



Hoja01_Herramientas_01

1.- Crear en la nube de Azure una máquina virtual (Azure Portal).

Con la cuenta de Educantabria puedes solicitar un crédito de 100\$ en el siguiente enlace:

<https://azure.microsoft.com/en-us/free/students/>

Una vez creada la cuenta nos vamos a conectar al portal de Azure y vamos a crear una máquina virtual de Linux, en nuestro caso: un Ubuntu 22.04LTS para realizar la instalación sigue los pasos que Microsoft nos facilita en la siguiente página web.

<https://learn.microsoft.com/es-es/azure/virtual-machines/>

Nos podemos dirigir al apartado entrenamiento autodirigido al curso “creación de una máquina virtual Linux en Azure.

Aquí nos ofrece una serie de guías para entender el portal de Azure y llegar a crear la máquina virtual

<https://learn.microsoft.com/es-es/training/paths/azure-linux/>

Os recomiendo ir estudiando cada uno de los puntos hasta llegar al Aprovisionamiento de una máquina virtual

<https://learn.microsoft.com/es-es/training/modules/create-linux-virtual-machine-in-azure/>

Primero de todo creamos una maquina virtual en azure con las especificaciones más bajas (así no nos consume tanto).

Terminamos de realizar la configuración de la máquina, en este caso vemos con que tipo de acceso nos queremos conectar.

Cuenta de administrador

Tipo de autenticación ⓘ

- ☒ Clave pública SSH
☐ Contraseña

ⓘ Ahora, Azure genera automáticamente un par de claves SSH y le permite almacenarlo para usarlo en el futuro. Es una forma rápida, sencilla y segura de conectarse a la máquina virtual.

Nombre de usuario * ⓘ

azureuser ✓

Origen de clave pública SSH

Generar un par de claves nuevo ✓

Tipo de clave SSH

- ☒ Formato RSA SSH
☐ Formato Ed25519 SSH

👉 Ed25519 ofrece mejor rendimiento y seguridad con un tamaño de clave más pequeño, mientras que RSA todavía se usa ampliamente, especialmente para sistemas y aplicaciones heredados.

Nombre de par de claves *

ServidorUbuntuAzure_key ✓

Reglas de puerto de entrada

Seleccione los puertos de red de máquina virtual que son accesibles desde la red Internet pública. Puede especificar acceso de red más limitado o granular en la pestaña Red.

Puertos de entrada públicos * ⓘ

- ☐ Ninguno
☒ Permitir los puertos seleccionados

Seleccionar puertos de entrada *

HTTP (80), SSH (22) ✓

ⓘ Se bloquea todo el tráfico de Internet de forma predeterminada. Puede cambiar las reglas del puerto de entrada en la página VM > Redes.

< Anterior

Siguiente: Discos >

Revisar y crear

Aquí vemos que todo se ha creado correctamente:

Inicio > Máquinas virtuales >

Crear una máquina virtual

Validación superada

Ayuda para crear una máquina virtual de bajo coste Ayuda para crear una VM optimizada para alta disponibilidad Ayudarme a elegir el tamaño de VM adecuado para mi carga de trabajo

Tamaño del disco del SO	Valor predeterminado de la imagen
Tipo de disco del sistema operativo	LRS de SSD prémium
Usar discos administrados	Sí
Eliminar disco de SO con VM	Habilitado
Disco de SO efímero	No

Redes

Red virtual	(nuevo) ServidorUbuntu-vnet
Subred	(nuevo) default (10.0.0.0/24)
IP pública	(nuevo) ServidorUbuntu-ip
Redes aceleradas	Desactivado
¿Quiere colocar esta máquina virtual como subyacente respecto a una solución de equilibrio de carga existente?	No
Eliminar IP pública y NIC cuando se elimine la VM	Deshabilitado

Administración

Microsoft Defender for Cloud	Ninguno
Identidad administrada asignada por el sistema	Desactivado
Inicio de sesión con Microsoft Entra ID	Desactivado
Apagado automático	Desactivado
Habilitar revisión en caliente	Desactivado
Opciones de orquestación de revisiones	Valor predeterminado de la imagen

