

Technical Document

NiagaraAX LDAP Guide

May 17, 2016



NiagaraAX LDAP Guide

Tridium, Inc.

3951 Westerre Parkway, Suite 350
Richmond, Virginia 23233
U.S.A.

Confidentiality

The information contained in this document is confidential information of Tridium, Inc., a Delaware corporation ("Tridium"). Such information and the software described herein, is furnished under a license agreement and may be used only in accordance with that agreement.

The information contained in this document is provided solely for use by Tridium employees, licensees, and system owners; and, except as permitted under the below copyright notice, is not to be released to, or reproduced for, anyone else.

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark notice

BACnet and ASHRAE are registered trademarks of American Society of Heating, Refrigerating and Air-Conditioning Engineers. Microsoft, Excel, Internet Explorer, Windows, Windows Vista, Windows Server, and SQL Server are registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Mozilla and Firefox are trademarks of the Mozilla Foundation. Echelon, LON, LonMark, LonTalk, and LonWorks are registered trademarks of Echelon Corporation. Tridium, JACE, Niagara Framework, NiagaraAX Framework, and Sedona Framework are registered trademarks, and Workbench, WorkPlaceAX, and AXSupervisor, are trademarks of Tridium Inc. All other product names and services mentioned in this publication that is known to be trademarks, registered trademarks, or service marks are the property of their respective owners.

Copyright and patent notice

This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc.

Copyright © 2016 Tridium, Inc. All rights reserved.

The product(s) described herein may be covered by one or more U.S. or foreign patents of Tridium.

CONTENTS

About this guide	iii
Document change log	iii
Chapter 1 Setup and configuration.....	1
Prerequisites.....	1
FAQs	2
Adding an LDAP user service	3
Setting up Fox and WebService properties.....	5
Setting up user prototypes	6
Setting up local users.....	7
Setting up simple authentication.....	9
Setting up Kerberos authentication.....	9
Setting up a client PC for Kerberos.....	10
Setting up access to the Key Distribution Center.....	12
Making sure you can connect using a browser	13
Configuring Firefox.....	13
Configuring Internet Explorer.....	14
Configuring Chrome security	15
Configuring Google Chrome startup arguments	15
Chapter 2 Introduction to LDAP	17
LDAP implementations	17
How LDAP benefits Niagara	18
Local vs LDAP users	18
Niagara properties and LDAP user attributes	19
Automatic new user creation	20
Kerberos and the single-sign-on feature	20
Logging in with found Kerberos credentials	21
Logging in without found Kerberos credentials	21
Using a browser and Kerberos to log in with a single sign on	22
Using a browser and only LDAP credentials to log in	23
Editing LDAP properties	24
Chapter 3 Reference	27
Components.....	27
Idap-ActiveDirectoryUserService.....	27
Idap-LdapUserService	30
Idap-LdapV2.....	35
Idap-LdapV3Ext.....	38
Idap-KerberosAuthenticator.....	43
Idap-PrototypeFolder.....	45
Idap-SimpleAuthenticator.....	45
Plugins (views).....	47
Idap-LdapUserManager	47

Glossary	51
Index.....	53

PREFACE

About this guide

LDAP (Lightweight Directory Access Protocol) manages user authentication using a database stored on a separate server. This guide introduces LDAP user authentication in the context of a Niagara network.

Document change log

This topic summarizes changes and additions made to this document.

May 17, 2016 — AX-3.8U1 updates:

- Reorganized some topics in the document.
- Added content to the `LdapUserService` topic, describing a new feature in AX-3.8U1, where multiple LDAP extensions can be added to the `LdapUserService` and configured for different LDAP servers.
- In the topic “Adding an LDAP User Service,” added notes to steps 3 and 10 regarding using multiple extensions.

November 6, 2013, initial version published:

- Updates to this *NiagaraAX LDAP / Active Directory Configuration Guide* replace the previous PDF-only *LDAP User Service Guide*.
- The content of this document is also available in the Workbench help system as *Doc Ldap AD* (module `docLdapAD`).

CHAPTER 1 SETUP AND CONFIGURATION

TOPICS COVERED IN THIS CHAPTER

Prerequisites

FAQs

Adding an LDAP user service

Setting up Fox and WebService properties

Setting up user prototypes

Setting up local users

Setting up simple authentication

Setting up Kerberos authentication

Setting up a client PC for Kerberos

Setting up access to the Key Distribution Center

Making sure you can connect using a browser

Configuring Firefox

Configuring Internet Explorer

Configuring Chrome security

Configuring Google Chrome startup arguments

With prerequisite information in hand, use this summary of the installation and configuration steps as a checklist to set up your stations for LDAP user management.

- Fulfill all prerequisites.
- Add the LDAP user service to each host (Supervisor and controller).
- Set up the related **Fox** and **WebService** properties in each host (Supervisor and controller)
- Set up NiagaraAX user prototypes.
- Set up a local super user and service user.
- Set up Kerberos authentication
- Set up your PC for Kerberos authentication.
- Set up each client and station to access the Key Distribution Center.
- Confirm your ability to connect using a browser.
- Configure your browser for LDAP support.

Prerequisites

Before you can configure your hosts for LDAP authentication your stations need to be licensed, you need to collect information from your LDAP and Kerberos administrators, as well as provide information to your LDAP administrator.

Licensing

Each Niagara platform (Supervisor and JACE) must be licensed for LDAP user services.

- The LDAPv2-compatible user services do not require host licensing. These are effectively the same LDAP user services provided since NiagaraAX-3.1. They do not offer Kerberos as an authentication choice.
- To use Kerberos authentication, your host platform must be licensed for LDAPv3. The following is an example of the license line:

```
<feature name="ldapv3" expiration="never" kerberos="true" parts="LDAPV3_PART"/>
```

NOTE: Kerberos authentication is not supported on any J9 Java VM platform (JACE-2/4/5 series).

LDAP environment and properties

Each Niagara host (Supervisor and controller) must be on a network with an existing LDAP server. The server must support LDAPv2 or later.

You need at least the following information from your LDAP system administrator:

- URL for the LDAP server (`ldap://your.domain.net:nnn` where `your.domain.net`:`nnn` is the URL for the LDAP server, and `nnn` is any port other than the standard, default LDAP port. To use a standard port (389, or 636 if you are using SSL/TLS), you do not need to include the port in the URL.
- User names for logging in to each station as they appear in the LDAP directory.

Information your LDAP system administrator may need from you

- The name of the user prototype (group) to associate with each user (such as, manager, operator, etc.).
- Your name for each station.

Kerberos prerequisites

You need the following information from your Kerberos administrator:

- Kerberos realm name (should be in UPPERCASE).
- Key Distribution Center URL.
- A service name (based on the station name you provided) for each station. This URL-style name must be set up by your Kerberos administrator on the LDAP server. This name should be in the form:

`http://somename.domain.com`

where `somename` is the name by which you will access your station via a browser, and `domain.com` is your realm.

This name must be trusted for delegation. If you are not planning for Kerberos authentication via the browser, you can use a regular user name (not a service).

- A keytab file or a password for each service name (station). Services typically require a keytab file, whereas users typically use a password.

FAQs

Use these questions and answers to broaden your understanding of LDAP and Niagara.

Q: Can I use SSL with LDAP?

A: Yes, in fact, you should configure platforms and stations for SSL (Secure Socket Layer) security.

For NiagaraAX running on newer JACE models and all Windows-based hosts (using the Hotspot JVM), refer to the *NiagaraAX SSL Connectivity Guide* for related details. For older JACE models (JACE-2 or JACE-4/5 series), which use the IBM J9 JVM, refer to the *NiagaraAX CryptoService (SSL)* engineering notes document. Be aware, however, that Kerberos is not supported on these older JACE models.

Q: Can a system use a combination of LDAP or Active Directory along with the network user feature in a NiagaraNetwork?

A: No. theNiagara network-user feature is incompatible with the LDAP user services (no hybrid system supported). All centralized user management is provided by the LDAP server, and each station requires a user service sourced from the **ldap** module. Local station users, which are unique to each station, are supported.

Q: Is Kerberos always associated with LDAP in Niagara?

A: Kerberos is an available authentication method for LDAPv3-compatible user services in the **ldap** module (**LdapV3UserService**, **LdapV3ADUserService**). Outside of these two LDAP-based user services, Kerberos is not used in Niagara. Other authentication methods are supported, including DIGEST-MD5, CRAM-MD5, and simple (clear text).

Q: Do the properties of an LDAP user service such as Password Strength and the various Lockout properties apply to LDAP users?

A: These properties apply only to local users created in the station. They do not apply to LDAP users. The same is true of the **Password Configuration** properties listed in the LDAP user service. These properties, which define periodic password expirations and enforce unique passwords, apply only to local station users, not to any LDAP users.

Q: Can a station support an older LDAPv2 level server or Active Directory using the newer LDAPv3-compatible user services (LdapV3UserService, LdapV3ADUserService)?

Yes. These LDAP user services are backwards-compatible with an LDAPv2-based system. However, Kerberos authentication is not available in this scenario.

Q: Can I configure my stations to run in FIPS mode (FIPS 140-2) and also use LDAPv3 with Kerberos authentication?

A: No. When running in FIPS mode, the set of permitted cryptographic algorithms is smaller—only algorithms that are FIPS-approved may be used. Due to this restriction, Kerberos cannot be used when running in FIPS mode, as the algorithms it requires are not supported by the FIPS cryptographic provider.

Adding an LDAP user service

For an existing station with the standard UserService (or an LDAP user service to be replaced by a different type), you should save and edit the station's config.bog file offline using Workbench. In the offline edit you swap in a different user service from the **ldap** palette.

Prerequisites:

Workbench is open.

You may be installing a station for the first time or upgrading an existing station.

Perform the following steps:

Step 1 Do one of the following:

- If you are upgrading the station on an existing controller, open a platform connection to the host and use the **Station Copier** tool to save a local copy of the station.
- If you are upgrading a Supervisor station located on a PC, open a local platform connection and stop the station using the **Application Manager** view.
- If you are configuring a brand new platform and station, continue with the next step.

Step 2 In the Workbench Nav tree, expand the host's file system and navigate to the **stations > stationname** folder (where *stationname* is the name of the station to which you are adding the service). Then expand the config.bog file to open its **Config > Services** container..

Step 3 Open the **ldap** palette in the side bar and determine which service you need.

NOTE: In AX-3.8U1, you can add multiple LDAP extensions to the **LdapUserService** and configure those for different LDAP servers. Note that this feature does not work with Kerberos extensions.

