KACE Appliance LDAP Reference Guide

V1.4

Brandon Whitman

The purpose of this guide is to help you with both common and advanced LDAP issues related to the KACE appliances. This guide will give you some background on LDAP as well as teach you to create search filters to assist you with configuring LDAP authentication, labels, and user imports. You will also learn the capabilities and limitations of LDAP on the KACE appliances.

The KACE Appliances allow you to use your existing LDAP credentials for login. The appliances support multiple LDAP systems like Microsoft Active Directory, OpenLDAP, and Novell eDirectory. It will work with any LDAP system that supports LDAP 3.0.

We will cover the LDAP capabilities of the KACE appliances in 4 sections.

- LDAP Authentication
- LDAP User Imports
- LDAP Labels
- LDAP Search Filters

LDAP Authentication will allow your users to login to the KACE appliance using their current LDAP credentials.

LDAP user imports will allow you to import a mass number of users into the K1000 series appliance, and pull in multiple bits of information from the users LDAP attributes.

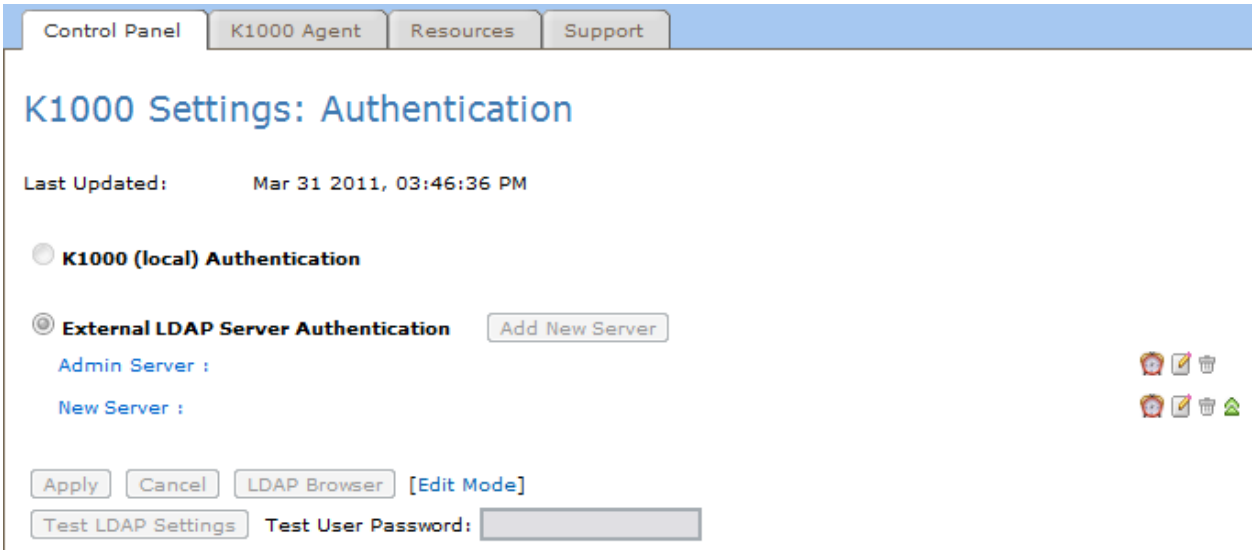
LDAP labels will allow you to group your users based on common LDAP attributes. Depending on your LDAP organization you may be able to group by such things as department, location, domain, etc.

Throughout this guide you will see the following Symbols.

-  This symbol will indicate a Best Practice for LDAP on KACE.
-  This symbol will indicate a Note of Interest for LDAP on KACE.

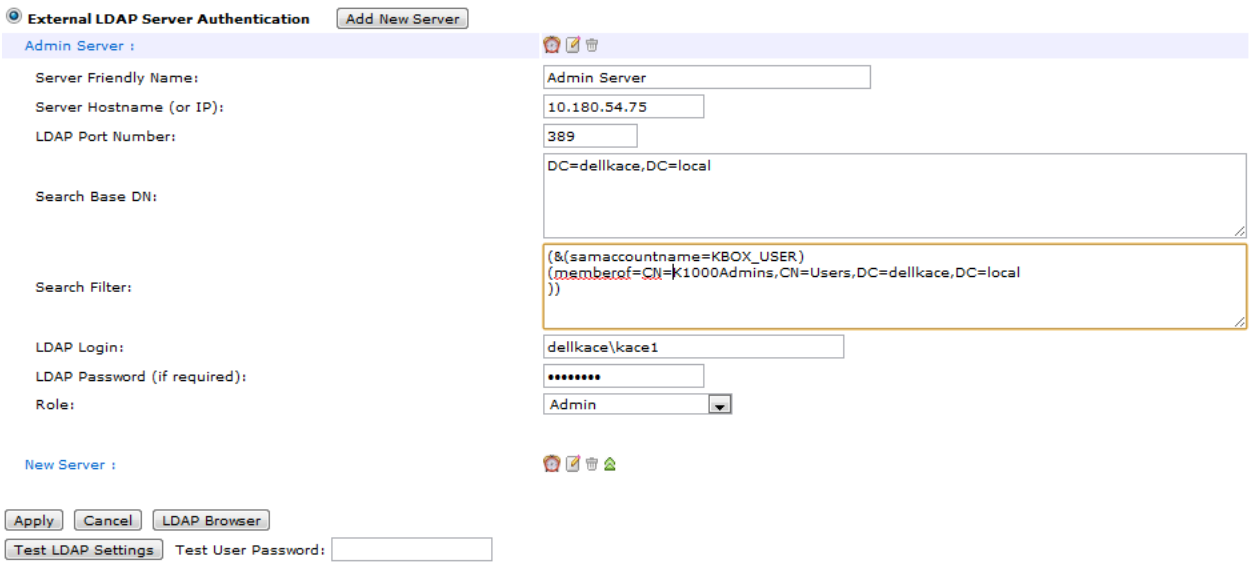
LDAP Authentication.

