



Arquitectura de Información en Web

Práctica 1

Gustavo Antonio Luna Maya

Isaac Eduardo Odriozola Díaz

José Antonio Pérez Durán

Primavera 2021

Índice

Práctica 1	1
Introducción	3
Contenido	3
Ventajas y Desventajas	3
Desarrollo o guía para la instalación y configuración del servidor	5
Crear una máquina virtual con Hyper-V.	5
Instalar el sistema operativo CentOS.	6
Instalar y configurar NginX.	7
Instalar y configurar Mariadb.	9
Comprobar el funcionamiento de PHP.	11
Configurar conexión HTTPS.	12
Configurar protocolo HTTP v2.	13
Conclusiones	14
Bibliografías	15

Introducción

En esta práctica se mostrará el proceso que se tiene que llevar a cabo para la instalación y configuración de un servidor con el uso de nativo de hyper-V de Windows 10 instalando el sistema operativo CentOS, instalando y configurando Nginx el cual se usará como servidor web HTTP, como base de datos para alojar nuestro archivo PHP usaremos el servicio de base de datos MariaDB, como ya mencionamos se realizará la configuración de la base de datos con el lenguaje de programación PHP, se configura la conexión HTTPS para obtener un certificado de seguridad y se configurara el protocolo HTTP v2. También explicaremos por qué se tomó la decisión de usar cada tecnología, servicio y lenguaje para realizar la configuración del servidor para tener una visión clara de las diferentes opciones que ofrece el mercado y cual puede tener un mayor beneficio para nosotros.

Contenido

Dentro de la actualidad podemos montar entornos para un desarrollo, en nuestro caso realizamos una máquina virtual con herramientas diferentes. Por un lado se implementó un entorno de desarrollo por medio de Hyper-V para obtener el servicio de servidor virtual debido a que nos permite trabajar simultáneamente de manera amigable, nos permite determinar un espacio para correr el servicio y especificar en su configuración el tipo de entorno que se va a estar utilizando.

El sistema operativo CentOS nos brinda la plataforma para poder montar nuestro entorno, nos da velocidad y estabilidad en el sistema Linux montado. Todo empieza a tomar mejor camino cuando utilizamos NginX porque nos permite alojar nuestra versión de PHP que se encargará de hacer todas las consultas y mostrará nuestras especificaciones a la hora de programar. Por otro lado, con MariaDB tenemos nuestra base de datos que se encarga de agrupar la información para poderla mandar nuevamente de vuelta al cliente siempre que sea solicitada.

Ventajas y Desventajas

VMware vs Hyper-v

Una de las ventajas de trabajar con Hyper-V que es una herramienta de virtualización en windows 10 es que se encuentra de forma nativa en el sistema operativo, solo es necesario activarla, claro que como desventaja pues es que solo se encuentra en el sistema operativo windows 10, Estas herramientas requieren como requisitos un procesador de 2 núcleos y como mínimo 4 GB de memoria RAM física, las dos herramientas cuentan con un nivel de personalización alto ya que podemos asignar la cantidad de núcleo de procesador, cantidad de memoria RAM, cantidad de memoria para almacenamiento entre otras características del sistema, VMware como ventaja tiene que puede ser usada tanto en sistemas operativos

windows como en sistemas operativos IOS pero como desventaja es un servicio para el cual es necesario adquirir una licencia con cierto costo.

Nosotros decidimos usar Hyper V por que nos permite ejecutar varios sistemas operativos como máquinas virtuales en Windows, en lo personal si un sistema operativo ofrece la herramienta en este caso Hyper V en windows aprovechamos mejor los recursos del sistema operativo ya que es una aplicación nativa y por ende tienden a tener un mejor rendimiento que las aplicaciones de terceros, algunas de las razones para usar la virtualización es que nos permite experimentar con otros sistemas operativos Hyper-V hace que sea muy fácil crear y eliminar diferentes sistemas operativos, también se pueden exportar e importar a cualquier otro sistema Hyper V, incluido Azure, sin dejar de mencionar la compatibilidad con la mayoría de los sistemas operativos como son Linux, FreeBSD, Windows.