If you are using Windows AD (Active Directory) choose one of the following:

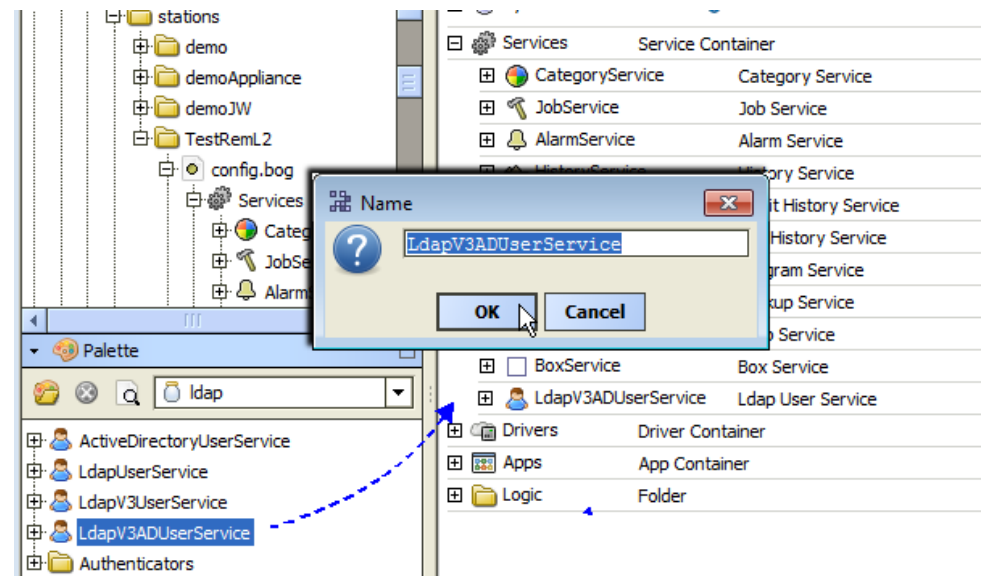
- **ActiveDirectoryUserService** supports LDAPv2 running without Kerberos authentication.
- **LdapV3ADUserService** supports Kerberos authentication.

If you are using one of the open source implementations (such as Apache Directory Server or OpenLDAP) choose one of the following:

- **LdapUserService** supports the LDAPv2 protocol without Kerberos authentication.
- **LdapV3UserService** supports the LDAPv3 protocol with Kerberos authentication.

Step 4 Drag or copy and paste the LDAP user service to the **Services** container.

The **Name** window opens.



Step 5 Type in a descriptive name or use the default name and click **OK**.

A new LDAP user service is now in the station's **Services** container.

Step 6 If a station's previous user service has local users and/or user prototypes that you would like to reuse, copy them and paste them at the same level under the new LDAP user service.

Step 7 In the station's config.bog file, select and delete the existing **UserService** component.

Step 8 To set the password for the admin user, double-click the LDAP user service and expand **admin** user.

Step 9 Assign a strong password to the local admin user.

Step 10 To use multiple LDAP extensions, simply drag the additional extensions into the **LdapUserService** at the same level as the first and configure as described above.

NOTE: Typically, multiple LDAP extensions are configured with the same domain name, but different server name(s). For example, the image below shows two Ldap-V3Ext extensions configured with different server names on the same domain (tridium.net).

Step 11 Right-click the config.bog file and click **Save**.

You may continue to work offline to do more configuration or open the station and continue configuring online.

Setting up Fox and WebService properties

The default **Authentication Scheme** for the **FoxService** is Digest, and for the **WebService** is Cookie Digest. Although these schemes provide superior communication security when compared to the basic schemes, they are not compatible with the LDAP user services. In addition to changing the default authentication schemes, you are strongly recommended to use SSL to encrypt system communications.

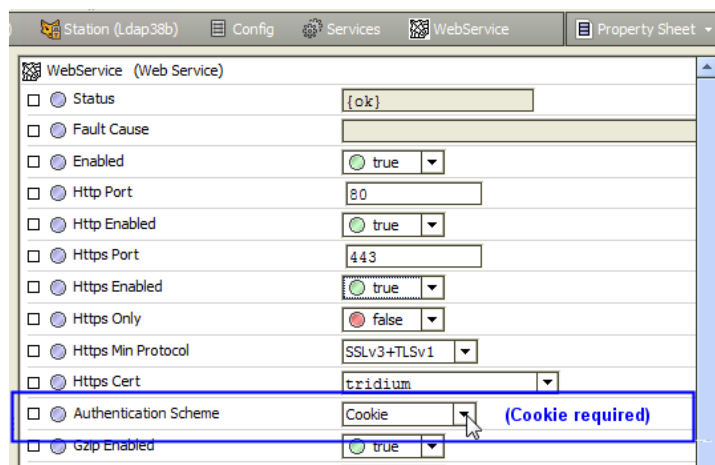
Step 1 Log in to Workbench and open the **FoxService** property sheet (right-click the **FoxService** node in the Nav tree and click **Views > Property Sheet**).

The property sheet opens.

Step 2 Change this property from Digest to Basic authentication.

Step 3 Confirm that **Foxs Enabled** is set to true.

Step 4 Open the **WebService** property sheet.



Step 5 Change the **WebService Authentication Property** from Cookie Digest to Cookie

Step 6 Confirm that **Https Enabled** is set to true.

Setting up user prototypes

When a new LDAP user logs in to a station for the first time, the system creates a user account (component) and names it based on the user name portion of the person's login credentials as stored on the LDAP server. The system populates the **Attr** (attribute) properties, such as **Full Name**, **Email**, and **Language**, directly from the LDAP server. It populates other properties, such as **Permissions**, from the local user prototype in the station.

Prerequisites: The station is open in Workbench.

A user prototype is a user record that has been pre-populated with the user properties that are not found on the LDAP server directory. Permissions are the most critical of these properties. Other user properties include **Nav File**, **Facets**, **Default Web Profile**, and **Mobile Web Profile**. This procedure documents how to set up user prototypes during initial system configuration.

Perform the following steps:

Step 1 In the Nav tree, expand an LDAP user service to see **User Prototypes**; then expand and double-click the **Default Prototype**.

- Step 2 Supply any property values that apply to all users and click **Save**.

Any LDAP users now inherit these settings, including permissions, when they log in.

- Step 3 To make a custom prototype, right-click the **Default Prototype** in the Nav tree and click **Duplicate**.

The **Name** window opens with the default name of `defaultPrototype1`.

- Step 4 Change this name to identify the user group (type of user) you are setting up, such as Manager, Operator, Engineer, etc. and click **OK**.

The **attr prototype** property (defined for each user on the LDAP server) assigns an individual user's custom prototype. This property usually defines the group to which the user belongs.

For example, if you have user prototypes named "sysIntegrator" and "buildingManager", an Ldap user who is a member of the buildingManager group on the LDAP server inherits permissions from the buildingManager prototype.

- Step 5 Repeat duplicating the **Default Prototype** and configuring properties until you have set up a separate prototype for all user groups.

LDAP users may belong to multiple groups on the LDAP server, but they can only be assigned one prototype. If an LDAP user belongs to multiple groups that match prototype names, the system uses the first prototype in the prototypes folder.

For example, if you have prototypes named "sysIntegrator" and "buildingManager", with "sysIntegrator" being first in the list, and an LDAP user who is a member of both groups on the LDAP server, the user inherits permissions from the "sysIntegrator" prototype.

- Step 6 When you are finished, save the station by right-clicking the station **Config** node on the Nav tree and clicking **Actions > Save**.

Setting up local users

A local user, such as the admin, or other super user, may log in to a station using the standard login. This type of login provides access to only the resources of the local station.

Prerequisites: Workbench is open and you are connected to the station

Step 1 To open the platform, choose one of the following:

- If you are working on a remote host, open a platform connection to the host. If you are upgrading this remote host, use the **Station Copier** tool to install the modified station back to the host.
- If you are working on a locally running station, such as on a Supervisor, open a local platform connection and **Start** the station from the **Application Manager** view.

Step 2 Allow sufficient time for the station to restart. If you are configuring a controller, a station transfer results in a controller reboot first.

Step 3 In Workbench, open a station connection to the host as the admin super user.

CAUTION: By default (unless you already changed this with the station opened offline), the admin user in any of the LDAP user services has a blank password—something you should definitely change immediately.

Step 4 Double-click the LDAP user service container in the Nav tree.

The LDAP user service property sheet opens.

NOTE: The **Password Strength** and **Lock Out**-related properties apply to local users only. LDAP users are configured in the LDAP server/system, and not Niagara. These properties also do not apply to network users, which are configured only if you are using the standard **UserService**. The **UserPrototypes** referenced here apply only to users supported by the LDAP user service.

Step 5 Create a new super user to replace the admin user.

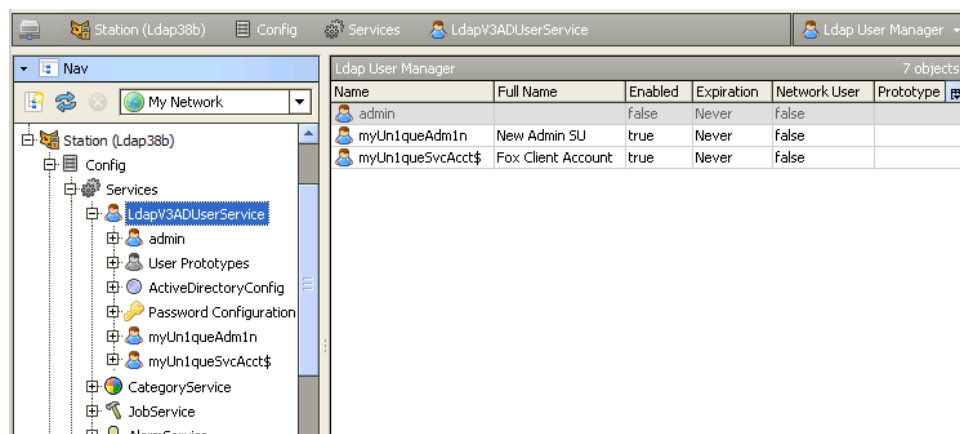
For this user you do not need to create LDAP attribute properties under the **ActiveDirectoryConfig** or **LdapConfig** containers. A local user does not appear in the LDAP directory on the LDAP server.

Step 6 After creating the new super user, disable the admin user (set the admin **Enabled** property to `false`).

NOTE: Do not disable the admin user before you create the new super user.

(For security reasons you should also consider disabling the guest user. The only people who should have access to any station are authorized local and LDAP users.)

Step 7 Create a new local service user as a super user. Assign a user name that easily identifies the host platform, and a strong password.



Step 8 Update the other stations in the network with the proper credentials under **Client Connection > NiagaraStation** to recognize the newly-configured station.

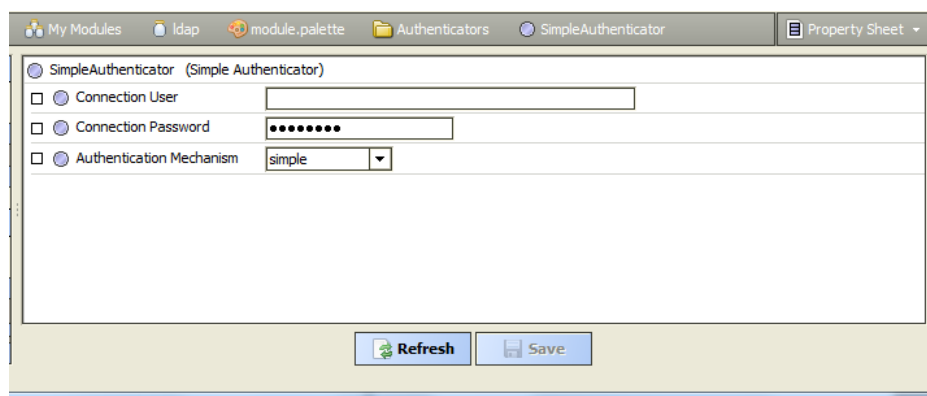
Your local station now has two local users in addition to the now disabled admin user and the standard guest user.

