6.3

Luna Imaging Inc.

Insight® + LUNA

LDAP Authentication

Revision 6.3

Insight® + LUNA

This document contains information proprietary to Luna Imaging, Incorporated (Luna). This document is provided to you as part of the Luna Imaging, Inc. licensed software and is governed under the same licensing restrictions that apply to that license. You may reproduce this documentation, which contains Luna Imaging, Inc. proprietary information for use within your organization. You may not disclose or distribute this documentation to third parties.

Even though Luna has tested the hardware and software and reviewed the documentation, Luna makes no warranty or representation, either express or implied, with respect to the hardware, software, or documentation, their quality, performance, merchantability, or fitness for a particular purpose. Luna has made every effort to keep the information in this manual current and accurate as of the date of publication or revision. However, Luna does not guarantee or imply that this document is error free or accurate with regard to any particular specification.

In no event will Luna be liable for direct, indirect, special, incidental, or consequential damages resulting from any defect in the hardware, software, or documentation, even if advised of the possibility of such damages. In particular, Luna shall have no liability for any programs or data stored in or used with Luna products, including the costs of recovering such programs or data.

No Luna agent, dealer, or employee is authorized to make any modification, extension, or addition to the above statements.

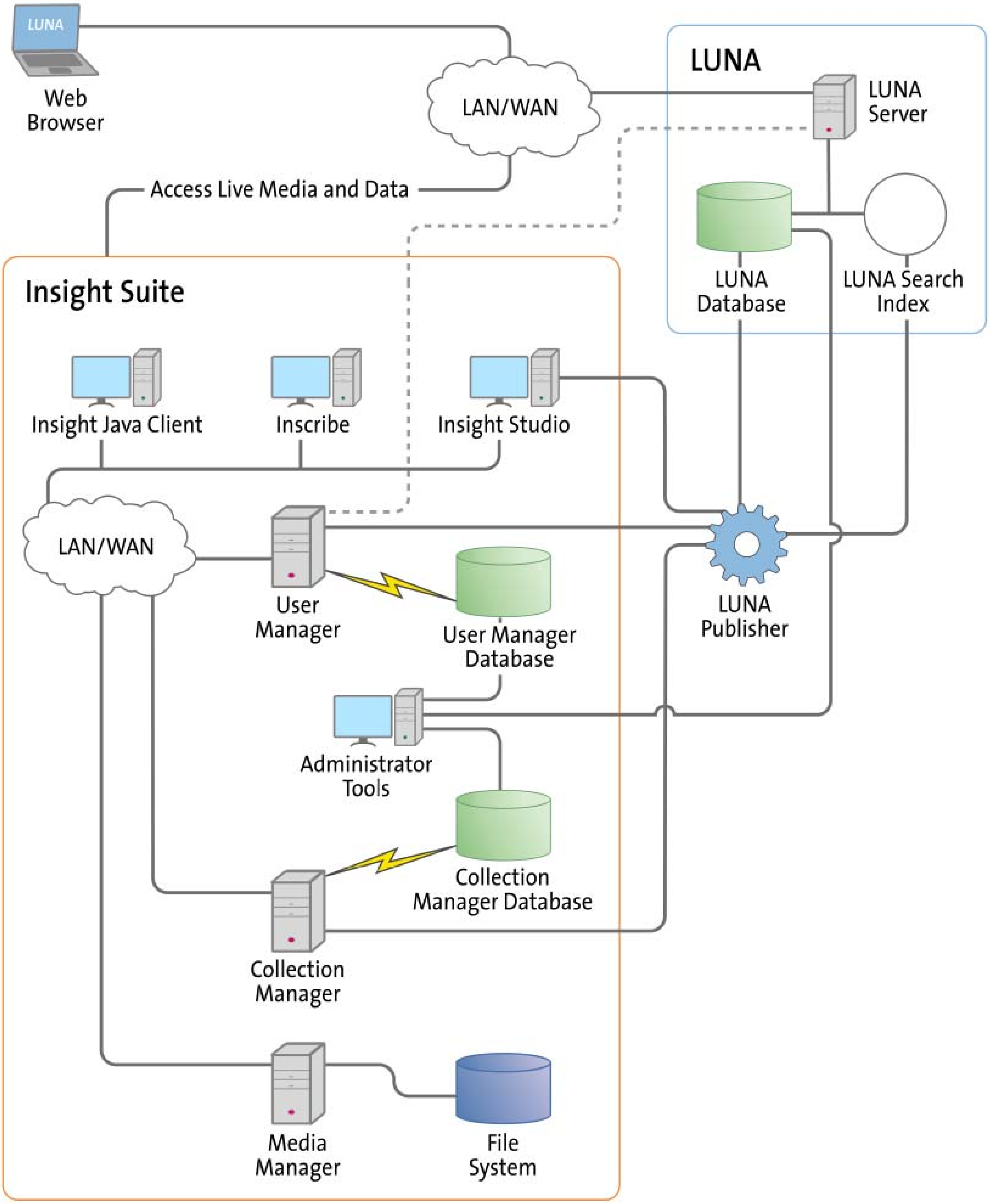
2702 Media Center Drive
los angeles, california 90065
phone 800.452.LUNA (5862) • fax .323.908.1441
www.lunaimaging.com

