

Tarea 8: Instalación de Nginx en linux

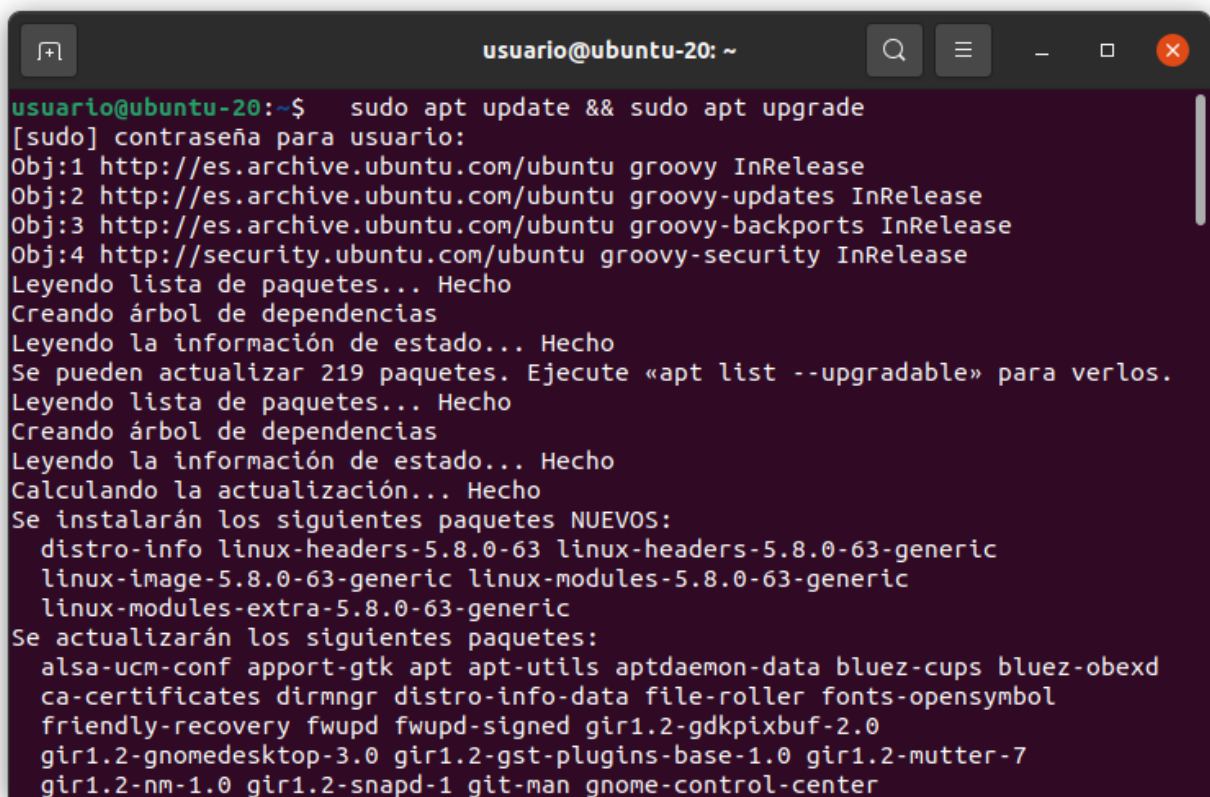


Índice

1. Actualización de repositorios
2. Instalación de Nginx
3. Aplicación de ajustes al firewall
4. Comprobación del servidor web
5. Administración del proceso de Nginx
6. Configuración de bloques del servidor

1. Actualización de repositorios

Primero vamos a actualizar el repositorio y el sistema operativo para que la instalación de apache sea mas segura.