Setting up simple authentication

A few properties need to be configured to set up simple authentication.

- Step 1 Log in to Workbench as a super user.
- Step 2 If the authenticator is the **KerberosAuthenticator**, delete it (right-click it and click **Delete**).
- Step 3 If needed, open the **Ldap** palette, expand the **Authenticators** folder, drag the **SimpleAuthenticator** to the station's **ActiveDirectoryConfig** or **LdapCconfig** component in the Nav tree, accept or name the authenticator and click **OK**.
- Step 4 Right-click the **SimpleAuthenticator** and click **Views > Property Sheet**.

The **SimpleAuthenticator** property sheet opens.



Step 5 Fill in the other properties and click **Save**.

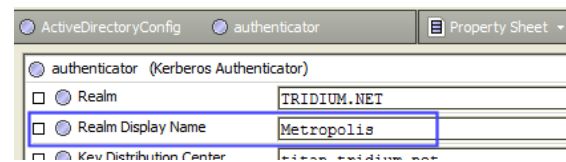
Setting up Kerberos authentication

A number of properties need to be configured to set up Kerberos authentication. The main ones include defining the realm, the domain name for the Key Distribution Center, and the station name as it appears in the Kerberos database.

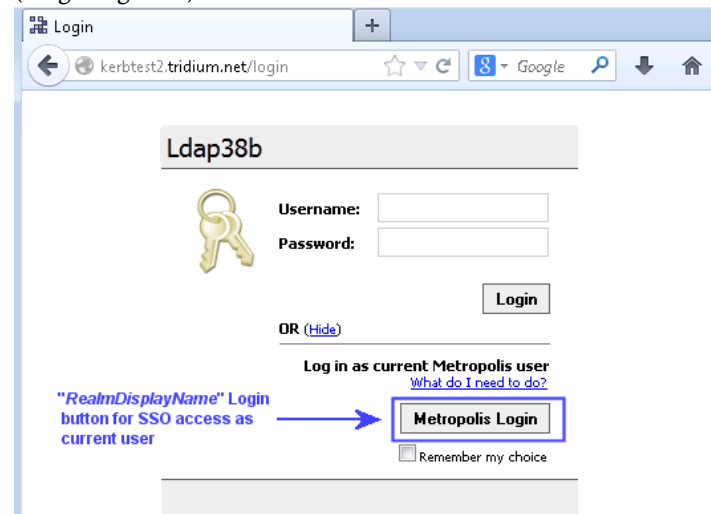
- Step 1 Log in to Workbench as a super user.

- Step 2 Right-click the **KerberosAuthenticator** under an LDAP user service in the Nav tree and click **Views > Property Sheet**.

The Kerberos properties are ready to configure.



The **Real Display Name** property configures what appears on and above the Kerberos login button when logging in to a station using Kerberos authentication and SSO (Single Sign On).



In the example the Realm Display Name on the login button is “Metropolis.”

- Step 3 Fill in the other properties and click **Save**.

NOTE: Kerberos is very particular about names. You must enter all Kerberos names exactly as they appear in the Kerberos database. Upper and lowercase can sometimes be an issue, so make sure you have an exact match.

Setting up a client PC for Kerberos

For any computer to access (as a client) a station that supports Kerberos authentication, you must update a Kerberos configuration file (`krb5`) in the PC with the default realm and which flags to set on acquired tickets. (Kerberos authentication requires the ability to acquire Kerberos tickets that can be forwarded.) In addition, you must update the Windows registry.

- Step 1 Locate your `krb5.ini` (Windows) or `krb5.conf` (Linux) file.

- On a Windows host, you may find the `krb5.ini` file in: `c:\winnt\krb5.ini` or `c:\windows\krb5.ini`.
- On a Linux host, find the file at: `/etc/krb5.conf`.

- Step 2 If you do not have a `krb5.ini` or `krb5.conf` file on your PC do one of the following:

- Create it and store it at one of the locations mentioned above.

- Create a `krb5.conf` file and store it in this location: `<java_home>\lib\security\krb5.conf` (Windows)
`<java_home>/lib/security/krb5.conf` (Linux and Solaris)

Use the `.conf` extension for all operating systems.

Step 3 Add this line to the `[libdefaults]` section of this file:

```
forwardable=true
```

If the `krb5` file does not have this section, add the following lines at the top of the file:

```
[libdefaults]
forwardable=true
```

If you created your own `krb5` file, the file requires only these two lines.

Step 4 Save the file.

If your PC is running Windows XP SP2 or higher, and you would like to access your native Kerberos ticket, you may set a registry key to allow Java to access the ticket.

Step 5 Before setting a registry key, back up your Windows registry.

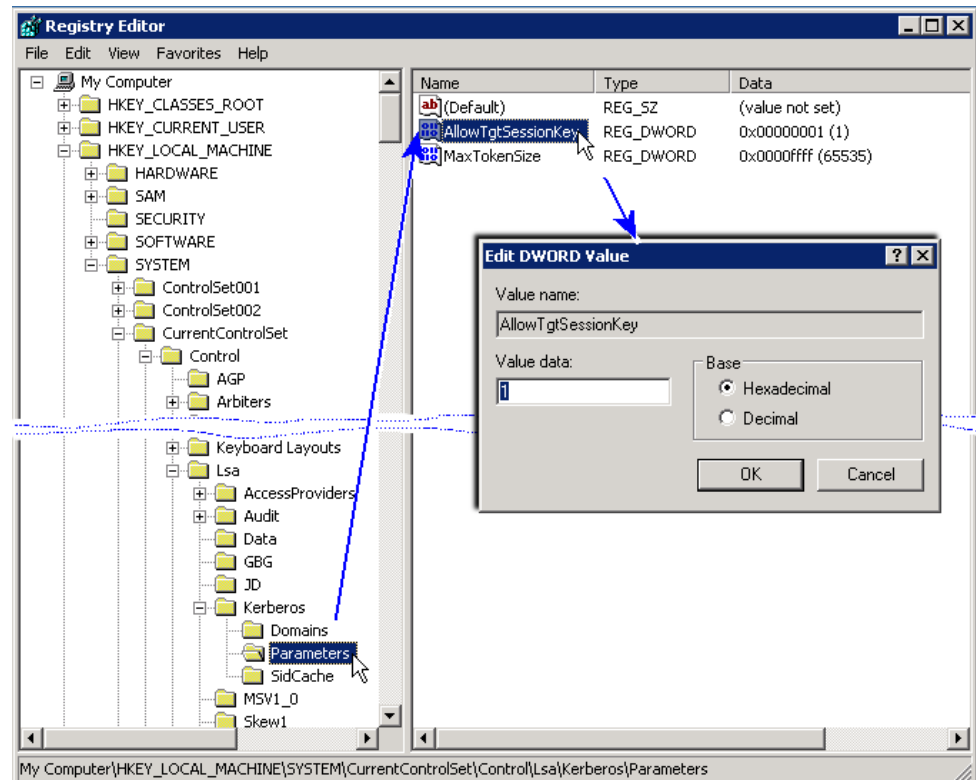
Step 6 To set the key, start the registry editor (Start > Run... `regedit`) and add or edit the following key:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\
Kerberos\Parameters
```

```
Value name: AllowTgtSessionKey
Value type: REG_DWORD
Value: 0x01
```

If you are configuring Windows XP, add or edit this key:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos
Value name: AllTgtSessionKey
Value type: REG_DWORD
Value: 0x01
```



NOTE: If it ever becomes necessary, you can return to the default Windows security setting by changing the value of this registry key to zero (0).

If you are working with Linux, some systems may require a more advanced `krb5.conf` file. If necessary, have your Kerberos administrator set this file up for you.

Setting up access to the Key Distribution Center

Kerberos authentication issues authentication tickets, which the system uses in a similar manner to private-key authentication. Ticket processing involves retrieving a key from a KDC (Key Distribution Center). Kerberos uses reverse DNS (Domain Name System) to find the referenced Key Distribution Center. You must specify a reverse DNS entry for both the client and station DNS servers. Otherwise, users are unable to acquire Kerberos tickets and log in. This procedure documents how to configure both a PC client and station to access a KDC. While modifying the `hosts` file is simple enough for a single station, and can be useful for testing your Kerberos setup, this approach can be tedious and prone to error when dealing with multiple stations and multiple client machines. Setting up DNS servers with reverse DNS entries is the best option, if available.

Perform the following steps:

Step 1 Contact your IT administrator to see if the appropriate entry exists on the LDAP server.

If you do not have a workable reverse DNS entry, you may configure an entry in the `hosts` file on each client PC and station. This entry maps the IP address of the Key Distribution Center.

On Windows PCs, the `hosts` file is located at `C:\Windows\System32\drivers\etc\hosts`.

On Linux hosts it is located at: `/etc/hosts`.

- Step 2 Add the following entry in your client `hosts` file:

```
nnn.nnn.nnn.nnn kdc.domain.net
```

where `nnn.nnn.nnn.nnn` is the IP address of the KDC and `kdc.domain.net` is the domain name.

- Step 3 On each platform, use the platform **TCP/IP Configuration** view (or equivalent view on the station's **TcpIpPlatformService**) to access and edit the `hosts` file with the same entry.

Making sure you can connect using a browser

Kerberos processes names and not IP addresses. The IP address of your PC must map to the name of the service you intend to use.

- Step 1 Attempt to connect to the station using a fully-qualified domain name: `http://some.domain.com/somepage`, where

`some.domain.com/somepage` is the LDAP server's domain name and home page name.

- Step 2 If you are unable to connect, edit your client PC's `hosts` file to add an entry similar to:

```
nnn.nnn.nnn.nnn some.domain.com,  
where nnn.nnn.nnn.nnn is the IP address of the LDAP server,  
and some.domain.com is the domain name of the server.
```

For example,
`172.16.10.10 kerbtest2.tridium.net`

This Ip address maps to the Kerberos service associated with `kerbtest2` on `tridium.net`.

NOTE: Configuring mapping in the `hosts` file is acceptable for testing, but not so good once the site is live and many people need to access it.

Configuring Firefox

Using Firefox for browser access (as a Kerberos authenticated LDAP user) to a station requires that you add to the browser's security configuration the stations to which you wish the browser to connect .

- Step 1 Open a Firefox window.

- Step 2 Type `about:config` in the location bar and press **Enter**.

If a warning appears, continue (promise to be careful).

- Step 3 In the **Search** box near the top of the page type `negotiate`.

This filters the Firefox configuration attributes to six or seven. You need to edit these entries:

```
network.negotiate-auth.delegation-uris  
network.negotiate-auth.trusted-uris
```

- Step 4 Include the URLs of the station(s) that the browser needs to be able to access. Use a comma to separate multiple stations.

For example, if two stations have the following URLs: `http://host1.domain.com/somepage`, and `http://host2.domain.com/somepage`, enter the URLs as follows:

`host1.domain.com, host2.domain.com`

.

Firefox is ready for Kerberos authentication and you should be able to log in to stations without being prompted for a user name and password.

NOTE: Only Windows-based stations (Supervisor, AX SoftJACE, or JACE-NXT) and Linux-based stations support the SSO (Single Sign On) feature from a browser. QNX-based JACE stations do not support this feature. Instead, they require an LDAP user login.

