

Niagara^{AX} FIPS 140 Configuration Guide

Information and/or specifications published here are current as of the date of publication of this document. Tridium, Inc. reserves the right to change or modify specifications without prior notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia. Products or features contained herein are covered by one or more U.S. or foreign patents. This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc. Complete confidentiality, trademark, copyright and patent notifications can be found at: <http://www.tridium.com/galleries/SignUp/Confidentiality.pdf>. © 2013 Tridium, Inc.

Entrust is a registered trademark of Entrust, Inc. in the U.S. and other countries. In Canada, Entrust is a registered trademark of Entrust Limited. JACE, Niagara Framework, Niagara AX Framework and the Sedona Framework are trademarks of Tridium, Inc.

In AX-3.8, a NiagaraAX host can be configured to run its station in “FIPS Mode”, using only cryptographic algorithms supplied by a FIPS 140-2 certified cryptographic module. This document explains how FIPS is used in NiagaraAX, how to get FIPS mode up and running, and some special considerations both when deploying FIPS, and when developing for NiagaraAX to be used in a FIPS environment.

See the following topics for more details:

- “About FIPS and NiagaraAX” on page 1
- “FIPS licensing and software distribution” on page 2
- “Installing the FIPS distribution” on page 2
 - “Make FIPS available to Workbench” on page 2
 - “Install FIPS to remote AX-3.8 hosts” on page 2
 - “Install FIPS to local host for station usage” on page 4
- “About a station in FIPS mode” on page 5
 - “FIPS station startup messages” on page 5
 - “Verifying FIPS mode” on page 5
- “Special considerations” on page 6
 - “Web authentication and FIPS” on page 6
 - “Kerberos authentication (in LDAP) and FIPS” on page 7
- “Developers notes on FIPS” on page 7
 - “Disallowed algorithms” on page 8
- “Document change log” on page 8

About FIPS and NiagaraAX

Collectively, FIPS (Federal Information Processing Standard) refers to U.S. government standards regulations, where in particular, FIPS 140 governs the use of encryption and cryptographic services used by hardware and software. To meet FIPS 140 accreditation, cryptographic modules undergo a thorough certification process by NIST (National Institute of Standards and Technology) to ensure that all cryptographic algorithms adhere to the government security guidelines. The current version of FIPS 140 is version 2, widely known as FIPS 140-2.

One of the features introduced in NiagaraAX 3.8 is a FIPS 140-2 compliant mode for stations. When running in “FIPS mode”, stations only use cryptographic algorithms supplied by a FIPS-certified cryptographic module.

NiagaraAX’s FIPS feature employs the JCA (Java Cryptography Architecture), which allows cryptographic algorithms to be requested without relying directly on a specific security provider. Instead, requests for specific algorithms go through an ordered list of installed providers, selecting the algorithm from the first provider with an implementation. Additional security providers may be installed as needed; similarly, undesired providers may be removed.

In AX-3.8, all NiagaraAX requests for cryptographic algorithms go through the JCA.

- In a standard station (running on a host *not* configured for FIPS), all the Sun (Oracle) built-in providers are available, as well as the “BouncyCastle” provider. Cryptographic algorithms are selected

from any of these providers.

- However, a station running in FIPS mode, i.e. on a host configured for FIPS, has most Sun cryptographic providers and services *stripped out*. Instead, providers in the third-party, FIPS-approved cryptography module (from Entrust[®] Inc.) are installed. Since all cryptographic algorithm requests through the JCA are restricted to installed providers, only FIPS-compliant algorithms are used.

Note: *In order to upgrade legacy (pre-AX-3.8) stations, and because of certain required Java core functions, a small number of non-FIPS approved algorithms are still available. These are listed in this document's "Developers notes on FIPS" subsection, "Disallowed algorithms" on page 8.*

NiagaraAX developers should note that although these algorithms are accessible through JCA calls, their use is not allowed in a FIPS environment—unless (for example) used to upgrade legacy systems.

FIPS licensing and software distribution

FIPS support in AX-3.8 is a *licensed feature*. In order for a station to run in FIPS mode, its NiagaraAX host platform must be licensed with the feature "fips140-2". An example license line is below.

```
<feature name="fips140-2" expiration="2014-09-13" lib="entrust" parts="PROTO-FIPS"/>
```

Along with this host licensing requirement, FIPS software must be separately installed on any host platform that is already running AX-3.8. The installation method varies between remote (typically JACE) platforms and local (e.g. Supervisor) platforms.