CentOS

Se usará el sistema operativo CentOS por su estabilidad ya que CentOS solo ejecuta las versiones más básicas y estables de los programas en el sistema operativo, de esta forma logramos obtener un mejor rendimiento y reducir el riesgo de bloqueos del sistema o caídas del sistema; Por su velocidad ya que si el procesador solo ejecuta las versiones más básicas y estables de los programas al ejecutar varias a la vez lo hace de una forma más consistente y por que los programas CentOS tiene una menor probabilidad de contar con bloqueos o provocar bloqueos en el sistema y por su confiabilidad ya que el equipo que desarrolla las actualización del sistema operativo tiene un ciclo de aproximadamente 5 años por lo que en pocas palabras cada cinco años no nos tendremos que preocupar por las notas de las actualizaciones y los problemas que cualquier cambio en el sistema podría afectar el funcionamiento o rendimiento de nuestros proyectos desarrollados en el sistema operativo.

Nginx

Usamos este servicio por delante de otros por ejemplo Apache ya que Nginx abre solo los hilos de ejecución justos y necesarios permitiendo servir millones de peticiones en un corto espacio de tiempo, ya que no requiere tiempo adicional para abrir nuevos procesos y además al no abrir nuevos procesos tampoco consume más memoria RAM.

La diferencia de rendimiento entre Nginx y Apache se nota, ya que el tiempo de respuesta conseguido por Nginx es casi un 150% más rápido que en el caso de Apache, también porque al usar PHP-FPM consume mucha menos memoria al procesar código PHP y funciona de forma independiente al servidor web, mediante el uso de procesos externos y sockets para funcionar. Referencia: Fontela, A. (2018, 24 abril). *Nginx: ventajas y desventajas*. Raiola Networks. <https://raiolanetworks.es/blog/nginx/#:%7E:text=La%20principal%20ventaja%20de%20Nginx,optimizando%20la%20entrega%20de%20contenidos>.

MariaDB

Utilizamos MariaDB frente a MySQL por que Mariadb ofrece medidas de seguridad más estrictas por ejemplo: Seguridad interna y verificación de contraseña, autenticación PAM y LDAP, roles de usuario, cifrado de la base de datos; También por su rendimiento que es mas rapido y mas eficiente comparado al de MySQL, ya que las dos opciones son ciertamente eficientes Mariadb a logrado permanecer como uno de los servicios de gestión de bases de datos más eficiente gracias a la selección de motores de bases de datos alternativos; Tambien por que tiene un mejor acceso a la asistencia al usuario, algo muy importante en esta clase de servicios ya que de ellos depende el correcto funcionamiento de nuestros sitios web o de nuestros proyectos.

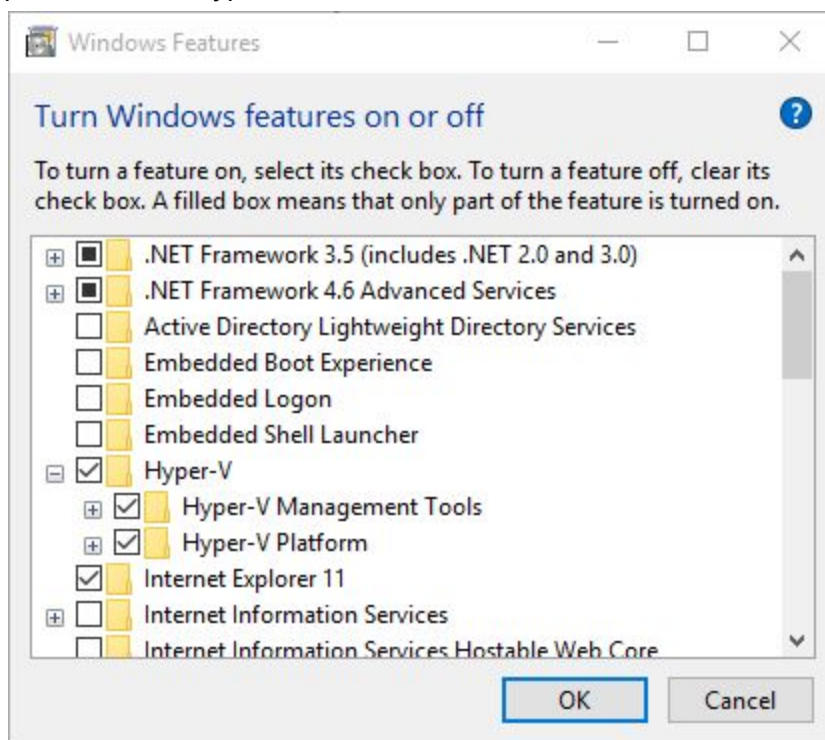
Desarrollo o guía para la instalacion y configuracion del servidor

