

## Configuración Servidor Web Centos 7

En esta práctica se va a configurar un servidor Web en un Servidor Centos 7. La organización de esta práctica se resume en los siguientes pasos:

1. Pasos Previos
2. Direccionamiento estático
3. Instalar el servicio SSH
4. Nombre de la máquina Servidor
5. Instalación servidor web nginx
6. Instale modo gráfico máquina cliente
7. Configuración Server-Blocks
8. Configuración de acceso por puerto distinto al 80. Por ejemplo, puerto 8080
9. Configuración segura. Puerto 443 (https)
10. Comprobación del funcionamiento
11. Consulta de Registros

### 1. Pasos Previos

Proceda a usar la instantánea del servidor y cliente en Centros básica, antes de la instalación de cualquier servicio.

1. Indique direccionamiento estático. Para las diferentes tarjetas de red tal como procedió en la práctica de configuración del Servicio DNS.
2. Abra el servidor Windows 2012 como servidor DNS. Coloque dicha IP como DNS de la máquina web de Centos.
3. Nombre a la máquina como wwwC7.grupoXX.net

Verifique que la tabla de encaminamiento es correcta.

### 2. Direccionamiento estático

<b>NIC1: VLAN Producción</b>	Dirección IP: 10.0.99.210 Máscara de red: 255.255.255.0 Puerta de Enlace: 10.0.99.1
<b>NIC2: VLAN GESTIÓN</b>	Dirección IP: 10.0.99.210

### 3. Instalar el servicio SSH

Instale servicio SSH y configure para que sólo se pueda acceder a través de la IP correspondiente a la VLAN de Gestión del Servidor Web.

Conéctese mediante Putty para la configuración del servidor.

#### 4. Nombre de la máquina Servidor

Nombre a la máquina como wwwC7.grupoXX.net

Reinicie la máquina para que se guarde la configuración realizada hasta el momento.

#### 5. Instalación servidor web nginx

Vamos a proceder a la instalación del servidor web nginx

Para habilitar el repositorio “yum” oficial de nginx para RHEL/CentOS 7, crea un fichero /etc/yum.repos.d/nginx.repo con el siguiente contenido.

```
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/mainline/centos/7/$basearch/
gpgcheck=0
enabled=1
```

Instalamos nginx:

```
[root@wwwc7 ~]# yum install nginx
```

De esta forma queda instalado el servicio web. Iniciamos el servicio:

```
[root@wwwc7 ~]# systemctl start nginx.service
```

Si tenemos instalado firewall

```
firewall-cmd --permanent --zone=public --add-service=http
firewall-cmd --permanent --zone=public --add-service=https
firewall-cmd --permanent --zone=public --add-ports=80/tcp
firewall-cmd --permanent --zone=public --add-ports=8080/tcp
firewall-cmd --permanent --zone=public --add-ports=443/tcp
```

```
firewall-cmd --reload
```

Desde el PC del aula, abra el navegador y acceda al servidor web: <http://10.xx.100.210>

Para que nginx empiece cada vez que reiniciamos la máquina.

```
systemctl enable nginx
```

#### 6. Instale modo gráfico máquina cliente

Para poder ejecutar un navegador en un cliente Centos, hemos de instalar modo gráfico, para ello:

```
yum groupinstall "GNOME Desktop" "Graphical Administration Tools"
yum groupinstall "Server with GUI"
```

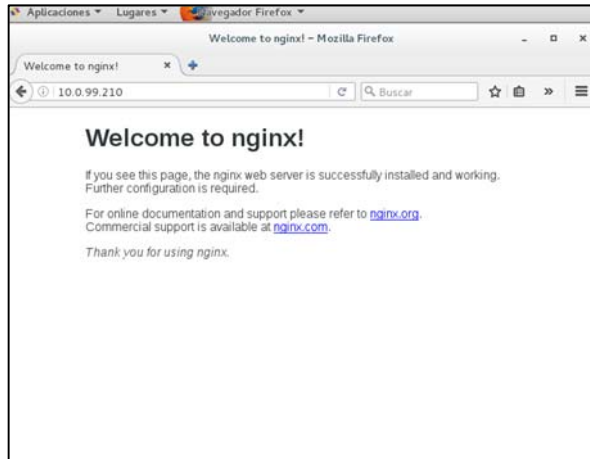
Por último habilitamos GUI sobre el sistema de arranque:

```
In -sf /lib/systemd/system/runlevel5.target /etc/systemd/system/default.target
```

Reiniciamos la máquina cliente.