LDAP authentication can be configured by navigating to Settings > User Authentication. Here you will see a list of servers that are setup to pass through authentication credentials to LDAP with a specific search filter to determine which type of user they are. These can be as simple as an Admin and a User server as you will see in the diagram below, or can be made much more granular depending on your needs.



To begin, click on the [Edit Mode] link and then select the External LDAP Server Authentication radio button. Now click on the small page icon to edit an existing server or click Add New Server to create a new one. We will use the existing ones in this example.

If you do not select the External LDAP Server Authentication radio button before you start creating your Authentication Servers then the changes will not be saved when you click Apply.



The above diagram shows a common LDAP authentication setup. It is set up to authenticate members of the K1000Admins group. This will allow users that are in the K1000Admins Group to login to the KACE appliance as a user with the Admin role.

A bit of vocabulary:

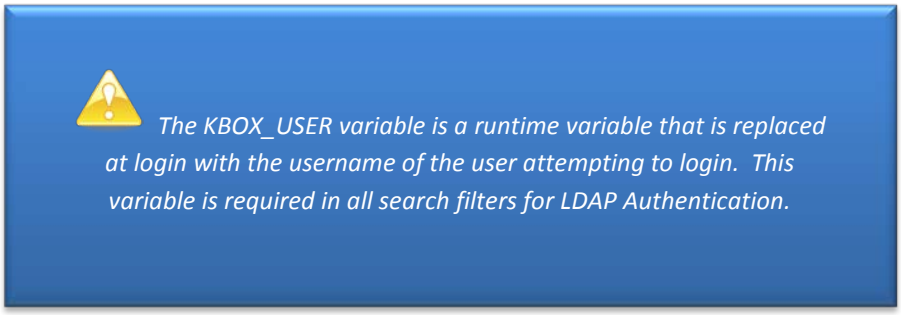
- *Search Base DN: The scope that the Search Filter will execute in.*
- *Search Filter: The criteria that must be met by a user in order to login to the KACE appliance.*
- *LDAP Login and Password: This must be an account that has, at minimum, full read access to the LDAP organization.*
- *Role: This designates the permissions the authenticated user will have upon login.*

Enter the required information such as the Server Friendly Name, the Server Hostname (or IP) and the LDAP port number (generally 389). To set your Search Base DN and the Search Filter you can either write it freehand or you can use the built-in LDAP Browser to assist you. Creating search filters with the LDAP Browser is covered on Page 12.

You can create as many authentication servers as necessary to have role specific logins for your users. This will allow users to automatically be assigned the proper role on login, even in between scheduled LDAP user imports.

Once your search filter is created, you will want to test that the server returns the correct number of results and that it is able to perform a successful authentication. Follow the instructions below.

- *In the Search Filter, Change the KBOX_USER variable to a * and then click Test LDAP Settings.*
    - *You should see the total number of records found shown in the test message below.*
- *To test for successful authentication, change the KBOX_USER variable to a specific user. For Example: **(samaccountname=Whitman)***
- *To test these settings:*
    - *Enter the password for the user you entered in the search filter, then press the Test LDAP Settings button.*
    - *You should see a successful test message in the box below.*
- *Make sure to change the username back to KBOX_USER before saving the settings.*



If you create your own LDAP authentication servers, make sure to delete any of the example servers that were pre-existing on the KACE appliance. Incorrect LDAP authentication servers can cause long delays in login time.

Now that users can successfully login, we will move onto LDAP user imports.

