



# DIARIO DE APRENDIZAJE

**Manuel Nieto Benítez**

Escuela Virgen de Guadalupe

## ÍNDICE

---

**01**

Servicio DNS

**06**

Servicio DNS

**02**

Servidor Web Apache

**07**

Servidor Web Apache

**03**

Directivas de Bloque

**08**

Directivas de Bloque

**04**

Webalizer

**09**

Webalizer

**05**

Resumen



# DESCRIPCIÓN DEL SERVICIO

## SERVICIO DNS

La tarea implica la instalación y configuración del servicio DNS (Sistema de Nombres de Dominio) en un servidor Ubuntu 16.04 para el dominio 2dawxx.net, donde xx representa el número de puesto del estudiante. El DNS desempeña la función de traducir nombres de dominio a direcciones IP y viceversa, lo que simplifica la navegación en internet y la comunicación entre dispositivos en una red.

## COMANDOS

- `hostnamectl set-hostname "nombre nuevo del hostname"` - Cambia el nombre del hostname de la máquina.
- `sudo apt-get update` y `sudo apt-get upgrade` - Actualiza el repositorio y realiza actualizaciones del sistema.
- `netstat -natp | grep named` y `netstat -naup | grep named` - Verifica los puertos TCP y UDP activos del servicio DNS.
- `systemctl status bind9` - Comprueba el estado del servicio DNS.
- `named-checkconf /etc/bind/named.conf.options` y `named-checkconf /etc/bind/named.conf.local` - Verifica errores de sintaxis en los archivos de configuración del DNS.
- `named-checkzone deap27.net /etc/bind/db.2daw27.net` y `named-checkzone 2daw27.net /etc/bind/db.10.2.13` - Verifica errores en las zonas directas e inversas.
- `ip a` - Muestra los interfaces de red levantados.
- `nano /etc/network/interfaces` - Accede al archivo de configuración de red para editar la configuración IP estática.
- `ping`, `nslookup` y `dig` - Herramientas para verificar la resolución de nombres y direcciones IP.

# CONFIGURACIÓN ADICIONAL

Configuración del adaptador de red como BRIDGE.

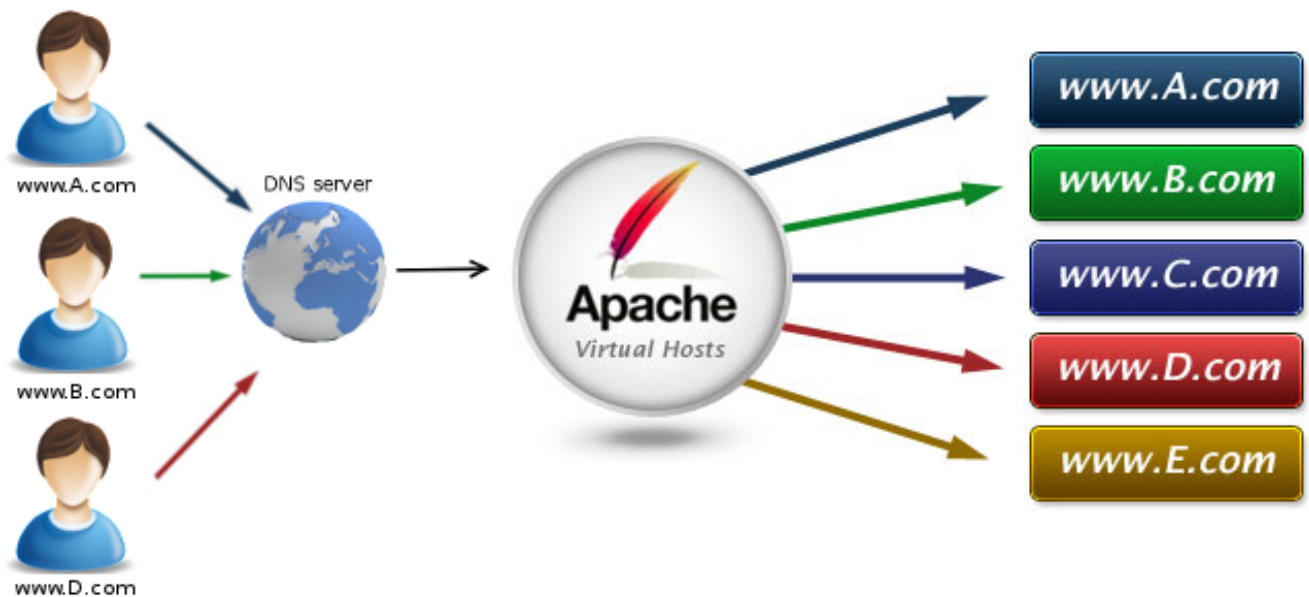
Configuración de la IP estática en el archivo `/etc/network/interfaces`.

