

Administración e Instalación de Linux CentOS (Windows 10)

Marco Andrés Vázquez Hernández

Práctica 1: Documentación.

Agosto de 2018

Instituto Politécnico Nacional

Prerequisitos (Virtual)

Instalación de Virtual Box (Windows 10)

Descargar la última versión de Virtual Box: <https://virtualbox/wiki/Downloads>.

Instalación de Vagrant (Windows 10)

Descargar la última versión de Vagrant: <https://vagrantup.com/downloads.html>.

Uso de Tools-CIC

Ejecutar desde la terminal (Powershell):

```
mkdir ~/Prac
cd ~/Prac
vagrant init nogala/centos --box-version 1
vagrant plugin install vagrant-vbguest
vagrant up
vagrant ssh
```

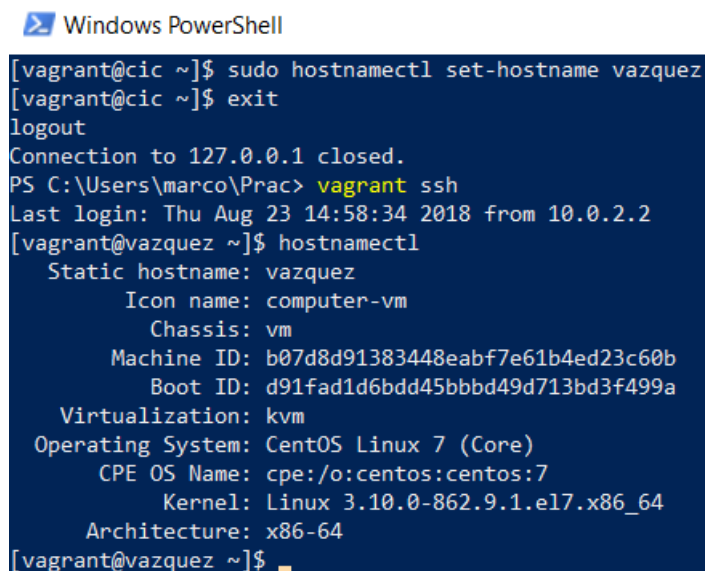
Nota: Hacer update y upgrade

```
sudo yum update
sudo yum upgrade
```

1) Cambio de nombre del host

```
sudo hostnamectl set-hostname vazquez
```

Para que el cambio tenga efecto se necesita reiniciar.



```
Windows PowerShell
[vagrant@cic ~]$ sudo hostnamectl set-hostname vazquez
[vagrant@cic ~]$ exit
logout
Connection to 127.0.0.1 closed.
PS C:\Users\marco\Prac> vagrant ssh
Last login: Thu Aug 23 14:58:34 2018 from 10.0.2.2
[vagrant@vazquez ~]$ hostnamectl
  Static hostname: vazquez
            Icon name: computer-vm
          Chassis: vm
    Machine ID: b07d8d91383448eabf7e61b4ed23c60b
       Boot ID: d91fad1d6bdd45bbbd49d713bd3f499a
  Virtualization: kvm
 Operating System: CentOS Linux 7 (Core)
    CPE OS Name: cpe:/o:centos:centos:7
          Kernel: Linux 3.10.0-862.9.1.el7.x86_64
    Architecture: x86-64
[vagrant@vazquez ~]$
```

2) Actualización y sincronización de la hora y fecha con servicio NTP

Para cambiar la hora se puede elegir de una lista:

```
timedatectl list-timezones
timedatectl set-timezone America/Mexico_City
hwclock --systohc
```

