

NeXT System Administration Release Notes: lookupd

Notes From Release 4.2

Changes to lookupd

A new configuration parameter has been added to **lookupd**. When cache validation is enabled, **lookupd** validates items retrieved from its cache. A fast validation check is implemented by each lookup agent independently. For example, the NetInfo agent compares the database checksum associated with a cached item with the current database checksum. In the new version of **lookupd**, the NetInfo agent keeps the last checksum fetched from each domain, and uses that value for a short time period instead of re-fetching

the checksum on each validation. The default time interval is 15 seconds. The time interval may be set by adding a "ValidationLatency" property to the lookupd configuration directory (**//locations/lookupd**) in NetInfo. The value of the property must be an integer. Setting ValidationLatency to 0 seconds causes validation against the source information system on every fetch from cache (as it is in OPENSTEP for Mach 4.0 and 4.1). Setting the value to a larger number effectively "caches" NetInfo checksums (or, in the case of the NIS agent, NIS map order numbers) for the indicated number of seconds. The ValidationLatency property is used by the NetInfo and NIS agents.

lookupd was updated in OPENSTEP for Mach Release 4.0. It has a much more sophisticated cache than the previous version of lookupd, with a good bit of design that should make it a winner on slow lines (SL/IP, PPP, or ISDN).

It is highly configurable. In the absence of any configuration, lookupd continues to function like the old lookupd. However, there's lots that you can do with the new version. Most of the configuration settings let you change the default lookup strategy and to control the various cache parameters.

The command line options used by earlier versions of lookupd have been superseded by the new configuration information that lookupd reads from NetInfo when it starts. In this release, lookupd does not restart in response to a SIGHUP signal (SIGHUP is ignored).

Notes From Release 4.1

Lookup Strategy

Internally, lookupd uses a set of software "agents" to get information. There are agents for NetInfo, Network Information Services (NIS), the Domain Name System (DNS), Flat Files, and a cache agent. When lookupd searches for information about an item, it queries the agents in order until the item is found. By default, lookupd first queries its cache agent, and then NetInfo. If the item is a host or network, lookupd will query the DNS agent last.

The lookup order is now configurable. For example, you might specify that lookupd queries the Cache, then NetInfo, then the Flat Files, then NIS. You may also change the lookup order for a particular category of item. The known categories are users, groups, hosts, networks, services, protocols, rpcs, mounts, printers, bootparams, bootp, aliases, and netgroups. You can set the lookup order (and some other configuration options) for all categories, and override them for individual categories.

Caching

There are caches for all categories of lookups. The caches are unlimited in capacity (although you can set a maximum size, see below). The default lookup order (see above) starts with the Cache agent for each lookup category. Caching may be disabled for all categories or for specific categories by removing the Cache agent from the lookup order.

Validation

When lookupd retrieves an item from cache, it validates the cached entry before returning it. If the entry is invalid, a fresh copy is fetched to replace the stale one in cache.

Validation is enabled by default, but may be disabled for all categories, or for specific categories. For example, since host names and addresses very rarely change, you might want to turn off cache validation in the host cache to save time and reduce network traffic. This is especially useful on slow network lines. If network access is fast, cache validation is inexpensive, and ensures that all data is correct.

The cache validation strategy in the NetInfo, NIS, and Flat File agents has been to make validation as fast and inexpensive as possible. At present, there is no cache validation for DNS. That means that objects retrieved from DNS will only be retrieved later on from the cache if cache validation is disabled for hosts.

Validation checks are done by testing if anything has changed in the data source. For

example, if you add a new host to a NetInfo domain, any cached entries from that domain will be invalidated, including user entries, host entries, and so on.

When a lookup agent (for example, the Flat File agent) fetches an item, it adds a validation "stamp" to the object before it's placed in the cache. In the case of the Flat File agent, the validation stamp is the file name and the file's last modification time. Later on, when lookupd retrieves an item from cache, it will ask the agent that originally fetched the object to validate it. For example, the Flat File agent validates an object by comparing the saved file modification time-stamp to the actual file's time-stamp. If the object is invalid, a fresh copy is fetched to replace the stale one in cache.

Time-to-live

When an object is placed in a cache, it is given a time-to-live. The default is 12 hours. Making things expire more quickly will cause the caches to stay smaller, but will result in more network traffic. Whenever an object is validated in a cache, its time-to-live is reset.

Configuration

All the configuration parameters are kept in NetInfo under the **//locations/lookupd** directory,

stored as values for various keys. There may also be subdirectories for each lookup category. For example: **//locations/lookupd/hosts**, **//locations/lookupd/users**, **//locations/lookupd/printers**, and so on. lookupd will climb the NetInfo hierarchy (starting at the local domain) to find this directory. If it doesn't find it, it will use a default configuration (see below).

The table below lists all the keys and the allowed values for each:

LogFile	Name of a log file that contains a copy of all messages sent to syslog. There is no default (i.e. no log file is kept).
MaxThreads	Maximum number of threads in the query dispatcher. The default is 16. Under very heavy load, only 5 or 6 threads are used, so 16 is more than enough.
MaxIdleThreads	When a thread finishes servicing a query, it will usually go back to the message queue to wait for another query. This setting limits the maximum number of idle threads waiting on the queue. If a thread finishes servicing a query and MaxIdleThreads are already waiting on the queue, the thread will exit. The default value is 16.
MaxIdleServers	The dispatcher uses a "server" object to actually answer a client

lookup. One server is required for each busy thread. The dispatcher keeps a pool of servers so that they can be re-used. This setting limits the maximum number of servers in the pool, waiting for a query to answer. The default value is 16.

ValidateCache	This boolean value determines whether cache validation is enabled for all cache categories. The default is "YES". Use "NO" to disable validation. The setting of this value may be over-ridden for specific cache categories (see below).
CacheCapacity	Maximum number of objects in the cache for each category (e.g. this many users, this many hosts, ...). When a cache is full, least-recently-used objects are removed when space is required. By default, there is no limit to the cache size.
TimeToLive	Time to live (measured in seconds) in cache when added. The default is 43200 seconds (12 hours).
lookupOrder	Sets the lookup order for all categories, although you may override this for specific categories. This key takes multiple values. The default for all categories except host and network lookups is CacheAgent then NIAgent. For hosts and networks, the default

lookup order is CacheAgent, NIAgent, then DNSAgent. Two other agents, YPAgent and FFAgent, are available for NIS and Flat File lookups.

Configuration options for specific categories reside in the directories:

/locations/lookupd/users, .../groups, .../hosts, .../networks, .../services, .../protocols, .../rpcs, .../mounts, .../printers, .../bootparams, .../bootp, .../aliases, and .../netgroups. The options that can be set per category are ValidateCache, CacheCapacity, TimeToLive, and lookupOrder.

Here's a sample configuration:

```
# nidump -r /locations/lookupd /
name = lookupd;
LogFile = /usr/adm/lookupd.log;
MaxThreads = 12;
lookupOrder = (CacheAgent, NIAgent);
CHILDREN = ({
    name = users;
    LookupOrder = (CacheAgent, NIAgent, YPAgent, FFAgent);
}, {
    name = hosts;
    LookupOrder = (CacheAgent, NIAgent, DNSAgent);
    ValidateCache = NO;
```


});