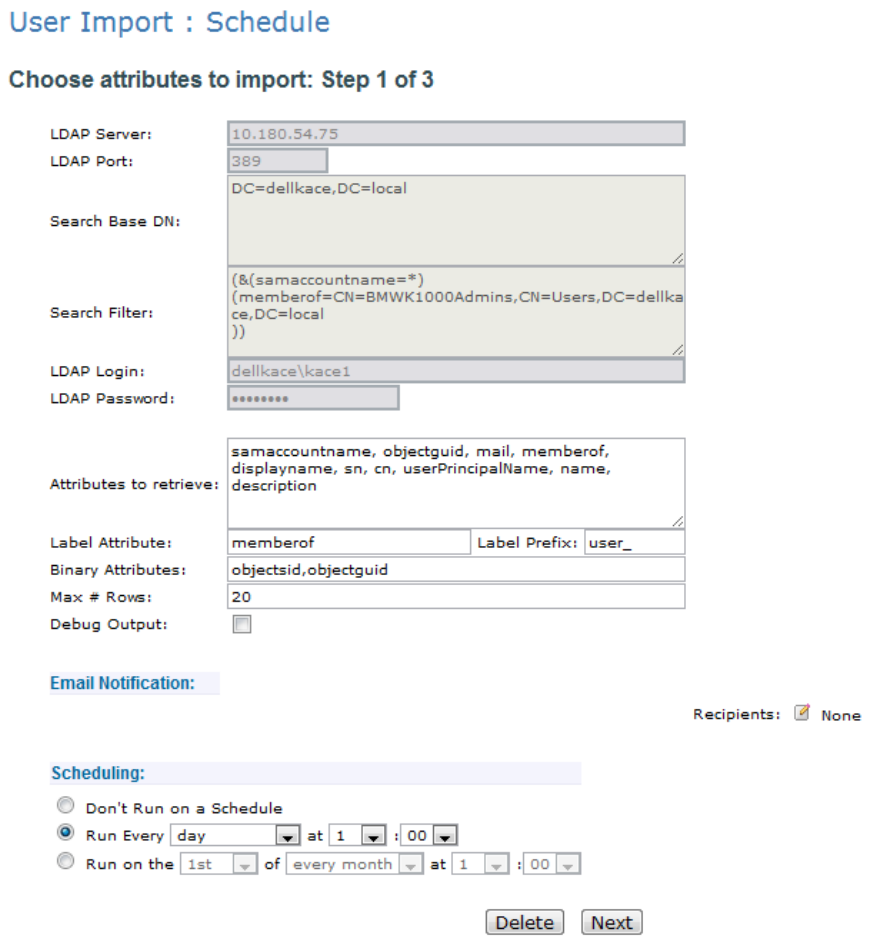
LDAP User Imports

Importing users is done on the K1000 for the purpose of having a user base for Administration and Helpdesk functions. Now that we can successfully authenticate users into the KBOX we need to import those users into the K1000 series appliance.

Navigate to Settings > User Authentication and click [Edit Mode] at the bottom of the page. Click on the small clock icon for the server you wish to schedule an import for. User Imports are a 3 step process.

Step 1: Choose attributes to import.

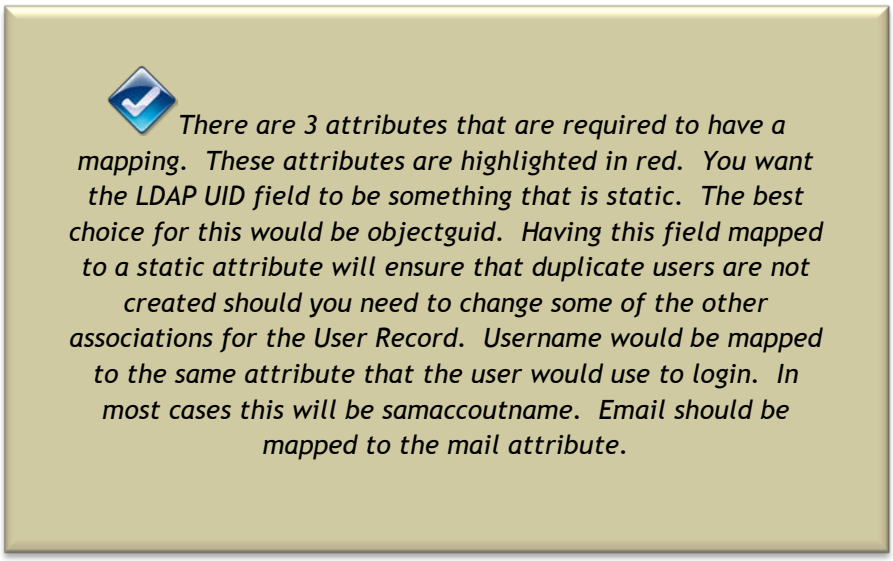
You will see the following screen and the import schedule will be prepopulated with several common attributes. Attributes will vary depending on the LDAP system you are using. We will focus on the most common Microsoft Active Directory attributes here.



You can add desired attributes to the list of Attributes to retrieve. Make sure each attribute is separated by a comma. A list of Microsoft Active Directory Attributes and their explanations can be found in the Appendix on Page 16.

Step 2: Define Mapping between User Attributes and LDAP attributes.

Here you can map the LDAP attributes that you selected for import to the User Record Fields in the K1000 appliance.



All the other attribute mappings are optional and can be set based on your needs. If you do not see an attribute you selected available for mapping then it was not present in the small amount of initial search results. There are two ways to remedy this. You can go back to Step 1 and increase the value in the Max # Rows field or you can go add a value to the desired field in LDAP for one of the users shown in the search results on Step 2.

Here is an example of the completed attribute mapping page. You can also specify the role you wish users to import as here. Once everything is associated as needed we can move on to review the import list and either save our settings or import now.

Roles will only apply on the FIRST import of a user. This import can happen as part of a scheduled import or by logging into the KBOX for the first time. Changing the role here then importing the users will NOT change the role for existing users.

Search Results (1 of 1)

#	Cn	Sn	Displayname	Memberof	Name	Objectguid
1	brandon whitman	Whitman	Brandon Whitman	BMWK1000Admins, Domain Admins	brandon whitman	d02d81df29863245ab385400ca635f4d

Step 3: Import Data into the K1000

Review your list of users and labels to imported and/or updated on this screen. This will give you information on new users to be imported, existing users (if any) to be updated, and any users that will not be imported due to invalid data. Users lacking any of the attributes in red will show under users with invalid data. This is most common when the user is lacking an email address. Users lacking an email address can still be imported to the KBOX but they will only import on login.

If everything on this page looks good then go ahead and click save or import now. Importing now will also save the settings of the import.

You should now be able to go to the Users tab under the Helpdesk module and see the listing of users that were imported.

KACE LDAP authentication and import Best Practices.

Authentication servers and user imports should always set to authenticate or import in order of hierarchy from highest to lowest. This will prevent users who should be admins from possibly importing as a less privileged role as some users may fit the criteria for multiple authentication servers.

