

---

# IAG Mirth Extension Deployment Guide

VERSION 1.0

---



## Copyright

*This document is CitiusTech Confidential and contains proprietary information, including trade secrets of CitiusTech. Neither the document nor any of the information contained in it may be reproduced or disclosed to any unauthorized person under any circumstances without the express written permission of CitiusTech.*

## Revision History

Document Version #	Revision Date	Prepared By	Approved By	Approval Date	Summary of Changes
1.0	01-07-2020	Akshaya Subramanian	Gaurav Chonkar	26-08-2020	First Version



## Contents

<b>1</b>	<b>PURPOSE .....</b>	<b>4</b>
<b>2</b>	<b>DEPLOYMENT REQUIREMENTS.....</b>	<b>5</b>
2.1	SERVER REQUIREMENTS .....	5
2.2	INSTALLATION PRE-REQUISITES.....	5
2.3	CLIENT ENVIRONMENT REQUIREMENTS.....	5
2.4	DEPLOYMENT USER REQUIREMENTS .....	5
<b>3</b>	<b>DEPLOYMENT STEPS .....</b>	<b>6</b>
3.1	VERIFYING THE CONTENTS OF THE ZIP FILE .....	6
3.2	IMPORTING THE CERTIFICATES.....	6
3.2.1	<i>For Windows</i> .....	6
3.3	INSTALLING THE EXTENSION.....	16
3.4	ANNEXURE .....	18



# 1 Purpose

This document provides guidelines to import the GIT Plugin for Mirth Connect.



## 2 Deployment Requirements

The following sections enlists the pre-requisites for the deployment:

### 2.1 Server Requirements

The GIT Mirth Extension is primarily a zip file that will get imported using the Mirth UI for Extensions.

The process to install the extension is similar to the other commercial extensions made available by Mirth Connect. Mirth User Guide can be referred for installation/uninstallation of the extension.

<Screenshot for Mirth Extensions Page>

### 2.2 Installation Pre-requisites

This section refers to the pre-requisites that would be required for the installation of the extension:

- Ensure GIT repository is available to connect to
- Ensure users have commit and push permissions to the GIT repository
- Ensure user has Mirth administration rights to be able to restart Mirth service, post import of the ZIP file
- The trusted certificate needs to be installed in the system's trust store

### 2.3 Client Environment Requirements

There is no specific requirements as this would reside in the Mirth Connect set up.

### 2.4 Deployment User Requirements

- The user needs to be a Mirth system administrator for installation and import of the ZIP file
- The user needs to import the certificate shared along with the ZIP file







## 3 Deployment Steps

Download the Mirth Extension zip from

<https://github.com/SalmanCitiustech/Mirth-Connect-Extension---GIT/tree/GitPluginZip>

### 3.1 Verifying the Contents of the ZIP File

The GIT plugin ZIP file would consist of the following:

Name	Type
 gitplugin-client	Executable Jar File
 gitplugin-server	Executable Jar File
 gitplugin-shared	Executable Jar File
 plugin	XML Document

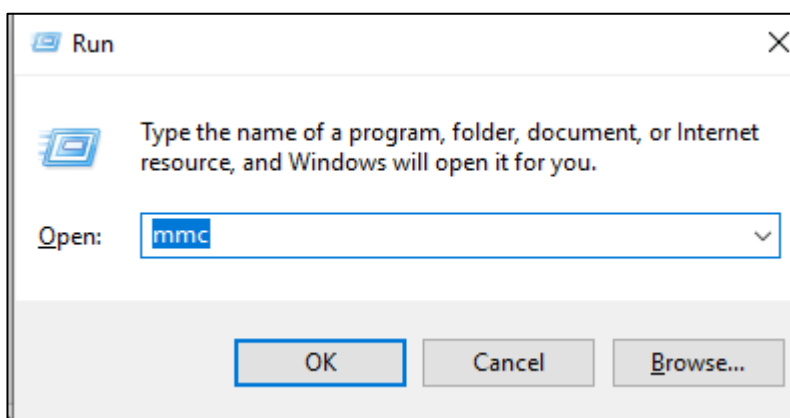
### 3.2 Importing the Certificates

Download the certificate shared along with the ZIP file and copy to some location on the Mirth server.

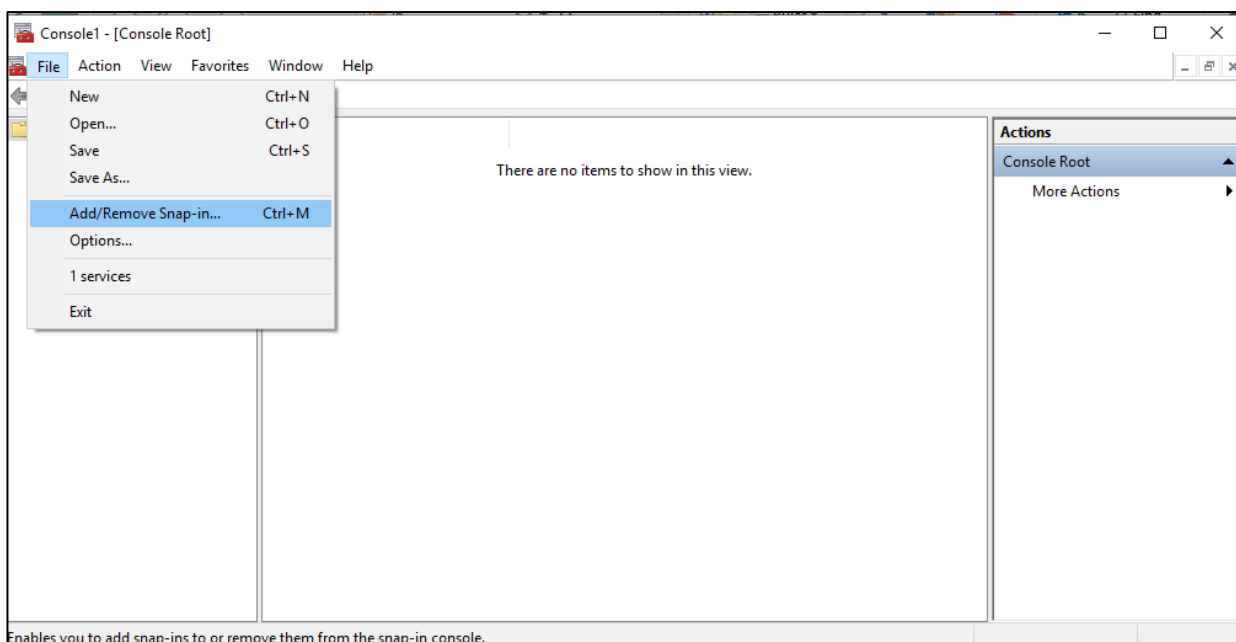
#### 3.2.1 For Windows

Start the MMC and perform the following steps:

1. Go to Windows, open the Run application and type **mmc**:

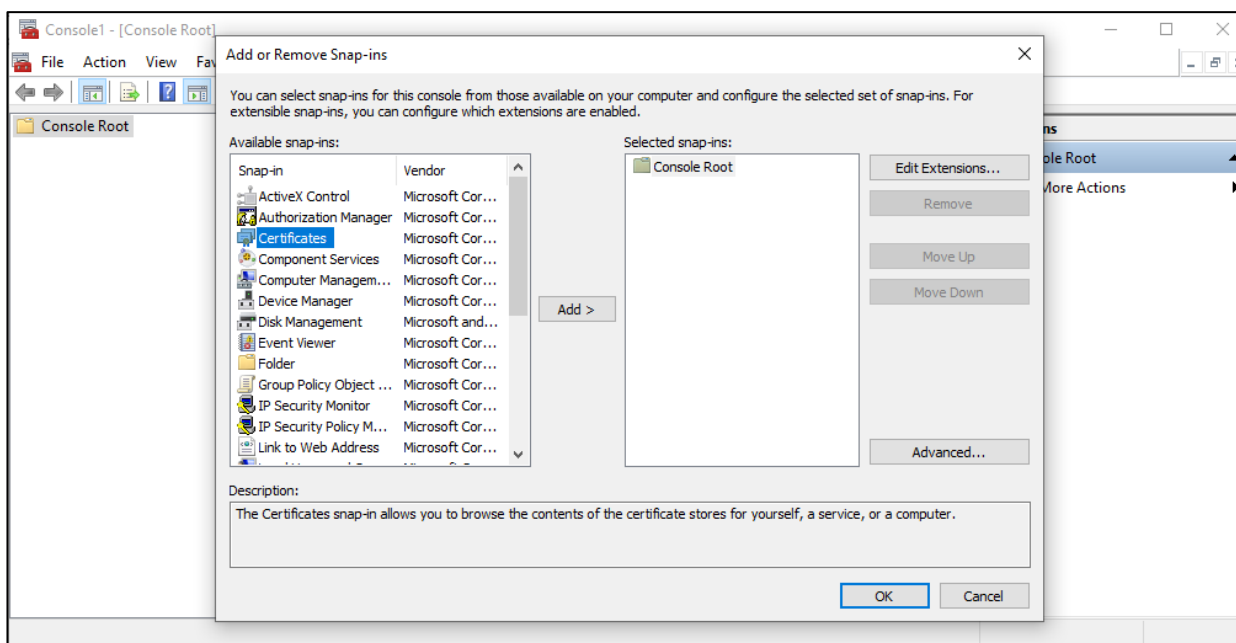


2. On the Console, click **File | Add/Remove Snap-in**:

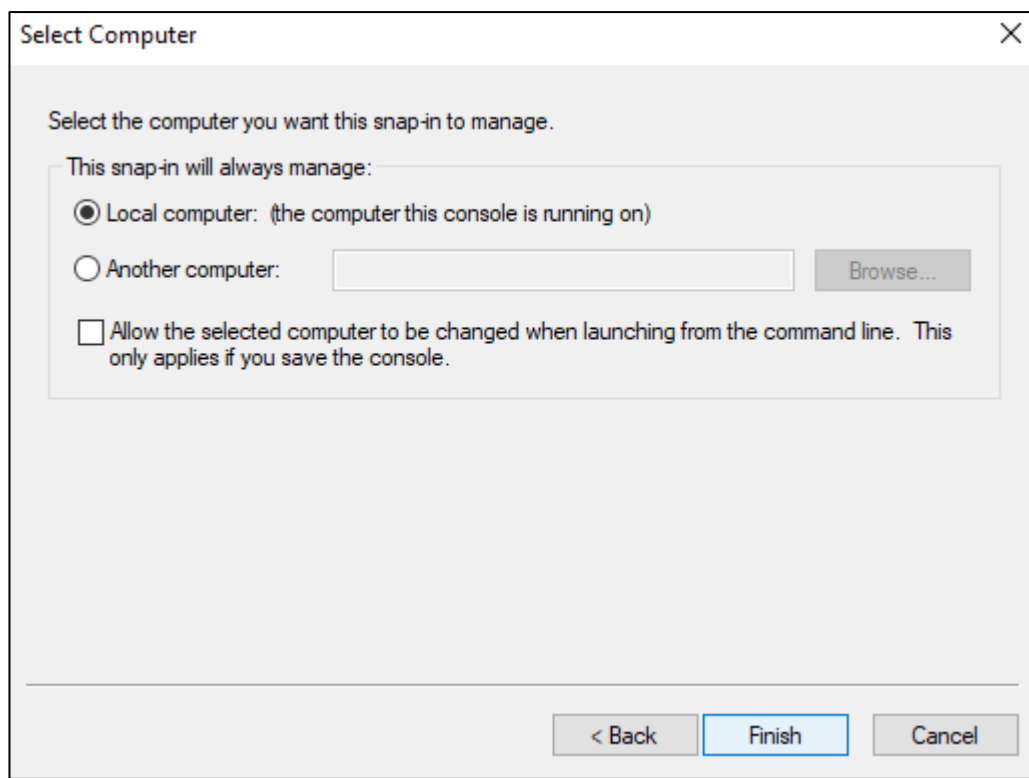


Enables you to add snap-ins to or remove them from the snap-in console.

