NAME

nisinit – NIS+ client and server initialization utility

SYNOPSIS

nisinit –**r**
nisinit –**p Y | D | N** *parent_domain host* . . .
nisinit –**c** –**H** *host* | –**B** | –**C** *coldstart*

AVAILABILITY

SUNWnisu

DESCRIPTION

nisinit initializes a machine to be a NIS+ client or an NIS+ root master server. It may be easier to use **nisclient**(1M) or **nisserver**(1M) to accomplish this same task.

OPTIONS

–**r** Initialize the machine to be a NIS+ root server. This option creates the file **/var/nis/data/root.object** and initialize it to contain information about this machine. It uses the **sysinfo**(2) system call to retrieve the name of the default domain.

To initialize the machine as an NIS+ root server, it is advisable to use the "–**r**" option of **nisserver**(1M), instead of using "**nisinit -r**".

–**p Y | D | N** *parent_domain host* . . .
This option is used on a root server to initialize a **/var/nis/data/parent.object** to make this domain a part of the namespace above it. Only root servers can have parent objects. A parent object describes the namespace ''above'' the NIS+ root. If this is an isolated domain, this option should not be used. The argument to this option tells the command what type of name server is serving the domain above the NIS+ domain. When clients attempt to resolve a name that is outside of the NIS+ namespace, this object is returned with the error **NIS_FOREIGNNS** indicating that a name space boundary has been reached. It is up to the client to continue the name resolution process.

The parameter *parent_domain* is the name of the parent domain in a syntax that is native to that type of domain. The list of host names that follow the domain parameter are the names of hosts that serve the parent domain. If there is more than one server for a parent domain, the first host specified should be the master server for that domain.

Y Specifies that the parent directory is a NIS version 2 domain.

D Specifies that the parent directory is a DNS domain.

N Specifies that the parent directory is another NIS+ domain. This option is useful for connecting a pre-existing NIS+ subtree into the global namespace.

Note that in the current implementation, the NIS+ clients do not take advantage of the –**p** feature. Also, since the parent object is currently not replicated on root replica servers, it is recommended that this option not be used.

–c Initializes the machine to be a NIS+ client. There are three initialization options available: initialize by coldstart, initialize by hostname, and initialize by broadcast. The most secure mechanism is to initialize from a trusted coldstart file. The second option is to initialize using a hostname that you specify as a trusted host. The third method is to initialize by broadcast and it is the least secure method.

–C *coldstart*
Causes the file *coldstart* to be used as a prototype coldstart file when initializing a NIS+ client. This coldstart file can be copied from a machine that is already a client of the NIS+ namespace. For maximum security, an administrator can encrypt and encode (with **uuencode**(1C)) the coldstart file and mail it to an administrator bringing up a new machine. The new administrator would then decode (with **uudecode**), decrypt, and then use this file with the **nisinit** command to initialize the machine as an NIS+ client. If the coldstart file is from another client in the same domain, the **nisinit** command may be safely skipped and the file copied into the **/var/nis** directory as **/var/nis/NIS_COLD_START**.

–H *hostname*
Specifies that the host *hostname* should be contacted as a trusted NIS+ server. The **nisinit** command will iterate over each transport in the **NETPATH** environment variable and attempt to contact **rpcbind**(1M) on that machine. This hostname *must* be reachable from the client without the name service running. For IP networks this means that there must be an entry in **/etc/hosts** for this host when **nisinit** is invoked.

–B Specifies that the **nisinit** command should use an IP broadcast to locate a NIS+ server on the local subnet. Any machine that is running the NIS+ service may answer. No guarantees are made that the server that answers is a server of the organization's namespace. If this option is used, it is advisable to check with your system administrator that the server and domain served are valid. The binding information can be dumped to the standard output using the **nisshowcache**(1M) command.

Note that **nisinit –c** will just enable navigation of the NIS+ name space from this client. To make NIS+ your name service, modify the file **/etc/nsswitch.conf** to reflect that. See **nsswitch.conf**(4) for more details.

RETURN VALUES

nisinit returns **0** on success and **1** on failure.

EXAMPLES

This example initializes the machine as an NIS+ client using the host *freddy* as a trusted server.

example# nisinit –cH freddy

This example sets up a client using a trusted coldstart file.

example# nisinit –cC /tmp/colddata

This example sets up a client using an IP broadcast.

example# nisinit −cB

This example sets up a root server.

example# nisinit −r

ENVIRONMENT

NETPATH This environment variable may be set to the transports to try when contacting the NIS+ server (see **netconfig**(4)). The client library will only attempt to contact the server using connection oriented transports.

FILES

/var/nis/NIS_COLD_START
This file contains a list of servers, their transport addresses, and their Secure RPC public keys that serve the machine's default domain.

/var/nis/data/root.object
This file describes the root object of the NIS+ namespace. It is a standard XDR-encoded NIS+ directory object that can be modified by authorized clients using the **nis_modify**( ) interface.

/var/nis/data/parent.object
This file describes the namespace that is logically above the NIS+ namespace. The most common type of parent object is a DNS object. This object contains contact information for a server of that domain.

/etc/hosts Internet host table.

SEE ALSO

nis+(1), **uuencode**(1C), **nisclient**(1M), **nisserver**(1M), **nisshowcache**(1M), **sysinfo**(2), **hosts**(4), **netconfig**(4), **nisfiles**(4)