



# ADMINISTRACIÓN DE INFRAESTRUCTURAS Y SISTEMAS INFORMÁTICOS (AISI)

Grado en Ingeniería Informática

Grado en Ingeniería Informática

Roberto R. Expósito ([roberto.rey.exposito@udc.es](mailto:roberto.rey.exposito@udc.es))

# PRÁCTICA 0

## Vagrant



# Objetivo

3

- El propósito de esta práctica es aprender a utilizar las opciones básicas de **Vagrant**, un *software* de virtualización de código abierto para la creación, gestión y configuración automatizada de entornos virtuales portables
  - Vagrant soporta múltiples proveedores:
    - VirtualBox, VMware, KVM, Docker, Hyper-V, AWS...



HashiCorp

# Vagrant

Development Environments Made Easy

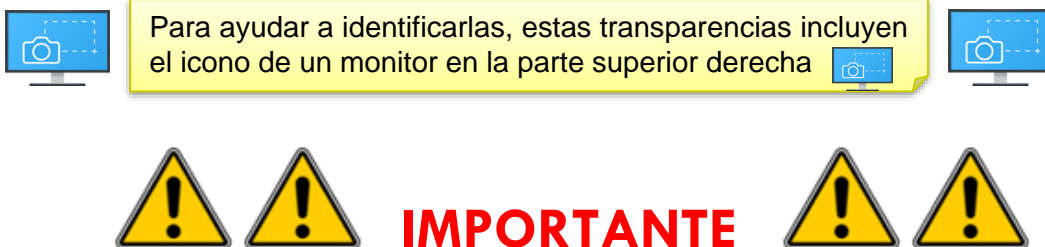
<https://www.vagrantup.com>



# Justificación de la práctica

4

- La **entrega** de la práctica consistirá en un **breve documento** en formato **PDF** que incluya las **todas capturas de pantalla** mostradas en las transparencias:
  - 15, 17, 22**



- En ocasiones, durante la práctica, se pide crear recursos con un nombre que empieza por un **prefijo** que contiene información del estudiante y del curso
- ES OBLIGATORIO** usar la siguiente nomenclatura para nombrar los recursos:  
*<iniciales del nombre y apellidos><curso>-<nombre del recurso>*
  - Ejemplo: El alumno Roberto Rey Expósito, que hace la práctica en el curso 2024/2025, utilizará el siguiente prefijo: **rre2425**
- NO RECORTES** las capturas de pantalla, **debe verse toda la información** que sea relevante para comprobar el trabajo realizado
- NO** seguir estas normas **IMPLICA UNA CALIFICACIÓN “C”** en la práctica



# Ejercicio 1: Instalación de VirtualBox/Vagrant

- Instala (o actualiza) [Oracle VirtualBox](https://www.virtualbox.org/) (versión  $\geq 7.1.6$ )
  - VirtualBox soporta CPUs **x86-64** (AMD/Intel) y SO Linux/Windows/macOS/Solaris
  - El soporte para CPUs **ARM64** (chips Apple M1/M2/M3) es **más limitado**
    - <https://www.virtualbox.org/manual/topics/Introduction.html#host-guest-matrix>
    - Entre otras cosas, el SO que ejecute las VM debe ser Linux y estar compilado para ARM64
- Instala [Vagrant](https://www.vagrantup.com/) (versión  $\geq 2.4.3$ ) y ejecuta un comando de prueba

```
rober@mordor:~$ vagrant version
Installed Version: 2.4.3
Latest Version: 2.4.3

You're running an up-to-date version of Vagrant!
rober@mordor:~$
```

- Instala estos dos *plugins* de Vagrant que usaremos en siguientes prácticas
  - *vagrant-hostmanager*: gestiona automáticamente el fichero de *hosts* de las VM (p.e. */etc/hosts* en Linux) y/o del equipo anfitrión para disponer de resolución DNS
    - <https://github.com/devopsgroup-io/vagrant-hostmanager>
  - *vagrant-vbguest*: gestiona la instalación y actualización automática de las *VirtualBox Guest Additions* en las VMs
    - <https://github.com/dotless-de/vagrant-vbguest>



