TRIDIUM niagara

Niagara 4 Hardening Guide

Tips to Secure a Niagara 4 System

NIAGARA 4 HARDENING GUIDE

TABLE OF CONTENTS

INTRODUCTION

This document describes how to implement security best practices in a Niagara 4 system. While it is impossible to make any system completely impenetrable, there are many ways to build up a system that is more resilient to attacks. In particular, this document describes how you can help make a Niagara 4 system more secure by carefully configuring and using:

- Passwords
- System Passphrase
- Platform Account Management
- Station Account Management
- Role and Permission Management
- Authentication
- TLS and Certificate Management
- Module Installation
- Additional Settings
- External Factors

Please note that while all of these steps should be taken to protect your Niagara 4 system, they do not constitute a magic formula. Many factors affect security – and vulnerabilities in one area can affect security in another; it doesn't mean much to configure a system expertly if your JACE is left physically unsecured where anyone can access it.

PASSWORDS

The Niagara 4 system typically uses passwords to authenticate "users" to a station or platform. It is particularly important to handle passwords correctly. If an attacker acquires a user's password, they can gain access to the system and have the same permissions as that user. In the worst case, an attacker might gain access to a Super User account or platform account and the entire system could be compromised.

Here are some of the steps that you can take to help secure the passwords in a Niagara 4 system:

- Use the Password Strength Feature
- Enable the Account Lockout Feature
- Expire Passwords
- Use the Password History
- Use the Password Reset Feature
- Leave the "Remember These Credentials Box Unchecked

USE THE PASSWORD STRENGTH FEATURE

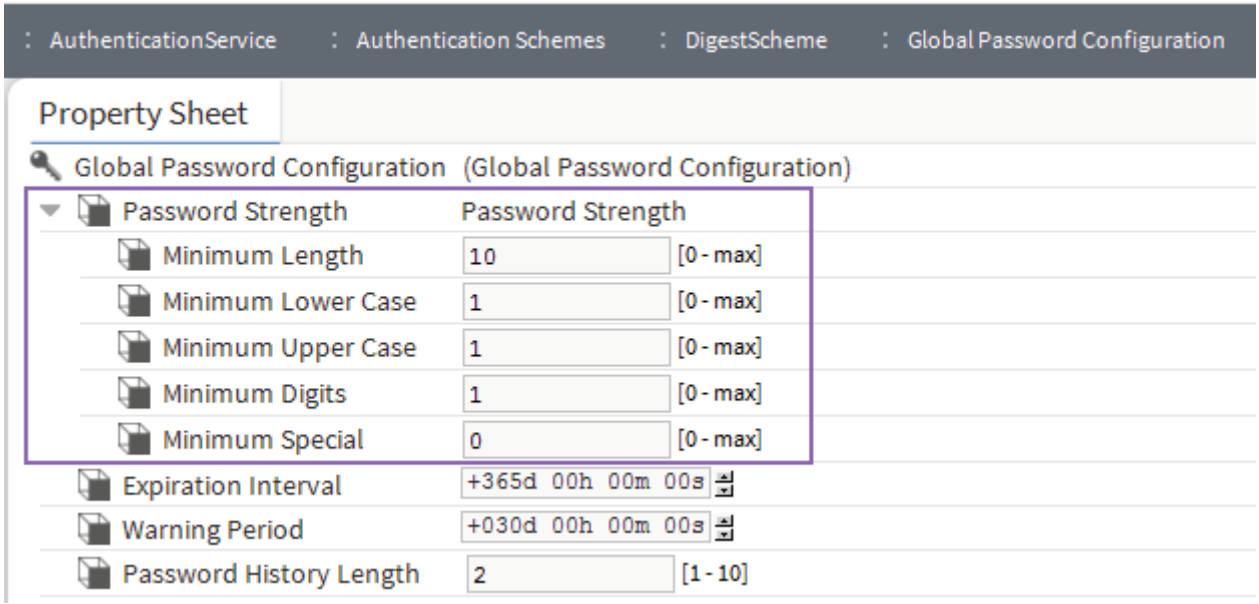
Many of the configurable authentication schemes in Niagara 4 support the notion of authenticating users with a password, but not all passwords are equally effective. Ensuring that users are choosing good, strong passwords is essential to securing a Niagara 4 system that uses password-based authentication schemes.

In Niagara 4, password strength is enforced by the “Password Strength” property on the authentication scheme “Global Password Configuration” property and the required password strength can be customized to meet the needs of each particular system. By default, passwords are required to be at least 10 characters in length, and contain at least 1 digit, 1 uppercase and 1 lowercase character. At the time of the writing of this document, this is the recommended industry standard for most applications. However, systems with higher security requirements can configure the “Password Strength” property to require a password strength that meets their needs.

Note that while password strength can be increased, it shouldn’t be reduced.

To change the required password strength, follow the steps described below.

1. **Go to the station’s AuthenticationService property sheet (Station > Config > Services > AuthenticationService).**
2. **Expand the “Authentication Schemes” folder and then expand the authentication scheme that you want to change.**
3. **Go to the “Global Password Configuration” property, expand the “Password Strength” property, and edit the fields as appropriate.**



4. Save the changes.

Note: This does not force a user whose password no longer meets the password strength requirement to change their passwords. If that user changes their password after the password strength requirements are modified, their new password will have to meet the new requirements.

STRONGER PASSWORDS

Even with good password strength requirements, there are some passwords that are stronger than others. It is important to educate users on password strength. Password strength requirements are not sufficient to ensure that actually strong passwords are used. For example, “Password10” satisfies all the requirements, but is actually a

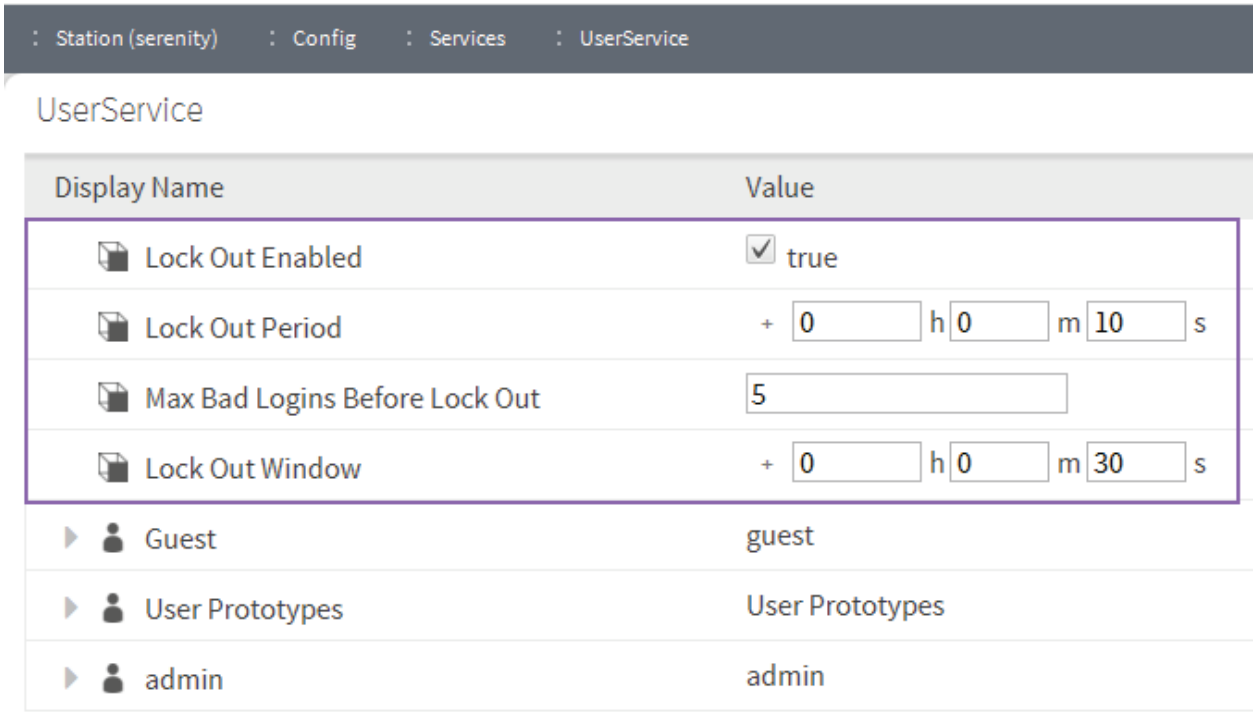
weak, easily hackable password. When creating a password follow the guidelines in Appendix A: Creating Strong Passwords That Are Actually Strong to help you generate stronger passwords.

ENABLE THE ACCOUNT LOCKOUT FEATURE

The user lockout feature allows the UserService to lock out a user after a specified number of failed login attempts. That user is not able to log back in to the station until the lockout is removed. This helps protect the Niagara 4 system against attackers trying to guess or "brute force" users' passwords.

Account Lock Out is enabled by default, but if it is not currently enabled, you can enable it as described below:

1. **Go to the station's UserService property sheet.**
2. **Set the "Lock Out Enabled" property to true.**



3. **Adjust the other lockout properties as necessary.**

   - **Lock Out Period.** This determines how long the user is locked out for. Even short periods (for example, 10 seconds) can be quite effective at blocking "brute force" attacks without inconveniencing users. However, more sensitive systems may warrant a longer lockout period.
   - **Max Bad Logins Before Lock Out.** This determines how many login failures are required before locking out the user.
   - **Lock Out Window.** The user is only locked out if the specified number of login failures occurs within the time set in the Lock Out Window. This helps separate suspicious activity (for example, 10 login failures in a few seconds) from normal usage (for example, 10 login failures over a year).

4. **Save the changes.**

EXPIRE PASSWORDS

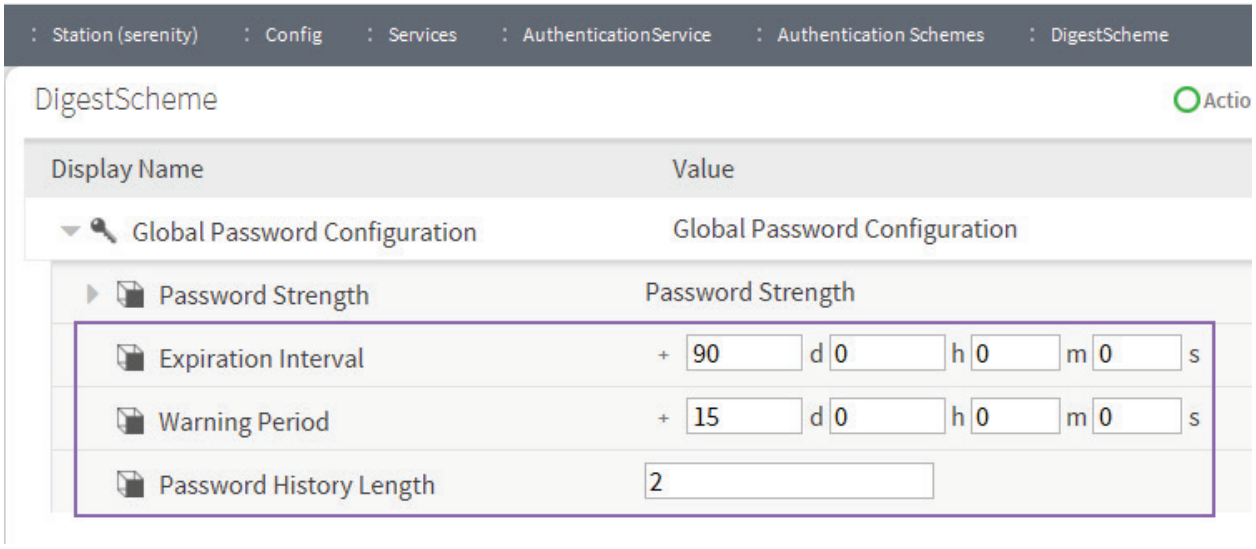
In Niagara 4, user passwords can be set to expire after a specified amount of time, or on a set date. This ensures that old passwords are not kept around indefinitely. If an attacker acquires a password, it is only useful to them until the password is changed. Expiration settings are configured on authentication schemes' Global Password Configuration property sheets as well as on individual user properties.

PASSWORD EXPIRATION: PASSWORD CONFIGURATION PROPERTY SHEET

Configure general password expiration settings in the UserService property sheet, as described below:

1. **Go to the station's AuthenticationService property sheet.**
2. **Go to the "Authentication Schemes" folder, and find the authentication scheme for which you want to modify the password expiration.**
3. **Expand the "Global Password Configuration" property, and configure the expiration settings as necessary.**

    - **Expiration Interval.** This property setting determines how long a password is used before it needs to be changed. The default is 365 days. You should change this to a lower value; ninety days is standard for many situations. NOTE: You must also set individual user password expiration dates (See Password Expiration: Edit Users Dialog Box).
    - **Warning Period.** Users are notified when their password is about to expire. The Warning Period specifies how far in advance the user is notified. Fifteen days generally gives the user enough time to change their password.