Para el servicio NTP se instalará chronyd

```
yum install -y chronyd
```

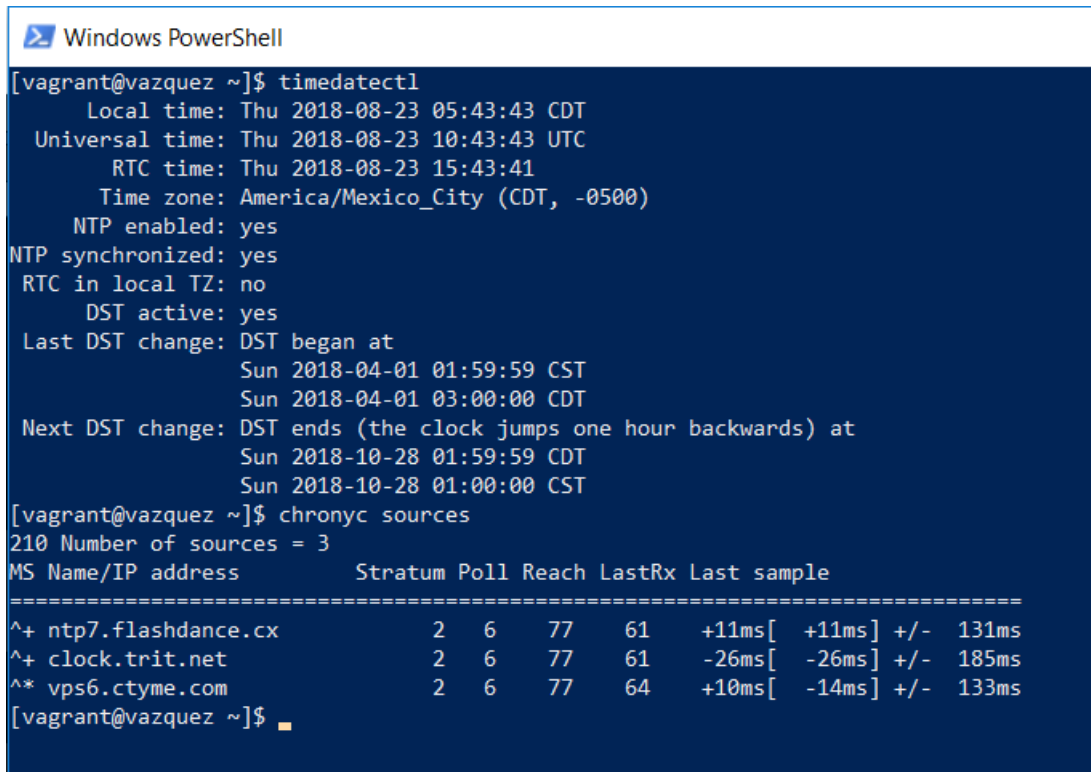
Se cambia el archivo /etc/chrony.conf con

```
server 0.mx.pool.ntp.org
server 1.north-america.pool.ntp.org
server 2.north-america.pool.ntp.org
```

```
nano /etc/chrony.conf
systemctl start chronyd
systemctl enable chronyd
```

Para verificar que se está funcionando correctamente y las fuentes que se están usando ejecutar:

```
timedatectl
chronyc sources
```



```
Windows PowerShell
[vagrant@vazquez ~]$ timedatectl
    Local time: Thu 2018-08-23 05:43:43 CDT
    Universal time: Thu 2018-08-23 10:43:43 UTC
    RTC time: Thu 2018-08-23 15:43:41
    Time zone: America/Mexico_City (CDT, -0500)
    NTP enabled: yes
NTP synchronized: yes
    RTC in local TZ: no
    DST active: yes
    Last DST change: DST began at
                     Sun 2018-04-01 01:59:59 CST
                     Sun 2018-04-01 03:00:00 CDT
    Next DST change: DST ends (the clock jumps one hour backwards) at
                     Sun 2018-10-28 01:59:59 CDT
                     Sun 2018-10-28 01:00:00 CST
[vagrant@vazquez ~]$ chronyc sources
210 Number of sources = 3
MS Name/IP address         Stratum Poll Reach LastRx Last sample
=====
^+ ntp7.flashdance.cx      2    6    77    61    +11ms[ +11ms] +/- 131ms
^+ clock.trit.net          2    6    77    61    -26ms[ -26ms] +/- 185ms
^* vps6.ctyme.com          2    6    77    64    +10ms[ -14ms] +/- 133ms
[vagrant@vazquez ~]$
```