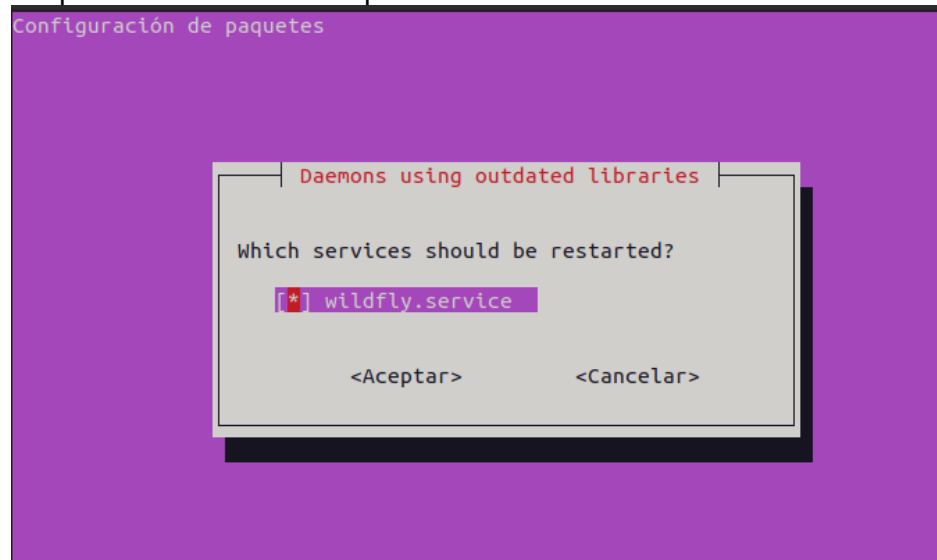
```
usuario@ubuntu-20: ~  
usuario@ubuntu-20:~$ sudo apt update && sudo apt upgrade  
[sudo] contraseña para usuario:  
Obj:1 http://es.archive.ubuntu.com/ubuntu groovy InRelease  
Obj:2 http://es.archive.ubuntu.com/ubuntu groovy-updates InRelease  
Obj:3 http://es.archive.ubuntu.com/ubuntu groovy-backports InRelease  
Obj:4 http://security.ubuntu.com/ubuntu groovy-security InRelease  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se pueden actualizar 219 paquetes. Ejecute «apt list --upgradable» para verlos.  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Calculando la actualización... Hecho  
Se instalarán los siguientes paquetes NUEVOS:  
  distro-info linux-headers-5.8.0-63 linux-headers-5.8.0-63-generic  
  linux-image-5.8.0-63-generic linux-modules-5.8.0-63-generic  
  linux-modules-extra-5.8.0-63-generic  
Se actualizarán los siguientes paquetes:  
  alsa-ucm-conf apport-gtk apt apt-utils aptdaemon-data bluez-cups bluez-obexd  
  ca-certificates dirmngr distro-info-data file-roller fonts-opensymbol  
  friendly-recovery fwupd fwupd-signed gir1.2-gdkpixbuf-2.0  
  gir1.2-gnomedesktop-3.0 gir1.2-gst-plugins-base-1.0 gir1.2-mutter-7  
  gir1.2-nm-1.0 gir1.2-snapd-1 git-man gnome-control-center
```

2. Instalación de Nginx

Ahora vamos a instalar nginx

```
usuario@ubuntu-20:~$ sudo apt install nginx
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream libnginx-mod-stream-geoip2
  nginx-common nginx-core
Paquetes sugeridos:
  fcgiwrap nginx-doc
```

Aceptamos las condiciones presentadas



3. Aplicacion de ajustes al firewall

Antes de iniciar Nginx, debemos aplicar algunos ajustes en el firewall para permitir su acceso utilizaremos el siguiente comando.

```
usuario@ubuntu-20:~$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
```

y permitimos el tráfico con el puerto 80

```
usuario@ubuntu-20:~$ sudo ufw allow 'Nginx HTTP'
Regla añadida
Regla añadida (v6)
```

luego verificamos el cambio y según el resultado que nos muestre indicara el tráfico de HTTP que se permite.

```
usuario@ubuntu-20:~$ sudo ufw status
Estado: activo

Hasta          Acción      Desde
-----
Apache         ALLOW       Anywhere
8083/tcp       ALLOW       Anywhere
Nginx HTTP     ALLOW       Anywhere
Apache (v6)    ALLOW       Anywhere (v6)
8083/tcp (v6)  ALLOW       Anywhere (v6)
Nginx HTTP (v6) ALLOW       Anywhere (v6)
```

