



IVANA SÁNCHEZ PÉREZ

2º ASIR

INTRODUCCIÓN

Vagrant es un software de código abierto que nos permite crear y mantener entornos de desarrollo portables, puede trabajar con VMware, VirtualBox, Hyper-V, KVM, AWS e incluso también con contenedores de Docker, por tanto, es ideal para simplificar la configuración de estos software de virtualización. Vagrant está escrito en lenguaje Ruby, pero se permite utilizar otros lenguajes de programación sin ningún problema.

Los mencionados archivos de configuración son denominados Vagrantfiles, estos Vagrantfiles pueden ser compartidos entre desarrolladores para replegar en sus equipos máquinas virtuales ya creadas. La utilización de Vagrant es útil en equipos de desarrollo montados por varias personas. Puesto que nos permite asegurarnos de que todos y cada uno de los integrantes trabajan con el mismo entorno de desarrollo.

Principales características

Además de solucionar problemas de compatibilidades de software con algunos sistemas operativos, nos permite que nuestros proyectos vayan acompañados de su entorno configurado, ya que los ficheros de configuración son archivos de texto plano y estos pueden ser versionados en plataformas como Git o subversión. De esta forma, permite que las personas que se incorporen en un proyecto ya comenzado tan solo tenga que descargar del repositorio, el proyecto, y ejecutar la orden Vagrant, con ello ya tendría listo su entorno de desarrollo para trabajar en el equipo.

Por defecto el comando Vagrant trabaja con VirtualBox, VirtualBox es un software de virtualización para la creación de máquinas virtuales. Pero Vagrant no nos limita solo a VirtualBox, podemos usar también VMware Workstation en Windows y VMware Fusion en MacOS, pero en MacOS además necesitamos un plugin de pago. A veces, algunas Boxes (entornos creados con Vagrant), podemos ejecutarlos en Parallels Desktop, otro asistente de virtualización pero este es de pago.

En cuanto a la arquitectura de Vagrant, utiliza «Provisioners» y también «Providers» como bloques de construcción para los entornos de desarrollo. Estos «Provisioners» son herramientas para que los usuarios puedan personalizar su configuración en entornos virtuales. Los «Providers» son los servicios que usa Vagrant internamente para configurar y crear los entornos virtuales. Un detalle importante es que VMware y AWS son soportados a través de plugins, pero no de forma nativa.

Otro aspecto muy destacable de Vagrant es la gran comunidad que hay detrás de esta herramienta, podremos acceder a una sección de «Docs» donde tendremos toda la documentación de la herramienta explicada en detalle, además, también tenemos una sección de «Community» donde podremos poner en los foros nuestras dudas. Por supuesto, este proyecto está más vivo que nunca, y tendremos desde la web oficial un enlace al proyecto oficial de

GitHub donde podremos acceder al código fuente, ver los fallos o bugs que hay y que se han solucionado, y poner dudas sobre el funcionamiento de una determinada función y mucho más. Por último, Vagrant es ampliamente utilizado por empresas tan conocidas como Mozilla, Expedia, Nokia o Disqus para el desarrollo interno de sus propias herramientas.

Qué ventajas tiene esta herramienta

Hay que tener en cuenta que los equipos que usan esta herramienta lo cierto es que lo tienen más sencillo a la hora de cambiar posibles archivos de configuración de este software y así poder ejecutarlos o comprobarlos cuando se quiera. Y todo con el objetivo de entender más a fondo cómo se desarrolla una máquina virtual.

Entre otras de sus ventajas, lo cierto es que también hay que señalar que permite que los equipos puedan incluso migrar a la nube o entre diferentes proveedores. Además de que no es necesario llegar a rediseñar el proceso que usan para poder construir las máquinas virtuales.

Por otro lado, las máquinas virtuales que se vayan a crear con esta herramienta, la realidad es que ofrecen un gran nivel de aislamiento. De esta manera se consigue aprovechar el soporte de virtualización especializado que está incorporado en las CPU modernas. Con esto se consigue que las VM y las herramientas que las creas, puedan llegar a ser una gran opción en cuestiones como la seguridad o el aislamiento.

Y entre otros de los diferentes beneficios que ofrece esta herramienta también se puede destacar que hace que sea más sencillo llevar a cabo el proceso de actualización del software del server que se esté utilizando en las diferentes estaciones de los desarrolladores. Por lo que facilita el trabajo en todo momento.

EJERCICIO

Primero intenté instalar vagrant en kvm, pero resultó ser una tarea imposible, e hiciera lo que hiciera siempre daba error. Me imagino por ser una anidación dentro de otra anidación.

