INTEGRATION OF A NETAPP STORAGE SYSTEM WITH A UNIX BASED LDAP SERVER

H.T. Sun, Network Appliance, Inc.

April 2006, TR-3464

TABLE OF CONTENTS

1. INTRODUCTION **3**

2. BENEFITS **3**

3. USING LDAP TO RETRIEVE USER INFORMATION **4**

4. USING LDAP FOR AUTHORIZATION AND AUTHENTICATION **5**

4.1 SETTING UP LDAP ON NETAPP STORAGE SYSTEM **6**

4.2 NAME SERVICE SWITCH (NSS) MAPPING **11**

4.3 SASL AND SSL **11**

4.4 USER MAPPING **14**

4.5 MAKING YOUR LDAP DEPLOYMENT MORE SECURE **16**

4.6 OTHER LDAP OPTIONS **17**

5. CONCLUSION **18**

APPENDIX **19**

A. GLOSSARY **19**

B. REFERENCES **20**

TABLE OF FIGURES

Figure 1) User Mapping in LDAP 5

Figure 2) NetApp Storage System Uses LDAP For Authorization and Authentication 7

Figure 3) Example of CA Root Certificate 13

Figure 4) LDAP User Mapping 15

1. INTRODUCTION

The Lightweight Directory Access Protocol (LDAP) is a common protocol interface to network directory services (NDS). Widely deployed NDSs are Domain Name Service (DNS), NIS (Network Information Service), etc. They provide the clients with information such as host IPs, usernames, passwords, home directories, etc.

LDAP is a network protocol for accessing information directories. It is based on the standards contained within the X.500 standard, but is much simpler and very efficient in serving data that does not need to be updated very often. LDAP runs over TCP/IP or other connection oriented transfer services. LDAP has the potential to consolidate existing NDSs into a single directory that can be accessed by LDAP clients. The LDAP clients can be email clients looking for email addresses, web browsers looking for host IP addresses or NFS clients looking for user IDs or group IDs and automounter maps, etc. Furthermore, LDAP provides the users with a hierarchical view of the company's organizations. You can look up employees' phone numbers, their e-mail addresses and organization groups, etc.

With all the benefits mentioned above, LDAP is quickly penetrating directory services space and being used to provide a more secure and robust alternative to traditional NIS user directory services based on the diverse forms of information it can store. In the case of LDAPv3[1], additional data security technologies such as Transport Layer Security (TLS), Secure Sockets Layer (SSL) and Simple Authentication and Security (SASL, see RFC 2222) are introduced to ensure secure data channels between clients and the LDAP server.

NetApp storage devices fully support any LDAPv3 compliant NDS (Data ONTAP® version 6.4 and later), as well as the RFC 2307 schema definitions to create entities within an LDAP directory to provide NIS compliant objects. NetApp storage system is also integrated with LDAP software by various vendors such as OpenLDAP, Sun™ Directory Server, Novell eDirectory, RedHat Directory Server and Microsoft® Windows® 2000 /2003 Active Directory.

This paper will discuss the necessary steps to integrate with UNIX based LDAP directories to provide name resolution and authentication. Integration with Microsoft Active Directory is outside the scope of this document. You may want to refer to TR-3458 - Unified Windows and UNIX Authorization Using Microsoft Active Directory LDAP as a Directory Store for information on integration with AD based LDAP. Customers are encouraged to read TR 3387 – Security in NFS Storage Networks for an overview of LDAP deployment before continuing.

This paper is intended for security-aware technical audiences who have hands-on LDAP knowledge and plan to integrate NetApp storage systems with their existing LDAP infrastructure.

2. BENEFITS

A number of significant benefits can be achieved by migrating to LDAP and integrating it with NetApp storage systems. These benefits are summarized as follows:

- **Security**

  Before a NetApp storage system gives user access to a file, it first needs to make sure that the user has proper rights to the data the user intends to access. To do this, the NetApp storage system retrieves the user's information, for example, user ID and group memberships, from the LDAP server. It then compares the uid against the file's access permissions stored in the inodes to determine if the user has permissions to read or modify the file (Authorization). Without LDAP in place, the file system is prone to unauthorized data access. Additionally, as mentioned previously, security elements such as SASL and SSL can be utilized to secure data transmission to and from the LDAP server. By contrast, SSL and SASL are not available in NIS and could result in security breaches.

- **Centralized Management**

[1] Although LDAP is a relatively new protocol, it has gained wide acceptance in the industry since its inception in 1993. The current version of the LDAP protocol is LDAPv3, and due to interoperability issues, LDAPv2 should be avoided if possible. LDAPv1 was never standardized.

Without LDAP, a NetApp storage system could store user information in the local **`/etc/passwd`** file. This will not cause any issue if you only have to manage a few NetApp storage systems. However, if the number of NetApp storage systems becomes greater as data grows, management of the password files can become a daunting and time consuming task.

With LDAP, all the administrator needs to do is to add/change user information stored in the centralized database and the updated information will be available to every NetApp storage system participating in the LDAP store. This greatly reduces the administrator's burden and improves efficiency.

- **Consolidation**

  As described in the Introduction section, LDAP consolidates network directory services such as DNS and NIS into one single database. Traditionally, these service components are separate entities that have their own databases running across the network. As such, more IT resources are needed in terms of deployment, management and troubleshooting. By combining these network services, LDAP offers a cost effective solution for managing the network directory services.

- **Multiprotocol**

  NetApp storage systems allow unified data access of both UNIX and Windows users by utilizing the user mapping feature in Data ONTAP. By default, Data ONTAP stores the user mapping information in a local database **`/etc/usermap.cfg`**. Just like the **`/etc/passwd`** file, management of this local database can become more difficult as the number of NetApp storage systems grows. To solve this problem, NetApp storage systems support user mapping attributes defined in the LDAP schemas to ease the administrator's workload.