4. **Save the changes.**

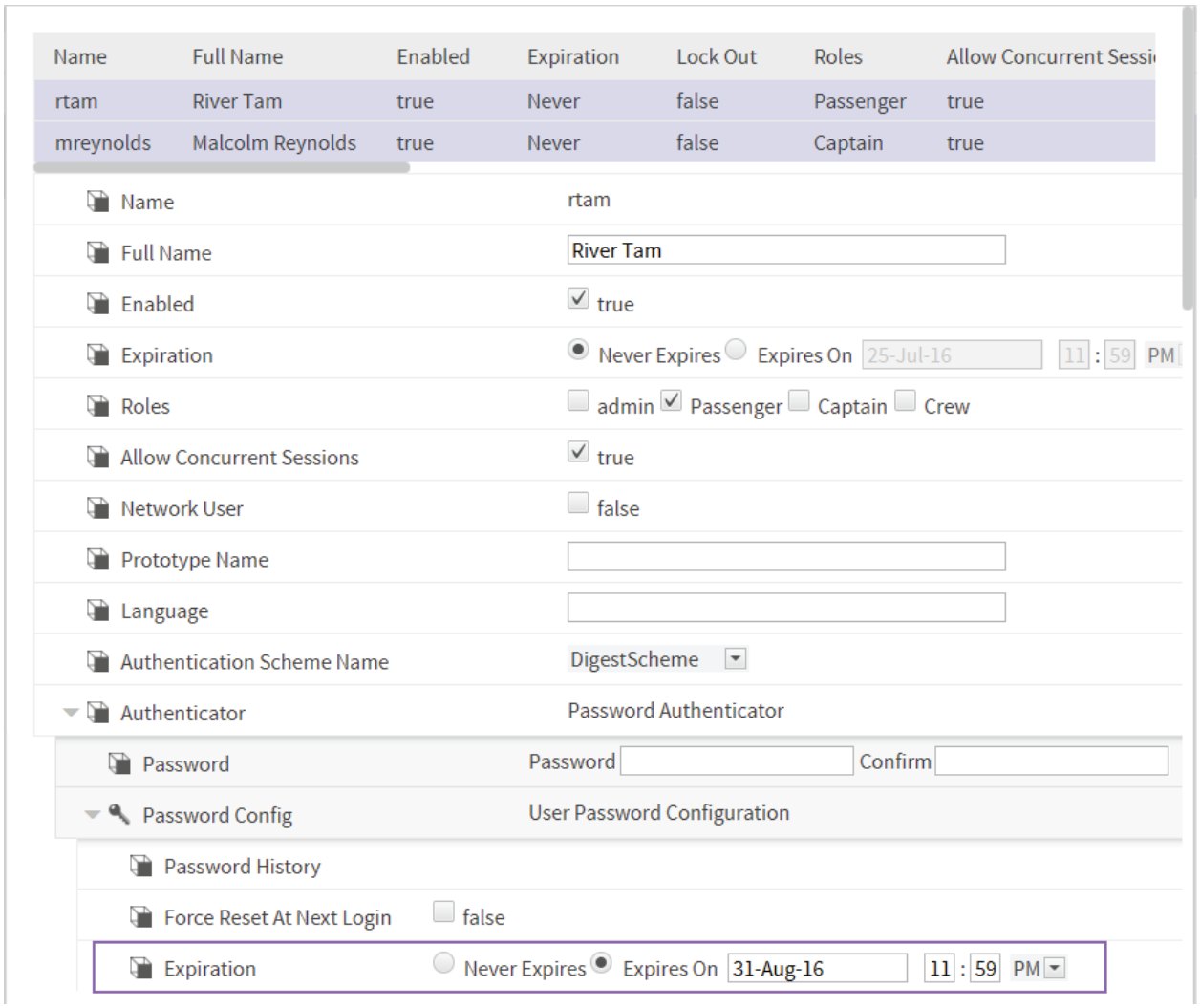
PASSWORD EXPIRATION: EDIT USERS DIALOG BOX

Password expiration may also be enabled on each user. If enabled on a user, the setting on the user takes precedence over the authentication scheme password expiration configuration. Once the password expires, the configuration on the user's authentication scheme is applied.

This property is available, by user, from the UserService property sheet but it may be more conveniently configured from the User Manager view, as described below:

To enable user password expiration, do the following:

1. **Select the User Manager view on the UserService (Station > Config > Services > UserService).**
2. **In the User Manager view, select one or more users and click the Edit button to open the Edit dialog box.**



3. **Choose "Expires On" for the Password Expiration option and set the expiration date at least 15 days into the future or perhaps equal to what you set for the "Password Configuration" Warning Period property.**

NOTE:

- *The default user "Password Expiration" property value is "Never Expires". To create new users with expiring passwords enabled, set the Password Configuration "Expiration" property (UserService > User Prototypes > Default Prototype > Password Configuration) to "Expires On" under the "Default Prototype" but be sure to actually set the "Expires On" date for each user.*
- *You could set the "Expires On" date to an arbitrary date far enough into the future that the user will likely have logged into the system before expiring and also set the "Force Reset At Next Login" to true so the user is forced to change their password on first login. This would then get their expiration in sync.*

4. **Save the changes. The next time the user changes their password, the expiration date is automatically updated to the UserService "Expiration Interval" added to current date and time.**

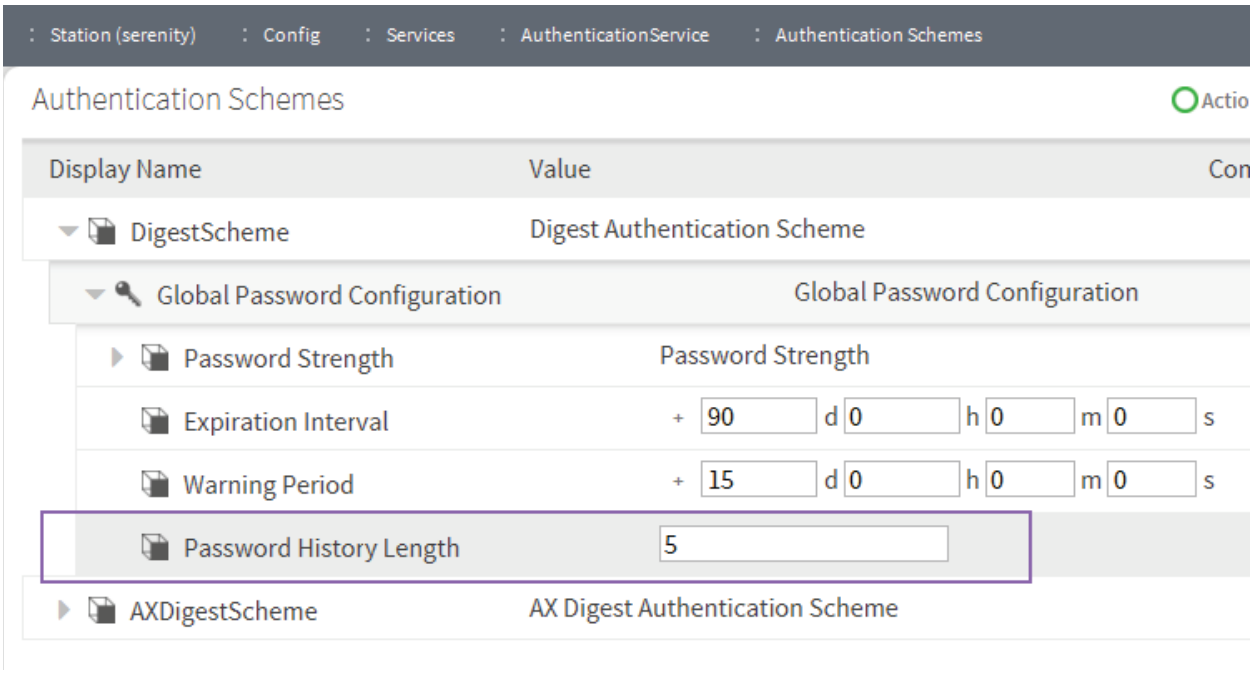
USE THE PASSWORD HISTORY

In Niagara 4, authentication schemes can be configured to remember users' previously used passwords. This password history is used to ensure that when a user changes his password, he or she does not choose a previously used password. Much like the password expiration feature, the password history helps prevent users from using passwords indefinitely. The default setting of "2" should always be changed to a reasonable number for your system.

Note: Password histories are tied to authentication schemes. Therefore, users with more sensitive accounts can have stronger authentication schemes with longer password histories.

To configure the password history, do the following:

- **Go to the AuthenticationService property sheet.**
- **Go to the "Authentication Schemes" folder, and find the Authentication Scheme whose password history you wish to modify.**
- **Expand the "Global Password Configuration" property.**



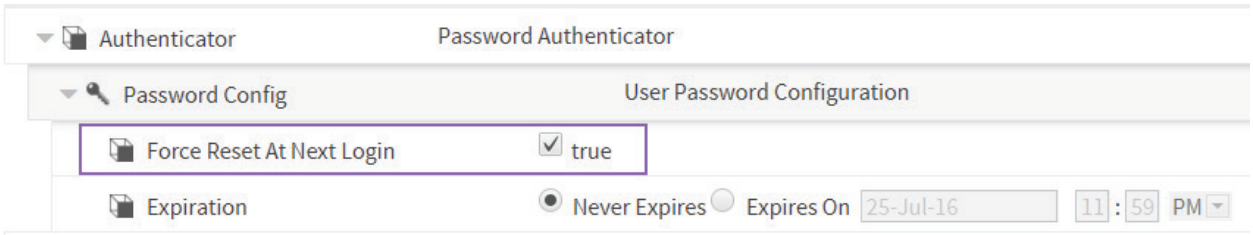
- **Set the "Password History Length" property to a non-zero value. This determines how many passwords are remembered. The maximum password history length is 10.**

USE THE PASSWORD RESET FEATURE

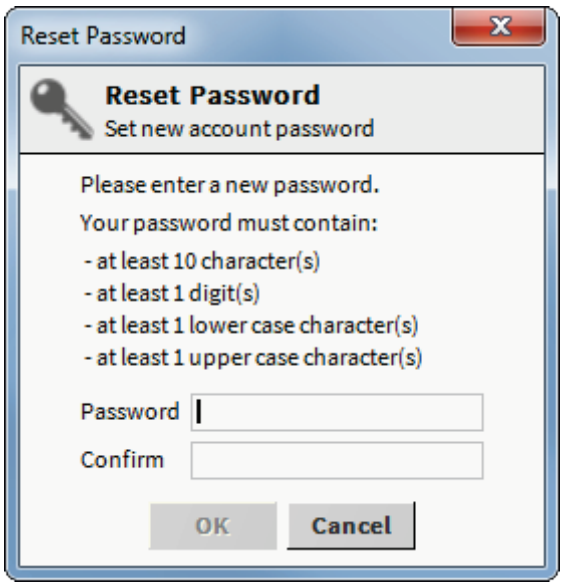
In Niagara 4, you can force users to reset their password. This is particularly useful when creating a new user. The first time a user logs in, he or she can create a brand new password known only to that user. The password reset feature is also useful to ensure that a new password policy is enforced for all users. For example, if a station is changed to require strong passwords, the existing passwords may not conform to the password policy. Forcing users to reset their passwords will ensure that after logging in to the station, their password conforms to the rules.

The following steps describe how to force a user to reset their password:

1. **Go to the user's property sheet view.**
2. **Expand the "Password Configuration" property.**
3. **Set the "Force Reset At Next Login" property to "True."**



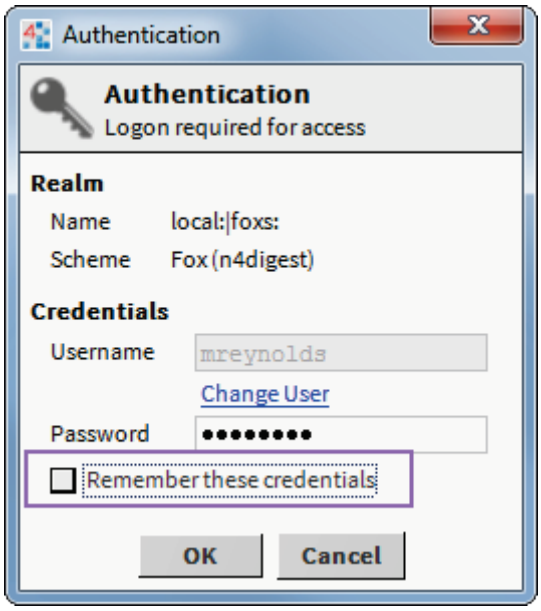
4. **The next time the user logs in they will be prompted to reset their password, as shown below. The user cannot access the station until resetting the password.**



To create new users with the "Force Reset At Next Login" property automatically set to "True," verify that the "Force Reset At Next Login" property is set to "True" on the "Default Prototype."

LEAVE THE "REMEMBER THESE CREDENTIALS" BOX UNCHECKED

When logging in to a Niagara 4 system via workbench, the login dialog includes a checkbox to "Remember these credentials." When checked, workbench will remember the credentials and use them to automatically fill in the login dialog box the next time the user tries to log in.



This option is provided for convenience. However, it is important to be aware that, if the box is checked, anyone with access to that workbench is able to log in using those credentials. For highly sensitive systems, privileged accounts, or unsecure computers, you should always leave the box unchecked.

NOTE: In Niagara 4, there is the "Allow User Credential Caching" property on the "General" tab in the Workbench Options dialog box (Tools > Options) which defaults to true. If you set that property to false, it will prevent a user from being able to even select the "Remember these credentials" check box in the login dialog.

SYSTEM PASSPHRASE

Niagara 4 uses a system passphrase to help protect the various sensitive data in a Niagara 4 system. This can include user passwords, Kerberos keytab files, backups, etc... In order to protect them, the data are encrypted using the system passphrase. The system passphrase is not associated with a user; it is used by the system to encrypt files. Because the passphrase is known by a human user, the data can be moved to another unit and decrypted there, provided the new system is provided with the correct system passphrase.

Because it is used to protect sensitive data, the system passphrase is also considered sensitive and should be protected. This section describes the various steps to take to keep your system passphrase safe.