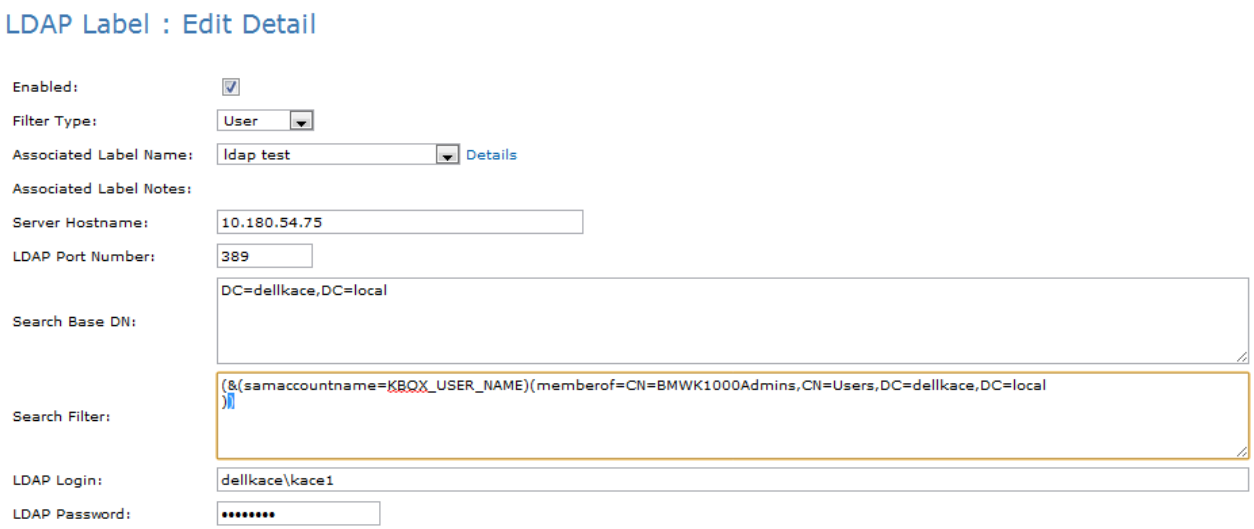
When setting up multiple imports using the SAME LDAP system, you will get an error stating that a scheduled import for this server already exists. To circumvent this error, modify the server hostname/IP. You can schedule up to 3 imports for the same server by using IP, Hostname, and FQDN.

LDAP Labels

LDAP Labels can be used just as other labels in the K1000 to automatically assign users or computers based on LDAP search filter criteria rather than manual assignment or assignment through database query based smart labels. The ease of implementing these labels is directly related to the organization of your LDAP system. If you have users or computers already in specified groups or organizational units then these can be very efficient for assigning users in the Helpdesk or for filtering computer for software or script deployment. We will cover basic LDAP labels in this section. You can create more powerful filters using the techniques covered in the LDAP Search Filters section later in this guide.

LDAP Labels are considered to be a type of smart label. Each smart label must have an associated Parent label. To create the parent label navigate to Home > Labels > Label Management. From the Choose Action drop-down menu choose Add New Label. Give the parent label a name, add notes if desired and click save. Next we will create the associated LDAP Label. To create the LDAP label navigate to Home > Labels > LDAP labels. From the Choose action drop-down menu choose Add New Label. Give the LDAP label a name. Choose the associated parent label that you just created from the Associated Label Name list. From here it will be much like creating an LDAP authentication server. Enter the Hostname or IP, the LDAP Port Number, Search Base DN, and Search filter. You can use the LDAP Browser to enter the Search Base DN and the Search Filter by clicking on the LDAP Browser button near the bottom of the page.

Here is an example of an LDAP Label that will assign our Admins based on the same search criteria that we used to create the Admin authentication server previously.



During the filter processing, the K1000 will replace all "KBOX_" defined variables with their respective runtime values. Currently supported variables for a Filter Type of **Machine** are:

KBOX_COMPUTER_NAME
KBOX_COMPUTER_DESCRIPTION
KBOX_COMPUTER_MAC
KBOX_COMPUTER_IP
KBOX_USERNAME
KBOX_USER_DOMAIN
KBOX_DOMAINUSER
KBOX_CUSTOM_INVENTORY_*

Should the external server require credentials for administrative login (aka non-anonymous login) please supply those credentials. If no LDAP user name is given, an anonymous bind will be attempted. Each LDAP filter may connect to a different LDAP/AD server.

The KBOX_CUSTOM_INVENTORY_* field can be used to check a custom inventory value. The * will be replaced with the Display Name of the custom inventory rule. Only allowed characters are [a-z0-9.-] anything else will be replaced with an '_' character.

Currently supported variables for a Filter Type of **User** are:

KBOX_USER_NAME
KBOX_FULL_NAME
KBOX_EMAIL
KBOX_DOMAIN
KBOX_BUDGET_CODE
KBOX_LOCATION
KBOX_WORK_PHONE
KBOX_HOME_PHONE
KBOX_MOBILE_PHONE
KBOX_PAGER_PHONE
KBOX_CUSTOM_1
KBOX_CUSTOM_2
KBOX_CUSTOM_3
KBOX_CUSTOM_4
KBOX_ROLE_ID
KBOX_LOCALE_BROWSER_ID
KBOX_LDAP_UID

Each LDAP Label must have one of the KBOX variable above as part of the search filter. This must match the associated attribute in the filter or the label will not work. If there is no KBOX variable in the search filter it will apply to all machines.