3. USING LDAP TO RETRIEVE USER INFORMATION

Let's take a look at how a NetApp storage system uses LDAP to retrieve user information in a multiprotocol environment. Suppose a user named James has a UNIX account ***james*** and a Windows account ***jim*** in Active Directory domain ***netapp***. Using the user mapping feature supported in Data ONTAP, James will be able to access his data from either a UNIX or Windows environment. The following flow chart (Figure 1) illustrates how the user mapping mechanism works in NetApp. The thick arrows indicate what really happens in our example.

In our example, a client requests to access a file by sending the uid to the NetApp storage system, if LDAP is the primary directory service in **`/etc/nsswitch.conf,`** the NetApp storage system then contacts the LDAP server to retrieve UNIX user ***james*** using the uid. If LDAP user mapping lookup is enabled (option **`ldap.usermap.enable`** set to on**`)`**, the NetApp storage system would further map ***james*** to Windows user ***jim*** in AD domain ***netapp***. Note that if LDAP user mapping lookup is not enabled, Data ONTAP uses local file **`/etc/usermap.cfg`** to resolve the mappings.

Once the user mapping is successful, NetApp storage system can then use the result to determine if a user has proper access rights to access data.

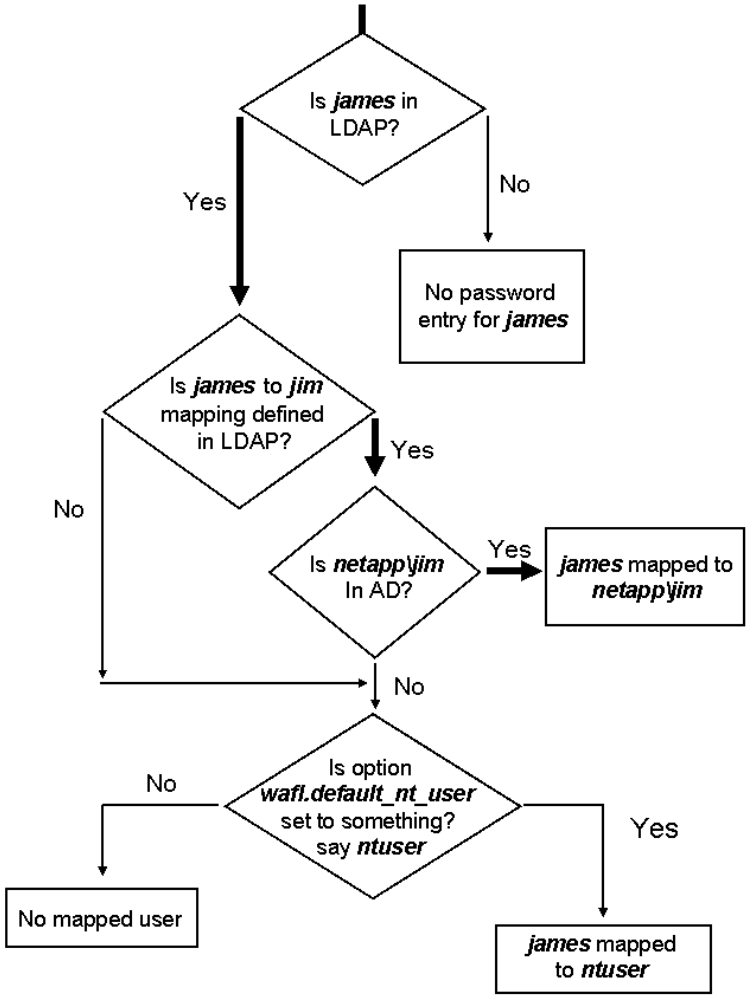

Figure 1) User Mapping in LDAP

4. USING LDAP FOR AUTHORIZATION AND AUTHENTICATION

It is often easy to get confused between authentication and authorization. Authorization is a process to determine if a user has the permissions to do certain things. By contrast, authentication is a process to verify a user's identity, in other words, to make sure a person is indeed who she or he said she or he is. For example, anyone over the age of 18 may apply for a driver's license from the Department of Motor Vehicles. The DMV authenticates the individual's identity and issues the driver's license. But whether the individual is authorized to purchase alcohol depends on if she or he is over the age of 21.

In the storage world, LDAP stores all the information needed to perform authorization or authentication tasks. Please note that it is the NetApp storage system, not the LDAP server, that performs authorization or authentication. Whenever necessary, the NetApp storage system contacts the LDAP server to retrieve information such as user passwords, user IDs, and group memberships for authentication/authorization purposes.

4.1 SETTING UP LDAP ON NETAPP STORAGE SYSTEM

Before we set up LDAP service in Data ONTAP, we need to understand how LDAP authorization and authentication work. Figure 2 below illustrates the steps involved for authorization using LDAP.

1. The client initiates a request to access data by providing user information such as numeric uid or gid to NetApp storage system.

2. NetApp storage system queries the LDAP server on the network. At this stage, Data ONTAP uses the logics described in Figure 1 above to proceed further. Note that in the process, both UNIX based LDAP and Active Directory based LDAP are contacted for proper mapping.

3. The LDAP server then returns the user information to NetApp storage system.

4. Finally, NetApp storage system compares the uid and gid against the access permissions stored in the inodes to determine if it should grant or deny access to the client.

It is important to note that steps 2 and 3 are usually skipped in a pure UNIX environment. In other words, if the clients and the volume/qtree are both UNIX based, Data ONTAP only needs to know the uid and gids (user may have membership with multiple groups) to authorize data access. Since the user information such as uid and gid is provided by the client, Data ONTAP does not need to query the LDAP server for other user information.

However, steps 2 and 3 will take place in the following situations:

- **When user mapping lookup is required –**
  Data ONTAP queries the LDAP server in a multiprotocol environment where lookups from UNIX to Windows or Windows to UNIX are necessary.