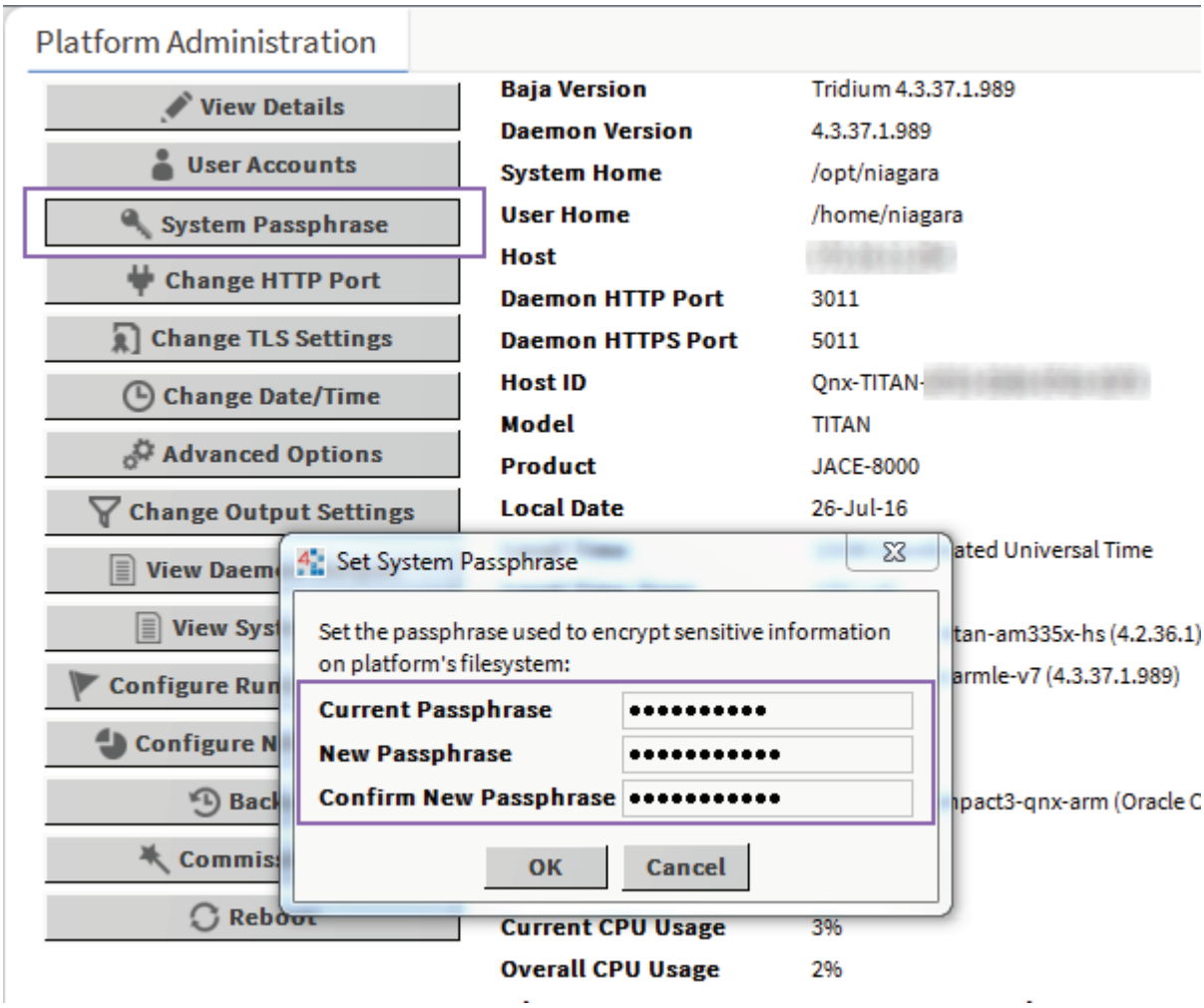
- Change the Default System Passphrase
- Use TLS To Set the System Passphrase
- Choose a Strong System Passphrase
- Protect the System Passphrase
- Ensure Platform Owner Knows the System Passphrase

CHANGE THE DEFAULT SYSTEM PASSPHRASE

Each JACE is shipped with a default system passphrase, “niagara.” When commissioning a new JACE, you should always change the system passphrase from the default to some new, unique passphrase. Default values are typically well known, and leaving the system passphrase at the default value leaves your sensitive data open to attack.

To change the system passphrase, follow the steps below:

1. **Open a platform connection and go to the “Platform Administration” view.**
2. **Click on “System Passphrase.”**



3. **Enter the old system passphrase. Enter the new system passphrase and confirm. The system passphrase must contain at least 10 characters, 1 digit, 1 lower case character and 1 upper case character.**

Note: You can easily tell if you're still using the default passphrase by going to the "Platform Administration" view. If you are using the default passphrase, a yellow warning box will be displayed in the bottom right indicating the problem.

USE TLS TO SET THE SYSTEM PASSPHRASE

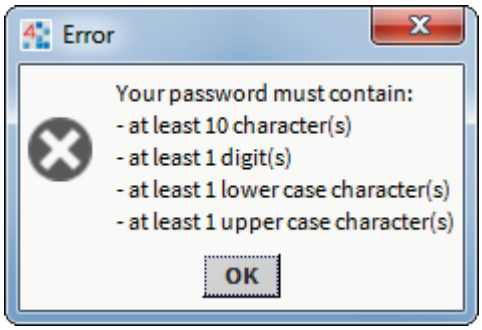
The system passphrase protects sensitive data; it must be protected. One way an attacker can attempt to acquire the system passphrase is by sniffing network traffic: although the password is sent across in encrypted format, it is sent in a clear text wrapper indicating that this is a password reset message.

Using TLS adds additional protection by encrypting the whole communication - an attacker wouldn't be able to tell which message is a password reset.

CHOOSE A STRONG SYSTEM PASSPHRASE

The system passphrase is used to protect important data. As a result, a strong passphrase should be selected. The system enforces the following passphrase requirements (see below):

- At least 10 characters long



- At least 1 digit
- At least 1 lower case character
- At least 1 upper case character

It is important to note that passphrase strength requirements are not sufficient to ensure that actually strong passphrases are used. See Appendix A: Creating Strong Passwords That Are Actually Strong for guidelines on creating strong passwords.

PROTECT THE SYSTEM PASSPHRASE

In addition to picking a strong system passphrase, users should take care to protect the system passphrase. The passphrase should not be written down or placed on a sticky note on the JACE. If forgetting the passphrase is truly a concern, it should be recorded in a proper key management system, or written down and locked away in a truly secure location (e.g. a safe).

ENSURE PLATFORM OWNER KNOWS THE SYSTEM PASSPHRASE

When installing a Niagara 4 system, it's not uncommon for the installer to be a different person than the owner or user of the platform. For example, many people hire system integrators to set up their Niagara 4 system. In these situations, it is important that once the system integrator is done, they provide the system owner with the system passphrase. The system owner should then change the system passphrase to something known only to them. This has several advantages:

- If something happens and a JACE can no longer be restored, a backup of the system can be restored to another device, but only if the system password is known. If the original system integrator cannot be brought back in, and the system owner doesn't know the password, their backups cannot be restored to a new JACE.

- The data protected by the system passphrase belongs to the system owner, and ideally should be protected by something only they know. This improves confidentiality of their data.

PLATFORM ACCOUNT MANAGEMENT

Platform accounts are highly sensitive accounts that can allow a user to modify or bring down the system. These platform accounts must be protected in order to maintain the confidentiality, integrity and availability of your Niagara 4 system.

This section describes steps that can be taken to secure your platform accounts:

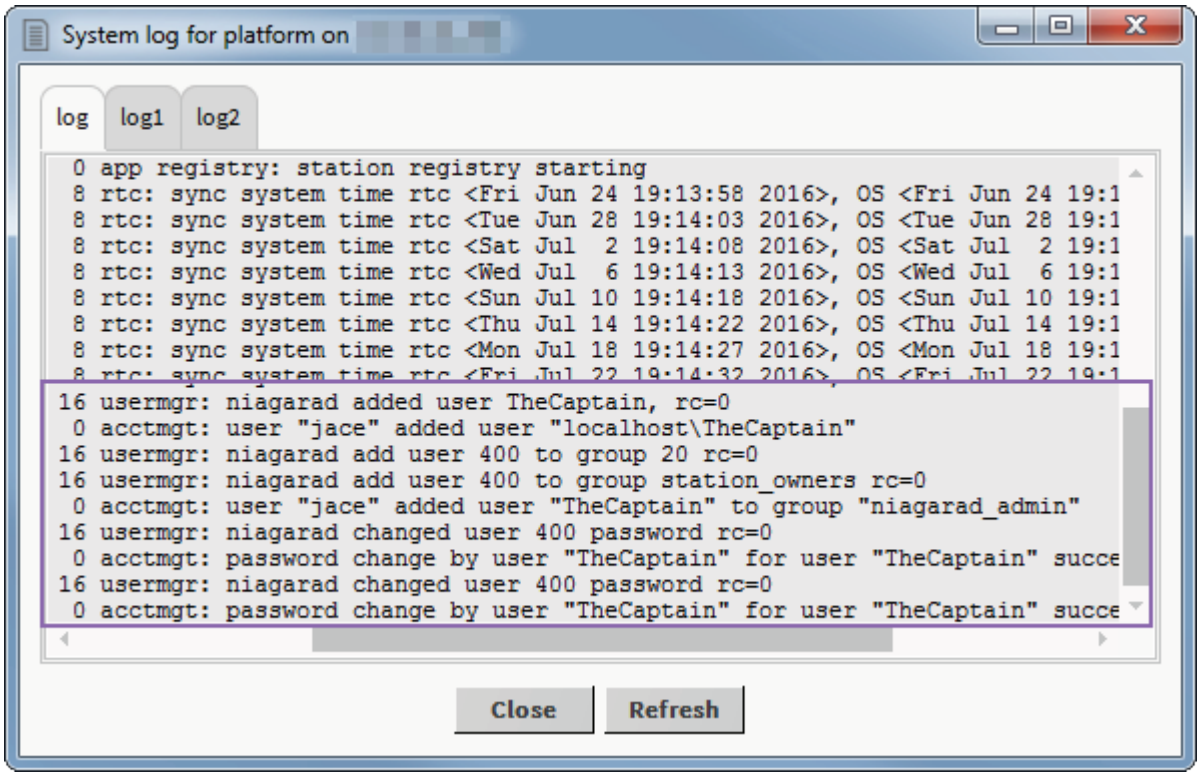
- Use a Different Account for Each Platform User
- Use Unique Account Names for Each Project
- Ensure Platform Owner Knows the Platform Credentials

USE A DIFFERENT ACCOUNT FOR EACH PLATFORM USER

In a Niagara 4 system, multiple platform users can be created for a JACE. Each platform user account should represent a single user. Different people should never share the same account. For example, rather than a general "PlatformAdmin" user that many administrators could use, each administrator should have their own, separate account.

There are many reasons for each platform user to have their own individual account:

- If each user has their own account, audit logs will be more informative. It will be easy to determine exactly which user did what. This can help detect if an account has been compromised. In the example below, it is easy to determine which changes were made by the user "jace", and which were made by the user "TheCaptain."



Note: Not all platform audit entries include the user who performed the action, but it's still a good idea to have a separate account for each user.

- If an account is removed, it does not inconvenience many users. For example, if a user should no longer have access to a station, deleting their individual account is simple. If it is a shared account, the only options are to change the password and notify all users, or to delete the account and notify all users. Leaving the account as-is is not an option – the goal is to revoke the user's access.

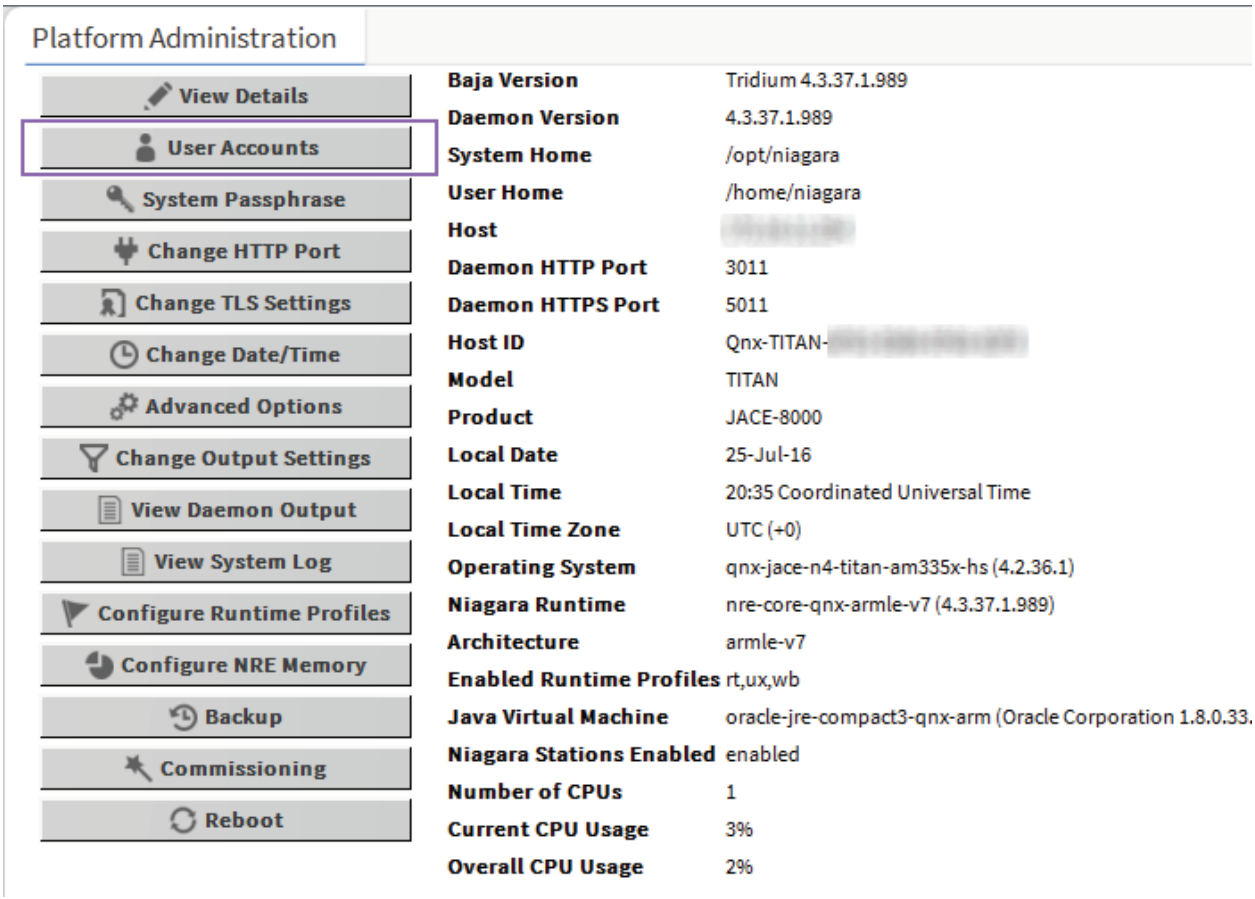
- A shared account means a shared password. It is an extremely bad security practice to share passwords. It makes it much more likely for the password to be leaked, and makes it more difficult to implement certain password best practices, such as password expiration.

Each different user should have a unique individual account.