Abrimos el navegador Aplicaciones → Internet → Firefox

Coloque como URL <http://10.XX.99.210>

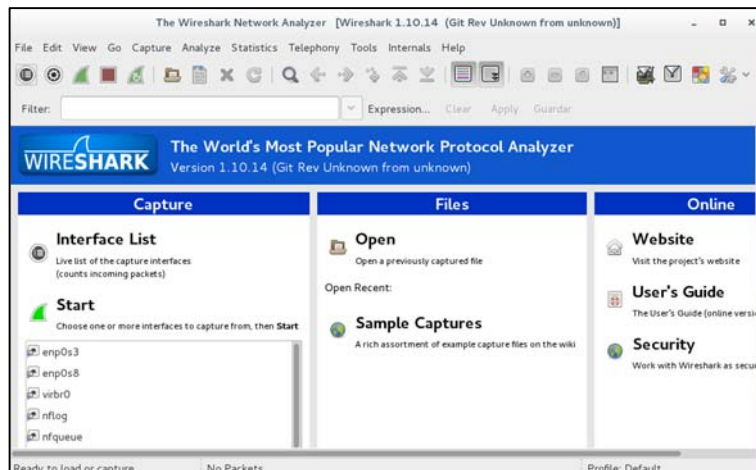


Con un detector podemos comprobar cómo se efectúan las comunicaciones y cómo funciona el protocolo. Usaremos la aplicación Wireshark:

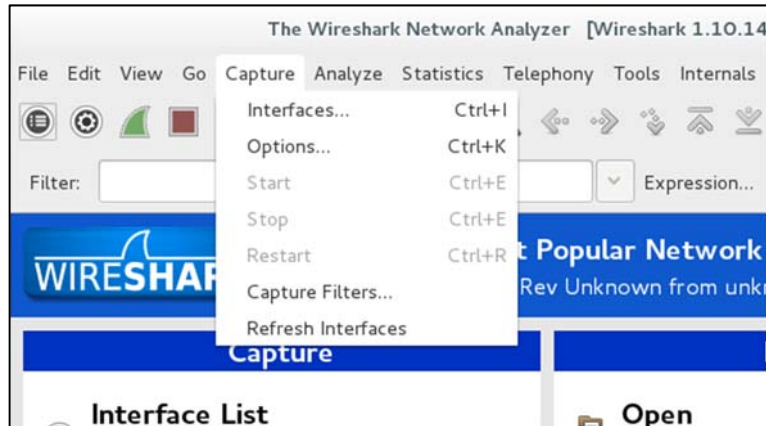
```
yum install wireshark
```

Ponemos en marcha la aplicación desde Aplicaciones → Internet → Wireshark (como root).

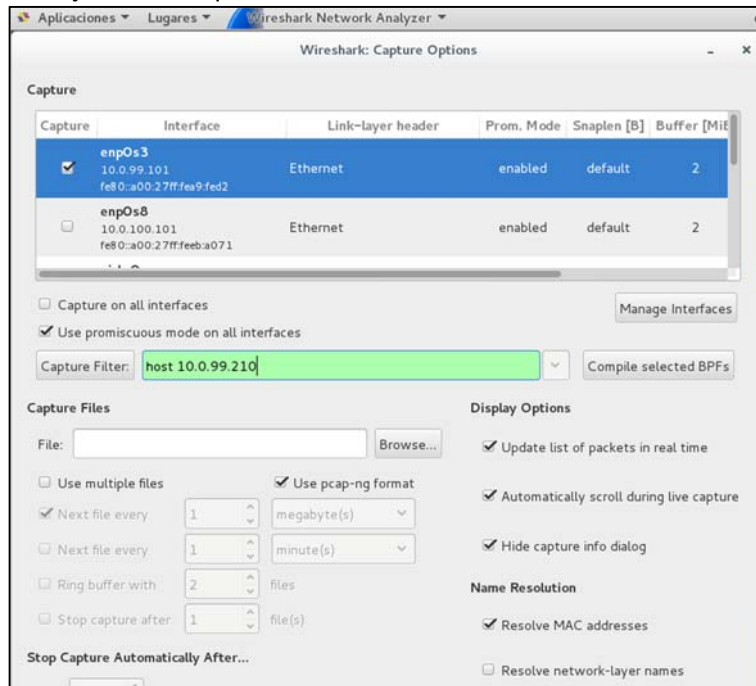
Tras pulsar en el icono de Wireshark aparece la siguiente pantalla, que es la que la aplicación usará como detector.