4. Comprobación del servidor web

Ahora vamos a comprobar que iniciamos Nginx

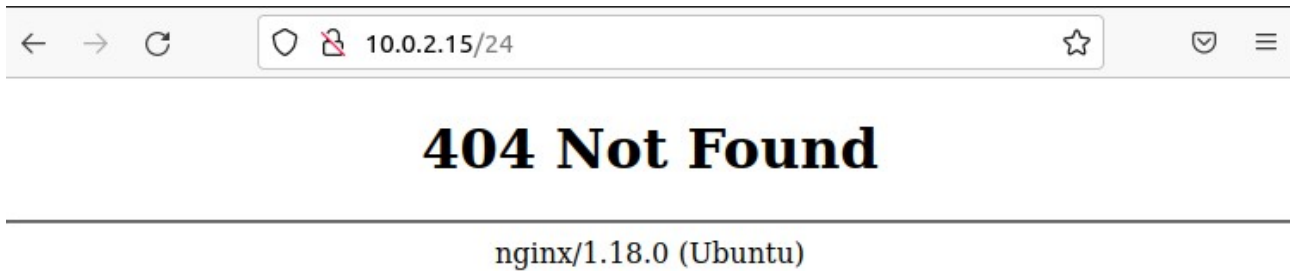
```
usuario@ubuntu-20:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset:
   Active: active (running) since Tue 2021-10-26 18:07:13 CEST; 18min ago
     Docs: man:nginx(8)
    Main PID: 3518 (nginx)
      Tasks: 3 (limit: 2308)
     Memory: 5.0M
    CGroup: /system.slice/nginx.service
            └─3518 nginx: master process /usr/sbin/nginx -g daemon on; master_
               └─3521 nginx: worker process
                 └─3522 nginx: worker process

oct 26 18:07:13 ubuntu-20 systemd[1]: Starting A high performance web server an
oct 26 18:07:13 ubuntu-20 systemd[1]: Started A high performance web server and
```

Posteriormente vamos a acceder a la pagina, para ello podremos conocer la ip escribiendo lo siguiente.

```
usuario@ubuntu-20:~$ curl -4 icanhazip.com
46.222.217.118
```

Introducimos la ip local en nuestro navegador



5. Administración del proceso de Nginx

Ahora que su servidor web está listo, revisaremos algunos de los comandos básicos de administración.

Para detener su servidor web, escriba lo siguiente:

```
usuario@ubuntu-20:~$ sudo systemctl stop nginx
[sudo] contraseña para usuario:
usuario@ubuntu-20:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset:
   Active: inactive (dead) since Tue 2021-10-26 18:53:06 CEST; 19s ago
     Docs: man:nginx(8)
   Process: 5666 ExecStop=/sbin/start-stop-daemon --quiet --stop --retry QUIT/
   Main PID: 3518 (code=exited, status=0/SUCCESS)

oct 26 18:07:13 ubuntu-20 systemd[1]: Starting A high performance web server an
oct 26 18:07:13 ubuntu-20 systemd[1]: Started A high performance web server and
oct 26 18:53:06 ubuntu-20 systemd[1]: Stopping A high performance web server an
oct 26 18:53:06 ubuntu-20 systemd[1]: nginx.service: Succeeded.
oct 26 18:53:06 ubuntu-20 systemd[1]: Stopped A high performance web server and
lines 1-12/12 (END)
```

Para iniciar el servidor web cuando no esté activo, escriba lo siguiente:

```

usuario@ubuntu-20:~$ sudo systemctl start nginx
usuario@ubuntu-20:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2021-10-26 18:54:22 CEST; 3s ago
     Docs: man:nginx(8)
   Process: 5771 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_proce
   Process: 5772 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (c
 Main PID: 5773 (nginx)
    Tasks: 3 (limit: 2308)
   Memory: 6.6M
    CGroup: /system.slice/nginx.service
            └─5773 nginx: master process /usr/sbin/nginx -g daemon on; master_
              └─5774 nginx: worker process
                └─5775 nginx: worker process

oct 26 18:54:22 ubuntu-20 systemd[1]: Starting A high performance web server and
oct 26 18:54:22 ubuntu-20 systemd[1]: Started A high performance web server and
lines 1-16/16 (END)

```

Para detener y luego iniciar el servicio de nuevo, escriba lo siguiente:

```

usuario@ubuntu-20:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2021-10-26 18:55:32 CEST; 3s ago
     Docs: man:nginx(8)
   Process: 5895 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_proce
   Process: 5896 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (c
 Main PID: 5897 (nginx)
    Tasks: 3 (limit: 2308)
   Memory: 3.5M
    CGroup: /system.slice/nginx.service
            └─5897 nginx: master process /usr/sbin/nginx -g daemon on; master_
              └─5898 nginx: worker process
                └─5899 nginx: worker process

oct 26 18:55:32 ubuntu-20 systemd[1]: Starting A high performance web server and
oct 26 18:55:32 ubuntu-20 systemd[1]: Started A high performance web server and
lines 1-16/16 (END)