# Ejercicio 1: Instalación de VirtualBox/Vagrant

- Las *VirtualBox Guest Additions* son un conjunto de controladores de dispositivo (*drivers*) y programas que se instalan en el SO que ejecuta la VM (el **SO invitado** o *guest*) para mejorar ciertos aspectos como:
  - La integración con el ratón
  - Mejor soporte y rendimiento del sistema de vídeo
  - Añadir la posibilidad de compartir directorios entre la VM (*guest*) y el equipo anfitrión (el *host*)
  - Añadir la posibilidad de copiar/pegar y arrastrar/soltar entre VM y *host*
- Puedes ver las características completas aquí:
  - <https://www.virtualbox.org/manual/UserManual.html#guestadditions>



# Ejercicio 1: Instalación de VirtualBox/Vagrant



**MUY IMPORTANTE**



- En **Windows 10/11** (Home/Pro), **VirtualBox** y otros hipervisores (p.e. VMware) **se pueden ver afectados** cuando el propio hipervisor de Microsoft (**Hyper-V**) se encuentra **activado**
  - Hyper-V "secuestra" el soporte *hardware* de virtualización de la CPU (VT-X/AMD-V) e impide que otros hipervisores lo usen adecuadamente, dando lugar a múltiples y diferentes problemas desde un bajo rendimiento, hasta incluso cuelgues/crasheos de las VM (estados "gurú")
  - En la barra de estado de una VM ejecutada con VirtualBox (parte inferior de la GUI) podemos comprobar si se está ejecutando con soporte completo de virtualización (figura de la izquierda, **icono "V"**), o si se ha detectado Hyper-V activado (figura de la derecha, **icono de la tortuga**)



- Si tienes Windows, es **MUY (MUY) RECOMENDABLE** desactivar **Hyper-V** y que tus VM se puedan ejecutar debidamente y sin errores ("sin la tortuga" en la barra de estado)
  - **Problema:** desactivar Hyper-V no siempre es un procedimiento "simple"
  - Tienes un **PDF en UDCOnline** donde describo una serie de configuraciones/comprobaciones a realizar, y el orden recomendado en el que deben hacerse
  - Puede que no sea necesario hacer todos los pasos, como ya se explica en el propio documento



## Ejercicio 2: Primeros pasos con Vagrant

- Sigue la guía [Quick Start](#) de la documentación de Vagrant (~15 minutos)
  - Sáltate las secciones "What is Vagrant", "Install Vagrant" y "Share an Environment"
- Explora los comandos de CLI más relevantes de Vagrant
  - <https://developer.hashicorp.com/vagrant/docs/cli>
- **¿Qué debes aprender?**
  - **Comandos:**
    - *init, up, ssh, box, provision, reload, status, global-status, validate, suspend, resume, halt, destroy*
  - **Conceptos:**
    - Proyecto Vagrant y la estructura/sintaxis básica del fichero *Vagrantfile*
    - Vagrant *boxes*, *providers* y [catálogo Cloud](#)
    - Carpetas sincronizadas entre VM/host y sus permisos
    - Configuración de la red virtual y redireccionamiento de puertos
    - Aprovisionamiento automatizado con *scripts* (externos, embebidos/*inline*)





## Ejercicio 2: Primeros pasos con Vagrant



### IMPORTANTE




- Las rutas de los proyectos Vagrant **NO DEBEN** contener espacios, acentos, eñes u otros caracteres “raros” (especialmente si usas **Windows**)
  - Si tienes un **antivirus** distinto al que incorpora **Windows**, asegúrate de configurarlo adecuadamente o desactivarlo de forma temporal para evitar bloqueos de conexiones al Vagrant *Cloud* durante la descarga de los *boxes*, entre otros problemas
- **NO** guardes el directorio de un proyecto Vagrant en una carpeta de tu equipo sincronizada con servicios de almacenamiento en la nube (p.e. **OneDrive**, iCloud)
- **NUNCA** elimines el directorio de un proyecto Vagrant sin eliminar antes sus VM ejecutando *vagrant destroy*
  - Tampoco es buena idea cambiarle el nombre a dicho directorio, copiarlo o moverlo a otra ruta si ya tiene VM creadas con Vagrant
- **NUNCA** inicies, pares, modifiques la configuración o elimines una VM de un proyecto Vagrant **directamente desde VirtualBox**
  - Solo usaremos VirtualBox para comprobar configuraciones de las VM creadas con Vagrant y/o en caso de que surjan problemas. Si fuese necesario realizar algún cambio directamente en VirtualBox, **se indicará de forma explícita en el enunciado**