Configuring Internet Explorer

To configure Internet Explorer on a client LDAP host to use Kerberos, you must change security settings.

Step 1 Open an Internet Explorer window.

Step 2 Using the menu bar, click **Tools > Internet Options**.

The **Internet Options** window opens.

Step 3 Click the **Security** tab and select the **Local intranet** zone.

Step 4 Click **Sites > Advanced**.

The **Add a website to this zone** window opens.

Step 5 Type in the URL for a station and click **Add**.

`http://host1.domain.com`

where `host1.domain.com` is the station's URL.

If you have multiple stations to add, continue typing in URLs and clicking **Add**

.

Step 6 To return to the **Security** tab, click **Close > OK**.

Step 7 With the **Local intranet** zone selected, click the **Custom level...** button.

The **Security Settings — Local intranet** window opens.

Step 8 To use Kerberos authentication without a prompt, scroll down to the **User Authentication** section (near the bottom), and click to enable **Automatic logon only in Intranet zone**.

If you prefer to be prompted, enable **Prompt for user name and password**.

Step 9 To close **Internet Options**, click **OK** twice.

Internet Explorer should now be ready for Kerberos authentication and you should be able to log in to stations without being prompted for a user name and password.

NOTE: Only Windows-based stations (Supervisor, AX SoftJACE, or JACE-NXT) and Linux-based stations support the SSO (Single Sign On) feature from a browser. QNX-based JACE stations do not support this feature. Instead, they require an LDAP user login.

Configuring Chrome security

Using Google Chrome for browser access (as a Kerberos authenticated LDAP user) to stations requires some client side setup.

NOTE: If you previously configured Internet Explorer to use Kerberos for LDAP access to stations, this may already be done. However, the Chrome startup arguments still need configuration.

Perform the following steps:

- Step 1 Open a Google Chrome window.
- Step 2 Click **Customize > Settings** or type `chrome:settings` in the location bar and press **Enter**.
The **Chrome Settings** page opens.
- Step 3 Near the bottom of the page, click **Show advanced settings...**, scroll down to the **Network** section and click the **Change proxy settings...** button.
The **Internet Options** window opens.
- Step 4 Click the **Security** tab, select the **Local intranet** zone, click **Sites > Advanced**.
The **Add website to this zone** window opens.
- Step 5 Type in the URL for a station and click **Add**.
`http://host1.domain.com`
If you have multiple stations to add, continue typing in URLs and clicking **Add**.
- Step 6 To return to the **Security** tab, click **Close > OK**.
- Step 7 With the **Local intranet** zone selected, click the **Custom level...** button.
The **Security Settings — Local intranet** window opens.
- Step 8 To use Kerberos authentication without a prompt, scroll down to the **User Authentication** section (near the bottom), and click to enable **Automatic logon only in Intranet zone**.
If you prefer to be prompted, enable **Prompt for user name and password**.
- Step 9 To close **Internet Options**, click **OK** twice.
- Step 10 Close all Chrome windows.

Configuring Google Chrome startup arguments

When you start Google Chrome after configuring its security properties, you need to add two arguments to the browser's white list.

- Step 1 Do one of the following:
 - If you are starting Chrome from a command line, append the arguments above to the end of the command.
 - If you are starting Chrome from a shortcut, continue with this procedure.
- Step 2 Right-click the shortcut used to start Chrome and click **Properties**.
- Step 3 From the **Shortcut** tab, click in the **Target** field and click **End** (to go to the end of the field).

Step 4 Append these arguments to the command after any quotation marks that may already be there.

- `-auth-negotiate-delegate-whitelist="host1.domain.com", host2.domain.com" etc.`
- `-auth-serverwhitelist="host1.domain.com", "host2.domain.com" etc.`

where the URL for the station(s) is in quotation marks. If you have multiple stations to define, use a comma to separate them.

Step 5 To save the shortcut, click **OK**.

Google Chrome should now be ready for Kerberos authentication. You should be able to log in to stations without being prompted for a user name and password.

NOTE: Only Windows-based stations (Supervisor, AX SoftJACE, or JACE-NXT) and Linux-based stations support the SSO (Single Sign On) feature from a browser. QNX-based JACE stations do not support this feature. Instead, they require an LDAP user login.

CHAPTER 2 INTRODUCTION TO LDAP

TOPICS COVERED IN THIS CHAPTER

LDAP implementations

How LDAP benefits Niagara

Local vs LDAP users

Niagara properties and LDAP user attributes

Automatic new user creation

Kerberos and the single-sign-on feature

Logging in with found Kerberos credentials

Logging in without found Kerberos credentials

Using a browser and Kerberos to log in with a single sign on

Using a browser and only LDAP credentials to log in

Editing LDAP properties

LDAP (Lightweight Directory Access Protocol) uses a separate server to provide an IP-network-accessible, hierarchical, and distributed database for storing information about authorized system users and their access privileges. Many network hosts can use the LDAP services, which are administered from a central location.

LDAP implementations

LDAP is an open, vendor-neutral, industry standard application protocol for accessing and maintaining distributed directory information services for an IP network. A common usage of LDAP is to provide a single sign on where a single user logs in to multiple network services using but one password.

Niagara supports two LDAP server implementations:

- Windows AD (Active Directory)

This widely-implemented type of LDAP server is a Microsoft-supplied service used on Windows domain networks, and is included in most Windows Server operating systems. AD provides an interface for these protocols: LDAP (LDAPv2 or LDAPv3) and Kerberos (for authentication). With AD, users can access resources anywhere on the network with a single login.

The Windows AD is structured as a hierarchical tree of objects.

To integrate a Windows AD system with a network of Niagara stations, you replace the standard **UserService** in each station with one of the following user services:

- **ActiveDirectoryUserService** is for ADs that support LDAPv2 running without Kerberos authentication.
- **LdapV3ADUserService** is for ADs that support Kerberos authentication. The host Niagara platform must be licensed for LDAPv3. If Kerberos authentication is used, the LDAPv3 requires the attribute: `Kerberos="true"`.
- Open source implementations

These implementations, including Apache Directory Server and OpenLDAP, support both LDAPv2 and LDAPv3 (with the possibility of Kerberos authentication).

Each of these implementations is structured as a hierarchical tree of objects. Each object has a set of attributes.

To integrate any of these LDAP implementations with a network of Niagara stations you replace each station's standard **UserService** with one of the following LDAP user services:

- **LdapUserService** supports the LDAPv2 protocol without Kerberos authentication.

- **LdapV3UserService** supports the LDAPv3 protocol with the availability of Kerberos authentication. If Kerberos authentication is used, the LDAPv3 requires the attribute: `Kerberos="true"`.

How LDAP benefits Niagara

LDAP communicates record-based, directory-like data between programs. It defines database access permissions and provides a schema, which is a way to describe the format and attributes of data stored in a server.

Corporate and campus installations that already use Windows Active Directory, or other LDAP-based directory services to manage user access across distributed network resources, can benefit from configuring Niagara stations to use an LDAP user service. Benefits include:

- Ease of implementation. Installations that already use Windows AD or an open-source implementation of LDAP can easily include Niagara stations in their existing user management configuration.
- Automatic new user account creation. When a user logs in to a station for the first time, the system automatically creates a user account (component) in the station and populates it with pre-defined properties (based on user prototype), such as permissions, and predefined LDAP properties (from the LDAP server), such as email address, full name, and language.
- Security. Kerberos authentication (available for LDAPv3-based AD or open source systems) offers a high level of security. Implementing Kerberos requires client setup of hosts and browsers.
- Simplified login. Current users may log in without needing to enter credentials.

NOTE: All stations on the network (both Supervisors and controllers) must use the LDAP server. The system does not support a mixture of stations using the standard **UserService** with other stations using an LDAP user service.

Local vs LDAP users

Once an LDAP user service is configured and running, most user access to a station comes from LDAP users. However, most configurations benefit from at least two regular station users that are not dependent upon LDAP server communications.

The two local users are:

- A replacement user for the admin user. The name “admin” is commonly used and easy for hackers to guess. Creating a new local super user with a unique name and strong password is a simple way to improve overall system security.
- A local service user you can reference in other remote stations when configuring the **Client Connection** properties under the remote station’s **NiagaraStation** device.

In theory, an LDAP user could serve as a service user, however, this is not recommended. A local service user makes the initial configuration of a **NiagaraNetwork** more straightforward and provides immunity from station-to-station communication issues that might arise, say from LDAP password expiration rules, or in the unlikely event of LDAP server problems.

NOTE: Do not allow any person to log in to the station using this user account. A service user is only for Fox station-to-station communications.

Niagara properties and LDAP user attributes

An LDAP server maintains a directory of information about system users. Each entry (record) in an LDAP directory consists of multiple attributes, which may or may not be assigned values. Users within a Niagara network require additional properties, such as permissions and facets that apply only within the Niagara context.

A sample LDAP user entry might contain the following information from the LDAP server:

Figure 1. Example LDAP directory record

```
User: jdoe
uid: jdoe
Fname: John Doe
Position: Software Engineer
Address: 123 Fake Street
Email: jdoe@email.com
Preferred Language: German
Groups: Engineering
```

Several key configuration properties in each of the Niagara LDAP user services correspond directly to the names of attributes in the LDAP directory. You can find them in the **ActiveDirectoryConfig** and **LdapConfig** containers (depending on the user service).

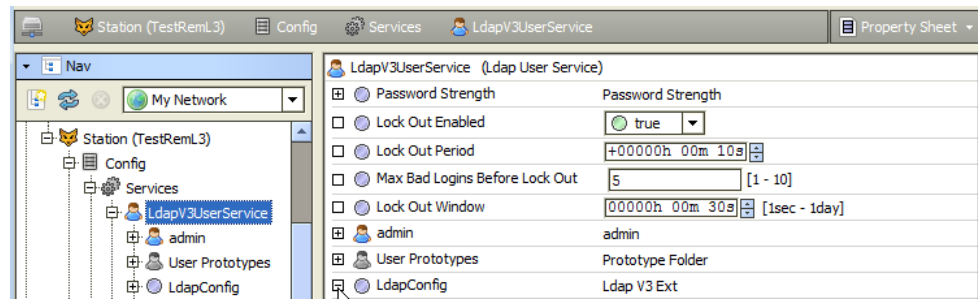
Figure 2. Attr (attribute properties in LdapConfig)

Property	Value
Enable Connection Pooling	true
Connection Url	ldap://zeus.example.net
SSL	false
User Login Attr	uid
User Base	DC=domain, DC=net
Attr Email	mail
Attr Full Name	fullName
Attr Language	preferredLanguage
Attr Cell Phone Number	member
Attr Prototype	mobile
Cache Expiration	1+00168h 00m 00s

The Niagara property names for these LDAP properties begin with **Attr** (attribute). These fields should correspond to the names of attributes in the LDAP directory. The system pulls the values for these properties from the LDAP directory on the LDAP server and uses them to fill out information about the user.

In the example above, the station user is `jdoe`. To populate the **Full Name** property value, you enter `displayName` in the **Attr Full Name** field.

The Niagara user properties that are not maintained by the LDAP server appear in the LDAP user service property sheets above the **ActiveDirectoryConfig** and **LdapConfig** containers.

