****

Tutorial

**BIG DATA** for Developers

Bugfixes a la VM, release 0.1

En este documento se detallan los bugs encontrados en la VM, release 0.1 y sus soluciones, las cuales serán implementadas en la próxima versión de la máquina virtual.

Agradecemos a los early adopters que no sólo nos informaron de los fallos sino que también nos brindaron muchas veces las soluciones.

A Uds., muchas gracias, you know who you are!

**Seteo de teclado dentro de la máquina virtual**

Dependiendo el teclado que usen y la configuración de sus sistemas posiblemente tengan que editar el archivo **/etc/sysconfig/keyboard** para ajustarlo a sus necesidades:

[training@hadoop-training ~]$ cat /etc/sysconfig/keyboard

KEYTABLE="es-latin1"

MODEL="pc105"

LAYOUT="es-latin1"

KEYBOARDTYPE="pc"

[training@hadoop-training ~]$

Para usar algunas variantes de teclado en español, no tradicionales, debería bastar con cambiar el valor de KEYTABLE por **es**:

KEYTABLE="es"

**Importación de la VM**

**\* MAC address inicializada**

Algunos usuarios comentaron la necesidad de inicializar la dirección MAC de la placa de red virtual de la VM para poder importar correctamente la máquina virtual. Recomendación: \*no lo hagan\* a menos que sea absolutamente necesario (por ejemplo que Virtualbox se queje).

En el 99% de los casos ***no es necesario hacerlo****,*sin embargo si ya lo hicieron tienen dos opciones:

A) Borrar la máquina virtual (asegurarse de quitar todos los archivos cuando nos pregunte) e importarla nuevamente.

B) Ejecutar los comandos que se describen a continuación para configurar una nueva interfaz de red dentro de la VM con la MAC address correcta. Si ya trabajaron en los ejemplos provistos en el training seguramente quieran aprovechar este camino (y de paso conocer un poco más sobre Linux / CentOS / Red Hat Enterprise Linux 6.6).

- Configurar una nueva interfaz de red para la VM -

Si inicializar la dirección MAC, una vez ingresen a la VM se van a encontrar que no tienen ninguna interfaz de red activada (excepto el propio loopback del sistema), pueden comprobarlo con el comando ifconfig:

[training@hadoop-training ssh]$ ifconfig

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:65536 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

Esto es así porque al cambiar la MAC address la configuración de la interfaz de red anterior (eth0) queda obsoleta. Para remediarlo rápidamente iniciar la VM, loguearse como root (usr: **root**, pwd: **root**) y ejecutar los siguientes comandos:

# dhclient

# cp /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth1

# sed -i "s/eth0/eth1/m" /etc/sysconfig/network-scripts/ifcfg-eth1

# sed -i "s/08\:00\:27\:F1\:41\:DE/$(ifconfig | grep HWaddr | cut -d " " -f11)/m" /etc/sysconfig/network-scripts/ifcfg-eth1

# reboot

(Si en algún momento del procedimiento reciben un error chequeen la sintaxis, especialmente la línea sed que escapa los dos puntos de la dirección MAC)

Al reiniciar, comprueben que la interfaz de red está correctamente activada:

[training@hadoop-training ssh]$ ifconfig

eth1 Link encap:Ethernet HWaddr 08:00:27:3C:13:14

inet addr:10.0.2.15 Bcast:10.0.2.255 Mask:255.255.255.0

inet6 addr: fe80::a00:27ff:fe3c:1314/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:438 errors:0 dropped:0 overruns:0 frame:0

TX packets:257 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:38260 (37.3 KiB) TX bytes:40470 (39.5 KiB)

lo Link encap:Local Loopback

inet addr:127.0.0.1 Mask:255.0.0.0

inet6 addr: ::1/128 Scope:Host

UP LOOPBACK RUNNING MTU:65536 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

A partir de ahora podrán ingresar por ssh, usar WebHDFS y acceder a la web.

