Initialize And Finalize Scripts: How do I handle network interfaces swapping names?

How do I handle network interfaces swapping names? eth0 keeps changing randomly on boot to eth1 and vice versa.

There are two ways to handle this issue:

1. Reorder the load order of the kernel modules,
2. Detect an invalid network interface order and reboot
    - via initialize script
    - via the init script

Reordering the kernel modules:

The reason NICs get reordered might be because each of them uses a different kernel module, say Module A and module B.

If those kernel modules are close together on the list of modules for a software image, then once in a while they might detect their respective hardware in a different order than the one implied from the ordering of the kernel modules themselves.

A way to deal with it is to separate those kernel modules with different kernel modules. This can be done either using CMGUI, or by means of CMSH up/down commands.

Below is a brief walkthrough for reordering the modules with cmsh:

```
[headnode]% softwareimage
[headnode->softwareimage]% use default-image
[headnode->softwareimage[default-image]]% kernelmodules
[headnode->softwareimage[default-image]->kernelmodules]% list

Module (key) Parameters
---------------------- ----------------------------------------------
xen-netfront
xen-blkfront
tun
nfs
NIC_MODULE_A
NIC_MODULE_B
tg3
sata_sil
ext2
ext3
ext4
forcedeth
bnx2
3w-xxxx
3w-9xxx
sata_svw
```

```
sata_nv
ahci
arcmsr
mptsas
mptspi
mptscsih
mpt2sas
sd_mod
ata_piix
aic7xxx
aic79xx
aacraid
megaraid_sas
igb
e1000e
sg
hpsa
cciss
qla3xxx
isci
xfs
ipmi_si
ipmi_devintf
hpilo
```

```
[headnode->softwareimage[default-image]->kernelmodules]% up NIC_MODULE_A
--or--
[headnode->softwareimage[default-image]->kernelmodules]% down NIC_MODULE_B
```

<repeat until the modules are further apart>

```
[headnode->softwareimage[default-image]->kernelmodules*]% commit
```

<wait until the initrd gets regenerated>

```
Mon Mar 11 16:10:52 2013 [notice] headnode: Initial ramdisk for image default-image is being
generated
<wait>
Mon Mar 11 16:12:18 2013 [notice] headnode: Initial ramdisk for image default-image was
regenerated successfully.
```

<test changes>

Detect invalid network interface order and rebooting

Another way to deal with the issue of NICs being identified in an incorrect order would be to

Initialize And Finalize Scripts: How do I handle network interfaces swapping names?

manually check the order when the node boots and reboot if the order is incorrect. This is only going to work if the hardware regularly changes its order of detection on boot, so if the kernel driver changes, and the new order favors the wrong NIC being identified first, you will end up with an endless reboot loop. So, some care is advised for the custom script.

Depending on the individual circumstances this should be done either via a custom initalize script, or via a modified init script of the initrd. The latter should work in most cases, whereas the former (easier to do), might not work if one of the interfaces which were identified in the incorrect order was the boot interface (again, this depends on individual circumstances).

Custom initialize script

When writing a custom initialize script use the "sreboot" command (that's not a typo) to reboot the node. Details on writing the initialize scripts can be found in the Administrator Manual.

Custom code in the initrd's init script

You can also insert custom code into the init script of the initrd which will detect when the interface were brought up in the incorrect order before the node installer starts, and will then reboot the node.

Note that (depending on the situation) a custom initialize script might also be used as an alternative to modifying the initrd's init script, however it is possible that due to NICs being brought up in the incorrect order the node installer will not reach the moment where the initialize script is run.

Please refer to the following KB article on how to manually insert code snippets into the initrd's init script.

http://kb.brightcomputing.com/faq/index.php?action=artikel&id=107

Using the guidelines in that KB article, insert your custom code snippet right before the following (existing) code fragment:

```
ifconfig $BOOTDEV mtu $MTU
if [ "$?" -ne "0" ]; then
  echo "WARNING: Setting MTU failed!"
else
  echo "Setting MTU $MTU."
fi
```

Initialize And Finalize Scripts: How do I handle network interfaces swapping names?

The exact location inside the init script is not that important (as long it is not in the code branch which gets executed by the cloud nodes).

The exact code which you need to write to test for network interfaces being out of order will, of course, be specific to your individual case. If you need any help with this, please contact our support.

Unique solution ID: #1105
Author: Piotr Wachowicz
Last update: 2013-03-27 11:22