

1

Optimizing Account Management 2.1

Account Management is a directory-enabled application that simplifies and unifies the management of user profiles on Windows* NT*, Solaris*, and Linux* networks. It leverages all the scalability, utility, and extensibility of NDS® eDirectory™ and adds crucial integration capability. With Account Management, you can eliminate many of the complexities of administering a mixed-platform network while smoothing over compatibility issues.

This guide contains information for optimizing configurable parameters in order to improve the performance of Account Management on Solaris and Linux systems.

The Account Management components should be installed on all Solaris/Linux systems that needs to be administered using eDirectory™. The installation of Account Management configures the system to use eDirectory instead of NIS, NIS+, or local /etc/passwd files. You can modify the system to use NIS or NIS+, if desired. You need to ensure that the names of UNIX* users, groups, and workstations are unique.

Performance Improvement Settings

The following is a list of the settings that will enable you to realize maximum performance gains from your installation of Account Management on Solaris and Linux. The guide is organized to first discuss settings on the Account Management workstation followed by the eDirectory server settings.

- ♦ “nscd Settings” on page 2
- ♦ “nds_uamcd Settings” on page 2
- ♦ “Access Speeds” on page 4
- ♦ “Setting the Preferred Server” on page 4

- ♦ “eDirectory Server Setting” on page 5

nscd Settings

Make sure that the Name Service Cache Daemon (nscd) is installed and running at all times.

Linux and Solaris systems provide a cache daemon, nscd, which provides a cache for the most common name service requests. The nscd daemon caches the profiles of user and group entries, improving the performance of operating system commands, such as, ls and ps. The nscd daemon can be configured using the /etc/nscd.conf file.

nscd Parameters

The following is a list of guidelines for the various nscd parameters.

- ♦ The value of the *suggested-size* parameter for passwd database should be a prime number greater than or equal to 1/4th of the users expected to be used.
- ♦ Increase the *positive-time-to-live* parameter for passwd and group database to, say, 3600 (1hour). Note that this might result in the cache reflecting stale information if the eDirectory database is modified.
- ♦ Increase the *negative-time-to-live* parameter for passwd and group database to, say, 600 (10 minutes). But this might also result in a newly added user not appearing for up to 10 minutes, if an attempt to read that user's non-existent information was done within 10 minutes.
- ♦ Increase the value of the *keep-hot-count* parameter to 200. nscd will keep upto 200 entries in the cache refreshed all the time.

NOTE: If any of the above parameters are changed, nscd has to be restarted.

nds_uamcd Settings

Make sure that the caching daemon, nds_uamcd, is running at all times.

Accessing eDirectory to get name service entries might cause a degradation in performance. nds_uamcd can persistently cache entries for workstations that are rebooted regularly. The persistent cache will improve access times, especially in setups with a large number of workstations.

You can enable or disable persistent caching to specify what information you want cached. If you enable persistent caching, all user profiles, group profiles and the fully distinguished names (FDNs) of User objects will be cached. If persistent caching is disabled, only the User FDNs will be cached.

After configuring account management, since persistent caching is enabled by default, the uam cache daemon automatically retrieves all user and group entries from eDirectory and builds the cache. If you want to disable persistent caching, set the parameter `n4u.uam.enable-persistent-cache=No` in the file `/etc/nds.conf` and restart the `nds_uamcd` daemon.

The settings for persistent caching, and the size of the cache for FDNs, user profiles and group profiles can be specified in the `/etc/nds.conf` file. Since the `nds_uamcd` daemon does not rescan the configuration file, you need to restart the `nds_uamcd` daemon after you make any modification to the configuration file.

Whenever the `pam_nds` and the `nss_nds` modules access eDirectory to get a User object, the `nds_uamcd` daemon updates the cache. `nds_uamcd` searches the cache before accessing eDirectory, thereby making the access quicker.

nds_uamcd Parameters

The following is a list of guidelines for the various `nds_uamcd` parameters.

- ◆ You can set the `n4u.uam.enable-persistent-cache` parameter to enable or disable persistent caching, to specify what information you want cached.
- ◆ You can use the `n4u.uam.suggested-size`, `n4u.uam.user-hash-size` and `n4u.uam.group-hash-size` parameters to set the cache size for user FDNs, user profiles and group profiles respectively. The cache size should be a prime number greater than or equal to 1/4th of the users in that workstation.
- ◆ Increase the value of the `n4u.uam.positive-time-to-live` parameter to, say, 3600 (1hour). Note that this might result in the cache reflecting stale information if the eDirectory database is modified.
- ◆ The `n4u.uam.persistent-cache-entries-aging-interval` parameter can be set to specify the interval after which the user and group entries will be deleted from the persistent cache.
- ◆ The `n4u.uam.persistent-cache-entries-refresh-period` parameter can be set to specify how periodically user and group entries should be refreshed from the eDirectory database. Setting a large value will result in less

network traffic and less load on the server, but the cache will reflect stale information if the eDirectory database is modified within that period.

- ♦ The *n4u.uam.persistent-cache-entries-refresh-flag* parameter can be set to specify whether all user and group entries or only those used in the current boot session are to be refreshed.
- ♦ Increase the value of the *n4u.uam.num-threads* parameter according to the number of concurrent applications running.
- ♦ Increase the value of the *n4u.uam.keep-hot-count* parameter to increase the number of entries that are always kept refreshed in the cache. The value should preferably be an integer, approximately equal to the number of entries used during the day, in your setup.

The *nds_uamcd* parameters can be configured using the *uamconfig* utility. Refer to the *nds_uamcd* man page for more details.

Access Speeds

- ♦ User or group profiles which are not inside the Account Management partition root will experience slower access rates. If possible, create all users in the Account Management partition root.
- ♦ The number of users in the same eDirectory group affects performance.

Setting the Preferred Server

Set the preferred server name in the */etc/nds.conf* file. This helps to locate the server faster than using the Service Location Protocol (SLP).

Account Management first communicates with the preferred server and tries to resolve all eDirectory queries. If the preferred server is down or does not correspond to the referred tree, Account Management tries to discover names using SLP. With the preferred server set, the name resolution mechanism is DNS followed by SLP. Typically, DNS-based name resolution is faster than SLP-based name resolution (especially in the absence of SLP DAs).

For example, the preferred server can be used to bypass SLP if all the User, Group, UNIX Workstation and UNIX Config objects are located in the same partition. In such a case, the preferred server can be set to the eDirectory server (Solaris or Linux) holding the relevant replica. Note that if Account Management and eDirectory Server are installed on the same system, the preferred server is automatically set to the local eDirectory Server.

eDirectory Server Setting

To optimize the performance of Account Management, you might need to specify the following settings on the eDirectory server also.

Creating Indexes

For optimal performance of Account Management, create indexes using ConsoleOne™1.2d or above. Add the uidNumber and gidNumber attributes for each server accessed by Account Management. These indexes are automatically created for the preferred server during installation.

To create indexes for other servers:

- 1** Right-click the server object > select Properties.
- 2** Select the Indexes tab > Add.
- 3** Enter a name for the index in the dialog box next to Index Name.
- 4** Select the uidNumber and gidNumber attributes from the drop down list of attributes.
- 5** Select Value from the drop down list next to Rule.

The status of the indexing operation will change from Bringing Online to Online once the process is completed.

6 Place Book Title Here