TABLE OF CONTENTS

Insight, LUNA and LDAP

This document is intended mainly for system administrators and secondarily for LUNA administrators to provide context and information on how to configure the NEW Insight and LUNA LDAP Authenticator. Readers are expected to have a basic knowledge of the UNIX / Windows Operating Systems as well as the structure of their institutions LDAP server.

Figure 1: Insight and LUNA Component Diagram



LUNA Components are as follows:

LUNA Publisher: Schedules and is used to pull data out of the Insight Collection Manager, storing it in an XML format. The LUNA Publisher then converts the archive data into a Lucene index for each collection and merges all Collection indexes into one single search index. Once merged, the LUNA Server is placed into a maintenance mode and the newly merged index replaces the existing one.

LUNA: LUNA is the dynamic web based front end to Insight collections. With LUNA, users can perform simple keyword searches as well as more complex structured queries, and control how they view, browse, and organize their results to create groups and presentations. LUNA includes the Web 2.0 tools your users are demanding which enable them to save and share their work by linking and embedding any view, image, group, or presentation.

LUNA Server: The LUNA Server is the server side component behind the LUNA client. Collections published to LUNA are indexed on the LUNA Server. The LUNA Publisher will automatically update the indexes based upon a schedule that you define.

Insight's Components are as follows:

User Manager: The User Manager functions as a single point of entry for all Insight Collections and Personal Insight Managers. It consolidates the functions of authentication and authorization for all shared resources within Insight and for LUNA. The User Manager can integrate with an existing security infrastructure if an institution already has a single sign-on solution in place. The User Manager also provides access to resources such as shared folders and groups within Insight. LUNA enables end-user management of viewing preferences, and storage folders for media groups and presentations.

Collection Manager: The Insight Collection Manager provides a common interface between client requests and the underlying data repository. Each Collection Manager may contain multiple collections and Virtual Collections of consistent or heterogeneous structures. The Collection Manager also functions as a broker between the clients (the Insight Java Client, Inscribe data editor, Studio (LUNA Publisher and XML Gateway) and the underlying database, coordinating search requests, and data updates.

Personal Insight Manager: The Personal Insight Manager is a specialized version of the Collection Manager which supports the creation of Personal Collections with the Insight Java Client.

Media Manager: The Insight Media Manager is built around a basic JSP Server, and manages access to Insight's media content. The Media Manager supports direct upload of processed content and also manages access to the JPEG2000 wavelet images that power Insight's Image Workspace.

Java Client: The Insight Java Client provides an additional interface for searching, viewing, and managing images, audio, video, and other media. The Java Client also provides end-user collection building tools with Personal Insight and the ability to create Virtual Collections – a subset of content with different access levels.

Inscribe®: The optional Inscribe data editor is an end-user cataloging tool designed to support all of an institution's cataloging needs. The Inscribe data editor includes built-in support for controlled vocabularies, date and numeric validation, and complex data relationships. Inscribe also supports an institution's workflow requirements by facilitating the publishing and review processes.