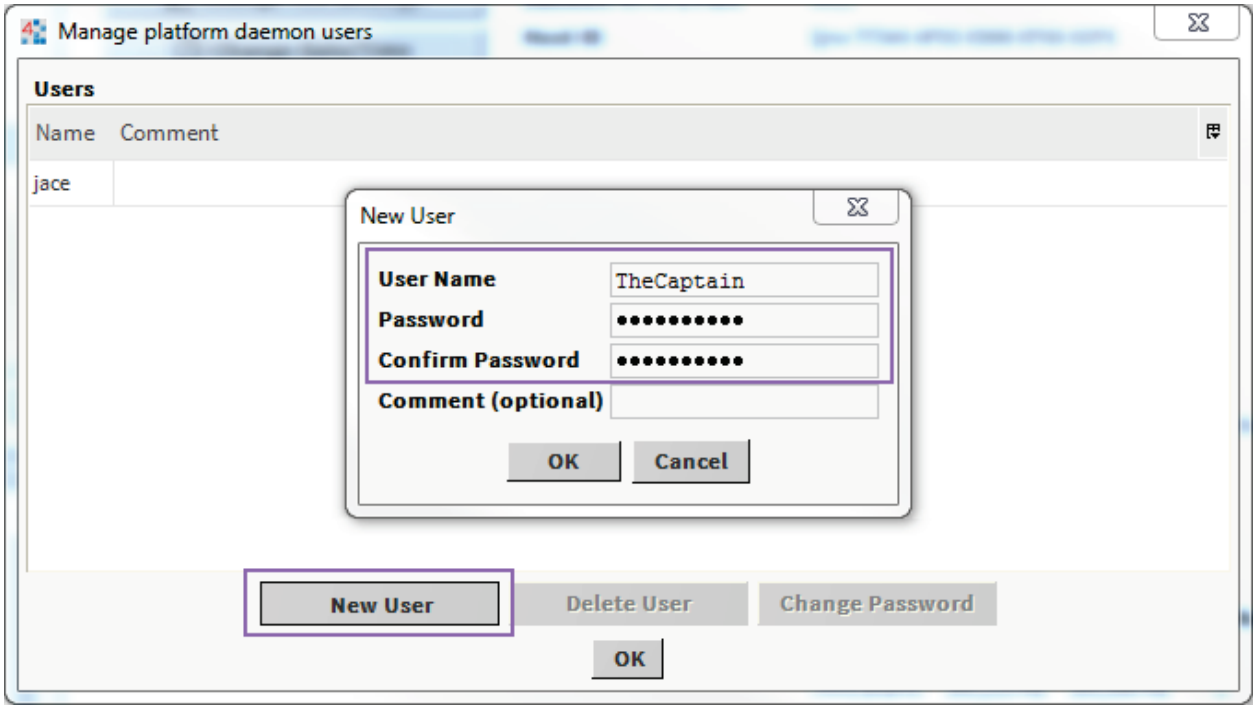
Note: Platform accounts are highly sensitive accounts. Malicious access to the platform can completely compromise the confidentiality, integrity, and availability of the platform. Therefore, you should only have a few authorized platform users, each of which should have their own unique account.

To create a new platform account, follow the steps described below:

1. **Open a platform connection to a JACE, and click on "User Accounts".**



2. **Select "New User". In the dialog that pops up, enter the new user's username and password. You can optionally provide a comment that will be shown in clear text in the platform user management dialog.**



3. **Click "OK".**

USE UNIQUE ACCOUNT NAMES FOR EACH PROJECT

It is a common (bad) practice that some system integrators often use the exact same platform credentials on every project they install. If one system is compromised, the attacker could potentially have credentials for access to many other projects installed by the same contractor.

ENSURE PLATFORM OWNER KNOWS THE PLATFORM CREDENTIALS

When installing a Niagara 4 system, it's not uncommon for the installer to be a different person than the owner or user of the platform. For example, many people hire system integrators to set up their Niagara 4 system. In these situations, it is important that once the system integrator is done, they provide the system owner with the platform credentials. The system owner should then change the platform credentials to something known only to them. This has several advantages:

- If a platform connection is required (e.g. for an update), but the original system integrator cannot be brought back in, the system owner can still perform the update, either themselves or using a new system integrator.
- The Niagara system and its data typically belong to the system owner, and ideally should be protected by something only they know. This improves confidentiality of their data.

STATION ACCOUNT MANAGEMENT

A Niagara 4 station has accounts, represented by users in the UserService. It is important that these accounts are properly managed. Failure to do so can make it easier for an attacker to penetrate the system, or make it more difficult to detect that an attack has occurred.

Some steps to help correctly manage user accounts are listed below.

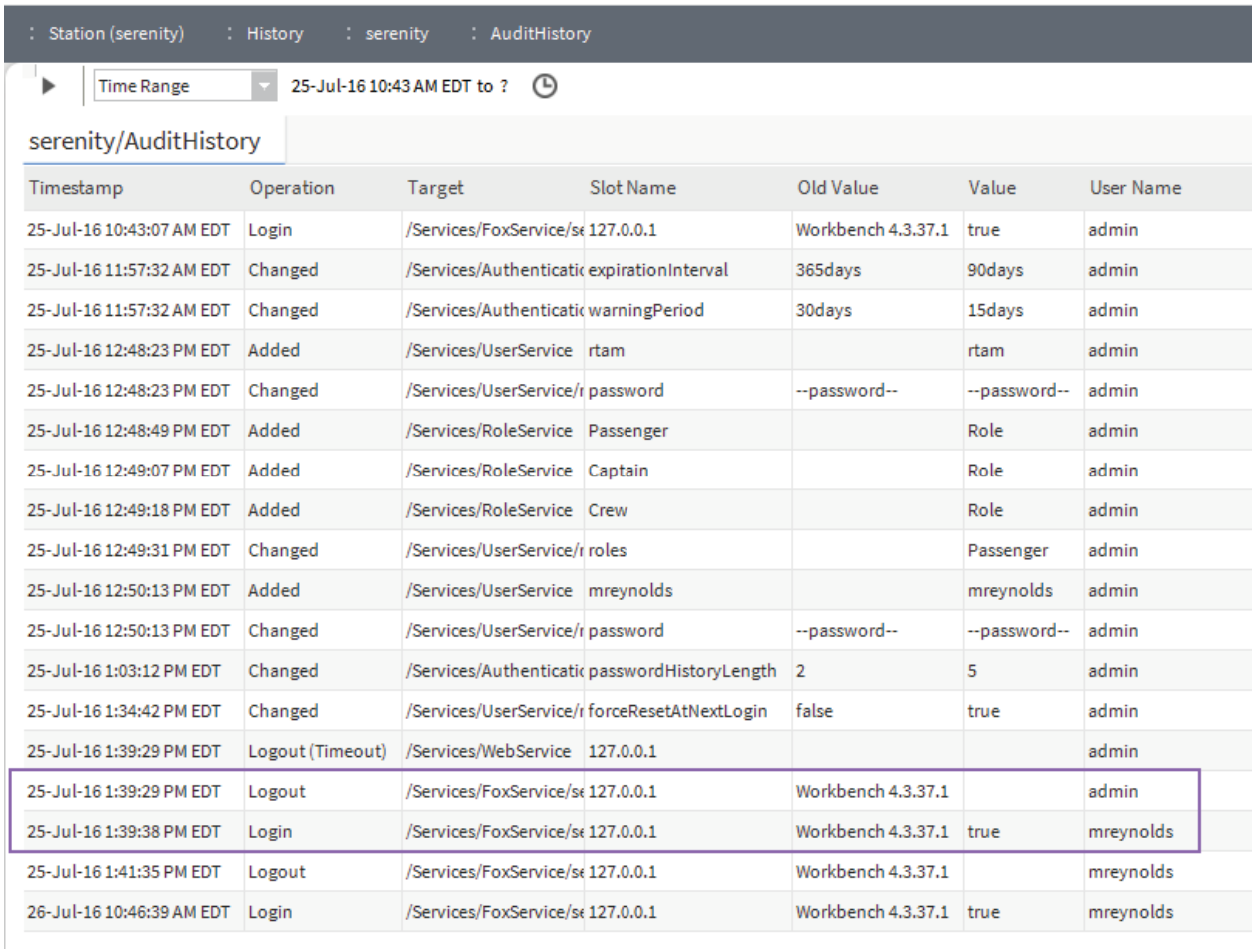
- Use a Different Account for Each Station User
- Use Unique Service Type Accounts for Each Project
- Disable Known Accounts When Possible
- Set Up Temporary Accounts to Expire Automatically
- Change System Type Account Credentials
- Disallow Concurrent Sessions When Appropriate

USE A DIFFERENT ACCOUNT FOR EACH STATION USER

Each user account in the UserService should represent a single user. Different people should never share the same account. For example, rather than a general “managers” user that many managers could use, each manager should have their own, separate account.

There are many reasons for each user to have their own individual account:

- If each user has their own account, audit logs will be more informative. It will be easy to determine exactly which user did what. This can help detect if an account has been compromised. In the example below, it is easy to determine which changes were made by the user “admin”, and which were made by the user “mreynolds.”


: Station (serenity) : History : serenity : AuditHistory

Time Range 25-Jul-16 10:43 AM EDT to ?

serenity/AuditHistory

Timestamp	Operation	Target	Slot Name	Old Value	Value	User Name
25-Jul-16 10:43:07 AM EDT	Login	/Services/FoxService/s	127.0.0.1	Workbench 4.3.37.1	true	admin
25-Jul-16 11:57:32 AM EDT	Changed	/Services/Authenticatic	expirationInterval	365days	90days	admin
25-Jul-16 11:57:32 AM EDT	Changed	/Services/Authenticatic	warningPeriod	30days	15days	admin
25-Jul-16 12:48:23 PM EDT	Added	/Services/UserService	rtam		rtam	admin
25-Jul-16 12:48:23 PM EDT	Changed	/Services/UserService/r	password	--password--	--password--	admin
25-Jul-16 12:48:49 PM EDT	Added	/Services/RoleService	Passenger		Role	admin
25-Jul-16 12:49:07 PM EDT	Added	/Services/RoleService	Captain		Role	admin
25-Jul-16 12:49:18 PM EDT	Added	/Services/RoleService	Crew		Role	admin
25-Jul-16 12:49:31 PM EDT	Changed	/Services/UserService/r	roles		Passenger	admin
25-Jul-16 12:50:13 PM EDT	Added	/Services/UserService	mreynolds		mreynolds	admin
25-Jul-16 12:50:13 PM EDT	Changed	/Services/UserService/r	password	--password--	--password--	admin
25-Jul-16 1:03:12 PM EDT	Changed	/Services/Authenticatic	passwordHistoryLength	2	5	admin
25-Jul-16 1:34:42 PM EDT	Changed	/Services/UserService/r	forceResetAtNextLogin	false	true	admin
25-Jul-16 1:39:29 PM EDT	Logout (Timeout)	/Services/WebService	127.0.0.1			admin
25-Jul-16 1:39:29 PM EDT	Logout	/Services/FoxService/s	127.0.0.1	Workbench 4.3.37.1		admin
25-Jul-16 1:39:38 PM EDT	Login	/Services/FoxService/s	127.0.0.1	Workbench 4.3.37.1	true	mreynolds
25-Jul-16 1:41:35 PM EDT	Logout	/Services/FoxService/s	127.0.0.1	Workbench 4.3.37.1		mreynolds
26-Jul-16 10:46:39 AM EDT	Login	/Services/FoxService/s	127.0.0.1	Workbench 4.3.37.1	true	mreynolds

- If an account is removed, it does not inconvenience many users. For example, if a user should no longer have access to a station, deleting their individual account is simple. If it is a shared account, the only options are to change the password and notify all users, or to delete the account and notify all users. Leaving the account as-is is not an option – the goal is to revoke the user's access.
- If each user has their own account, it is much easier to tailor permissions to precisely meet their needs. A shared account could result in users having more permissions than they should.
- A shared account means a shared password. It is an extremely bad security practice to share passwords. It makes it much more likely for the password to be leaked, and makes it more difficult to implement certain password best practices, such as password expiration.

Each different user should have a unique individual account. Similarly, users should never use accounts intended for station-to-station connections. Station-to-station connections should have their own accounts.

USE UNIQUE SERVICE TYPE ACCOUNTS FOR EACH PROJECT

It is a common (bad) practice that some system integrators often use the exact same system (station to station) credentials on every project they install. If one system is compromised, the attacker could potentially have credentials for access to many other projects installed by the same contractor.

DISABLE KNOWN ACCOUNTS WHEN POSSIBLE

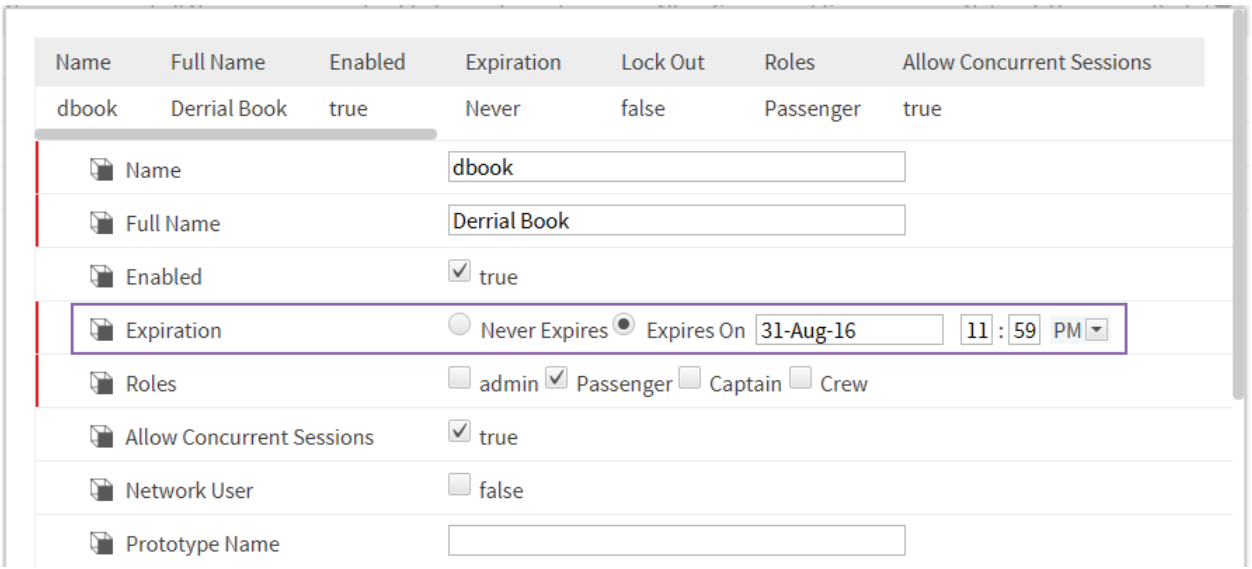
In Niagara 4, it is possible to disable the default admin account. The admin account is a known account name in a Niagara 4 system. If the admin or any other known account name is enabled a potential hacker need only guess the user's password. Note that you will not be able to disable the admin user account until you have created another super user account.