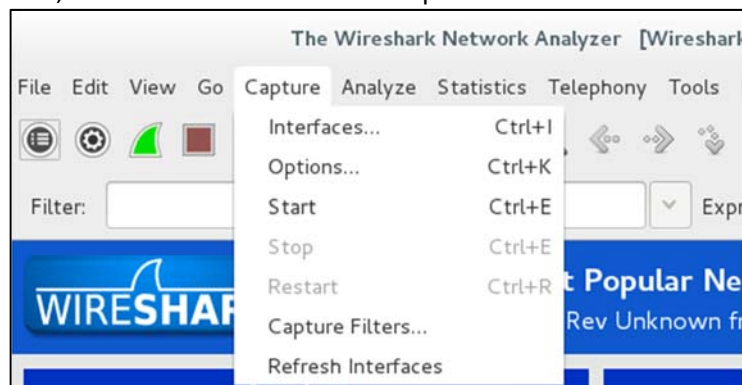
Ponemos en marcha el detector pulsando en Capture → Options



A continuación aparece una pantalla, donde hemos de seleccionar la interface, clicamos en enps03, y como "Capture Filter" indicamos que sólo queremos que capture tráfico relacionado con el servidor. Colocamos la orden host 10.XX.99.210 (dirección IP del servidor). En la figura siguiente queda reflejado como quedaría.

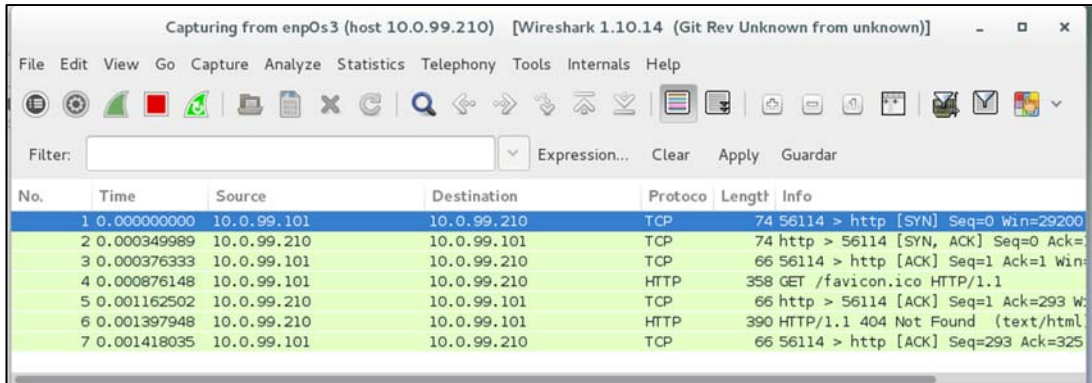


Cerramos la pantalla, clic en X. Hacemos click en Capture → Start



Abrimos de nuevo el navegador, con la URL <http://10.XX.99.210>.

A continuación nos aparece la captura de los paquetes. En la siguiente figura se puede observar cómo funciona el protocolo http con las comunicaciones sin cifrar.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.99.101	10.0.99.210	TCP	74	56114 > http [SYN] Seq=0 Win=29200
2	0.000349989	10.0.99.210	10.0.99.101	TCP	74	http > 56114 [SYN, ACK] Seq=0 Ack=
3	0.000376333	10.0.99.101	10.0.99.210	TCP	66	56114 > http [ACK] Seq=1 Ack=1 Win=
4	0.000876148	10.0.99.101	10.0.99.210	HTTP	358	GET /favicon.ico HTTP/1.1
5	0.001162502	10.0.99.210	10.0.99.101	TCP	66	http > 56114 [ACK] Seq=1 Ack=293 W=
6	0.001397948	10.0.99.210	10.0.99.101	HTTP	390	HTTP/1.1 404 Not Found (text/html
7	0.001418035	10.0.99.101	10.0.99.210	TCP	66	56114 > http [ACK] Seq=293 Ack=325

## 7. Configuración Server-Blocks

Desactive selinux:

```
[root@wwwC7 /]# vi /etc/selinux/config
```

Coloque la directiva en disabled → SELINUX=disabled

Reinicie la máquina wwwc7.