Note for any type of AX-3.8 host, all software to implement the FIPS feature is contained in a single NiagaraAX distribution (.dist) file:

```
entrust-fips-generic.dist
```

Obtain this file through your sales channel to make available on your AX-3.8 Workbench PC. Once you have this file, see ["Installing the FIPS distribution"](#).

Installing the FIPS distribution

The following procedures provide step-by-step details, and apply to AX-3.8 systems only:

- ["Make FIPS available to Workbench"](#)
- ["Install FIPS to remote AX-3.8 hosts"](#)
- ["Install FIPS to local host for station usage"](#)

Make FIPS available to Workbench

Do the following to make the FIPS distribution available to your AX-3.8 Workbench.

Note: *Prerequisite: Get the FIPS distribution file. See ["FIPS licensing and software distribution"](#).*

Making FIPS available to Workbench


- Step 1 If AX-3.8 Workbench is running, save all work and exit Workbench.
- Step 2 Copy the FIPS .dist file into your *niagaraHome/sw/inbox* subdirectory.
- Step 3 Open (or restart) AX-3.8 Workbench.

Install FIPS to remote AX-3.8 hosts

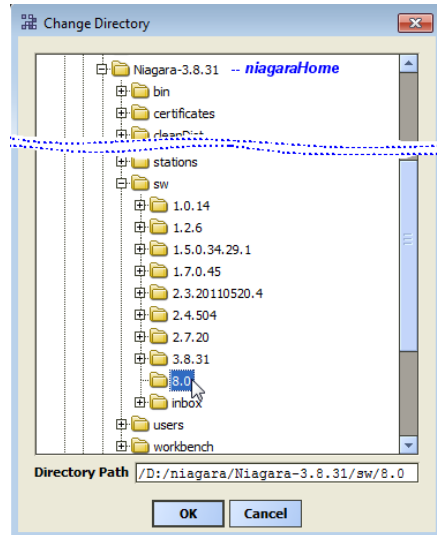
Do the following to install the FIPS distribution to remote AX-3.8 hosts (typically JACEs).

Note: *Prerequisite: Have the FIPS distribution file in Workbench. See ["Make FIPS available to Workbench"](#).*

Installing FIPS to a remote AX-3.8 host (e.g. JACE)

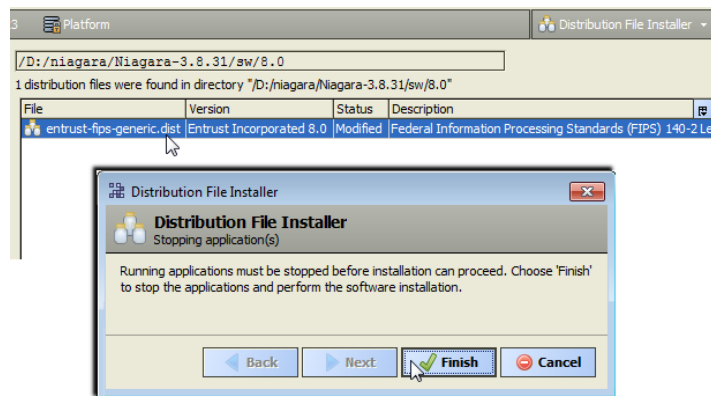
- Step 1 Using AX-3.8 Workbench, open a platform connection to the remote host.
- Step 2 If you have purchased a FIPS license upgrade for this host, but not installed it yet, you can do that now in this same platform session (*before* installing the FIPS distribution).
If the license is already installed, or you wish to do this later, go to [Step 4](#).
- Step 3 Go to the platform **License Manager** view for the remote host, and install the updated license.
Note: *Answer "No" when the **Licensing Complete** dialog appears asking to "Reboot now?" The necessary reboot happens automatically later, after you complete the next steps.*
- Step 4 Go to the platform **Distribution File Installer** view for the remote host.
Click the  **Choose Directory** button at the bottom of the view.

Step 5 In the **Change Directory** dialog, navigate to and expand your *niagaraHome/sw* folder.



Select the *niagaraHome/sw/8.0* folder.

Step 6 In the **Distribution File Installer**, select the *entrust-fips-generic.dist* file, then click **Install**.