Así que procedo a instalarlo directamente en virtualbox a través de Centos. Primero descargo el archivo de la página oficial

Instalar Vagrant 2.4.3 (última versión)

macOS

Gestor de paquetes

```
brew tap hashicorp/tap
brew install hashicorp/tap/hashicorp-vagrant
```

Descarga binaria

AMD64 Versión: 2.4.3 [Descargar](#)

ARM64 Versión: 2.4.3 [Descargar](#)

Ventanas

Descarga binaria

AMD64 Versión: 2.4.3 [Descargar](#)

I686 Versión: 2.4.3 [Descargar](#)

Acerca de Vagrant

Vagrant es la utilidad de línea de comandos para administrar el ciclo de vida de las máquinas virtuales.

Documentos destacados

- Instalar
- Interfaz de línea de comandos
- Cajas
- Redes
- Carpets sincronizadas
- Proveedores

Registro de vagabundos de HCP

Cajas virtuales para Linux, Laravel y cualquier entorno de desarrollo

[Prueba HCP Vagrant Registry gratis](#)

Lo instalo. Una vez instalado, en el cmd, creamos una carpeta que llamaré vagrant\vm1, que es donde irá mi caja de vagrant y dónde me ubicaré con un cd.

Símbolo del sistema

```
Microsoft Windows [Versión 10.0.22631.1]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\IVANA>cd ..
C:\Users>cd ..
C:\>mkdir vagrant\vm1
C:\>cd vagrant\vm1
C:\vagrant\vm1>
```

Ejercicio10. Scripts

Nombre	Fecha de modificación	Tipo	Tamaño
iMyFone D-Back for Windows	13/10/2024 11:28	Carpeta de archivos	
iMyFone_Backup	13/10/2024 11:28	Carpeta de archivos	
Intel	08/12/2024 19:57	Carpeta de archivos	
netrequestIson	15/11/2024 11:38	Carpeta de archivos	
OneDriveTemp	05/10/2024 20:00	Carpeta de archivos	
PerfLogs	07/05/2022 7:24	Carpeta de archivos	
ProgramData	27/11/2024 21:37	Carpeta de archivos	
Recovery	05/10/2024 12:48	Carpeta de archivos	
SQL2022	27/11/2024 21:52	Carpeta de archivos	
Usuarios	05/10/2024 13:11	Carpeta de archivos	
vagrant	12/12/2024 0:06	Carpeta de archivos	
Windows	27/11/2024 21:53	Carpeta de archivos	
xampp	27/11/2024 22:27	Carpeta de archivos	

Vuelvo a la página oficial de vagrant y pincho en cajas para descargarme una de ellas. En mi caso he elegido una de Centos 7.

Descubre las cajas Vagrant

Buscar...

Proveedor: Arquitectura: Ordenar por: Más descargados

Mostrando 25 resultados

Nombre de la caja	Última versión	Descargas	Último lanzamiento	Proveedores	Arquitecturas
Ubuntu / Trusty64	20191107.0.0	30,797,988	Hace 265 semanas	caja virtual	desconocido
laravel / granja	14.0.2	14,601,774	Hace 327 días	paralelas, caja virtual, libvirt, escritorio vmware	amd64, brazo64, desconocido
hashicorp / precisa64	1.1.0	6,816,287	Hace 561 semanas	vmware_fusión, caja virtual, hiperv	desconocido
centos / 7	2004.01	6,065,825	Hace 238 semanas	libvirt, caja virtual, escritorio vmware, hiperv, vmware_fusión, estación de trabajo vmware	desconocido
Ubuntu / xenial64	20211001.0.5	3,632,443	Hace 53 días	caja virtual	desconocido
Puppet / ubuntu1404-x64	20161102	2,523,446	Hace 423 semanas	escritorio vmware, caja virtual, paralelas	desconocido

centos / 7

Descripción general

Descargas: 6065826, Versiones: 35, Último lanzamiento: Hace 239 semanas

Caja Vagrant de CentOS Linux 7 x86_64

Última versión

2004.01 Liberado

Creado 15 de mayo de 2020, 2020:26

Imágenes de Vagrant de CentOS Linux 7/x86_64 actualizadas al 30/04/2020 y basadas en CentOS 7.8.2003.

Las notas de la versión detalladas se publican en <https://blog.centos.org/2020/05/updated-centos-vagrant-images-available-v2004-01/>

Los siguientes proveedores están habilitados: virtualbox, libvirt, hyperv y vmware_desktop (este último debería funcionar para vmware_fusion, vmware_workstation y vmware_desktop).

