

NeXT System Administration Release Notes: NetInfo

This file contains release notes for NetInfo. Items specific to later releases are listed first, and notes from previous releases follow.

Notes Specific to Release 3.3

Changes to NetInfo internals

These changes improve performance and enhance system administration:

- Clone servers check the master database to make sure the master has correct configuration information for the clone. Errors are written to **syslog**.
- A new NetInfo binding strategy improves performance in wide-area networks. Servers preferentially bind to parent servers on the local host, if possible. If binding fails or isn't possible on the local host, a second bind request is sent to all servers on the same subnet, including servers on the local host. If binding fails or isn't possible on the local subnet, then bind requests are sent to all known parent servers.
- The text of NetInfo messages sent to **syslog** have been improved to make them easier to

understand.

- NetInfo client processes detect rogue servers at the time of their initial connection. Clients print a log message if they detect a rogue server, and they reconnect to another server rather than remain connected to the original server.
- NetInfo clients reconnect to a new server using the same strategy outlined above for binding. A client tries to connect to a server first on the local host, then on the local subnet, and finally to any server.
- **lookupd** has been modified to send a trace of all its current NetInfo server connections to **syslog** upon receipt of a **USR1** signal. The trace is logged at **LOG_INFO** priority.
- **lookupd** prints all arguments to its log file if logging is enabled. (In previous releases, some arguments weren't logged).
- **lookupd** logging can be disabled or reenabled on the fly. You can toggle logging on and off by sending **lookupd** a **USR2** signal. If logging is started with the **-I** option rather than **-L**, logging is disabled until you start it with a **USR2** signal.
- PIDs for **nibindd** and **lookupd** are written to **/etc/nibindd.pid** and **/etc/lookupd.pid**.
- **nibindd** has been changed to allow remote operations, including creating and deleting clone servers. You can access this new functionality in NetInfoManager.

Domain Name System (DNS) configuration

- Resolver configuration can be maintained in NetInfo. The resolver searches for the NetInfo directory **/locations/resolver** following the typical search strategy of checking the local domain and then climbing the NetInfo hierarchy. The directory should contain a domain property that has the DNS domain name as its value as well as a nameserver property that contains the IP address of one or more nameservers as values.

New command-line utilities

There are new **niutil** options in Release 3.3 as well as three new NetInfo command-line utilities:

- The **niutil** command supports the following new operations:

±appendprop	append a value to a property
±insertprop	append a value to a property only if it's a new value
±insertval	insert a value at a given index in a property
±destroyval	remove a value at a given index in a property
±renameprop	rename a property key
±rparent	get a server's current NetInfo parent
±resync	force master and clone servers to resynchronize databases
±statistics	print server version number and database checksum
±readprop	prints the values associated with a single property in the specified directory
±readval	prints a single value associated with a property in the specified directory
±domainname	prints a domain's name

You can also specify a user name and password on the command line for authentication, and you can specify a connection timeout value. For information on how to do this, invoke the **niutil** command without specifying any arguments and see the resulting usage message.

- You can use the **nigrep** command to search for regular expressions in a NetInfo domain.
- You can use the **nifind** command to search for directories with given key-value pairs. **nifind** normally searches the local NetInfo domain, then climbs the hierarchy to the root domain. You can also use **nifind** to search for directories in an entire hierarchy.
- You can use the **nireport** command much as you use **niutil -list**, but you can specify multiple

keys. **nireport** produces a tab-separated list of the values for those keys in all subdirectories of a given directory.

Bug fixes

The following changes and bug fixes been made to NetInfo since Release 3.2.

- **nibindd** correctly restarts all NetInfo servers on a host upon receipt of a HUP signal.
- The `_writers` property has been removed from all NetInfo directories except **/localconfig/screens** and **/locations/renderers/localhost**. This improves NEXTSTEP system security.
- Access to **/etc/netinfo** is limited to the **root** user and members of the **wheel** group.

Known Problems

- A bug in **netinfod** causes the server to incorrectly manage a directory's subdirectory list if you create a subdirectory with no properties. Avoid removing all the properties from a directory using **niutil** or NetInfoManager. If you're using the **netinfo(3)** access library to write programs that access NetInfo, don't create zero-sized directories or remove all the properties from a directory.

Notes Specific to Release 3.2

New Features

The following features have been added to NetInfo since Release 2.0.

Performance improvements

A number of performance improvements have been made to ensure that NetInfo scales efficiently to networks containing hundreds or thousands of computers.