1. Crear una máquina virtual con Hyper-V.

Para comenzar tenemos que habilitar la característica hyper-v desde la consola de comandos o CMD, escribiendo en el navegador de windows o barra de búsqueda "CMD", en la consola escribiremos el siguiente comando:

```
DISM /Online /Enable-Feature /All /FeatureName:Microsoft-Hyper-V
```

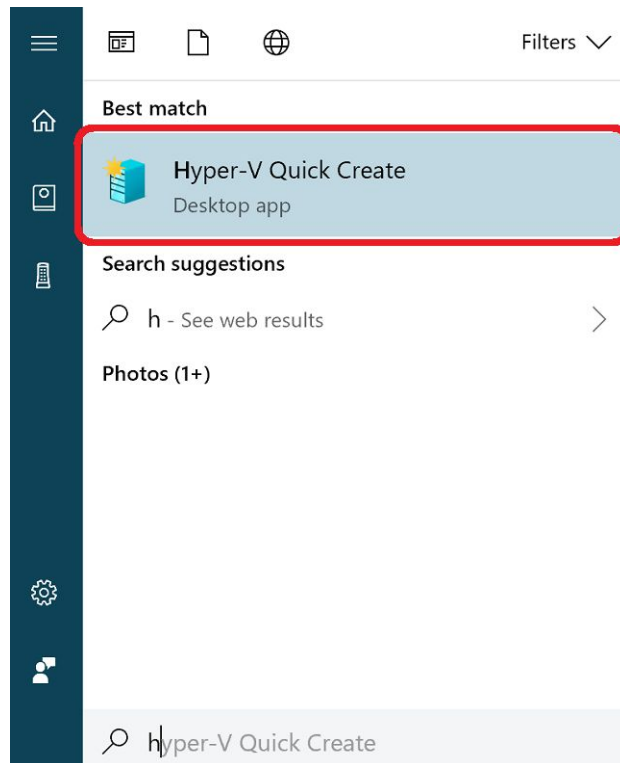
despues click derecho sobre el botón windows y seleccionar la opción de aplicaciones y características, seleccionar programas y características y activar características de windows para activar el hyper-v.



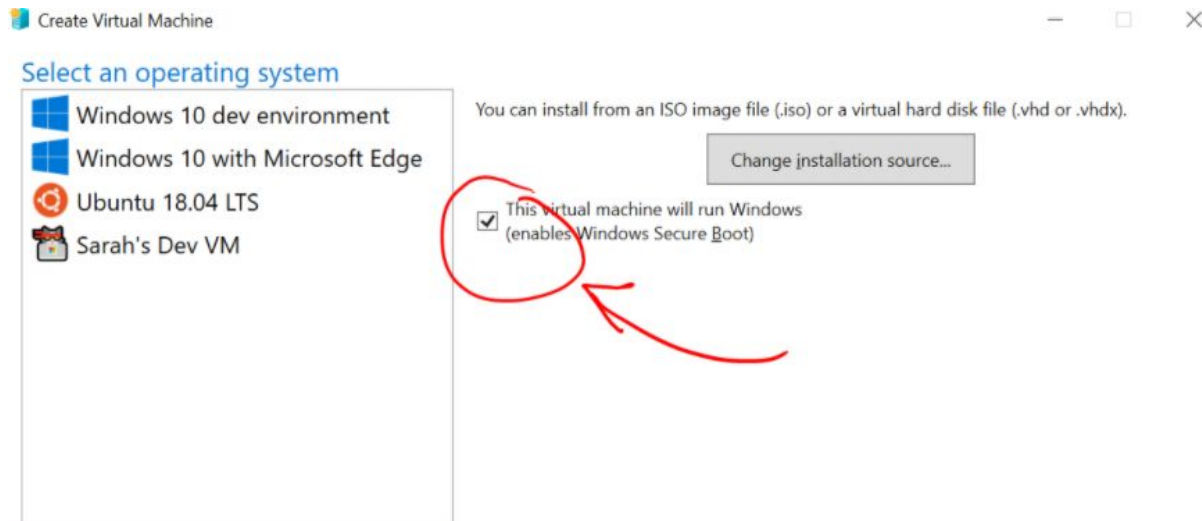
Listo ya tenemos activada la característica de windows 10 Hyper-v ahora configuraremos la máquina virtual.