3.a) Instalación de Java

Se revisa la versión más reciente de Java desde el navegador y se realiza un wget (si no se tiene wget se instala):

```
yum install wget
wget --no-cookies --no-check-certificate --header "Cookie:
gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie"
"http://download.oracle.com/otn-pub/java/jdk/8u181-b13/96a7b8442fe848ef90c96a2fad6ed6d1/
jdk-8u181-linux-x64.rpm"
```

Se instala localmente desde la carpeta en donde lo descargamos:

```
yum localinstall -y jdk-8u181-linux-x64.rpm
```

Revisar y cambiar a la última versión de java con los comandos:

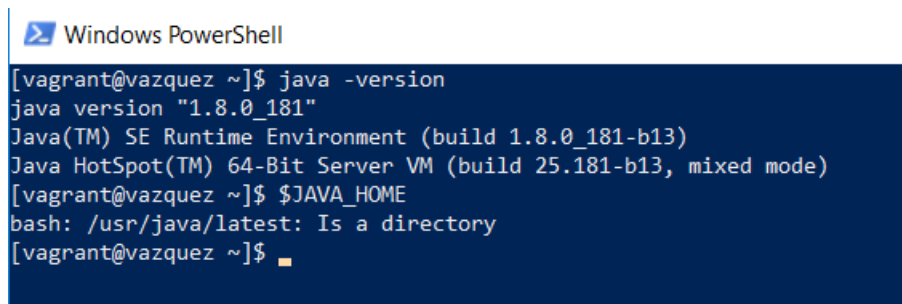
```
cd /usr/java
ll
alternatives --config java
```

Crear la variable de entorno HOME_JAVA (como root)

```
echo "export JAVA_HOME=/usr/java/latest" >> /etc/environment
```

Para revisar que la instalación haya sido correcta:

```
java -version
echo $JAVA_HOME
```



```
Windows PowerShell
[vagrant@vazquez ~]$ java -version
java version "1.8.0_181"
Java(TM) SE Runtime Environment (build 1.8.0_181-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.181-b13, mixed mode)
[vagrant@vazquez ~]$ $JAVA_HOME
bash: /usr/java/latest: Is a directory
[vagrant@vazquez ~]$
```

3.b) Instalación de Maven

Ejecutar los comandos:

```
yum install yum-utils -y
sudo yum install yum-utils -y
sudo yum groupinstall "Development Tools" -y
cd /usr/local/src
wget http://www-eu.apache.org/dist/maven/maven-3/3.5.4/binaries/apache-maven-3.5.4-bin.tar.gz
tar -xvzf apache-maven-3.5.4-bin.tar.gz
rm apache-maven-3.5.4-bin.tar.gz
mv apache-maven-3.5.4 apache-maven
```

Para cambiar los paths, crear un archivo en /etc/profile.d con nombre maven.sh que contenga lo siguiente:

```
cd /etc/profile.d/
nano maven.sh
```

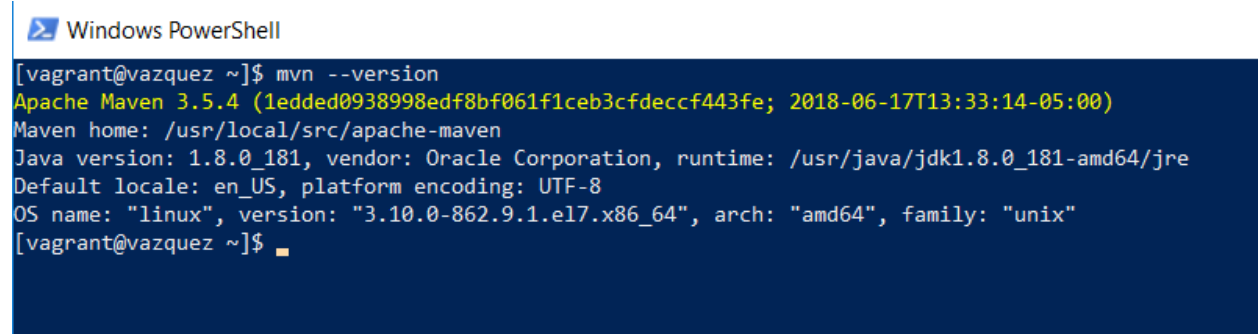
```
# Apache Maven Environment Variables
# MAVEN_HOME for Maven 1 - M2_HOME for Maven 2
export M2_HOME=/usr/local/src/apache-maven
export PATH=${M2_HOME}/bin:${PATH}
```

