Technical Document

# Niagara IEEE 802.1X Configuration Guide

**May 1, 2019**

# Niagara IEEE 802.1X Configuration Guide

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

## Confidentiality



## Trademark notice



## Copyright and patent notice

# Contents

# About this guide

This topic contains important information about the purpose, content, context, and intended audience for this document.

### Product Documentation

This document is part of the Niagara technical documentation library. Released versions of Niagara software include a complete collection of technical information that is provided in both online help and PDF format. The information in this document is written primarily for Systems Integrators. In order to make the most of the information in this book, readers should have some training or previous experience with Niagara 4 or NiagaraAX software, as well as experience working with JACE network controllers.

### Document Content

This document describes how to use the Niagara IEEE 802.1X Wired Authentication standard on JACE-8000 and Edge devices. This document introduces the standard and explains how to install and configure IEEE 802.1X on the host.

# Document change log

Updates (changes/additions) to this guide are listed below.

### May 1, 2019

Initial Niagara 4 publication.

# Related documentation

- The *Niagara Station Security Guide*
- *JACE-8000 Install and Startup Guide*
- *Niagara Edge 10 Install and Startup Guide*

### Additional information

The following links to Wikipedia articles provide additional information about IEEE 802.1X wired authentication.

- IEEE 802.1X: https://en.wikipedia.org/wiki/IEEE_802.1X
- Extensible Authentication Protocol: https://en.wikipedia.org/wiki/Extensible_Authentication_Protocol
- wpa_supplicant: https://en.wikipedia.org/wiki/Wpa_supplicant
- Protected Extensible Authentication Protocol: https://en.wikipedia.org/wiki/Protected_Extensible_Authentication_Protocol

# Chapter 1 Preparation and installation

**Topics covered in this chapter**

- ♦ Requirements
- ♦ Installing software
- ♦ Setting up security

In Niagara 4.8 and later, there is added support for the IEEE 802.1X Wired Authentication standard on JACE-8000 and Edge platforms. On the primary adapter of a JACE-8000 or Edge device, you have the option to enable 802.1X security on the device.

IEEE 802.1X is a wired authentication protocol that protects a closed network from unauthenticated access. In 802.1X terminology, the remote device or "client" is referred to as the "supplicant". The JACE/Edge supplicant device requests network access via a port on the secure network. Once successfully authenticated to the port, the supplicant device can access the network.

Topics in this chapter explain how to install software, and set up security on the device using the **IEEE 802.1X Configuration** view in Workbench.

This information is not required for Supervisor stations.

## Requirements

Niagara 4.8 makes it possible to configure a JACE-8000 or Edge device to run as a supplicant (client) on an IEEE 802.1X network.

This document assumes that you have experience with network authentication, and understand the properties that need to be configured. Consult your local IT network administrator if you have any questions.

### Hardware requirements

A JACE-8000 or Edge device.

**NOTE:** IEEE 802.1X is supported only on the device's Primary Ethernet adapter.

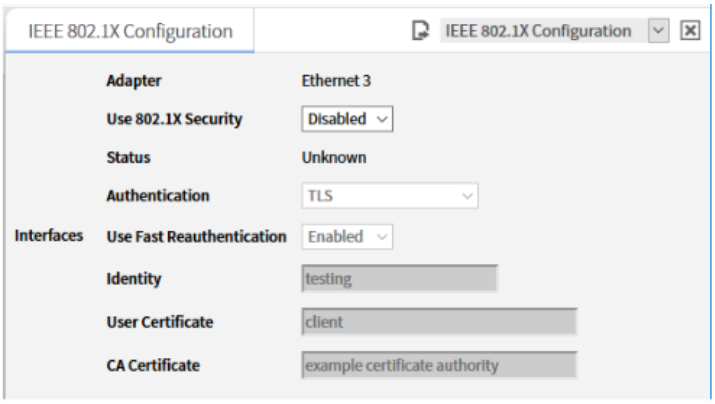
**NOTE:** For Edge devices, IEEE 802.1X is supported only in isolated mode. It is not supported on an Edge device in daisy-chain mode.

### Software requirements

- You must have a properly licensed Niagara 4.8 release installed and running.
- You need to have platform credentials in order to configure the device for IEEE 802.1X communications since this is configured only at the platform level.
- No modules are required for the platform-level connection and configuration. However, for the station-level **`IEEE 802.1X Platform Service Plugin`**, you will need to make sure that the platIEEE8021X (-rt, -wb) modules are available in the Workbench environment so that the views are available. However, it is not a requirement that these modules are installed to the supplicant device (only the Workbench environment).

**NOTE:** The station-level view (shown right) is a read-only view which can be useful for confirming connection status.

Figure 1 Configurable Workbench view (left), Read-only station-level view (right)




### License requirements

The Niagara 4.8 `ieee8021x` license feature must be installed on the device.