### ▪ Resolución nombres (Servidor DNS-Windows)

A continuación, vamos a crear los siguientes dominios:

1. al1.grupo0.net (máquina [www.al1.grupo0.net](http://www.al1.grupo0.net))
2. al2.grupo0.net (máquina [www.al2.grupo0.net](http://www.al2.grupo0.net))

Para ello nos vamos al servidor DNS de Windows y procedemos a la configuración de la resolución de nombres para cada máquina.

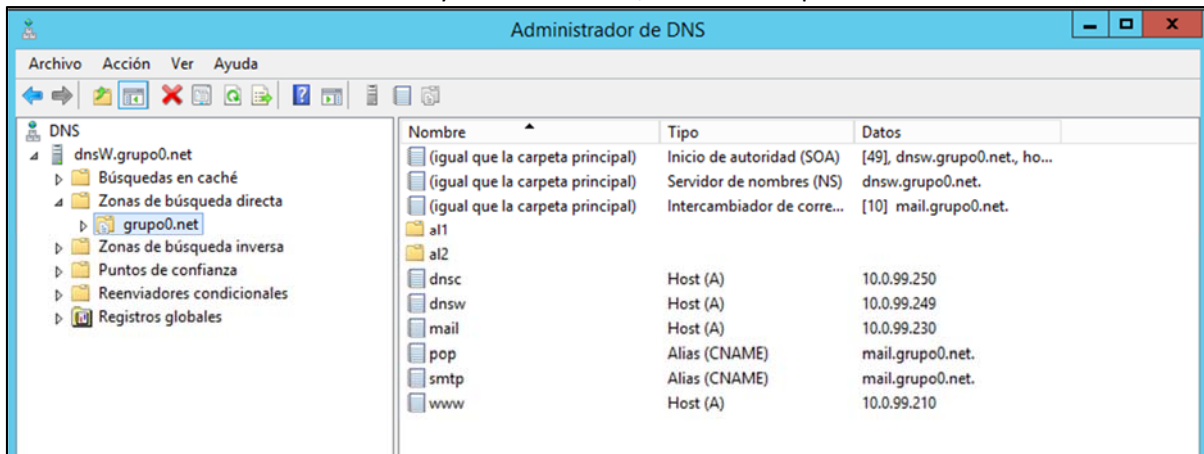
Abrir Administrador DNS → Zona de búsqueda directa → grupo0.net → clic botón derecho → Alias nuevo (CNAME) → Completamos los apartados para al1.grupo0.net.

Lo repetimos para crear ahora el alias [www.al1.grupo0.net](http://www.al1.grupo0.net)

DNS → Zona de búsqueda directa → grupo0.net → clic botón derecho → Alias nuevo (CNAME) → Completamos los apartados para [www.al1.grupo0.net](http://www.al1.grupo0.net).

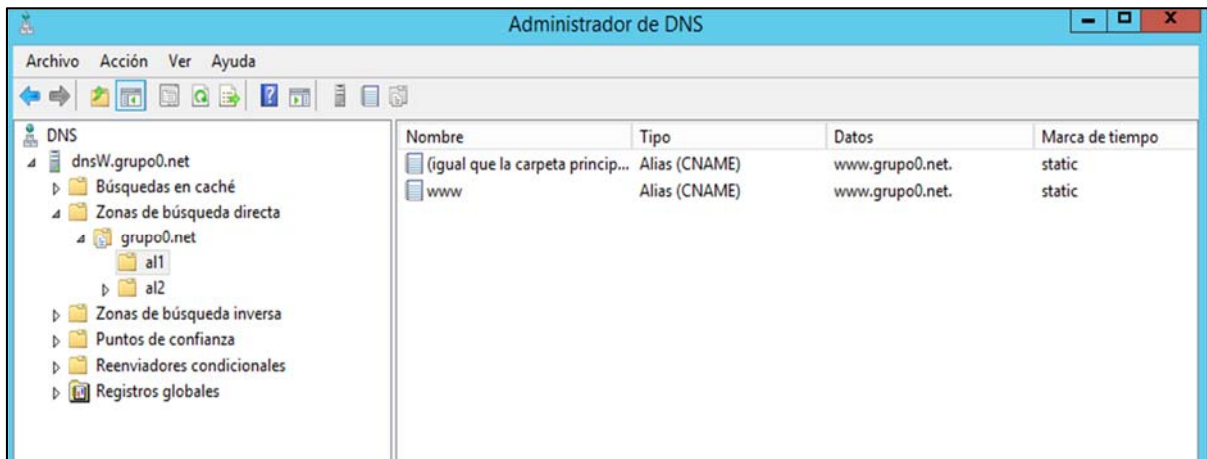
Repetir para el subdominio al2.grupo.net.