2. Instalar el sistema operativo CentOS.

Para instalar el sistema operativo CentOS es necesario cumplir con el paso anterior, buscar en la barra de búsqueda de windows “Hyper-V Quick Create”



Continuamos con la configuración del sistema operativo, para esto previamente tendríamos que haber descargado el sistema operativo o imagen de disco CentOS para cargarlo en la configuración de Hyper V.



3. *Instalar y configurar NginX.*

Primero tenemos que asegurarnos de tener actualizado el repositorio de paquetes del sistema operativo con los siguientes comandos:

```
sudo apt update
```

```
sudo apt upgrade
```

Después necesitamos incluir en la terminal la siguiente línea de comando para proceder con la instalación de NginX:

```
sudo apt install nginx
```

Podemos habilitar NginX de la siguiente manera:

```
sudo systemctl enable nginx
```

Una vez que queramos correr NginX podemos comprobarlo a través del siguiente comando para correrlo:

```
sudo systemctl start nginx
```

Para comprobar que esté todo adecuadamente configurado solo es necesario comprobar el status de NginX, lo podemos realizar con este comando:

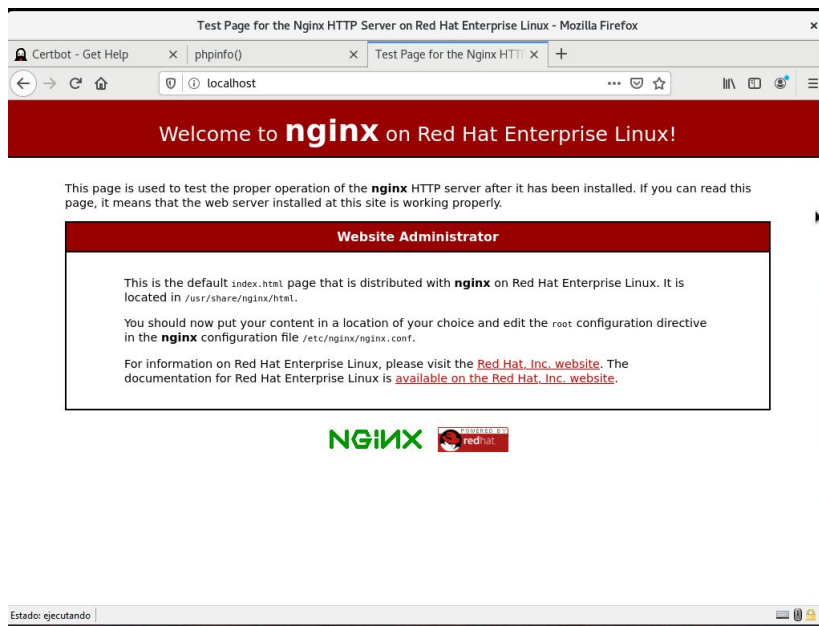
```
systemctl status nginx
```

Como salida esperamos obtener un mensaje de este tipo:

```
antonio@localhost:/etc/nginx/conf.d
Archivo  Editar  Ver  Buscar  Terminal  Ayuda

feb 12 04:11:10 localhost.localdomain systemd[1]: Starting The nginx HTTP and reverse proxy process:
feb 12 04:11:11 localhost.localdomain nginx[1343]: nginx: the configuration file /etc/nginx/nginx.conf was not found (2)
feb 12 04:11:11 localhost.localdomain nginx[1343]: nginx: configuration file /etc/nginx/nginx.conf not found (2)
feb 12 04:11:11 localhost.localdomain systemd[1]: Started The nginx HTTP and reverse proxy process:
[root@localhost conf.d]# systemctl status mariadb
● mariadb.service - MariaDB 10.3 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2021-02-12 04:11:10 EST; 6h ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 1152 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 30 (limit: 10779)
    Memory: 81.6M
    CGroup: /system.slice/mariadb.service
            └─1152 /usr/libexec/mysqld --basedir=/usr
```

