Certificado HTTPS a nuestros proyectos Web con Nginx

Cristhian Santa Cruz
hace 2 años

Antes de comenzar esta guía, se asume que contamos ya con un proyecto publicado utilizando **Nginx y Linux Ubuntu.**

Comencemos configurando primero nuestro sistema operativo, por defecto Ubuntu server o las distribuciones para servidores de Ubuntu, solamente aceptan peticiones **HTTP**, debido a que para aceptar peticiones **HTTPS** requieren de certificaciones, por esta razón habilitaremos solicitudes de tipo **HTTPS** o **SSL**.

Ejecutemos en la terminal del servidor:

```
sudo ufw status
```

Con esto veremos las opciones por defecto que dejo la instalación de **Nginex** en nuestro servidor:

```
Status: active
To Action From
-- ------ ----
22/tcp ALLOW Anywhere
22/tcp (v6) ALLOW Anywhere (v6)
```

El resultado debería ser parecido a lo de arriba, lo importante es que **status sea activo**, en caso no fuera así debemos ejecutar en la terminal:

```
sudo ufw enable
```

Luego volver a ejecutar **sudo ufw status** y veremos que nos indica que el status ahora es activo.

Ahora vamos a habilitar la peticiones **HTTPS** con **Nginx** y eliminar configuraciones redundantes.

```
sudo ufw allow 'Nginx Full'
```

```
sudo ufw delete allow 'Nginx HTTP'
```

Ahora si volvemos a ejecutar: **sudo ufw status** veremos algo como:

```
Status: active

To                         Action      From
--                         ------      ----
22                         ALLOW       Anywhere
443                        ALLOW       Anywhere
21/tcp                     ALLOW       Anywhere
80                         ALLOW       Anywhere
Nginx Full                 ALLOW       Anywhere
22 (v6)                    ALLOW       Anywhere (v6)
80 (v6)                    ALLOW       Anywhere (v6)
443 (v6)                   ALLOW       Anywhere (v6)
Nginx Full (v6)            ALLOW       Anywhere (v6)
21/tcp (v6)                ALLOW       Anywhere (v6)
```

Hasta aquí ya habilitamos peticiones **HTTPS** en nuestro servidor con **Nginx**. Ahora veamos como obtenemos el certificado **SSL**.

Obtener certificado SSL con cerbot

Para obtener nuestro certificado SSL utilizaremos una herramienta llamada **Cerbot**, que nos ayudara a obtener un certificado mediante Let's Encrypt de una forma muy sencilla, por tanto instalemos **Cerbot** en nuestro servidor:

```
sudo add-apt-repository ppa:certbot/certbot
```

Nos pedirá que presionemos ENTER y luego ejecutamos:

```
sudo apt update
```

Ahora instalamos Cerbot:

```
    sudo apt install python-certbot-nginx
```

Terminada la instalación, solo nos queda obtener los certificados, para ello será necesaria que contemos con un dominio o sub dominio asociado al proyecto que tenemos publicado con Nginx, para mi caso yo usare el sub dominio: **servicios.neunapp.com**

En la terminal ejecutamos:

```
    sudo certbot --nginx -d servicios.neunapp.com
```

Si tuviéramos más proyectos con más dominios en el servidor sería algo como:

```
    sudo certbot --nginx -d servicios.neunapp.com -d dominio1.neun
```

La primera vez que ejecutemos esto, pedirá que ingresemos un correo electrónico, de preferencia se debe poner el correo electrónico propietario de los dominios o sub dominios que estemos utilizando, una vez que **cerbot** confirme que los dominios son nuestros saltara en la terminal algo como:

```
Please choose whether or not to redirect HTTP traffic to HTTPS, re
-------------------------------------------------------------------
1: No redirect - Make no further changes to the webserver configur
2: Redirect - Make all requests redirect to secure HTTPS access. C
-------------------------------------------------------------------
Select the appropriate number [1-2] then [enter] (press 'c' to can
```

Ahí nos pregunta si dejamos que todas las peticiones http se redirijan a peticiones https,

como recomendación te sugiero elegir la opción 2, para que siempre aceptes ambas peticiones y redirijas siempre a https.

Escribimos 2 o 1 y luego pulsamos ENTER, con ello nos pintara un mensaje de confirmación en la terminal, similar a:

```
IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/example.com/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/example.com/privkey.pem
   Your cert will expire on 2018-07-23. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot a
   with the "certonly" option. To non-interactively renew *all* of
   your certificates, run "certbot renew"
 - Your account credentials have been saved in your Certbot
   configuration directory at /etc/letsencrypt. You should make a
   secure backup of this folder now. This configuration directory
   also contain certificates and private keys obtained by Certbot
   making regular backups of this folder is ideal.
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/don
   Donating to EFF:                    https://eff.org/donate-le
```

Ahora ya tienes los certificados **SSL** para los dominios que hayas especificado, ahora solo queda activar la renovación automática de estos certificados

```
    sudo certbot renew --dry-run
```

Bien, ahora si revisas tus archivos de configuración de **Nginx**, verás que se agregó código que tú no pusiste, es el código generado por **cerbot** para los certificados SSL o para acceso **HTTPS**.

Recomendación:

Si tienes varios sub dominios tienes que crear un certificado para cada uno de ellos, ya que **cerbot** no te dejara utilizar el mismo certificado para diferentes archivos de configuración Nginx.

Recuerda que en el canal de youtube hay un video realizado esta guía en un servidor con Digital Ocean. (https://m.do.co/c/d464e42336a5)

¿La respuesta fue util? Si No