### Certificate requirements

**NOTE:** You will need to coordinate with your local IT department/network administrator for the following items:

- The authentication scheme required by the network (e.g., EAP-TLS, PEAP, etc.), used in configuring the supplicant device for IEEE 802.1X communications
- A client certificate (`*.pem` format) for the supplicant device, the "identity" associated with that certificate, and possibly a "private key password".

  **NOTE:** The client certificate should include the client's private key and it may include the optional "private key password". If the certificate uses one, then that private key password is required in order to use the certificate.
- A CA certificate (`*.pem` format). The CA certificate is a certificate that can confirm the identity of the server to the supplicant. Both of the certificates are used in establishing a connection to the 802.1X network.

Once you obtain the client and CA certificates, save the `*.pem` files somewhere on the Workbench (Supervisor) file system. When setting up security on the supplicant device, the client and CA certificate files will be imported to the device's **Certificate Management** view **User Key Store** and **User Trust Store** respectively.

## Installing software

This procedure covers installing the software on the supplicant device via commissioning.

**Prerequisites:**

- The Supervisor installation (Workbench) used to commission the JACE/Edge device is has the Niagara 4.8 (or later) release installed.
- If you wish to have the station-level configuration view, the latest `ieee8021X-*.jar` files should be in the `!modules` folder for the release of Niagara you are using, where `!` replaces the folder path.

**NOTE:** The Nav Container view for the platform is always available for the Niagara 4.8 Workbench. If the Workbench environment has the necessary modules, no additional modules need to be installed for the platform-level **IEEE 802.1X Configuration** view.

Step 1 Connect the Supervisor and supplicant device to an open (not secure) network.

Step 2 Download from Niagara Central the latest Niagara 4.8 installer (or the module files: `platIEEE8021X-rt.jar` and `platIEEE8021X-wb.jar`, if the latest version of Niagara 4.8 is already installed).

Step 3 Run `setup.exe` or save the modules in the `Niagara\version\modules` folder, where `version` is the release version number of Niagara.

Step 4 Commission the supplicant device, installing `platIEEE8021X-rt.jar` during the software step.

**NOTE:** If not done already, be sure to install the Niagara 4.8 `ieee8021x` license feature on the supplicant device.

Step 5 Once commissioning is complete and the device reboots, connect to the platform and continue the installation by setting up security, and configuring IEEE 802.1X properties (described in separate procedures following this one).

When you have configured the supplicant device, disconnect from the open port and connect it to an authenticated network segment. If you can access the supplicant device using the expected IP address, your configuration has worked. You may need to use a serial cable to diagnose problems.

**NOTE:** See the System Shell menu for added IEEE 802.1X options.

## Setting up security

Before you can configure IEEE 802.1X properties you need to import to the remote device's Certificate Manager a client certificate and a CA certificate. The client certificate is used to authenticate the device to the 802.1X port on the network. The CA certificate is used to validate the server that the device is connecting to. The certificates must be imported to the device, prior to configuring the IEEE 802.1X properties.

**Prerequisites:**

- You are working in Workbench on a computer that is not on the Internet and is in a physically secure location.
- You have already obtained from the local IT network administrator a client certificate (with private key password if required) for each JACE/Edge device, and a CA certificate, and saved the certificate `*.pem` files on the Workbench local file system.

  **NOTE:** Note that the it may not be necessary to have a separate client certificate (i.e. different identity) for each device, you could re-use a client certificate across multiple devices if your local IT network administrator allows that.
- You have platform credentials for the remote device.
- You have already commissioned the remote device with required software and license feature and rebooted the device.

Step 1 In Workbench open a platform connection to the remote device and in the **Nav Container View**, double-click **Certificate Management** to open the view.

Step 2 In the **Certificate Management** view for the device, on the **User Key Store** tab, click **Import**.

Step 3 In the **Certificate Import** window, navigate to locate the client certificate `*.pem` file for the device (previously stored locally on the Workbench file system), select the file and click **Open**.

Step 4 If prompted for the certificate's **Private Key Password**, enter the password and click **OK**.

**NOTE:** This step is necessary only if the certificate uses a private key password. If it does not, then you will not be prompted to enter one.

Step 5 When the selected certificate's properties display in the **Certificate Import** window, click **OK** to complete importing the certificate.

The client certificate is imported to the device's Key store.

Step 6 In the **Certificate Management** view, click the **User Trust Store** tab, and click **Import**.

Step 7 In the **Certificate Import** window, navigate to locate the CA certificate `*.pem` file for the device (stored locally on the Workbench file system), select the file and click **Open**.

Step 8 When the selected CA certificate's properties display in the **Certificate Import** window, click **OK** to complete importing the certificate.

The CA certificate is imported to the device's Trust store.

# Chapter 2 Configuring adapter settings

**Topics covered in this chapter**

♦ Configuration examples
♦ platIEEE8021X-IEEE8021XDaemonSessionPlugin

Once the required certificates are imported to the remote device's **Certificate Manager**, you will use the Workbench **IEEE 802.1X Configuration** view to configure the 802.1X properties on the primary adapter for the device.

Step 1 Click the **Use 802.1X Security** checkbox to enable this feature.