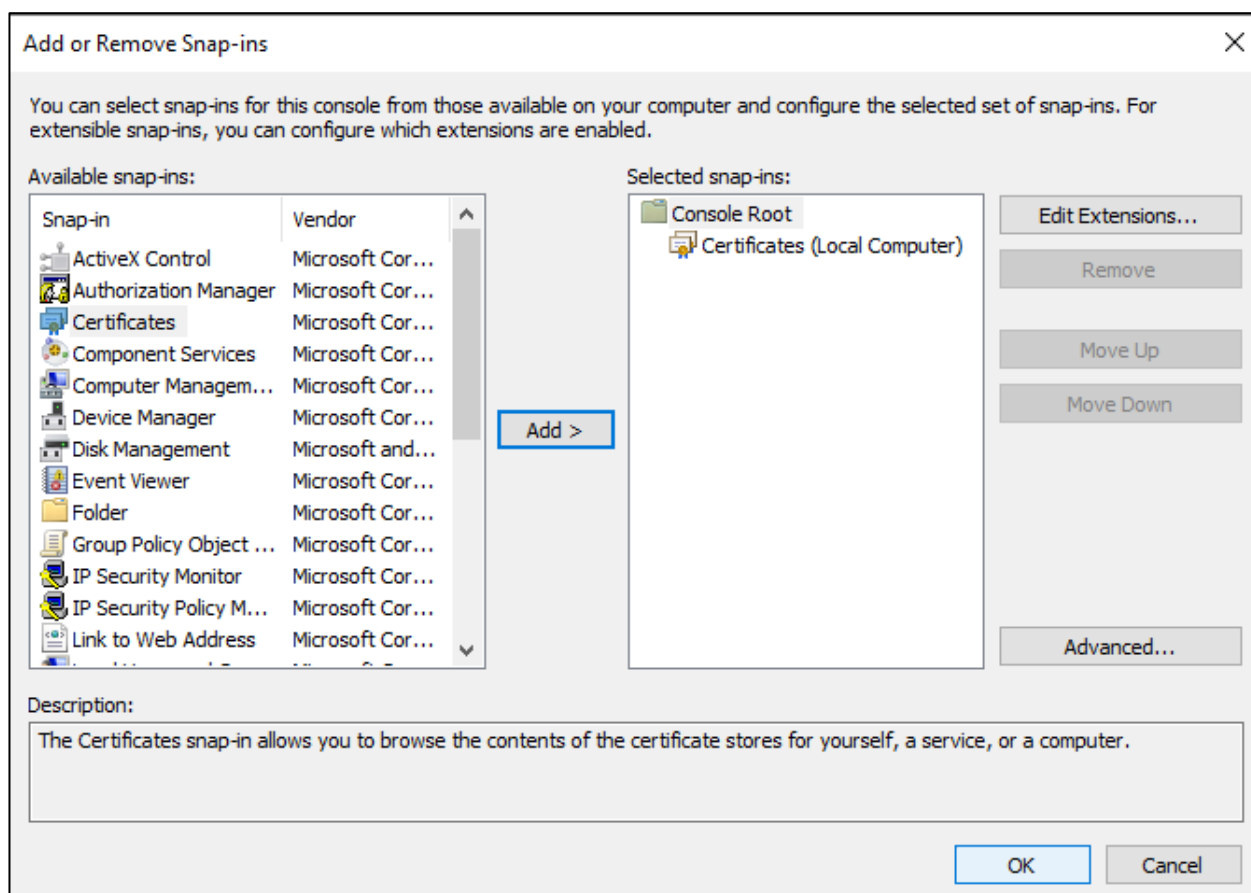
3. Select **Certificates** and click **Add**:



4. Select the **Service Account** or the **Computer Account**:

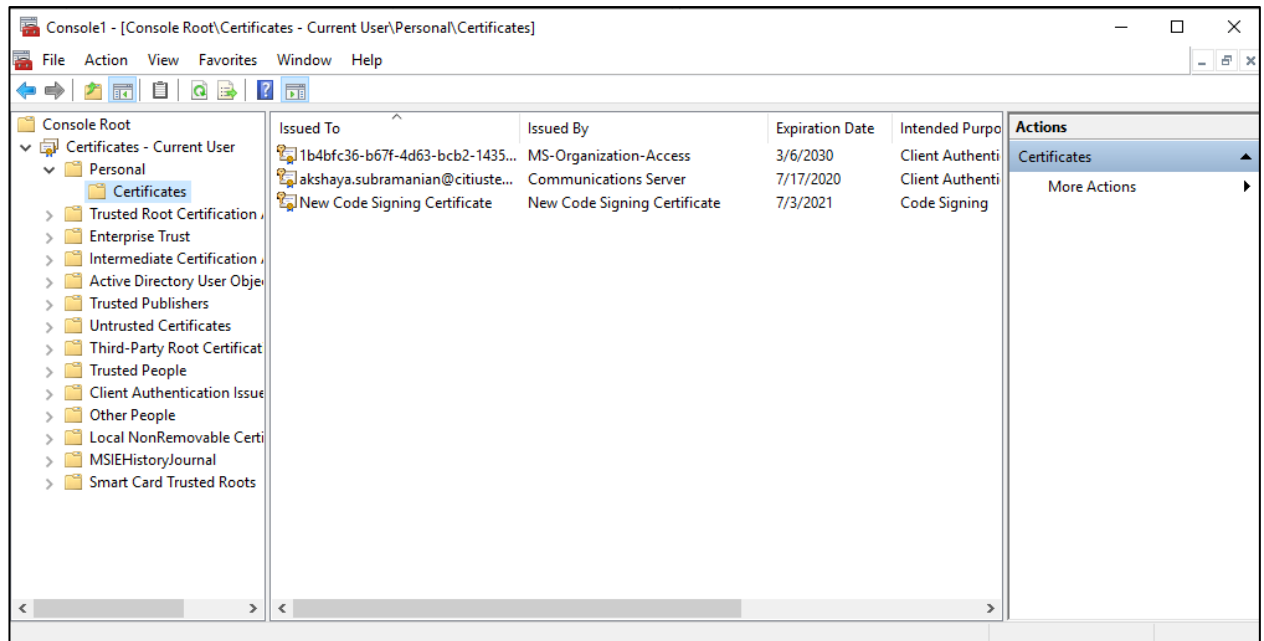


5. Click **OK**:

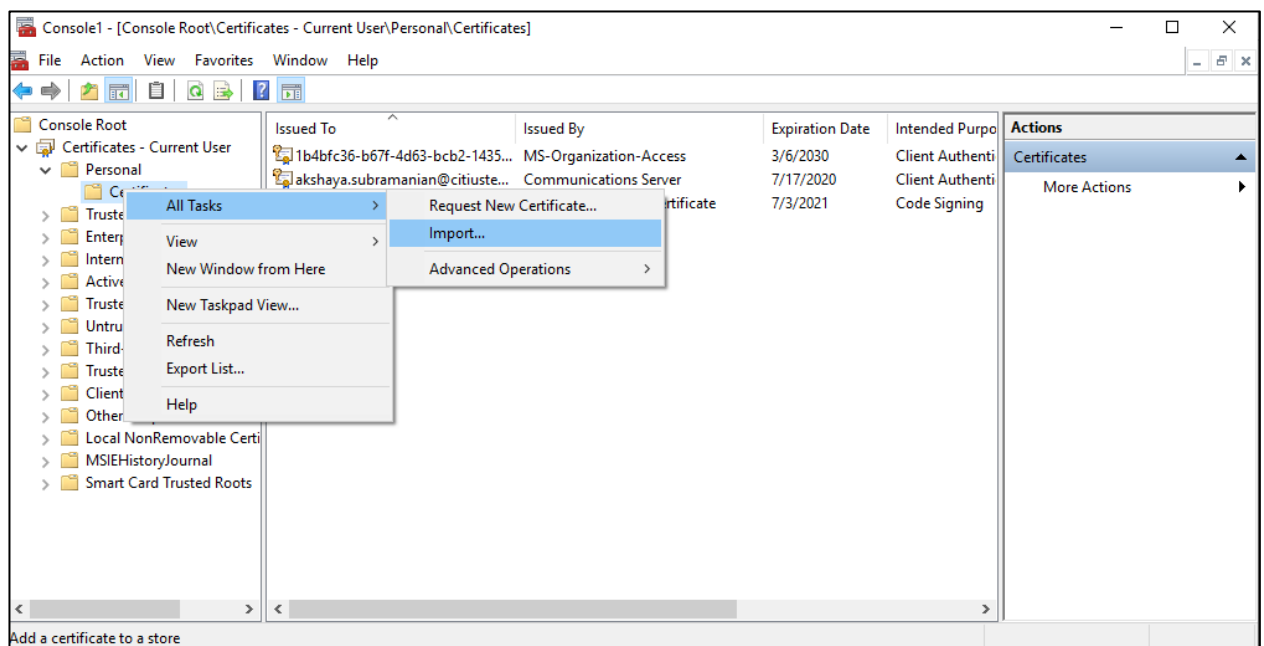


6. Select **Certificates** -> **Personal** -> **Certificates**:



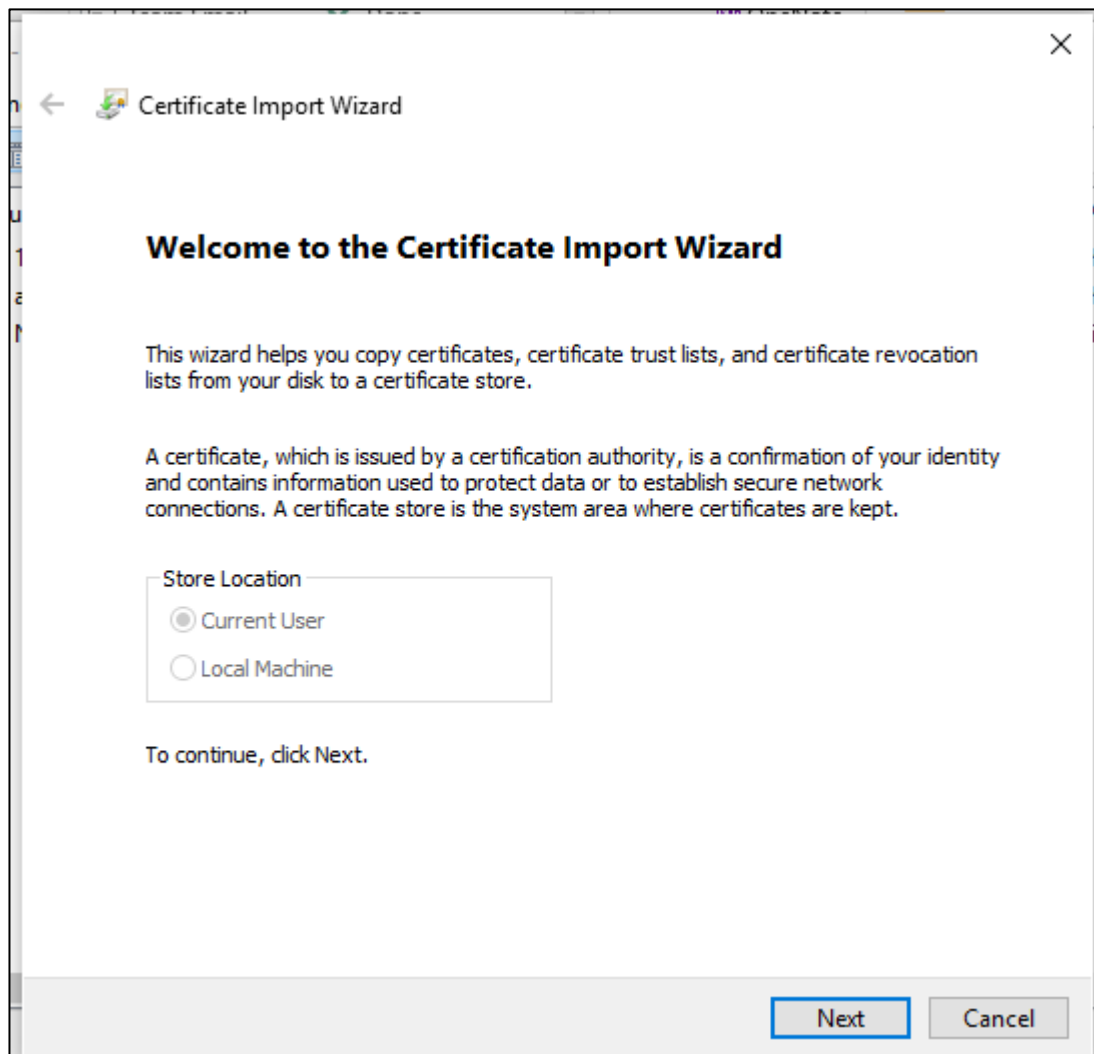


7. Right-click **Certificates**, select **All Tasks**, and click **Import**:





8. A Certificate Import Wizard gets launched, click **Next**.





9. Choose the certificate that is part of the deployment package:

The screenshot shows the 'Certificate Import Wizard' window. At the top, there is a back arrow and the title 'Certificate Import Wizard'. Below this, the section 'File to Import' is displayed with the instruction 'Specify the file you want to import.' A horizontal line separates this section from the input area. In the input area, there is a label 'File name:' followed by a text box and a 