As shown above, a confirmation popup appears. Click **Finish** to start the installation.

An **Installing Distribution** popup shows you the installation progress, where the host reboots when installation is complete. Click **Close**.

Any station started after the reboot will be running in FIPS mode. See [“About a station in FIPS mode”](#) on page 5.

Note: As for any AX-3.8 station, if browser client access is needed using “Web Workbench” (*wbApplet*), any browser client PC needs Java “Unlimited Strength Policy Files” installed for this to work. For related details, see [“Additional client-side Java installation steps”](#) in the NiagaraAX 2013 Security Updates document.

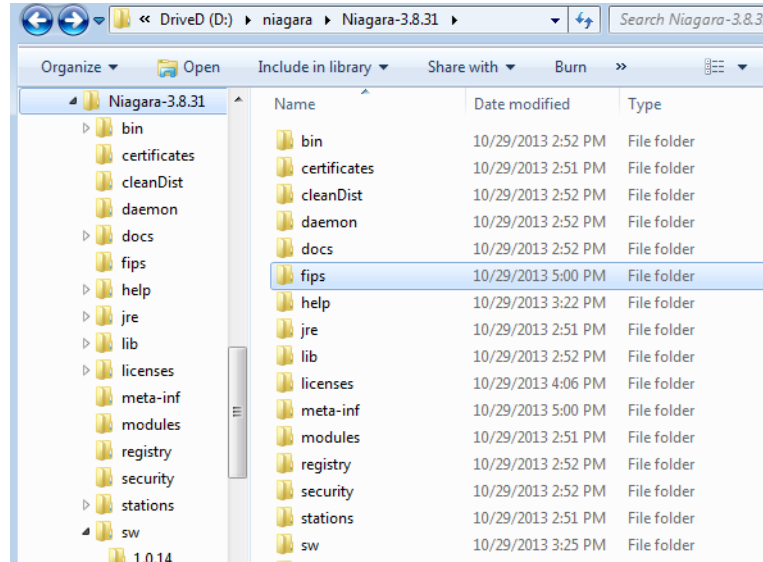
Install FIPS to local host for station usage

Do the following to make the FIPS distribution available to a station running on your local host (PC).

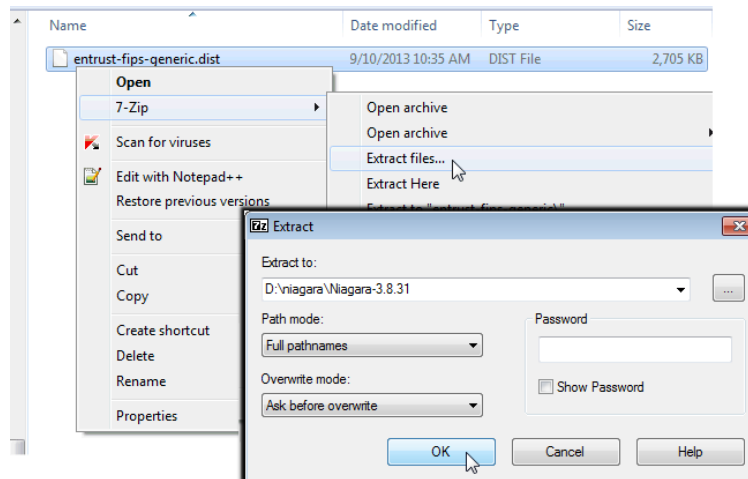
Note: Prerequisite: Get the FIPS distribution file. See “[FIPS licensing and software distribution](#)” on page 2.

Installing FIPS to local host for station usage

- Step 1 Extract the files in the `entrust-fips-generic.dist` directly into your `niagaraHome/` folder. This should create a new “fips” subfolder.



Note: A NiagaraAX `.dist` file is like a zip archive file, where if necessary you can change the file extension from `.dist` to `.zip` in order to extract its files using Windows Explorer or a utility like “WinZip”. Note that some archive utilities like “7-Zip” recognize `.dist` files as archives, such that you can extract files directly.



- Step 2 If a station is running on your local host, *restart* it. Any station started after the reboot will be running in FIPS mode. See “[About a station in FIPS mode](#)” on page 5.

