NAME rpc.nisd, nisd – NIS+ service daemon

SYNOPSIS **/usr/sbin/rpc.nisd** [**–ACDFhlv**] [**–Y** [**–B** [**–t** *netid*]]] [**–d** *dictionary*] [**–L** *load*]
[**–S** *level*]

AVAILABILITY SUNWnisu

DESCRIPTION

The **rpc.nisd** daemon is an RPC service that implements the NIS+ service. This daemon must be running on all machines which serve a portion of the NIS+ namespace.

rpc.nisd is usually started from a system startup script.

The –**B** option causes **rpc.nisd** to start an auxiliary process, **rpc.nisd_resolv**, which provides ypserv compatible DNS forwarding for NIS host requests. **rpc.nisd_resolv** can also be started independently. See **rpc.nisd_resolv**(1M) for more information on using **rpc.nisd_resolv** independently.

OPTIONS

–**A** Authentication verbose mode. The daemon logs all the authentication related activities to **syslogd**(1M) with **LOG_INFO** priority.

–**C** Open diagnostic channel on **/dev/console**.

–**D** Debug mode (don't fork).

–**F** Force the server to do a checkpoint of the database when it starts up. Forced checkpoints may be required when the server is low on disk space. This option removes updates from the transaction log that have propagated to all of the replicas.

–**h** Print list of options.

–**v** Verbose. With this option, the daemon sends a running narration of what it is doing to the syslog daemon (see **syslogd**(1M)) at **LOG_INFO** priority. This option is most useful for debugging problems with the service (see also –**A** option).

–**Y** Put the server into NIS (YP) compatibility mode. When operating in this mode, the NIS+ server will respond to NIS Version 2 requests using the version 2 protocol. Because the YP protocol is not authenticated, only those items that have read access to nobody (the unauthenticated request) will be visible through the V2 protocol. It supports only the standard Version 2 maps in this mode (see –**B** option and **NOTES** in **ypfiles**(4)).

–**B** Provide ypserv compatible DNS forwarding for NIS host requests. The DNS resolving process, **rpc.nisd_resolv**, is started and controlled by **rpc.nisd**. This option requires that the **/etc/resolv.conf** file be setup for communication with a DNS nameserver. The **nslookup** utility can be used to verify communication with a DNS nameserver. See **resolv.conf**(4) and **nslookup**(1M).

–t *netid* Use *netid* as the transport for communication between **rpc.nisd** and **rpc.nisd_resolv**. The default transport is **ticots**(7D) (**tcp** on SunOS 4.x systems).

–d *dictionary*
Specify an alternate dictionary for the NIS+ database. The primary use of this option is for testing. Note that the string is not interpreted, rather it is simply passed to the **db_initialize** function. See **nis_db**(3N).

–L *number* Specify the ''load'' the NIS+ service is allowed to place on the server. The load is specified in terms of the *number* of child processes that the server may spawn. This *number must* be at least 1 for the callback functions to work correctly. The default is 128.

–S *level* Set the authorization security level of the service. The argument is a number between 0 and 2. By default, the daemon runs at security level 2.

- **0** Security level 0 is designed to be used for testing and initial setup of the NIS+ namespace. When running at level 0, the daemon does not enforce any access controls. Any client is allowed to perform any operation, including updates and deletions.
- **1** At security level 1, the daemon accepts both **AUTH_SYS** and **AUTH_DES** credentials for authenticating clients and authorizing them to perform NIS+ operations. This is not a secure mode of operation since **AUTH_SYS** credentials are easily forged. It should not be used on networks in which any untrusted users may potentially have access.
- **2** At security level 2, the daemon accepts only **AUTH_DES** credentials for authentication and authorization. This is the highest level of security currently provided by the **NIS+** service. This is the default security level if the **–S** option is not used.

EXAMPLES

The following example sets up the NIS+ service.

example% rpc.nisd

The following example sets up the NIS+ service, emulating YP with DNS forwarding.

example% rpc.nisd –YB

ENVIRONMENT

NETPATH The transports that the NIS+ service will use can be limited by setting this environment variable (see **netconfig**(4)).

FILES

/var/nis/parent.object This file contains an XDR encoded NIS+ object that describes the namespace above a root server. This parent namespace may be another NIS+ namespace or a foreign namespace such as one served by the Domain Name Service. It is only present on servers that are serving the root of the namespace.

/var/nis/root.object This file contains an XDR encoded NIS+ object that describes the root of the namespace. It is only present on servers that are serving the root of the namespace.

/etc/init.d/rpc initialization script for NIS+

SEE ALSO

nis_cachemgr(1M), **nisinit**(1M), **nissetup**(1M), **nslookup**(1M), **syslogd**(1M), **rpc.nisd_resolv**(1M), **rpc.nispasswdd**(1M), **nis_db**(3N), **netconfig**(4), **nisfiles**(4), **resolv.conf**(4), **ypfiles**(4), **ticots**(7D)