Studio: Studio provides collection administrators with the tools to build and manage collections. Studio is a wizard-based tool supporting batch processing of media, batch import of metadata, user management, catalog template creation and management, and other basic functions. The LUNA Publisher is accessible from within the Studio client.

Administrator Tools: The Insight Administrator Tools allow a collection administrator to dynamically manage collections. Components of the Insight Administrator Tools include tools for batch media processing, user and resource management, data indexing, and configuring of the search and user interface.

JPEG2000 Wavelet Encoder: The Administrator Tools and Studio include a JPEG2000 Wavelet Encoder. Wavelet encoded images improve performance when accessing large images, reduce storage requirements, and optimize network usage on image delivery.

Insight XML Gateway: The XML Gateway provides a web-services based interface for searching and retrieving content stored within Insight collections. The XML Gateway is middleware which allows backend interaction between Insight and other XML aware applications.

Introduction to Insight and LUNA's new LDAP Authenticator

This new LDAP authenticator was designed to be flexible and fairly simple to implement on your Insight and LUNA systems. Both Insight and LUNA have very different architectures and this authenticator was designed to work with both in a similar fashion.

The current legacy authenticators will continue to work as they have but now you have a choice to implement this new authenticator and we hope that it will serve your needs.

We would like to point out a few differences between the two platforms in which this authenticator works and what is required to make full use of it.

Insight

Only compatible with 6.0.1+ clients (Insight, Insight Studio and Inscribe) and servers.

This means if you have legacy clients, 6.0.0 and older, this authentication method will not work with them.

You need to upgrade any client that you wish to use this new authenticator with.

Rules consistent with our SimpleLDAP still apply. You will need the LDAP user mirrored in the Insight User Manager to authorize collection access.

LUNA

Version 6.0.1+

In LUNA we have added the ability to assign a credential to a specific pattern based on what is returned by the LDAP server.

This gives you great flexibility to assign privileges to groups of users and never have to edit what's in your LDAP tree.

Configuring Insight

Introduction

The following examples are configured in the InsightUserServer.dat of your 6.0.1+ Insight User Manager.

Active Directory LDAP example

```
################################################################################
#
# Security Settings
#
################################################################################

#-------------------------
# Shared Settings
#-------------------------

##
# AuthenticationHandler - the full class name of the authentication handler
# to use for authentication.
# Default Example (for normal Insight authentication): AuthenticationHandler =
com.luna.insight.client.security.DefaultAuthenticationHandler
#
# Kerberos/LDAP Example:
# AuthenticationHandler = com.luna.insight.client.security.KerberosAuthenticationHandler
#
# Simple LDAP Example:
# AuthenticationHandler = com.luna.insight.client.security.SimpleLDAPAuthenticationHandler

AuthenticationHandler = com.luna.insight.client.security.LDAPAuthenticationSearchHandler


##
# AuthorizationHandler - the full class name of the authorization handler to use for
authorization.
# Default Example (for normal Insight authentication): AuthorizationHandler =
com.luna.insight.client.security.DefaultAuthorizationHandler
#
# Kerberos/LDAP Example:
# AuthorizationHandler = com.luna.insight.client.security.LDAPAuthorizationHandler
#
# Simple LDAP Example:
# AuthorizationHandler = com.luna.insight.client.security.SimpleLDAPAuthorizationHandler

AuthorizationHandler = com.luna.insight.client.security.SimpleLDAPAuthorizationHandler


##
# LdapURL - The URL of the LDAP server.
# (note: under Windows 2000/Active Directory, this is the address of the Active Directory
machine prepended with ldap:// )
# Example: LdapURL = ldap://ldap.lunaimaging.com


#-----------------------------------------------------------------------------------
# LDAP SETTING
#
# This setting is the latest setting for LDAP since Insight 6.x.
# The SIMPLE LDAP SETTINGS listed bellow is supported as backward
# compatibility.
#-----------------------------------------------------------------------------------

# Shared setting.
# See the 'Shared Settings' section.
#

LdapURL = ldap://ldap.luna.edu

#
# Shared setting.
# See 'KERBEROS & LDAP SETTINGS - PART 1: LDAP' section.
```

```
#

BaseDN = dc=luna,dc=edu

# Shared setting.
# See 'SIMPLE LDAP SETTINGS' section.
#
LoginSSL = 0

#
# The attribute, insightUser holds the login name that a user
# entered at login time.
# This attribute should be used as it is unless the insightUser
# attribute is already being used in the LDAP for a different purpose.
#

LdapUserAttribute = insightUser

# Example for LDAPS
#
# LdapURL = ldaps://ldap.luna.edu
# LoginSSL = 1
#
# **VERY IMPORTANT** Sign lunacacert in clients' install directory
```