SET UP TEMPORARY ACCOUNTS TO EXPIRE AUTOMATICALLY

In some cases, you may need to set up an account for a user who only temporarily needs access. For example, an auditor may need an account to inspect the system. In these situations, a new account should be created and set up to expire automatically when it is no longer needed, using the "Expiration" property. This ensures that no accounts are accidentally left enabled.

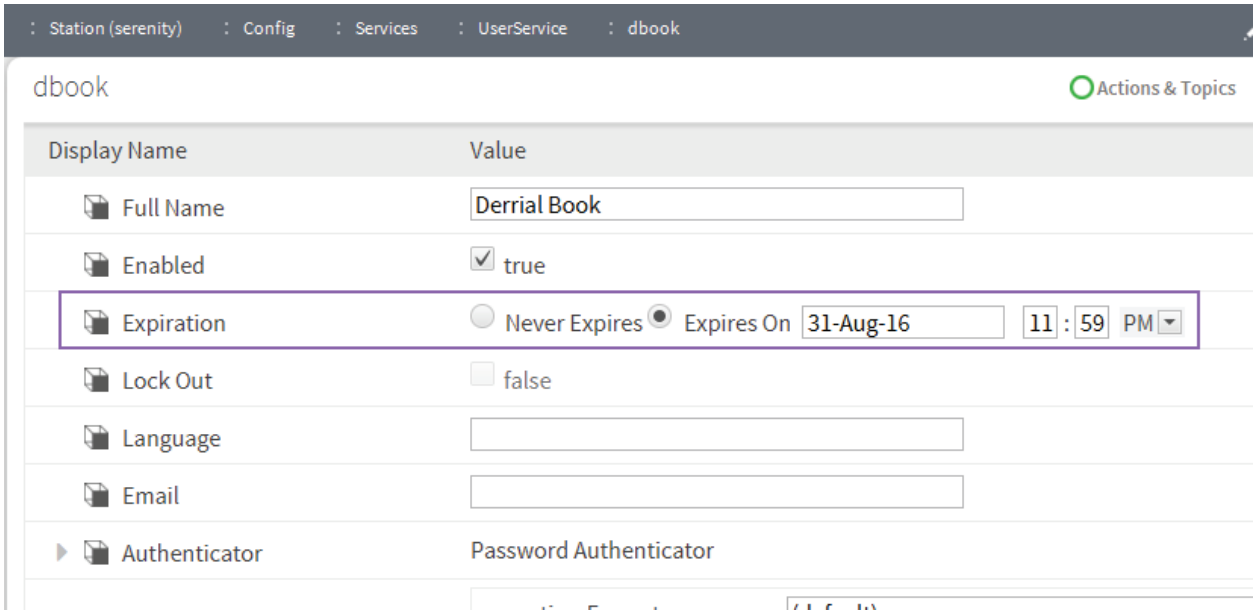
To set up an account to expire, follow the instructions below.

1. **Go the UserService, and create a new User.**
2. **In the user creation pop up dialog, set the "Expiration" property to the date the user will no longer require access.**



Alternatively, if the user is already created, you can follow the following steps.

1. **Go to the user's property sheet in the UserService.**
2. **Edit the "Expiration" property to be the date the user will no longer require access.**



CHANGE SYSTEM TYPE ACCOUNT CREDENTIALS

It may be necessary to periodically change the system type account credentials (station to station, station to rdbms, etc). For example, if an employee who is knowledgeable of the system type credentials is terminated, you may want to change those credentials. Also, in most cases, it is better to configure a system type account with non-expiring passwords, so that those passwords expiring silently do not affect system operation.

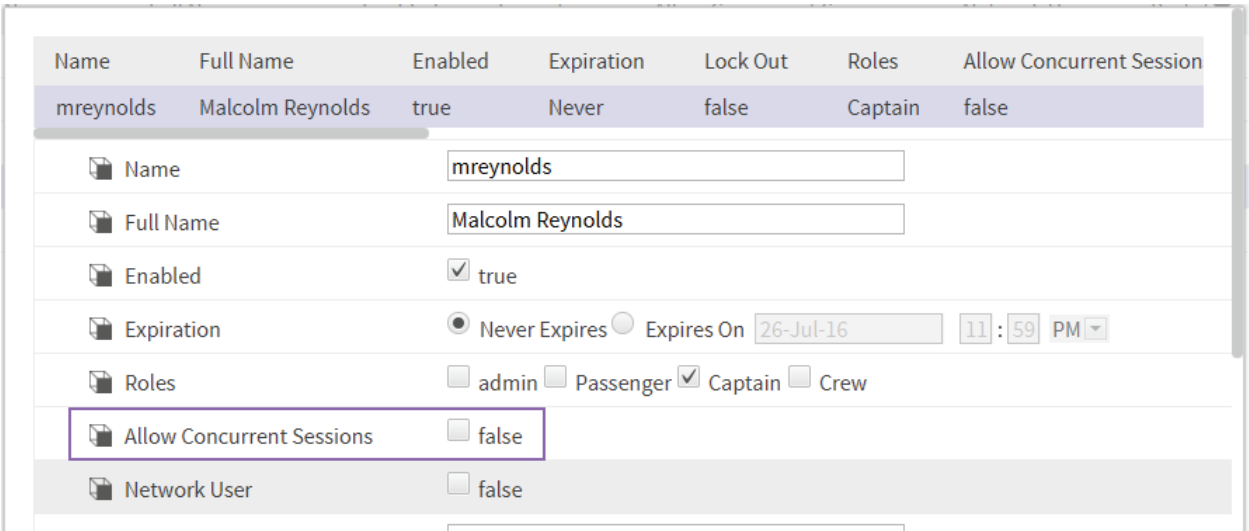
DISALLOW CONCURRENT SESSIONS WHEN APPROPRIATE

In Niagara 4, users can, by default, log in from multiple clients at the same time. For example, a user could be logged from two different workstations, or from two different browsers on the same workstation. However, certain accounts may be more sensitive and may require extra protection. If you know that a user will only ever be logged in from one client at a time, you can disable the ability to run concurrent sessions. If a user is logged in, and the same user logs in from a different workstation, the original session will be disconnected with a message informing the user why. This has several advantages:

- It helps prevent sessions being left open unattended. If a user goes home and forgets to end their session at the office, it will automatically be terminated if they log in from home.
- It notifies the user of suspicious activity. If a user's session is disconnected unexpectedly, this can indicate that an unauthorized person has accessed their account. The user can quickly change their password, or alert the system administrator to disable their account.

To disallow concurrent sessions, follow the steps below.

1. **In the UserManager view, double-click on the user for which you wish to disallow concurrent sessions.**
2. **In the popup dialog, set the "Allow Concurrent Sessions" property to "false."**



ROLE & PERMISSION MANAGEMENT

In Niagara 4, user permissions are managed by roles and the RoleService. Permissions are assigned to roles, and roles can be assigned to one or more users. It is important to manage roles and permissions properly. Failure to do so can result in users having more permissions than they need, which can result in accidental or malicious security breaches.

Some steps to help properly manage roles and permissions are listed below.

- Configure Roles with Minimum Required Permissions
- Assign Minimum Required Roles to Users
- Use the Minimum Possible Number of Super Users
- Require Super User Permissions for Program Objects
- Use the Minimum Required Permissions for External Accounts

CONFIGURE ROLES WITH MINIMUM REQUIRED PERMISSIONS

When creating a new role, think about what the users who will be assigned that role needs to do in the station, and then assign the minimum permissions required to do that job. For example, a user who only needs to acknowledge alarms does not need access to the UserService or the WebService. Giving non-required permissions increases the chance of a security breach. The user might inadvertently (or purposefully) change settings that they should not change. Worse, if the account is hacked, more permissions give the attacker more power.

CREATE NEW CATEGORIES

In the Category Service, you should create categories as needed to ensure that users have access only to what they absolutely need.

For more information on setting categories and permissions, refer to the “Authorization Management” section and various subsections in the *Station Security Guide*.

ASSIGN MINIMUM REQUIRED ROLES TO USERS

Users can be assigned one or more roles. This allows you to create roles corresponding to discrete tasks (e.g. AlarmManager or LightTechnician). Users should only be assigned the roles that they need to complete their required tasks. As does assigning too many permissions to a role, assigning too many roles to a user increases the chance of a security breach.

USE THE MINIMUM POSSIBLE NUMBER OF SUPER USERS

Only assign a Super User role when absolutely necessary. A Super User is an extremely powerful account – it allows complete access to everything. A compromised Super User account can be disastrous. Only the system administrator should have access a Super User account.

It is a good practice for system administrators to have two accounts. One account for normal use, and the other for use in emergency situations.

Although it can be very tempting to take the easy route and create a single Super User role and assign it to each user, doing so puts your system at risk. Instead, create a set of roles that allow you to easily assign the permissions your users require.

REQUIRE SUPER USER PERMISSIONS FOR PROGRAM OBJECTS

Program Objects are special components in a Niagara 4 station that have certain special permissions granted to them (in particular the ability to run external executables).

While Program Objects are restricted to Super Users by default, it is possible to lift this restriction by editing the <niagara_home>\lib\system.properties file. To ensure that the restriction is in place, verify that the line "niagara.program.requireSuperUser=false" is commented out (using the # character) as shown below:

```
# When this line is set to false, the restriction that only
# super users can add/edit program objects and robots in a
# running station will be lifted.  The default value is true,
# meaning that only super users can add/edit program objects (and robots).
#niagara.program.requireSuperUser=false
```

NOTE: Although only Super Users should be allowed to edit Program Objects, it can be acceptable for other users to invoke the Program Object's "Execute" action.

USE THE MINIMUM REQUIRED PERMISSIONS FOR EXTERNAL ACCOUNTS

Some stations use accounts for external servers – for example, an RdbmsNetwork with a SqlServerDatabase must specify a username and password for the SQL server. This account is used when connecting to the server to read from or write to the database.

NOTE: References in this section are to permissions on the external server, and not permissions on the Niagara 4 station.

These and any other external accounts should always have the minimum permissions needed for the required functionality. That way, if the station is compromised or an exploit is discovered, the external server is better protected: an attacker gaining control of an SQL administrator user could wreak havoc, reading confidential

information or deleting important data; on the other hand, an attacker gaining control of a restricted user has much less power.

When configuring a Niagara 4 station, be sure to understand exactly what tasks the external account needs to be able to perform, and create a user with the minimum rights and permissions required to perform those tasks.

AUTHENTICATION

Niagara 4 stations have a pluggable authentication system that can support many different authentication schemes at once. These schemes determine how a client talks to the station and how the user's credentials are transmitted to the station for proof of identity. Be sure to use the strongest authentication policies to increase protection for user credentials, keeping those accounts safer from attacks.

The following steps help secure the authentication system.

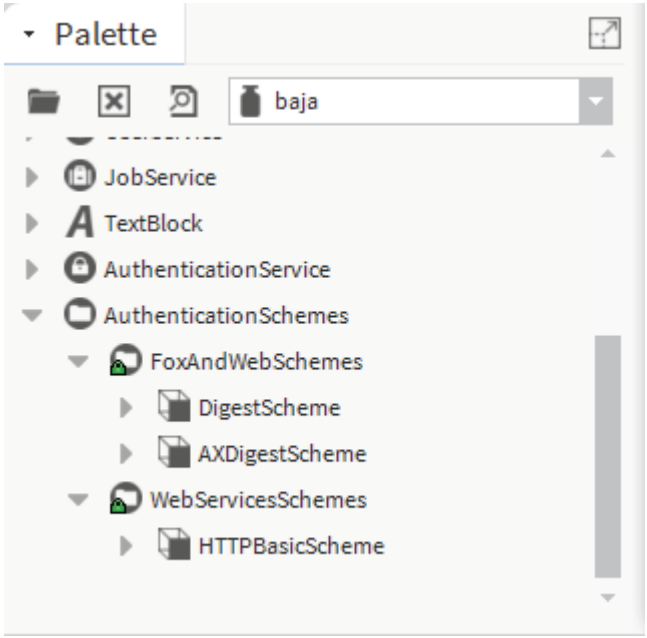
- Use an Authentication Scheme Appropriate for the Account Type
- Remove Unnecessary Authentication Schemes

USE AN AUTHENTICATION SCHEME APPROPRIATE FOR THE ACCOUNT TYPE

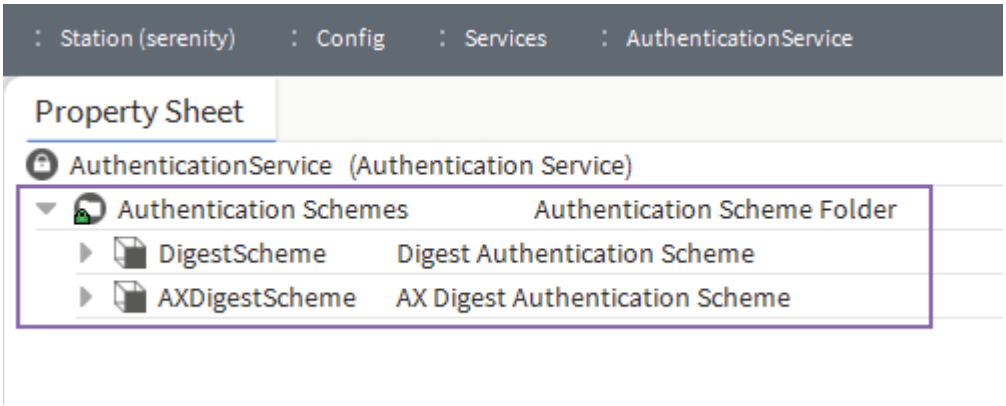
In Niagara 4, the type of authentication used is determined by the user account. Sensitive accounts should use stronger authentication types. Accounts for simple devices that can't do anything else can use less robust authentication schemes, but should have roles with as few permissions as possible.

Authentication schemes can be added to the station via the Authentication Service, as shown below.