Figure 3. NiagaraAX user properties

Automatic new user creation

All users must exist in the LDAP directory on the LDAP server. When a new employee joins your team, make sure you set them up in the LDAP server before they attempt to log in to a station. An appropriate user prototype that contains default Niagara properties for each type of user should exist in each station. (User prototypes allow you to group users, for example: manager, operator, engineer, etc.).

When a new user logs in to a station for the first time, the Niagara system automatically creates a new user account (component) in the station. It uses the user name that the person logged in with (the person's user login name on the LDAP server) as the account name. The system populates (maps) this component's properties from two sources:

- It populates the properties the attr properties with attributes supplied from the LDAP server.

One of those attributes identifies the group within your organization to which the user belongs. This attribute is the **Attr Prototype**.

- Niagara uses the **Attr Prototype** name to identify the user prototype in the station from which to populate the component's local user properties, including user permissions, facets, Nav file, default Web and Mobile profiles, and other specific properties required by a station.

For Active Directory, this is the `memberOf` attribute.

NOTE: For the automatic populating of Niagara user properties to work, the name of the **Attr Prototype** in the LDAP server must exactly match the name of a user prototype in the station.

- If the **Attr Prototype** does not match a user prototype name or this property is blank, the system uses the **Default Prototype** as its source.

An LDAP user may be a member of multiple groups.

Kerberos and the single-sign-on feature

Niagara supports Kerberos authentication when logging in to a station. Kerberos is a widely used authentication protocol that helps to keep your credentials and station safe.

SSO (Single Sign On) is an access control feature of Kerberos that allows the automatic logging in to multiple related, but independent software systems. When you use a browser to log in with LDAP and Kerberos, you provide a single set of credentials and receive a ticket, which allows you to automatically gain access to all networked stations. SSO also makes it possible to log in to individual stations without being prompted for user name or password each time.

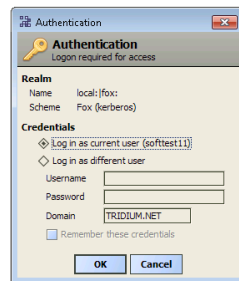
Logging in with found Kerberos credentials

Kerberos is an open-source computer network authentication protocol that uses tickets to verify the identity of users before allowing them to access network resources.

Prerequisites: Your client host (your PC) is part of the same LDAP realm as the station.

Step 1 Launch Workbench and open a platform or station.

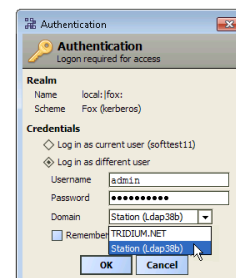
If Workbench is able to acquire your native Kerberos credentials, it displays the **Authentication** window.



Step 2 Do one of the following:

- If you are the current user, click **OK** to log in.
- To log in as a different LDAP user or as a local user, click to select **Log in as different user**.

The system enables the credential fields.



On the left is the login as a different LDAP user. On the right is the login as a local station user.

Step 3 If you are logging in as a different user do one of the following.

- To log in as a different LDAP user with Kerberos authentication, leave the **Domain** field set to the current realm as shown on the left above.
- To log in as a local station user, use the drop-down list to change the **Domain** name to your station name (in parentheses) as shown on the right above.

Step 4 Enter your user name and password.

You are logged in using Kerberos authentication.

Logging in without found Kerberos credentials

There may be any number of reasons why the system may not be able to find a user's Kerberos credentials on the LDAP server. The LDAP server may not contain a record for the user, or there may be some other issue.

Step 1 Launch Workbench and open a platform or station.

If Workbench is unable to acquire your native Kerberos credentials, it displays a simple login window.



The example on the left is the network login window when Kerberos credentials are not found. On the right is the local login window when Kerberos credentials are not found. These windows do not provide an option to log in as the current user.

Step 2 Do one of the following:

- To log in using Kerberos authentication (left screen capture above), leave the **Domain** field set to the current realm and enter your credentials (your LDAP user name and password).
- To log in as a local station user (right screen capture above), use the drop-down list to change the **Domain** name to your station name (in parentheses) and enter your local credentials.

Using a browser and Kerberos to log in with a single sign on

This procedure provides steps to log in with a single sign on.

Prerequisites: Your controller is a Hotspot JACE (including the JACE 3/6/7 series) that supports Kerberos tokens sent by browsers as required by the SSO login.

Step 1 Launch a browser and open a platform or station.

By default you may see a login window similar to the following:



Step 2 Do one of the following:

- To log in as the current user, click the realm display login button.

For this choice to work, the station must reside on the same realm that you are on. For example, if you are logged in to the FACTORY realm, the system cannot use your credentials to access a station set up for the HQ realm.

CAUTION: If a station is not set up to use the same realm as your currently-logged-in user, you enter your Kerberos/LDAP credentials directly into the credentials fields, and click the upper **Login** button, the system sends your credentials to the station as plain text, where the station then takes care of the Kerberos authentication. You are strongly recommended to use SSL (NiagaraAX) or TLS (Niagara 4). These protocols encrypt the transmission to, at least, protect the integrity of your password.

- For SSO access, go to the `/login-kerb` page (instead of this default `/login` page) and the system directly logs you in to the station without having to click a **Login** button.

Step 3 If you can successfully log in using SSO and want to bypass the login window in the future, click to select the Remember my choice check box at the bottom of the window.

This is effectively the same as going to the station's `/login-kerb` page.

Step 4 If you are not using the SSO feature, click the OR (Hide) link.

This reduces the size of the login Window to include just credentials.

The screenshot shows a login window titled 'Ldap38b'. It contains a yellow key icon, a 'Username:' field, a 'Password:' field, and a 'Login' button. Below the password field is a link that says 'OR (Show)'.

Step 5 To restore the login window, click the OR (Show) link.

Browser cache maintains the configuration of the last-used login window.

NOTE: The SSO feature from a browser does not work if accessing a station on a QNX-based JACE. It only works on a station running a Windows-based host.

Using a browser and only LDAP credentials to log in

There may be several reasons to log in with your LDAP credentials each time you access a controller or station. One reason is that your controller is QNX-based and uses Java-5, which does not support Kerberos tokens sent by browsers. Other reasons are that you wish to log in as a different LDAP user, or you wish to log in as a local station user (such as admin).

Step 1 Open your browser and enter the IP address for your controller in the locator field.

The system displays a simple login window.

The screenshot shows a login window titled 'Ldap38b'. It contains a yellow key icon, a 'Username:' field, a 'Password:' field, and a 'Login' button.

Step 2 Enter your credentials and click the **Login** button.

The SSO login features (toggled with the Show and Hide links) do not appear in the login window to a QNX-based JACE station.

CAUTION: Anytime you chose to log in by entering credentials and clicking the **Login** button), you are strongly recommend to use communication security (SSL/TLS), so as to at least protect your password with encryption.

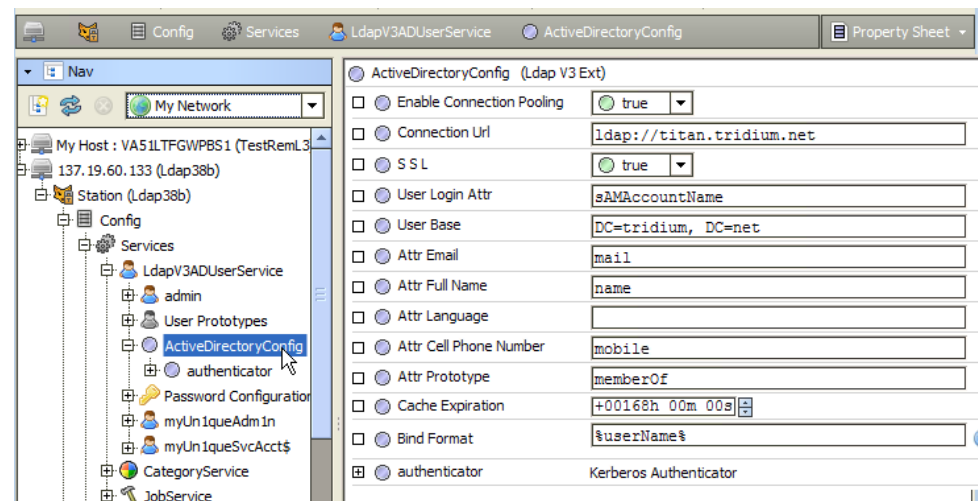
Editing LDAP properties

While the system populates the LDAP attribute properties with data from the LDAP server, you may from time to time need to edit individual LDAP properties. Attribute changes you make under the **ActiveDirectoryConfig** and **LdapConfig** containers of the LDAP user service property sheets are updated on the LDAP server.

Prerequisites: The station is open in Workbench.

- Step 1 Expand the LDAP user service in the Nav tree and double-click the **ActiveDirectoryConfig** or **LdapConfig** container.

The LDAP property sheet opens.



- Step 2 Configure the properties for your first user and click **Save**.

The next step is to configure authentication. The default authenticator is **KerberosAuthenticator**. Instead, you can use a **SimpleAuthenticator**.

- Step 3 If you are using **SimpleAuthenticator**, right-click the default **KerberosAuthenticator** and click **Delete**, then drag the **SimpleAuthenticator** from the **ldap** palette to the **ActiveDirectoryConfig** or **LdapConfig** property sheet and name it appropriately.

- Step 4 To view authentication properties, double-click the authenticator container.

The authenticator property sheet opens.

The image shows two overlapping configuration windows from a software interface. The top window is titled 'SimpleAuthenticator (Simple Authenticator)' and contains the following fields: 'Connection User' (empty), 'Connection Password' (masked with dots), and 'Authentication Mechanism' (set to 'simple'). The bottom window is titled 'KerberosAuthenticator (Kerberos Authenticator)' and contains the following fields: 'Realm' (set to 'EXAMPLE.COM'), 'Realm Display Name' (empty), 'Key Distribution Center' (set to 'kdc.example.com'), 'Station Kerberos Name' (set to 'station name'), 'Station Kerberos Password' (masked with dots), and 'Key Tab Location' (set to 'null'). Both windows have 'Refresh' and 'Save' buttons at the bottom. The interface also shows a breadcrumb trail at the top: 'My Modules' > 'ldap' > 'module.palette' > 'Authenticators' > 'SimpleAuthenticator'.

NOTE: You are strongly recommended to use SSL/TLS with simple authentication. With SSL/TLS, at least the system communicates credentials in clear text over an encrypted connection.

CHAPTER 3 REFERENCE

TOPICS COVERED IN THIS CHAPTER

Components Plugins (views)

The topics that follow provide detailed documentation for each component and plugin that supports this system feature.

Components

Components include services, folders and other model building blocks associated with a module. You may drag them to a property or wire sheet from a palette.

The descriptions included in the following topics appear as headings in documentation. They also appear as context-sensitive help topics when accessed by:

- Right-clicking on the object and selecting **Views > Guide Help**
- Clicking **Help > Guide On Target**

Following is a list of the components in the **template** module:

Idap-ActiveDirectoryUserService

This component supports LDAPv2 client access to a Windows AD (Active Directory) server. Connection to this server uses simple authentication. The child **ActiveDirectoryConfig** component holds all properties needed to configure connection to the AD server.

