

1 Problemática en equipos linux cando se cambia a MAC da Máquina Virtual

1.1 Sumario

- 1 Edición curso Febreiro - 2015
- 2 Introducción
- 3 Configuración de partida
- 4 Nova Mac
- 5 Solución

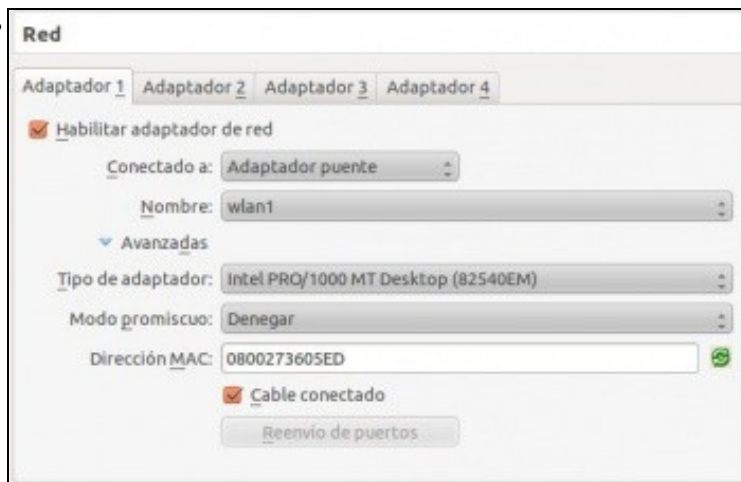
1.2 Edición curso Febreiro - 2015

- Coas novas versións de Ubuntu e Zentyal non se teñen os problemas que se describen a continuación cando unha equipo cambia de tarxeta de rede e por tanto cambia a súa MAC.
- Por tanto, déixase este material a modo de material de apoio por se en videiras versións dos SOs, volvera pasar o que a continuación se describe.

1.3 Introducción

- Para afondar dun xeito maior sobre o cambio de MAC nun sistema Linux, amosase unha práctica real coa MV Zentyal.
- Linux asocia os enderezos MAC a interfaces de rede internos do S.O.
 - ♦ Así a primeira tarxeta ethernet, por exemplo, que teña o ordenador terá unha MAC e esta estará, por defecto, asociada a **eth0**.
 - ♦ Se se cambiase a tarxeta do ordenador esta nova tarxeta tería a súa propia MAC, e linux asociaría esa nova MAC a **eth1**.
- Cal é problema?: que se ten asociada unha **configuración IP** a eth0 e se cambia a tarxeta asociada a eth0, esta non vai estar activa e a nova eth1 non estará configurada.
- A continuación vaise usar 'zserver para ver a problemática. Vaise facer en modo comandos, porque así tamén vale para cando se instale un Linux Server que non teña contorno gráfico.
- **IMPORTANTE:** asegúrate de que tes unha copia da máquina (clonada ou servizo virtualizado) e antes comezar con estes pasos tes unha **instantánea** da mesma.

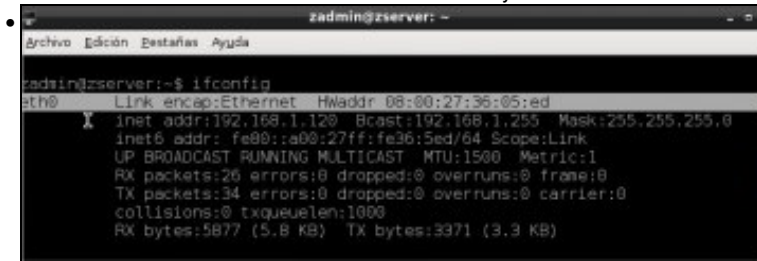
1.4 Configuración de partida



Revisar o enderezo MAC da máquina **zserver**.



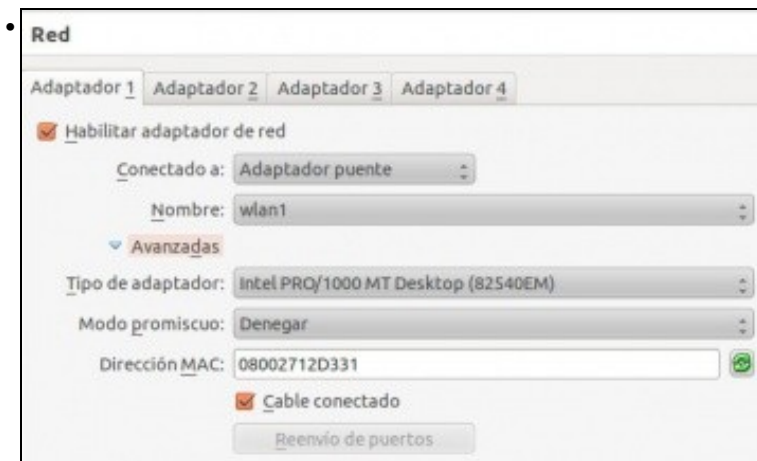
Revisar en **interfaces** do **módulo de rede** de Zentyal como **eth0** está asociada á unha IP de xeito manual.