# Ejercicio 3: Proyecto Vagrant con VirtualBox

10

- Usando la plantilla proporcionada en el [repositorio de la práctica](#), completa el *Vagrantfile* para **desplegar una VM Linux** con la siguiente configuración:

Box	<b>debian/bookworm64</b> (box disponible en el <a href="#">catálogo Cloud</a> ) Especifica en el <i>Vagrantfile</i> la última versión del box disponible en el catálogo <i>Cloud</i> y desactiva la comprobación de actualizaciones (parámetros <i>box_version</i> y <i>box_check_update</i> , respectivamente)
Hostname 	<b>Debes usar tu prefijo.</b> Ejemplo: alumno Roberto Rey Expósito usaría <b>rre2425</b> como <i>hostname</i>
CPU/Memoria	[1-2] cores / [1-2] GiB
Red	1 red privada configurada mediante DHCP 1 red privada configurada con IP estática: <b>192.168.56.10/24</b> Puerto <b>8080</b> del equipo anfitrión ( <i>host</i> ) <b>redireccionado</b> al puerto <b>80</b> de la VM
Disco	Tamaño: <b>2 GiB</b> . Formato del disco virtual: <b>VMDK</b> (ver siguiente transparencia) Disco conectado a una <b>controladora de tipo SAS de un puerto</b> que deberás añadir previamente a la VM
Aprovisionamiento	<ol style="list-style-type: none"><li>Configura el <b>shell script embebido (inline)</b> para que se ejecute únicamente en el primer despliegue (opción <i>run</i> de <i>config.vm.provision</i>). Completa el script para: <b>1)</b> instalar (<i>apt</i>) el servidor web Apache con soporte PHP (paquetes: <i>apache2</i>, <i>php</i>, <i>php-mysql</i>); <b>2)</b> iniciar el servicio <i>apache2</i> y activar su inicio automático en el arranque del SO (<i>systemctl</i>); <b>3)</b> formatear (<i>mkfs</i>) el disco SAS (<i>/dev/sdb</i>) añadido a la VM usando el sistema de ficheros <i>ext4</i>; <b>4)</b> crear el punto de montaje del disco: <b>/mnt/X</b> (sustituye <b>X</b> por tu <b>prefijo</b>)</li><li>Configura la <b>ejecución en cada despliegue de la VM del shell script externo</b> disponible en el repositorio, encargado de montar el disco SAS en la ruta creada previamente, entre otras cosas. Como único <b>argumento al script</b>, debes pasarle el <b>punto de montaje</b> del disco. <b>NO modifiques</b> el script proporcionado</li></ol>
Carpeta sincronizada	Comparte la carpeta <i>html</i> de tu equipo que contiene una página web con la siguiente ruta de la VM: <i>/var/www/html</i> . Establece permisos 755 y 644 para el directorio ( <i>dmode</i> ) y los ficheros ( <i>fmode</i> ), respectivamente (ver opción <i>mount_options</i> de <i>config.vm.synced_folder</i> )
Plugins	No modifiques la configuración para el plugin <i>vagrant-vbguest</i> . Para el plugin <i>vagrant-hostmanager</i> , activa la gestión tanto en la VM ( <i>manage_guest</i> ) como en el <i>host</i> ( <i>manage_host</i> )



# Ejercicio 3: Proyecto Vagrant con VirtualBox

## ● ¿Qué debes aprender?

### ● Opciones del *Vagrantfile*:

- *config.vm.box*, *config.vm.hostname*, *config.vm.network*, *config.vm.synced\_folder*, *config.vm.provision*, *config.vm.provider*


### ● Personalización con *VboxManage*:

- **Opción:** *{provider}.customize*
- **Comandos:** *storagectl*, *createmedium*, *storageattach*
  - Deberás usar los **tres comandos previos** para, respectivamente:
    1. Añadir a la VM una controladora de disco de tipo SAS con un puerto de conexión
    2. Crear un disco virtual con formato VMDK y el tamaño indicado en la tabla previa
    3. Conectar el disco virtual al puerto disponible en la controladora SAS
  - Ver la referencia proporcionada en la última transparencia sobre la sintaxis del comando *VBoxManage*