Agradecemos sus comentarios y sugerencias, únase a nosotros en la lista centos-devel (<https://lists.centos.org/>), o en #Centos-devel en irc.freenode.net

hiperv

Alojado externamente (https://cloud.centos.org/centos/7/vagrant/x86_64/images/CentOS-7-x86_64-Vagrant-2004.01.hvm.xz)

Descargar

Cómo utilizar esta caja con Vagrant

Paso 1

Opción 1: Crear un Vagrantfile e iniciar el cuadro

```
vagrant init centos/7 --box-version 2004.01
```

Opción 2: Abra el archivo Vagrantfile y reemplace el contenido con lo siguiente

```
Vagrant.configure("2") do |config|
  config.vm.box = "centos/7"
  config.vm.box_version = "2004.01"
end
```

Paso 2

Activa tu máquina virtual

```
vagrant up
```

Volvemos a abrir nuestro cmd y configuramos

```
C:\>cd vagrant\vm1
C:\vagrant\vm1>vagrant -v
Vagrant 2.4.3
C:\vagrant\vm1>vagrant box add centos/7
==> box: Loading metadata for box 'centos/7'
box: URL: https://vagrantcloud.com/api/v2/vagrant/centos/7
This box can work with multiple providers! The providers that it
can work with are listed below. Please review the list and choose
the provider you will be working with.
1) hyperv
2) libvirt
3) virtualbox
4) vmware_desktop
Enter your choice: 3
==> box: Adding box 'centos/7' (v2004.01) for provider: virtualbox
box: Downloading: https://vagrantcloud.com/centos/boxes/7/versions/2004.01/providers/virtualbox/unknown/vagrant.box
Download redirected to host: cloud.centos.org
box:
box: Calculating and comparing box checksum...
==> box: Successfully added box 'centos/7' (v2004.01) for 'virtualbox'!
C:\vagrant\vm1>
```

Iniciamos vagrant

```
C:\vagrant\vm1>vagrant init centos/7
A 'Vagrantfile' has been placed in this directory. You are now
ready to 'vagrant up' your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
'vagrantup.com' for more information on using Vagrant.
C:\vagrant\vm1>
```

```
Símbolo del sistema
C:\vagrant\vm1>vagrant init centos/7
A 'Vagrantfile' has been placed in this directory. You are now
ready to 'vagrant up' your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
'vagrantup.com' for more information on using Vagrant.

C:\vagrant\vm1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: E0EA-47B2

Directorio de C:\vagrant\vm1

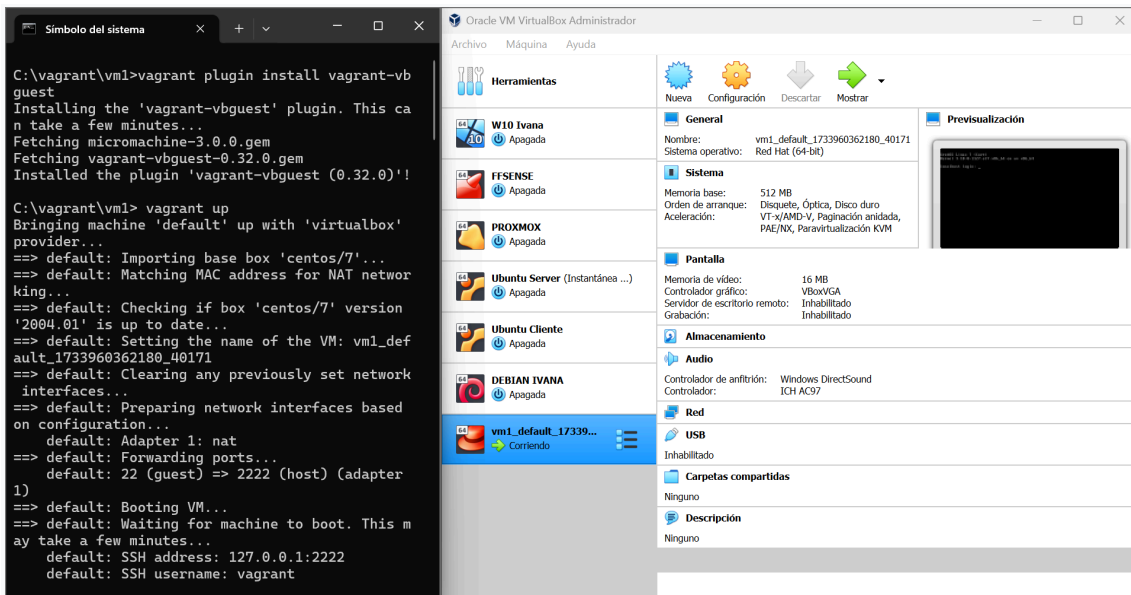
12/12/2024  00:30    <DIR>          .
12/12/2024  00:06    <DIR>          ..
12/12/2024  00:30                3.459 Vagrantfile
                1 archivos          3.459 bytes
                2 dirs  341.011.714.048 bytes libres

C:\vagrant\vm1>
```