Activación del reenviador y configuración de las zonas directas e inversas en los archivos de configuración del DNS.

Teclado en español:

```
root@deap12:~# sudo dpkg-reconfigure keyboard-configuration_
```

Tree es una herramienta útil para visualizar la estructura de directorios en forma de árbol: `sudo apt-get install tree`





# DESCRIPCIÓN DEL SERVICIO

---



## SERVICIO WEB APACHE

En esta actividad, se aborda la configuración del servidor web Apache en un entorno virtualizado utilizando Ubuntu Server 16.04. Se instala y configura la herramienta Webalizer para analizar los registros de acceso del servidor web, y se ajustan las directivas de rendimiento en el archivo de configuración principal de Apache.

Una vez instalado Webalizer, se pueden generar informes detallados sobre el tráfico y la actividad en el servidor, lo que facilita el monitoreo y la toma de decisiones informadas. Además, ajustar las directivas de rendimiento en Apache permitirá optimizar el uso de recursos y mejorar la velocidad de respuesta del servidor. Con estas configuraciones adecuadas, se logrará tener un servidor web eficiente y seguro en el entorno virtualizado con Ubuntu Server 16.04.

## COMANDOS

---

- Instalación del Servidor Apache: `sudo apt-get update` `sudo apt-get install apache2`
- **Consultar estados:**
  - `systemctl status apache2`
  - `systemctl start apache2`
  - `systemctl stop apache2`
  - `systemctl restart apache2`
- **Configuración de Apache(contexto de servidor):**  
`nano /etc/apache2/apache2.conf`

# ESTRUCTURA DE DIRECTORIOS Y ARCHIVOS PRINCIPALES



**Directorio raíz de Apache:** /etc/apache2

- **Archivos de configuración:**
  - apache2.conf
  - ports.conf
  - conf-available/ (Directorio de configuraciones disponibles)
  - conf-enabled/ (Directorio de configuraciones habilitadas)
  - sites-available/ (Directorio de configuraciones de sitios disponibles)
  - sites-enabled/ (Directorio de configuraciones de sitios habilitados)
- **Carpeta como htdocs (DocumentRoot):** /var/www
- **Ruta de logs del sistema:** /var/log/apache2
- **Archivo de configuración Virtual Host por defecto:** /etc/apache2/sites-enabled/000-default.conf
  
- **Archivo de Configuración del Contexto de Servidor:**
  - Ruta: /etc/apache2/apache2.conf
  - Directiva: MaxKeepAliveRequests
  
- **Configuración del Contexto Virtual Host:**
  - Archivo por defecto: /etc/apache2/sites-enabled/000-default.conf
  - Directiva: DirectoryIndex

## DIRECTIVAS CONFIGURACIÓN RENDIMIENTO SERVIDOR:

1. Configuración del Tiempo de Espera: Timeout 180
2. Límite de Peticiones Simultáneas por Conexión: MaxClientsPerChild 50
3. Tiempo de Espera entre Peticiones: KeepAliveTimeout 10
4. Límite Máximo de Conexiones Simultáneas: MaxClients 200

# DIRECTIVAS DE BLOQUES



## - **Directory:**

- Propósito: Define la configuración para un directorio específico en el sistema de archivos.

- Ejemplo:

```
<Directory /var/www/html>  
    Options -Indexes +FollowSymLinks  
    AllowOverride None  
    Require all granted  
</Directory>
```

## - **Files:**

- Propósito: Define la configuración para archivos específicos en el sistema de archivos.

- Ejemplo:

```
<Files "example.html">  
    Require all granted  
</Files>
```

## - **VirtualHost:**

- Propósito: Define la configuración para un host virtual específico.

- Ejemplo:

```
<VirtualHost *:80>  
    ServerName example.com  
    DocumentRoot /var/www/example.com/public_html  
    ErrorLog /var/www/example.com/logs/error.log  
    CustomLog /var/www/example.com/logs/access.log combined  
</VirtualHost>
```

## - **DirectoryMatch y FilesMatch:**

- Similar a `<Directory>` y `<Files>`, pero permite utilizar expresiones regulares para coincidir con múltiples directorios o archivos.