**\* No se puede importar la VM a VirtualBox**

Revisar (**Windows**):

A) Estar corriendo la última versión de VirtualBox, en este caso 4.3.20. Si no es así descargarla desde este link: <http://download.virtualbox.org/virtualbox/4.3.20/VirtualBox-4.3.20-96997-Win.exe>

B) La característica Hyper-V de Windows debe estar deshabilitada.

C) Controlar que en el BIOS/UEFI esté habilitada la opción para virtualización (o deshabilitada la opción de 'seguridad' asociada)

Revisar (**Linux**):

A) Estar corriendo la última versión de VirtualBox, en este caso 4.3.20.

**Problemas al implementar los ejercicios**

De forma predeterminada la VM tiene habilitado un máximo de 512MB de RAM para evitar comerse toda la memoria de sus sistemas cuando hacen el training, como reportaron en el thread posiblemente tengan que ampliar el valor a 1GB o más:

1. Con la VM detenida ir al panel de configuración

2. En la solapa System pueden alterar la cantidad de memoria disponible para la VM de acuerdo a sus requerimientos

**Error al conectarse al WebHDFS corriendo en la VM desde el host**

Debemos agregar una regla de port forwarding a VirtualBox y acomodar el firewall de la VM.

A) Agregar la regla correspondiente de forwarding (redirección) al NAT de la VM en la configuración de VirtualBox:

a.1.) Asegurarse que la VM está detenida, abrir la ventana de configuración

a.2.) Sección Red → (en la ventana de la derecha) click en Avanzadas → Reenvío de puertos

a.3.) Click en el botón **+** → Agregar una regla con los siguientes datos:

Nombre: WebHDFS, protocolo: TCP, IP host: 127.0.0.1,

puerto host: 50070, IP guest: 10.0.2.15, puerto guest: 50070

a.4.) Guardar los cambios e iniciar la VM

B) Una vez dentro de la VM hay dos caminos a seguir los cuales tienen implicancias enormemente distintas desde el punto de vista de la seguridad informática.

La opción ideal para mantener el entorno asegurado vía el firewall es agregar las excepciones necesarias para que el mismo permita la conexión entrante al puerto 50070, sin embargo esto significa que más adelante cuando necesiten habilitar otro puerto tengan que nuevamente gestionar iptables.

En nuestro caso ya que se trata de un entorno de testing vamos a simplemente deshabilitar el firewall de forma que se pueda usar la VM de la forma más práctica y transparente posible.

b.1.) Comprobamos que el firewall está corriendo:

[root@hadoop-training training]# /etc/init.d/iptables status

Tabla: filter

Chain INPUT (policy ACCEPT)

num target prot opt source destination

1 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED

2 ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0

3 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0

4 ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:22

5 REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)

num target prot opt source destination

1 REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)

num target prot opt source destination

b.2.) Lo desactivamos:

[root@hadoop-training ~]# /etc/init.d/iptables stop

iptables: Poniendo las cadenas de la política ACCEPT: filte[ OK ]

iptables: Guardando las reglas del cortafuegos: [ OK ]

iptables: Descargando módulos: [ OK ]

[root@hadoop-training ~]# /etc/init.d/iptables status

iptables: El cortafuegos no está ejecutándose.

[root@hadoop-training ~]#

Nota: en este punto, si tenemos realizada la redirección del puerto 50070 y Apache Hadoop / YARN funcionando ya deberíamos poder acceder al módulo web.

b.3.) Deshabilitamos el firewall para las sucesivas sesiones:

[root@hadoop-training training]# chkconfig iptables off

Al chequear el estado de los servicios corriendo con el comando chkconfig la línea correspondiente a iptables debería verse así:

iptables 0:desactivado 1:desactivado 2:desactivado 3:desactivado 4:desactivado 5:desactivado 6:desactivado

(Los números denotan los niveles de ejecución de Linux para los que está desactivado el firewall)