

**Universidad Tecnológica de Panamá**  
**Facultad de Ingeniería de Sistemas Computacionales**

**By Gabriel Diaz**

**Resumen**

Una guía paso a paso para la instalación del servidor web Nginx en una distribución gnu-linux.

**Instalación de servidor NGINX en Linux (Ubuntu, Ubuntu Server o distribuciones basadas en Debian)**

1. Es recomendable logearse como usuario root.
2. El siguiente paso antes de realizar la instalación siempre es necesario realizar un update de los paquetes con el comando:

`$apt update`

O también con

`$sudo apt-get update`

3. Luego proceder con la instalación mediante el comando:

`$apt install nginx`

O también con

`$sudo apt-get install nginx`

4. Verificando el estado del servicio mediante los comandos para nginx:

- Encender el servicio:

`$systemctl start nginx.service`

- Parar el servicio:

`$systemctl stop nginx.service`

- Mostrar el estado del servicio:

```
$systemctl status nginx.service
```

- Recargar el servidor (se debe ejecutar siempre luego de realizar algún cambio de configuración)

```
$nginx -s reload
```

También

```
$systemctl reload nginx.service
```

- Verificar el archivo de configuración de arranque del servidor:

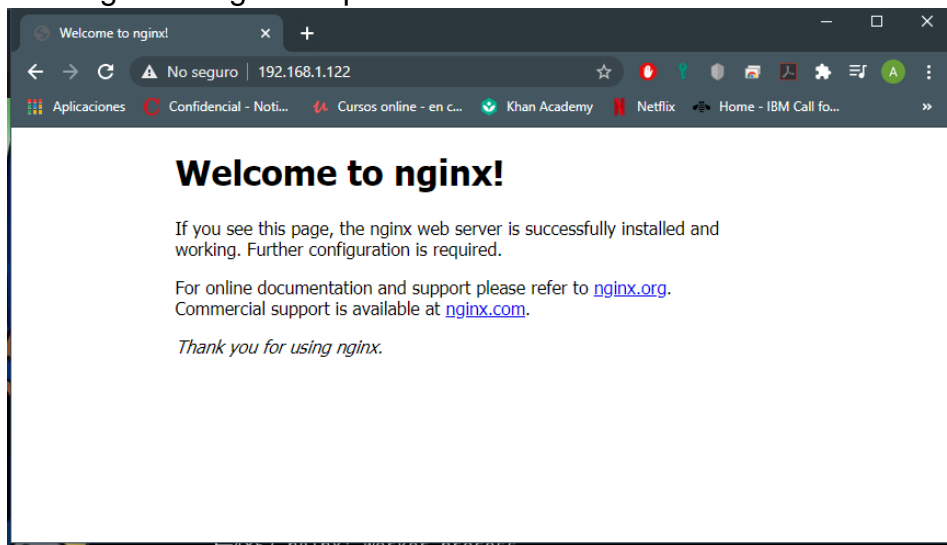
```
$ /usr/sbin/nginx -t
```

#o bien, indicando el fichero de configuración

```
$/usr/sbin/nginx -c /etc/nginx/nginx.conf -t
```

5. Realizar la verificación del servidor desde el navegador web accediendo a la dirección de IP del servidor o también escribiendo "http://localhost"

Se cargará la siguiente pantalla:



¿Como saber la IP del servidor?

Instalar el paquete net-tool

\$apt install net-tools

Luego ingresar el comando

\$ifconfig

Desplegara la información sobre las interfaces que existen en el servidor

```
root@gab-nginx-server:/home/gabox# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:d8:c1:c5:a7 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.122 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a00:27ff:fea1:f2b1 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:a1:f2:b1 txqueuelen 1000 (Ethernet)
    RX packets 59545 bytes 86722226 (86.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6560 bytes 545110 (545.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 138 bytes 11746 (11.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 138 bytes 11746 (11.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@gab-nginx-server:/home/gabox#
```

IP  
Servidor

## Opcional

### Creando un nuevo dominio web en NGINX

1. Dirigirse a la carpeta que contiene los directorios que almacenan los archivos de la pagina default, mediante el comando:

\$cd /var/www

Si realizamos un ls nos mostrara el archivo html de la página de bienvenida.

```
root@gab-nginx-server:/home/gabox# cd /var/www
root@gab-nginx-server:/var/www# ls
html
root@gab-nginx-server:/var/www#
```

2. Creamos un nuevo directorio donde guardaremos los archivos para la nueva página.

```
root@gab-nginx-server:/var/www# mkdir mipagina
root@gab-nginx-server:/var/www# ls
html  mipagina
root@gab-nginx-server:/var/www#
```

3. Cambiamos los permisos de usuario de la carpeta

```
root@gab-nginx-server:/var/www# chmod -R 755 mipagina/
```

4. Accedemos a la carpeta creada y luego creamos un archivo index.html que contendrá la estructura de nuestra nueva pagina

```
root@gab-nginx-server:/var/www# cd mipagina
root@gab-nginx-server:/var/www/mipagina# nano index.html_
```

En el archivo index.html escribimos el código html

A screenshot of the GNU nano 4.8 text editor. The title bar shows 'index.html'. The editor content consists of three lines: '<html>', '<p> Hello World </p>', and '</html>'. The bottom status bar displays various keyboard shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, M-U Undo, ^X Exit, ^R Read File, ^\ Replace, ^U Paste Text, ^T To Spell, and ^\_ Go To Line, M-E Redo. A small tooltip '[ Read 5 lines ]' is visible above the 'Cut Text' shortcut.

