# HTTPS Certificate Rotation

Sam Menown - 2023-10-19 - Comments (0) - Autoform DM FAQs

This article contains information and guidance on how to rotate an expired / expiring or temporary HTTPS certificate.

## Pre-requisites

- A JKS or PFX Keystore containing a new, valid certificate and private key along with the alias & password for the Keystore.

## Introduction

From Autoform DM v9 onwards, both standalone and clustered deployments are now set up with HTTPS as standard. During installation, you are given the opportunity to provide a custom certificate for your server, or if you do not have one available, the installer can generate a temporary certificate in order to speed up deployment time as well as configure your system for HTTPS communication.

The temporary certificate will remain valid for one year from the creation date, after which time it will need to be rotated out for a new, valid certificate. Whilst we recommend that you replace any temporary certificates as soon as possible after installation with custom certificates, we appreciate that in some circumstances, a temporary certificate is sufficient for some user's needs (e.g. test environments).

The process for rotating your expired, expiring or temporary certificate for both standalone and clustered servers is detailed below.

## Installation

The process of rotating your HTTPS certificate is straightforward and should not take more than a few minutes. As noted in the pre-requisites you will need your new Keystore and access to the node.properties file for your server installation. This Keystore will need to

contain both the new certificate and the private key and can be either in JKS format or PFX format (typically a certificate exported from a Windows environment will in PFX format).

For standalone and clustered installations, both the Keystore and node.properties are located alongside each other.

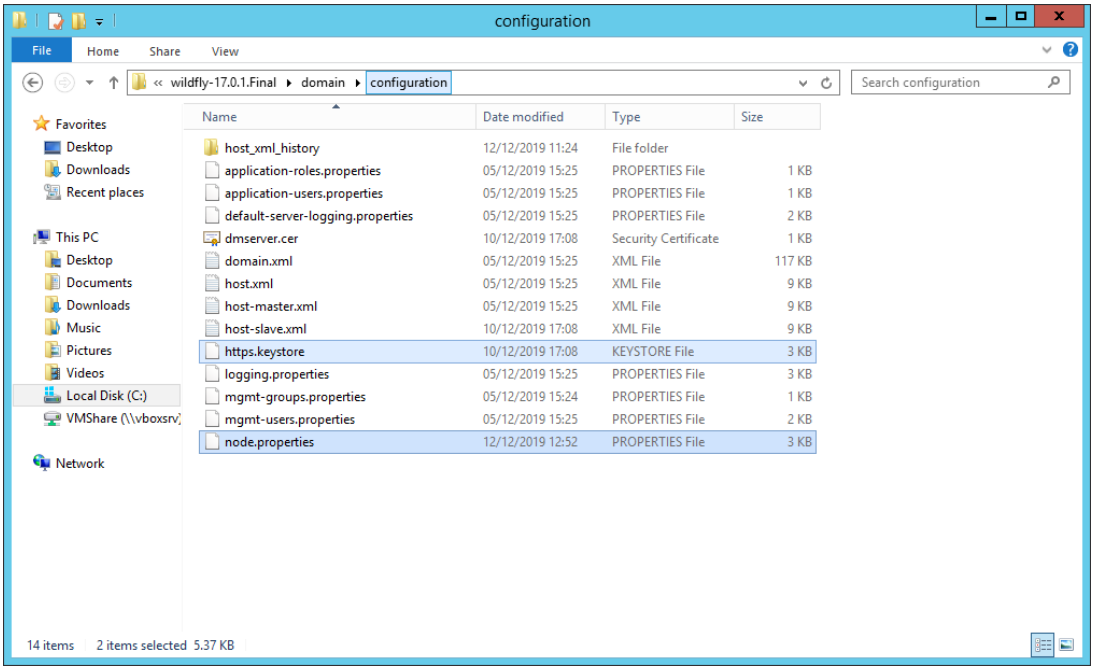
With a clustered installation, the certificate rotation must be carried out on the load balancer.
Example:

**Standalone Installation**

`C:\Program Files\EFS Technology\AUTOFORM DM\Server_9\wildfly-x.x.x.Final\standalone\configuration\`

**Cluster Installation (Load Balancer Path)**

`C:\Program Files\EFS Technology\AUTOFORM DM\Server_9 - Cluster_<cluster name> - Role_LB - <node name>\wildfly-x.x.x.Final\domain\configuration\`




**Keystore**

Autoform DM is set to look for a Keystore called `https.keystore` in the above folder. Therefore in order to carry out the rotation, follow these steps:

1. Rename the existing (old) file (for example https.expired), then locate your new Keystore and copy it into the appropriate directory, depending on your installation type.

2. Rename your new Keystore to `https.keystore`.

**node.properties**

Once you have copied and renamed your keystore, you need to update the existing keystore alias and password to match those for your new keystore. These credentials are stored in the node.properties file which is located in the same directory.

1. Open the file with a txt / xml editor and update the values for `https.keystore.alias` and `https.keystore.password` (see example below):

Example

```
#These values should be updated if switching to use your own
certificates/keystore from an auto-generated one
# Certificate alias and password for the keystore
# https.keystore.alias=dm_generated_cert
# https.keystore.password=dmcert

 https.keystore.alias=examplealias
 https.keystore.password=Securepassword
```

2. Save your file, then restart the Autoform DM service for the changes to take effect. For a clustered deployment, you only need to restart the Load Balancer service.

Your new certificate should now be active for your standalone / clustered deployment.

**[Optional] Validate Keystore**

If you have forgotten your Keystore alias or if you want to validate the Keystore, use the command below which will prompt for your password:

```
\jdk-11.0.4_11\bin\keytool.exe -list -keystore \path\to\keystore
```

This will list the alias for the certificate and verify the Keystore format. Please note that the listed certificate entry should be of type `PrivateKeyEntry`.