Step 2 Click the **Authentication** dropdown list and click on the required method (TLS, Tunneled TLS, or Protected EAP).

Step 3 Click in the **Identity** field, and enter the identity associated with the client certificate.

"Identity" is indicated during certificate creation. Coordinate with the network administrator who provided the certificate for this value.

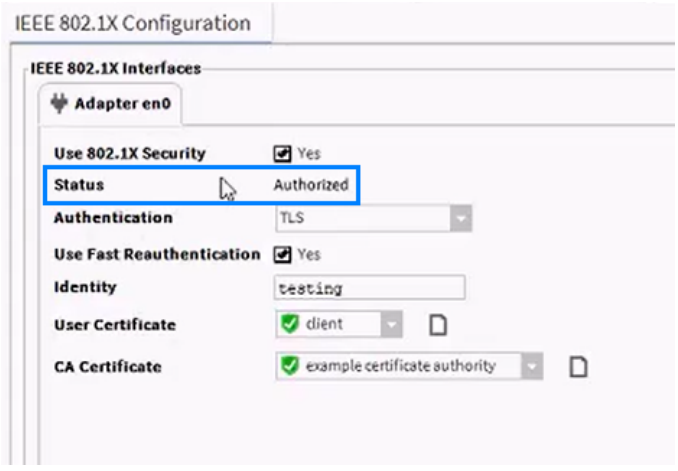
Step 4 Click on the **User Certificate** dropdown and click to select the client certificate that you previously imported.

Step 5 Click on the **CA Certificate** dropdown and click to select the CA certificate that you previously imported.

Step 6 Click **Save**.

**NOTE:** On saving your changes a device reboot is required. This is because the 802.1X connection is established during booting. If the device is unable to authenticate to the 802.1X port on the network, the connection will fail.

Once the reboot completes, open the **IEEE 802.1X Configuration** view to confirm that the device Status is "Authorized", which tells you that the supplicant device is correctly configured and connected on the primary adapter with 802.1X.



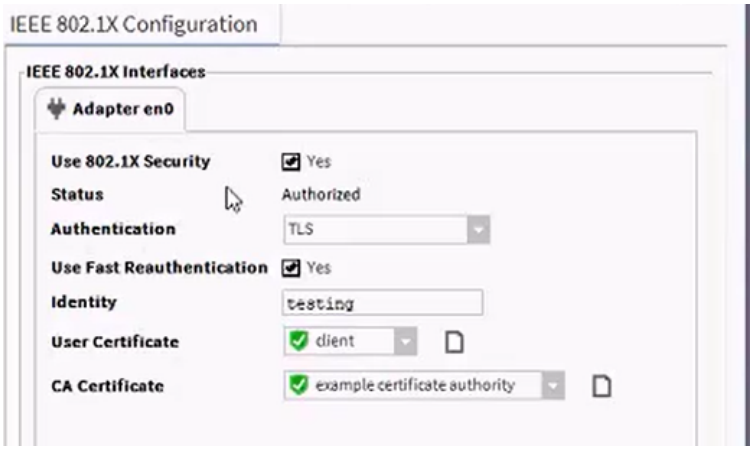

**NOTE:** If the device is configured with the wrong certificates, wrong credentials, or a scheme that is not supported, the connection will fail and you will see Status is "Unauthorized".

Once the supplicant device is successfully connected on the network, it behaves the same as any other device on a non-protected network.

# Configuration examples

The following examples are provided to show the primary adapter configured for IEEE 802.1X using several different authentication methods. Note that the certificate drop-downs are populated with certificates that you installed through the Certificate Management Platform Service.

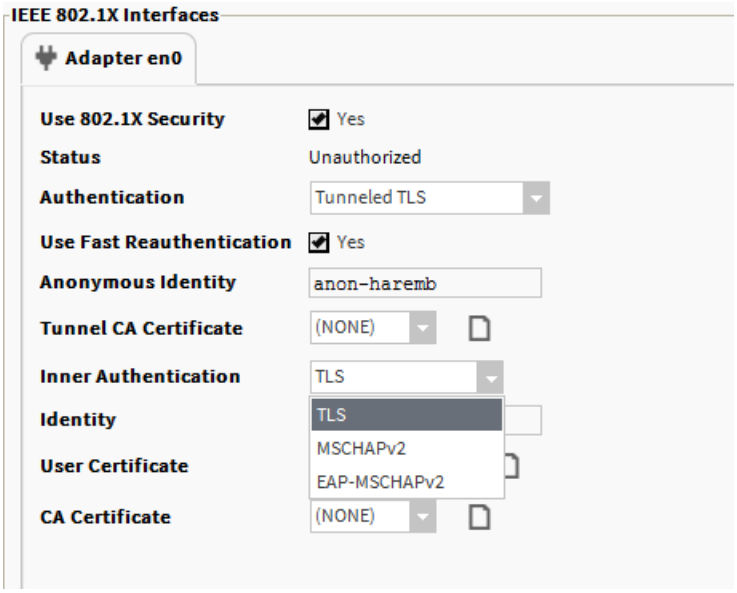
## Primary adapter configured to use IEEE 802.1X with EAP-TLS




The certificate drop-downs are populated with certificates you installed through the Certificate Management Platform Service.