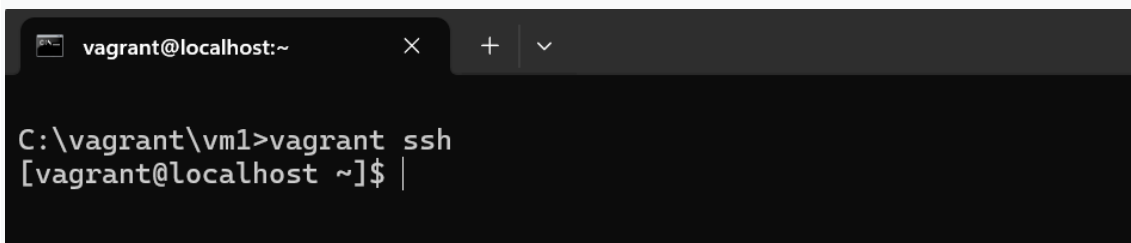
Ahora procedemos a conectarnos a VirtualBox con el comando `vagrant plugin install vagrant-vbguest`

```
Símbolo del sistema
C:\vagrant\vm1>vagrant plugin install vagrant-vbguest
Installing the 'vagrant-vbguest' plugin. This can take a few minutes...
Fetching micromachine-3.0.0.gem
Fetching vagrant-vbguest-0.32.0.gem
Installed the plugin 'vagrant-vbguest (0.32.0)'
```

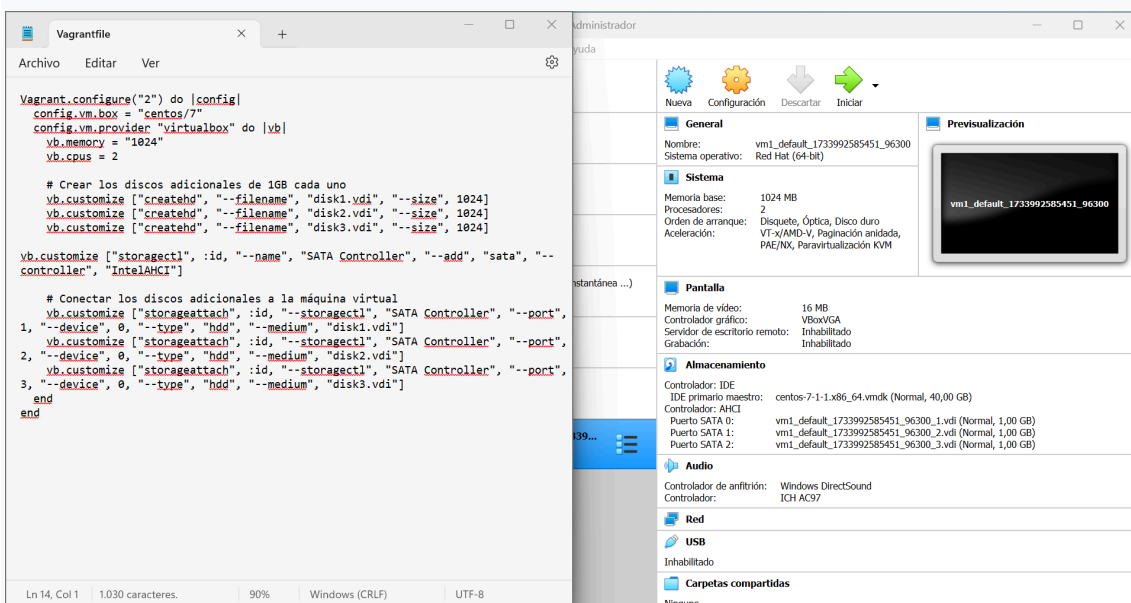
La levantamos y aparecerá ya creada en VirtualBox



Ya instalada, procedemos a comunicarnos con ella por protocolo ssh



Los discos los cree en el archivo Vagrantfile, pero en la máquina centos los tuve que añadir manualmente.