Supervisión

Alertas	Desactivado
Diagnósticos de arranque	Activado
Habilitar diagnósticos del SO invitado	Desactivado
Habilitar supervisión de estado de la aplicación	Desactivado

Opciones avanzadas

Extensiones	Ninguno
Aplicaciones de máquina virtual	Ninguno
Cloud-init	No
Datos de usuario	No
Tipo de controladora de disco	SCSI
Grupo con ubicación por proximidad	Ninguno
Grupo de reserva de capacidad	Ninguno

< Anterior Siguiente > **Crear**

Descargamos la clave para luego conectarnos mediante SSH:

Generar un par de claves nuevo

i Un par de claves SSH contiene una clave pública y una privada. **Azure no almacena la clave privada.** Una vez creado el recurso de clave SSH, no podrá volver a descargar la clave privada. [Más información](#)

Descargar la clave privada y crear el recurso

Volver a la creación de una máquina virtual

para acceder a esa máquina aconsejan utilizar SSH como figura en el documento, pero si tienes problemas tienes otras opciones disponibles token o bastión aquí figura como configurarlo. Configuración a través de token

<https://docs.github.com/es/authentication/keeping-your-account-and-data-secure/managing-yourpersonal-access-tokens>

Configuración a través de bastion, ESTA CONFIGURACIÓN CONSUME MUCHOS RECURSOS SOLO REALIZARLA EN CASO DE NO UTILIZAR SSH

<https://learn.microsoft.com/es-es/azure/bastion/quickstart-host-portal>

¡OJO! Tener en cuenta que dependiendo de la máquina que creamos a la hora de elegir el almacenamiento, los discos, la memoria nos va a cobrar más por favor revisa el coste y piensa que solo vamos a realizar despliegues de aplicaciones web.

2.- Una vez creada la máquina virtual Instalaremos

A modo de guía proponemos realizar los siguientes pasos:

- Es recomendable activar el firewall ufw , y permitir el OpenSSH

```
$ufw app list
```

Para activar la aplicación OpenSSH

```
$ufw allow OpenSSH
```

Para activar el firewall

```
$ufw enable
```

Para ver el estado

```
$ ufw status
```

- Apache está disponible en los repositorios de software predeterminados de Ubuntu, lo que permite instalarlo con las herramientas convencionales de administración de paquetes. Comencemos actualizando el índice de paquetes locales para que reflejen los últimos cambios anteriores:

```
$sudo apt update
```

- A continuación, instalamos el paquete apache2

```
$sudo apt install apache2
```

- Revisaremos el firewall ufw. Ufw tiene tres perfiles disponibles para Apache, utilizaremos el perfil Apache que solo abre el puerto 80 (tráfico web no cifrado)

```
$sudo ufw allow Apache
```

Se puede verificar con el comando

```
$sudo ufw status
```

- Comprobaremos el servidor web. Al final del proceso de instalación, Ubuntu 20.04 inicia Apache. El servidor web ya debería estar activo.

```
$sudo systemctl status apache2
```

Ahora nos conectamos a la máquina que acabamos de crear:

```

azureuser@ServidorUbuntu: ~$
System information as of Tue Sep 24 08:12:49 UTC 2024

System load:  0.0          Processes:           107
Usage of /:   5.0% of 28.02GB Users logged in:      0
Memory usage: 61%         IPv4 address for eth0: 10.0.0.4
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@ServidorUbuntu:~$

```

La actualizamos y la ponemos a punto e instalamos lo que nos pedían:

```

azureuser@ServidorUbuntu: ~$
Get:31 http://azure.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]
Get:32 http://azure.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:33 http://azure.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:34 http://azure.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:35 http://azure.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:36 http://azure.archive.ubuntu.com/ubuntu noble-security/main amd64 Packages [377 kB]
Get:37 http://azure.archive.ubuntu.com/ubuntu noble-security/main Translation-en [81.5 kB]
Get:38 http://azure.archive.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4516 B]
Get:39 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 Packages [270 kB]
Get:40 http://azure.archive.ubuntu.com/ubuntu noble-security/universe Translation-en [113 kB]
Get:41 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 Components [8632 B]
Get:42 http://azure.archive.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [10.1 kB]
Get:43 http://azure.archive.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [353 kB]
Get:44 http://azure.archive.ubuntu.com/ubuntu noble-security/restricted Translation-en [68.1 kB]
Get:45 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [10.9 kB]
Get:46 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2808 B]
Get:47 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get:48 http://azure.archive.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B]
Fetched 29.0 MB in 5s (5492 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
26 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ufw is already the newest version (0.36.2-6).
ufw set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 26 not upgraded.
azureuser@ServidorUbuntu:~$