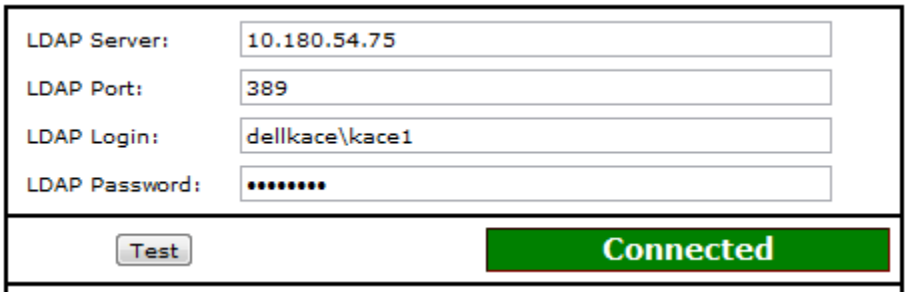
LDAP User Labels will only be applied to users on login. If you have users that do not login to the KBOX then LDAP labels will not be the most efficient way to label your users.

LDAP Machine Labels will apply to computers at check-in. Do not be alarmed if you create a label then check it only to find that there are few or no machines that have been applied the label. It will take a full check in cycle for all the machines to be applied to the label.

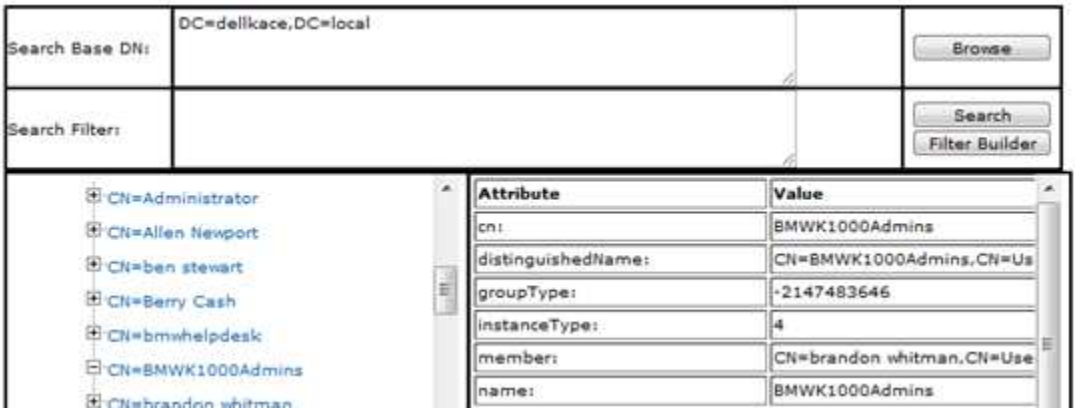
LDAP Search Filters

The ability to write search filters is the backbone of successful usage of LDAP on the KACE appliances. In this section we will cover using the LDAP Browser and Filter Builder to form LDAP search filters along with a little bit of freehand writing that will make this process easier. It is a little convoluted as you will create part of the search filter to return the correct number of search results then combine that filter with the required KBOX variable to complete it and make it work with the appliance.

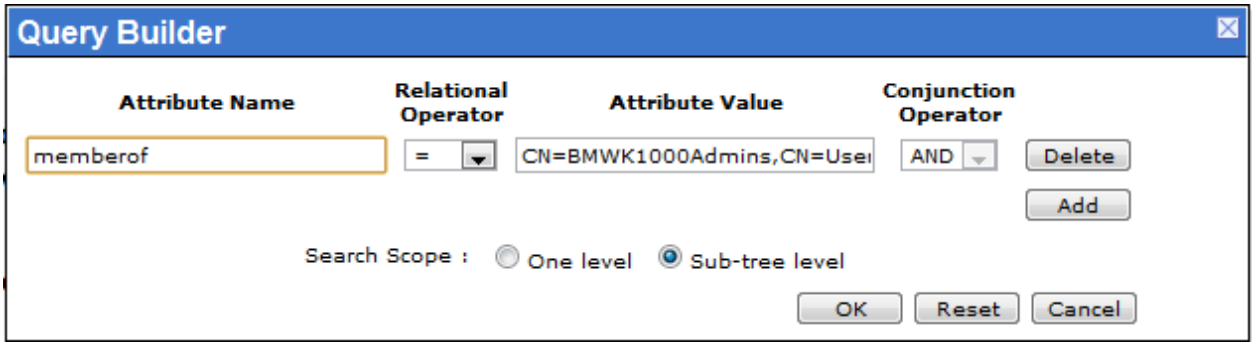
From the setup of LDAP authentication servers, LDAP labels and Step 1 of the User import process you will see a button labeled LDAP Browser. Click that button to access the browser. You will need to enter your LDAP server details here if they are not already present. Click test to test your connection. Once it shows connected as below you may click next.