Continuamos escribiendo en nuestro buscador de internet la siguiente dirección IP `127.0.0.1` o `localhost`. Nos mostrará una ventana como la siguiente:



Finalmente en nuestra consola de comandos escribiremos el siguiente código para hacer que el usuario “www-data” (Nginx User) sea el usuario ruta o root:

```
sudo chown www-data:www-data /usr/share/nginx/html -R
```

4. Instalar y configurar Mariadb.

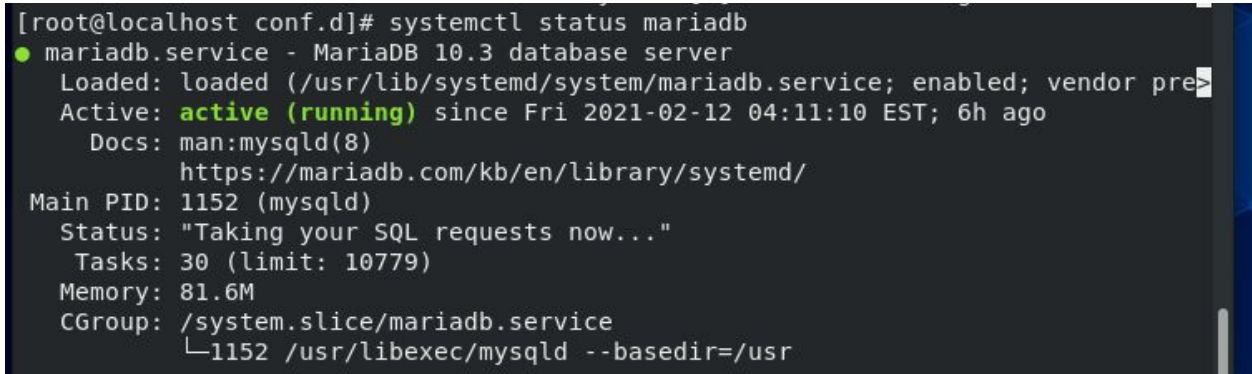
Continuamos con la instalacion y configuracion del servicio de gestión a bases de datos MariaDB, empecemos instalando el cliente con el siguiente código desde la terminal:

```
sudo apt install mariadb-server mariadb-client
```

Después de su instalación el servidor de MariaDB se tendría que actualizar automaticamente y lo podemos comprobar con el siguiente codigo:

```
systemctl status mariadb
```

Espera ver como salida lo siguiente:

A screenshot of a terminal window showing the output of the command 'systemctl status mariadb'. The output indicates that the mariadb.service is loaded and active (running). It shows details such as the main PID (1152), status ('Taking your SQL requests now...'), tasks (30), memory (81.6M), and CGroup. The terminal background is dark with light-colored text.

```
[root@localhost conf.d]# systemctl status mariadb
● mariadb.service - MariaDB 10.3 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor pre>
   Active: active (running) since Fri 2021-02-12 04:11:10 EST; 6h ago
     Docs: man:mysql(8)
           https://mariadb.com/kb/en/library/systemd/
   Main PID: 1152 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 30 (limit: 10779)
    Memory: 81.6M
    CGroup: /system.slice/mariadb.service
            └─1152 /usr/libexec/mysqld --basedir=/usr
```

Si no esta corriendo lo podemos hacer con el siguiente comando:

```
sudo systemctl start mariadb
```

Si quieres habilitar MariaDB al arrancar la máquina virtual o el sistema operativo escribe el siguiente comando:

```
sudo systemctl enable mariadb
```

Ahora podemos correr el script de seguridad después de la instalación:

```
sudo mysql_secure_installation
```