CREAMOS EL RAID 5

Comienzo instalando el comando mdadm

```
vagrant@localhost:~  
# geographically close to the client. You should use this for CentOS  
updates  
# unless you are manually picking other mirrors.  
#  
# If the mirrorlist= does not work for you, as a fall back you can tr  
y the  
# remarked out baseurl= line instead.  
#  
#  
[base]  
name=CentOS-7.8.2003 - Base  
baseurl=http://vault.centos.org/7.8.2003/os/$basearch/  
gpgcheck=1  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7  
  
[updates]  
name=CentOS-7.8.2003 - Updates  
baseurl=http://vault.centos.org/7.8.2003/updates/$basearch/  
gpgcheck=1  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7  
  
[extras]  
name=CentOS-7.8.2003 - Extras  
baseurl=http://vault.centos.org/7.8.2003/extras/$basearch/  
gpgcheck=1  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7  
  
[centosplus]  
name=CentOS-7.8.2003 - CentOSPlus  
baseurl=http://vault.centos.org/7.8.2003/centosplus/$basearch/  
gpgcheck=1  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7  
  
-- INSERT --
```



```
vagrant@localhost:~$ sudo yum install -y mdadm
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package mdadm.x86_64 0:4.1-4.el7 will be installed
--> Processing Dependency: libreport-fsfilesystem for package: mdadm-4.1-4.el7.x86_64
--> Running transaction check
---> Package libreport-fsfilesystem.x86_64 0:2.1.11-53.el7.centos will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                        Arch      Version                               Repository      Size
=====
Installing:
mdadm                          x86_64    4.1-4.el7                            base            439 k
Installing for dependencies:
libreport-fsfilesystem        x86_64    2.1.11-53.el7.centos                 base            41 k
=====

Transaction Summary
=====
Install 1 Package (+1 Dependent package)

Total download size: 479 k
Installed size: 1.0 M
Downloading packages:
(1/2): libreport-fsfilesystem-2.1.11-53.el7.centos | 41 kB  00:00
(2/2): mdadm-4.1-4.el7.x86_64.rpm                  | 439 kB  00:01
-----
Total                                              289 kB/s | 479 kB  00:01
```

Creamos nuestro RAID 5

```
vagrant@localhost:~$ sudo mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/sda1 /dev/sdb1 /dev/sdc1
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 1045504K
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
[vagrant@localhost ~]$
```

Hacemos las particiones de los discos. Para crear el RAID5 necesitare 3 discos, ya que utiliza un algoritmo de paridad distribuido entre los discos, lo que permite la tolerancia a fallos en un solo disco. Lo que significa que si uno de los discos falla, el arreglo puede seguir funcionando, ya que la paridad (información redundante) se distribuye en los discos restantes.

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo fdisk /dev/sda  
sudo: fdisk: command not found  
[vagrant@localhost ~]$ clear  
[vagrant@localhost ~]$ sudo fdisk /dev/sda  
Welcome to fdisk (util-linux 2.23.2).  
  
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.  
  
Device does not contain a recognized partition table  
Building a new DOS disklabel with disk identifier 0x17ee44ac.  
  
Command (m for help): n  
Partition type:  
   p   primary (0 primary, 0 extended, 4 free)  
   e   extended  
Select (default p): p  
Partition number (1-4, default 1): 1  
First sector (2048-2097151, default 2048):  
Using default value 2048  
Last sector, +sectors or +size{K,M,G} (2048-2097151, default 2097151)  
: G  
Last sector, +sectors or +size{K,M,G} (2048-2097151, default 2097151)  
:  
Using default value 2097151  
Partition 1 of type Linux and of size 1023 MiB is set  
  
Command (m for help): t  
Selected partition 1  
Hex code (type L to list all codes): fd  
Changed type of partition 'Linux' to 'Linux raid autodetect'  
  
Command (m for help): w  
The partition table has been altered!
```

Verificar el RAID

Primero su configuración

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo mdadm --detail /dev/md0  
/dev/md0:  
    Version : 1.2  
    Creation Time : Thu Dec 12 09:38:54 2024  
    Raid Level : raid5  
    Array Size : 2091008 (2042.00 MiB 2141.19 MB)  
    Used Dev Size : 1045504 (1021.00 MiB 1070.60 MB)  
    Raid Devices : 3  
    Total Devices : 3  
    Persistence : Superblock is persistent  
  
    Update Time : Thu Dec 12 09:39:21 2024  
    State : clean  
    Active Devices : 3  
    Working Devices : 3  
    Failed Devices : 0  
    Spare Devices : 0  
  
    Layout : left-symmetric  
    Chunk Size : 512K  
  
Consistency Policy : resync  
  
    Name : localhost.localdomain:0 (local to host localhos  
t.localdomain)  
    UUID : 16fb74dc:1f8004a1:5504b256:243c4f6f  
    Events : 18  
  
    Number Major Minor RaidDevice State  
      0      8       1         0  active sync  /dev/sda1  
      1      8      17         1  active sync  /dev/sdb1  
      3      8      33         2  active sync  /dev/sdc1  
[vagrant@localhost ~]$
```

Y segundo su estado

```
vagrant@localhost:~  
[vagrant@localhost ~]$ cat /proc/mdstat  
Personalities : [raid6] [raid5] [raid4]  
md0 : active raid5 sdc1[3] sdb1[1] sda1[0]  
      2091008 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3]  
      [UUU]  
  
unused devices: <none>  
[vagrant@localhost ~]$
```

¿Qué diferencia existe entre RAID 5 y RAID 1?