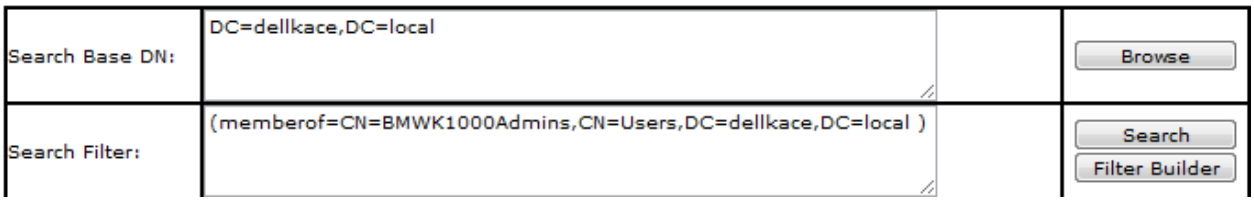
We are now in the LDAP Browser. In this example we will use the same KACE Admins group that we have used previously in this document. We will show how to navigate to a group in order to create a search filter for that group. In the LDAP Browser, remove any entry in the search filter box. Your Search Base DN should already be populated. Click Browse and then navigate to the group you wish to create the filter against. (in this case BMWK1000Admins). Select the group and copy the entry next to the distinguishedName attribute. See the graphic below.



Next we will use the filter builder to create the search filter that will search within the specified group. Enter the attribute "memberof" in the Attribute Name field and copy the Distinguished Name into the Attribute Value. This will make the search filter look for users that are a member of the specified group. It will look like this.



Click OK and now your LDAP browser should look something like this.



Press the search button to display the results. If the number of results shown is correct then we can now complete our search filter with the KBOX_USER variable.

You will add the "(&(samaccountname=KBOX_USER)" at the beginning of the search filter. You will also add a single ")" to the end.

EXAMPLE

Before:

(memberof=CN=BMWK1000Admins,CN=Users,DC=dellkace,DC=local)

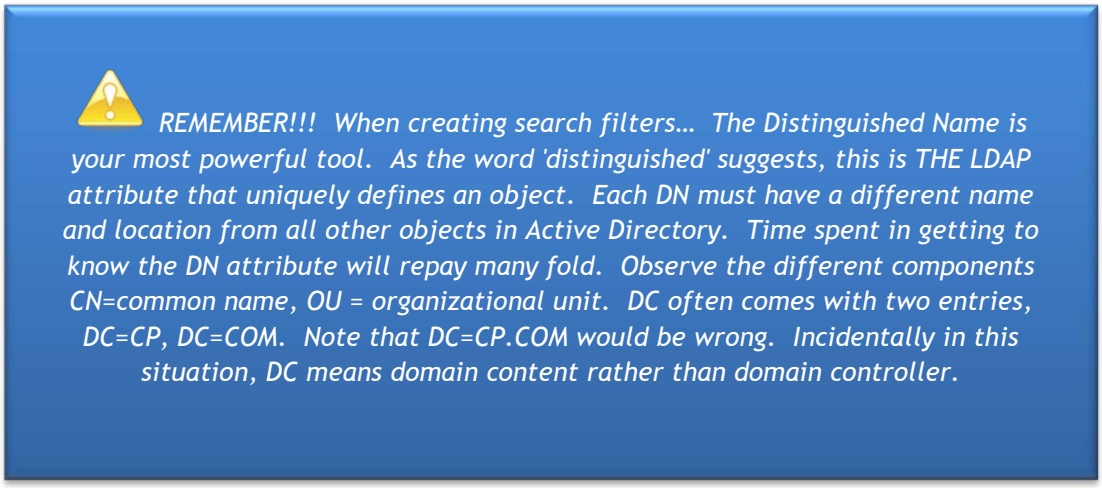
After:

(&(samaccountname=KBOX_USER)(memberof=CN=BMWK1000Admins,CN=Users,DC=dellkace,DC=local))

Click next. Confirm that the settings are correct on this page and click next again. Users that are a part of the group specified in the search filter will now be able to login to the KBOX interface.

Let's Get Crafty

Here are some other example queries. You can use the concepts shown here to create your own more powerful search filters.



One common request is to have an authentication server for all users. This can be easily done using a Base DN that is as wide as possible (in our case this would be DC=dellkace,DC=local) and a search filter containing just (samaccountname=KBOX_USER). This will, in fact, return all users. However, it will also return service accounts, deactivated users, and others that we do not want to allow login to the KBOX. Often the desired list of users is the membership of the Domain Users security group. Members of this group however do not have a memberof attribute for this group. In order to get the membership of the group you must use the primarygroupid attribute, which for the Domain Users group is 513.

(&(samaccountname=*)(&(objectCategory=user)(primarygroupid=513))

If you also want to exclude the few system mailboxes and some service accounts that end up in the Domain Users group, the following search filter will allow login of only users who are actual people (not service accounts) and are active within Active Directory.

Search Base DN: DC=dellkace,DC=local

Search Filter:
(&(samaccountname=KBOX_USER)(&(objectCategory=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2))))

If you want to search for all active users within a specific OU then you can limit the scope of the search using the Search Base DN as shown below.

Search Base DN: OU=Location1,OU=Users,DC=dellkace,DC=local

Search Filter:
(&(samaccountname=KBOX_USER)(&(objectCategory=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2))))

To search for all users in a specific group (useful for admins or other role specific logins) Use a filter similar to the one below. Remember when searching for users in groups that you MUST use the Distinguished Name of the group and to keep your search base as wide as possible as all users may not be in the same path as their group.

Search Base DN: DC=dellkace,DC=local

Search Filter:
(&(samaccountname=KBOX_USER)(memberof=CN=BMWK1000Admins,CN=Users,DC=dellkace,DC=local))

If you want to search for multiple groups then you would want to use the filter builder in the LDAP browser to create your initial search filter then add the KACE variable as we did on Page 13.