Note: As for any AX-3.8 station, if browser client access is needed using “Web Workbench” (`wbApplet`), any browser client PC needs Java “Unlimited Strength Policy Files” installed for this to work. For related details, see “[Additional client-side Java installation steps](#)” in the NiagaraAX 2013 Security Updates document.

About a station in FIPS mode

If the host platform is licensed for FIPS and has the FIPS distribution installed, then any station started on that platform runs in FIPS mode, without any additional steps needed.

See the following for more details:

- [“FIPS station startup messages”](#)
- [“Verifying FIPS mode”](#)

FIPS station startup messages

Upon station startup, station output (visible in the host platform’s **Application Director** view) indicates FIPS status with *one* of the following messages:

- **FIPS providers successfully loaded.**
This indicates the station has successfully started in FIPS mode.
MESSAGE [13:21:23 30-Oct-13 EDT] [sys.registry] Loaded [669ms]
MESSAGE [13:21:59 30-Oct-13 EDT] [sys] FIPS providers successfully loaded.
- **FIPS module is present but FIPS is not licensed.**
This indicates that the FIPS providers were found, but the host platform is not licensed for FIPS. The station was *not* started in FIPS mode.
MESSAGE [13:21:23 30-Oct-13 EDT] [sys.registry] Loaded [669ms]
MESSAGE [13:21:59 30-Oct-13 EDT] [sys] FIPS module is present but FIPS is not licensed.
To fix this, update the host’s license to include the “fips104-2” feature.
- **FIPS is licensed but FIPS module is not present.**
This indicates the host platform is licensed for FIPS, but the FIPS providers could not be found. The station was *not* started in FIPS mode.
MESSAGE [13:21:23 30-Oct-13 EDT] [sys.registry] Loaded [669ms]
MESSAGE [13:21:59 30-Oct-13 EDT] [sys] FIPS is licensed but FIPS module is not present.
To fix this, ensure that FIPS was correctly installed. See [“Installing the FIPS distribution”](#) on page 2.

Note that a station running in FIPS mode *does not*, for the most part, *appear to run differently* from a station not in FIPS mode. Although different FIPS-compliant cryptographic algorithms are used in a FIPS station, this should not result in behavioral changes. A few exceptions to this follow:

- SSL version 3.0 is not permitted with FIPS, which requires a minimum of TLS version 1.0. In order to ensure FIPS compliance, the “Https Min Protocol” property of the station’s WebService is automatically set to TLSv1 and made read-only when a station is in FIPS mode. Similarly, the “Foxy Min Protocol” property in the station’s FoxService is also set to TLSv1 and is read-only.
The platform’s protocol in “Platform SSL Settings” is also set to TLSv1 and made read-only—see this via the **Platform Administration** view, “Change SSL Settings”. (Note although FIPS mode is not claimed on platform connections, for the most part, only FIPS-compliant algorithms are used).
- Since FIPS does not permit SSL version 3.0, the list of allowed ciphers suites has been reduced in FIPS mode to include only TLS version 1.0 cipher suites.
- Any functionality relying on a non-FIPS compliant algorithm will no longer work in a station running in FIPS mode.

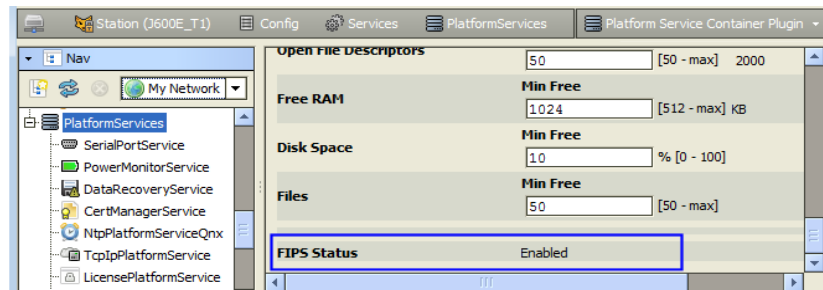
Verifying FIPS mode

Because a station running in FIPS mode demonstrates few behavioral differences from a station not running in FIPS mode, you have several ways to check the FIPS status of a station at any time. These are in *addition* to the station startup messages (see [“FIPS station startup messages”](#) on page 5).

Use the following methods to verify a station’s FIPS status:

- The station’s **PlatformServices** container has a new “FIPS Status” property, found at the bottom of the service’s default view (Platform Services Container Plugin). See [Figure 1](#).