- **UNIX users access files with NTFS security style permissions exported through NFS –**
  Since the files exported through NFS could have NTFS security style permissions and exist in either an NTFS or Mixed security style volume/qtree, Data ONTAP needs to know appropriate Windows user mapping for the UNIX user in order to authorize the user for data access. For a detailed description regarding Data ONTAP security styles (UNIX, NTFS, Mixed), please refer to TR 3014.

- **NetApp storage system is configured to use LDAP for password authentication –**
  This scenario pertains to using LDAP for authentication during cifs setup process.

  ```
  fas3020*> cifs setup
  .............

  (1) Active Directory domain authentication (Active Directory domains only)
  (2) Windows NT 4 domain authentication (Windows NT or Active Directory
      domains)
  (3) Windows Workgroup authentication using the filer's local user accounts
  (4) /etc/passwd and/or NIS/LDAP authentication

  Selection (1-4)? [1]: 4
  ```

  In this scenario, suppose ldap is the primary directory service configured in **`/etc/nsswitch.conf`**. NetApp storage system will retrieve user password information from the LDAP directory store in steps 2 and 3 to authenticate the user.

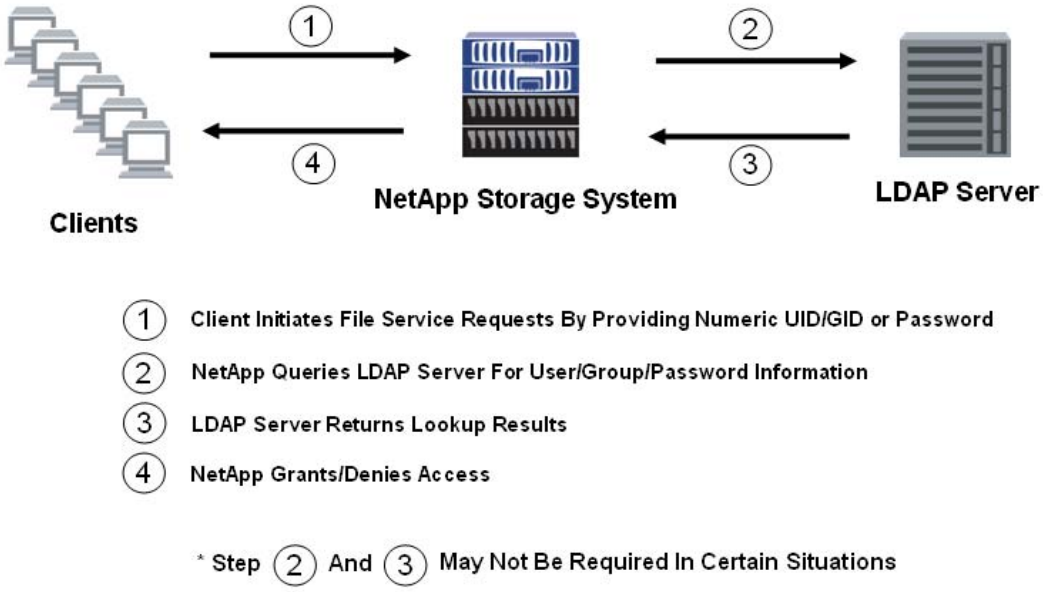

Figure 2) NetApp Storage System Uses LDAP For Authorization and Authentication

Now we are ready to configure NetApp storage system to perform LDAP lookups on user, group and netgroup. Please make sure that the primary directory services for these lookups are set up as ldap in **/etc/nsswitch.conf:**

```
fas3020*> rdfile /etc/nsswitch.conf
passwd: ldap nis files
group: ldap nis files
netgroup: ldap nis files
```

User Lookup

Since OpenLDAP is an open source product and runs on a majority of UNIX platforms, we will use OpenLDAP for the examples demonstrated below. Other LDAP servers should have similar schema definitions unless noted otherwise.

Let's say a user named James Dole has his information stored in the LDAP database in the LDAP Interchange Format (LDIF) as follows. LDIF is described in RFC 2849 and contains a number of attributes as in the example shown below. The attributes are either defined in your vendor-supplied schemas or can be customized by the LDAP administrator.

```
dn: cn=James Dole,ou=People,dc=netapp,dc=com
objectClass: posixAccount
objectClass: OpenLDAPperson
objectClass: shadowAccount
cn: James Dole
givenName: James
uid: james
title: Product and Partner Engineer
postalAddress: 495 East Java Dr, Sunnyvale, CA 94089
userPassword:: e01ENX1KSFFodkZpMWFJOXlvaHFFPVo1NmNRPT0=
mail: jdole@netapp.com
telephoneNumber: 408-822-6000
```

```
uidNumber: 6000
gidNumber: 300
homeDirectory: /home/james
loginShell: /bin/sh
```

We now need to configure NetApp storage system to retrieve above information from the LDAP server. Although there are many attributes pertaining to user James, the most important attributes for LDAP authorization are the **uid** and **uidNumber**. Suppose the search base is "**dc=netapp,dc=com**" and there are two LDAP servers – **ldap1.netapp.com** as the master server and **ldap2.netapp.com** as the replication server. You can specify a preferred LDAP server using the **ldap.servers.preferred** option to ensure that the preferred server is always contacted first. The preferred server has to be one of the servers listed in option **ldap.servers.** The following are the options that you need to specify in Data ONTAP in order to retrieve the information correctly.

```
fas3020*> options ldap.base "dc=netapp,dc=com"
fas3020*> options ldap.enable on
fas3020*> options ldap.port 389
fas3020*> options ldap.servers  "ldap1.netapp.com ldap2.netapp.com"
fas3020*> options ldap.servers.preferred ldap1.netapp.com
```

The following table describes the options in more detail:

Option	Description	Example
ldap.base	The base distinguished name to use for common LDAP lookup. You can also specify scope for each filter. The scope value determines how detailed the search results you want LDAP to display. The scope can be one of these three choices: **BASE**, **ONELEVEL** or **SUBTREE** <default>.	**dc=netapp,dc=com** **dc=netapp,dc=com:ONELEVEL**
ldap.enable	Turns LDAP lookup off or on. An entry must also be made in **/etc/nsswitch.conf** file to use LDAP for this purpose.	**on**
ldap.port	The port to use for LDAP queries. This defaults to 389, LDAP's well-known port assignment.	**389**
ldap.servers	List of servers to use for LDAP queries. Use quotation marks to indicate servers whose names have embedded spaces or commas. You can specify multiple servers for redundancy.	**ldap1.netapp.com** **ldap1.netapp.com ldap2.netapp.com**
ldap.servers.preferred	List of preferred LDAP servers. Use this list to indicate servers	**Ldap1.netapp.com**

	that are on faster links if any of the servers listed in ldap.servers is on a WAN link or is for some other reason considered slower or less reliable.	

To verify that the above configuration works correctly, you can use the **getXXbyYY** command (available in advanced mode) in Data ONTAP to retrieve user data from the LDAP server.

To look up by the login name:

```
fas3020*> getXXbyYY getpwbyname_r james
pw_name = james
pw_passwd = {MD5}LHQhgFi1aI5yokqEAZ56cQ==
pw_uid = 6000, pw_gid = 300
pw_gecos =
pw_dir = /home/james
pw_shell = /bin/sh
```

Or to look up by the **uid**:

```
fas3020*> getXXbyYY getpwbyuid_r 6000
pw_name = james
pw_passwd = {MD5}LHQhgFi1aI5yokqEAZ56cQ==
pw_uid = 6000, pw_gid = 300
pw_gecos =
pw_dir = /home/james
pw_shell = /bin/sh
```

Note that the password is hashed to prevent from being human readable.

Group Lookup

Suppose you have another entry in LDIF that defines a group named ***webteam*** and its members are ***james*** and ***emily***.

```
dn: cn=webteam,ou=Groups,dc=netapp,dc=com
objectClass: posixGroup
objectClass: top
cn: webteam
gidNumber: 300
memberuid: uid=james,ou=Groups,dc=netapp,dc=com
memberUid: uid=emily,ou=Groups,dc=netapp,dc=com
```

Again, in Data ONTAP, you can use the **getXXbyYY** command to verify the group lookup.
To look up a user's gid:

```
Fas3020*> getXXbyYY getgrlist james
pw_name = james
Groups: 300
```

Or to look up group name by gid:

```
fas3020*> getXXbyYY getgrbygid 300
name = webteam
gid = 300
```

Or to look up gid by group name:

```
fas3020*> getXXbyYY getgrbyname webteam
name = webteam
gid = 300
```

Netgroup Lookup

Netgroups are mostly used for host lookup in NFS mount. In the NFS export table (**/etc/exports**), you may configure the exports based on netgroups in certain situations, for example, the number of hosts becomes too large to maintain effectively. In this scenario, you can take advantage of netgroup to shorten the export table by letting LDAP store the netgroup and its members (host, user or domain). NetApp storage system is compliant with RFC 2307 and can look up netgroup information in the LDAP directory store. Suppose you have server hosts in netgroup **server-hosts** and web hosts in netgroup **web-hosts**, and both **server-hosts** and **web-hosts** are in netgroup **all-hosts**. The structure is defined in the LDAP as follows:

```
dn: cn=server-hosts,ou=Netgroup,dc=netapp,dc=com
objectClass: top
objectClass: nisNetgroup
cn: server-hosts
nisNetgroupTriple: (server1,,)
nisNetgroupTriple: (server2,,)
nisNetgroupTriple: (server3,,)
nisNetgroupTriple: (server4,,)
nisNetgroupTriple: (server5,,)
nisNetgroupTriple: (server6,,)

dn: cn=web-hosts,ou=Netgroup,dc=netapp,dc=com
objectClass: top
objectClass: nisNetgroup
cn: web-hosts
nisNetgroupTriple: (www1,,)
nisNetgroupTriple: (www2,,)
nisNetgroupTriple: (www3,,)
nisNetgroupTriple: (www4,,)

dn: cn=all-hosts,ou=Netgroup,dc=netapp,dc=com
objectClass: top
objectClass: nisNetgroup
cn: all-hosts
memberNisNetgroup: server-hosts
memberNisNetgroup: web-hosts
```

To look up web host www1's membership in LDAP, you can do the following:

```
fas3020*> getXXbyYY netgrp web-hosts www1
client www1 is in netgroup web-hosts
fas3020*> getXXbyYY netgrp all-hosts www1
client www1 is in netgroup all-hosts
fas3020*> getXXbyYY netgrp server-hosts www1
client www1 is not in netgroup server-hosts
```

The results confirm **www1**'s membership in netgroup **all-hosts** and **web-hosts** but not **server-hosts.**

In the example, if you have an export table that looks like this originally:
/vol/volx -rw=server1:server2:server3:server4:server5:server6,ro=www1:www2:www3:www4

it is much shorter than original one if netgroup was used:
/vol/volx -rw=@server-hosts,ro=@web-hosts

Note that the **"@"** sign is used here to distinguish netgroup from host.