# Ejercicio 3: Proyecto Vagrant con VirtualBox

12

- Personaliza la página web que debe mostrar el servidor web Apache que ejecutará la VM
  - En el directorio *html*, abre el fichero *index.php* con un **editor de texto en tu equipo** para incluir tu nombre y apellidos
  - **Debes modificar únicamente la variable *name* (línea 9)** 
- Despliega la VM con Vagrant, conéctate a ella por *ssh* y comprueba:
  - Que el *hostname* de la VM es el correcto (ver *prompt* y/o ejecutar *hostname*)
  - El montaje del disco SAS que has configurado (comando *df*)
  - Que el servidor web Apache está en ejecución (comando *systemctl*)
  - La carpeta sincronizada entre *host* y VM
  - Las interfaces de red (comando *ip addr*)
  - El contenido del fichero creado por el *shell script* externo en el disco SAS



En las siguientes tres transparencias tienes ejemplos de las comprobaciones que debes realizar



# Ejercicio 3: Proyecto Vagrant con VirtualBox

Hostname correctamente configurado

```
vagrant@rre2425:~$ hostname  
rre2425
```

Carpeta sincronizada (/var/www/html). Disco SAS (/dev/sdb) montado en la ruta correcta. Sobre el montaje en la ruta /vagrant, ver comentario y figura inferior

```
vagrant@rre2425:~$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
udev            962M   0    962M   0% /dev  
tmpfs           197M  500K  197M   1% /run  
/dev/sda1       98G   1.3G   92G   2% /  
tmpfs           984M   0    984M   0% /dev/shm  
tmpfs           5.0M   0    5.0M   0% /run/lock  
tmpfs           197M   0    197M   0% /run/user/1000  
vagrant         183G   54G   129G  30% /vagrant  
var_www_html    183G   54G   129G  30% /var/www/html  
/dev/sdb        2.0G   28K   1.8G   1% /mnt/rre2425
```

Servidor web Apache en ejecución (servicio apache2)

```
vagrant@rre2425:~$  
vagrant@rre2425:~$ systemctl status apache2  
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)  
   Active: active (running) since Thu 2025-01-16 15:43:45 UTC; 39s ago  
     Docs: https://httpd.apache.org/docs/2.4/  
   Main PID: 9556 (apache2)  
     Tasks: 6 (limit: 2307)  
    Memory: 15.9M  
       CPU: 45ms  
   CGroup: /system.slice/apache2.service  
           └─9556 /usr/sbin/apache2 -k start  
             └─9558 /usr/sbin/apache2 -k start  
               └─9559 /usr/sbin/apache2 -k start  
                 └─9560 /usr/sbin/apache2 -k start  
                   └─9561 /usr/sbin/apache2 -k start  
                     └─9562 /usr/sbin/apache2 -k start
```

La VM puede acceder a la página web disponible en el equipo anfitrión

```
vagrant@rre2425:~$  
vagrant@rre2425:~$ ls -l /var/www/html/  
total 20  
drwxr-xr-x 1 vagrant vagrant 4096 Nov 18  2023 css  
-rw-r--r-- 1 vagrant vagrant 3963 Oct 19  2023 db-get-data.php  
-rw-r--r-- 1 vagrant vagrant  248 Oct 18  2023 get-info.php  
drwxr-xr-x 1 vagrant vagrant 4096 Nov 18  2023 img  
-rw-r--r-- 1 vagrant vagrant 1087 Jan 15 15:43 index.php  
vagrant@rre2425:~$
```



¿Qué vemos cuando listamos la ruta /vagrant desde la VM y por qué?. Repasa este [apartado de la Quick Start](#) de Vagrant si no entiendes el motivo

```
vagrant@rre2425:~$ ls /vagrant/  
Vagrantfile  dbserver  diskVM-SAS.vmdk  html  provisioning  
vagrant@rre2425:~$
```