Permitir la ejecución de tal archivo con:

```
sudo chmod +x maven.sh
```

Para revisar que este instalado correctamente ejecutar:

```
mvn --version
```

A terminal window titled "Windows PowerShell" showing the output of the command 'mvn --version'. The output displays the Apache Maven version as 3.5.4 with a SHA-256 hash, the Maven home directory, Java version 1.8.0_181, default locale en_US, and OS information (Linux 3.10.0-862.9.1.el7.x86_64, amd64 architecture).

```
[vagrant@vazquez ~]$ mvn --version
Apache Maven 3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T13:33:14-05:00)
Maven home: /usr/local/src/apache-maven
Java version: 1.8.0_181, vendor: Oracle Corporation, runtime: /usr/java/jdk1.8.0_181-amd64/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "3.10.0-862.9.1.el7.x86_64", arch: "amd64", family: "unix"
[vagrant@vazquez ~]$
```

3.c) Instalación de Python (Desde Código Fuente)

Instalar prerequisites:

```
sudo yum install gcc openssl-devel bzip2-devel libffi-devel -y
```

Buscar en internet la última versión de Python (Gzipped source tarball) y extraer en /usr/local/src

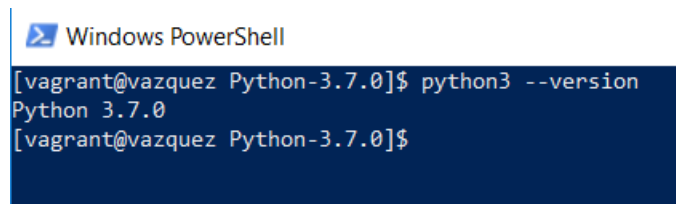
```
cd /usr/local/src/
sudo wget https://www.python.org/ftp/python/3.7.0/Python-3.7.0.tgz
sudo tar xvzf Python-3.7.0.tgz
```

Entrar en la carpeta y configurar con los siguientes comandos

```
cd Python-3.7.0/
sudo ./configure
sudo make
sudo make install
```

Para verificar que está instalado:

```
python3 --version
```

A terminal window titled "Windows PowerShell" showing the output of the command 'python3 --version'. The output displays the Python version as 3.7.0.

```
[vagrant@vazquez Python-3.7.0]$ python3 --version
Python 3.7.0
[vagrant@vazquez Python-3.7.0]$
```

4) Configuración de las herramientas de rendimiento y administración del sistema (tmux, htop, git)

Para instalar tmux:

```
sudo yum install epel-release
sudo yum install tmux
```

para instalar htop:

```
wget dl.fedoraproject.org/pub/epel/7/x86_64/Packages/e/epel-release-7-11.noarch.rpm
sudo rpm -ihv epel-release-7-11.noarch.rpm
sudo yum install htop
```

