

Se pide realizar los siguientes ejercicios:

1. Crea una máquina Ubuntu Server 22 de nombre AnsibleXXX donde XXX son tus iniciales. La máquina dispondrá de: (2 ptos.)

a) 2 GB de RAM

b) 2 CPU's

c) Una interfaz de red sólo anfitrión en la red 192.168.1XX.0/24 y con ip 192.168.1XX.100. (La XX será el número de tu PC de clase, único por cada puesto suponiendo que van del 1 al 20)

d) El hostname de la máquina será SerAnsXXX

e) Carpeta compartida tal que el directorio app de la carpeta donde tenemos nuestro proyecto vagrant debe montarse en la carpeta /source de la máquina Ubuntu.

f) Además se actualizará el gestor de paquetes de apt y se instalará el servicio web Apache

```
E: > Vagrant > AnsibleMFGH > Vagrantfile
1  Vagrant.configure("2") do |config|
2      config.vm.define "AnsibleMFGH" do |ansible|
3          ansible.vm.box = "bento/ubuntu-22.04"
4          ansible.vm.hostname = "SerAnsMFGH"
5          ansible.vm.network "private_network", ip: "192.168.106.100"
6          ansible.vm.provider "virtualbox" do |vb|
7              vb.memory = "2048" # 2 GB de RAM
8              vb.cpus = 2        # 2 CPU
9          end
10
11         ansible.vm.synced_folder "./app", "/source"
12
13         ansible.vm.provision "shell", inline: <<-SHELL
14             apt-get update
15             apt-get install -y apache2
16         SHELL
17     end
18 end
```

```

PS E:\Vagrant\AnsibleMFGH> vagrant up
Bringing machine 'AnsibleMFGH' up with 'virtualbox' provider...
==> AnsibleMFGH: Importing base box 'bento/ubuntu-22.04'...
==> AnsibleMFGH: Matching MAC address for NAT networking...
==> AnsibleMFGH: Checking if box 'bento/ubuntu-22.04' version '202407.23.0' is up to date...
==> AnsibleMFGH: Setting the name of the VM: AnsibleMFGH_AnsibleMFGH_1737306061404_67522
Vagrant is currently configured to create VirtualBox synced folders with
the 'SharedFoldersEnableSymlinksCreate' option enabled. If the Vagrant
guest is not trusted, you may want to disable this option. For more
information on this option, please refer to the VirtualBox manual:

  https://www.virtualbox.org/manual/ch04.html#sharedfolders

This option can be disabled globally with an environment variable:

  VAGRANT_DISABLE_VBOXSYMLINKCREATE=1

or on a per folder basis within the Vagrantfile:

  config.vm.synced_folder '/host/path', '/guest/path', SharedFoldersEnableSymlinksCreate: false
==> AnsibleMFGH: Clearing any previously set network interfaces...
==> AnsibleMFGH: Preparing network interfaces based on configuration...
  AnsibleMFGH: Adapter 1: nat
  AnsibleMFGH: Adapter 2: hostonly
==> AnsibleMFGH: Forwarding ports...
  AnsibleMFGH: 22 (guest) => 2222 (host) (adapter 1)
==> AnsibleMFGH: Running 'pre-boot' VM customizations...
==> AnsibleMFGH: Booting VM...
==> AnsibleMFGH: Waiting for machine to boot. This may take a few minutes...
  AnsibleMFGH: SSH address: 127.0.0.1:2222
  AnsibleMFGH: SSH username: vagrant
  AnsibleMFGH: SSH auth method: private key
  AnsibleMFGH: Warning: Connection reset. Retrying...
  AnsibleMFGH:
  AnsibleMFGH: Vagrant insecure key detected. Vagrant will automatically replace
  AnsibleMFGH: this with a newly generated keypair for better security.
  AnsibleMFGH:
  AnsibleMFGH: Inserting generated public key within guest...
  AnsibleMFGH: Removing insecure key from the guest if it's present...
  AnsibleMFGH: Key inserted! Disconnecting and reconnecting using new SSH key...
==> AnsibleMFGH: Machine booted and ready!
==> AnsibleMFGH: Checking for guest additions in VM...
==> AnsibleMFGH: Setting hostname...
==> AnsibleMFGH: Configuring and enabling network interfaces...
==> AnsibleMFGH: Mounting shared folders...
  AnsibleMFGH: E:/Vagrant/AnsibleMFGH/app => /source
  AnsibleMFGH: E:/Vagrant/AnsibleMFGH => /vagrant
==> AnsibleMFGH: Running provisioner: shell...
  AnsibleMFGH: Running: inline script