Search Base DN: DC=dellkace,DC=local

Search Filter (before KACE variable):
(|(memberof=CN=Group2,CN=Users,DC=dellkace,DC=local)(memberof=CN=Group1,CN=Users,DC=dellkace,DC=local))

Search Filter (after KACE variable): (&(samaccountname=KBOX_USER)
(|(memberof=CN=Group2,CN=Users,DC=dellkace,DC=local)(memberof=CN=Group1,CN=Users,DC=dellkace,DC=local)))

Notice how all we did was add “(&(samaccountname=KBOX_USER)” at the beginning of the search filter and a single “)” to the end. This is the simplest way to ensure that complex search filters will work correctly. Filter the users first and make sure your string is correct then add the KACE variable.

Nested Groups

Nested groups can be a pain point when creating search filters. A search filter will search exactly what you tell it to. It will go no further. If you have a group who's membership is other groups that contain users you can use a string to search through the groups. Here is an example.

(&(samaccountname=KBOX_USER)(memberof:1.2.840.113556.1.4.1941:=CN=nestedgroup,CN=Users,DC=whitman,DC=com))

It is the ":1.2.840.113556.1.4.1941:" after the memberof attribute that walks the chain of ancestry in objects all the way to the root until it finds a match.

Multiple OU's

There is not a way to query multiple specific OU's for authentication. You can target a single OU with the search base but there is not a way to specify multiple ones. You may see information that says you can search an OU using a memberOf attribute, but as an OU does not have "membership" then this will not work.

For Example

"(|(memberOf=OU=sales,DC=kace,DC=com)(memberOf=OU=support,DC=kace,DC=com))(samaccountname=KBOX_USER))"

The above query will result in an error when testing the search string.
You may also not use wildcards in a search filter that utilizes any distinguished names.

For Example

"(|(memberOf=OU=sales,DC=kace,DC=com)(memberOf=*OU=support,DC=kace,DC=com))(samaccountname=KBOX_USER))"

The Wildcard in this search filter will not be processed as it is part of a distinguished name. This search filter will error as well.

Appendix A

Useful LDAP Attributes

LDAP Attribute	*Example*
CN - Common Name	CN=Brandon Whitman. Actually, this LDAP attribute is made up from givenName joined to SN.
description	What you see in Active Directory Users and Computers. Not to be confused with displayName on the Users property sheet.
displayName	displayName=Brandon Whitman. If you script this property, be sure you understand which field you are configuring. DisplayName can be confused with CN or description.
DN - also distinguishedName	DN is simply the most important LDAP attribute. CN=Brandon Whitman,OU=Users,DC=dellkace,DC=local
givenName	Firstname also called Christian name
name	Name=Brandon Whitman. Exactly the same as CN.
objectCategory	Defines the Active Directory Schema category. For example, objectCategory=Person
objectClass	objectClass=User. Also used for Computer, organizationalUnit, even container. Important top level container.
physicalDeliveryOfficeName	Office! on the user's General property sheet
sAMAccountName	sAMAccountName=bwhitman. Old NT 4.0 logon name, must be unique in the domain. Can be confused with CN.
SN	SN=Whitman. This would be referred to as last name or surname.
userAccountControl	Used to disable an account. A value of 514 disables the account, while 512 makes the account ready for logon.

userPrincipalName	userPrincipalName = bwhitman@dellkace.local Often abbreviated to UPN, and looks like an email address. Very useful for logging on especially in a large Forest. Note UPN must be unique in the forest.
Examples of Exchange Specific LDAP attributes	
legacyExchangeDN	Legacy distinguished name for creating Contacts. In the following example, Guy Thomas is a Contact in the first administrative group of GUYDOMAIN: /o=GUYDOMAIN/ou=first administrative group/cn=Recipients/cn=Guy Thomas
mail	An easy, but important attribute. A simple SMTP address is all that is required bwhitman@dellkace.local
mailNickname	Normally this is the same value as the sAMAccountName, but could be different if you wished. Needed for mail enabled contacts.
proxyAddresses	As the name 'proxy' suggests, it is possible for one recipient to have more than one email address. Note the plural spelling of proxyAddresses.
showInAddressBook	Displays the contact in the Global Address List.
Other LDAP Attributes	
c	Country or Region
company	Company or Organization Name
department	Useful category for filling in and using to filter
homephone	Home phone number
location	Important, particularly for printers
manager	Boss, manager
mobile	Mobile phone number
objectClass	Usually user or computer

OU	Organizational Unit
postalCode	ZIP or postal code
st	State, province or county
streetAddress	First line of address
telephoneNumber	Office Phone
dNSHostname	Computer.dellkace.local

LDAP Search Functions.

And, Or and Not will commonly be used in your filters, especially when your filter contains a large number of groups. Approximately equal to is not used often but can be useful when trying to filter by locations that may start with the same name such as KACE lab, KACE research, KACE management, etc.

