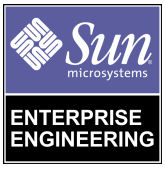
Solaris™ 8 Operating Environment Additions to `sysidcfg`

By Rob Snevely - Enterprise Technology Center

Sun BluePrints™ OnLine - March 2000



`http://www.sun.com/blueprints`

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Part No.: 806-4879-10
Revision 01, March 2000

Solaris™ 8 Operating Environment Additions to `sysidcfg`

In the October 1999 article titled *JumpStart™: NIS and `sysidcfg`* (`http://www.sun.com/software/solutions/blueprints/1099/jumpstart.pdf`), I wrote about how to configure NIS for JumpStart™ software and the use of `sysidcfg` to enable hands-off installations of Solaris™ Operating Environment. There are two additions to the install process for Solaris 8 Operating Environment (IPv6 and Kerberos Security Policies). The configuration of both of these can now be handled at install time via `sysidcfg`.

As a quick refresher, `sysidcfg` is a configuration file that provides information needed for a configuration, such as: system_locale, timezone, terminal, name_service, timehost, and root_password. This information is stored in a file that must be called `sysidcfg` and is exported to the machine being installed. For example, to create a `sysidcfg` file for the machine sephora, you can have a filesystem on the install server `/export/SYSIDCFG` which is exported and mountable by the machine sephora. You would create another directory "`mkdir -p /export/SYSIDCFG/sephora`" and place the file called `sysidcfg` in that newly created directory. The contents of the file would look something like this.

```
system_locale=en_US
terminal=sun
timezone=US/Pacific
name_service=NIS
timehost=harpers
```

When running the `add_install_client` script you would add "`-p harpers:/export/SYSIDCFG/sephora`" to the commandline.

```
./add_install_client -c harpers:/export/jumpstart -p harpers:/
export/SYSIDCFG/sephora sephora sun4u
```

New for Solaris 8 Operating Environment

Network_interface and IPv6 `sysidcfg` will alllow the configuration of a single network interface. You can only configure a single interface, as only one instance of a keyword, in this case "network_interface", is valid and only the first instance found will be used.

The basic structure is: `network_interface=VALUE {hostname=HOSTNAME ip_address=IP_ADDRESS, netmask=NETMASK, protocol_ipv6=YES or NO}`

For the network_interface VALUE, you can specify either the device name. (`hme1` or `qfe3`) or you can use the keyword PRIMARY. The PRIMARY keyword implies the first interface that is seen, usually `hme0`.

The values for hostname and ip_address should be included for the sake of clarity. However, the current HOSTNAME and IP_ADDRESS will be used at installation by default, if no other are specifed. Now, how or why should this matter? In some environments you do not do installations on a production network. It can cause unplanned spikes in network utilization. So, you might have 2 physical interfaces on a machine. One interface will be used to do the installation on a non-production network. However, you would want to configure the network interface of the machine being installed to use the interface on the production network.

The value for NETMASK is the decimal notation for the netmask. So if your default netmask is FFFFFF00, the value in the `sysidcfg` file would be netmask=255.255.255.0. Again, the `sysidcfg` file will override any value that exists in they netmask.byaddr NIS map. However, if the no netmask is defined in the `sysidcfg` file, the netmask entry in `netmasks.byaddr` will be used.

IPv6 expands the network layer address space from 32bits to 128bits. So this function is defined by the `network_interface` entry in the `sysidcfg` file. There are a couple of ways to handle this. You can specify the interface instance, which you might want to do if you are installing a machine where you are using an interface that is not the primary interface, or if you only have one interface on the machine you can use the keyword "primary". The flag to use IPv6 is binary, yes or no.

Here are a few examples.

- `network_interface=primary {protocol_ipv6=no}` (This is the simplest case)
- `network_interface=hme1 {hostname=sephora2 ip_address=192.29.209.46 netmask=255.255.255.0 protocol_ipv6=no}` (This would set the `hme1` interface instance.)
- `network_interface=qfe3 {hostname=sephora-ipv6 protocol_ipv6=yes}` This would set the `qfe3` interface instance to be configured for IPv6, but, it would use other Name Services to get the ip address and netmask.

Kerberos Security Polices

The Kerberos system authenticates individual users in a network environment. After authenticating yourself to Kerberos, you can use the Kerberos authentication option of network services such as NFS. You can also use Kerberos to do remote functions like `rsh` and `rcp` with having to use a password and better still without having to use `.rhosts` files.

If you do not use Kerberos you can just put the following in the `sysidcfg` file.

```
security_policy=none
```

If you do use Kerberos you use the following configuration. However, the entries will be specific to your Kerberos environment.

```
security_policy=kerberos {default_realm=Your_Kerberos_Realm
admin_server=Your_Kerberos_Admininstration_Server
kdc=KK0
kdc=KK1
kdc=KK2
kdc=KK3
kdc=KK4
kdc=KK5}
```

For more information on Kerberos, you can look at the following man pages in Solaris 8 Operating Environment.

Note – Note: The `-s SECTION#` flag tells the man command which man page section to use. This is very useful when looking up a command or system call that has more then one instance, like `rsh`, `write`, or `kerberos`.

```
man -s 1 kerberos
man -s 1 kinit
man -s 1 klist
man -s 1 kdestroy
man -s 1m kerbd
man -s 4 kerb.conf
man -s 4 krb.realms
```

So here is a working sample of a `sysidcfg` that will enable complete hands-off installation of Solaris 8 Operating Environment.

```
system_locale=en_US
terminal=xterms
timezone=US/Pacific
name_service=NIS
timehost=henry8
network_interface=primary {protocol_ipv6=no}
security_policy=none
```

With the above information , and information from previous Sun BluePrints™ OnLine articles, you can now do a fully hands-off installation of Solaris 8 Operating Environment. Using JumpStart software and `sysidcfg` will provide uniform Solaris Operating Environment installations and save you time.

Author's Bio: Rob Snevely

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