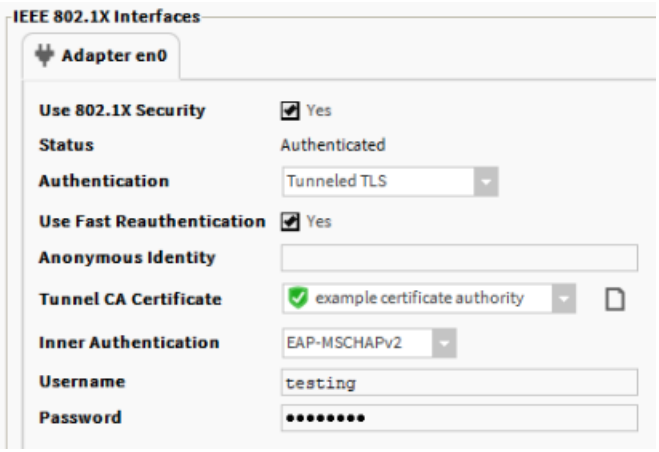
## Primary adapter configured to use IEEE 802.1X with Tunneled TLS and TLS as the inner authentication method

Figure 2 Adapter configured to use Tunneled TLS and TLS method



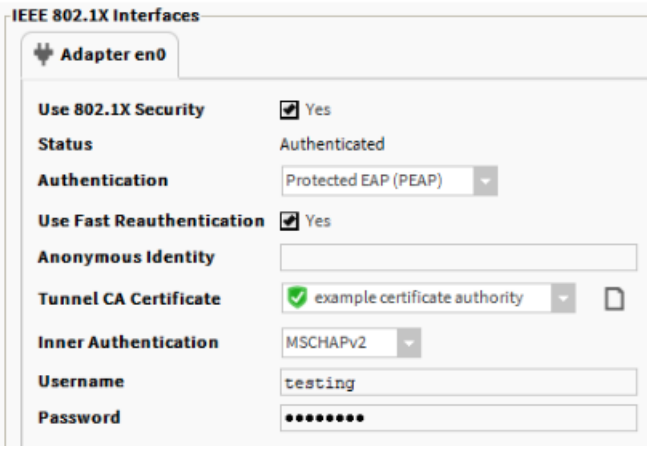
### Primary adapter configured to use IEEE 802.1X with Tunneled TLS and EAP–MSCHAPv2 as the inner authentication method

Figure 3 Adapter configured to use Tunneled TLS and EAP–MSCHAPv2 method




### Primary adapter configured to use IEEE 802.1X with PEAP and MSCHAPv2 as the inner authentication method

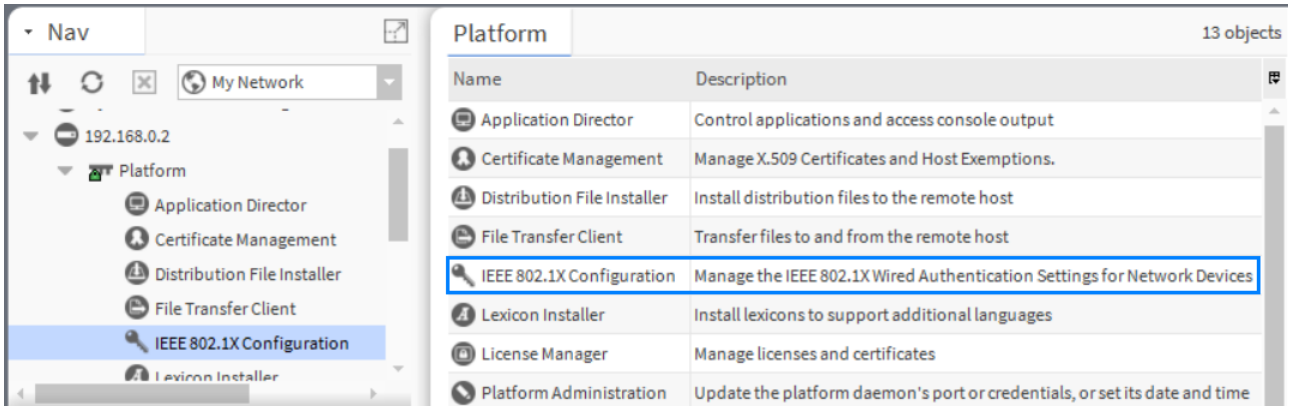
Figure 4 Adapter configured to use PEAP and MSCHAPv2 method



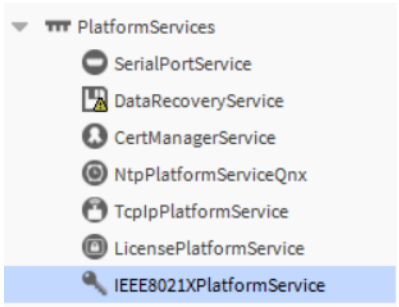

## platIEEE8021X-IEEE8021XDaemonSessionPlugin

In Niagara 4.8 and later, the **IEEE 802.1X Configuration** view is the main view for configuring a JACE-8000 or Edge device for communications on a 802.1X protected network. This view is available from the platI-EEE8021X module.