# DIRECTIVAS DE BLOQUES

## - **DirectoryIndex:**

- Define el archivo que se servirá cuando se solicita un directorio sin un archivo específico.

## - **DocumentRoot:**

- Establece el directorio raíz para un host virtual o servidor.

## - **FollowSymLinks:**

- Permite que Apache siga enlaces simbólicos dentro del sistema de archivos.

## - **AllowOverride:**

- Define qué directivas pueden ser sobrescritas por archivos .htaccess en un directorio específico.

## - **Require:**

- Especifica los requisitos de acceso para un recurso en particular.

## - **Options:**

- Configura las opciones disponibles para un directorio o ubicación específica.

```
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/prueba
    DirectoryIndex prueba.html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
    <Directory /var/www/html>
        Options -Indexes
    </Directory>
```

```
root@deap12:~# service apache2 restart
root@deap12:~#
```



## DESCRIPCIÓN DEL SERVICIO

# — Webalizer

Webalizer es una herramienta de análisis de registros de acceso a servidores web. Su función principal es procesar los archivos de registro generados por el servidor web y presentar la información de manera visual y fácil de entender en forma de estadísticas y gráficos.

## FUNCIONES PRINCIPALES

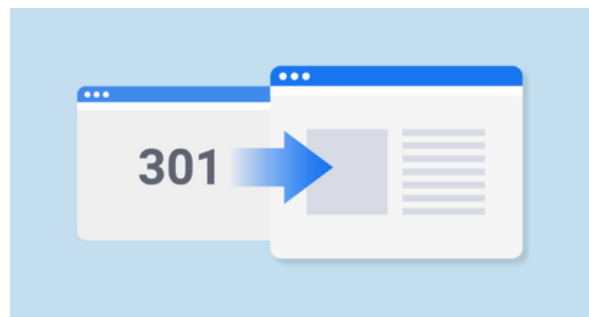
- **Monitoreo del tráfico web:** Webalizer permite a los administradores de sitios web analizar la interacción de los usuarios, identificar áreas de mejora y entender la popularidad de las páginas.
- **Optimización del contenido:** Permite identificar páginas populares, términos de búsqueda, y áreas del sitio que necesitan atención.
- **Seguimiento de campañas de marketing:** Ayuda a evaluar la efectividad de campañas en línea como anuncios pagados o redes sociales.
- **Detección de problemas de servidor:** Permite identificar problemas de rendimiento o seguridad como ataques DDoS.
- **Análisis de registros de acceso:** Examina información de solicitudes recibidas por el servidor como direcciones IP de visitantes y páginas solicitadas.
- **Generación de estadísticas:** Proporciona datos detallados sobre el tráfico web incluyendo visitas, páginas vistas, visitantes únicos, navegadores y sistemas operativos utilizados.

## COMANDOS

- Instalación: `sudo apt-get install webalizer`
- Archivo de Configuración: `/etc/webalizer/webalizer.conf`
- Directorio de Salida: `sudo mkdir -p /var/www/html/webalizer`
- Acceso: `http://dirección_servidor_Linux/webalizer/index.html`

# REDIRECCIONES

---



Cuando un usuario ingresa una dirección web, el servidor web Apache redirige al navegador a una nueva ubicación si es necesario, asegurando que llegue al contenido correcto.

Esto se logra a través de códigos de estado HTTP, como el 301 para redirecciones permanentes o el 302 para redirecciones temporales. De esta manera, el servidor indica al navegador la acción a seguir para garantizar una experiencia de navegación fluida y eficiente para el usuario. La capacidad de redireccionamiento del servidor Apache es una herramienta fundamental para mantener actualizada la estructura de un sitio web y facilitar la navegación de los visitantes.

## - **Redirect:**

- Esta directiva redirige una URL a otra ubicación especificada. Se utiliza para redireccionar solicitudes de los usuarios a diferentes direcciones URL.

## - **Alias:**

- Define un alias (nombre alternativo) para una ruta específica en el servidor web. Se utiliza para asociar una URL con una ubicación diferente en el sistema de archivos.

## - **ErrorDocument:**

- Permite que Apache siga enlaces simbólicos dentro del sistema de archivos.

## - **LogLevel:**

- Define el nivel de detalle para los mensajes de registro (logging) del servidor web Apache. Se utiliza para especificar qué tipo de eventos se registrarán, como información de depuración (debug), advertencias o errores.