- The protocol between NetInfo master and clone servers has been improved to eliminate excessive data transfers, and the distribution of NetInfo changes has been streamlined. When a number of NetInfo changes are made in succession, as often occurs when high-level applications are used, **netinfod** will coalesce the individual changes into a single composite update. Since only one transaction with each clone server is needed to distribute the composite update, the overhead associated with updating clones is reduced.

In addition, the distribution of updates is now multi-threaded. Modifications made to the master database are distributed immediately to clones; an update need not wait for prior updates to finish before it is handled. Update threads operate independently, except that updates are guaranteed to arrive at each individual clone in the correct order.

- The disk format for NetInfo databases has changed to increase the default record size. In most cases, this reduces the number of files in a database by about 90 percent and decreases the cost of large searches by about 25 percent. Release 3 NetInfo is compatible with both new and old database formats, and all versions of NetInfo can be used together on a network, regardless of the revision level or database format of the individual servers. All configurations, including master and clone servers which use different versions of software and different database formats, are supported.
- In order to deter NetInfo servers from binding to inappropriate parent servers over a slow network link, NetInfo servers now check for the existence of a local clone of the parent domain before broadcasting to find a suitable parent server. Since clients typically inherit the bindings

of their NetInfo servers, this can help reduce the frequency with which NetInfo clients bind to relatively distant servers.

- Two sources of excess network traffic have been quashed: the automatic reload of **lookupd**'s user information cache every half hour, and the execution of **atrun** by **cron** every fifteen minutes. The user information cache in Release 3 is not reloaded unless actively used, and **cron** has been modified to avoid using library routines which access NetInfo when possible.
- Several caches have been added to **lookupd**, the NetInfo daemon which accesses NetInfo, NIS, and DNS information on behalf of the UNIX libraries. The most frequent requests seek information about the current host, the currently logged-in user, and the **root** account. In Release 3, **lookupd** answers these requests from caches, avoiding use of the network completely. The logged-in user and **root** caches persist for at most twenty minutes; they are explicitly reset when a user logs in or out, or when a password change is made. Local host information is never reset, and the computer must be restarted if its Internet address is changed.
- UNIX groups, mount points, printer database entries, and password file entries are also cached, but **lookupd** checks NetInfo to be sure the cached information is valid before these caches are used. In Release 3, about 80 percent of **lookupd** requests are satisfied from a cache.
- If NetInfo is the only source for user and group information on your network, some additional optimizations are possible due to the database-like nature of NetInfo. To determine which UNIX groups a user belongs to, for example, it is no longer necessary to enumerate all groups across the network. This helps to speed up logging in and the UNIX **su** command.

Other changes and enhancements

- **lookupd** supports three ways of handling the **getpwent** family of library routines. By default,

lookupd maintains a cache of user data which is checked for accuracy only after servicing a client request. If the data have become outdated and a certain amount of time has passed since the last update (thirty minutes by default, settable with the **-m** option), the cache is reloaded. Since the client request is answered before the cache is reloaded, it is possible for a second enumeration of user data to yield different results. The **-f** option forces the cache to be reloaded on a regular schedule, and **-m 0** turns it off completely.

- **lookupd** may be passed a **-L filename** argument to ask that all lookup requests be logged to the specified file. Cache hits are marked with an asterisk. The **lookupd** log can become quite large over time.
- **niload** now allows the use of a tagged domain specification.
- The interface between the UNIX libraries and **lookupd** is considered private API, and the header files which define it have been removed.
- The file **/etc/exports** is obsolete, and export information is now stored in NetInfo. The **niload** and **nidump** commands can be used to load or dump export files in the standard UNIX format.
- Each user record in NetInfo now has a user-writable subdirectory called "info", which may be used by applications to store configuration information. **niload** will create these info subdirectories when a UNIX passwd file is loaded.
- To force **lookupd** to rebind its connections to NetInfo servers, it can be restarted by sending it a SIGHUP (signal number 1). To restart all NetInfo servers on a computer, kill **nibindd** with a SIGTERM and restart it by running **/usr/etc/nibindd** from a shell.
- Release 3 includes several security enhancements which may be enabled through NetInfo. To turn on security features, create a security_options property in the **root** directory of a NetInfo domain and add one or more of the following values:

login_auditing	Asks that all successful logins, including those to the workspace, be recorded in the standard UNIX files, and that unsuccessful logins be reported through syslog.
secure_passwords	Raises the standards of obscurity which passwords must meet in order to be accepted by the Preferences application and the passwd command. With secure_passwords on, all passwords must be at least six characters long and contain non-alpha characters.
lockout	When a login attempt fails, progressively increases the delay enforced between login attempts.
all	Selects all current and future options.

There is no way to turn off a security option. A computer uses all the security options enabled in its local domain and all of its parent domains.