```



Herramientas



AnsibleMFGH_AnsibleMFGH_1737306061404_67522

Corriendo

```

PS E:\Vagrant\AnsibleMFGH> vagrant ssh
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-116-generic x86_64)

```

```

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

```

System information as of Sun Jan 19 05:03:49 PM UTC 2025

```

System load:  0.39           Processes:            162
Usage of /:   12.5% of 30.34GB Users logged in:        0
Memory usage: 12%           IPv4 address for eth0: 10.0.2.15
Swap usage:   0%

```

This system is built by the Bento project by Chef Software
More information can be found at <https://github.com/chef/bento>

Use of this system is acceptance of the OS vendor EULA and License Agreements.

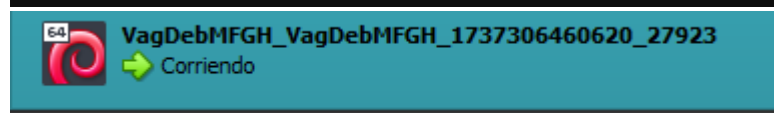
vagrant@SerAnsMFGH:~\$

2. Crea una máquina Debian 10 de nombre VagDebXXX donde XXX son tus iniciales. La máquina dispondrá de: (2 ptos.)

- a) 4 GB de RAM
- b) 2 CPU's
- c) Una interfaz de red sólo anfitrión en la red 192.168.1XX.0/24 y con ip 192.168.1XX.150.
- d) Se debe inhabilitar el uso de la carpeta compartida por defecto /vagrant
- e) El hostname de la máquina será VdebXXX

```
E: > Vagrant > VagDebMFGH > Vagrantfile
1  Vagrant.configure("2") do |config|
2      config.vm.define "VagDebMFGH" do |debian|
3          debian.vm.box = "bento/debian-10"
4          debian.vm.hostname = "VdebMFGH"
5          debian.vm.network "private_network", ip: "192.168.106.150"
6          debian.vm.provider "virtualbox" do |vb|
7              vb.memory = "4096" # 4 GB de RAM
8              vb.cpus = 2        # 2 CPU
9          end
10
11      debian.vm.synced_folder ".", "/vagrant", disabled: true
12  end
13  end

PS E:\Vagrant\VagDebMFGH> vagrant up
Bringing machine 'VagDebMFGH' up with 'virtualbox' provider...
==> VagDebMFGH: Box 'bento/debian-10' could not be found. Attempting to find and install...
VagDebMFGH: Box Provider: virtualbox
VagDebMFGH: Box Version: >= 0
==> VagDebMFGH: Loading metadata for box 'bento/debian-10'
VagDebMFGH: URL: https://vagrantcloud.com/api/v2/vagrant/bento/debian-10
==> VagDebMFGH: Adding box 'bento/debian-10' (v202309.08.0) for provider: virtualbox
VagDebMFGH: Downloading: https://vagrantcloud.com/bento/boxes/debian-10/versions/202309.08.0/providers/virtualbox/unknown/vagrant.box
VagDebMFGH:
==> VagDebMFGH: Successfully added box 'bento/debian-10' (v202309.08.0) for 'virtualbox'!
==> VagDebMFGH: Importing base box 'bento/debian-10'...
==> VagDebMFGH: Matching MAC address for NAT networking...
==> VagDebMFGH: Checking if box 'bento/debian-10' version '202309.08.0' is up to date...
==> VagDebMFGH: Setting the name of the VM: VagDebMFGH_VagDebMFGH_1737306460620_27923
==> VagDebMFGH: Fixed port collision for 22 => 2222. Now on port 2200.
==> VagDebMFGH: Clearing any previously set network interfaces...
==> VagDebMFGH: Preparing network interfaces based on configuration...
VagDebMFGH: Adapter 1: nat
VagDebMFGH: Adapter 2: hostonly
==> VagDebMFGH: Forwarding ports...
VagDebMFGH: 22 (guest) => 2200 (host) (adapter 1)
==> VagDebMFGH: Running 'pre-boot' VM customizations...
==> VagDebMFGH: Booting VM...
==> VagDebMFGH: Waiting for machine to boot. This may take a few minutes...
VagDebMFGH: SSH address: 127.0.0.1:2200
VagDebMFGH: SSH username: vagrant
VagDebMFGH: SSH auth method: private key
```