<and> ::= '&'
<or> ::= '|'
<not> ::= '!'
<equal> ::= '='
<approx> ::= '~='
<ge> ::= '>='
<le> ::= '<='
<any> ::= '*'

Appendix B

Common OpenLDAP Attributes

Abbrev.	Name	objectClass	Description	Schema
c	countryName	country	2 character country code defined in ISO 3166	core.schema
cn	commonName	person organizationalPerson organizationalRole groupOfNames applicationProcess applicationEntity posixAccount device		core.schema
dc	domainComponent	dcObject	any part of a domain name e.g. domain.com, domain or com	core.schema
-	facsimileTelephoneNumber	residentialPerson organizationalRole organizationalPerson		core.schema
co	friendlyCountryName	friendlyCountry	full name of country	cosine.schema
gn	givenName	inetOrgPerson	First or given name	core.schema
homePhone	homeTelephoneNumber	inetOrgPerson		cosine.schema
-	jpegPhoto	inetOrgPerson	jpg format photo	inetorgperson.schema
l	localityName	locality organizationalPerson		core.schema
mail	rfc822Mailbox	inetOrgPerson	email address e.g. joe@smokeyjoe.com	core.schema
mobile	mobileTelephoneNumber	inetOrgPerson	mobile or cellular phone number	cosine.schema
o	organizationName	organization	Organization name or even organisational name	core.schema
ou	organisationalUnitName	organizationUnit	Usually department or any sub entity of larger entity	core.schema

-	owner	groupOfNames device groupOfUniqueNames		core.schema
pager	pagerTelephoneNumber	inetOrgPerson		cosine.schema
-	postalAddress	organizationalPerson		core.schema
postalCode	postalCode	organizationalPerson	Post Code or ZIP	core.schema
sn	surname	person	surname or family name	core.schema
st	stateOrProvinceName	organizationalPerson		core.schema
street	streetAddress	organizationalPerson		core.schema
-	telephoneNumber	organizationalPerson		core.schema
userPassword	-	organization organizationalUnit person dmd simpleSecurityObject domain posixAccount	User password for some form of access control	core.schema
uid	userid	account inetOrgPerson posixAccount	various - mostly username or other unique value	core.schema

OpenLDAP Object Classes

This list show the mandatory MUST and optional MAY attributes in some commonly used objectclasses

Name	Must	May	Schema
account	userid	description $ seeAlso $ localityNam e $ organizationName $ organizatio nalUnitName $ host	cosine.schema
country	c	searchGuide $ description	core.schema
dcObject	dc	-	core.schema
Device	Cn	serialNumber $ seeAlso $ owner $ ou $ o $ l $ description	core.schema
friendlyCountry	friendlyCountyName		cosine.schema
groupOfNames	member $ cn	businessCategory $ seeAlso $ owner $ ou $ o $ description	core.schema
groupOfUniqueNames	uniqueMember $ cn	businessCategory $ seeAlso $ owner $ ou $ o $ description	core.schema

inetOrgPerson	-	audio $ businessCategory $ carLicense $ departmentNumber $ displayName $ employeeNumber $ employeeType $ givenName $ homePhone $ homePostalAddress $ initials $ jpegPhoto $ labeledURI $ mail $ manager $ mobile $ o $ pager $ photo $ roomNumber $ secretary $ uid $ userCertificate $ x500uniqueIdentifier $ preferredLanguage $ userSMIMECertificate $ userPKCS12	inetorgperson.schema
locality	-	street $ seeAlso $ searchGuide $ st $ l $ description	core.schema
organizationalPerson	-	title $ x121Address $ registeredAddress $ destinationIndicator $ preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $ telephoneNumber $ internationaliSDNNumber $ facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $ postalAddress $ physicalDeliveryOfficeName $ ou $ st $ l	core.schema
Organization	o	userPassword $ searchGuide $ seeAlso $ businessCategory $ x121Address $ registeredAddress $ destinationIndicator $ preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $ telephoneNumber $ internationaliSDNNumber $ facsimileTelephoneNumber $ street $ postOfficeBox $ postalCode $ postalAddress $ physicalDeliveryOfficeName $ st $ l $ description	core.schema
organizationalRole	cn	x121Address $ registeredAddress $ destinationIndicator $ preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier $ telephoneNumber $ internationaliSDNNumber $ facsimileTelephoneNumber $ seeAlso $ roleOccupant $ preferredDeliveryMethod $ street $ postOfficeBox $ postalCode $ postalAddress $ physicalDeliveryOfficeName $ ou $ st $ l $ description	core.schema
organizationalUnit	ou	userPassword $ searchGuide $ seeAlso $ businessCategory $ x121Address $ registeredAddress $ destinationIndicator $ preferredDeliveryMethod $ telexNumber $ teletexTerminalIdentifier	core.schema

		$ telephoneNumber
		$ internationaliSDNNumber
		$ facsimileTelephoneNumber $ street
		$ postOfficeBox $ postalCode
		$ postalAddress
		$ physicalDeliveryOfficeName $ st $ l
		$ description	

person	sn $ cn	userPassword $ telephoneNumber $ seeAlso $ description	core.schema

posixAccount	cn $ uid $ uidNumber
	$ gidNumber
	$ homeDirectory
		userPassword $ loginShell $ gecos
		$ description
			nis.schema

residentialPerson		businessCategory $ x121Address
		$ registeredAddress
		$ destinationIndicator
		$ preferredDeliveryMethod
		$ telexNumber
		$ teletexTerminalIdentifier
		$ telephoneNumber
		$ internationaliSDNNumber
		$ facsimileTelephoneNumber
		$ preferredDeliveryMethod $ street
		$ postOfficeBox $ postalCode
		$ postalAddress
		$ physicalDeliveryOfficeName $ st $ l
			core.schema

Change log:

V1.0 Initial creation 6/17/11
V1.1 Added appendix 6/19/11
V1.2 Added support for nested groups 8/18/11

References:

MS LDAP Attributes: http://www.computerperformance.co.uk/Logon/LDAP_attributes_active_directory.htm

OpenLDAP Attributes: http://www.zytrax.com/books/ldap/ape/#attributes