Al terminar de escribir guardamos con CTRL + O y para salir CTRL + X

5. Para habilitar la nueva pagina procederemos a cambiar al directorio de configuración de NGNIX mediante el comando

```
$cd /etc/nginx
```

En este directorio se encuentran las carpetas de configuración y el archivo nginx.conf de guarda la configuración global del servidor

```
root@gab-nginx-server:/var/www/mipagina# cd /etc/nginx/
root@gab-nginx-server:/etc/nginx# ls
conf.d      koi-utf    modules-available  proxy_params  sites-enabled  win-utf
fastcgi.conf  koi-win    modules-enabled     scgi_params   snippets
fastcgi_params mime.types nginx.conf          sites-available uwsgi_params
root@gab-nginx-server:/etc/nginx# _
```

6. Para habilitar nuestra nueva pagina solo necesitamos acceder a la carpeta sites-aviable, donde encontraremos un archivo con el nombre default.

```
root@gab-nginx-server:/etc/nginx# cd sites-available/
root@gab-nginx-server:/etc/nginx/sites-available# ls
default
```

7. Realizaremos una copia del archivo default para que contenga la configuración de nuestra nueva página.

```
root@gab-nginx-server:/etc/nginx/sites-available# cp default mipagina
root@gab-nginx-server:/etc/nginx/sites-available# ls
default  mipagina
root@gab-nginx-server:/etc/nginx/sites-available# _
```

8. Procedemos a editar el archivo copiado

```
root@gab-nginx-server:/etc/nginx/sites-available# nano mipagina
```

9. Se nos desplegaran muchas líneas de texto, de las cuales solo necesitaremos editar las siguientes:

- 1) Se configura el puerto 80 que corresponde al protocolo http
- 2) Opcional. Se omite la línea.
- 3) Se ubica el directorio que creamos para nuestra pagina en el paso 2.
- 4) Se estable un nombre de dominio para nuestra página.

```
GNU nano 4.8 mipagina Modified
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
    listen 80;
    #listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/mipagina;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name mipagina.com;

    location / {
```

Se guarda la configuración `ctrl + O` y cerramos con `ctrl + X`

10. Crear un enlace simbólico del archivo de configuración de nuestra página que se guardara en el directorio `sites-enabled`, con el siguiente comando

```
root@gab-nginx-server:/etc/nginx/sites-available# ln -s /etc/nginx/sites-available/mipagina /etc/nginx/sites-enabled/mipagina
root@gab-nginx-server:/etc/nginx/sites-available# _
```

11. Como ultima configuración se tiene que editar el archivo `host`

```
root@gab-nginx-server:/etc/nginx/sites-available# nano /etc/hosts_
```

Agregamos el nuevo dominio de nuestra página (ej: `example.com`)

```
GNU nano 4.8 /etc/hosts
127.0.0.1 localhost
127.0.0.1 mipagina.com
127.0.1.1 gab-nginx-server

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Guardamos y cerramos

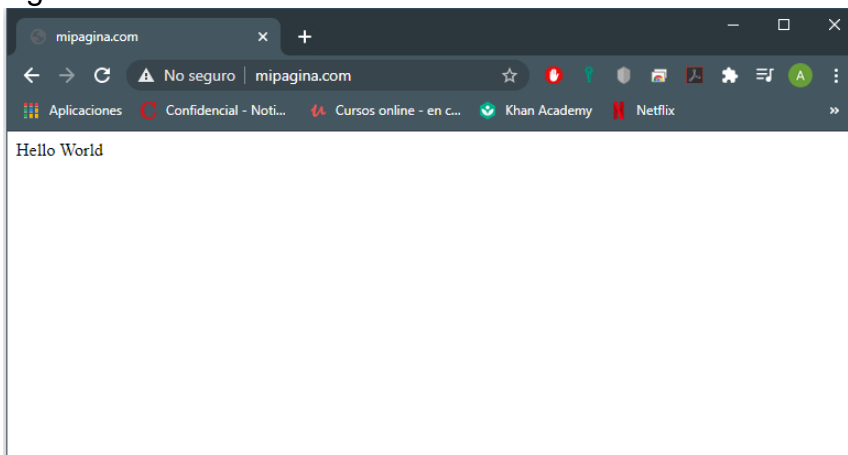
Y reiniciamos nuestro servidor mediante el comando

```
$systemctl restart nginx.service
```

O también

```
$nginx -s reload
```

12. Ingresar la dirección de nuestra página en el navegador web y debería mostrar lo siguiente



### Instalar dependencias y librerías de python:

```
$sudo apt install python3-pip python3-dev build-essential libssl-dev libffi-dev python3-setuptools
```

```
$sudo apt install python3-venv
```

Instal flask