1. **Go to the station's AuthenticationService, and go to the "AuthenticationSchemes" folder.**
2. **Open the palette for the module which contains your authentication scheme. Niagara 4 comes with authentication schemes built in the 'baja' and 'ldap' modules.**



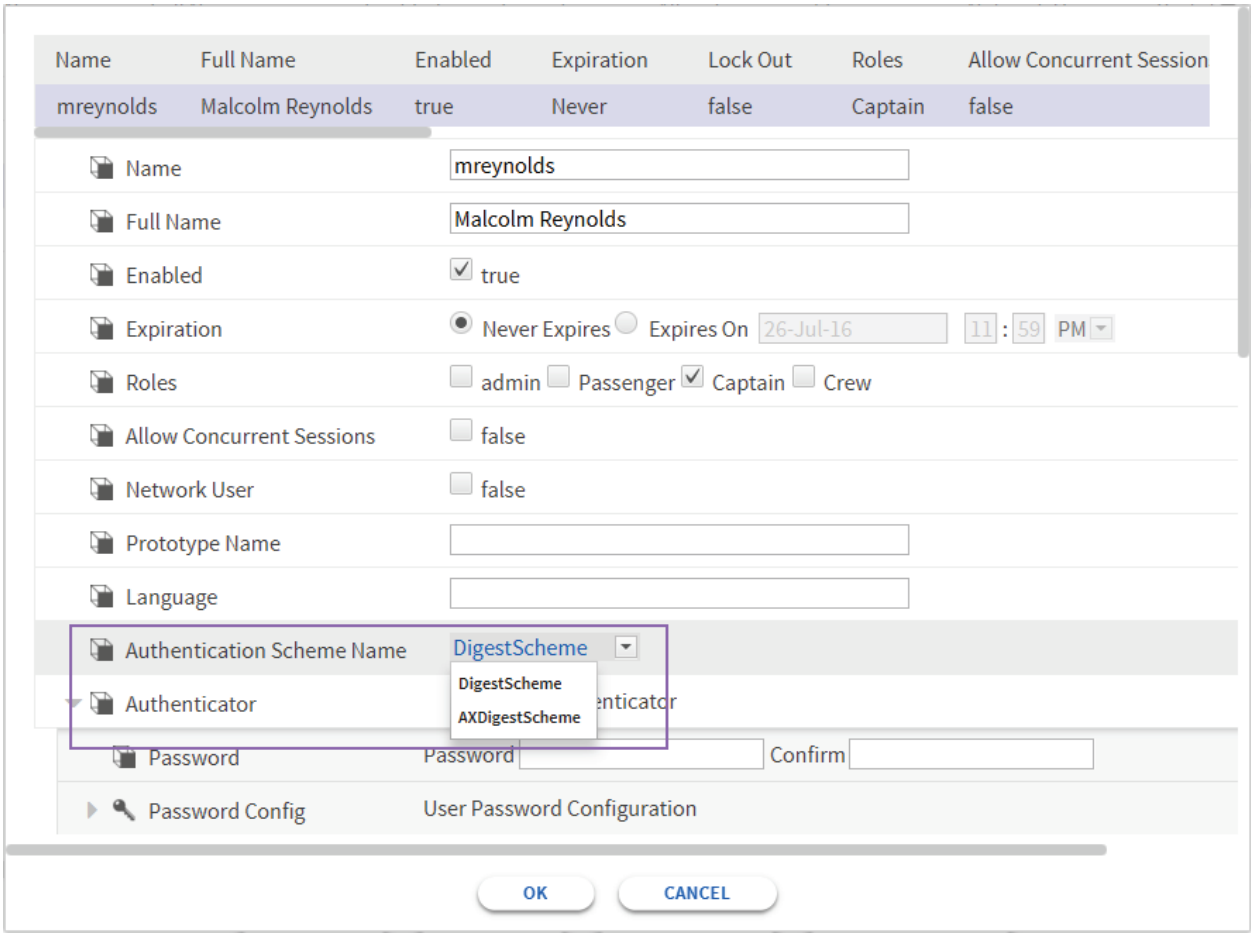
3. **Drag the authentication scheme you want your station to support to the "AuthenticationSchemes" folder and configure it as appropriate.**



Note: You can have multiple instances of the same authentication scheme type, configured differently. For example, you could have multiple DigestAuthenticationSchemes configured with different password strength requirements. Or, you could have different LdapAuthenticationSchemes pointing to different LDAP servers.

To configure your user account to use a particular authentication scheme, follow the steps below.

1. **Select the user you wish to configure in the UserService UserManager view.**
2. **Choose the authentication scheme you want to associate with that user from the "Authentication Scheme Name" property.**



Note: Certain authentication schemes (e.g. LdapAuthenticationScheme) support the notion of remote users. For these authentication schemes, it is not required to create the user ahead of time. When an unknown user attempts to log in, the scheme will automatically be attempted and the user will be created and configured on a successful login.

REMOVE UNNECESSARY AUTHENTICATION SCHEMES

A Niagara 4 station should only support the authentication schemes that it needs. Every new authentication scheme installed increases the station's attack surface: it provides a new point of entry for an attacker to attempt to exploit.

For example, every Niagara 4 station comes with the Digest and AXDigest authentication schemes installed by default. The AXDigestScheme allows Niagara AX stations to connect to a Niagara 4 station. If your station will not have Niagara AX stations connecting to it, you should remove the AXDigestScheme from the AuthenticationService.

To delete a scheme, simply delete it from the AuthenticationSchemes folder.

TLS & CERTIFICATE MANAGEMENT

NOTE: In late 2014, the POODLE vulnerability was discovered in SSLv3. As a result, SSLv3 support was removed from Niagara 4.

Transport Layer Security (TLS) provides communication security over a network by encrypting the communication at a lower level than the actual data being communicated. This allows secure transmission of unencrypted data (for example, the username and password in LDAP authentication) over an encrypted connection. TLS as a protocol replaces its predecessor, Secure Sockets Layer (SSL); however, because TLS originally evolved from the SSL standard, the terms "TLS" and "SSL" are often used interchangeably. Although many people still refer to TLS as "SSL", it is important to know that the latest version of SSL as a protocol (SSLv3) is not considered secure, and it is important to use the latest version of TLS available.

Using TLS protects data from anyone who might be eavesdropping and watching network traffic. It also provides proof of identity, so that an attacker cannot impersonate the server to acquire sensitive data. When possible, **always** use TLS.

Niagara 4 provides several opportunities for using TLS. You should use these options whenever they are feasible. Niagara 4 TLS options are listed below:

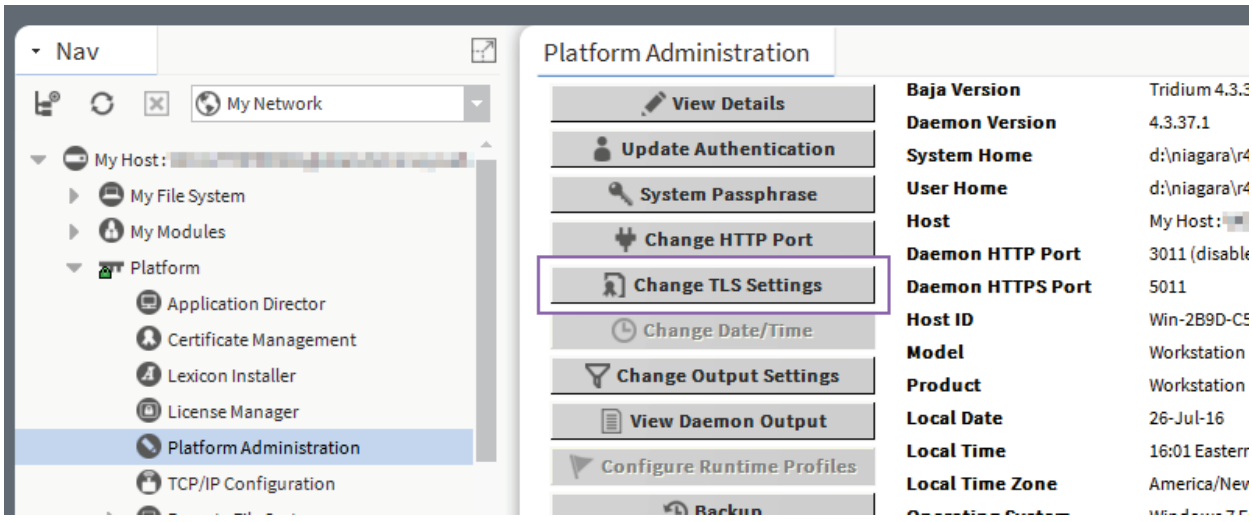
- Enable Platform TLS Only
- Enable Fox TLS Only
- Enable Web TLS Only
- Enable TLS on Other Services
- Set Up Certificates

ENABLE PLATFORM TLS ONLY

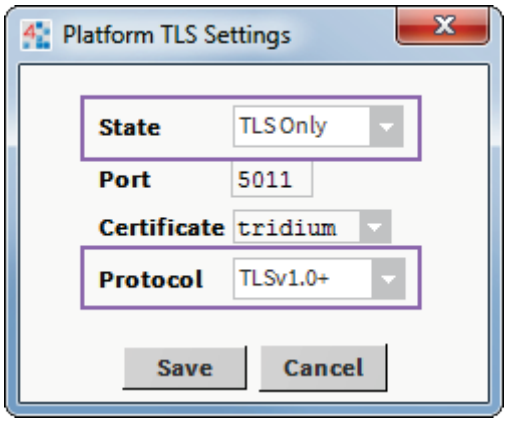
In Niagara 4, TLS can be enabled for platform connections.

To enable platform TLS, do the following:

1. **Open a platform connection.**
2. **Navigate to the "Platform Administration" view, and select "Change TLS Settings".**



3. A "Platform TLS Settings" dialog opens. Select "TLS Only" from the "State" drop down menu.



4. **Adjust the other fields as necessary.**

   - **Port.** The default port (5011) is generally acceptable, but may need to be changed due to IT constraints.

   - **Certificate.** This allows you to select the certificate you want to use for TLS. Note: the default self-signed tridium certificate only provides encryption and does not provide for server identity verification.
     Refer to the *Station Security Guide* for more details on certificates.

   - **Protocol.** This specifies which protocols are allowed. You can choose to use TLSv1.0 or higher, TLSv1.1 and higher, or TLSv1.2 only. IT or contractual constraints may require you to pick a particular setting.

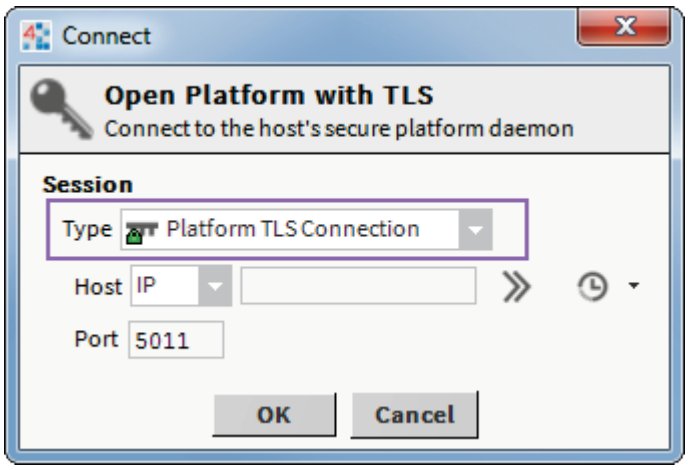
5. **Click the "Save" button. Close the platform connection.**

NOTE: If "State" is set to "Enabled" rather than "TLS Only," regular platform connections (not over TLS) are still permitted. Unless absolutely required, this should not be allowed. It places the burden of remembering to use TLS on the user initiating the connection – this can easily be forgotten, compromising security.

With TLS enabled for platform connections, a platform connection over TLS can be opened, as described below:

1. **Open the "Open Platform" dialog box.**

2. **Under the "Session" section, change the "Type" field to "Platform TLS Connection." Note that the dialog is updated.**

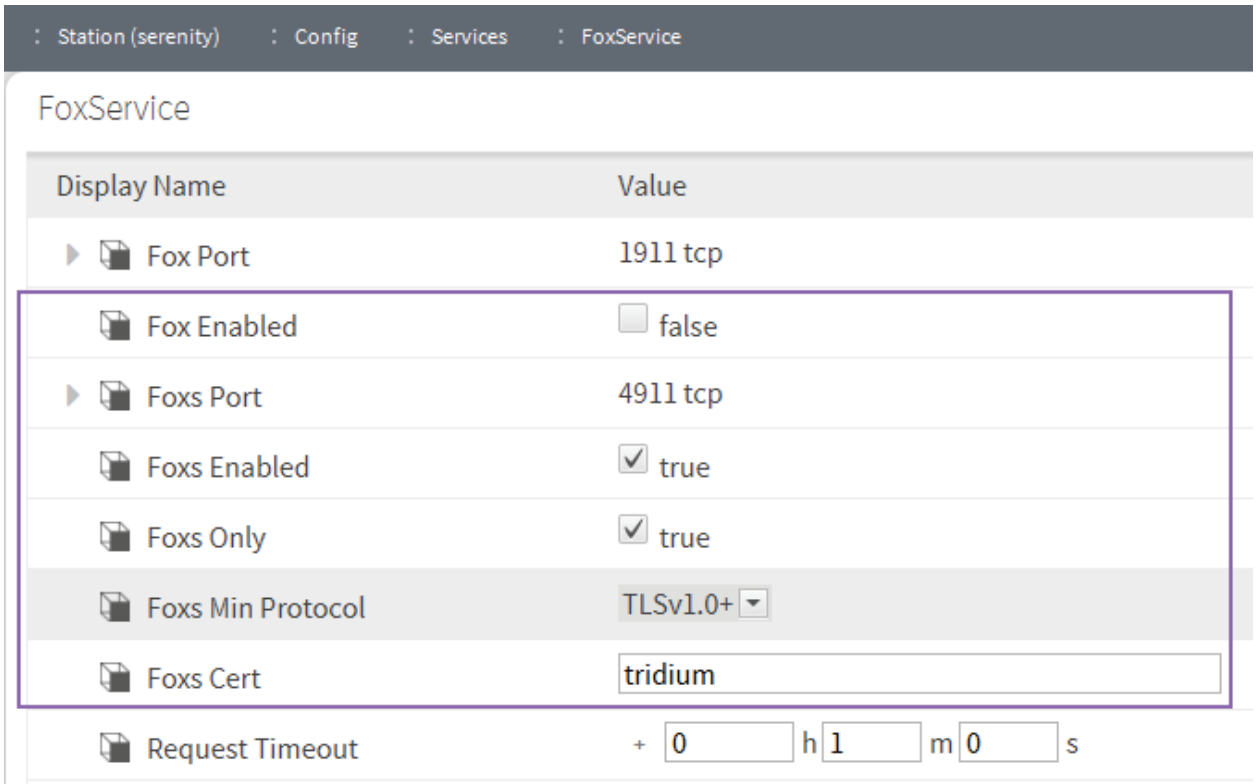


3. **Enter the IP, port and credentials for the platform and click "OK."**

ENABLE FOX TLS ONLY

In Niagara 4, TLS can be enabled for Fox connections, as outlined below.

