra el editor.

Verificamos que no existe ningún error en Nginx.

```
nginx -t
```

Ejemplo valido…

```
[root@sololinux ~]# nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Recargamos la configuración de Nginx.

```
service nginx reload
```

ejemplo de salida correcto…

```
[root@sololinux ~]# service nginx reload
Reloading nginx configuration (via systemctl): [ OK ]
```

Ya lo tenemos instalado y funcionando. Puedes verificar que un sitio web en particular usa Gzip con el siguiente comando.

```
curl -s -H "Accept-Encoding: gzip" -I https://midominio.com --insecure | grep -i Content-Encoding
```

Como ejemplo verificamos que sololinux.es hace uso de Gzip.

```
[root@sololinux ~]# curl -s -H «Accept-Encoding: gzip» -I https://www.sololinux.es –insecure | grep -i Content-Encoding
Content-Encoding: gzip
```

Correcto, la salida nos indica que **sololinux.es** trabaja con **Gzip**.

Merry Christmas

Happy New Year