## - **ErrorLog:**

- Especifica la ubicación del archivo de registro de errores del servidor web Apache. Indica dónde se guardarán los mensajes de error generados por el servidor.

## - **CustomLog:**

- Configura el archivo de registro de accesos personalizado del servidor web Apache. Permite especificar dónde se almacenarán los registros de acceso generados por las solicitudes de los usuarios.

# AUTENTICACIÓN

---

La autenticación Apache permite proteger el acceso a ciertos recursos del servidor mediante la solicitud de credenciales de usuario. Cuando un cliente intenta acceder a un recurso protegido, Apache le solicita un nombre de usuario y una contraseña. Este proceso garantiza que solo los usuarios autorizados puedan acceder a la información protegida.

En el caso de la autenticación básica, las credenciales se envían al servidor en texto plano, mientras que en la autenticación digest se utiliza un algoritmo de hash para proteger las credenciales durante la transmisión. Esto asegura un mayor nivel de seguridad en comparación con la autenticación básica.

La configuración de la autenticación se realiza mediante la definición de grupos de usuarios y la asignación de permisos de acceso a los recursos del servidor. Esto permite controlar quién puede acceder a qué recursos y garantiza la seguridad de la información almacenada en el servidor web Apache.

## - Módulos estáticos cargados:

- Utiliza el comando `apache2ctl -M` para mostrar los módulos estáticos cargados en Apache.

## - Paquetes disponibles en el repositorio de Ubuntu Server:

- Emplea el comando `apt-cache search apache2` para listar los paquetes disponibles en el repositorio de Ubuntu Server relacionados con Apache.

## - Activación y configuración del módulo `server-status` y `server-info`:

- Utiliza la directiva `<Location>` en el archivo de configuración de Apache para habilitar el acceso a `/server-status` y `/server-info`.

Inicia los manejadores de los módulos correspondientes para habilitar los servicios.

Restringe el acceso a estas rutas solo a la máquina física mediante la configuración adecuada.

## - Habilitación del módulo `userdir`:

- Habilita el módulo `userdir` utilizando el comando `a2enmod userdir`.
- Verifica en el archivo de configuración del módulo `userdir` que el uso de directorios personales esté permitido para todos los usuarios excepto para `root`.
- Crea el directorio `/home/usuarioxx/public_html` según corresponda.

# AUTENTICACIÓN

---

## - Instalación de servicios adicionales:

- Instala MySQL y PHP utilizando los comandos `apt-get install mysql-server mysql-client php-mysql` y `apt-get install php libapache2-mod-php php-mcrypt php-mysql`.

Configura el servidor para que presente un archivo PHP por defecto cuando se accede a la dirección IP del servidor.

## - Instalación de phpMyAdmin:

- Instala phpMyAdmin utilizando el comando `apt-get install phpmyadmin`.
- Verifica el acceso a phpMyAdmin desde la máquina física.
- 

## - Configuración de autenticación Apache:

- Configura la autenticación básica y digest según corresponda para los directorios y usuarios especificados.
- Crea archivos de texto con las credenciales de los usuarios en los directorios correspondientes.
- Restringe el acceso a las rutas necesarias y define los grupos de usuarios según los requerimientos.

# DIRECTIVAS

## Autenticación Básica:

- `AuthType`: Define el tipo de autenticación como básica.
- `AuthName`: Especifica el nombre del área protegida.
- `AuthUserFile`: Indica la ubicación del archivo de contraseñas que contiene las credenciales de los usuarios.
- `Require`: Establece los requisitos de autenticación para acceder al recurso, como la autenticación válida del usuario.

# AUTENTICACIÓN

## Autenticación Digest:

- AuthType: Define el tipo de autenticación como Digest.
- AuthName: Especifica el nombre del área protegida.
- AuthDigestFile: Indica la ubicación del archivo de contraseñas para la autenticación digest.
- Require: Establece los requisitos de autenticación para acceder al recurso.

## Autenticación Digest:

- <Location>: Define la ubicación en la que se aplicarán las directivas de autenticación.
- Require: Establece los requisitos de autenticación específicos para una URL o un conjunto de URLs.