4.2 NAME SERVICE SWITCH (NSS) MAPPING

Although Data ONTAP is RFC 2307 compliant, the LDAP server that NetApp storage system talks to may have its own set of custom schemas compatible with RFC 2307. It is also possible that the LDAP server is a Windows domain controller running Microsoft Services for UNIX (MSSFU). In such circumstances, Data ONTAP provides a set of LDAP related options allowing Name Service Switch (NSS) mapping to overcome this problem. These options are shown below along with their default values.

```
ldap.nssmap.attribute.gecos  gecos
ldap.nssmap.attribute.gidNumber gidNumber
ldap.nssmap.attribute.groupname cn
ldap.nssmap.attribute.homeDirectory homeDirectory
ldap.nssmap.attribute.loginShell loginShell
ldap.nssmap.attribute.memberNisNetgroup memberNisNetgroup
ldap.nssmap.attribute.memberUid memberUid
ldap.nssmap.attribute.netgroupname cn
ldap.nssmap.attribute.nisNetgroupTriple nisNetgroupTriple
ldap.nssmap.attribute.uid    uid
ldap.nssmap.attribute.uidNumber uidNumber
ldap.nssmap.attribute.userPassword userPassword
ldap.nssmap.objectClass.nisNetgroup nisNetgroup
ldap.nssmap.objectClass.posixAccount posixAccount
ldap.nssmap.objectClass.posixGroup posixGroup
```

Suppose that on the LDAP server, the name of user identification attribute is "**userid**" instead. You would need to set option **ldap.nssmap.attribute.uid** to **userid** in order to do proper nss mapping.

```
fas3020*> options ldap.nssmap.attribute.uid userid
```

For those who are running Windows Active Directory services with Services for UNIX to emulate an UNIX LDAP server, the nss mapping values need to be specified as follows in order for the LDAP lookups to be successful:

```
ldap.nssmap.attribute.gecos  name
ldap.nssmap.attribute.gidNumber msSFU30GidNumber
ldap.nssmap.attribute.groupname cn
ldap.nssmap.attribute.homeDirectory msSFU30HomeDirectory
ldap.nssmap.attribute.loginShell msSFU30LoginShell
ldap.nssmap.attribute.memberNisNetgroup msSFU30MemberNisNetgroup
ldap.nssmap.attribute.memberUid msSFU30MemberUid
ldap.nssmap.attribute.netgroupname name
ldap.nssmap.attribute.nisNetgroupTriple msSFU30MemberOfNisNetgroup
ldap.nssmap.attribute.uid    sAMAccountName
ldap.nssmap.attribute.uidNumber msSFU30UidNumber
ldap.nssmap.attribute.userPassword msSFU30Password
ldap.nssmap.objectClass.nisNetgroup msSFU30NisNetgroup
ldap.nssmap.objectClass.posixAccount User
ldap.nssmap.objectClass.posixGroup Group
```

Please refer to the Data ONTAP manual page reference documentation available on the NOW™ site for more information on the above options.

4.3 SASL AND SSL

By default, Data ONTAP uses Simple Authentication and Sockets Layer (SASL) when initiating LDAP communication to prevent LDAP data from being transmitted over the network in clear text. If SASL support is not available on the LDAP server, Data ONTAP then falls back to simple authentication (or simple bind), which means password is transmitted over the wire as plain text. Data ONTAP version 7.0.1 and later supports LDAP over SASL using the digest MD5 mechanism.

In addition to SASL support, Data ONTAP version 7.1 and later supports LDAP encryption over Secure Sockets Layer (SSL) to ensure that the transmission of data is secure. Generally speaking, SSL requires more CPU cycles to encrypt every piece of information transmitted across the network including password and data. SASL, on the

other hand, can be configured to only encrypt the password during the authentication phase and leave the data transmission unencrypted if the data is meant to be public (for example, username, title and phone number, etc).

Enabling SASL

First you need to add an administrative account in the LDAP server's database for SASL authentication. For example:

```
dn: uid=ntap,ou=People,dc=netapp,dc=com
objectclass: posixAccount
objectclass: organizationalPerson
objectclass: OpenLDAPperson
cn: SASL account
userpassword: secret
uid: ntap
```

In Data ONTAP, you need to set option **ldap.name** to the **uid** attribute (user's login name) of this account. The **ldap.passwd** option needs to be set to the value of the **userpassword** attribute (***secret***) of this account.

```
fas3020*> options ldap.name ntap
fas3020*> options ldap.passwd secret
```

The following table has more detailed description for the above options.

Option	Description	Example
ldap.name	The username to use for the administrative queries necessary to look up UIDs and GIDs given a username. Best practice is to make this a user with read-only access to the database.	**ntap**
ldap.passwd	The password to use for the administrative user. This will always display as six '*****'s when listing the options.	**secret**

At this point you can use the **getXXbyYY** command described in section 4.1 above along with the **pktt** command to capture the packets for LDAP connection to verify that the communication is indeed encrypted.

Digital Certificate and Certificate Authority (CA)

Before you can use SSL on NetApp storage system, you need to install a self-signed certificate in Data ONTAP. A self-signed certificate is also called a root certificate. A root certificate could either be signed by a commercial certificate authority (CA) such as RSA, Verisign, etc. or the CA of your company, usually the IT organization. A CA manages digital certificate application, including issuance and revocation of the certificate. Figure 3 below is a root certificate of RSA security. The unencrypted part of the certificate can be retrieved from the encrypted portion of the certificate between the lines "**-----BEGIN CERTIFICATE-----**" and "**-----END CERTIFICATE-----**" using the X.509 digital certificate standard. The digital certificate contains information about the issuer, expiration date, public key and so on.