# Ejercicio 3: Proyecto Vagrant con VirtualBox

## ● Interfaces de red de la VM

```
vagrant@rre2425:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
        valid_lft 85969sec preferred_lft 85969sec
    inet6 fd00::a00:27ff:fe8d:c04d/64 scope global dynamic mngtmpaddr
        valid_lft 85969sec preferred_lft 13969sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    link/ether 08:00:27:2d:4c:99 brd ff:ff:ff:ff:ff:ff
    altname enp0s8
    inet 192.168.56.4/24 brd 192.168.56.255 scope global dynamic eth1
        valid_lft 440sec preferred_lft 440sec
    inet6 fe80::a00:27ff:fe2d:4c99/64 scope link
        valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    link/ether 08:00:27:d5:14:44 brd ff:ff:ff:ff:ff:ff
    altname enp0s9
    inet 192.168.56.10/24 brd 192.168.56.255 scope global eth2
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed5:1444/64 scope link
        valid_lft forever preferred_lft forever
vagrant@rre2425:~$
```

Interfaz de red *localhost* →

Interfaz de red creada por Vagrant (por defecto) para acceso a Internet desde la VM →

Interfaz de red creada en el *Vagrantfile* y configurada por DHCP (tu IP podría ser distinta) →

Interfaz de red creada en el *Vagrantfile* y configurada con IP estática (tu IP debe ser la misma) →



Comprueba en VirtualBox el tipo de adaptador de red que se ha creado para todas las interfaces




# Ejercicio 3: Proyecto Vagrant con VirtualBox



- Contenido del disco SAS montado en `/mnt/X`

```
vagrant@rre2425:~$ ls -l /mnt/rre2425/
total 20
-rw-r--r-- 1 root root 1754 Jan 16 16:00 info
drwx----- 2 vagrant vagrant 16384 Jan 16 16:00 lost+found
vagrant@rre2425:~$ cat /mnt/rre2425/info
Linux rre2425 6.1.0-27-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.115-1 (2024-11-01) x:
uid=0(root) gid=0(root) groups=0(root)
Thu Jan 16 16:00:32 UTC 2025
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda 8:0 0 100G 0 disk
└─sda1 8:1 0 100G 0 part /
sdb 8:16 0 2G 0 disk /mnt/rre2425
Debian 12.8
H/W path Device Class Description
=====
/0/100/1/1 storage PnP device PNP0700
/0/100/d scsi0 storage 82801HM/HEM (ICH8M/ICH8M-E) SATA Controller
/0/100/16 scsi1 storage SAS1068 PCI-X Fusion-MPT SAS
<?php header("Cache-Control: no-cache, must-revalidate");?>
<!DOCTYPE html>
<html>
<head>
<title>GEI AISI</title>
<link href="css/bootstrap.min.css" rel="stylesheet">
<script type="text/javascript">
function init() {
    var name = "Roberto Rey Expósito";
    document.getElementById("myName").innerHTML = name;
}
function getURL() {
    document.write(window.location.href);
}
function getTime() {
    document.getElementById("current_date").innerHTML = Date();
}
</script>
</head>
<body onload="init()">
<div class="container" style="text-align: center;">
<div class="jumbotron">
<p></p>
<p><u>GEI AISI: 2024/2025</u></p>
<p></p>
<p>Página web de <span id="myName"></span></p>
<p><script>getURL();</script></p>
<h4><div id="current_date"><script>getTime();</script></div></h4>
<?php include('get-info.php');?></div>
</div>
</body>
</html>
```



El fichero *info* es creado por el *shell script* proporcionado en el repositorio de la práctica y que se ejecuta como parte del aprovisionamiento de la VM. Entre otras cosas, este *script* muestra el contenido del fichero `/var/www/html/index.php`



# Ejercicio 3: Proyecto Vagrant con VirtualBox

- Comprueba el correcto funcionamiento del servidor web Apache:
  - Desde el navegador de tu *host*, accede a la URL: <http://localhost:8080>
  - Desde la *VM*, usa *curl* para hacer una petición GET a la URL: <http://localhost>
    - En ambos casos, debes ver el contenido personalizado del fichero *index.php*
- Pruebas adicionales que puedes hacer desde el navegador de tu *host*
  - <http://localhost>
  - <http://192.168.56.10:8080>
  - <http://192.168.56.10>
  - <http://X:8080>
  - <http://X>



Trata de razonar si el acceso debería funcionar en cada caso de prueba



Siempre que te encuentres con una "X", deberás sustituirla por tu **prefijo**



¿Qué ocurre si haces un *ping* desde tu *host* a X? En caso de obtener respuesta, ¿qué dirección IP responde y por qué?