Estas directivas permiten configurar la autenticación básica y digest en Apache, restringiendo el acceso a los recursos del servidor web y asegurando que solo los usuarios autorizados puedan acceder a ellos.

```

GNU nano 2.2.2      Archivo: 000-default      Modificado

<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Order deny,allow
        Deny from all
        AllowOverride AuthConfig
    </Directory>
    <Directory /var/www/>
        Order allow,deny
        allow from all
        Options None
    </Directory>

    <Directory /var/www/scripts/>
        Order allow,deny
        allow from all
        Options +ExecCGI
    </Directory>

    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
    </Directory>

[ 47 líneas leídas ]
^G Ver ayuda  ^O Guardar    ^R Leer Fich  ^Y RePág.    ^K Cortar Tex ^C Pos actual
^X Salir      ^J Justificar ^W Buscar     ^V Pág. Sig. ^U PegarTxt   ^T Ortografía
  
```



# SSL/TLS

---

El servicio HTTPS (SSL/TLS) en un servidor web Apache se utiliza para proporcionar comunicaciones seguras a través de Internet. HTTPS cifra la información transmitida entre el servidor y el cliente, protegiendo así la privacidad y seguridad de los datos. Esto es especialmente importante cuando se trata de información confidencial, como contraseñas, datos financieros o personales.

Los sitios web que utilizan HTTPS muestran un candado en el navegador, lo que indica que la conexión es segura y autenticada. Esto ayuda a generar confianza en los usuarios y protege contra ataques como el robo de datos o la suplantación de identidad.

## COMANDOS

### **a2ensite:**

- Este comando habilita un sitio web (host virtual) en Apache.

### **netstat -ltn:**

- Se utiliza para listar los puertos en los que el servidor está escuchando conexiones.

**openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout myserver.key -out myserver.crt:**

- Genera un certificado autofirmado junto con su clave privada utilizando OpenSSL.

## SSL/TLS



### DIRECTIVAS

#### **SSLEngine:**

- Habilita o deshabilita el motor SSL/TLS para un servidor virtual.

#### **SSLCertificateFile:**

- Especifica la ruta al archivo que contiene el certificado SSL.

#### **SSLCertificateKeyFile:**

- Indica la ruta al archivo que contiene la clave privada del certificado SSL.

**<VirtualHost>:** Define la configuración de un host virtual, incluyendo la configuración específica para SSL/TLS.

**ServerName:** Especifica el nombre del servidor para el host virtual.

**DocumentRoot:** Establece el directorio raíz para el host virtual.

**<Location>:** Define la configuración de una ubicación específica dentro del host virtual, en este caso, para la redirección de URLs en HTTPS.



Los Virtual Hosts en Apache permiten alojar múltiples sitios web en un solo servidor físico. Esto se logra mediante la asignación de nombres de dominio o direcciones IP específicas a diferentes conjuntos de archivos y directorios en el servidor.

## COMANDOS

### **a2ensite:**

- Este comando habilita un sitio web (host virtual) en Apache.

### **netstat -ltn:**

- Se utiliza para listar los puertos en los que el servidor está escuchando conexiones.

### **cp**

- Se utiliza para copiar archivos o directorios. En este caso, se usa para copiar el archivo de configuración del host virtual por defecto a una nueva ubicación.

```
GNU nano 2.5.3      File: /etc/apache2/sites-available/evg.es.conf
<VirtualHost 192.168.201.67:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    ServerName evg.es
```

# HOSTS

---

## DIRECTIVAS

### <VirtualHost>:

- Define la configuración de un host virtual en Apache.

### ServerName:

- **Especifica el nombre del servidor para el host virtual.**

### DocumentRoot:

- **Establece el directorio raíz para el host virtual.**

### Alias:

- Crea un alias para un directorio. Se utiliza para definir alias para subdirectorios en el host virtual

### ServerAlias:

- Especifica otros nombres de servidor que el servidor puede usar para identificarse.

### <Directory>:

- Define las directivas de configuración que se aplican a un directorio específico.

### AllowOverride:

- Permite que las directivas en archivos .htaccess sobrescriban la configuración del servidor.

```
Alias "/apunteshome" "/home/apunteshome"

<Directory /var/www/html/evg>
    Options Indexes FollowSymLinks
    AllowOverride All
    DirectoryIndex indexevg.es
    Require all granted
</Directory>
```

# LO APRENDIDO

---

Manejo de ubuntu server:

- Principalmente el uso de sus archivos de configuración para servidor DNS y servicio web Apache.
- Comandos útiles a la hora de administrar un servidor.
- Aprender a usar documentación sobre un tema concreto.

# RECURSOS

---

Lo más utilizado para realizar las pruebas han sido:

- Documentación oficial
- Comunidad
- Chat GPT

# DIFICULTADES

---

Dificultades:

- Entender qué hace cada comando que utilizas.
- Encontrar información concreta.