Figure 1 FIPS Status parameter in station's Platform Service Container Plugin (at bottom)

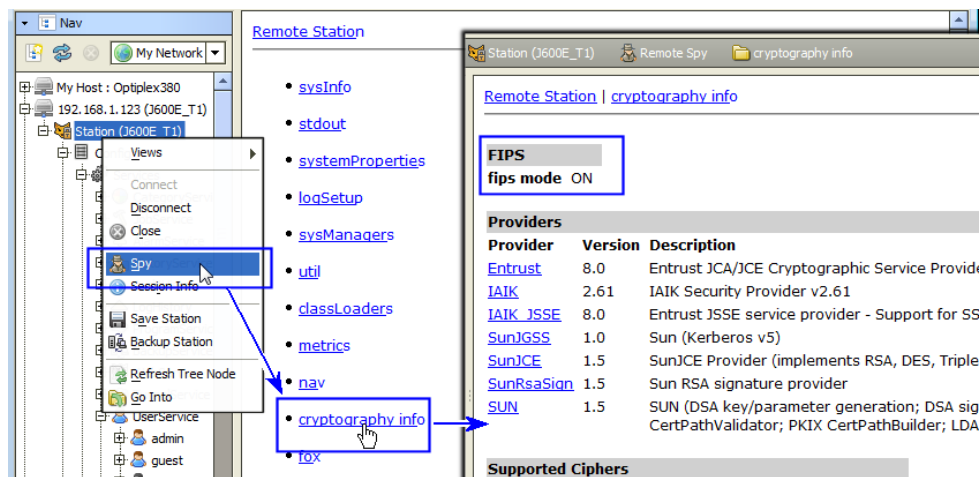


Note: This property is not present unless the host is licensed for FIPS or the FIPS distribution has been installed.

The FIPS Status property has one of the following values:

- Enabled — Indicates the station is running in FIPS mode.
- Disabled (Licensed/JAR Not Installed) — Indicates the host platform is correctly licensed, but the FIPS distribution can not be found. The station is *not* running in FIPS mode.
- Disabled (JAR Installed/Not Licensed) — Indicates the FIPS distribution was correctly installed, but the host platform is not licensed for FIPS. The station is *not* running in FIPS mode.
- Providing that the platCrypto module is installed in the host, the station's "spy page" includes a "cryptography info" link you can view. See Figure 2.

Figure 2 Station's spy page includes "cryptography info" page with FIPS status and provider details



- As shown, the "fips mode" entry displays "ON" if the station is running in FIPS mode, and "OFF" otherwise. In addition, a station running in FIPS mode has the Entrust, IAIK and IAIK_JSSE providers listed, as shown in Figure 2.

Special considerations

There are a few special considerations when using FIPS, say from a transition from "non-FIPS" to FIPS or when using LDAP with Kerberos authentication. See the following sections for more details:

- "Web authentication and FIPS"
- "Kerberos authentication (in LDAP) and FIPS"

Web authentication and FIPS

When using the "Cookie Digest" authentication scheme in a station's **WebService**, the required client-side cryptography is supplied by Javascript libraries, rather than the JCA security providers—whether the station is running in FIPS mode or not. These Javascript libraries are *not* FIPS compliant.

Therefore, if FIPS compliance is strictly required, you must change the "Authentication Scheme" property of the station's **WebService** to "Cookie", which does not use *any* cryptography (and therefore cannot use any non-FIPS compliant cryptographic algorithms). However, because the "Cookie" authentication scheme sends a user's credentials to the station in clear text, you should *not* use this method without also configuring the station's WebService for HTTPS (TLSv1).

Kerberos authentication (in LDAP) and FIPS

In addition to “FIPS mode” ability, AX-3.8 also introduces support for Kerberos authentication when the station uses an LDAP (or Active Directory) user service. For complete details, refer to the *NiagaraAX LDAP/Active Directory Configuration Guide*.