Cuando solicite introducir la contraseña como usuario raíz presionaremos la tecla Enter y después la letra Y para responder Yes a la pregunta Set root password? como se muestra en la siguiente imagen:

```
linuxbabe@bionic: ~  
linuxbabe@bionic:~$ sudo mysql_secure_installation  
  
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!  
  
In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
you haven't set the root password yet, the password will be blank,  
so you should just press enter here.  
  
Enter current password for root (enter for none): Press Enter  
OK, successfully used password, moving on...  
  
Setting the root password ensures that nobody can log into the MariaDB  
root user without the proper authorisation.  
  
Set root password? [Y/n] y Enter Y to set root password  
New password:  
Re-enter new password:
```

Después presionaremos la tecla Enter para que se elimine el usuario anónimo y la base de datos de prueba de mariaDB:

```
linuxbabe@bionic: ~  
  
By default, a MariaDB installation has an anonymous user, allowing anyone  
to log into MariaDB without having to have a user account created for  
them. This is intended only for testing, and to make the installation  
go a bit smoother. You should remove them before moving into a  
production environment.  
  
Remove anonymous users? [Y/n] Press Enter  
... Success!  
  
Normally, root should only be allowed to connect from 'localhost'. This  
ensures that someone cannot guess at the root password from the network.  
  
Disallow root login remotely? [Y/n] Press Enter  
... Success!
```

```
By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Press Enter
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Press Enter
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
linuxbabe@bionic:~$
```

Para iniciar sesión en MariaDB con el usuario con el que inicias sesión en el sistema operativo utiliza el siguiente comando:

```
sudo mariadb -u root
```

Para salir de MariaDB:

```
exit;
```

Para revisar o verificar la versión del servicio:

```
mariadb --version
```

5. Comprobar el funcionamiento de PHP.

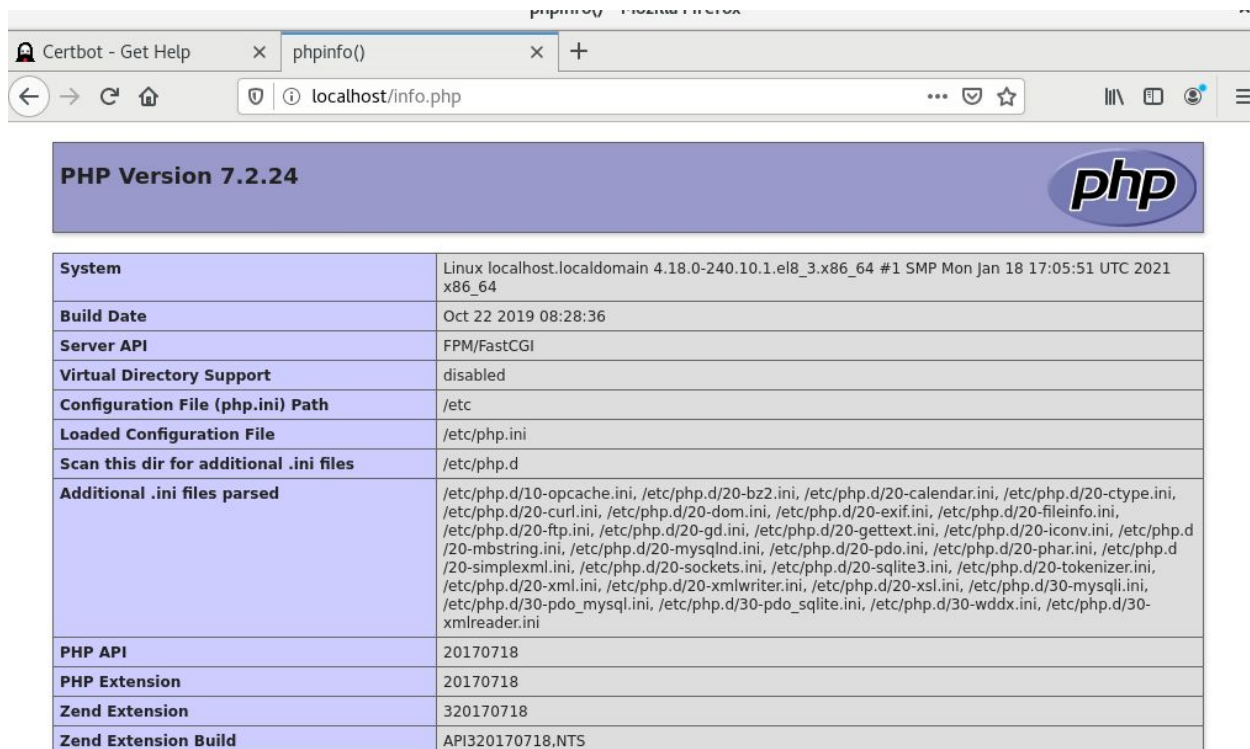
Para poder comprobar si NginX está corriendo adecuadamente PHP debemos seguir los siguientes pasos. Debemos crear un archivo que se llame info.php desde la raíz del directorio:

```
sudo nano /usr/share/nginx/html/info.php
```

