==========
NIS+
==========
NIS Daemons
----
ypserv - NIS Server Process
ypbind - Binding porcess
ypxfrd - High speed map transfer
rpc.yppasswdd - NIS password update daemon
rpc.ypupdated - Modifies other maps i.e "publickey"
KEYSRV: Daemon for storing logged in users' private encryption keys
(NIS+ ?)

Essential NIS daemons
----
portmap, yppasswdd, ypserv
- ypbind and ypxfrd

Proper reboot order for machines running NIS, NFS and hosting /home:

NIS, NFS, mount /home

Restart NIS Daemon
----
1. Log into NIS Master server
2. cd /usr/lib/netsvc/yp
3. ./ypstop
4. ./ypstart

Locate a machine's NIS Maps:

% ypwhich –m

Check NIS daemon status

% rpcinfo -p
Should see portmap, yppasswdd, ypserv, ypbind, ypxfrd/ypxfr
Note: ypbind and ypxfrd won't load until the main NIS daemons
(portmap, yppasswdd and ypserv) are running.

Display a host's IP address:

% ypcat hosts | grep -i nodename (NIS)

Find Home Directory Mount Point for A User:

% ypcat auto_home | grep userid

Change Local Root Password

```
# passwd –r files root
Set Root Password to never expire
# passwd –x –1 root
Set Global Password Aging Values
- Modify /etc/default/passwd
Shadow File Note: A “–1” in the /etc/shadow file indicates password
aging is off.

NSCD – Name Service Cache Deamon
--------------------------------
- This is a password, group, and host lookup caching service
- Note: Mostly superseded by SSSD, but you may find it in NIS+
implementations.
Start/Stop NSCD:
% /etc/init.d/nscd stop
% /etc/init.d/nscd start

Update NIS+ Directory Object Public Key
-----------------------------------------------
- Must be at Security Mode 0.  The default is Security Mode 2.
1. Change to Security Mode 0:
  a. Find all rpc.nisd processes:
  % ps –ef | grep –i prc.nisd
  b. Kill all running rpc.nisd processes:
  % kill –9 <PID_from_step_a>
  c. Bring NIS back up in Security Mode 0:
  % /usr/sbin/rpc.nisd –s 0
2. Unset the current secret key:
  # keylogout –f
3. Change DES key:
  a. # nisaddcred des
  b. # nisupdkeys ‘nisdefaults –d’
  c. # nisupdkeys org_dir ‘nisdefaults –d’
  d. # nisupdkeys groups_dir ‘nisdefaults –d’
  e. # /usr/sbin/keyserv

Dump NIS+ Tables
----------------------
1. Log into the NIS+ Master
2. Choose a table dumping method
  a. Dump all NIS+ tables: # /usr/cm/sa/bin/dump_tables
      Will dump to /var/tables by default.  Use –l <path><filename> to
change.
  b. Dump a specific NIS+ table:
      # /usr/cm/sa/bin/dump_tables <table_name>
3. Make you table change, and push the new tables to NIS+
  # /usr/cm/sa/bin/push_table <table_name>

NISCACHE – NIS+ directory cache daemon “niscachemgr”
-----------------------------------------------------------------
```

- Clear out NISCACHE:
  1. Remove nis cache files: # rm /var/nis/*DIRCACHE
  2. Restart NIS+ nis_cachemgr:
      ps –ef | grep cache (look for “nis_cache” or “nis_cachemgr” and note the PID)
      kill -9 <pid_from_ef>

Display NIS+ defaults: nisdefaults

Push updates from NIS+ Master to a Slave Server: /usr/lib/nis/nisping

Update DEC credentials: # /usr/lib/nis/nisclient –vco <userid>

Remove NIS+ credentials:
nisaddcred –r <userid>.some.domain

Cycle NIS+ Daemon:
Solaris 9: /usr/sbin/rpc.nisd
Solaris 10: svcadm restart svc:/network/rpc/nisplus:default

Change NIS+ Table:
/usr/cm/sa/bin/edit_table <table_name>

The Cold-Start File and Directory Cache: cold_start

When a client is initialized, it is given a cold-start file. The cold-start file gives a client a copy of a directory object that it can use as a starting point for contacting servers in the namespace. This directory object contains master and replica server address, public keys, and other information. A cold-start file is used to initialize a client's directory cache. The directory cache, managed by an NIS+ facility called the cache manager, stores the directory objects that enable a client to send its requests to the proper servers.

Create NIS+ local and DES key credentials
---
1. Add local credentials:
nisaddcred –p <UID> -P <userid>.<domain>. Local
nisaddcred –p unix.<UID>@<domain> -P <userid>.<domain>. Des
Enter an initial password at the prompt.