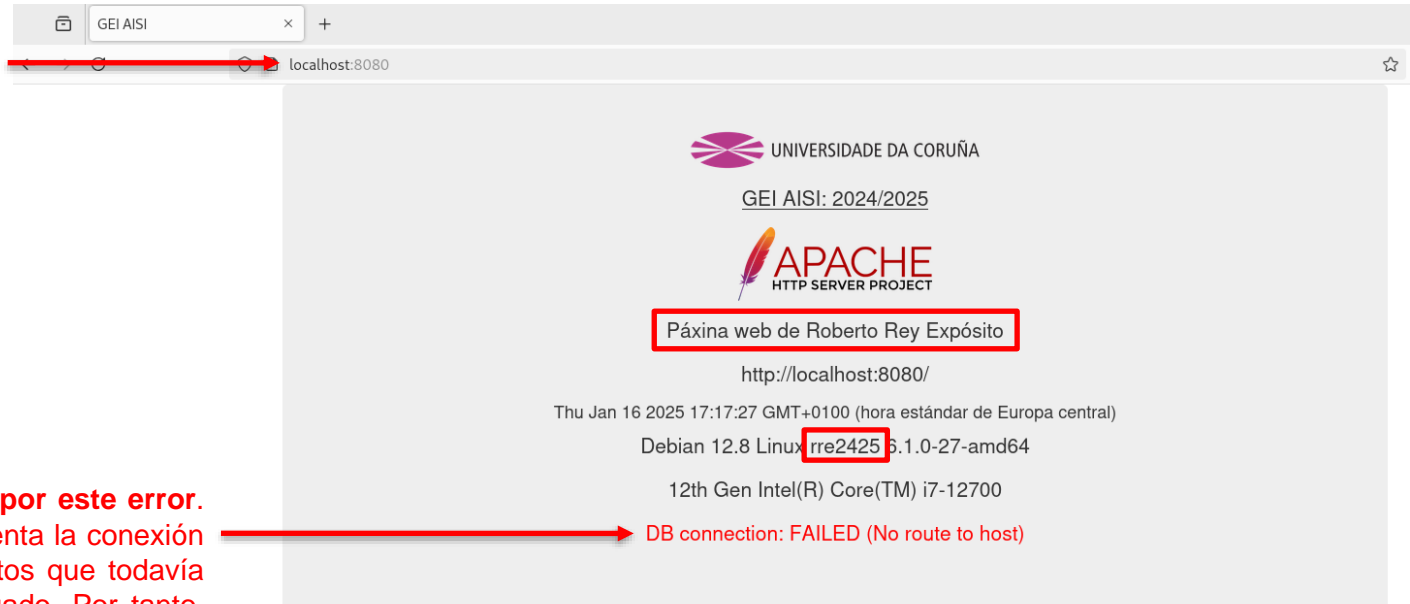




# Ejercicio 3: Proyecto Vagrant con VirtualBox



Acceso al servidor Apache desde el navegador del equipo anfitrión (host)



No te preocupes por este error. La página web intenta la conexión a una base de datos que todavía no hemos desplegado. Por tanto, la conexión fallará



Las capturas de pantalla que incluyas en tu PDF deben mostrar la misma información y/o comandos que los ejemplos que se proporcionan en el enunciado de todas las prácticas de la asignatura, excepto por cuestiones relacionadas con el nombre de los recursos donde deberás usar el que corresponda en cada caso. **NO es necesario** que encuadres en rojo tal y como se hace en los ejemplos proporcionados. Esto se hace para resaltar determinada información que **deberás comprobar en tus propias capturas**