```

Seguimos añadiendo el OpenSSH

```

azureuser@ServidorUbuntu:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
azureuser@ServidorUbuntu:~$ sudo ufw allow OpenSSH
Rule added
Rule added (v6)
azureuser@ServidorUbuntu:~$

```

Vemos el estado del apache2


```
azureuser@ServidorUbuntu:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-09-24 08:21:13 UTC; 36s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3198 (apache2)
    Tasks: 55 (limit: 460)
   Memory: 5.4M (peak: 6.4M)
      CPU: 38ms
   CGroup: /system.slice/apache2.service
           └─3198 /usr/sbin/apache2 -k start
             └─3201 /usr/sbin/apache2 -k start
               └─3202 /usr/sbin/apache2 -k start

Sep 24 08:21:13 ServidorUbuntu systemd[1]: Starting apache2.service - The Apache HTTP Server...
Sep 24 08:21:13 ServidorUbuntu systemd[1]: Started apache2.service - The Apache HTTP Server.
azureuser@ServidorUbuntu:~$
```

3.- Comprobaremos desde nuestro ordenador que accedemos a la página web de Apache en Ubuntu 22.04

Mira cual es la ip de la máquina azure creada y pon esa IP como pública.
Y en cualquier navegador de tu pc pon `http:// XX.XX.XX.XXX` Adjunta la pantalla donde se vea tu ip y la página web.

Agregamos la regla de seguridad de entrada:

 **Agregar regla de seguridad de entrada** ✕

Origen ⓘ

Any

Intervalos de puertos de origen * ⓘ

*

Destino ⓘ

Any

Servicio ⓘ

Custom

Intervalos de puertos de destino * ⓘ

80 ✓

Protocolo

☒ Any

☐ TCP

☐ UDP

☐ ICMPv4

Acción

☒ Permitir

☐ Denegar

Prioridad * ⓘ

100 ✓

Nombre *

ConexionHTTP ✓

Descripción

Regla para permitir la entrada a conexiones HTTP


CA

88

11

Agregar

Cancelar

 Enviar comentarios


```

azureuser@ServidorUbuntu: ~$ sudo systemctl status apache2
No containers need to be restarted.

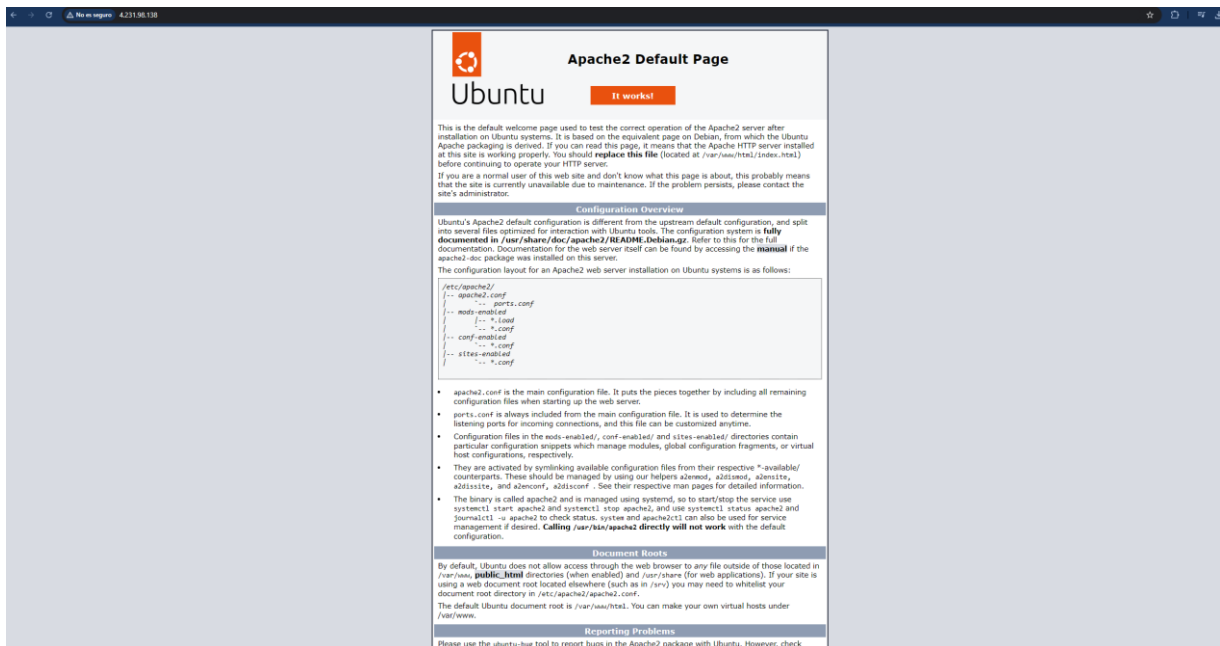
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
azureuser@ServidorUbuntu:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-09-24 08:21:13 UTC; 36s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 3198 (apache2)
    Tasks: 55 (limit: 460)
  Memory: 5.4M (peak: 6.4M)
     CPU: 38ms
    CGroup: /system.slice/apache2.service
            └─3198 /usr/sbin/apache2 -k start
              └─3201 /usr/sbin/apache2 -k start
                └─3202 /usr/sbin/apache2 -k start

Sep 24 08:21:13 ServidorUbuntu systemd[1]: Starting apache2.service - The Apache HTTP Server...
Sep 24 08:21:13 ServidorUbuntu systemd[1]: Started apache2.service - The Apache HTTP Server.
azureuser@ServidorUbuntu:~$ sudo ufw allow 80/tcp
Rule added
Rule added (v6)
azureuser@ServidorUbuntu:~$ sudo restart status apache2
sudo: restart: command not found
azureuser@ServidorUbuntu:~$ sudo systemctl start+ apache2
Unknown command verb 'start+', did you mean 'start'?
azureuser@ServidorUbuntu:~$ sudo systemctl start apache2
azureuser@ServidorUbuntu:~$

```

Como podemos ver nos funciona correctamente:



4.- Configuraremos un host virtual

Ubuntu 22.04 tiene habilitado un bloque de servidor por defecto, que está configurado para proporcionar documentos del directorio `/var/www/html`.

Sin embargo, nosotros nos vamos a crear otro directorio de publicación web.

- Crearemos un directorio dentro de `/var/www` y como nombre tu numero de clase y nombre

```
$sudo mkdir /var/www/dawxxxxxx
```

- Pondremos los permisos adecuados, por ejemplo 755 (permitir al propietario leer, escribir y ejecutar los archivos, y a la vez conceder solo permisos de lectura y ejecución a los grupos y terceros).

```
$sudo chmod -R 755 /var/www/dawxxxxxx
```

- A continuación, cree una página de ejemplo index.html utilizando nano como editor por ejemplo. Crea una página Html5, que tenga como título “Bienvenido a la página web de DAWXX” y que tenga una etiqueta h1 donde escribas “el virtual host de xxxxx está bien configurado funciona”
- En lugar de modificar el archivo de configuración predeterminado situado en /etc/apache2/sites-available/000-default.conf vamos a copiarlo y crear nuestro fichero de configuración del virtualhost que vamos a crear.
- Crea el fichero de copia en el mismo directorio donde se encuentra el de por defecto y con la misma extensión.
- Edita el fichero de configuración y cambia la directiva DocumentRoot por el directorio creado por nosotros.
- Habilitaremos el archivo con la herramienta a2ensite

```
$sudo a2ensite fichero creado.conf
```

- Deshabilitamos el sitio predeterminado definido en 000-default.conf.

```
$sudo a2dissite 000-default.conf
```

- A continuación, puedes comprobar que no hay errores de configuración en el fichero con el comando

```
$sudo apache2ctl configtest
```

