

Q. When I reboot after performing:

```
niload -d hosts / < /etc/hosts
```

why does the NetInfo server display the following message?

```
cannot find parent NetInfo server, still looking...  
Press 'c' to continue boot without parent server.  
See your system administrator if you need help.
```

A. During the **niload -d** command the entry for the local host in the NetInfo root domain got deleted, or perhaps a new entry was created without the **serves** property. In either case, without the **serves** property, a machine can't find its parent server.

niload -d deletes entries that are in the directory but not in the input (which in this case is /etc/hosts). Thus either the machine's entry is completely lost (if the corresponding

entry is not in /etc/hosts), or else the machine's entry is overwritten by the new entry, deleting the **serves** property in the process. (**niload** is not capable of adding properties other than the standard Unix ones, which are **ip_address** and **name** in this case.)

There are two different procedures for fixing the problem, depending on whether the **serves** property got corrupted on a NetInfo **client** or on a **server**.

1. If the Machine is a Client:

The procedure to fix the problem is easier in this case, since the root domain can be accessed through NetInfoManager. Run NetInfoManager on the NetInfo server and add the missing **machines** subdirectory or the missing **serves** property (of the form **hostname/local**) to the root domain:

- Launch NetInfoManager.
- Open the / (root) domain.

- Edit Properties for *hostname*, where *hostname* is the name of your client (see following section if there is no entry for *hostname*).
- Make a **serves** property.
- Change newValue of the **serves** property to *hostname/local*.
- Reboot *hostname*.

If there is no **machines** entry for *hostname*, create one:

- Create a new **machines** subdirectory for *hostname*.
- Change the **machines** subdirectory from newValue to *hostname*.
- Edit Properties for *hostname*.
- an **ip_address** property with a value of the form xxx.yy.zzz.aaa, replacing xxx.yy.zzz.aaa with the correct ip-address.
- a **serves** property with the value *hostname/local*
- Reboot *hostname*.

2. If the Machine is a Server:

If the machine is the server, the procedure is a little more involved:

- Boot the machine, and when you get the error message, type "c" to continue. Once the machine is up, log in. Open a Shell or Terminal and **su** to root if you are not already logged in as root.
- Run **niutil** to see what has happened to the machine *hostname*. Replace *hostname* with the name of your machine:

```
#niutil -read -t localhost/network /machines/hostname
```

You might get an output that looks like this:

```
name: hostname  
ip_address: 192.72.11.1
```

```
en_address: 00:00:0f:00:3:3
bootfile: mach
bootparams: private=server:/clients/hostname root=server:/
```

Note the missing **serves** property! If your output looks like the above, add the missing **serves** property by entering the following command:

```
#niutil -createprop -t localhost/network /machines/hostname serves\
hostname/local ./network
```

and then reboot the machine.

If instead of the above output, you got the following output:

```
niutil: can't open localhost/network:/machines/hostname
```

then the **machines** subdirectory for your machine *hostname* was blown away. To recreate this directory do the following:

- Create a host directory for *hostname*:

```
#niutil -create -t localhost/network /machines/hostname
```

- Add the correct **serves** property for *hostname*:

```
#niutil -createprop -t localhost/network /machines/hostname serves\  
hostname/local ./network
```

- Add the **ip_address** property for *hostname*, replacing *xxx.yy.zzz.aaa* with the correct IP address:

```
#niutil -createprop -t localhost/network /machines/hostname\  
ip_address xxx.yy.zzz.aaa
```

- After these changes, reboot the machine:

#reboot

QA515

Valid for 1.0, 2.0, 3.0, 3.1