```

Si solo está realizando cambios en la configuración, Nginx a menudo puede volver a cargarse sin perder las conexiones. Para hacer esto, escriba lo siguiente:


```

usuario@ubuntu-20:~$ sudo systemctl reload nginx
usuario@ubuntu-20:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2021-10-26 18:55:32 CEST; 56s ago
     Docs: man:nginx(8)
   Process: 5895 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_proce
   Process: 5896 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (c
   Process: 6011 ExecReload=/usr/sbin/nginx -g daemon on; master_process on; -
   Main PID: 5897 (nginx)
    Tasks: 3 (limit: 2308)
   Memory: 5.0M
   CGroup: /system.slice/nginx.service
           └─5897 nginx: master process /usr/sbin/nginx -g daemon on; master_
             └─6012 nginx: worker process
               └─6013 nginx: worker process

oct 26 18:55:32 ubuntu-20 systemd[1]: Starting A high performance web server an
oct 26 18:55:32 ubuntu-20 systemd[1]: Started A high performance web server and
oct 26 18:56:24 ubuntu-20 systemd[1]: Reloading A high performance web server a
oct 26 18:56:24 ubuntu-20 systemd[1]: Reloaded A high performance web server an
lines 1-10/10 (END)

```

De forma predeterminada, Nginx está configurado para iniciarse automáticamente cuando lo haga el servidor. Si no es lo que quiere, deshabilite este comportamiento escribiendo lo siguiente:

```

usuario@ubuntu-20:~$ sudo systemctl disable nginx
Synchronizing state of nginx.service with SysV service script with /lib/systemd/
systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable nginx
Removed /etc/systemd/system/multi-user.target.wants/nginx.service.

```

Para volver a habilitar el servicio de modo que se cargue en el inicio, puede escribir lo siguiente:

```

usuario@ubuntu-20:~$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /lib/systemd/
systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /lib
/systemd/system/nginx.service.

```

6. Configuración de bloques del servidor

Ahora crearemos el directorio para your_domain como se muestra a continuación, usando el indicador -p para crear cualquier directorio principal necesario:

```

usuario@ubuntu-20:~$ sudo mkdir -p /var/www/your_domain/html

```


Luego asignamos la propiedad al directorio con la variable \$USER

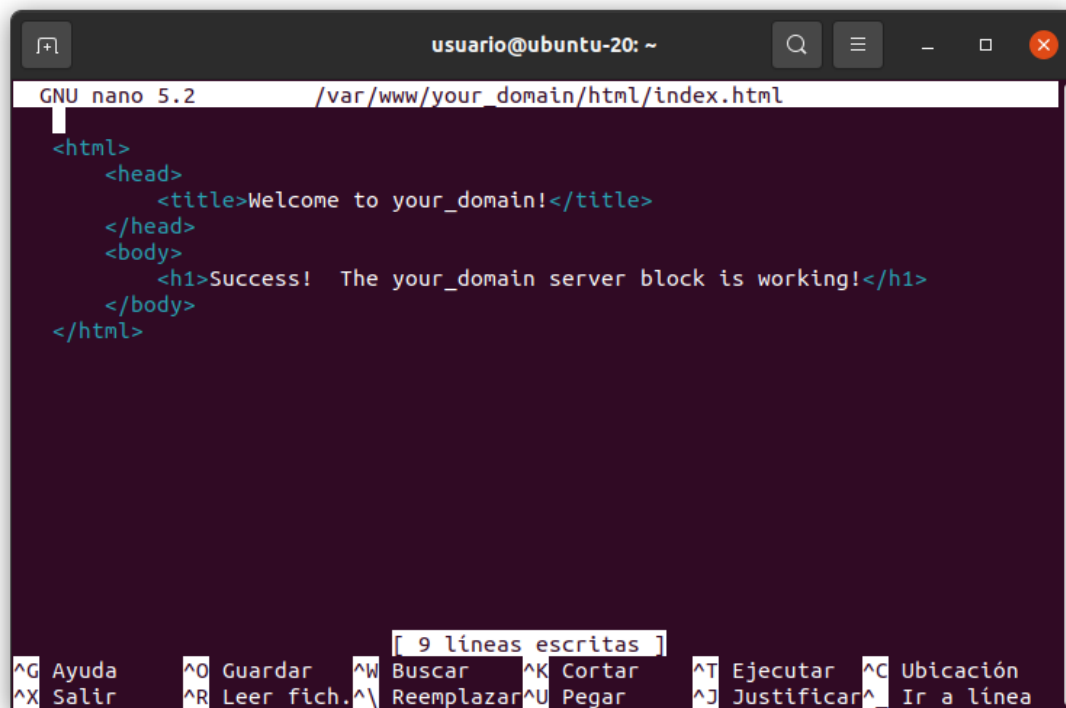
```
usuario@ubuntu-20:~$ sudo chown -R $USER:$USER /var/www/your_domain/html
```