- Para que tenga efecto hay que Reiniciar Apache

```
$sudo systemctl restart apache2
```

```
azureuser@ServidorUbuntu:~$ sudo systemctl start apache2
azureuser@ServidorUbuntu:~$ sudo mkdir /var/www/dawAdrian
azureuser@ServidorUbuntu:~$ sudo chmod -R 755 /var/www/dawAdrian/
azureuser@ServidorUbuntu:~$ sudo nano /var/www/dawAdrian/index.html
```

```

GNU nano 7.2 /var/www/dawAdrian/index.html *
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mi Servidor Web</title>
</head>
<body>
  <h1>Bienvenido a mi servidor web</h1>
</body>
</html>

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/_ Go To Line M-E Redo      M-G Copy

```

```

azureuser@ServidorUbuntu:~$ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/copiaConfiguracion.conf
azureuser@ServidorUbuntu:~$

```

```

GNU nano 7.2 /etc/apache2/sites-available/copiaConfiguracion.conf
# 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/daw2Adrian

# 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

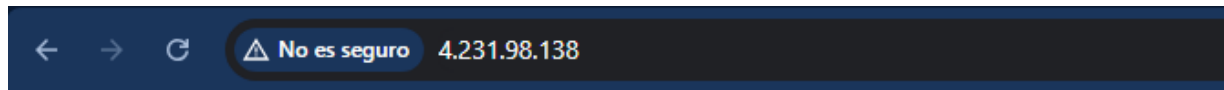
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# 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
</VirtualHost>

[Wrote 29 lines]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/_ Go To Line M-E Redo      M-G Copy

```

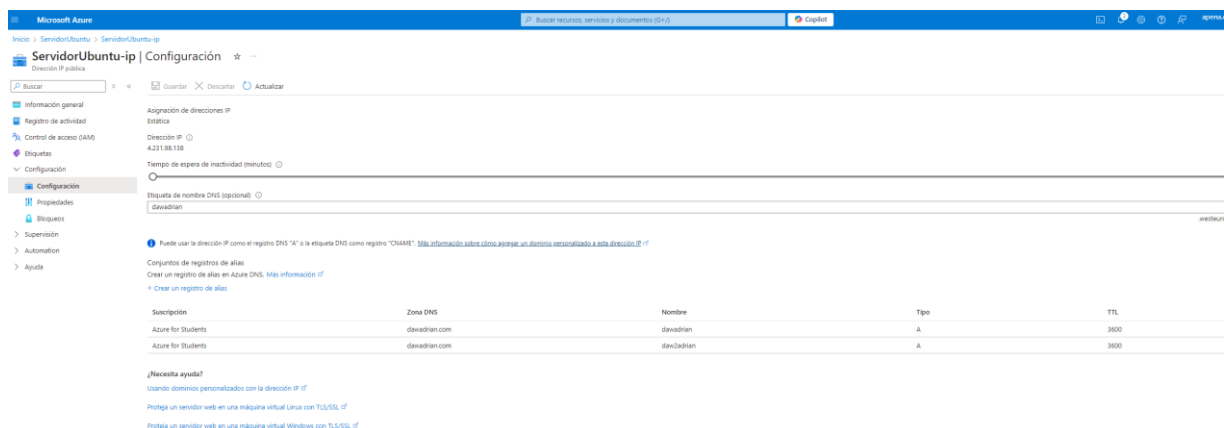
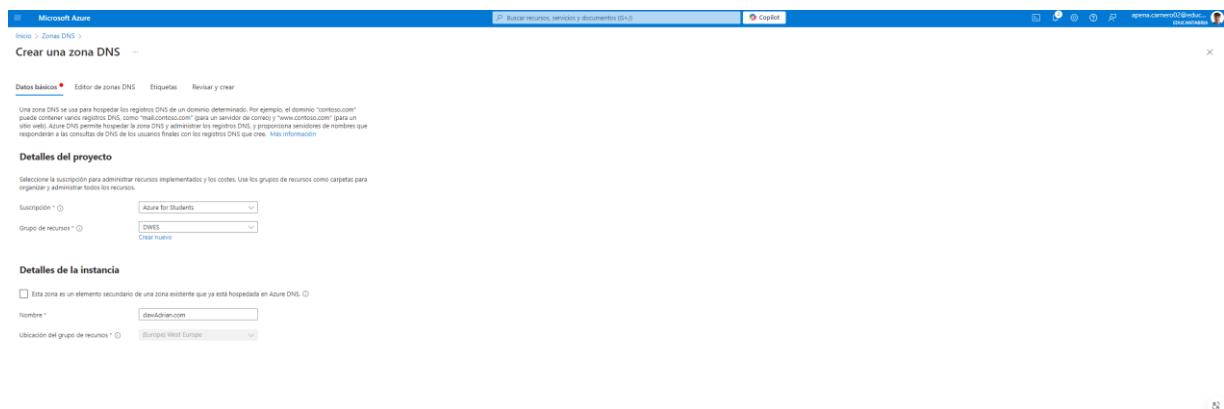
5.- Comprobaremos desde nuestro ordenador que accedemos a la página web creada por nosotros. Mira cual es la ip de la máquina azure creada. Y en cualquier navegador de tu pc pon `http:// XX.XX.XX.XXX` Adjunta la pantalla donde se vea tu ip y la página web.