**Figure 5** Access IEEE 802.1X Configuration view from Nav Tree or Nav Container View

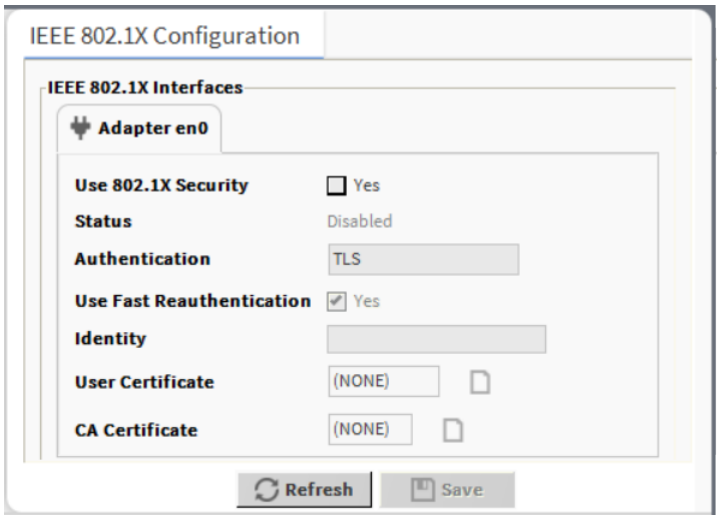



**NOTE:** If platIEEE8021X (-rt, -wb) modules are installed on the supplicant device, you can also access a read-only view of the IEEE 802.1X settings and connection status in the Nav Tree under the station's PlatformServices node, as shown here.




The IEEE 802.1X settings are configured in the Workbench environment on the platform's primary adapter via the **IEEE 802.1X Configuration** view. You can access the view from the platform's **Nav Container View** or from a node in the Nav tree.

**Figure 6** IEEE 802.1X Configuration view




## About authentication methods

IEEE 802.1X uses Extensible Authentication Protocol (EAP) to provide security. The available EAP authentication methods are:

- **EAP-TLS** is certificate-based and mutual authentication of client-to-server and server-to-client. It relies on client-side and server-side certificates to perform authentication.
- **Tunneled TLS** provides for certificate-based, mutual authentication of the client and server through an encrypted channel (or tunnel); a means to derive dynamic, per-user, per-session WEP keys; and requires only server-side certificates.
- **Protected EAP (PEAP)** provides a method to transport secure authentication data using tunneling between PEAP clients and an authentication server.

### 802.1X properties for device's primary adapter

| Name | Value | Description |
|---|---|---|
| Use 802.1X Security | Yes, No (default) | Enables/disables use of this feature. Indicates whether IEEE 802.1X is being used on the platform |
| Status | Disabled (default), Authorized, Unauthorized, Unknown, Unlicensed | Read only value, indicates current network connection status. |
| Authentication | TLS (default), Tunneled TLS, Protected EAP | Choose the EAP method required by the network. |
| Use Fast Reauthentication | Yes (default), No | By default, fast re-authentication is enabled for all EAP methods that support it. This variable can be used to disable fast re-authentication. Normally, disabling this is only necessary if your network infrastructure (RADIUS) does not support Fast Re-authentication. |
| Identity | string | Identity string for EAP. This is indicated during client certificate creation. It can be obtained from the local IT network administrator. |
| User Certificate | | Select the client certificate alias for the EAP. The certificate should be in PEM format with a `.pem` file extension. The client certificate (with private key password if the certificate uses one) for each device, obtained from the local IT network administrator, is required. This field is populated with certificates available in the platform's Certificate Manager User Key Store. |
| CA Certificate | | Select the Certificate Authority (CA) certificate alias to be used for the EAP. This certificate should be in PEM format with a `.pem` file extension. This required cert is the CA certificate provided by the network administrator. This field is populated with certificates available in the platform's Certificate Manager User Trust Store. |

### Additional properties for the Tunneled TLS and Protected EAP authentication methods

| Name | Value | Description |
|---|---|---|
| Anonymous Identity | string | This is the string for EAP (to be used as the unencrypted identity with EAP types that support different tunnelled identity, e.g., EAP-TTLS) |
| Tunnel CA Certificate | | This is used in inner authentication with TLS tunnel when using EAP-TTLS or EAP-PEAP. This CA certificate is required. There can be one or more trusted CA certificates. |
| Inner Authentication | TLS (default), MSCHAPv2, EAP-MSCHAPv2 | The specified authentication scheme to be used "inside" the tunnel for schemes like PEAP and Tunneled TLS. |

# Chapter 3 Troubleshooting

**Topics covered in this chapter**

♦ Runtime behavior
♦ Additional details in system shell
♦ Uninstall

### Configuration view under the Platform node in the Nav tree is not available or servlet <ieee8021X> not started, missing license feature "ieee8021x"

Use of the ieee8021X module requires a license. Contact your sales representative.