'Browse...' button. Below the text box, a 'Note' states: 'More than one certificate can be stored in a single file in the following formats:'. This is followed by three bullet points: 'Personal Information Exchange- PKCS #12 (.PFX,.P12)', 'Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)', and 'Microsoft Serialized Certificate Store (.SST)'. At the bottom of the window, there are 'Next' and 'Cancel' buttons. The 'Next' button is highlighted with a blue border.

← Certificate Import Wizard

**File to Import**  
Specify the file you want to import.

File name:  Browse...

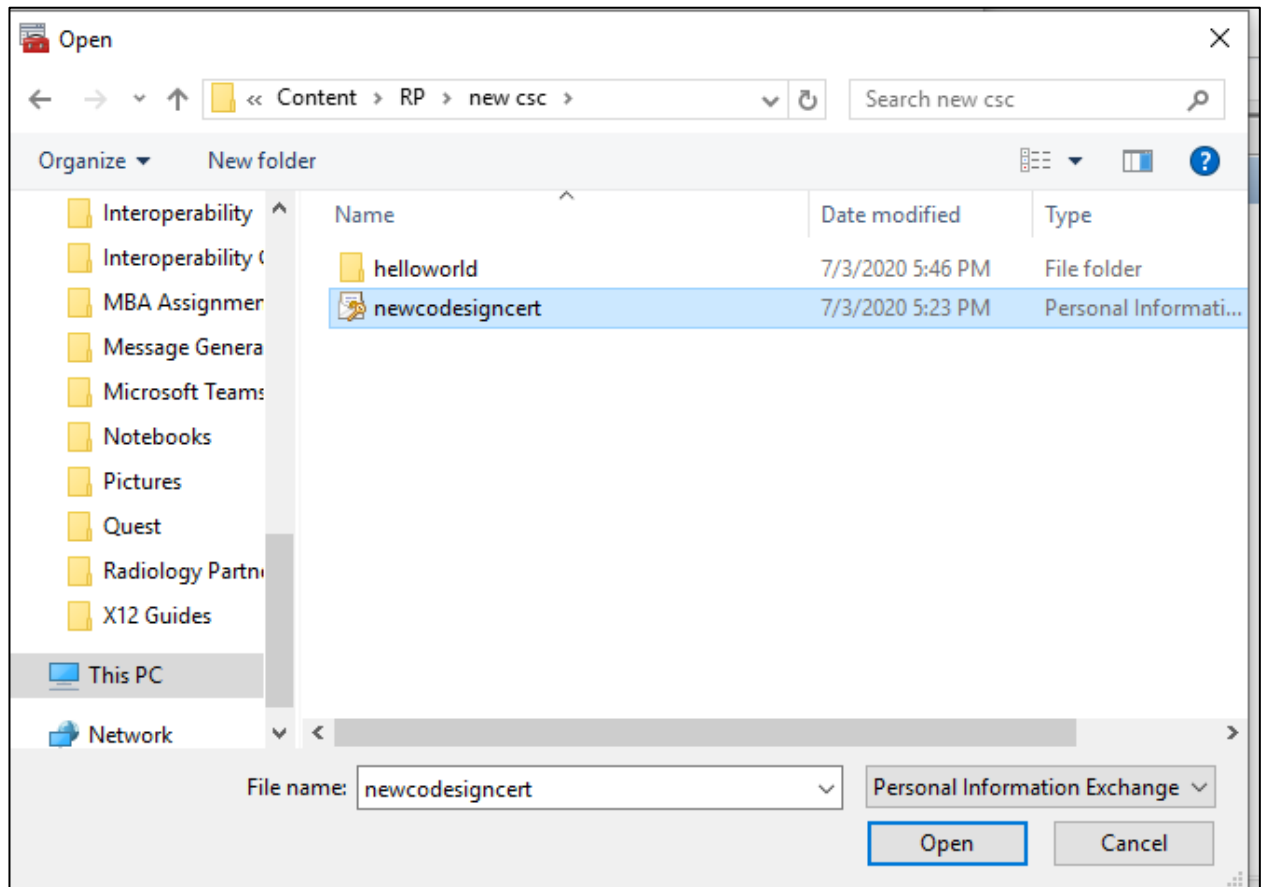
Note: More than one certificate can be stored in a single file in the following formats:

- Personal Information Exchange- PKCS #12 (.PFX,.P12)
- Cryptographic Message Syntax Standard- PKCS #7 Certificates (.P7B)
- Microsoft Serialized Certificate Store (.SST)