If a need for both Kerberos and FIPS with NiagaraAX arises, keep the following in mind:

- At the time of this document, the combination of LDAP with Kerberos along with FIPS is *untested*.
- Because a FIPS mode station can only use FIPS-compliant algorithms, the LDAP and Kerberos servers must also support FIPS algorithms. This is a known *problem* for all versions of Windows Active Directory, which support only DES and RC4 (neither of which are FIPS-compliant algorithms).
- Therefore, Kerberos together with FIPS is not possible without meeting these requirements:
 - The LDAP and Kerberos servers must support either 3DES *or* AES. If the system includes Hotspot QNX-based JACE platforms (JACE-3/6/7 series), only 3DES can be used.
 - NiagaraAX hosts must support Kerberos (AX-3.8 “Hotspot JVM” platforms only, note that “J9 JVM platforms, i.e. JACE-2/4/5 series, do not support Kerberos authentication).
- In order to use Kerberos and FIPS, it may be necessary to enable the use of stronger encryption on the Kerberos server. This is something you would typically need to have done by the Kerberos administrator at the installation site.

- In order to ensure that only FIPS algorithms are used when doing Kerberos authentication, Workbench can be set up to request only certain specific FIPS encryption types. You do this by editing the `krb5.conf` file, described in the *NiagaraAX LDAP/Active Directory Configuration Guide*.

Add the following lines to the `[libdefaults]` section of this file, to restrict which encryption types are allowed by a client:

```
[libdefaults]
default_tkt_enctypes = aes256-cts aes128-cts des3-cbc-sha1
default_tgs_enctypes = aes256-cts aes128-cts des3-cbc-sha1
permitted_enctypes = aes256-cts aes128-cts des3-cbc-sha1
```

These entries will restrict the ciphers used to AES-128, AES-256, or 3DES. Note that AES-128 and AES-256 are not supported on a QNX-based JACE platform, which must use the (last) 3DES cipher

Developers notes on FIPS

Note: *These notes are intended for advanced NiagaraAX developers only, and may be moved at a later date to another document with similar type content.*

When developing code intended for use in a FIPS environment (station running in FIPS mode), you *must* take steps to ensure that *no* “non-FIPS compliant” cryptographic algorithms are used.

In order to simplify writing code for FIPS environments, we have made use of the JCA (Java Cryptography Architecture). In the JCA, security providers are added to or removed from the framework as needed. Different providers may implement different cryptographic algorithms, or they may provide different implementations of the same algorithms. Requests for specific algorithms are then made through the JCA.

For example, if an AES-256 cipher is needed, you can call:

```
Cipher cipher = Cipher.getInstance("AES256");
```

Although it is possible to use `Cipher cipher = Cipher.getInstance("AES256", "Entrust");` to request a cipher from a specific provider, this should almost never be used. Because we use different providers for FIPS-mode and non-FIPS mode, requesting a specific provider can result in code that only works in one environment.

In the JCA, security providers are ordered; when a request for an algorithm is made, the JCA goes through the ordered list of providers and returns the first implementation it finds. In our implementation of FIPS, we have ensured that the FIPS providers are always first in the list. Therefore, FIPS-compliant algorithms will always be selected when possible.

In addition, when running in FIPS mode, most non-FIPS algorithms have been stripped out from the security providers (exceptions are listed in the Disallowed Algorithms section). This ensures that any request for a non-FIPS algorithm will generate an exception, so that no non-FIPS cryptographic algorithm usages are inadvertently introduced.

When writing code for a FIPS environment, all cryptographic algorithms requests must go through the JCA. Also see “[Disallowed algorithms](#)” on page 8.

Disallowed algorithms

Although most non-FIPS compliant algorithms are stripped out of the JCA security providers when running in FIPS mode, it is necessary for a small subset of non-FIPS algorithms to remain. In some cases, this is for compatibility with older systems (e.g. to decrypt old BOG files). In other cases, Java needs specific non-FIPS algorithms, for example to load and verify security providers.

- **Ciphers**
 - Blowfish
- **Macs**
 - HMAC/MD5
 - HmacMD5
- **Message Digests**
 - MD5
 - SSL3-SHAMD5
- **Signatures**
 - MD5withRSA

These non-FIPS algorithms are listed above. Although these algorithms are available for use through JCA calls, these should never be used—*unless* it is to upgrade an older system that uses non-FIPS algorithms. For example, you could decrypt using the Blowfish cipher, but you could not encrypt with it.

See “[Developers notes on FIPS](#)” on page 7 for related information.

Document change log

Updates (changes/additions) to this *NiagaraAX FIPS 140 Configuration Guide* document are listed below.

- Publication: November 6, 2013
Initial document, which applies to AX-3.8 only.