Open LDAP example

```
################################################################################
#
# Security Settings
#
################################################################################

#-------------------------
# Shared Settings
#-------------------------

##
# AuthenticationHandler - the full class name of the authentication handler
# to use for authentication.
# Default Example (for normal Insight authentication): AuthenticationHandler =
com.luna.insight.client.security.DefaultAuthenticationHandler
#
# Kerberos/LDAP Example:
# AuthenticationHandler = com.luna.insight.client.security.KerberosAuthenticationHandler
#
# Simple LDAP Example:
# AuthenticationHandler = com.luna.insight.client.security.SimpleLDAPAuthenticationHandler

AuthenticationHandler = com.luna.insight.client.security.LDAPAuthenticationSearchHandler


##
# AuthorizationHandler - the full class name of the authorization handler to use for
authorization.
# Default Example (for normal Insight authentication): AuthorizationHandler =
com.luna.insight.client.security.DefaultAuthorizationHandler
#
# Kerberos/LDAP Example:
# AuthorizationHandler = com.luna.insight.client.security.LDAPAuthorizationHandler
#
# Simple LDAP Example:
# AuthorizationHandler = com.luna.insight.client.security.SimpleLDAPAuthorizationHandler

AuthorizationHandler = com.luna.insight.client.security.SimpleLDAPAuthorizationHandler



#-----------------------------------------------------------------------------------
# LDAP SETTING
#
# This setting is the latest setting for LDAP since Insight 6.x.
# The SIMPLE LDAP SETTINGS listed below is supported as backward
# compatibility.
#-----------------------------------------------------------------------------------

# Shared setting.
# See the 'Shared Settings' section.
#
LdapURL = ldap://ldap.luna.edu
```

```
#
# Shared setting.
# See 'KERBEROS & LDAP SETTINGS - PART 1: LDAP' section.
#
BaseDN = dc=luna,dc=edu


# Shared setting.
# See 'SIMPLE LDAP SETTINGS' section.
#
LoginSSL = 0

#
# The attribute, insightUser holds the login name that a user
# entered at login time.
# This attribute should be used as it is unless the insightUser
# attribute is already being used in the LDAP for a different purpose.
#
LdapUserAttribute = insightUser


#
# LDAP User Search and Authentication.
#
# This part is mainly for Unix OpenLDAP although it can be used for Windows Active
# directory.
# The syntax of LdapSearchFilter x is based on RFC 2254.
# See: http://www.ietf.org/rfc/rfc2254.txt
#
# The x in LdapSearchFilter x specifies the search order.
# If the user entry cannot be located by one search, continue search
# by specifying the next search filter with LdapSearchFilter 2,
# LdapSearchFilter 3, so forth. In most cases, using only the search filter (uid={0})
# would suffice.
#
# After LdapSearchFilters run out or if there is no LdapSearchFilter,
# Insight uses the values of LdapSecurityPrincipal and
# LdapSecurityPrincipalAttributes to bind a specific user's DN by
# binding the user's login name and password.
#
#
LdapSearchFilter 1= (uid={0})
LdapSearchAttributes 1=insightUser
LdapSecurityPrincipal= {0}
LdapSecurityPrincipalAttributes= dn

# Example for LDAPS
#
# LdapURL = ldaps://ldap.luna.edu
# LoginSSL = 1
#
# Sign lunacacert in clients' install directory
```

Configuring LUNA

Introduction

The configuration for LUNA is done in the luna-security.xml (6.3+) file located in the following two places

- <LUNA Install Directory>/jboss/server/default/deploy/luna.war/WEB-INF folder
- <LUNA Install Directory>/tomcat/luna_apps/luna.war/WEB-INF

In the section that follows, only sections that are pertinent to the configuration are shown.