### I am unable to verify that IEEE 802.1X wired authentication is working.

Log in to your own network infrastructure and check out the error messages coming from the attempted connection.

## Runtime behavior

You should be able to confirm that IEEE 802.1X is enabled during runtime. When you save IEEE 802.1X configuration changes, immediately confirm an update of the `/etc/IEEE 802.1X.conf` file. The location of the file on the supplicant device filesystem is `/etc/ieee8021x/wpa_supplicant_[adapter_name].conf`. The settings contained in this file reflect the current values in the **IEEE 802.1X Configuration** view. The settings are not applied, however, until after a reboot.

When the QNX 7 network stack starts up, it checks for the configuration file. If it finds the file, the stack launches wpa_supplicant. You should be able to confirm through `pidin arg` that the application started with the following parameters:

`/etc/ieee8021x/wpa_supplicant_en0.conf (EDGE)` or `/etc/ieee8021x/wpa_supplicant_dm0.conf (TITAN)`

- `-P` determines where the pid (process identifier) for the wpa_supplicant instance will be located.
- `-i` determines which interface the wpa_supplicant should use for IEEE 802.1X authentication.
- `-D` specifies which driver should be used with wpa_supplicant.
- `-B` specifies that the wpa_supplicant should be launched as a daemon process.
- `-c` specifies the configuration file to be used for wpa_supplicant.

If you properly configured the wpa_supplicant, it is fairly easy to determine if it is working. If the results of `ifconfig` indicate that the adapter has an IP address, then wpa_supplicant is working. If you are unable to communicate to a JACE through TCP/IP protocols, then wpa_supplicant is not properly configured.

You should also be able to determine that wpa_supplicant started from the following initialization message:

```
launching 802.1x supplicant for interface en0
```

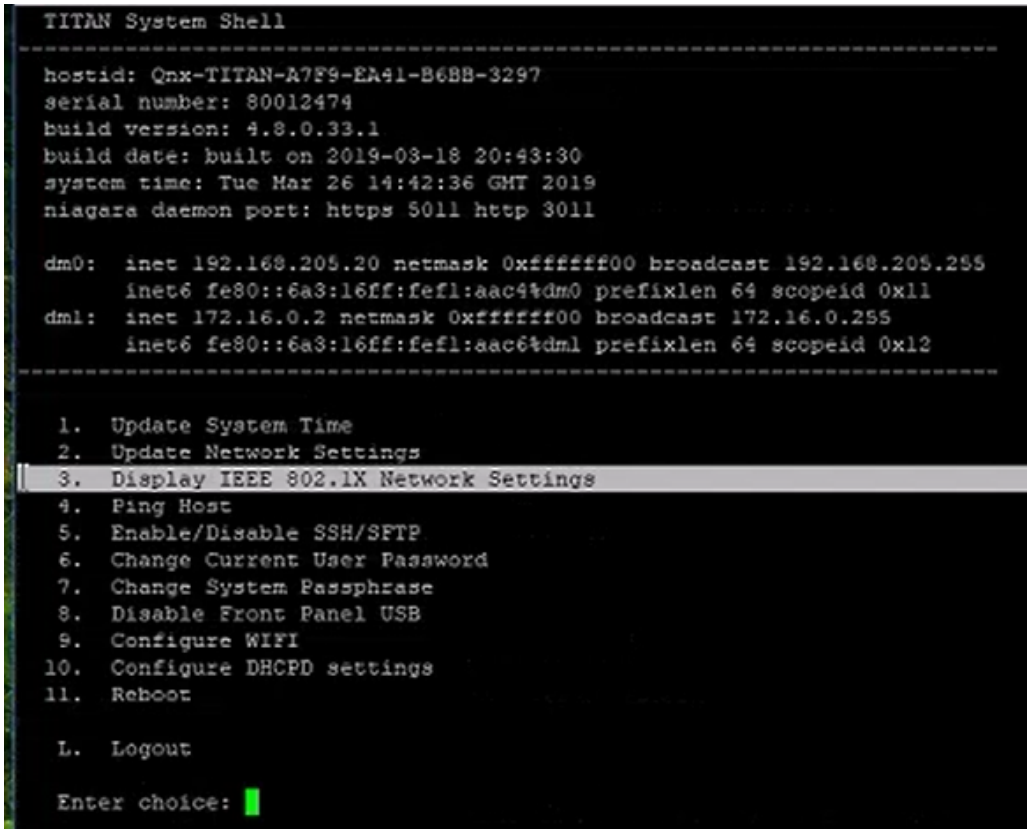
## Additional details in system shell

In the Niagara 4.8 System Shell, there is added support for IEEE 802.1X.