Tras cerrar el Administrador de DNS y volverlo a abrir, nos deberá quedar:



Nombre	Tipo	Datos
(igual que la carpeta principal)	Inicio de autoridad (SOA)	[49] dnsw.grupo0.net, ho...
(igual que la carpeta principal)	Servidor de nombres (NS)	dnsw.grupo0.net.
(igual que la carpeta principal)	Intercambiador de corre...	[10] mail.grupo0.net.
al1		
al2		
dnsc	Host (A)	10.0.99.250
dnsw	Host (A)	10.0.99.249
mail	Host (A)	10.0.99.230
pop	Alias (CNAME)	mail.grupo0.net.
smtp	Alias (CNAME)	mail.grupo0.net.
www	Host (A)	10.0.99.210

Si pinchamos en la carpeta al1, veremos:



Reinicie el servicio DNS. (Herramientas Administrativas->Servicios->Servidor DNS)

#### ▪ Creación Server-Blocks para cada dominio

Con ello ya hemos creado los DNS de nuestros dominios o subdominios apuntando a la misma máquina web. El concepto servidor de bloques consiste en que el servidor web mostrará webs distintas según la url de origen o el puerto.

Para proceder adecuadamente abrimos el fichero,

```
[root@wwwC7 /]# vi /etc/nginx/nginx.conf
```

```
paubernabeu — root@wwwC7:/ — ssh root@10.0.100.210 — 116x44
# For more information on configuration, see:
# * Official English Documentation: http://nginx.org/en/docs/
# * Official Russian Documentation: http://nginx.org/ru/docs/

user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;

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

    # Load modular configuration files from the /etc/nginx/conf.d directory.
    # See http://nginx.org/en/docs/nginx_core_module.html include
    # for more information.
    include /etc/nginx/conf.d/*.conf;

    server {
        listen 80 default_server;
        listen [::]:80 default_server;
        server_name _;
        root /usr/share/nginx/html;

        # Load configuration files for the default server block.
        include /etc/nginx/default.d/*.conf;

    }
}

-- INSERT --
```

En la primera línea no comentada tenemos `user=nginx`. Observamos que en Centos 7, el usuario es `nginx` (en centos 6 por ejemplo es `www-data`). Este usuario es importante saberlo para dar los permisos adecuados a las carpetas donde se colgará el correspondiente sitio web para cada DOMINIO.

Más abajo, tenemos `include /etc/nginx/conf.d/*.conf`. Este es el lugar donde podemos dejar los ficheros de configuración de cada DOMINIO, si bien podemos dejarlo en el sitio queramos, pero el path de donde estén deberá añadirse con `include`.

De todos modos, lo más habitual es crear las carpetas `sites-available` y `sites-enabled` (este segundo con enlaces simbólicos hacia `sites-enabled`).

### **Primer paso:**

Creamos los directorios:

- `mkdir /etc/nginx/sites-available`
- `mkdir /etc/nginx/sites-enabled`

Añadiremos en `nginx.conf` la línea:

- `include /etc/nginx/sites-enabled/*.conf`

```
include /etc/nginx/conf.d/*.conf;
include /etc/nginx/sites-enabled/*.conf;

server{
    listen      80 default_server;
    listen      [::]:80 default_server;

    server_name _;
    root        /usr/share/nginx/html;
```

A continuación, muestra con la directiva `server{ }` el servidor por defecto. Y el lugar donde tiene la página web por defecto (fichero `index.html`). En este caso lo expresa con la directiva `root`. Las webs están pues en `/usr/share/nginx/html`.

Nosotros vamos a configurar las diferentes webs virtuales por debajo de `/usr/share/nginx`. Podría estar en otro lugar y aquí es, para cada caso, donde deberíamos indicar dónde guarda el sitio web de cada DOMINIO.