After you decide the CA you want to use, you need to configure the LDAP server so that it has the knowledge of the root certificate of the CA. You also need to generate a certificate signing request using the LDAP server's fully qualified domain name (FQDN) as the common name and send it over to your CA to sign. Lastly, you will need to install the signed certificate on the LDAP server.

```
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            02:ad:66:7e:4e:45:fe:5e:57:6f:3c:98:19:5e:dd:c0
        Signature Algorithm: md2WithRSAEncryption
        Issuer: C=US, O=RSA Data Security, Inc., OU=Secure Server Certification Authority
        Validity
            Not Before: Nov  9 00:00:00 1994 GMT
            Not After : Jan  7 23:59:59 2010 GMT
        Subject: C=US, O=RSA Data Security, Inc., OU=Secure Server Certification Authority
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (1000 bit)
                Modulus (1000 bit):
                    00:92:ce:7a:c1:ae:83:3e:5a:aa:89:83:57:ac:25:
                    01:76:0c:ad:ae:8e:2c:37:ce:eb:35:78:64:54:03:
                    e5:84:40:51:c9:bf:8f:08:e2:8a:82:08:d2:16:86:
                    37:55:e9:b1:21:02:ad:76:68:81:9a:05:a2:4b:c9:
                    4b:25:66:22:56:6c:88:07:8f:f7:81:59:6d:84:07:
                    65:70:13:71:76:3e:9b:77:4c:e3:50:89:56:98:48:
                    b9:1d:a7:29:1a:13:2e:4a:11:59:9c:1e:15:d5:49:
                    54:2c:73:3a:69:82:b1:97:39:9c:6d:70:67:48:e5:
                    dd:2d:d6:c8:1e:7b
                Exponent: 65537 (0x10001)
    Signature Algorithm: md2WithRSAEncryption
        65:dd:7e:e1:b2:ec:b0:e2:3a:e0:ec:71:46:9a:19:11:b8:d3:
        c7:a0:b4:03:40:26:02:3e:09:9c:e1:12:b3:d1:5a:f6:37:a5:
        b7:61:03:b6:5b:16:69:3b:c6:44:08:0c:88:53:0c:6b:97:49:
        c7:3e:35:dc:6c:b9:bb:aa:df:5c:bb:3a:2f:93:60:b6:a9:4b:
        4d:f2:20:f7:cd:5f:7f:64:7b:8e:dc:00:5c:d7:fa:77:ca:39:
        16:59:6f:0e:ea:d3:b5:83:7f:4d:4d:42:56:76:b4:c9:5f:04:
        f8:38:f8:eb:d2:5f:75:5f:cd:7b:fc:e5:8e:80:7c:fc:50
MD5 Fingerprint=74:7B:82:03:43:F0:00:9E:6B:B3:EC:47:BF:85:A5:93
-----BEGIN CERTIFICATE-----
MIICNDCCAaECEAKtZn5ORf5eV288mBle3cAwDQYJKoZIhvcNAQECBQAwXzELMAkG
A1UEBhMCVVMxIDAeBgNVBAoTF1JTQSBEYXRhIFNlY3VyaXR5LCBJbmMuMS4wLAYD
VQQLEyVTZWN1cmUgU2VydmVyIENlcnRpZmljYXRpb24gQXV0aG9yaXR5MB4XDTk0
MTEwOTAwMDAwMFoXDTEwMDEwNzIzNTk1OVowXzELMAkGA1UEBhMCVVMxIDAeBgNV
BAoTF1JTQSBEYXRhIFNlY3VyaXR5LCBJbmMuMS4wLAYDVQQLEyVTZWN1cmUgU2Vy
dmVyIENlcnRpZmljYXRpb24gQXV0aG9yaXR5MIGbMA0GCSqGSIb3DQEBAQUAA4GJ
ADCBhQJ+AJLOesGugz5aqomDV6wIAXYMra6OLDfO6zV4ZFQD5YRAUcm/jwjiioII
0haGN1XpsSECrXZogZoFokvJSyVmIlZsiAeP94FZbYQHZXATcXY+m3dM41CJVphI
uR2nKRoTLkoRWZweFdVJVCxzOmmCsZc5nG1wZ0j3S3WyB57AgMBAAEwDQYJKoZI
hvcNAQECBQADfgBl3X7hsuyw4jrg7HFGmhkRuNPHoLQDQCYCPgmc4RKz0Vr2N6W3
YQO2WxZpO8ZECAyIUwxrl0nHPjXcbLm7qt9cuzovk2C2qUtN8iD3zV9/ZHuO3ABc
1/p3yjkWWW8O6tO1g39NTUJWdrTJXwT4OPjr0l91X817/OWOgHz8UA==
-----END CERTIFICATE-----
```

Figure 3) Example of CA Root Certificate

Installing the Root Certificate in Data ONTAP

The CA establishes a trust relationship with the LDAP server by signing the LDAP server's certificate request. This is a two-way trust relationship in that the CA "trusts" the LDAP server and the LDAP server is "trusted" by the CA. NetApp storage system, as an LDAP client, "trusts" the CA by installing the CA's root certificate in Data ONTAP. This chain of trust relationship is called "chaining" in cryptography. In this scenario, NetApp storage system has chained from its trusted copy of the CA's public key (embedded in the certificate) to a trusted copy of the LDAP server's public key. As a result, a secured communication channel can be established between NetApp storage system and the LDAP server. An analogy would be using a web browser to access a secure banking site, the web server at the banking site has a signed certificate issued by a commercial CA. And the browser has a built-in root certificate from the same CA that it trusts. A secure communication based upon HTTPS can therefore be accomplished.

Suppose that you have the root certificate named **`cacert.pem`**. You would need to use the **`keymgr`** command in Data ONTAP to install it. Copy the certificate to the **`/etc`** directory in root volume on NetApp storage system, then issue the following command:

```
fas3020*> keymgr install root /etc/cacert.pem
```

Verify that the certificate is installed correctly:

```
fas3020*> keymgr list root
```

```
Existing certificate file(s):
Name          Common Name          Size          Expiration Date
cacert.pem    netapp.com           1139          Dec 29 01:55:32 2006 GMT
```

Now we are ready to enable the encryption method of our choice using either SSL or SASL.

Enabling SSL

Now that the certificates are in place, you need to enable LDAP over SSL by switching on the **ldap.ssl.enable** option. The value of **ldap.port** will change automatically from 389 to 636, which is the standard TCP port for LDAP over SSL (ldaps).

```
fas3020*> options ldap.ssl.enable on
fas3020*> options ldap.port
ldap.port                    636
```

Again, at this point you can use the **getXXbyYY** command described in section 4.1 above along with the **pktt** command to capture the packets for LDAP connection to verify that the communication is indeed encrypted.