Ahora daremos permisos de leer escribir y ejecutar archivos al siguiente directorio

```
usuario@ubuntu-20:~$ sudo chmod -R 755 /var/www/your_domain
```

A continuación creamos un index.html con cualquier editor con el siguiente texto.

```
usuario@ubuntu-20:~$ nano /var/www/your_domain/html/index.html
```



The screenshot shows a terminal window with the nano text editor open. The title bar indicates the user is 'usuario@ubuntu-20' and the current file is '/var/www/your_domain/html/index.html'. The editor shows the following HTML code:

```
<html>
  <head>
    <title>Welcome to your_domain!</title>
  </head>
  <body>
    <h1>Success! The your_domain server block is working!</h1>
  </body>
</html>
```

At the bottom of the editor, a status bar shows '[9 líneas escritas]' and a list of keyboard shortcuts: ^G Ayuda, ^O Guardar, ^W Buscar, ^K Cortar, ^T Ejecutar, ^C Ubicación, ^X Salir, ^R Leer fich., ^\ Reemplazar, ^U Pegar, ^J Justificar, ^_ Ir a línea.

Ahora

crearemos un nuevo archivo de configuración.

```
usuario@ubuntu-20:~$ sudo nano /etc/nginx/sites-available/your_domain
```

```

GNU nano 5.2 /etc/nginx/sites-available/your_domain

server {
    listen 84;
    listen [::]:84;

    root /var/www/your_domain/html;
    index index.html index.htm index.nginx-debian.html;

    server_name your_domain www.your_domain;

    location / {
        try_files $uri $uri/ =404;
    }
}

[ 14 líneas escritas ]
^G Ayuda  ^O Guardar  ^W Buscar  ^K Cortar  ^T Ejecutar  ^C Ubicación
^X Salir  ^R Leer fich. ^\ Reemplazar ^U Pegar  ^J Justificar ^_ Ir a línea

```

A continuación, habilitaremos el archivo creando un enlace entre él y el directorio sites-enabled, en el cual Nginx obtiene lecturas durante el inicio

```

usuario@ubuntu-20:~$ sudo ln -s /etc/nginx/sites-available/your_domain /etc/nginx/sites-enabled/

```

Para evitar un problema de memoria de depósito de hash que pueda surgir al agregar nombres de servidor, es necesario aplicar ajustes a un valor en el archivo `/etc/nginx/nginx.conf`. Abra el archivo y quite el `#` que se sitúa en `server_names_hash_bucket`.

```

GNU nano 5.2 /etc/nginx/nginx.conf

##
# Basic Settings
##

sendfile on;
tcp_nopush on;
types_hash_max_size 2048;
# server_tokens off;

server_names_hash_bucket_size 64;
# server_name_in_redirect off;

include /etc/nginx/mime.types;
default_type application/octet-stream;

##
# SSL Settings
##

[ 83 líneas escritas ]
^G Ayuda  ^O Guardar  ^W Buscar  ^K Cortar  ^T Ejecutar  ^C Ubicación
^X Salir  ^R Leer fich. ^\ Reemplazar ^U Pegar  ^J Justificar ^_ Ir a línea

```

```
usuario@ubuntu-20:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
usuario@ubuntu-20:~$ sudo systemctl restart nginx
usuario@ubuntu-20:~$
```

Ahora reiniciamos Nginx

Y por ultimo comprobamos que cargue y funcione la página.

A screenshot of a web browser window. The address bar shows the IP address 127.0.0.1:8084. The main content area displays the text "Success! The your_domain server block is working!" in a bold, black, sans-serif font.

Success! The your_domain server block is working!