Figure 4. ActiveDirectoryUserService (ActiveDirectoryConfig) properties

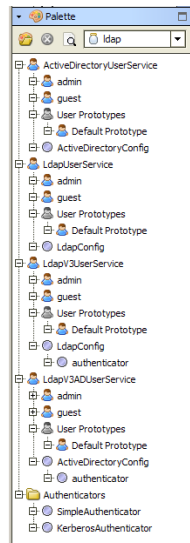
Property	Value
Enable Connection Pooling	<input checked="" type="checkbox"/> true
Connection Url	ldap://domain.net
SSL	<input type="checkbox"/> false
User Login Attr	sAMAccountName
User Base	DC=domain, DC=net
Attr Email	mail
Attr Full Name	name
Attr Language	
Attr Cell Phone Number	mobile
Attr Prototype	memberOf
Cache Expiration	+00168h 00m 00s
Domain	domain.net

Property	Value	Description
Enable Connection Pooling	true or false	Setting this property to true allows connections to be shared and reused. This can improve performance.
Connection URL	ldap://your.domain.net or ldap://your.domain.net:nnn	Identifies the URL (your.domain.net) for the LDAP server. Standard LDAP ports are 389, or 636 (if using SSL). If the server uses a non-standard port, include the port (your.domain.net:nnn) in the URL, for example, ldap://your.domain.net.999..
SSL	true or false	Enables and disables secure communication. If set to true, make sure that SSL (3.8) or TLS (4.0) is enabled in the station's FoxService (for Workbench-to-station access) and WebService (for browser-to-station access).
User Login Attr	Text For AD this value defaults to sAMAccountName.	Identifies the specific attribute in the LDAP directory to store the LDAP user logon name. For AD servers this is always sAMAccountName. For OpenLDAP servers, it would be uid.
User Base	Domain components	Identifies the sub-tree of the LDAP server in which users who can access this station are found. At the very least it must contain the domain components of the server's domain, for example: DC=domain, CD=net
Attr Email	Email address The AD default value is: mail.	Identifies the specific attribute in the LDAP directory to store the user's LDAP email address. This value populates the Niagara user's Email property.
Attr Full Name	Text The AD default value is: name.	Identifies the specific attribute in the LDAP directory to store the user's full name. This value populates the Niagara user's Full Name property.

Property	Value	Description
Attr Language	Two-letter language code The AD default is blank.	Identifies the specific attribute in the LDAP directory to store the user's language. This value populates the Niagara user's Language property.
Attr Cell Phone Number	Telephone number The AD default is mobile.	Identifies the attribute in the LDAP directory that stores the user's mobile phone number. This value populates the Niagara user's Cell Phone Number property.
Attr Prototype	Text The AD default is memberOf.	<p>Identifies the User Prototype with which the system populates a new user's local properties.</p> <p>If this property is blank or the name does not match any user prototype, the system uses the Default Prototype to populate local user properties.</p> <p>If a user belongs to multiple user groups (user prototypes), the top-to-bottom order of prototypes determines which prototype the system uses. If the value of a user prototype property changes, the system dynamically updates user properties accordingly.</p>
Cache Expiration	Date and time	<p>Defines a future date after which the system no longer stores a user's password in cache. When an LDAP server is unavailable a user can still log on with the cached credentials until this date and time.</p> <p>This property applies to Kerberos authentication even though the station never receives the user's password. Instead, the station verifies the corresponding Kerberos user ticket against the cached user information.</p>
Domain	text	Supplies the domain name used to contact the LDAP server.

Ldap-LdapUserService

The **ldap** palette provides four LDAP user services.



Each user service replaces a station's standard **UserService** component and each provides essentially the same configuration properties.

The **ActiveDirectoryUserService** and **LdapUserService** user services support LDAPv2 only. While the **LdapV3UserService** and **LdapV3ADUserService** user services (and the authenticators) apply to systems using LDAPv3—with or without Kerberos authentication.

Use the table below to determine which LDAP user service to drag to your **Services** folder. Start by identifying your implementation in the first column and the version of LDAP your server supports in the second column. Use the third column to identify which service to install from the **ldap** palette. The Extension name column identifies the container name in the property sheet that contains the main LDAP properties you need to configure.

Implementation	Ldap version	LdapUserService	Extension name
Windows AD	LDAPv2	ActiveDirectoryUserService	ActiveDirectoryConfig
Open source (OpenLDAP or OpenDS)	LDAPv2	LdapUserService	LdapConfig
Open source (OpenLDAP or OpenDS)	LDAPv3	LdapV3UserService	LdapConfig
Windows AD	LDAPv3	LdapV3ADUserService	ActiveDirectoryConfig

The properties at the top of each LDAP user service property sheet apply to local users only. They are the same user properties found in the **UserService**.

Using multiple LDAP extensions

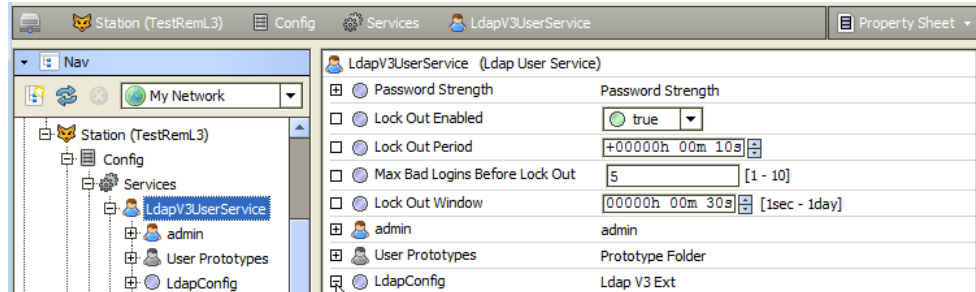
Update release, AX-3.8U1, added support for using multiple LDAP extensions. This is useful when it is necessary to log onto multiple Active Directory servers in order to validate users.

Simply drag multiple LDAP extensions from the **ldap** palette to the **LdapUserService** and then configure the extensions for different LDAP servers. Once configured, the feature functions as described here:

- A new LDAP user will try each extension until they can successfully log in.
- Once the user is logged in, they are assigned a dynamic property corresponding to the name of the extension he/she logged in with. On subsequent logins, only the assigned extension is used.
- If the assigned extension is renamed or removed, the user's property is automatically renamed or removed.

NOTE: Due to the differences in login dialogs, this feature does not work with Kerberos extensions.

Figure 5. LDAP user service properties that apply to local users only



Property	Value	Description
Password Strength	n/a	This is a container with five properties that define the minimum characters/types required in station user passwords. User additions or edits of any user's password require an entry that meets these minimum specifications, otherwise an error popup window results. See Password Strength, page 33
Lock Out Enabled	true or false	If enabled (<code>true</code>), a number of consecutive authentication failures temporarily disables logon access to the user account, for the duration of the lock out period (next property). Using lock out makes it difficult to automate the guessing of passwords. NOTE: Each user has a Clear Lock Out action.
Lock Out Period	hours minutes seconds (default is 10 seconds)	If lock out is enabled, this property defines the period of time a user account is locked out before being reset. While locked out, any logon attempt (even a valid one) is unsuccessful.

Property	Value	Description
		<p>NOTE: The default Lock Out value guards against an automated, brute-force password attack, where a computer application issues hundreds of logon attempts a second. The 10 second latency is thwarts such an attack, as the attacker must wait 10 seconds after each five unsuccessful logon attempts. If deemed necessary, you can adjust to guard against human attack.</p>
Max Bad Logins Before Lock Out	number from 1 — 10 (default is 5)	If lock out is enabled, in conjunction with the Lock Out Window , this property specifies the number of consecutive failed user logon attempts that trigger a lock out after a window of time.
Lock Out Window	hours minutes seconds (default is 30 seconds)	<p>If lock out is enabled, and a user fails to log on successfully before the Max Bad Logins Before Lock Out window (period) expires, the user is locked out for the Lock Out Period duration.</p> <p>The system enforces changes to lock out properties the next time the user logs in. For example, if Max Bad Logins Before Lock Out is set to 5, user ScottF fails to log on four times within the Lock Out Window, and an admin-level user changes Max Bad Logins Before Lock Out to 3, the change does not lock ScottF out. User ScottF still has one more chance to log on before getting locked out.</p> <p>If ScottF's fifth attempt to log on fails, the system locks him out the next time he attempts to log on because five failed attempts is greater than or equal to the Max Bad Logins Before Lock Out of 3.</p>

Property	Value	Description
admin		Provides a section for the admin user.
User Prototypes		See User Prototypes, page 33
LdapConfig		LdapV2 (LdapConfig) is the child "LdapConfig" container under the LdapUserService (LDAPv2 support only) and contains various properties that define the LDAP attribute and connection methods.

Password Strength

Property	Value	Description
Max Bad Logins Before Lock Out	number from 1 — 10 (default is 5)	If lock out is enabled, in conjunction with the Lock Out Window , this property specifies the number of consecutive failed user logon attempts that trigger a lock out after a window of time.
Minimum Length	10	Sets the required number of total characters.
Minimum Lower Case	1	Sets the required number of lower case characters (a-z).
Minimum Upper Case	1	Sets the required number of upper case (Z-A).
Minimum Digits	1	Sets the required number of digits (0–9).
Minimum Special	0	Sets the required number of non-alphabetic symbols.

User Prototypes

User Prototypes under an LDAP user service operate differently than they do under the standard **baja UserService**, where only the **Default Prototype** matters. However, just as with the standard **UserService**, the system always uses the LDAP user service's **Default Prototype** as the template for all user property values when creating any local user.

Under the **baja UserService** the system assigns any additional (non-default) prototypes to network users. Network users are not supported by the **LdapUserService**.

If, after an LDAP user accesses the station and the system creates a user account component (using whatever source property values), you subsequently change a property in the user prototype, this change dynamically updates in the corresponding user account. For example, if you increase permissions in the user prototype, the permissions for any user that the system originally sourced from the user prototype should also change to match.

Figure 6. User properties with permissions window

The screenshot shows a 'User properties with permissions' window. The main window has several sections: Name (ScottF), Full Name (Scott Free), Enabled (true), Expiration (Never Expires), Permissions (Super User), Network User (false), Prototype Name, Language, Password, Email (sfree@metropolis.net), Cell Phone Number (555-555-5555), Facets, and Nav File (file:^nav/Demo.nav). A 'Permissions' sub-window is open, showing a table of permissions for various categories and operators. A blue arrow points from the 'Permissions' field in the main window to the 'Permissions' sub-window.

Category	Operator			Admin		
	R	W	I	R	W	I
User	✓	✓	✓	✓	✓	✓
Admin	✓	✓	✓	✓	✓	✓
Category 3	✓	✓	✓	✓	✓	✓
Category 4	✓	✓	✓	✓	✓	✓
Category 5	✓	✓	✓	✓	✓	✓
Category 6	✓	✓	✓	✓	✓	✓
Category 7	✓	✓	✓	✓	✓	✓
Category 8	✓	✓	✓	✓	✓	✓

Property	Value	Description
Full Name	Text	The first and last name of the user.
Enabled [general]	true or false	Activates and deactivates use of the function.
Expiration	Radio buttons: Never Expires (default) Expires On: DD-MM-YYYY HH:MM AM/PM indicator time zone	Sets an optional date and time when user attributes are no longer valid.
Permissions	Super User check box	To the right of the check box the system summarizes the categories and their assigned permissions. Click the chevron to open the Permissions window
Language	Two-character code	Identifies the language spoken by the user.
Email	name@domain.com	Defines a user's email address.
Password	a minimum of 10 characters using: at least one UPPER CASE letter, at least one lower case letter, and at least one digit (numeral)	Two fields allow you to create and verify a strong password. The password must <i>match</i> in both password fields. The characters you enter display as asterisks (*).