**NOTE:** Connection to the 802.1X network occurs at the time the platform boots. After configuring the primary adapter, you can use a serial connection to observe what happens in the boot log during the reboot. This can be useful for troubleshooting purposes.

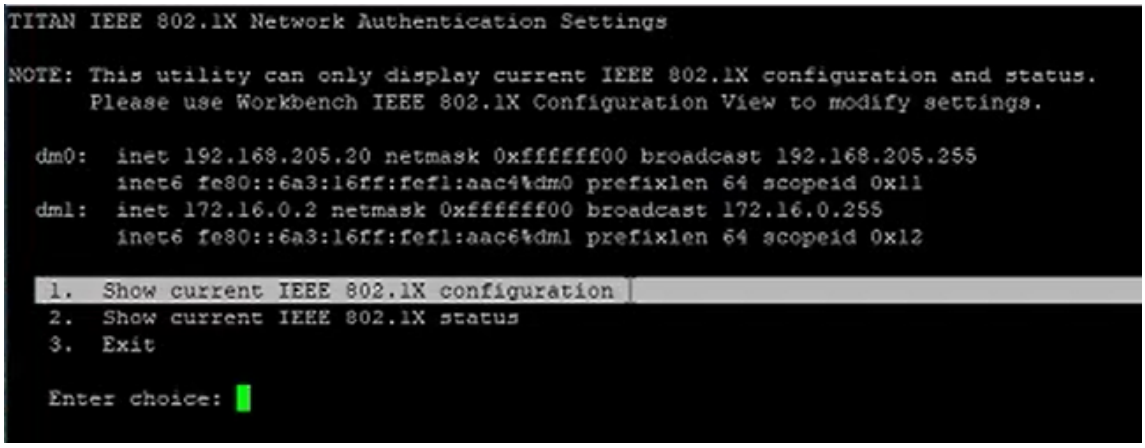
If working at the serial shell level, you have the option to display the IEEE 802.1X settings.

Figure 7 System Shell Menu with added option to display IEEE 802.1X settings



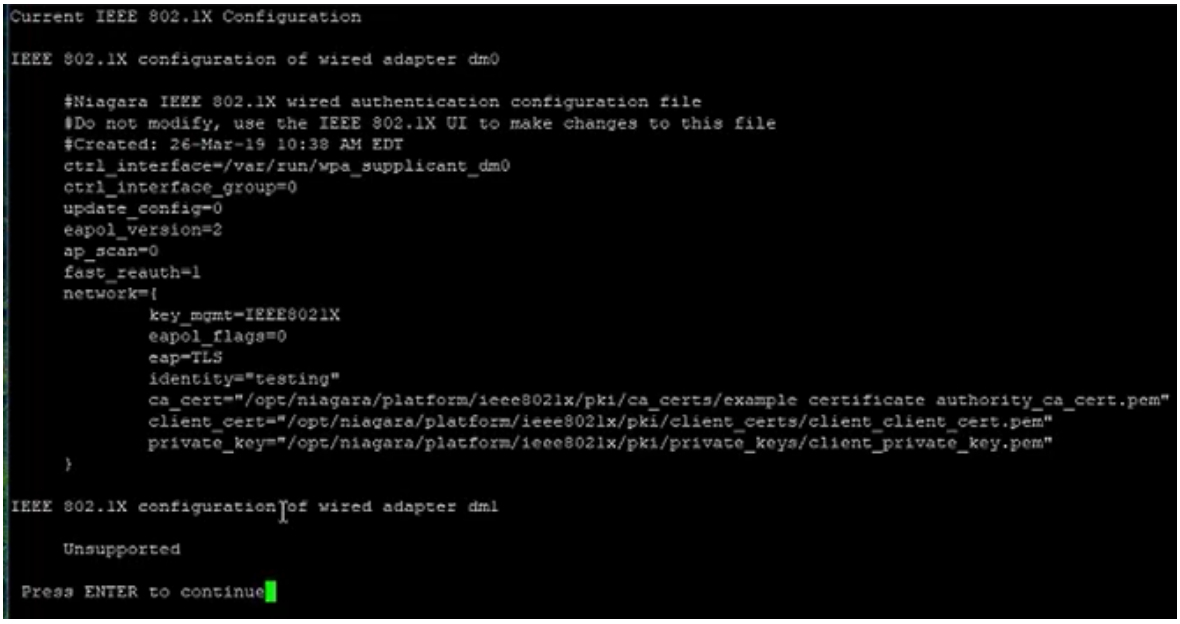

When you type 3 and press Enter, a secondary menu provides options to view the current configuration settings as well as view the current status.