Bienvenido a mi servidor web

También podemos crear un DNS en Azure y así no utilizar la dirección Pública, pon como DNS tu número de clase más nombre y prueba que accedes a través de él.

Adjunta la pantalla donde se vea tu dns y la página web





6.- Ahora instala php en la máquina Azure

Pon todos los pasos que realizas y crea una página php que nos muestre la función phpinfo() para verificar que este bien instalado. Mira la versión de php instalada.

```
azureuser@ServidorUbuntu: ~$ sudo apt update
Memory usage: 65%          IPv4 address for eth0: 10.0.0.4
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.
9 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***
Last login: Tue Sep 24 10:20:34 2024 from 95.121.82.157
azureuser@ServidorUbuntu:~$ sudo apt update
Hit:1 http://azure.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://azure.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:4 http://azure.archive.ubuntu.com/ubuntu noble-security InRelease [126 kB]
0% [Working]
Get:5 http://azure.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [530 kB]
Get:6 http://azure.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [374 kB]
Get:7 http://azure.archive.ubuntu.com/ubuntu noble-security/main amd64 Packages [377 kB]
Fetched 1533 kB in 1s (1846 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
9 packages can be upgraded. Run 'apt list --upgradable' to see them.
azureuser@ServidorUbuntu:~$ sudo apt install php libapache2-mod-php php-mysql
```

```
azureuser@ServidorUbuntu: ~$ sudo apt-get install php8.3
Setting up php (2:8.3+93ubuntu2) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for php8.3-cli (8.3.6-0ubuntu0.24.04.1) ...
Processing triggers for libapache2-mod-php8.3 (8.3.6-0ubuntu0.24.04.1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Pending kernel upgrade!
Running kernel version:
 6.8.0-1014-azure
Diagnostics:
  The currently running kernel version is not the expected kernel version 6.8.0-1015-azure.

Restarting the system to load the new kernel will not be handled automatically, so you should consider rebooting.

Restarting services...

Service restarts being deferred:
systemctl restart networkd-dispatcher.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
azureuser@ServidorUbuntu:~$ sudo nano /var/www/
dawAdrian/ html/
azureuser@ServidorUbuntu:~$ sudo nano /var/www/dawAdrian/info.php
```

```
GNU nano 7.2 /var/www/dawAdrian/info.php
<?php
phpinfo();
?>
```

[Wrote 3 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^_ Go To Line	M-E Redo	M-G Copy

The screenshot displays a web browser window showing the PHP Version 8.3.6 system information page. The page is titled "PHP Version 8.3.6" and features the PHP logo. It lists various system details, including the operating system (Linux), build system (Linux), server API (Apache 2.0 handler), and configuration file (php.ini). The page also includes a section for the Zend Engine, which lists various extensions and their versions. Below the system information, there is a "Configuration" section for the apache2handler, which lists various settings such as the server name, server administrator, and various timeouts.

On the right side of the screenshot, there is a terminal window showing system logs. The logs indicate that the system is processing triggers for php8.3-libs and php8.3-common. It also shows that the system is scanning for updates and that the currently running kernel version is not the expected kernel version 6.9. The logs end with the message "Restarting services..."

NOTA: realiza un documento de guía con las pantallas donde se muestre los comandos ejecutados además de las pantallas web solicitadas.