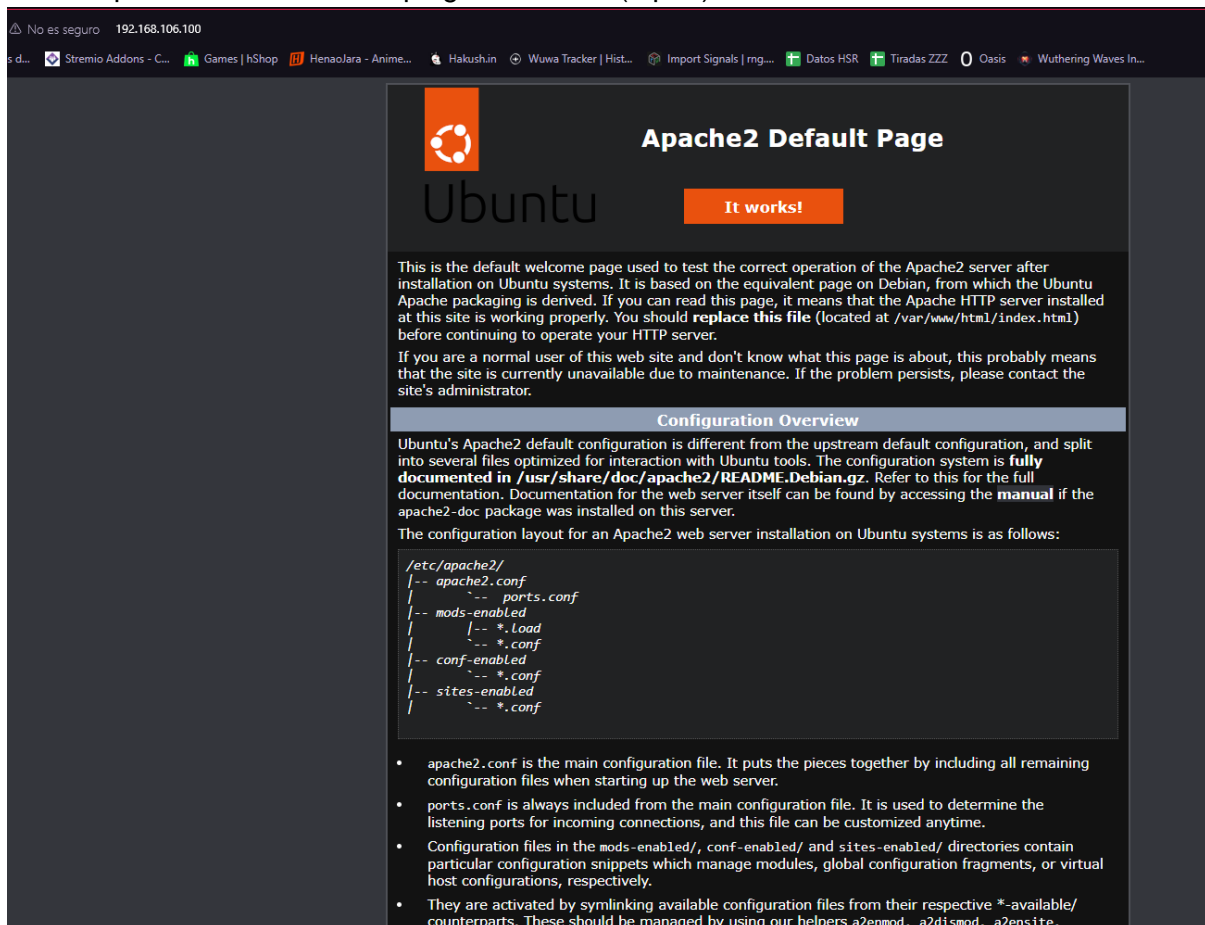
```
PS E:\Vagrant\VagDebMFGH> vagrant ssh
Linux VdebMFGH 4.19.0-25-amd64 #1 SMP Debian 4.19.289-2 (2023-08-08) x86_64

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
vagrant@VdebMFGH:~$
```