1. **Open a station connection.**
2. **Open Config > Services > FoxService property sheet.**
3. **Set "Foxs Enabled" to "true."**
4. **Set "Foxs only" to "true."**



5. **Adjust the other Foxs settings as necessary.**

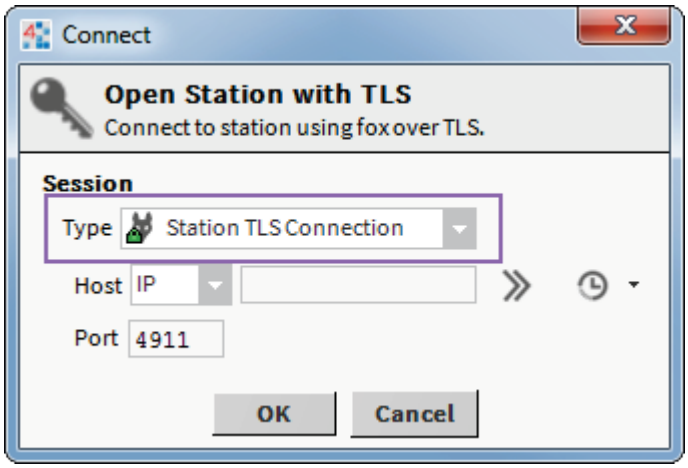
    - **Foxs Port.** The default port (4911) is generally acceptable, but may need to be changed due to IT constraints.
    - **TLS Min Protocol.** This determines what the minimum acceptable TLS version to use is.
    - **Foxs Cert.** This allows you to select the certificate you want to use for TLS. See the Station Security Guide for more details on certificates.

6. **Save the settings and close the station connection.**

NOTE: If "Foxs only" is not set to true, regular fox connections (not over TLS) are permitted. Unless absolutely required, this configuration should not be allowed, because it places the burden of remembering to use TLS on the user initiating the connection. This can easily be forgotten, compromising security. Leaving the "Fox Enabled" property set to true with "Foxs Only" also set to true provides a redirect to the Foxs port if a client attempts to make an unsecure Fox connection.

Now that TLS is enabled, a Foxs (Fox over TLS) connection can be opened, as described below:

1. **Open the "Open Station" dialog box.**
2. **Under the "Session" section, change the "Type" field to "Station TLS Connection." Note that the dialog box is updated.**



3. **Enter the IP, port and credentials for the station and click "OK".**

NOTE: A fox connection over TLS has a tiny lock on the fox icon ( ).

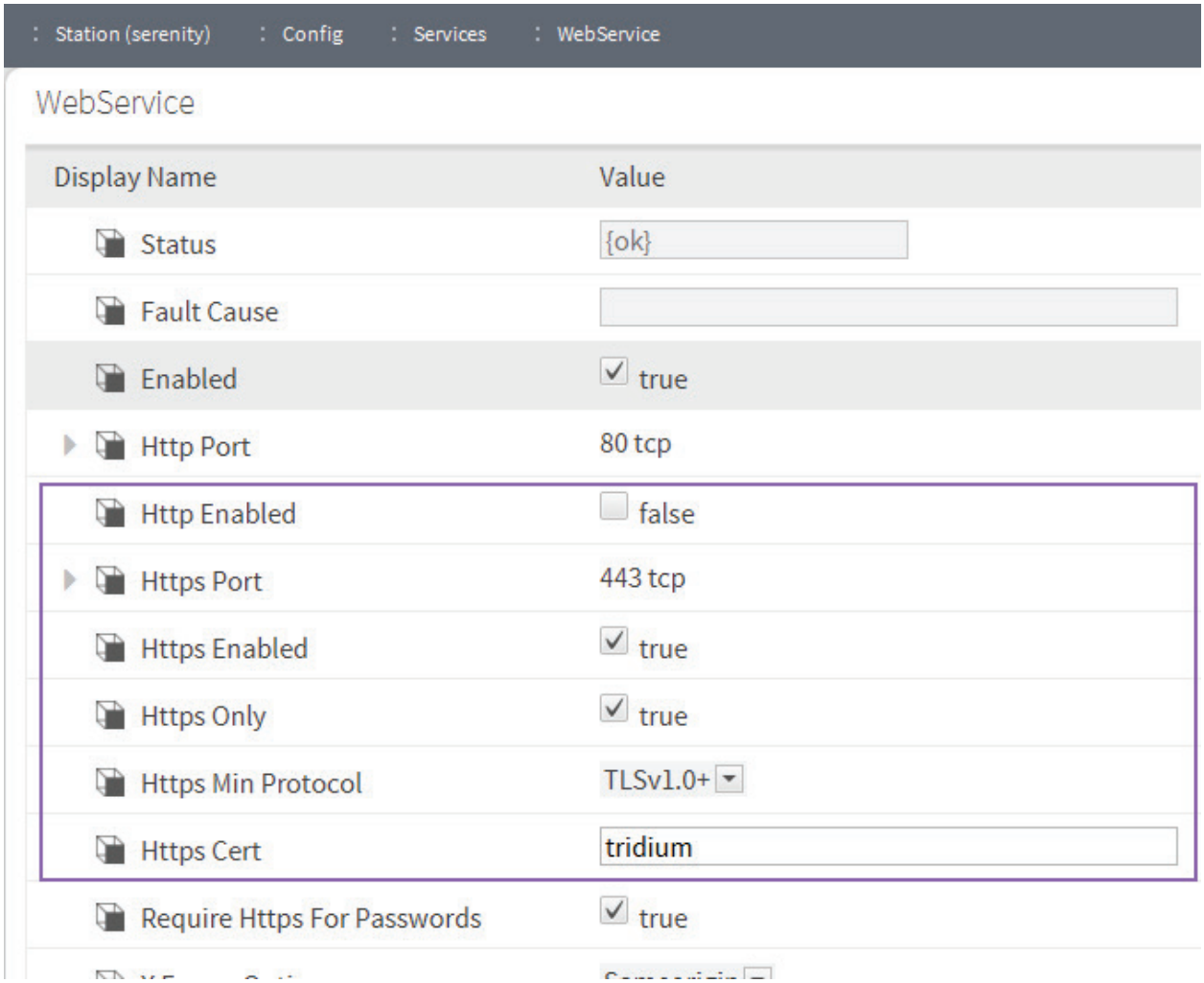
ENABLE WEB TLS ONLY

The steps to follow to enable TLS over HTTP are outlined below:

1. **Open a station connection.**
2. **Open Config > Services > WebService property sheet.**

3. **Set the "Https Enabled" property to "true."**

4. **Set the "Https Only" property to "true."**



5. **Adjust the other Https settings as necessary.**

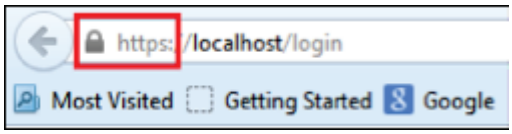
   - **Https port.** The default port (443) is generally acceptable, but may need to be changed due to IT constraints.
   - **TLS Min Protocol.** This determines what the minimum acceptable TLS version to use is.
   - **Https Cert.** This property allows you to select the certificate you want to use for TLS. Note that the default self-signed "tridium" only provides encryption and does not provide server identity verification. See the *Station Security Guide* for more details on certificates.

6. **Save the settings.**

NOTE: That if "Https only" is not set to true, regular http connections (not over TLS) will still be permitted. Unless absolutely required, this should not be allowed, because it places the burden of remembering to use TLS on the user initiating the connection – this can easily be forgotten, compromising security.

Now that TLS is enabled, an HTTPS connection can be opened. Here's how:

1. **Open a browser.**
2. **Navigate to the station's login page. If the server's certificate was signed by a valid CA then you probably will not see a prompt.**
3. **If prompted, you need to make your decision on whether or not to accept the Certificate based on an understanding of the circumstances. See the Station Security Guide for more details.**
4. **Note that you now have an https connection.**



ENABLE TLS ON OTHER SERVICES

There are a number of services in Niagara 4 that communicate with an outside server. For example, the EmailService OutgoingAccount and IncomingAccount both contact an email server. This connection is not the same as the fox or http connection used by the client to talk to the station, and TLS is handled separately for these types of connections. When setting up a new service on a station, check to see if it includes a TLS option. If TLS is an option, make sure that it is enabled. If needed, contact the IT department and make sure that the server the station needs to talk to supports TLS.

See the *Station Security Guide* and *User Guide* for details about setting up email with TLS features.

SET UP CERTIFICATES

Niagara 4 includes tools to help with certificate management. Certificates are required for TLS, and should be set up properly.

NOTE: Default certificates are self-signed and can only be used for encryption, not for server identity verification.

There are many things to consider when setting up certificates, and a full discussion is beyond the scope of this document. See the *Station Security Guide* for more information about correctly setting up certificates for a Niagara 4 system.

MODULE INSTALLATION

When installing modules in Niagara 4, there are extra steps that you can take to make sure that the modules you are installing will not negatively impact the security of your Niagara 4 system.

This step is listed below.

- Verify Module Permissions

VERIFY MODULE PERMISSIONS

Niagara 4 introduced the Java Security Manager, which places restrictions on who can run which code. Many modules do not have the permissions to run code that handles sensitive data or accesses files. This helps protect Niagara 4 systems from inadvertent or malicious tampering.

Starting in Niagara 4 version 4.2, modules can request additional permissions to the baseline granted to all modules. These permissions allow modules to perform certain specific tasks such as authenticating users via an authentication scheme, opening sockets, or reading system properties.

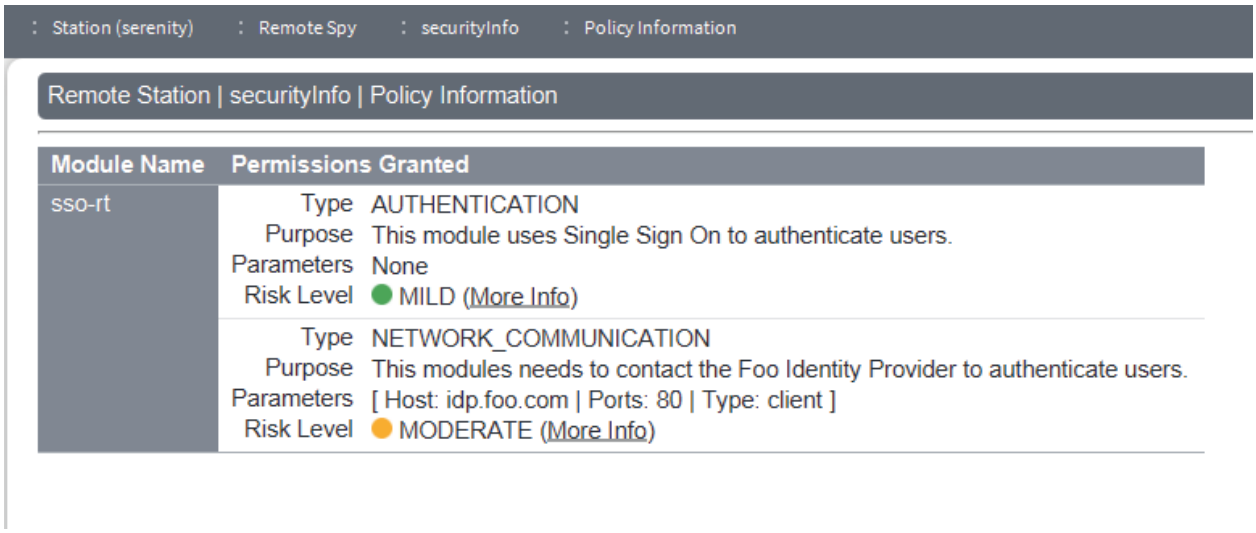
When installing new modules, care should be taken to inspect what permissions these modules are requesting and make sure that they match up with the functionality the module claims. For example, a module claiming to add a new UI scheme should probably not be opening a socket to www.super-suspicious-URL.com.

To verify what permissions modules have been granted, follow the steps below.

1. **Go to the station or Workbench's spy page.**

   NOTE: Modules request permissions for Workbench and stations separately so both should be verified.

2. **Go to securityInfo > Policy Information. The example below shows how the "sso-rt" module might request AUTHENTICATION and NETWORK_COMMUNICATION permissions in order to perform its Single Sign On functionality.**



3. **Verify that the permissions granted to the module match up with its intended functionality. In particular, validate that the "Purpose" field indicates a legitimate need for the permissions.**

ADDITIONAL SETTINGS

In addition to the settings discussed in previous sections, there are a few general settings to configure in order to secure a Niagara 4 system. These don't fall under a specific category like TLS or passwords, but are important to security.

- Require Signed Program Objects/Robots
- Disable SSH and SFTP

- Disable Unnecessary Services
- Configure Necessary Services Securely
- Update Niagara 4 to the Latest Release

REQUIRE SIGNED PROGRAM OBJECTS AND ROBOTS

Starting in Niagara 4.2, various components such as program objects and robots can be signed by a code signing certificate. A signed program object or robot will run only if the certificate that it was signed with is present in the Certificate Management's trust stores. Unsigned objects can always run. By default, signing is not required, but you can require program objects and robots to be signed by adding the "program.requireSigning=true" system property to your system.properties file.

NOTE: In future Niagara 4 releases, program object and robot signing may be required.