Note that you can run SASL and SSL independently or simultaneously. SASL is not required to run SSL and vice versa. Also note that CA is not required for SASL deployment. By contrast, CA is necessary to successfully deploy SSL.

4.4 USER MAPPING

In a heterogeneous environment, a physical user may have a user id for UNIX account and another for Windows account as discussed in section 3 above. Traditionally, you can use the **/etc/usermap.cfg** file to define the mapping between the two accounts (see TR 3014 for more information). However, since **usermap.cfg** is local, as the number of the storage systems grows, management will become a challenge. To address this issue, Data ONTAP performs symmetric and asymmetric lookups of the centralized user mapping data within LDAP to provide the same function. User mapping support using LDAP is available in Data ONTAP 6.5.1 and later.

Symmetric and Asymmetric Lookup

The purpose of this paper is to describe using UNIX LDAP server only to look up user mapping. It therefore falls into the category of "symmetric lookup". Symmetric lookup uses just a single directory server to map Windows accounts to UNIX users, or to map UNIX users to Windows accounts. In contrast to symmetric lookup, asymmetric lookup involves both UNIX LDAP server (for lookups from UNIX to Windows) and Windows Active Directory Server (for lookups from Windows to UNIX). Figure 4 below illustrates how user mapping works in Data ONTAP for symmetric lookup using UNIX clients. Before you can use user mapping, make sure the following conditions are met. Note that you are required to extend the directory schema to describe the user mapping:

- The UNIX user information is expected to be found in attributes described by RFC 2307.
- The Windows account name is found in a user object attribute that is defined by the option **ldap.usermap.attribute.windowsaccount**[2]. This attribute is defined by a custom schema extension.

[2] In contrast, option **ldap.usermap.attribute.unixaccount** is used by Windows clients in AD domain to perform symmetric lookups.

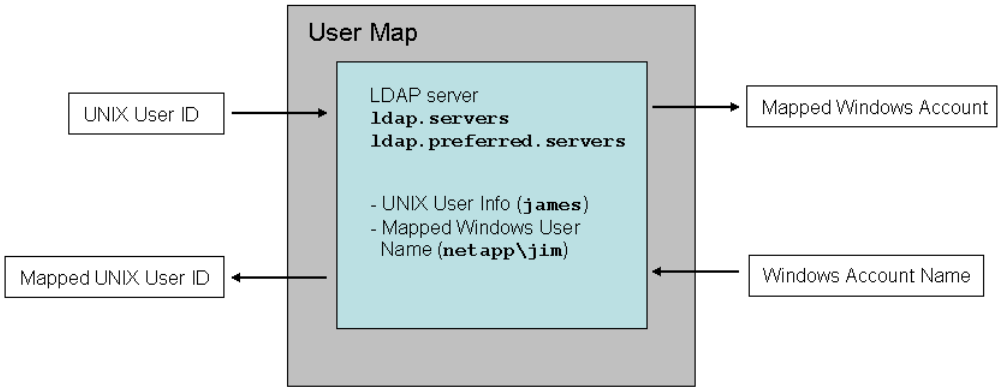

Figure 4) LDAP User Mapping

Implementation

- Enable user mapping lookup:

  ```
  fas3020*> options ldap.usermap.enable on
  ```

- Symmetric lookup is specified by the option **ldap.usermap.symmetriclookup** being set to "**yes**".

  ```
  fas3020*> options ldap.usermap.symmetriclookup yes
  ```

- Extend your schema to include an attribute for the user mapping, for example, **winAcctName:**

  ```
  dn: cn=James Dole,ou=People,dc=netapp,dc=com
  objectClass: posixAccount
  objectClass: OpenLDAPperson
  objectClass: shadowAccount
  cn: James Dole
  givenName: James
  title: Product and Partner Engineer
  uid: james
  winAcctName: netapp\jim
  mail: jdole@netapp.com
  uidNumber: 6000
  gidNumber: 300
  ```

- Set option **ldap.usermap.attribute.windowsaccount** to **winAcctName:**

  fas3020*> options ldap.usermap.attribute.windowsaccount winAcctName

Verification

The **wcc** command (WAFL® credential cache) enables you to check if the user mapping is functioning correctly. Assuming that the NetApp storage system has joined Windows AD domain ***netapp***, this is done through the **cifs setup** command and by setting option **ldap.ADdomain** to domain ***netapp***.

For user lookup from UNIX to Windows:

```
fas3020*> wcc –u james
(NT - UNIX) account name(s):  (NETAPP\jim - james)
        ***************
        UNIX uid = 6000

        NT membership
                NETAPP\jim
                BUILTIN\Users
        User is also a member of Everyone, Network Users,
        Authenticated Users
        ***************
```

For user lookup from Windows to UNIX:

```
fas3020*> wcc –s jim
(NT - UNIX) account name(s):  (NETAPP\jim - james)
        ***************
        UNIX uid = 6000

        NT membership
                NETAPP\jim
                BUILTIN\Users
        User is also a member of Everyone, Network Users,
        Authenticated Users
        ***************
```

The outputs show that the NetApp storage system is able to find proper user mapping that we defined on the LDAP server.

Trouble Shooting

If for some reason, the user mapping does not work correctly after following the above steps, you can turn on option **cifs.trace_login** so Data ONTAP can display more debugging information for trouble shooting.

```
fas3020*> wcc -u james

Wed Mar 15 12:59:39 PST [fas3020:
auth.trace.authenticateUser.loginTraceMsg:info]: AUTH: LSA lookup: Located
account "netapp\jim" in domain "NETAPP"..
```

4.5 MAKING YOUR LDAP DEPLOYMENT MORE SECURE

Here are a couple of tips that you can use to make your LDAP deployment more secure:

Running LDAP on non-standard port

By default LDAP service listens on port 389 for unencrypted connection and port 636 for encrypted connection. You can select an unprivileged port number (>1024) to run LDAP service and specify this value to the **ldap.port** option. By communicating through a non-standard port number, you could avoid exposing your LDAP communication to unsolicited port scanning software in the network.

