JumpStart™: NIS and `sysidcfg`

By Rob Snevely - Enterprise Technology Center

Sun BluePrints™ OnLine - October 1999

Please
Recycle

Adobe PostScript

JumpStart™: NIS and `sysidcfg`

In the Solaris™ 2.1 release, Sun Microsystems introduced `auto_install`, which is now known as, JumpStart™. JumpStart technology allows automation of the install process, while simultaneously allowing specific customization of a Solaris operating environment installation. This has three major benefits for I.T. professionals:

- It simplifies custom installations of Solaris software, including adding or deleting of specific software packages, disk partitioning, and other non-Solaris software as well.
- It provides a simple way to help ensure that a set of machines are all installed exactly the same.
- It saves time.

The specific requirements of the JumpStart profiles depend on the environment the machines are used in. There are as many different environments as there are Sun customers. However, this article focuses on getting a machine to the point where you can use the install profile of your choice in a totally automated way. Getting to that point is common to virtually all Solaris operating environment installations.

Booting a Machine off the Net

Two information services are required to boot a machine off the net for an installation: `ethers` and `bootparams`. The information for either or both of these can be located either in a NIS map or a file contained in `/etc` on the Solaris operating environment installation server.

ethers

Ethers is the Ethernet (MAC) address of the machine and that machine's hostname. For a machine with the hostname sephora, the entry in the `ethers` file `or ethers NIS map` would look as follows:

```
8:0:20:8e:e8:71 sephora
```

Why do you need this in the first place? The machine's only hardware-level unique identifier is this Ethernet address. The map entry says this piece of hardware also has an IP address, which is represented by its hostname, sephora.

```
# ypmatch sephora hosts
192.29.208.127   sephora
#
```

Using the information in `ethers` you can now tell a machine what its IP address is given its specific Ethernet address.

bootparams

Because we are getting all the information and file systems off of a file system on the network, we must tell sephora where to mount these files systems from. That is the function of `bootparams`. sephora's `bootparams` entry looks as follows:

```
sephora
root=wessex-208:/export/share/build/s998/s998_u3/11/
Solaris_2.7/Tools/Boot
install=wessex-208:/export/share/build/s998/s998_u3/11
boottype=:in rootopts=:rsize=32768
```

However, you don't need to enter this information yourself because the `add_install_client` script does it automatically and puts the info into `/etc/bootparams`.

Now that we have the two network information files we need to boot, should this information be placed in NIS maps? While there is no truly right or wrong answer, the best practice is not to put this information in NIS maps, unless you have an environment with multiple install servers serving the same network *and* you have access to only one of the install servers. This forces a bit of order and makes the people responsible for both install servers work together and creates fewer problems

with both servers trying to respond to the same request. In most cases, you just have `/etc/ethers` and `/etc/bootparams` files and set `/etc/nsswitch.conf` to use them. Make the following changes to the `/etc/nsswitch.conf` file:

ethers: files

bootparams: files

Now you can boot the machine off the net. However, to have a complete hands-off installation, you need a few more pieces. When the Solaris software boots up it wants to know several pieces of information in order to put a machine on the net. If you are running NIS, you probably already have IP address information in a hosts map. There are a few other maps you will need. One is `netmasks.byname`. This tells the machine how its netmask should be set. The format for the map entry is:

NetworkAddress Netmask

You can specify this by class A, class B, or class C address.

So all of the following are valid:

```
192.0.0.0      255.255.255.0 # Class A Address
192.29.0.0     255.255.255.0 # Class B Address
192.29.206.0   255.255.255.0 # Class C Address
```

However, you should only set the NetAddress as high as the network class address you have. If your company has its own Class B address for example, and, all nets have the same netmask, then you could just have one entry

192.29.0.0 255.255.255.0

Several changes were made between the Solaris 2.5.1 release and Solaris software 2.6 and beyond.

Solaris™ 2.5.1 Software

When you boot Solaris 2.5.1 software, you will be asked for the following information:

■ Terminal Type:

You will be asked this only if a keyboard and monitor are NOT attached to the machine, and you must provide a response at the console. The boot process will suspend until this question is answered. There is no NIS map to configure this. This prohibits complete hands-off installation of Solaris 2.5.1 software if you do not have

a keyboard and monitor attached to the system. The good news is that this was changed in the Solaris 2.6 release and beyond. If the keyboard and monitor are attached then a Sun Terminal Type is presumed.

- Country/Locale:

Information needed for your localized version of Solaris software. This can be handled by a NIS map called locale.byname. The entry in the map is just one line and is one of the valid locale values for Solaris 2.5.1 software. So for the English locale the value is C.

Below is the language and locale map values.

TABLE 1   Locale Map Values

Language	Valid Locale Value
Chinese	zh
English (default)	C
French	fr
German	de
Japanese	ja
Korean	ko
Latin American	es
Swedish	sv
Taiwanese	zh_TW

- Timezone:

What timezone are you in. This can be handled by the creation of a timezone.byname map. The entry would look like this.