- RAID 5 :

- **Número mínimo de discos** : 3.
- **Distribución de datos y paridad** : En RAID 5, los datos y la paridad se distribuyen a lo largo de todos los discos. Esto proporciona tolerancia a fallos (puede fallar un disco) y optimiza el uso del espacio de almacenamiento.
- **Capacidad** : La capacidad utilizable es igual a la cantidad de discos menos uno. Por ejemplo, si tienes 3 discos de 1TB cada uno, la capacidad utilizable es de 2TB (ya que 1TB se utiliza para almacenar la paridad).
- **Rendimiento** : Buen rendimiento en lectura y escritura debido a la distribución de datos y paridad.
- **RAID 1** :
 - **Número mínimo de discos** : 2.
 - **Espejo (mirroring)** : En RAID 1, los datos se duplican de forma exacta en dos discos. Esto significa que cada disco tiene una copia exacta de los datos, proporcionando redundancia.
 - **Capacidad** : La capacidad utilizable es igual a la del disco más pequeño, ya que los datos se replican en todos los discos del arreglo. Por ejemplo, si tienes 2 discos de 1TB, la capacidad utilizable será de 1TB, ya que los datos se duplican.
 - **Rendimiento** : Las lecturas pueden ser rápidas, ya que los datos se pueden leer desde cualquier disco, pero las escrituras son más lentas, ya que deben escribirse en ambos discos simultáneamente.

La capacidad que tiene el RAID que he creado es de 2 GB

$$\text{Capacidad útil} = (3 - 1) \times 1\text{GB} = 2\text{GB}$$

Crear un Volumen Lógico (LVM)

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo pvcreate /dev/md0  
Physical volume "/dev/md0" successfully created.  
[vagrant@localhost ~]$ sudo vgcreate vg_raid5 /dev/md0  
Volume group "vg_raid5" successfully created
```

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo lvcreate -n lv_raid5 -L 500M vg_raid5  
WARNING: xfs signature detected on /dev/vg_raid5/lv_raid5 at offset 0  
. Wipe it? [y/n]: y  
Wiping xfs signature on /dev/vg_raid5/lv_raid5.  
Logical volume "lv_raid5" created.  
[vagrant@localhost ~]$ sudo mkfs.xfs /dev/vg  
[vagrant@localhost ~]$ sudo mkfs.xfs /dev/vg_raid5/lv_raid5  
meta-data=/dev/vg_raid5/lv_raid5 isize=512    agcount=8, agsize=16000  
blks  
          =                               sectsz=512    attr=2, projid32bit=1  
          =                               crc=1         finobt=0, sparse=0  
data      =                               bsize=4096    blocks=128000, imaxpct=  
25  
          =                               sunit=128     swidth=256 blks  
naming    =version 2                     bsize=4096    ascii-ci=0 ftype=1  
log       =internal log                 bsize=4096    blocks=624, version=2  
          =                               sectsz=512    sunit=8 blks, lazy-coun  
t=1  
realtime  =none                          extsz=4096     blocks=0, rtextents=0  
[vagrant@localhost ~]$ sudo mkdir -p /mnt/raid5  
[vagrant@localhost ~]$ sudo mount /dev/vg_raid5/lv_raid5 /mnt/raid5  
[vagrant@localhost ~]$
```

```
vagrant@localhost:~  
[vagrant@localhost ~]$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
devtmpfs        489M   0  489M   0% /dev  
tmpfs           496M   0  496M   0% /dev/shm  
tmpfs           496M  6.8M  489M   2% /run  
tmpfs           496M   0  496M   0% /sys/fs/cgroup  
/dev/sdd1       40G   3.1G   37G   8% /  
tmpfs           100M   0  100M   0% /run/user/1000  
/dev/mapper/vg_raid5-lv_raid5 498M   26M  473M   6% /mnt/raid5  
[vagrant@localhost ~]$ df -h | grep /mnt/raid5  
/dev/mapper/vg_raid5-lv_raid5 498M   26M  473M   6% /mnt/raid5  
[vagrant@localhost ~]$
```