Always use SASL or SSL to secure your LDAP communications

Although encryption in LDAP is optional, it is highly recommended to enable either SASL or SSL to secure communications between NetApp storage system and the LDAP server. Please note that SSL requires the installation of Certificate Authority (CA) while SASL does not.

4.6 OTHER LDAP OPTIONS

The following LDAP related options are not discussed in this paper but in general, they allow more flexibility in a complex LDAP infrastructure:

Option	Description	Example
ldap.base.group	The base distinguished name to use for group lookups, this option will override **ldap.base** option. The format of the base string is: "(filter1):scope1;(filter2):scope2;". The scope can be one of those three choices: **BASE, ONELEVEL** or **SUBTREE**. The default scope is **SUBTREE** if it is not specified.	**dc=netapp,dc=com** **dc=netapp,dc=com:ONELEVEL**
ldap.base.netgroup	The base distinguished name to use for netgroup lookups, this option will override **ldap.base** option. The format of the base string is: "(filter1):scope1;(filter2):scope2;". The scope can be one of those three choices: **BASE, ONELEVEL** or **SUBTREE**. The default scope is **SUBTREE** if it is not specified.	**ou=Groups,dc=netapp,dc=com**
ldap.base.passwd	The base distinguished name to use for user password lookups, this option will override the **ldap.base** option. The format of the base string is: "(filter1):scope1;(filter2):scope2;". The scope can be one of those three choices: **BASE, ONELEVEL** or **SUBTREE**. The default scope is **SUBTREE** if it is not specified.	**ou=People,dc=netapp,dc=com**
ldap.usermap.base	The base distinguished name to use	**ou=Engineering,dc=netapp,dc=com**

	for ldap usermapping. The format of the base string is: "(filter1):scope1;(filter2):scope2;". The scope can be one of those three choices: **BASE, ONELEVEL** or **SUBTREE**. The default scope is **SUBTREE** if it is not specified.	

For more information about these options and other options that we have discussed in this paper, you can find useful descriptions in "*Using LDAP services*" section of the File Access and Protocols Management Guide under Chapter "*File Sharing Between NFS and CIFS*" on the NOW site.

5. CONCLUSION

LDAP plays a critical role as a network directory service and has the potential to replace NIS completely. LDAP is more secure and contains consolidated network information usually served up by individual services such as DNS and NIS. A successful integration of LDAP services and NetApp storage system ensures not only higher quality of secure directory service but also better access control for authorized users while providing network file services.

APPENDIX

A. GLOSSARY

CA – (Certificate Authority)
An issuer of Security Certificates used in SSL connections.

Digital Certificate
An attachment to an electronic message used for security purposes. The most common use of a digital certificate is to verify that a user sending a message is who he or she claims to be, and to provide the receiver with the means to encode a reply.

Directory Service
A directory service is a software application—or a set of applications—that stores and organizes information about a computer network's users and network shares, and that allows network administrators to manage users' access to the shares. Additionally, directory services act as an abstraction layer between users and shared resources.

DNS – (Domain Name Service)
An Internet service that translates domain names into IP addresses.

INODE
Data structures that contain information about files in the file systems. Each file has an inode and is identified by an inode number (i-number) in the file system where it resides. inodes provide important information on files such as user and group ownership, access mode (read, write, execute permissions) and type.

LDIF – (LDAP Data Interchange Format)
A data interchange format for exporting data from and importing data to LDAP servers. It conveys directory information or a description of a set of changes made to directory entries. The data are represented in plain text form.

NIS – (Network Information Service)
Sun Microsystems' "Yellow Pages" (YP) client-server directory service protocol for distributing system configuration data such as user and host names between computers on a computer network.

NTFS – (New Technology File System)
The standard file system of Windows NT® and its descendants Windows 2000, Windows XP and Windows Server 2003. NTFS has strong security enhancements over early Microsoft FAT (File Allocation Table) file system.

RFC – (Request for Comments)
A series of notes about the Internet, started in 1969. An Internet Document can be submitted to the Internet Engineering Task Force (IETF) by anyone, but the IETF decides if the document becomes an RFC. Eventually, if it gains enough interest, it may evolve into an Internet standard.

SASL – (Simple Authentication and Security Layer)
Originating with RFC 2222, written by John Myers while at Netscape Communications, SASL is a method for adding authentication support to connection-based protocols. SASL takes effect when a protocol initiates a command for identifying and authenticating a user to a server.

Schema
The structure of a database system, described in a formal language supported by the database management system (DBMS). In a relational database, the schema defines the tables, the fields in each table, and the relationships between fields and tables. RFC 2252 provides the necessary knowledge for understanding the majority of OpenLDAP's schema files.

SSL – (Secure Sockets Layer)
A protocol designed by Netscape Communications to enable encrypted, authenticated communications across the Internet.

WAFL – (Write Anywhere File Layout)

The microkernel of NetApp storage appliance's operating system – Data ONTAP. WAFL was specifically designed to work in a network file server appliance. WAFL and RAID were designed together to avoid the performance problems that most file systems cause with RAID.

X.500

Originated in RFC 1943. An ISO standard that defines how global directories should be structured. X.500 directories are hierarchical with different levels for each category of information.

X.509

Originated in RFC 2459. A widely used standard for defining digital certificates.

B. REFERENCES

- TR 3387 – Security in NFS Storage Networks
- TR 3445 – Data ONTAP: Best Practices For Security Configuration
- TR-3014 -- Multiprotocol Data Access: NFS, CIFS, and HTTP
- Data ONTAP File Access and Protocols Management Guide
- Data ONTAP Commands: Manual Page Reference
- OpenLDAP Administrator's Guide