Figure 8 System shell 802.1X status



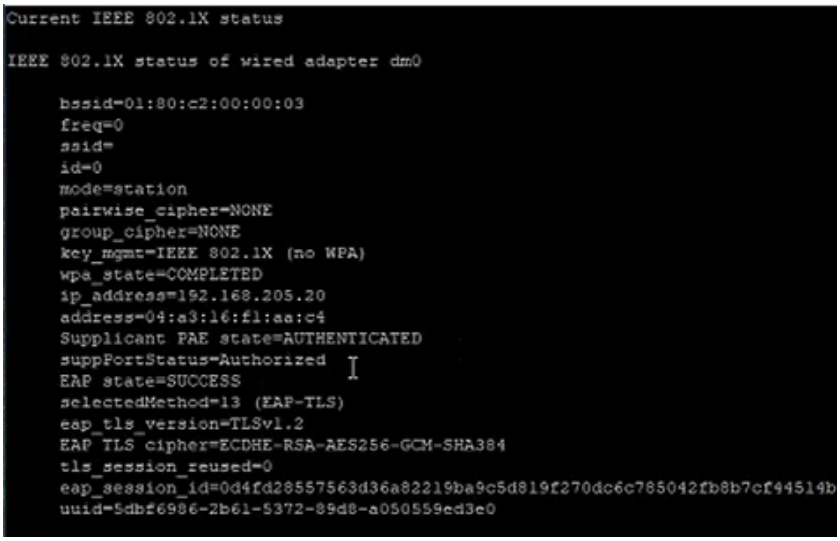

Option 1 shows the current IEEE 802.1X configuration file. This allows you to see the information that is displayed in the UI down at the serial level. It shows the current configuration file that is in use for the primary adapter, and that the secondary adapter does not support 802.1X connections.

Figure 9 Current IEEE 802.1X configuration file




Option 2 shows the status of the supplicant in use. This allows you to see detailed information about the primary adapter. In the following example it shows that the supplicant port state is “Authenticated” that the EAP state is “Successful”; it shows the kind of method that was used, “EAP-TLS”; and the version of that method, “TLSv1.2”.

Figure 10 Status of the supplicant in use




## Uninstall

To uninstall a module from a host, use the Software Manager to remove the module from the `!modules` folder, where `!` replaces the file path, and move the device to a port that does not require (or provide) IEEE 802.1X wired authentication.

# Chapter 4 About IEEE 802.1X Wired Authentication

**Topics covered in this chapter**

♦ How it works
♦ IEEE 802.1X and Niagara
♦ About Wired Authentication and SSL connectivity

The IEEE 802.1X standard provides a method to authenticate client devices that are physically connected a network. This type of authentication prohibits unauthorized connections via an Ethernet cable.

Secure networks require hardware (switches, bridges, gateways) located in a secure building, use encryption for wireless transmissions, and maintain strong access credentials (usernames and passwords). Even with these precautions, a would-be hacker could plug a cable directly in to a network switch port, receive an IP address, and capture all non-encrypted traffic on the network. The implementation of the IEEE 802.1X standard protects the network from just such an attack by requiring any connected device to successfully authenticate itself.

## How it works

IEEE 802.1X client authentication relies on an authentication server to validate or reject the client device credentials. While a full understanding of exactly how IEEE 802.1X authentication works is not necessary to use it, it may help during configuration and testing to have a general sense of what is occurring. For more detailed explanations, do an internet search.

IEEE 802.1X provides a way to authenticate the clients that are physically connected to a network. This type of authentication prohibits an unauthorized device (for example, a laptop operated with malicious intent) from connecting to a network.

Three actors play roles in the authentication drama:

- The *supplicant* is a device (computer, JACE, etc.) that connects to the network.
- The *authenticator* is the network port or WiFi access point to which the device connects.
- The *authentication server* is a host on the network (RADIUS server) that is capable of verifying the identity of the device.

Before a device is allowed to access any resource on the network, the supplicant's credentials must be authenticated. The authenticator forwards the supplicant's credentials to the authentication server. If the authentication server accepts the supplicant's credentials, the authenticator gives the supplicant access to network resources. Otherwise, the supplicant does not gain access.

## IEEE 802.1X and Niagara

In Niagara 4.8 and later, there is added support for IEEE 802.1X wired authentication on the JACE-8000 and Edge platforms. Support for the standard allows the JACE-8000/Edge and Supervisor platforms to join IEEE 802.1X authenticated networks.

The JACE/Edge devices licensed to use the IEEE 802.1X module contain a supplicant, and the means to configure the supplicant to communicate with the authenticator. The following components support the Niagara 4.8 IEEE 802.1X implementation:

- The supplicant, **wpa_supplicant**, negotiates the key with a WiFi Protected Access (WPA) authenticator and controls the roaming and IEEE 802.11 authentication/association of the WLAN driver. **wpa_supplicant** is designed to be a "daemon" program that runs in the background and acts as the back-end component controlling the wireless connection.

- Instances of the configuration file, `wpa_supplicant.conf`, allow users to configure the **wpa_supplicant**.

The `platIEEE8021X.jar` (-rt, -wb) module supports editing the configuration file, `wpa_supplicant.conf`.

## About Wired Authentication and SSL connectivity

The Niagara 4.8 implementation of the IEEE 802.1X standard for wired authentication makes use of Niagara's SSL technology. Client certificates must be signed by a recognized Certificate Authority. A signed CA certificate is required for IEEE 802.1X to work.

# Index