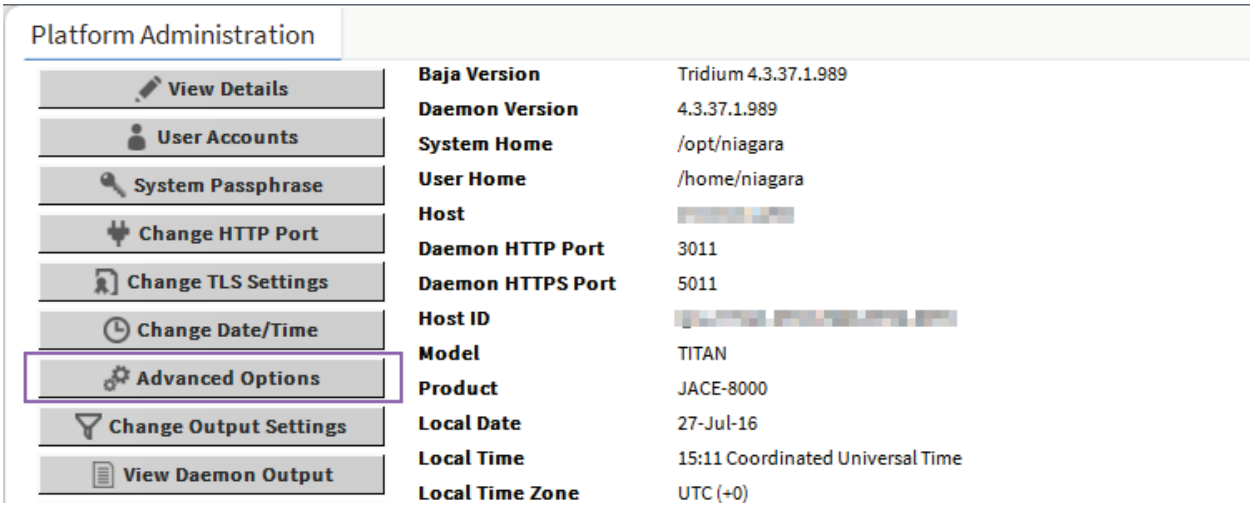
Requiring signed program objects ensures that only program objects and robots from trusted sources are allowed to run, and reduces the risk of malicious code being run on your Niagara 4 system.

DISABLE SSH AND SFTP

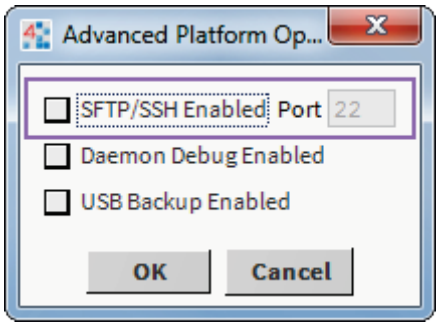
SFTP (Secure File Transfer Protocol) and SSH (Secure Shell) access to a JACE are disabled by default and should remain disabled unless necessary for troubleshooting or as directed by Tridium technical support. This helps prevent unauthorized access to the JACE. Enabling SFTP or SSH on a JACE poses a very significant security risk.

To ensure that SFTP and SSH are disabled on a JACE, follow these steps:

1. **Open a platform connection to the JACE controller.**
2. **In the Platform Administration view, click on "Advanced Options."**



3. **When the "Advanced Platform Options" dialog box opens, make sure that the "SFTP/SSH Enabled" box is not selected.**

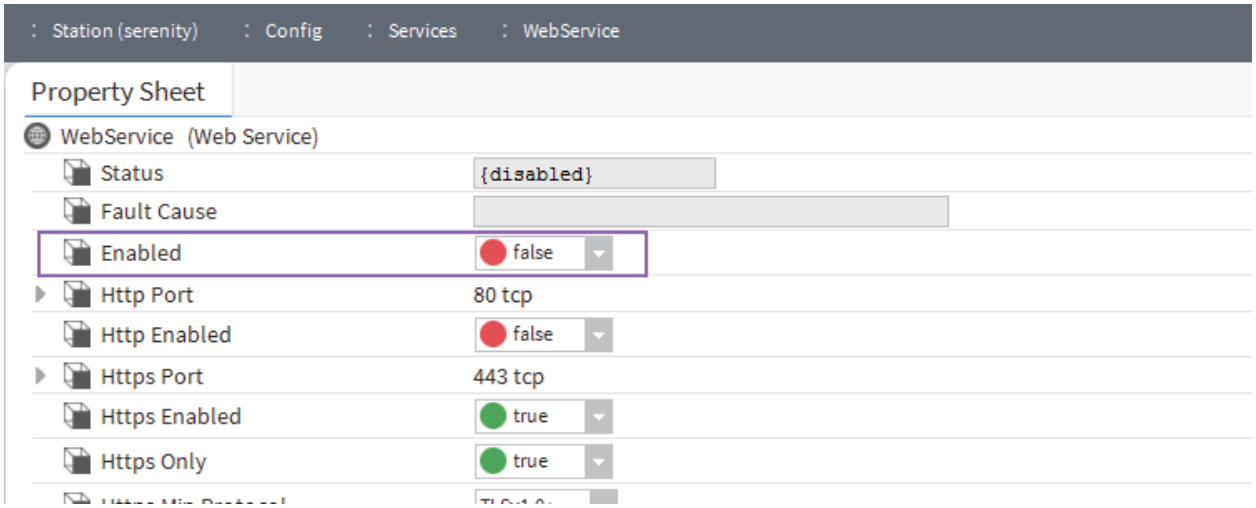


DISABLE UNNECESSARY SERVICES

When setting up a Niagara 4 station, either after creating a new station or copying an existing one, many services may already be installed and enabled in the Services folder. However, not every station has the same requirements. Services that are not required for what the station needs to do should be removed or disabled. This helps improve security by providing fewer openings for a potential attacker to exploit.

For example, if the station is not intended to be accessed via the web, then you should disable the WebService. This will prevent potential attackers from using the web to attempt to penetrate the station. The same consideration should be given to the other services.

To disable a service, either remove it from the station by deleting it, or go to the service's property sheet and look for an "Enabled" property. If one exists, set it to false, as shown below for WebService.



Figuring out what services are required means planning ahead of time how the station is intended to be used. Remember, a service can always be added or enabled, so it is best to start with only the services known to be required, and add services later as necessary.

CONFIGURE NECESSARY SERVICES SECURELY

If a service is required, care should be taken to configure that service securely. Different services have different settings affecting security, and the relevant documentation should be consulted when configuring a new service.

For example, when configuring the WebService, in addition to configuring TLS, the following settings should be on considered:

- The 'X Frame Options' property, set to 'SameOrigin' by default for backwards compatibility, should be set to 'Deny' when possible to protect against Cross Frame Scripting (XFS) attacks.
- 'Show Stack Traces' should be set to 'false' unless specifically debugging an issue, as a stack trace could reveal information that an attacker could use.
- The 'Require Https for Passwords' property should be set to 'true'. This enforces a TLS connection to perform operations such as updating a password.

UPDATE NIAGARA 4 TO THE LATEST RELEASE

Niagara 4 updates often include a number of important fixes, including security fixes. Niagara 4 systems should always be updated as soon as possible to ensure the best available protection. This is very important. Older releases may have known vulnerabilities – these are fixed as soon as possible, but if a system is not updated, it does not get the fixes.

EXTERNAL FACTORS

In addition to station and platform settings, there are some external factors to consider when securing a Niagara 4 system.

- Install JACEs in a Secure Location
- Make Sure that Stations Are Behind a VPN

INSTALL JACES IN A SECURE LOCATION

Restricting physical access to JACE controllers is essential to security. If an attacker can physically connect to the JACE using a cable, they can gain complete control of the system. This could potentially be disastrous. Keep JACEs secure in a locked room with restricted access.

MAKE SURE THAT STATIONS ARE BEHIND A VPN

A station exposed to the Internet is a station at risk. Anyone who discovers the station's IP address can attempt an attack, either to gain access to the system or to bring the system down. Even stations that have been configured to use TLS only are at risk for a denial-of-service attack. Keeping stations behind a properly configured VPN ensures that they are not exposed, reducing the system's attack surface. For more information, see "Using a VPN with Niagara Systems" available from the Niagara Framework Software Security Resource Center on Niagara Community.

Do not assume that because you have not shared the station's IP address with anyone that it cannot be discovered – that is not the case. **There are tools that already exist** to discover exposed Niagara 4 systems without knowing the IP addresses beforehand.

APPENDIX A: CREATING STRONG PASSWORDS THAT ARE ACTUALLY STRONG

Most Niagara 4 systems which use passwords enforce some password strength, which may or may not be customizable. However, password strength requirements alone are not sufficient to ensure that a password is truly strong. A good example is “Password10”: it satisfies all the password strength requirements, but is actually a weak password that is easy to crack. Dictionary words followed by a few numbers are an extremely common password pattern and will be quickly guessed by an attacker.

When creating passwords, the following guidelines can help generate stronger passwords:

- A random string of characters, including digits and uppercase, lowercase and special characters, (e.g. s13pj96t!cD) is typically a strong password. However, these can be hard to remember.
- A long, nonsensical sentence (e.g. “I happily tarnished under 21 waterlogged potatoes, which meet up on Sundays”) can be used as is. For systems that restrict password length, it can be contracted to include only the first character of each word (e.g. “Ihtu21wp,wmuoS”). These are difficult for attackers to guess, but are typically easy (albeit silly) for users to remember.

Note: when picking a sentence as a passphrase, it is best to avoid well-known phrases and sentences, as these may be included in dictionary attacks (e.g. “Luke, I am your father”).

- A string of random words (e.g. “coffee Strange@ Halberd 11 tortoise!”) provides a much longer password that a single word or a random string of characters. However, password crackers are becoming more aware of this technique, and inserting few random numbers and symbols in there can help.

Remember, a good password is easy for a user to remember, but difficult for an attacker to guess.

APPENDIX B: BLACKLIST SENSITIVE FILES AND FOLDERS

In Niagara, a blacklist feature is available. Many of the files listed in the blacklist are blocked by the Security Manager in Niagara 4. However, there may be cases where it is useful to blacklist additional files and/or folders.

When you implement this feature, files and folders on the blacklist are not accessible remotely through the station. This helps to protect sensitive files from being tampered with. For example, if an attacker is able to get into the station using a web connection, access to any file in the blacklist is still denied.

Some folders are always blacklisted, such as the following: /backups, /bin, /daemon, /files, /jre, /modules, /registry, /security, /users and /workbench.

Refer to the "system.properties notes" section in the *Platform Guide* for more details about the location of the system.properties file, blacklisting and more notes and cautions about editing the file.

Additional files may be blacklisted by editing the system.properties file, as described below.

To edit the system.properties blacklist

1. **Open the system.properties file.**
2. **Uncomment the "niagara.remoteBlacklist.fileNamePatterns" line and add any file patterns that should be blacklisted (for example, *.bog).**

```
# The following property allows for specification of additional
# file name patterns to blacklist from remote station access.
# File name patterns are delimited by a semicolon, and follow the format
# defined in javax.baja.util.PatternFilter. For example, a value of
# *.txt;*.xml would restrict any text or xml file from being accessed
# remotely through the station (ie. from the web or through a fox
# connection in Workbench).
#niagara.remoteBlacklist.fileNamePatterns=*.bog
```

3. **Uncomment the "niagara.remoteBlacklist.filePaths" line and add any folders that should be blacklisted (for example, !lib).**

```
# The following property allows for specification of additional
# file paths to blacklist from remote station access (ie. from the
# web or through a fox connection in Workbench).
# File paths are delimited by a semicolon, and follow the body format
# defined in javax.baja.file.FilePath. For example, a value of
# !licenses;!modules would restrict access to the licenses and modules
# directories under the Niagara sys home.
#niagara.remoteBlacklist.filePaths=!lib
```

4. **The station must be restarted before changes to system.properties become effective.**

The added file patterns or folders depend on the particular Niagara installation. Consider what needs to be protected and does not absolutely need to be accessed remotely.

APPENDIX C: HARDENING CHECKLIST

This section presents the information in the Niagara 4 Hardening Guide in a convenient checklist. The list can be used to verify that all the described steps to secure your Niagara 4 system have been followed.

The checklist is included for convenience. However, it is important to remember that the goal is not to check boxes on a list. You need to have a good understanding of the security reasoning behind each of the boxes. Moreover, security is an ongoing process. You should always be on the lookout for areas in which you can improve security, whether they are on the list or not.

- ❑ Passwords
  - ❑ Use the Password Strength Feature
  - ❑ Enable the Account Lockout Feature
  - ❑ Expire Passwords
  - ❑ Use the Password History
  - ❑ Use the Password Reset Feature
  - ❑ Leave the “Remember These Credentials” Box Unchecked
- ❑ System Passphrase
  - ❑ Change the Default System Passphrase
  - ❑ Use TLS To Set the System Passphrase
  - ❑ Choose a Strong System Passphrase
  - ❑ Protect the System Passphrase
  - ❑ Ensure Platform Owner Knows the System Passphrase
- ❑ Platform Account Management
  - ❑ Use a Different Account for Each Platform User
  - ❑ Use Unique Account Names for Each Project
  - ❑ Ensure Platform Owner Knows the Platform Credentials
- ❑ Station Account Management
  - ❑ Use a Different Account for Each Station User
  - ❑ Use Unique Service Type Accounts for Each Project
  - ❑ Disable Known Accounts When Possible
  - ❑ Set Up Temporary Accounts to Expire Automatically
  - ❑ Change System Type Account Credentials
  - ❑ Disallow Concurrent Sessions When Appropriate
- ❑ Role & Permission Management
  - ❑ Configure Roles with Minimum Required Permissions
  - ❑ Assign Minimum Required Roles to Users
  - ❑ Use the Minimum Possible Number of Super Users

- ❑ Require Super User Permissions for Program Objects
- ❑ Use the Minimum Required Permissions for External Accounts

❑ Authentication

- ❑ Use an Authentication Scheme Appropriate for the Account Type
- ❑ Remove Unnecessary Authentication Schemes

❑ TLS & Certificate Management

- ❑ Enable Platform TLS Only
- ❑ Enable Fox TLS Only
- ❑ Enable Web TLS Only
- ❑ Enable TLS on Other Services
- ❑ Set Up Certificates

❑ Module Installation

- ❑ Verify Module Permissions

❑ Additional Settings

- ❑ Require Signed Program Objects and Robots
- ❑ Disable SSH and SFTP
- ❑ Disable Unnecessary Services
- ❑ Configure Necessary Services Securely
- ❑ Update Niagara 4 to the Latest Release

❑ External Factors

- ❑ Install JACEs in a Secure Location
- ❑ Make Sure that Stations Are Behind a VPN