US/Pacific Eng.Sun.COM # Sun West Coast

You can also find the listings of all timezone in `/usr/share/lib/zoneinfo`.

- Date:

What is the correct date and time. The only way to handle this in the Solaris 2.5.1 release is by declaring a host the timehost. To make the machine with the hostname vouge the timehost, you would make the following changes in the NIS hosts table and remake the map.

Old Entry

192.29.208.126 vouge

New Entry

192.29.208.126 vouge timehost

Do this only if all machines in that NIS domain are in the same timezone. Getting the date and time from a machine over the net is fine, but the time will be delayed by a greater amount the more network hops the timehost is from the machine being installed.

Making the above changes will allow the most hands-off installation of Solaris 2.5.1 software.

New Functionality in Solaris™ 2.6 Software and Beyond

While configuring hands-off installation is possible with the Solaris 2.5.1 release, it lacks a certain degree of flexibility. Many of these issues have been addressed in the Solaris 2.6 software and beyond through the use of a `sysidcfg` file.

There are six primary functions provided by the `sysidcfg` file that we will cover here. These six functions in cooperation with a netmasks map enable complete hands-off installation of Solaris 2.6 and Solaris 7 software. These are: system_locale, timezone, terminal, name_service, timeserver, and root_password.

Note – If you have an environment where Solaris 2.5.1 software is mixed with Solaris 2.6 software and beyond, you can keep the above NIS maps in effect, and the `sysidcfg` file will override any NIS maps that might be used.

To use a `sysidcfg` file, a couple of rules must be followed: First, the file must be placed in a directory that is exported at least to the machine(s) that will be using it. The filename must be `sysidcfg`. And you must give another argument, `-p` `MACHINE:/path/to/sysidcfg/directory`, to `add_install_client`. So the command line to add the machine sephora using a JumpStart profile and the `sysidcfg` file would look like as follows.

```
add_install_client -c vouge:/export/jumpstart -p harpers:/
export/sysidcfg/sephora sephora sun4u
```

Both of the arguments point to the directories where the files reside and not to the files themselves. In this example, the rules file for the JumpStart profile is in `/export/jumpstart` on the machine vouge, and the `sysidcfg` file is in

`/export/sysidcfg/sephora` on the machine harpers, and both of these directories are exported to the machine sephora. For information on exporting files systems look at the man pages for `share and dfstab`.

`sysidcfg` File Entries Format

system_locale=en_US

This sets the locale to the 8 bit US/English locale. The list of locales reside in the `/usr/lib/locale` directory.

timezone=US/Pacific

This sets the timezone to Pacific Standard Time and includes the use of Daylight Savings Time. The list of available timezone resides in the `/usr/share/lib/zoneinfo` file.

teminal=xterms

This sets the default terminal type for the console. If you use, xterms as your terminal of choice you probably want to set this to xterms or vt100. A list of all valid terminal types can be found in the `/usr/share/lib/terminfo` directory.

name_service=NIS

This sets the name service to use. There are only four valid choices. The values are NIS, NIS+, OTHER, and NONE. The two most common would be NIS (Network Information Services, also known as YP), and NIS+, which is a new version of the network information nameservice that includes enhancements in security, and the administration of the service.

timehost=localhost

This sets the date and time of the machine. If localhost is specified it is assumed that the machine has the correct time. Additionally, a different machines hostname or IP address could be used and this would alleviate the need to define a single timehost in the NIS hosts map for all machines in that NIS domain.

root_password="13 character encryption for a password found in the /etc/shadow file"

If this value is set, the encrypted entry is put into the `/etc/shadow` file on the installed machine as root's password. This prohibits the machine from stopping at boot up after installation and asking for a root password. However, I do not recommend using it for a couple of reasons. First, having all machines with the same root password is not good security policy. Also, the encrypted entry is left in the

sysidcfg file on the server, making it easier to grab the encryption key for cracking. The encrypted entry is placed on the newly installed machine in `/etc/sysidcfg` and the file is readable by the world. Again making it easier to grab the encrypted entry for cracking. If you choose to use this feature, make sure that the `sysidcfg` file on the server is removed after the machine has been installed or at least make sure that the `root_password` entry is removed. And be sure to change the permissions to 600, readable only by root, on the `/etc/sysidcfg` file on the newly installed machine.

Now you have all the information you need to do complete hands-off installations of Solaris software. This will help ensure that the machines in your environment are more uniform in their installation and most importantly save you time.

Author's Bio: Rob Snevely

Rob is a member of Sun's Enterprise Technology Centers technical staff. He has over 10 years experience in UNIX system administration, networking, and performance tuning. His major responsibilities include architecting and designing data center and network architectures.