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo mdadm --manage /dev/md0 --add /dev/sdd1  
mdadm: Cannot open /dev/sdd1: Device or resource busy  
[vagrant@localhost ~]$ sudo lsof | grep /dev/sdd1  
sudo: lsof: command not found  
[vagrant@localhost ~]$ lsblk  
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT  
sda                                  8:0      0   1G  0 disk  
└─sda1                              8:1      0 1023M  0 part  
   └─md0                             9:0      0    2G  0 raid5  
      └─vg_raid5-lv_raid5            253:0    0   500M  0 lvm    /mnt/raid5  
sdb                                  8:16     0    1G  0 disk  
└─sdb1                              8:17     0 1023M  0 part  
   └─md0                             9:0      0    2G  0 raid5  
      └─vg_raid5-lv_raid5            253:0    0   500M  0 lvm    /mnt/raid5  
sdc                                  8:32     0    1G  0 disk  
└─sdc1                              8:33     0 1023M  0 part  
   └─md0                             9:0      0    2G  0 raid5  
      └─vg_raid5-lv_raid5            253:0    0   500M  0 lvm    /mnt/raid5  
sdd                                  8:48     0   40G  0 disk  
└─sdd1                              8:49     0   40G  0 part  /  
[vagrant@localhost ~]$ |
```

Montar Marcar un Disco como Estropeado

Voy a estropear el disco /dev/sdb

```
vagrant@localhost:~$ sudo mdadm --manage /dev/md0 --fail /dev/sdb1
mdadm: set /dev/sdb1 faulty in /dev/md0
[vagrant@localhost ~]$ sudo mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
  Creation Time : Thu Dec 12 09:38:54 2024
    Raid Level : raid5
    Array Size : 2091008 (2042.00 MiB 2141.19 MB)
  Used Dev Size : 1045504 (1021.00 MiB 1070.60 MB)
    Raid Devices : 3
    Total Devices : 3
 Persistence : Superblock is persistent

 Update Time : Thu Dec 12 10:12:34 2024
      State : clean, degraded
Active Devices : 2
Working Devices : 2
Failed Devices : 1
Spare Devices : 0

 Layout : left-symmetric
 Chunk Size : 512K

Consistency Policy : resync

        Name : localhost.localdomain:0 (local to host localhost.localdomain)
        UUID : 16fb74dc:1f8004a1:5504b256:243c4f6f
        Events : 20

Number   Major   Minor   RaidDevice State   /dev/sda1
  0         8         1         0   active sync
-         0         0         1   removed
  3         8        33         2   active sync   /dev/sdc1

EVENTS : 20

Number   Major   Minor   RaidDevice State   /dev/sda1
  0         8         1         0   active sync
-         0         0         1   removed
  3         8        33         2   active sync   /dev/sdc1
  1         8        17         -   faulty   /dev/sdb1
[vagrant@localhost ~]$
```

Verificamos acceso al fichero

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo touch /mnt/raid5/prueba.txt  
[vagrant@localhost ~]$ ls /mnt/raid5/  
prueba.txt  
[vagrant@localhost ~]$
```

Retiramos el disco estropeado

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo mdadm --manage /dev/md0 --remove /dev/sdb  
1  
mdadm: hot removed /dev/sdb1 from /dev/md0  
[vagrant@localhost ~]$
```

añadimos un nuevo disco al raid

```
vagrant@localhost:~  
[vagrant@localhost ~]$ sudo mdadm --manage /dev/md0 --add /dev/sdb1  
mdadm: added /dev/sdb1  
[vagrant@localhost ~]$
```

Verificamos la sincronización

Como he hecho varias pruebas, tengo dos discos borrados


```
vagrant@localhost:~  
/dev/md0:  
    Version : 1.2  
    Creation Time : Thu Dec 12 09:38:54 2024  
    Raid Level : raid5  
    Array Size : 2091008 (2042.00 MiB 2141.19 MB)  
    Used Dev Size : 1045504 (1021.00 MiB 1070.60 MB)  
    Raid Devices : 3  
    Total Devices : 3  
    Persistence : Superblock is persistent  
  
    Update Time : Thu Dec 12 11:36:14 2024  
    State : clean, FAILED  
    Active Devices : 1  
    Working Devices : 1  
    Failed Devices : 2  
    Spare Devices : 0  
  
    Layout : left-symmetric  
    Chunk Size : 512K  
  
Consistency Policy : resync  
  
    Name : localhost.localdomain:0 (local to host localhost.localdomain)  
    UUID : 16fb74dc:1f8004a1:5504b256:243c4f6f  
    Events : 84  
  