### ▪ **Configuración sitio `al1.grupo0.net` en sites available:**

```
cd /etc/nginx/sites-available/
vi al1_grupo0.net.conf
```

Como mínimo, le indicamos que escuche por el puerto 80, que el nombre del sitio web es `al1.grupo0.net` y su alias [www.al1.grupo0.net](http://www.al1.grupo0.net) (`server_name`) y donde está el `index.html` de ese sitio web (`root`).

```
server{
```





listen 80;

```
server_name al1.grupo0.net www.al1.grupo0.net;
```

```
location /{
    root /usr/share/nginx/al1.grupo0.net/html;
    index index.html index.htm index.php;
}
}
```

Creamos enlace simbólico en sites-enabled:

```
ln -s /etc/nginx/sites-available/al1.grupo0.net.conf /etc/nginx/sites-enabled/
```

- **Creación de la carpeta /usr/share/nginx/al1.grupo0.net/html con los permisos adecuados**  
(usa sudo si el usuario no es root)

```
mkdir -p /usr/share/nginx/al1.grupo0.net/html
```

```
chown -R nginx:nginx /usr/share/nginx/al1.grupo0.net
```

```
chown -R nginx:nginx /usr/share/nginx/al1.grupo0.net/html/
```

```
chmod -R 755 /usr/share/nginx/al1.grupo0.net
```

```
chmod -R 755 /usr/share/nginx/al1.grupo0.net/html/
```

Repite el proceso para el dominio al2.grupo0.net

- **Crear contenido para el sitio web**

```
vi /usr/share/nginx/al1.grupo0.net/html/index.html
```

```
<!DOCTYPE html>
<html lang=es>
  <head>
    <meta charset="UTF-8">
    <title>al1.grupo0.net</title>
  </head>
  <body>
    <h1>En el sitio al1.grupo0.net</h1>
    <p>Bienvenido a este sitio web (Virtual)</p>
  </body>
</html>
~
~
```

Repita el proceso para el dominio al2.grupo0.net



Reiniciamos el servicio nginx:

```
systemctl restart nginx
```

### **Comprobación funcionamiento:**

Vamos a nuestra máquina cliente, <http://www.al1.grupo0.net>



Debe funcionar también para <http://al1.grupo0.net>



Para el sitio al2 ([www.al2.grupo0.net](http://www.al2.grupo0.net))



Para <http://al2.grupo0.net>



Para el sitio [www.grupo0.net](http://www.grupo0.net), va a la página nginx por defecto



## 8. Configuración de acceso por puerto distinto al 80. Por ejemplo, puerto 8080

Creamos el fichero:

```
/etc/nginx/sites-available/grupo0.net.8080.conf
```

```
server{
    listen 8080 default_server;
    server_name grupo0.net www.grupo0.net;

    location /{
        root /usr/share/nginx/grupo0.net.8080/html;
        index index.html index.htm index.php;
    }
}
```

Creamos enlace simbólico:

```
ln -s /etc/nginx/sites-available/grupo0.net.8080.conf /etc/nginx/sites-enabled/
```

Creamos el directorio `/usr/share/nginx/grupo0.net.8080/html`. Le damos propietario al usuario `nginx`. Le damos permiso `755`.

Creamos el fichero `index.html`

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF8">
    <title>Por el puerto 8080</title>
</head>
<body>
<h1>Conexión por defecto nginx por el puerto 8080</h1>
<p>Server nginx over Centos 7.0</p>
</body>

<html>
```



Comprobamos que para cualquier <http://mydomain:8080> se nos abre la página web por defecto.

Queremos que cada dominio pueda tener acceso a su página específica, en este caso vamos a hacer que apunte al mismo sitio web. El sitio para cada dominio.

Queremos que [www.al1.grupo0.net:8080](http://www.al1.grupo0.net:8080) y el sitio [al1.grupo0.net:8080](http://al1.grupo0.net:8080) apunten al mismo sitio que [www.al1.grupo0.net](http://www.al1.grupo0.net) y no al sitio por defecto del puerto 8080.

Para ello:

```
vi /etc/nginx/sites-available/al1.grupo0.net.conf
```

```
server{
    listen 80;
    listen 8080;
    server_name al1.grupo0.net www.al1.grupo0.net;

    root /usr/share/nginx/al1.grupo0.net/html;
}
```

Añadimos escuche también por el puerto 8080, si quisiéramos un sitio diferente, nueva etiqueta `server{}` con un `root` apuntando a una nueva carpeta.

Repetimos lo mismo para el dominio [al2.grupo0.net](http://al2.grupo0.net).

## 9. Configuración segura. Puerto 443 (https)

Para activar el servidor web seguro con el protocolo HTTPS tenemos que configurar el servidor para que escuche por el puerto TCP 443. Dejando el puerto 80 para el protocolo http.

Antes de nada, vamos a crear los certificados para los sitios [www.al1.grupo0.net](http://www.al1.grupo0.net) y [www.al2.grupo0.net](http://www.al2.grupo0.net).

Dominio [al1.grupo0.net](http://al1.grupo0.net)

Creamos el directorio ssl dentro de nginx

```
mkdir -p /etc/nginx/ssl/al1
cd /etc/nginx/ssl/al1
```

Generamos el certificado que se usará en las conexiones HTTPS a través del protocolo SSL.



**openssl genrsa -des3 -out ssl.al1.key 2048**

Generating RSA private key, 2048 bit long modulus

.....+++

.....+++

e is 65537 (0x10001)

Enter pass phrase for ssl.al1.key: **XXXX**

Verifying - Enter pass phrase for ssl.al1.key: **XXXX**

**Realizamos la solicitud de firma:**

openssl req -new -key ssl.al1.key -out ssl.al1.csr

Enter pass phrase for ssl.al1.key:

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [XX]:es

State or Province Name (full name) []:Alacant

Locality Name (eg, city) [Default City]:Alcoi

Organization Name (eg, company) [Default Company Ltd]:EPSA

Organizational Unit Name (eg, section) []:DCOM

Common Name (eg, your name or your server's hostname) []:www.al1.grupo0.net ←colocar el dominio con www

Email Address []:admin@al1.grupo0.net

Please enter the following 'extra' attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

Una vez que tenemos el fichero ssl.al1.csr, que es la petición de certificado, hemos de enviarlo a una entidad certificadora. Esta entidad certificadora será la encargada de firmarla y de certificar la validez. Con el fin de simplificar este paso, puedo con el servidor hacer una autofirma del fichero, obteniendo así el fichero firmado ssl.al1.crt.

openssl x509 -req -days 365 -in ssl.al1.csr -signkey ssl.al1.key -out ssl.al1.crt

Signature ok

subject=/C=es/ST=Alacant/L=Alcoi/O=EPSA/OU=DCOM/CN=www.al1.grupo0.net/emailAddress=admin@al1.grupo0.net



Getting Private key

Enter pass phrase for ssl.al1.key:XXXX

Para evitar ahora que cada vez reiniciemos el servicio nginx nos pida “pass phrase”.

Creamos una copia de .key como .original

```
cp -v ssl.al1.{key,original}
```

```
«ssl.al1.key» -> «ssl.al1.original»
```

```
openssl rsa -in ssl.al1.original -out ssl.al1.key
```

```
Enter pass phrase for ssl.al1.original:
```

```
writing RSA key XXXX
```

Finalmente borramos ssl.al1.original

```
rm ssl.al1.original
```

Por seguridad cambiamos los permisos de acceso al fichero únicamente para el usuario de la manera siguiente:

```
chmod 400 ssl.al1.key
```

Abrimos `/sites_available/al1.grupo0.net.conf`, para crear el sitio seguro para [www.al1.grupo0.net](http://www.al1.grupo0.net) (mismo nombre que el **common name** del certificado creado). Con las etiquetas `ssl_certificate` y `ssl_certificate_key`. Le indicamos donde está la clave privada y la firmada.

Añadimos las líneas:

```
server{  
    listen 443 ssl;  
    server_name www.al1.grupo0.net;  
  
    location /{  
        root /usr/share/nginx/al1.grupo0.net/html;  
        index index.htm index.html index.php;  
    }  
  
    ssl_certificate /etc/nginx/ssl/al1/ssl.al1.crt;  
    ssl_certificate_key /etc/nginx/ssl/al1/ssl.al1.key;  
  
}
```

El certificado sólo es válido para la URL [www.al1.grupo0.net](http://www.al1.grupo0.net). Pero no para la URL `al1.grupo0.net`. Para que sea válido añadimos las siguientes líneas de código en `al1.grupo0.net.conf`