# Ejercicio 3: Proyecto Vagrant con VirtualBox

```
vagrant@rre2425:~$ curl http://localhost  Acceso al servidor Apache  
desde la propia VM usando curl  
<!DOCTYPE html>  
<html>  
<head>  
  <title>GEI AISI</title>  
  <link href="css/bootstrap.min.css" rel="stylesheet">  
  <script type="text/javascript">  
    function init() {  
      var name = "Roberto Rey Expósito";  
      document.getElementById("myName").innerHTML = name;  
    }  
    function getURL() {  
      document.write(window.location.href);  
    }  
    function getTime() {  
      document.getElementById("current_date").innerHTML = Date();  
    }  
  </script>  
</head>  
<body onload="init()">  
  <div class="container" style="text-align: center;"><div class="jumbotron">  
    <p></p>  
    <p><u>GEI AISI: 2024/2025</u></p>  
    <p></img></p>  
    <p>Página web de <span id="myName"></span></p>  
    <p><script>getURL();</script></p>  
    <h4><div id="current_date"><script>getTime();</script></div></h4>  
    <p>Debian 12.8 Linux rre2425 6.1.0-27-amd64</p><p> 12th Gen Intel(R) Core(TM) i7-12700</p>  
<p style='color: red;'>DB connection: FAILED (No route to host)</p></div></div>  
</body>  
</html>  
vagrant@rre2425:~$ _
```



# Ejercicio 3: Proyecto Vagrant con VirtualBox

## ● Abre VirtualBox para ver la configuración final de la VM

Fíjate que el nombre que tiene la VM en VirtualBox no tiene por qué ser el mismo que el *hostname* configurado en el SO que ejecuta dicha VM (aunque se podrían configurar para que coincidan). El nombre de la VM en VirtualBox se configura en el *Vagrantfile* con un parámetro distinto al *hostname* (revisa el *Vagrantfile* para ver entender cómo se configuran ambos)

El *box* usado incluye una controladora SATA en la que se encuentra conectado el disco (100 GB) donde está instalado el SO invitado (Debian) que ejecuta esta VM

En esta práctica hemos añadido a la VM una nueva controladora de tipo SAS y conectado a ella un disco de 2 GB

Fíjate en la configuración de la red virtual. Este [enlace](#) puede ayudarte a entender dicha configuración

Tenemos dos carpetas sincronizadas entre la VM y el *host*, aunque **solo hemos configurado una de forma explícita** en el *Vagrantfile*. **¿Por qué hay dos?.** Haz click en *Carpetas compartidas* para ver más detalles sobre ambas

The screenshot shows the VirtualBox settings for a VM named 'AISI-P0-rre2425'. The settings are organized into several tabs, with the following configurations highlighted by red boxes and arrows:

- General:** The 'Nombre' (Name) is 'AISI-P0-rre2425'.
- Sistema:** The 'Memoria base' (Base memory) is '2048 MB' and the 'Procesadores' (Processors) is '2'.
- Almacenamiento:** The 'Controlador' (Controller) is 'SATA Controller' and the 'Puerto SATA 0' (SATA 0 Port) is 'box.vmdk (Normal, 100,00 GB)'. The 'Controlador' (Controller) is 'SAS Controller' and the 'Puerto SAS 0' (SAS 0 Port) is 'diskVM-SAS.vmdk (Normal, 2,00 GB)'.
- Red:** The 'Adaptador 1' (Adapter 1) is 'Intel PRO/1000 MT Desktop (NAT)', 'Adaptador 2' (Adapter 2) is 'Intel PRO/1000 MT Desktop (Adaptador solo anfitrión, «vboxnet0»)', and 'Adaptador 3' (Adapter 3) is 'Intel PRO/1000 MT Desktop (Adaptador solo anfitrión, «vboxnet0»)'.
- Carpetas compartidas:** The 'Carpetas compartidas' (Shared folders) are '2'.



# Ejercicio 3: Proyecto Vagrant con VirtualBox



Prueba a detener y/o suspender la VM con Vagrant y volver a desplegarla. Comprueba que el servidor web sigue funcionando. Fíjate también en los aprovisionamientos que se ejecutan en cada caso e investiga como podemos **forzar el aprovisionamiento** de una VM ya desplegada

```
[rober@oceania p0-vagrant]$ vagrant status  
Current machine states:
```

```
default                running (virtualbox)
```

The VM is running. To stop this VM, you can run `vagrant halt` to shut it down forcefully, or you can run `vagrant suspend` to simply suspend the virtual machine. In either case, to restart it again, simply run `vagrant up`.

```
[rober@oceania p0-vagrant]$ vagrant suspend  
==> default: Saving VM state and suspending execution...  
[rober@oceania p0-vagrant]$ vagrant status  
Current machine states:
```

```
default                saved (virtualbox)
```