Property	Value	Description
		An error popup reminds you if you attempt to enter a password that does not meet minimum rules.
Facets — Time format	reuse	Defines how the time displays: using AM and PM (12-hour clock; the default) or military time (24-hour clock).
Facets — Unit Conversion	Options: English (default) Metric None	Identifies which system to use.
Nav File	filename	Identifies a special XML file that resides on the file system—not in the station database. This file provides navigation in a hierarchical tree structure.
Prototype Name	text	The name to assign to this prototype.
Network User		
Email	name@domain.com	Defines a user's email address.
Network User		
Cell Phone Number	telephone number format	Identifies a user's smart phone.

Ldap-LdapV2

This component supports LDAPv2 client access to an open source LDAP server, such as OpenLDAP or OpenDS. Connection to the LDAP server uses simple authentication. The child **LdapConfig** component contains all properties needed to configure connection and authentication to the LDAP server.

Figure 7. LDAP user service

module.palette LdapUserService LdapConfig Property Sheet

LdapConfig (Ldap V2)

- ☒ Enable Connection Pooling true
- ☐ Connection Url ldap://domain.net
- ☐ SSL false
- ☐ User Login Attr uid
- ☐ User Base DC=domain, DC=net
- ☐ Attr Email
- ☐ Attr Full Name
- ☐ Attr Language
- ☐ Attr Cell Phone Number
- ☐ Attr Prototype
- ☐ Cache Expiration +00168h 00m 00s
- ☐ Connection User
- ☐ Connection Pwd

Refresh Save

Property	Value	Description
Enable Connection Pooling	true or false	Setting this property to true allows connections to be shared and reused. This can improve performance.
Connection URL	ldap://your.domain.net or ldap://your.domain.net:nnn	Identifies the URL (your.domain.net) for the LDAP server. Standard LDAP ports are 389, or 636 (if using SSL). If the server uses a non-standard port, include the port (your.domain.net:nnn) in the URL, for example, ldap://your.domain.net.999..
SSL	true or false	Enables and disables secure communication. If set to true, make sure that SSL (3.8) or TLS (4.0) is enabled in the station's FoxService (for Workbench-to-station access) and WebService (for browser-to-station access).
User Login Attr	Text For AD this value defaults to sAMAccountName.	Identifies the specific attribute in the LDAP directory to store the LDAP user logon name. For AD servers this is always sAMAccountName. For OpenLDAP servers, it would be uid.

Property	Value	Description
User Base	Domain components	Identifies the sub-tree of the LDAP server in which users who can access this station are found. At the very least it must contain the domain components of the server's domain, for example: DC=domain, CD=net
Attr Email	Email address The AD default value is: mail.	Identifies the specific attribute in the LDAP directory to store the user's LDAP email address. This value populates the Niagara user's Email property.
Attr Full Name	Text The AD default value is: name.	Identifies the specific attribute in the LDAP directory to store the user's full name. This value populates the Niagara user's Full Name property.
Attr Language	Two-letter language code The AD default is blank.	Identifies the specific attribute in the LDAP directory to store the user's language. This value populates the Niagara user's Language property.
Attr Cell Phone Number	Telephone number The AD default is mobile.	Identifies the attribute in the LDAP directory that stores the user's mobile phone number. This value populates the Niagara user's Cell Phone Number property.
Attr Prototype	Text The AD default is memberOf.	<p>Identifies the User Prototype with which the system populates a new user's local properties.</p> <p>If this property is blank or the name does not match any user prototype, the system uses the Default Prototype to populate local user properties.</p> <p>If a user belongs to multiple user groups (user prototypes), the top-to-bottom order of prototypes determines which prototype the system uses. If the value of a user prototype property changes, the system dynamically updates user properties accordingly.</p>

Property	Value	Description
Cache Expiration	Date and time	<p>Defines a future date after which the system no longer stores a user's password in cache. When an LDAP server is unavailable a user can still log on with the cached credentials until this date and time.</p> <p>This property applies to Kerberos authentication even though the station never receives the user's password. Instead, the station verifies the corresponding Kerberos user ticket against the cached user information.</p>
Connection User	text	<p>Defines the user name for the initial LDAP server connection. It may be required if users, who will be logging in, are in different sub-trees of the LDAP directory. If the LDAP server supports anonymous connections, leave this property empty (blank). When used, requires a valid user name in the LDAP server. The system uses this name to connect to the server for authentication.</p>
Connection Pwd	text	<p>The password for the user specified in property Connection User. When used, requires a valid password in the LDAP server. The system uses this password to connect to the server for authentication.</p>

Ldap-LdapV3Ext

Due to the popularity of Windows Server products with LDAPv3 capability, the **LdapV3ADUserService** may be the most frequently used of the Niagara **LdapUserService** components. Its sister service, **LdapV3UserService** provides the same LDAPv3 support for open system authentication schemes, such as OpenLDAP and OpenDS

Both user services support LDAPv3 client access to the LDAP server and are backwards compatible with LDAPv2 systems. The child **LdapConfig** component holds all properties needed to configure connection to the LDAP server, including an authenticator container.

By default, Kerberos authentication is used for both services. This authentication scheme requires a server that supports LDAPv3. If needed, after copying one of these user services to the **Services** container in the station, you can replace the default authenticator with the SimpleAuthentication from the **ldap** palette.

Figure 8. Example of LdapV3UserService properties

The screenshot shows a configuration window titled "module.palette" with tabs for "LdapV3ADUserService" and "ActiveDirectoryConfig". The "ActiveDirectoryConfig" tab is selected, showing a list of properties for "Ldap V3 Ext". The properties are as follows:

Property	Value
Enable Connection Pooling	true
Connection Url	ldap://domain.net
SSL	false
User Login Attr	sAMAccountName
User Base	DC=domain, DC=net
Attr Email	mail
Attr Full Name	name
Attr Language	
Attr Cell Phone Number	mobile
Attr Prototype	memberOf
Cache Expiration	+00168h 00m 00s
Bind Format	%userName%
authenticator	Kerberos Authenticator

At the bottom of the window are "Refresh" and "Save" buttons.

Property	Value	Description
Enable Connection Pooling	true or false	Setting this property to true allows connections to be shared and reused. This can improve performance.
Connection URL	ldap://your.domain.net or ldap://your.domain.net:nnn	Identifies the URL (your.domain.net) for the LDAP server. Standard LDAP ports are 389, or 636 (if using SSL). If the server uses a non-standard port, include the port (your.domain.net:nnn) in the URL, for example, ldap://your.domain.net.999..
SSL	true or false	Enables and disables secure communication. If set to true, make sure that SSL (3.8) or TLS (4.0) is enabled in the station's FoxService (for Workbench-to-station access) and WebService (for browser-to-station access).
User Login Attr	Text For AD this value defaults to sAMAccountName.	Identifies the specific attribute in the LDAP directory to store the LDAP user logon name. For AD servers this is always sAMAccountName. For OpenLDAP servers, it would be uid.

Property	Value	Description
User Base	Domain components	Identifies the sub-tree of the LDAP server in which users who can access this station are found. At the very least it must contain the domain components of the server's domain, for example: DC=domain, CD=net
Attr Email	Email address The AD default value is: mail.	Identifies the specific attribute in the LDAP directory to store the user's LDAP email address. This value populates the Niagara user's Email property.
Attr Full Name	Text The AD default value is: name.	Identifies the specific attribute in the LDAP directory to store the user's full name. This value populates the Niagara user's Full Name property.
Attr Language	Two-letter language code The AD default is blank.	Identifies the specific attribute in the LDAP directory to store the user's language. This value populates the Niagara user's Language property.
Attr Cell Phone Number	Telephone number The AD default is mobile.	Identifies the attribute in the LDAP directory that stores the user's mobile phone number. This value populates the Niagara user's Cell Phone Number property.
Attr Prototype	Text The AD default is memberOf.	Identifies the User Prototype with which the system populates a new user's local properties. If this property is blank or the name does not match any user prototype, the system uses the Default Prototype to populate local user properties. If a user belongs to multiple user groups (user prototypes), the top-to-bottom order of prototypes determines which prototype the system uses. If the value of a user prototype property changes, the system dynamically updates user properties accordingly.

Property	Value	Description
Cache Expiration	Date and time	<p>Defines a future date after which the system no longer stores a user's password in cache. When an LDAP server is unavailable a user can still log on with the cached credentials until this date and time.</p> <p>This property applies to Kerberos authentication even though the station never receives the user's password. Instead, the station verifies the corresponding Kerberos user ticket against the cached user information.</p>
Bind Format (3.8)	BFormat (Baja Format) syntax with a default value of %userName%	<p>This property applies to Ldap V3 only. Every LDAP server is different. For the most part, a user base and logon name are sufficient to find a user in the LDAP directory. However, when using DIGEST authentication, it may be necessary to specify the exact format of the logon name to send to the server. In Active Directory (AD) 2000, this might be: %username%@domain.com. Later versions of AD would reject this format, however, they would accept username based on how the server stores passwords. Bind Format allows you to specify how to send the name to the server, for example, using a BFormat this property would be: %username%@domain.net or cn=%username, %userBase%. For details, see the engineering notes document, <i>BFormat (Baja Format) Property Usage</i>.</p> <p>If you are not using Kerberos authentication, but instead are using SimpleAuthenticator, you need to specify the exact format of the logon name. This format depends on the LDAP server and may be required more often when the</p>

Property	Value	Description
		<p>Authentication Choice in the SimpleAuthenticator is set to DIGEST MD5. In some cases, just the user base and logon name may be sufficient to find a user in the LDAP directory.</p> <p>The default value may handle most cases, especially with an AD. However, if you are using the SimpleAuthenticator and choose DIGEST MD5 for authentication, this property may need to be changed. For example, in one AD 2000 implementation, a change was necessary to: username@domain.com.</p> <p>In another implementation (using an OpenLDAP server and SimpleAuthenticator with DIGEST MD5 encryption) required being set to: %userLoginAttr%=%userName%, %userBase%.</p> <hr/> <p>NOTE: For assistance, consult with your onsite LDAP administrator for assistance if the value of this property needs to be changed.</p> <hr/>
Connection User	text	<p>Defines the user name for the initial LDAP server connection. It may be required if users, who will be logging in, are in different sub-trees of the LDAP directory. If the LDAP server supports anonymous connections, leave this property empty (blank). When used, requires a valid user name in the LDAP server. The system uses this name to connect to the server for authentication.</p>
Connection Pwd	text	<p>The password for the user specified in property Connection User. When used, requires a valid password in the LDAP server. The system uses this</p>

Property	Value	Description
		password to connect to the server for authentication.
Authentication Mechanism	Simple, CRAM-MD5, and DIGEST-MD5	LDAP v3 supports SASL (Simple Authentication and Security Layer) mechanisms. Simple sends the user name and password to the server in clear text while CRAM-MD5 and DIGEST-MD5 obscures the password for security.
Domain	text	Supplies the domain name used to contact the LDAP server.