Next Cancel

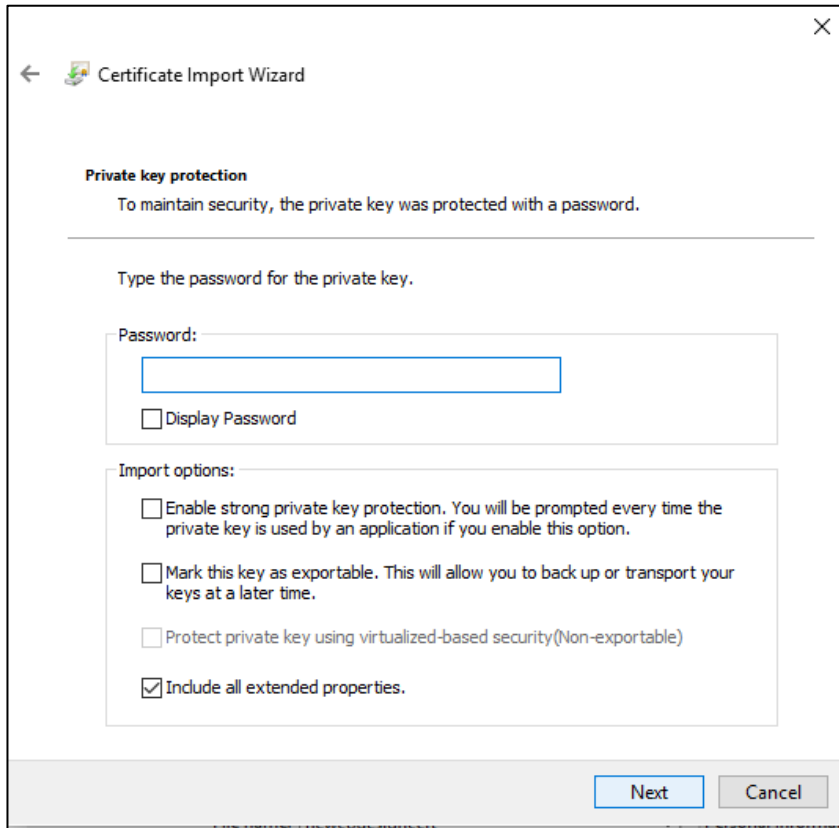


10. Browse to the location where the certificate file is saved and click **Open**:



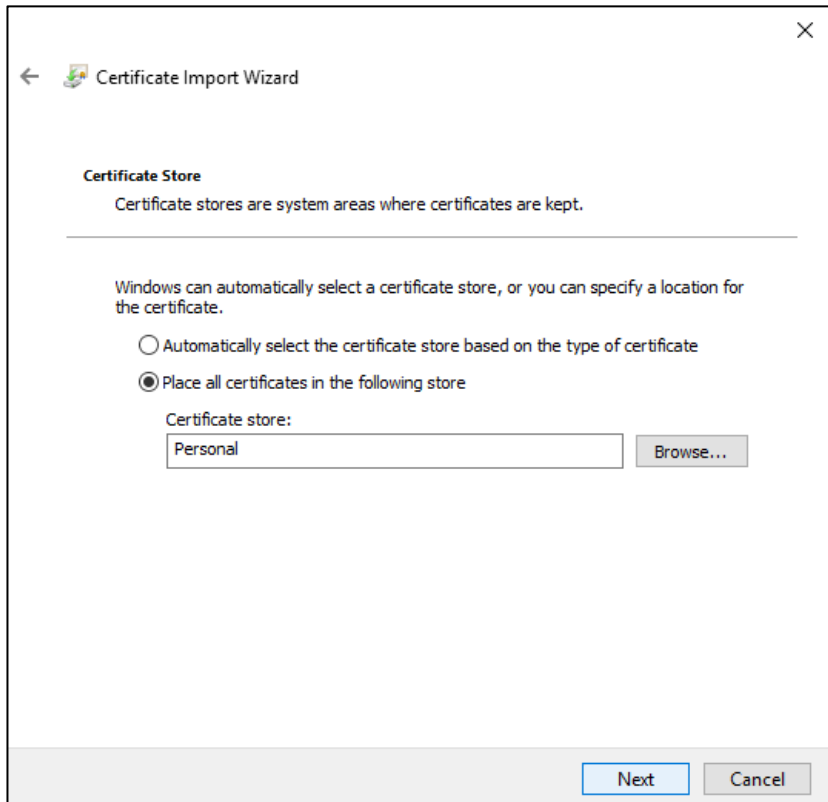


11. Click **Next** and type the private password that will be shared separately:



The screenshot shows the 'Certificate Import Wizard' window. The title bar includes a back arrow, a certificate icon, and the text 'Certificate Import Wizard'. The main content area is titled 'Private key protection' and contains the text 'To maintain security, the private key was protected with a password.' Below this is a section 'Type the password for the private key.' with a 'Password:' label and an empty text box. A checkbox labeled 'Display Password' is below the text box. Further down is a section 'Import options:' with four checkboxes: 'Enable strong private key protection...' (unchecked), 'Mark this key as exportable...' (unchecked), 'Protect private key using virtualized-based security(Non-exportable)' (unchecked), and 'Include all extended properties.' (checked). At the bottom right are 'Next' and 'Cancel' buttons.

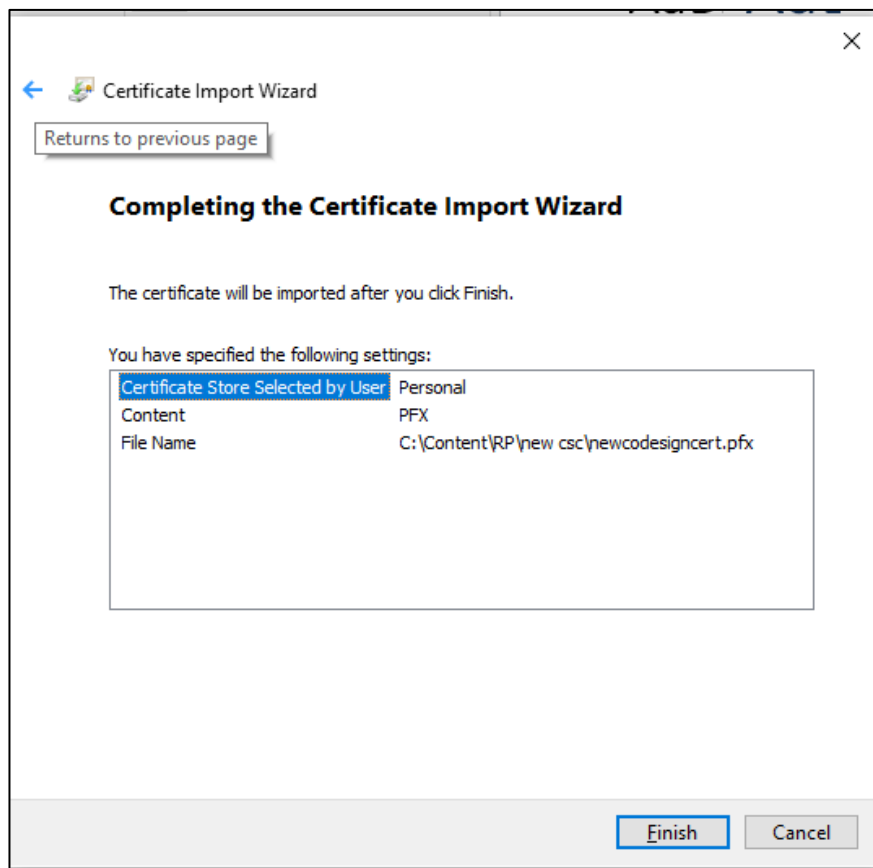
12. Click **Next** and choose the **Personal** Certificate store:



The screenshot shows the 'Certificate Import Wizard' window. The title bar includes a back arrow, a certificate icon, and the text 'Certificate Import Wizard'. The main content area is titled 'Certificate Store' and contains the text 'Certificate stores are system areas where certificates are kept.' Below this is a section 'Windows can automatically select a certificate store, or you can specify a location for the certificate.' with two radio buttons: 'Automatically select the certificate store based on the type of certificate' (unselected) and 'Place all certificates in the following store' (selected). Below the selected radio button is a 'Certificate store:' label and a text box containing 'Personal'. A 'Browse...' button is to the right of the text box. At the bottom right are 'Next' and 'Cancel' buttons.

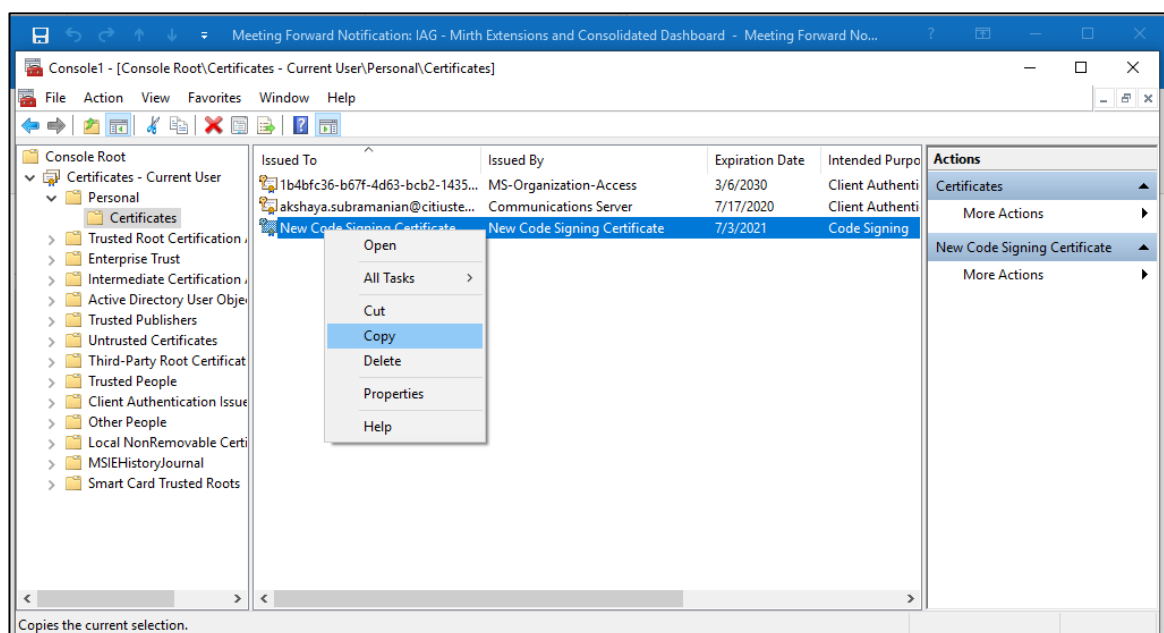


13. Click **Next** and then click **Finish**:

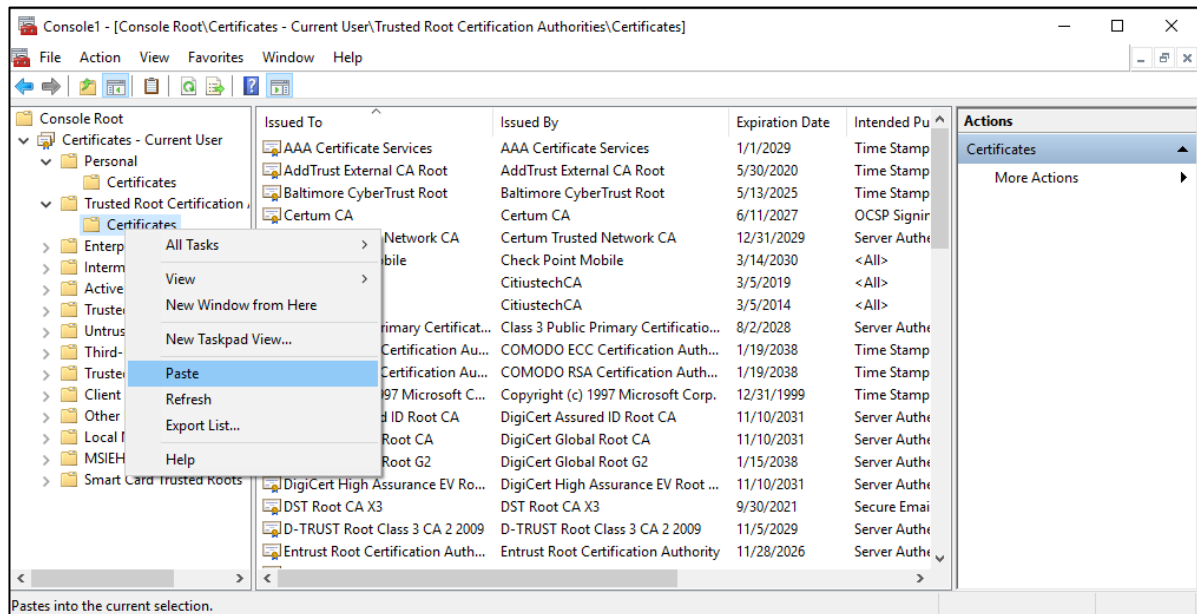


You should see the certificate installed in the Personal store.