```
server{  
    listen 443;  
    server_name al1.grupo0.net;
```



```
return 301 https://www.$server_name$request_uri;  
}
```

De tal forma, que el navegador reenviará la página para [www.al1.grupo0.net](http://www.al1.grupo0.net)

## 10. Comprobación del funcionamiento

Abrimos el navegador en la máquina cliente, e introducimos la dirección del servidor <https://www.al1.grupo0.net>. Una vez, introducida la dirección, el servidor nos indica que se trata de un servidor HTTPS y que la conexión no es segura (al estar configurado de manera incorrecta, es autofirmado).

Para (Mozilla FireFox):



Clicamos en Avanzado, y nos informa es autofirmado (self-signed)



Confiaremos en él y clicamos en Añadir Excepción:



Al clicar en VER, podemos ver el certificado:



Confirmamos excepción de seguridad y la página es cargada. Recordad hemos dicho va al mismo sitio que con http.



Comprobemos el funcionamiento para <https://al1.grupo0.net>

Nos aparecerán las misma confirmaciones caso anterior, y tras validar, podéis observar apunta a <https://www.al1.grupo0.net>.



Repita el proceso para el dominio al2.grupo0.net

## 11. Consulta de Registros

Con el fin de tener el máximo control de los servidores deberemos saber dónde se encuentran los ficheros de registros o logs. Cualquier problema que tenga el servidor web se almacena en estos ficheros.

Dichos ficheros se encuentran en `var/log/nginx` (tal como nos lo indica el fichero de configuración `nginx.conf`).

Para ver su contenido:

### **tail -f /var/log/nginx/access.log**

```
10.0.99.101 - - [02/Oct/2016:20:27:26 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:28:31 +0200] "GET / HTTP/1.1" 200 216 "-" "Mozilla/5.0 (X11;
Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:28:31 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:28:32 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:28:42 +0200] "GET / HTTP/1.1" 301 185 "-" "Mozilla/5.0 (X11;
Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:28:42 +0200] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11;
Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:31:37 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:33:55 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:35:27 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
10.0.99.101 - - [02/Oct/2016:20:35:27 +0200] "GET /favicon.ico HTTP/1.1" 404 169 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:45.0) Gecko/20100101 Firefox/45.0" "-"
```

### **tail -f /var/log/nginx/error.log**

```
2016/10/02      20:26:57      [error]      4676#4676:      *17      open()
"/usr/share/nginx/all.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:
10.0.99.101, server: all.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:
"www.all.grupo0.net"
2016/10/02      20:26:57      [error]      4676#4676:      *17      open()
"/usr/share/nginx/all.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:
10.0.99.101, server: all.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:
"www.all.grupo0.net"
2016/10/02      20:27:01      [error]      4676#4676:      *18      open()
"/usr/share/nginx/all.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:
10.0.99.101, server: all.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:
"www.all.grupo0.net:8080"
2016/10/02      20:27:26      [error]      4676#4676:      *21      open()
"/usr/share/nginx/all.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:
```



10.0.99.101, server: www.al1.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host: "www.al1.grupo0.net"

2016/10/02 20:28:31 [error] 4676#4676: \*25 open()  
"/usr/share/nginx/al1.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: www.al1.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al1.grupo0.net"

2016/10/02 20:28:32 [error] 4676#4676: \*25 open()  
"/usr/share/nginx/al1.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: www.al1.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al1.grupo0.net"

2016/10/02 20:31:37 [error] 4676#4676: \*28 open()  
"/usr/share/nginx/al1.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: al1.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al1.grupo0.net"

2016/10/02 20:33:55 [error] 4676#4676: \*30 open()  
"/usr/share/nginx/al1.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: al1.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al1.grupo0.net"

2016/10/02 20:35:27 [error] 4676#4676: \*32 open()  
"/usr/share/nginx/al2.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: www.al2.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al2.grupo0.net"

2016/10/02 20:35:27 [error] 4676#4676: \*32 open()  
"/usr/share/nginx/al2.grupo0.net/html/favicon.ico" failed (2: No such file or directory), client:  
10.0.99.101, server: www.al2.grupo0.net, request: "GET /favicon.ico HTTP/1.1", host:  
"www.al2.grupo0.net"