3. Prueba el correcto funcionamiento del servicio Apache en Ubuntu. Comprueba que las dos máquinas virtuales hacen ping entre ellas. (1 pto.)



SerAns hace ping con VagDeb

```
vagrant@SerAnsMFGH:~$ ping -c 4 192.168.106.150
PING 192.168.106.150 (192.168.106.150) 56(84) bytes of data.
64 bytes from 192.168.106.150: icmp_seq=1 ttl=64 time=1.02 ms
64 bytes from 192.168.106.150: icmp_seq=2 ttl=64 time=0.620 ms
64 bytes from 192.168.106.150: icmp_seq=3 ttl=64 time=0.833 ms
64 bytes from 192.168.106.150: icmp_seq=4 ttl=64 time=0.625 ms

--- 192.168.106.150 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 0.620/0.774/1.020/0.165 ms
```

VagDeb hace ping con SerAns

```
vagrant@VdebMFGH:~$ ping -c 4 192.168.106.100
PING 192.168.106.100 (192.168.106.100) 56(84) bytes of data.
64 bytes from 192.168.106.100: icmp_seq=1 ttl=64 time=1.17 ms
64 bytes from 192.168.106.100: icmp_seq=2 ttl=64 time=0.574 ms
64 bytes from 192.168.106.100: icmp_seq=3 ttl=64 time=0.578 ms
64 bytes from 192.168.106.100: icmp_seq=4 ttl=64 time=1.36 ms

--- 192.168.106.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 235ms
rtt min/avg/max/mdev = 0.574/0.921/1.364/0.354 ms
```