    Number   Major   Minor   RaidDevice State  
    0         8       1       0       active sync /dev/sda1  
    -         0       0       1       removed  
    -         0       0       2       removed
```

```
vagrant@localhost:~  
[vagrant@localhost ~]$ cat /proc/mdstat  
Personalities : [raid6] [raid5] [raid4]  
md0 : active raid5 sdb1[4](F) sdc1[3](F) sda1[0]  
      2091008 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/1]  
      [U__]  
  
unused devices: <none>  
[vagrant@localhost ~]$
```

Creamos un disco reserva

```
vagrant@localhost:~$ sudo mdadm --manage /dev/md0 --add /dev/sdd1
mdadm: added /dev/sdd1
```

Simulamos el fallo de otro disco

```
vagrant@localhost:~$ sudo mdadm --manage /dev/md0 --fail /dev/sdc1
mdadm: set /dev/sdc1 faulty in /dev/md0
[vagrant@localhost ~]$
```

Verificamos la sincronización con el disco reserva

```
Every 2.0s: cat /proc/mdstat Thu Dec 12 10:29:52 2024

Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 sdb1[4] sdc1[3](F) sda1[0]
      2091008 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/2]
      [UU_]
unused devices: <none>
```

Agregamos otro disco a Vagrant

```
config.vm.provider "virtualbox" do |vb|
  vb.memory = "1024"
  vb.cpus = 2

  # Añadir el controlador SATA
  vb.customize ["storagectl", :id, "--name", "SATA Controller", "--add", "sata", "--controller", "IntelAHCI"]

  # Crear discos virtuales de 1 GB cada uno
  vb.customize ["createhd", "--filename", "disk1.vdi", "--size", 1024]
  vb.customize ["createhd", "--filename", "disk2.vdi", "--size", 1024]
  vb.customize ["createhd", "--filename", "disk3.vdi", "--size", 1024]

  # Adjuntar los discos virtuales al controlador SATA
  vb.customize ["storageattach", :id, "--storagectl", "SATA Controller", "--port", 1, "--device", 0, "--type", "hdd", "--medium", "disk1.vdi"]
  vb.customize ["storageattach", :id, "--storagectl", "SATA Controller", "--port", 2, "--device", 0, "--type", "hdd", "--medium", "disk2.vdi"]
  vb.customize ["storageattach", :id, "--storagectl", "SATA Controller", "--port", 3, "--device", 0, "--type", "hdd", "--medium", "disk3.vdi"]
  vb.customize ["storageattach", :id, "--storagectl", "SATA Controller", "--port", 4, "--device", 0, "--type", "hdd", "--medium", "disk4.vdi"]
end
```

```
[vagrant@localhost ~]$ sudo mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
  Creation Time : Thu Dec 12 19:53:01 2024
    Raid Level : raid5
    Array Size : 2091008 (2042.00 MiB 2141.19 MB)
  Used Dev Size : 1045504 (1021.00 MiB 1070.60 MB)
    Raid Devices : 3
  Total Devices : 4
 Persistence : Superblock is persistent

    Update Time : Thu Dec 12 21:01:18 2024
      State : clean
 Active Devices : 3
Working Devices : 4
 Failed Devices : 0
 Spare Devices : 1


    Layout : left-symmetric
   Chunk Size : 512K

Consistency Policy : resync

           Name : localhost.localdomain:0 (local to host localhost.localdomain)
          UUID : f1912857:52ae9bd5:fe3bb347:b87e287e
        Events : 78

   Number   Major   Minor   RaidDevice State
     0         8       1         0     active sync  /dev/sda1
     1         8      17         1     active sync  /dev/sdb1
     3         8      33         2     active sync  /dev/sdc1

     4         8      49         -     spare   /dev/sdd1
[vagrant@localhost ~]$
```

Redimensionar

```
vagrant@localhost:~
[vagrant@localhost ~]$ sudo xfs_growfs /mnt/raid5
meta-data=/dev/mapper/vg_raid5-lv_raid5 isize=512    agcount=8, agsize=16000 blks
       =                                           sectsz=512   attr=2, projid32bit=1
       =                                           crc=1        finobt=0 spinodes=0
data     =                                           bsize=4096   blocks=128000, imaxpct=
25       =                                           sunit=128    swidth=256 blks
naming   =version 2                             bsize=4096   ascii-ci=0 ftype=1
log      =internal                             bsize=4096   blocks=624, version=2
       =                                           sectsz=512   sunit=8 blks, lazy-coun
t=1
realtime =none                                 extsz=4096   blocks=0, rtextents=0
data blocks changed from 128000 to 522240
[vagrant@localhost ~]$
```