Na liña de comandos executar **ifconfig** e observar como **eth0** está asociada á MAC da primeira imaxe e á IP que se configurou cando se instalou a MV.

1.5 Nova Mac

- Apagar a MV e xerar unha nova MAC, isto tamén vale cando se copia/clona/importa un servizo virtualizado dunha MV e se cambia a MAC, cousa que se debe facer nunha aula se o Profesor pasa a mesma MV a todo o alumnado.



Premer no botón de xerar MACs



Fixarse en Zentyal como amosa **eth1** no canto de **eth0**. Iso é porque á tarxeta que tiña MAC á que estaba asociada **eth0** xa non está no ordenador.

```

zadmin@zserver: ~
└─┬─ Archivo Edición Bstañas Ayuda
   │
   │ zadmin@zserver:~$ 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:16436 Metric:1
   │    RX packets:1587 errors:0 dropped:0 overruns:0 frame:0
   │    TX packets:1587 errors:0 dropped:0 overruns:0 carrier:0
   │    collisions:0 txqueuelen:0
   │    RX bytes:642644 (642.6 KB) TX bytes:642644 (642.6 KB)
   │
   │ zadmin@zserver:~$

```

No terminal executar **ifconfig**, xa non amosa **eth0**, pero tampouco **eth1**, iso é porque **eth1** non está activa.

```

zadmin@zserver: ~
└─┬─ Archivo Edición Bstañas Ayuda
   │
   │ zadmin@zserver:~$ ifconfig -a
   │ eth1 Link encap:Ethernet HWaddr 08:00:27:12:d3:31
   │      BROADCAST MULTICAST MTU:1500 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:1000
   │      RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
   │
   │ lo Link encap:Local Loopback
   │    inet addr:127.0.0.1 Mask:255.0.0.0

```

Ao executar **ifconfig -a** amósanse os interfaces tanto activos como inactivos, e aí está **eth1**. Observar a MAC de eth1.

1.6 Solución

- No arquivo `/etc/udev/rules.d/70-persistent-net.rules` é onde se asocia o interface **ethX** coa tarxeta de rede e a súa MAC.
- Botémoslle un ollo, para iso imos usar o editor de texto **nano** e o comando **sudo** para poder editar ese arquivo como administrador.

```

zadmin@zserver: ~
└─┬─ Archivo Edición Bstañas Ayuda
   │
   │ zadmin@zserver:~$ sudo nano /etc/udev/rules.d/70-persistent-net.rules
   │ [sudo] password for zadmin:

```

Nun terminal executar: **sudo nano /etc/udev/rules.d/70-persistent-net.rules** e introducir o contrasinal de administrador.

```

zadmin@zserver: ~
└─┬─ Archivo Edición Bstañas Ayuda
   │
   │ GNU nano 2.2.2 File: /etc/udev/rules.d/70-persistent-net.rules
   │
   │ # This file was automatically generated by the /lib/udev/write_net_rules
   │ # program, run by the persistent-net-generator.rules rules file.
   │ #
   │ # You can modify it, as long as you keep each rule on a single
   │ # line, and change only the value of the NAME= key.
   │
   │ # PCI device 0x8086:0x100e (e1000)
   │ SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR(address)==08:00:27:36:53:00, NAME="eth0"
   │
   │ # PCI device 0x8086:0x100e (e1000)
   │ SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR(address)==08:00:27:12:d3:31, NAME="eth1"

```

Neste editor non funciona o rato, moverse coas teclas do cursor. Observar que hai dúas tarxetas (PCI_device) e se nos movemos polas liñas imos ver as MACs asociadas a cada tarxeta (ATTR{address}=.....)

```

zadmin@zserver: ~
└─┬─ Archivo Edición Bstañas Ayuda
   │
   │ GNU nano 2.2.2 File: /etc/udev/rules.d/70-persistent-net.rules
   │
   │ # This file was automatically generated by the /lib/udev/write_net_rules
   │ # program, run by the persistent-net-generator.rules rules file.
   │ #
   │ # You can modify it, as long as you keep each rule on a single
   │ # line, and change only the value of the NAME= key.
   │
   │ # PCI device 0x8086:0x100e (e1000)
   │ NAME="eth0"
   │
   │ # PCI device 0x8086:0x100e (e1000)
   │ SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR(address)==08:00:27:12:d3:31, NAME="eth1"