4. Construye el siguiente entorno multimáquina con Vagrant: (4 pts.)

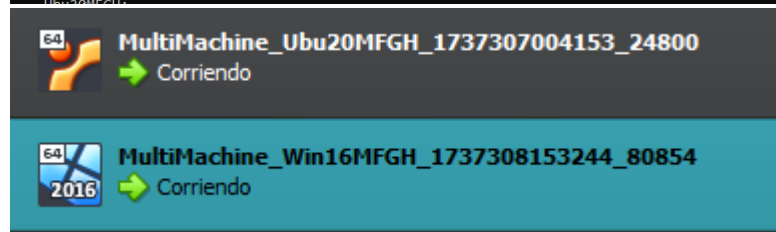
- a) Máquina servidora con Ubuntu Server 20, 1GB RAM, 1 CPU, interfaz de red en la 172.30.0.0/16 ip 172.30.0.200 y hostname Ubu20MFGH, con instalación de apache
- b) Máquina cliente con Windows Server 2016, 2 GB RAM, 1 CPU, interfaz de red en la 172.30.0.0/16 ip 172.30.0.250 y hostname Win16MFGH

```
Vagrantfile X
E: > Vagrant > MultiMachine > Vagrantfile
1  Vagrant.configure("2") do |config|
2      config.vm.define "Ubu20MFGH" do |ubuntu|
3          ubuntu.vm.box = "bento/ubuntu-20.04"
4          ubuntu.vm.hostname = "Ubu20MFGH"
5          ubuntu.vm.network "private_network", ip: "172.30.0.200"
6          ubuntu.vm.provider "virtualbox" do |vb|
7              vb.memory = "1024" # 1 GB de RAM
8              vb.cpus = 1        # 1 CPU
9          end
10
11         ubuntu.vm.provision "shell", inline: <<-SHELL
12             apt-get update
13             apt-get install -y apache2
14             SHELL
15         end
16
17         config.vm.define "Win16MFGH" do |windows|
18             windows.vm.box = "peru/windows-server-2016-standard-x64-eval"
19             windows.vm.hostname = "Win16MFGH"
20             windows.vm.network "private_network", ip: "172.30.0.250"
21             windows.vm.provider "virtualbox" do |vb|
22                 vb.memory = "2048" # 2 GB de RAM
23                 vb.cpus = 1        # 1 CPU
24             end
25         end
26     end
27
```

```

PS E:\Vagrant\MultiMachine> vagrant up
Bringing machine 'Ubu20MFGH' up with 'virtualbox' provider...
Bringing machine 'Win16MFGH' up with 'virtualbox' provider...
==> Ubu20MFGH: Box 'bento/ubuntu-20.04' could not be found. Attempting to find and install...
Ubu20MFGH: Box Provider: virtualbox
Ubu20MFGH: Box Version: >= 0
==> Ubu20MFGH: Loading metadata for box 'bento/ubuntu-20.04'
Ubu20MFGH: URL: https://vagrantcloud.com/api/v2/vagrant/bento/ubuntu-20.04
==> Ubu20MFGH: Adding box 'bento/ubuntu-20.04' (v202407.23.0) for provider: virtualbox (amd64)
Ubu20MFGH: Downloading: https://vagrantcloud.com/bento/boxes/ubuntu-20.04/versions/202407.23.0/providers/virtualbox/amd64/vagrant.box
Ubu20MFGH:
==> Ubu20MFGH: Successfully added box 'bento/ubuntu-20.04' (v202407.23.0) for 'virtualbox (amd64)'!
==> Ubu20MFGH: Importing base box 'bento/ubuntu-20.04'...
==> Ubu20MFGH: Matching MAC address for NAT networking...
==> Ubu20MFGH: Checking if box 'bento/ubuntu-20.04' version '202407.23.0' is up to date...
==> Ubu20MFGH: Setting the name of the VM: MultiMachine_Ubu20MFGH_1737307004153_24800
==> Ubu20MFGH: Fixed port collision for 22 => 2222. Now on port 2201.
==> Ubu20MFGH: Clearing any previously set network interfaces...
==> Ubu20MFGH: Preparing network interfaces based on configuration...
Ubu20MFGH: Adapter 1: nat
Ubu20MFGH: Adapter 2: hostonly
==> Ubu20MFGH: Forwarding ports...
Ubu20MFGH: 22 (guest) => 2201 (host) (adapter 1)
==> Ubu20MFGH: Running 'pre-boot' VM customizations...
==> Ubu20MFGH: Booting VM...
==> Ubu20MFGH: Waiting for machine to boot. This may take a few minutes...
Ubu20MFGH: SSH address: 127.0.0.1:2201
Ubu20MFGH: SSH username: vagrant
Ubu20MFGH: SSH auth method: private key
Ubu20MFGH:

```



```

PS E:\Vagrant\MultiMachine> vagrant ssh Ubu20MFGH
>>
Vagrant failed to initialize at a very early stage:

There is a syntax error in the following Vagrantfile. The syntax error
message is reproduced below for convenience:

```

```

E:/Vagrant/MultiMachine/Vagrantfile:1: invalid multibyte char (UTF-8)

```

```

PS E:\Vagrant\MultiMachine> vagrant ssh Ubu20MFGH

```

```

>>

```

```

Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-189-generic x86_64)

```

```

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

```

```

System information as of Sun 19 Jan 2025 05:47:23 PM UTC

```

```

System load:  0.0      Processes:            138
Usage of /:   12.6% of 30.34GB Users logged in:        0
Memory usage: 22%      IPv4 address for eth0: 10.0.2.15
Swap usage:   0%

```

```

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento

```

```

Use of this system is acceptance of the OS vendor EULA and License Agreements.