Active Directory LDAP example

```
<!--
    The properties of ldapAuthenticationSearch need to be specified in the
    same way as Insight User Manager 6.x. See the InsightUserServer.dat file.

    Example for Active Directory
   -->
  <bean id="ldapAuthenticationSearch"
class="com.luna.insight.client.security.ldap.LDAPAuthenticationSearchImpl">
    <property name="ldapUserAttribute" value="insightUser"/>
    <property name="ldapURL" value="ldap://ldap.luna.edu"/>
    <property name="baseDN" value="dc=luna,dc=edu"/>
    <property name="loginSSL" value="0"/>
    <property name="trustStorePath" value=""/>
<!--
    Example for LDAPS:
      <property name="ldapUserAttribute" value="insightUser"/>
      <property name="ldapURL" value="ldaps://ldap.luna.edu"/>
      <property name="baseDN" value="dc=luna,dc=edu"/>
      <property name="loginSSL" value="1"/>
      <property name="trustStorePath" value="C:/myStoreFile">
-->
  </bean>
```

Assigning credentials by pattern matching in user's DN for Active Directory LDAP

The ldapSearchFilters and credentialsConditions properties work in the same way as ldapAuthenticationSearch.

In the following example, pattern matching is applied to the DN returned after applying the search filter (sAMAccountName=(0)), where sAMAccountName is the login name for every user.

This is done in the order from top to bottom, and when there is a match, the corresponding credential under "mappedValues" is assigned and the rest of the conditions are ignored. If there is no match, it is assigned a default credential ID set above.

Regular expression can be used in conditions.

Syntax for "conditions":
  <attribute value>=<pattern>

For example, in condition "distinguishedName=CN=admin,OU=Art History*", we are trying to match the <attribute value> "distinguishedName" against the <pattern> "CN=admin,OU=Art History*"

Example DN's and expected results from the credential mappings set below:

```
distinguishedName: CN=admin,OU=Art History Department,OU=College of Letters
and Science,DC=luna,DC=edu
```

admin will get credential 4

```
distinguishedName: CN=John Smith,OU=Art History Department,OU=College of
Letters and Science,DC=luna,DC=edu
```

jsmith will get credential 3

```
distinguishedName: CN=Jenny Anderson,OU=Art History Department,OU=College of
Letters and Science,DC=luna,DC=edu
```

janderson will get credential 3

```
distinguishedName: CN=Jane Doe,OU=English Department,OU=College of Letters
and Science,DC=luna,DC=edu
```

jdoe will get credential 3

```
distinguishedName: CN=Jill Emerson,OU=French Department,OU=College of Letters
and Science,DC=luna,DC=edu
```

jemerson will get credential 2

NOTE:
For Windows Active Directory, it should be enough to specify only 1 entry, (sAMAccountName={0}) in ldapSearchFilters. Unlike OpenLDAP, specifying complicated or wrong search condition for Active Directory results in time-out and considerable system delay.

The syntax of LdapSearchFilter_x is based on RFC 2254.
See: http://www.ietf.org/rfc/rfc2254.txt

```
  <!-- Example for Active Directory
  -->
  <bean id="ldapCredentialsMap"
class="com.lunaimaging.insight.core.domain.authenticators.ldap.LdapCredentialsMap">
    <property name="ldapSearchFilters">
      <list>
        <value>(sAMAccountName={0})</value>
      </list>
    </property>
    <property name="ldapSearchAttributes">
      <list>
        <value>insightUser</value>
      </list>
    </property>
    <property name="conditions">
      <list>
        <value>distinguishedName=English Department</value>
        <value>distinguishedName=CN=admin,OU=Art History*</value>
        <value>distinguishedName=Art History*</value>
        <value>distinguishedName=College of Letters and Science</value>
      </list>
    </property>
    <property name="mappedValues">
      <list>
        <value>3</value>
        <value>4</value>
        <value>3</value>
        <value>2</value>
      </list>
    </property>
  </bean>
```