```

finalmente ao final de cada liña está o nome do interface que se vincula a esa tarxeta de rede. A primeira está vinculada a **eth0**...

```
zadmin@zserver: ~
└─$ nano /etc/udev/rules.d/78-persistent-net.rules
GNU nano 2.2.2 File: /etc/udev/rules.d/78-persistent-net.rules
# This file was automatically generated by the /lib/udev/write_net_rules
# program, run by the persistent-net-generator.rules rules file.
#
# You can modify it, as long as you keep each rule on a single
# line, and change only the value of the NAME= key.
#
# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:36:53:00", NAME="eth1"
#
# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:12:34:56", NAME="eth2"
```

e a segunda a eth1.

```
zadmin@zserver: ~
└─$ nano /etc/udev/rules.d/78-persistent-net.rules
GNU nano 2.2.2 File: /etc/udev/rules.d/78-persistent-net.rules Modified
# This file was automatically generated by the /lib/udev/write_net_rules
# program, run by the persistent-net-generator.rules rules file.
#
# You can modify it, as long as you keep each rule on a single
# line, and change only the value of the NAME= key.
#
# PCI device 0x8086:0x100e (e1000)
#SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:36:53:00", NAME="eth0"
#
# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:12:34:56", NAME="eth2"
```

Pois ben, se se comenta (cun # -sostenido-) á asociación da primeira tarxeta con eth0 e ...

```
zadmin@zserver: ~
└─$ nano /etc/udev/rules.d/78-persistent-net.rules
GNU nano 2.2.2 File: /etc/udev/rules.d/78-persistent-net.rules Modified
# This file was automatically generated by the /lib/udev/write_net_rules
# program, run by the persistent-net-generator.rules rules file.
#
# You can modify it, as long as you keep each rule on a single
# line, and change only the value of the NAME= key.
#
# PCI device 0x8086:0x100e (e1000)
#SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:36:53:00", NAME="eth0"
#
# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:12:34:56", NAME="eth2"
```

se cambia o nome da asociación na segunda tarxeta a eth0. Xa estaría vinculada eth0 á nova MAC.

```
zadmin@zserver: ~
└─$ nano /etc/udev/rules.d/78-persistent-net.rules
GNU nano 2.2.2 File: /etc/udev/rules.d/78-persistent-net.rules Modified
# This file was automatically generated by the /lib/udev/write_net_rules
# program, run by the persistent-net-generator.rules rules file.
#
# You can modify it, as long as you keep each rule on a single
# line, and change only the value of the NAME= key.
#
# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:36:53:00", NAME="eth0"
#
# PCI device 0x8086:0x100e (e1000)
#SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:12:34:56", NAME="eth0"
```

Saír do editor premendo CTRL+X, indicar **yes** cando pregunte se se desexan gardar os cambios e gardar o arquivo.



Reiniciar o equipo e en Zentyal xa se pode observar que o interface é eth0. Pero se se fai un **ifconfig** verase que non ten IP.

```
zadmin@zserver: ~
Archivo Edición Pestañas Ayuda
zadmin@zserver:~$ sudo /etc/init.d/networking restart
* Reconfiguring network interfaces...
ssh stop/waiting
ssh start/running, process 1682
zadmin@zserver:~$
```

Unha vez reiniciado o equipo debería ter eth0 vinculado coa súa IP, pero non é así, para solucionalo reiníciase o servizo de rede co comando: **sudo /etc/init.d/networking restart**.

```
zadmin@zserver: ~
Archivo Edición Pestañas Ayuda
zadmin@zserver:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:12:d3:31
          inet addr:192.168.1.128  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe12:d331/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:63  errors:0  dropped:0  overruns:0  frame:0
          TX packets:67  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0  txqueuelen:1000
          RX bytes:6670 (6.6 KB)  TX bytes:5460 (5.4 KB)
```

Agora si que está todo OK.

Outra opción, sería configurar o interface eth1, como se fixo nuns pasos previos con eth0.

-- Antonio de Andrés Lema e Carlos Carrión Álvarez -- (2 nov 2011).