```

```

vagrant@Ubu20MFGH:~$

```

MultiMachine_Win16MFGH_1737308153244_80854 [Corriendo] - Oracle VM VirtualBox

Máquina Ver Entrada Dispositivos Ayuda



Recycle Bin



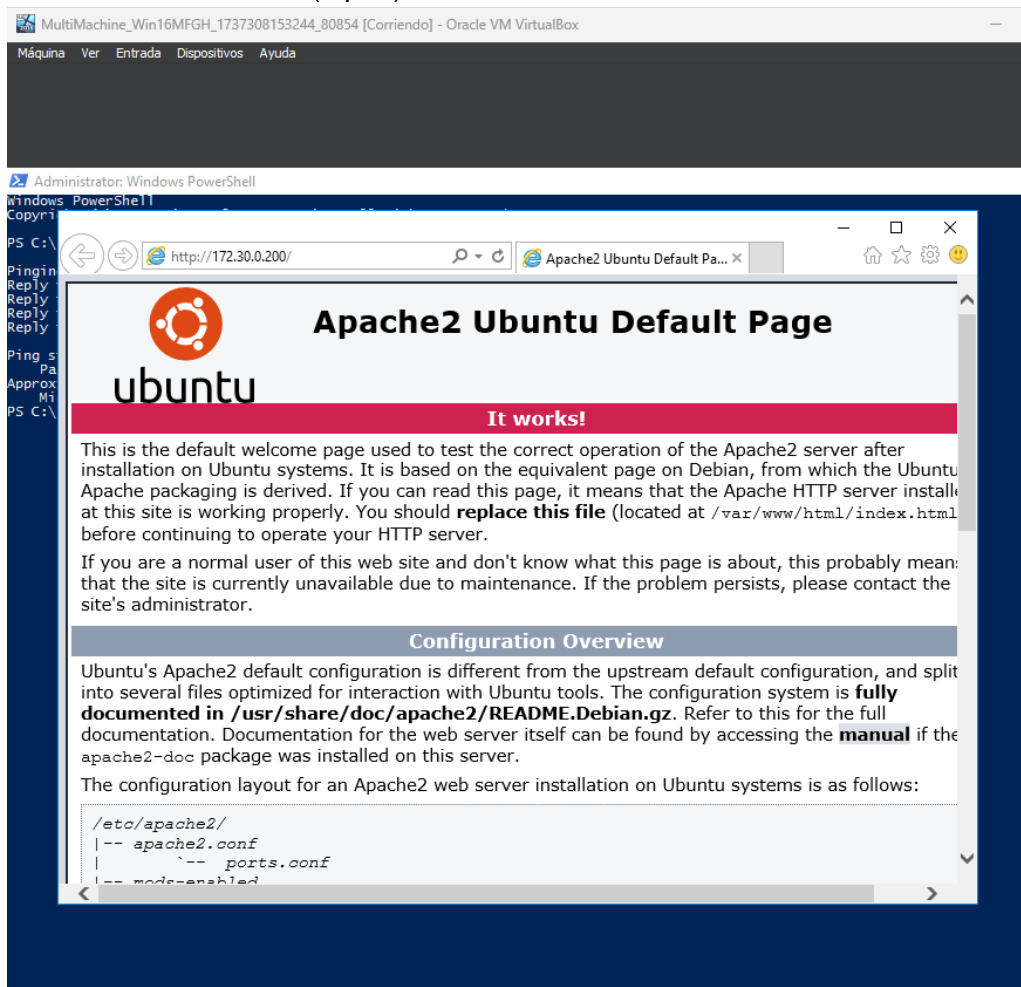

```
MultiMachine_Win16MFGH_1737308153244_80854 [Corriendo] - Oracle VM VirtualBox
Máquina Ver Entrada Dispositivos Ayuda

Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\vagrant> ping 172.30.0.200

Pinging 172.30.0.200 with 32 bytes of data:
Reply from 172.30.0.200: bytes=32 time=1ms TTL=64
Reply from 172.30.0.200: bytes=32 time<1ms TTL=64
Reply from 172.30.0.200: bytes=32 time<1ms TTL=64
Reply_
```

5. Prueba el correcto funcionamiento del servicio Apache en Ubuntu 20 desde la máquina Windows Server 2016. (1 pto.)



MultiMachine_Win16MFGH_1737308153244_80854 [Corriendo] - Oracle VM VirtualBox

Máquina Ver Entrada Dispositivos Ayuda

Administrator: Windows PowerShell

Windows PowerShell

Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\> http://172.30.0.200/ Apache2 Ubuntu Default Pa...

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|   |-- mods-enabled
```