Idap-KerberosAuthenticator

This component provides Kerberos authentication for system users.

Kerberos is a computer network authentication protocol that works on the basis of so-called tickets to allow nodes communicating over a network that is not secure to prove their identity to one another in a secure manner. Aimed primarily at a client-server model, it provides mutual authentication—both the user and the server verify each other's identity. (Wikipedia)

Figure 9. KerberosAuthenticator properties

Property	Value	Description
Realm	UPPERCASE letters EXAMPLE.COM	Identifies the system on which the LDAP server resides. You get this information from your Kerberos administrator.
Realm Display Name	Optional text the default is blank	When using a web browser, this property specifies the text to display instead of Realm in the lower SSO area of the station login window. If you leave this property blank, the system uses the domain name as defined by

Property	Value	Description
		the Realm property on the button. For example: TRIDI-UM.NET Login .
Key Distribution Center	Text Example: kd.example.com	Specifies the name of the Kerberos Key Distribution Center that the system contacts to get a ticket, which, like a key, is used to authenticate the user to the NiagaraAX system. You get this information from your Kerberos administrator.
Station Kerberos Name	Text	<p>As part of securely delegating Kerberos tickets, this property represents the station as a user in the Kerberos database. If logging in only via Workbench, this user can be any user or service in the Kerberos directory.</p> <p>However, if the user logs in via a browser, the user must be a service in the form: HTTP/service-Name.domain.com, where service-Name.domain.com is how the station is to be accessed in the browser, (for example, http://jacekerb1.mydomain.com).</p> <p>The service name for the station Kerberos name typically omits a bit of the normal http URL syntax, for example: http://jacekerb1.mydomain.net instead of http://jacekerb1.mydomain.net. You may need to ask the Kerberos administrator to create the service for you in the Kerberos database.</p> <hr/> <p>NOTE: Kerberos is very particular about names. You must enter the station name in the “Station Kerberos Name” property exactly as it appears in the Kerberos database. Upper/lowercase can sometimes be an issue, so make sure you have an exact match.</p> <hr/>

Property	Value	Description
Station Kerberos Password	Text default: blank	Specifies the password for the Kerberos station user identified by the Station Kerberos Name property. If you are using a keytab file, you can leave this property blank.
KeyTab Location or Key Tab File	File name	<p>Kerberos services usually do not use a password to authenticate. Instead, they use a file that contains a key table (keytab file). To authenticate from a web browser you must specify an associated service in the Station Kerberos Name property and reference a keytab file supplied by the Kerberos administrator.</p> <p>Copy that keytab file to a secure location on the NiagaraAX platform, somewhere under the station's file space. For example, copy it into the root of the station's file space. For the KeyTab Location property, use the File Ord Chooser to browse to the keytab file and select it. Again, if you are using a keytab, you can leave the Station Kerberos Password property blank (default).</p>

Idap-PrototypeFolder

This component provides a folder you can use to group LDAP user service components in a station.

Idap-SimpleAuthenticator

This component provides simple authentication for system users.

SASL (Simple Authentication and Security Layer) is a framework for authentication and data security in Internet protocols. It de-couples authentication mechanisms from application protocols, in theory allowing any authentication mechanism supported by SASL to be used in any application protocol that uses SASL. Authentication mechanisms that support SASL can provide a data security layer offering data integrity and data confidentiality services. DIGEST-MD5 is an example of a mechanism that can provide a data-security layer. Application protocols that support SASL typically also support Transport Layer Security (TLS) to complement the services offered by SASL. (Wikipedia)

Figure 10. SimpleAuthenticator properties

Property	Value	Description
Connection User	Text	Specifies the name of the user for the initial connection. This name may be required if the LDAP users logging in are in different sub-trees of the LDAP directory. If the LDAP server supports anonymous connections, this property may not be required and can be left blank.
Connection Password	Text	Specifies the Connection User's password.
Authentication Mechanism	Simple (default) DIGEST-MD5 CRAM-MD5 None	<p>Simple: the password is passed as clear text during authentication.</p> <p>DIGEST-M5: the password is encrypted using Digest-MD5.</p> <p>CRAM-MD5: the password is encrypted using CRAM-MD5.</p> <p>None: no authentication is used.</p> <hr/> <p>CAUTION: You are strongly encouraged to use SSL (3.8) or TLS (4.0) with simple authentication. Sending credentials in clear text is not secure. Using a security protocol at least secures the connection.</p>

Plugins (views)

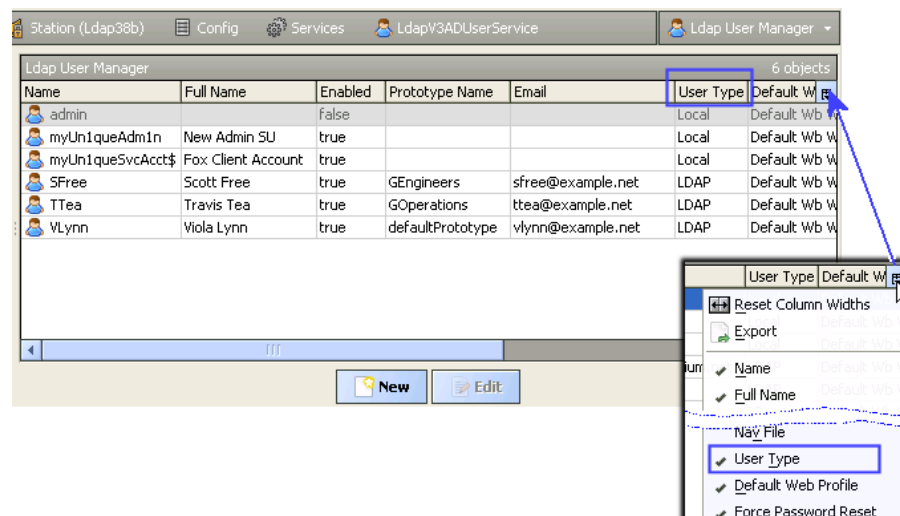
Plugins provide views of components and can be accessed in many ways. For example, double-click a component in the Nav tree to see its default view. In addition, you can right-click on a component and select from its **Views** menu.

For summary documentation on any view, select **Help > On View (F1)** from the menu or press **F1** while the view is open.

Ldap-LdapUserManager

The **Ldap User Manager** is the default view on any of the LdapV2 or LdapV3 user service components (installed from the **ldap** module). This view manages LDAP users as well as local station users. For example, you can add new local users and edit or delete existing local users. In regard to local users, the **Ldap User Manager** works identically to the standard (baja) **User Manager**.

Figure 11. Ldap User Manager view



Column	Value	Description
Full Name	Text	The first and last name of the user.
Enabled [general]	true or false	Activates and deactivates use of the function.
Expiration	Radio buttons: Never Expires (default) Expires On: DD-MM-YYYY HH:MM AM/PM indicator time zone	Sets an optional date and time when user attributes are no longer valid.
Lock Out Enabled	true or false	If enabled (true), a number of consecutive authentication failures temporarily disables logon access to the user account, for the duration of the lock out period (next property). Using lock out makes it difficult to

Column	Value	Description
		<p>automate the guessing of passwords.</p> <hr/> <p>NOTE: Each user has a Clear Lock Out action.</p> <hr/>
Permissions	Super User check box	To the right of the check box the system summarizes the categories and their assigned permissions. Click the chevron to open the Permissions window
Network User		
Prototype Name	text	The name of the prototype to assign to this user.
Language	Two-character code	Identifies the language spoken by the user.
Password	a minimum of 10 characters using: at least one UPPER CASE letter, at least one lower case letter, and at least one digit (numeral)	<p>Two fields allow you to create and verify a strong password.</p> <p>The password must <i>match</i> in both password fields. The characters you enter display as asterisks (*).</p> <p>An error popup reminds you if you attempt to enter a password that does not meet minimum rules.</p>
Email	name@domain.com	Defines a user's email address.
Facets	<i>trueText</i> (default) or <i>falseText</i>	<p>Facets contain additional data applied to input and output values.</p> <ul style="list-style-type: none"> <i>trueText</i> is the text to display when output is true <i>falseText</i> is the text to display when output is false. <p>For example, you might want to set the facet <i>trueText</i> to display "ON" and the facet <i>falseText</i> to display "OFF."</p> <p>"Units of measurement" is also a type of facet. You can view Facets on the Slot sheet and edit them from a component Property sheet by clicking the >> icon</p>

Column	Value	Description
		to display the Config Facets window.
Nav File	filename	Identifies a special XML file that resides on the file system—not in the station database. This file provides navigation in a hierarchical tree structure.
User Type		

GLOSSARY

LDAP user	A user whose access permissions are managed by an LDAP (Lightweight Directory Access Protocol) server.
local user	A user that has been set up using the standard NiagaraAX user properties. The system cannot validate a local user against the LDAP database. This type of user can only log in to a local host. Two local users are common: the admin user and a service user. The admin user logs in to the host during initial setup and rarely thereafter to update NiagaraAX software. A service user represents the host to other remote users.
realm	A set of NiagaraAX stations that share the same Kerberos database. The Kerberos database resides on the LDAP server.
service user	A user created within a station for the sole purpose of representing the station as the client of another remote host. No one ever logs in to host as the service user. It represents the host to other remote hosts on the network when configuring the remote host's Client Connection properties under the NiagaraAXStation device.
super user	A type of local user that has full read-write permissions within a station. This user can set up and update station software as well as create users and manager access permissions. The default admin user is an example of a super user.

INDEX

A

about this guide.....	iii
ActiveDirectoryUserService	27
Authentication Scheme property	5

B

browser	
login.....	23
browser configuration	13
browserlogin.....	20, 22

C

checklist.....	1
Chrome	
configuring.....	15
Components	27
configuration	1

D

DNS configuration.....	12
document change log	iii

F

FAQ	2
Firefox	
configuring.....	13

G

Google Chrome	
configuring.....	15

I

Internet Explorer	
configuring.....	14

K

Kerberos	
configuring the client PC.....	10
krb5.conf	10
krb5.ini.....	10
login.....	21
setting up authentication.....	9
tickets.....	12
Key Distribution Center	12

L

LDAP	
attributes	19
benefits.....	18
configuring the component.....	24
definition	17
LDAP implementations	17
Ldap User Manager	47
LDAP user service	
adding to a station.....	3
LDAP users	18
ldap-ActiveDirectoryUserService	27
ldap-KerberosAuthenticator.....	43
ldap-LdapUserService	30
ldap-LdapV2.....	35
ldap-PrototypeFolder	45
ldap-SimpleAuthenticator.....	45
LdapV3ADUserService.....	38
LdapV3UserService.....	38
local service user	
configuring.....	7
local user	
prototype	6
local users	18
login	
without found Kerberos	
authentication.....	21

P

PC setup for Kerberos	10
plugins.....	47
prerequisites	1

R

reference	27
-----------------	----

S

setup	1
simple authentication	
setting up	9
single sign on	20, 22
SSO	20, 22

T

tickets used in Kerberos authentication	12
--	----

U

user groups.....	20
user prototype	
prototypes for user properties	6
user prototypes	20
users	
LDAP	18
userslocal.....	18

V

views	47
-------------	----