Al abrir el archivo, escribimos el siguiente encabezado para comprobar que PHP efectivamente esté funcionando:

```
<?php phpinfo(); ?>
```

Una vez guardado el archivo, podemos ir a la siguiente referencia “localhost/info.php” y nos debe aparecer una ventana similar a la siguiente:



PHP Version 7.2.24	
System	Linux localhost.localdomain 4.18.0-240.10.1.el8_3.x86_64 #1 SMP Mon Jan 18 17:05:51 UTC 2021 x86_64
Build Date	Oct 22 2019 08:28:36
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-wddx.ini, /etc/php.d/30-xmlreader.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718,NTS

En nuestro caso la versión que tenemos para PHP es la 7.2.24 y podemos leer sus especificaciones.

6. Configurar conexión HTTPS.

Para poder implementar el protocolo HTTPS es sumamente sencillo, la cuestión es entender los conceptos uno de ellos es el certificado SSL y el dominio el cual tienen un costo, pero para la configuración de esta práctica se hizo un SSL y un dominio local para fines prácticos.

Se necesita un certificado SSL para obtener seguridad en tu navegación con HTTP, es posible cifrar la conexión por medio de estos certificados autofirmados:

```
~$ sudo openssl req -newkey rsa:2048 -nodes -keyout
/etc/pki/tls/private/centos8.local.key -x509 -days 365 -out
/etc/pki/tls/certs/centos8.local.crt
```

CentOS 8 nos preguntará por el dominio de la máquina, <<Common Name>> de manera semejante a lo siguiente:

```
...
Common Name (eg, your name or your server's hostname) []:centos8.local
...
```

El archivo principal se edita con la siguiente línea:

```
~$ sudo nano /etc/nginx/nginx.conf
```

Posteriormente debemos acceder a la configuración de SSL:

```
...
# Settings for a TLS enabled server.
#
#    server {
#        listen      443 ssl http2 default_server;
#        listen      [::]:443 ssl http2 default_server;
#        server_name _;
#        root         /usr/share/nginx/html;
#
#        ssl_certificate "/etc/pki/nginx/server.crt";
#        ssl_certificate_key "/etc/pki/nginx/private/server.key";
#
#    }
...
```

Para descomentar las siguientes líneas:

```
...
    ssl_certificate "/etc/pki/tls/certs/centos8.local.crt";
    ssl_certificate_key "/etc/pki/tls/private/centos8.local.key";
...
```

Guardamos los cambios:

```
~$ sudo systemctl reload nginx
```

7. Configurar protocolo HTTP v2.

Podemos ver desde NginX la versión de HTTP con el siguiente comando:

```
sudo nginx -V
```

Si la versión es al menos 1.9.5 o posterior, nos indicará que está adecuadamente instalado:

```
--with-http_v2_module
```

Luego es necesario configurar el servidor con alguna de estas dos opciones de comando, dependiendo el caso de si tienes un sitio predefinido o no:

```
sudo vi /etc/nginx/conf.d/yourdomain.conf
-----
sudo vi /etc/nginx/sites-available/yourdomain.conf
```

En el servidor debe aparecer la siguiente línea que indica que HTTP v2 está corriendo adecuadamente:

```
server {
    listen 443 ssl http2;
    server_name www.linuxbabe.com;

    ssl_certificate /etc/ssl/fullchain.pem;
    ssl_certificate_key /etc/ssl/privkey.pem;

    .....
}
```