Open LDAP example

```
  <bean id="authenticatorSearchLDAP"
class="com.lunaimaging.insight.core.domain.authenticators.LdapSearchAuthenticator">
    <property name="defaultCredentialsId" value="2"/>
    <property name="ldapAuthenticationSearch">
      <ref bean="ldapAuthenticationSearch" />
    </property>
    <property name="ldapSearchMap">
      <ref bean="ldapCredentialsMap"/>
    </property>
  </bean>

  <!--
    The properties of ldapAuthenticationSearch need to be specified in the
    same way as Insight User Manager 6.x. See the InsightUserServer.dat file.

    Example for OpenLDAP:
  -->

    <bean id="ldapAuthenticationSearch"
class="com.luna.insight.client.security.ldap.LDAPAuthenticationSearchImpl">
      <property name="ldapUserAttribute" value="insightUser"/>
      <property name="ldapURL" value="ldap://ldap.luna.edu"/>
      <property name="baseDN" value="dc=luna,dc=edu"/>
      <property name="loginSSL" value="0"/>
      <property name="trustStorePath" value=""/>

<!--
    Example for LDAPS:
      <property name="ldapUserAttribute" value="insightUser"/>
      <property name="ldapURL" value="ldaps://ldap.luna.edu"/>
      <property name="baseDN" value="dc=luna,dc=edu"/>
      <property name="loginSSL" value="1"/>
      <property name="trustStorePath" value="/usr/local/insight/LunaImaging/myStoreFile">
-->

      <property name="ldapSearchFilters">
        <list>
          <value>(uid={0})</value>
        </list>
      </property>
      <property name="ldapSearchAttributes">
        <list>
          <value>insightUser</value>
        </list>
      </property>
      <property name="ldapSecurityPrincipal" value="{0}"/>
      <property name="ldapSecurityPrincipalAttributes" value="dn"/>
    </bean>
```

Assigning credentials by pattern matching in user's DN for OpenLDAP

The ldapSearchFilters and credentialsConditions properties work in the same way as ldapAuthenticationSearch.

OpenLDAP:

In the following example, pattern matching is applied to the DN returned after applying the search filter (uid=(0)), where uid is the login name for every user.

This is done in the order from top to bottom, and when there is a match, the corresponding credential under "mappedValues" is assigned and the rest of the conditions are ignored. If there is no match, it is assigned a default credential ID set above.

Regular expression can be used in conditions.

Syntax for "conditions":
<attribute value>=<pattern>

for example, in "dn=uid=admin,ou=Art History*",
we are trying to match the <attribute value> "dn" against the <pattern> "uid=admin,ou=Art History*"

Example DN's and expected results from the credential mappings:

```
dn: uid=admin,ou=Art History Department,ou=College of Letters and
Science,dc=luna,dc=edu
```

admin will get credential 4

```
dn: uid=jsmith,ou=Art History Department,ou=College of Letters and
Science,dc=luna,dc=edu
```

jsmith will get credential 3

```
dn: uid=janderson,ou=Art History Department,ou=College of Letters and
Science,dc=luna,dc=edu
```

janderson will get credential 3

```
dn: uid=jdoe,ou=English Department,ou=College of Letters and
Science,dc=luna,dc=edu
```

jdoe will get credential 3

```
dn: uid=jemerson,ou=French Department,ou=College of Letters and
Science,dc=luna,dc=edu
```

jemerson will get credential 2

The syntax of LdapSearchFilter_x is based on RFC 2254.
See: http://www.ietf.org/rfc/rfc2254.txt

```
    <bean id="ldapCredentialsMap"
class="com.lunaimaging.insight.core.domain.authenticators.ldap.LdapCredentialsMap">
      <property name="ldapSearchFilters">
        <list>
          <value>(uid={0})</value>
        </list>
      </property>
      <property name="ldapSearchAttributes">
        <list>
          <value>insightUser</value>
        </list>
      </property>
      <property name="conditions">
        <list>
          <value>dn=English Department</value>
          <value>dn=uid=admin,ou=Art History*</value>
          <value>dn=Art History*</value>
          <value>dn=College of Letters and Science</value>
        </list>
      </property>
      <property name="mappedValues">
        <list>
          <value>3</value>
          <value>4</value>
          <value>3</value>
          <value>2</value>
        </list>
      </property>
     </bean>
```