To resume this VM, simply run `vagrant up`.

```
[rober@oceania p0-vagrant]$ vagrant up  
Bringing machine 'default' up with 'virtualbox' provider...  
==> default: Resuming suspended VM...  
==> default: Booting VM...  
==> default: Waiting for machine to boot. This may take a few minutes...  
default: SSH address: 127.0.0.1:2222  
default: SSH username: vagrant  
default: SSH auth method: private key  
==> default: Machine booted and ready!  
==> default: [vagrant-hostmanager:guests] Updating hosts file on active guest virtual machines...  
==> default: [vagrant-hostmanager:host] Updating hosts file on your workstation (password may be required)...  
==> default: Machine already provisioned. Run `vagrant provision` or use the `--provision`  
==> default: flag to force provisioning. Provisioners marked to run always will still run.  
==> default: Running provisioner: shell...  
default: Running: /tmp/vagrant-shell120240117-1198397-sfifwl.sh
```



¿Qué diferencia hay entre suspender y detener una VM con Vagrant?



# Ejercicio 4: Despliegue de la BBDD

21

- Usando el proyecto Vagrant proporcionado en la carpeta *dbserver* de la práctica, despliega la VM que ejecutará una base de datos [MariaDB](#)
  - El aprovisionamiento de esta VM se encarga de instalar y configurar MariaDB, creando automáticamente una BBDD usando tu **prefijo** como nombre
  - En el directorio *html*, abre el fichero *db-get-data.php* con un **editor de texto en tu equipo** y modifica la variable *database* (línea 2) para configurar el nombre de tu BBDD **sustituyendo X por tu prefijo**
- Una vez desplegada la VM *dbserver*, conéctate a ella por *ssh* y comprueba que la BBDD está en ejecución (comando *systemctl*, servicio *mariadb*)

```
vagrant@dbserver:~$ systemctl status mariadb
● mariadb.service - MariaDB 10.11.6 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-01-16 16:44:41 UTC; 1min 29s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
   Process: 3017 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysql
   Process: 3018 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION
   Process: 3020 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR= ||
   Process: 3094 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION
   Process: 3096 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
 Main PID: 3085 (mariabdd)
    Status: "Taking your SQL requests now..."
     Tasks: 9 (limit: 1100)
    Memory: 150.5M
       CPU: 254ms
    CGroup: /system.slice/mariadb.service
            └─3085 /usr/sbin/mariabdd

vagrant@dbserver:~$
```



# Ejercicio 4: Despliegue de la BBDD



- Desde el navegador de tu equipo, accede de nuevo al servidor Apache

Si quieres, puedes probar a añadir un nuevo contacto o modificar uno existente

Last name	First name	Phone	Email	Admin
Doe	Jane	010-110-1101	janed@someotheraddress.org	<a href="#">Add Contact</a>
Johnson	Roberto	123-456-7890	roberto@someaddress.com	<a href="#">Edit</a> <a href="#">Remove</a>



Es posible realizar **despliegues de múltiples VM** en el mismo proyecto Vagrant (es decir, usando un único fichero *Vagrantfile*). En esta primera práctica hemos definido ambas VM en proyectos Vagrant separados, por simplicidad. En prácticas posteriores veremos ejemplos de despliegues multi-VM usando un único *Vagrantfile*



# Referencias

23

- Configuración de carpetas sincronizadas
  - [https://developer.hashicorp.com/vagrant/docs/synced-folders/basic\\_usage](https://developer.hashicorp.com/vagrant/docs/synced-folders/basic_usage)
- Configuración de la red
  - <https://developer.hashicorp.com/vagrant/docs/networking>
- Configuración del proveedor VirtualBox
  - <https://developer.hashicorp.com/vagrant/docs/providers/virtualbox/configuration>
- Configuración del aprovisionamiento
  - <https://developer.hashicorp.com/vagrant/docs/provisioning>
- Aprovisionamiento con *shell scripts*
  - <https://developer.hashicorp.com/vagrant/docs/provisioning/shell>
- Comando *VBoxManage* de VirtualBox
  - <https://www.virtualbox.org/manual/ch08.html>