Luego es necesario recargar NginX para poder observar los cambios y listo:

```
sudo service nginx reload
-----
sudo systemctl reload nginx
```

Conclusiones

Gustavo Antonio Luna Maya

La instalación y configuración de un servidor aparte reforzar nuestros conocimientos y habilidades en informática nos permite contar con un ambiente de desarrollo propio en el cual podemos experimentar con funciones, desarrollar nuevos proyectos, probar de forma controlada y eficaz nuestros proyectos, y al tener una guía en la cual se representan los pasos a seguir para la implementación de este servidor lo vuelve una tarea muy sencilla ya que después de la segunda o tercera vez que necesitemos configurar un servidor de este tipo probablemente sea una tarea que nos tome poco menos de treinta minutos, la cual consideramos es una herramienta con mucha importancia.

Odriozola Díaz Isaac Eduardo:

La aplicación de una máquina virtual aporta habilidades para poder montar un entorno de desarrollo adecuado, estos entornos nos pueden hacer aprender las tecnologías del momento que pueden dar solución a muchos problemas cotidianos. Los proyectos deben ser bien implementados, cuando aplicamos las herramientas necesarias podemos aplicar todos los conocimientos de programación que hemos obtenido a lo largo de la carrera. Para finalizar es necesario comentar que el desarrollo y aplicación de estos entornos no es fácil y solo las personas que conocen del tema son las indicadas para desarrollarlo adecuadamente y de forma funcional.

José Antonio Pérez Durán

La práctica me pareció sumamente importante para nuestro perfil, nos ayuda a poder entender cómo la tecnología avanza en la forma de cómo podemos montar un servidor y los servicios que requerimos para nuestro entorno de programación un ejemplo de esto fue Hyper-V que en los últimos años ha tenido actualizaciones el cual permite que sea más fácil montar las máquinas virtuales y soportar más sistemas operativos, por otro lado el SO CentOS8 me pareció genial cada vez vemos como el UX y UI toma más protagonismo su simpleza y minimalismo hace no echas de menos a windows o mac OS, el servidor NGINX me pareció estupendo tiene muchos parámetros de seguridad y muy buen soporte para las empresas eso hace que en la actualidad muchas empresas lo implementan, el gestor de bases de datos MariaDB es uno de mis favoritos y sin duda para el entorno de desarrollo con bases de datos relacionales es la mejor opción y ya para concluir PHP el lenguaje favorito de los developer backend (broma) sigue teniendo mucha relevancia en el desarrollo web. y la seguridad creo que es lo más importante algo que me gusto mucho fue poder ver como funciona HTTP y HTTPS V2 aunque nos complicamos un poco en la configuración ami en lo personal me deja una grata satisfacción al final de esta práctica creo que ahora podemos montar lo que sea.

Bibliografías

Rachele, W. (s. f.). *Ejemplos de programas para la administración de bases de datos*. Techlandia. Recuperado 11 de febrero de 2021, de https://techlandia.com/ejemplos-programas-administracion-bases-datos-lista_126285/

Ortiz, A. E. (2020, 16 junio). *3 ventajas de usar MariaDB sobre MySQL*. Blog HostDime Argentina, servidores dedicados. <https://www.hostdime.com.ar/blog/3-ventajas-de-usar-mariadb-sobre-mysql/>

Fontela, A. (2018, 24 abril). *Nginx: ventajas y desventajas*. Raiola Networks. <https://raiolanetworks.es/blog/nginx/#:%7E:text=La%20principal%20ventaja%20de%20Nginx,optimizando%20la%20entrega%20de%20contenidos.>

Instalador, E. (2019, 3 octubre). *Cómo instalar Nginx en CentOS 8*. Cómo instalar... https://comoinstalar.me/como-instalar-nginx-en-centos-8/#Como_configurar_seguridad_HTTPS_SSL

Guoan, X. (2016, 29 junio). *What are SPDY and HTTP/2 and How to Enable Them on Nginx*. LinuxBabe. <https://www.linuxbabe.com/nginx/what-are-spdy-and-http2-and-how-to-enable-them-on-nginx>

S. (2018, 25 junio). *Introduction to Hyper-V on Windows 10*. Microsoft Docs. <https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/about/>