14. Right-click the certificate and click **Copy**, as shown in the following screenshot:



15. Click **Trusted Root Certification Authorities -> Certificates**.
16. Right-click and select **Paste**:



The certificate would get copied in this location.

## 3.2.2 For Linux

Steps to add certificate to Linux: -

1. Copy the Code signing certificate .pfx and .jks files to Linux.

#Generate crt and key from pfx file

2. `openssl pkcs12 -in newcodesigncert.pfx -nokeys -out Certificate.crt -nodes`
3. `openssl pkcs12 -in newcodesigncert.pfx -nocerts -out Key.pem -nodes`

#Verifying

4. `openssl x509 -text -noout -in Certificate.crt`

or

`openssl x509 -noout -modulus -in Certificate.crt | openssl md5`

#Importing crt file to keystore jks file --- keep your own alies in below command

5. `keytool -import -trustcacerts -alias mycert -file Certificate.crt -keystore newcodesign.jks -- added to keystore`

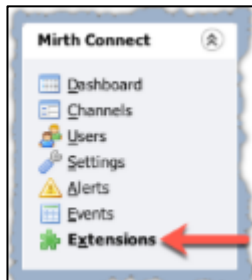
#View list of keystore certificates

6. `keytool -list -keystore newcodesign.jks -storepass <password>`



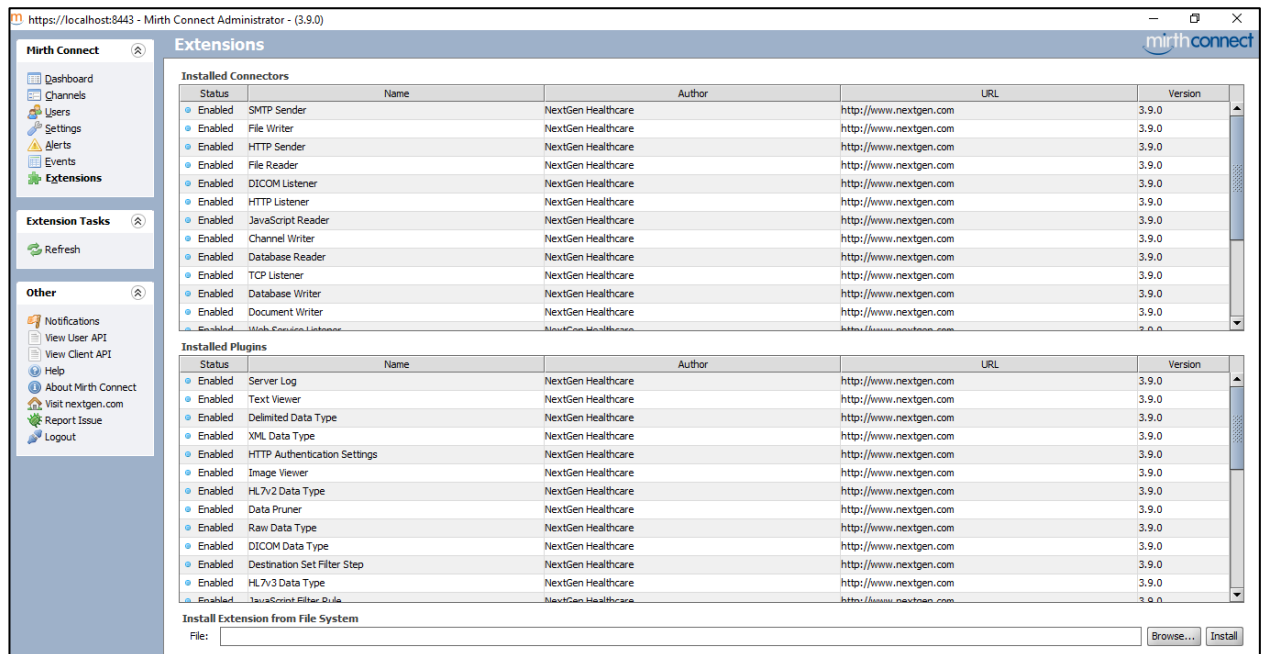
### 3.3 Installing the Extension

1. Launch the Mirth Connect administrator and login to the Mirth UI.
2. Click **Extensions** on the task pane at the upper left:

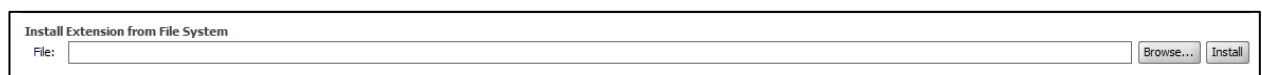


Extensions section is separated into the following categories:

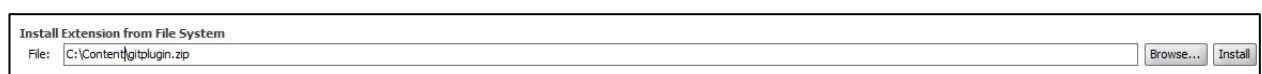
- Installed Connectors
- Installed Plugins
- Installing Extensions from file system
- Extension Tasks



3. Navigate to the **Install Extension from File System** at the bottom of the screen:



4. Click **Browse** and select the **GITPlugin.zip** file from the respective folder:



5. Click **Install**. The user gets a notification to restart the server:





The Mirth Connect Server and Administrator must be restarted before your changes will take effect.

Install Extension from File System

File:  Browse... Install

6. Restart the NextGen Connect Integration Engine server and launch the Mirth Connect Administrator. The user will be able to see the new extension listed in the Installed Plugins table.

## 3.4



## Appendix

### Licenses

The Software may contain third party software which requires notices and/or additional terms, conditions and licenses. Such required third party software notices and/or additional terms, conditions and licenses can be located [here](#).



## 3.5 Annexure

### Deployment Checklist:

This section provides the template for maintaining the details for each environment mentioned in section [2.1](#) (Separate row to be created for each environment).

Sr No	Client Name	Environment	Machine Name (IP Address)	Processor	# of Cores	Operating System	RAM	Total Hard Disk Space	Third Party Software Deployed	Solution Component Deployed