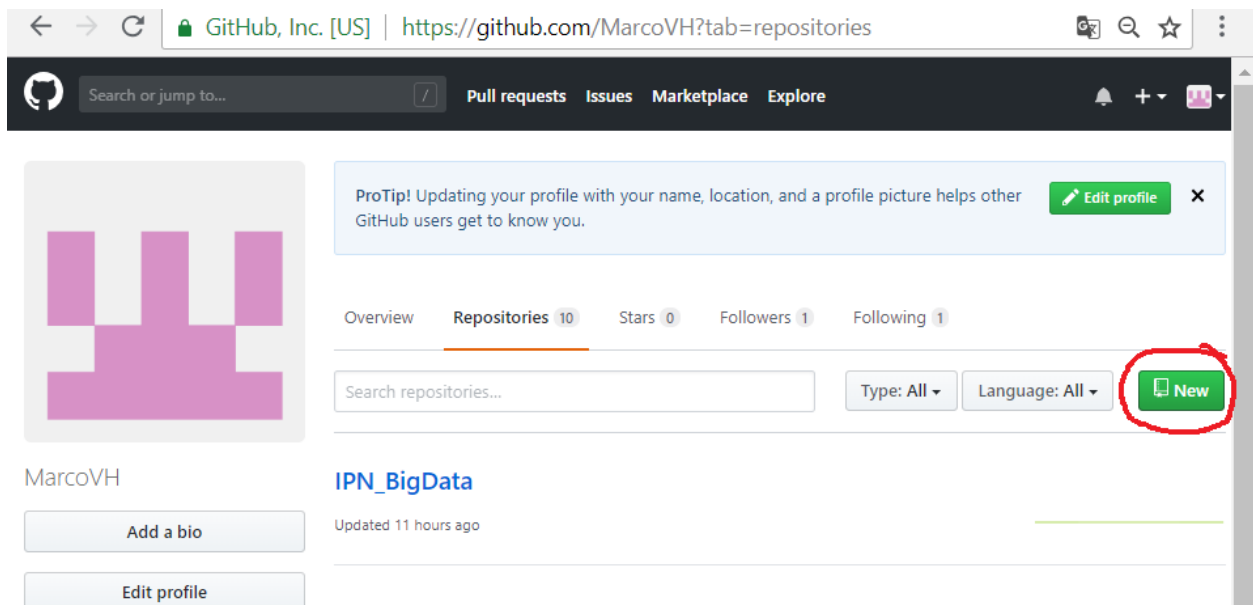
Para git se recomienda crear otro (super)usuario:

```
sudo useradd marco
sudo usermod -G wheel marco
sudo passwd marco
su marco
```

Para instalar git y configurar el usuario:

```
sudo yum install git -y
sudo git config --global user.name "MARCOVH"
sudo git config --global user.email "marcovazquezh@gmail.com"
sudo git config --list
```

Para crear un nuevo repositorio crearlo en la cuenta en github.com:



Crear la carpeta en la maquina junto con un archivo README.md y configurarlo para git con commit y push.

```
sudo mkdir Practica-1
cd Practica-1
sudo nano README.md
sudo git init
sudo git add README.md
```

```
sudo git commit -m "primer commit"
sudo git remote add origin https://github.com/MarcoVH/Practica-1.git
sudo git push -u origin master
```

Si todo es correcto la pantalla mostrará:

```
[marco@vazquez ~]$ cd Practica-1
[marco@vazquez Practica-1]$ sudo nano README.md
[marco@vazquez Practica-1]$ git init
/home/marco/Practica-1/.git: Permission denied
[marco@vazquez Practica-1]$ sudo git init
Initialized empty Git repository in /home/marco/Practica-1/.git/
[marco@vazquez Practica-1]$ sudo git add README.md
[marco@vazquez Practica-1]$ sudo git commit -m "primer commit"
[master (root-commit) 32f4548] primer commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
[marco@vazquez Practica-1]$ sudo git remote add origin https://github.com/MarcoVH/Practica-1.git
[marco@vazquez Practica-1]$ push -u origin master
bash: push: command not found
[marco@vazquez Practica-1]$ sudo git push -u origin master
Username for 'https://github.com': marcovh
Password for 'https://marcovh@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 242 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/MarcoVH/Practica-1.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
[marco@vazquez Practica-1]$
```

Y el repositorio en github tendrá el README.md

5) Instalación y configuración del servidor de web (httpd)

NOTA: Para que funcione con vagrant, el archivo Vagrantfile debe de tener la línea “config.vm.network”forwarded__port“, guest: 80, host: 8080” no comentada.

Para instalar httpd

```
sudo yum install httpd
sudo service httpd start
```

Para checar que esté funcionando correctamente se ejecuta

```
sudo service httpd status
```

También tiene que aparecer la página de prueba de Apache HTTP server poniendo en el navegador local-host:8080 (de acuerdo al puerto que se configure en el Vagrantfile para virtualizaciones):

