



ADSS Web RA Server 2.9.10  
Installation  
Guide

---

ASCERTIA LTD

FEBRUARY 2026

Document Version - 1.0.7

---

© Ascertia Limited. All rights reserved.

This document contains commercial-in-confidence material. It must not be disclosed to any third party without the written authority of Ascertia Limited.

---

Commercial-in-Confidence

## Table of Contents

<b>1</b>	<b>Introduction.....</b>	<b>3</b>
1.1	Scope .....	3
1.2	Intended Readership .....	3
1.3	Technical Support.....	3
1.4	Glossary .....	4
<b>2</b>	<b>System Requirements.....</b>	<b>5</b>
2.1	Hardware Prerequisites .....	5
2.2	Software Prerequisites.....	5
2.3	Application Development feature in IIS .....	7
2.4	Microsoft .Net Core 9.0. Runtime & Hosting Bundle .....	8
2.5	Microsoft IIS URL Rewrite Module 2.1 .....	10
2.6	Unlock system.webServer/serverRuntime section in IIS .....	12
2.7	SMTP Server.....	13
2.8	Database .....	13
<b>3</b>	<b>Installation Modules.....</b>	<b>14</b>
<b>4</b>	<b>ADSS Web RA Installation on Windows Server.....</b>	<b>15</b>
4.1	Regular Release Installation .....	15
4.2	Uninstalling Regular Release .....	21
<b>5</b>	<b>ADSS Web RA Installation on Linux System.....</b>	<b>26</b>
5.1	Prerequisites for Linux Installation .....	26
5.2	Pre-Installation Steps.....	29
5.3	Configuring Installation Parameters in install.json file .....	31
<b>6</b>	<b>Appendix.....</b>	<b>42</b>
6.1	Troubleshooting.....	42
6.2	Troubleshooting for Linux.....	44
6.3	Configurations used for Simple Certificate Enrollment Protocol (SCEP) .....	47
6.4	SSL Certificates.....	48
6.5	SSL Configuration for Linux .....	51
6.6	Importing Root and Intermediate Certificates .....	52
6.7	Generate a Self -Signed Certificate .....	55
6.8	Generate a CSR for an SSL Certificate.....	58

# 1 Introduction

Registration Authority (RA) is another important component of PKI along with Certificate Authority (CA). CA is primarily responsible to create and revoke certificates, but complex business scenarios demand more than just the creation of certificates. Their responsibilities now include but not limited to managing users, certificate creation requests and revocation of certificates.

Businesses in the modern world require strong control over these processes along with the complete audit trail, to maintain the irrefutable evidence of these activities for future. Such additional controls and management are covered by an RA. An RA is therefore responsible to verify a user and their certificate request, and then inform the CA to issue the requested certificate.

An RA receives a request for digital certificate and verifies the user requesting the certificate. The user verification can be done manually through face to face interaction or electronically by using other mediums like phone, video conferencing, mail or courier that is acceptable to the RA as a secured medium. Once RA approves the user, it informs the CA to issue the certificate for the user. The RA then obtains the user certificate from the CA, and sends it to the user using a secure medium.

## 1.1 Scope

This manual describes how to install ADSS Web RA Server.

ADSS Web RA comprises five components and the installation procedure for all are covered herein:

- **Web** interface that provides user services on desktop browsers.
- **Admin** console that provides system administration and configuration.
- **API** that utilises the ASP.NET Web API framework to provide a REST architecture.
- **Device** is used to manage device enrolment for certificate creation.
- **Windows Enrolment** is used to manage certificate renewal or auto-enrolment on a Windows machine.

## 1.2 Intended Readership

This manual is intended for administrators responsible for installation and initial configuration. It is assumed that the reader has a good understanding of web applications running on IIS, digital signatures, digital certificates and IT security.

## 1.3 Technical Support

If technical support is required, Ascertia has a dedicated support team providing debugging and integration assistance as well as general customer support. Ascertia Support can be accessed through [Ascertia Ticketing System](#) or email address: [support@ascertia.com](mailto:support@ascertia.com)

Ascertia provides formal support agreements with all product sales. Contact [sales@ascertia.com](mailto:sales@ascertia.com) for further details.

A Product Support Questionnaire should be completed in order to provide Ascertia Support having information about your system environment, along with details of any issues encountered. When requesting help, it is always important to confirm these details:

- System platform.
- ADSS Web RA version number.
- Details of the specific issue and relevant steps taken to reproduce it if possible.
- Database vendor, version and patch level.
- Product log files.

## 1.4 Glossary

ADSS Web RA	A short form of Unified Web Registration Authority
Cert	A short form of Digital Certificate
DBMS	Database Management System
HSM	Hardware Security Module
HTTP	Hyper Text Transfer Protocol
HTTP/S	HTTP over SSL/TLS connection
SSL	Secure Sockets Layer

## 2 System Requirements

System Requirements includes hardware and software requirements both.

### 2.1 Hardware Prerequisites

Components	Requirements
<b>Hard Disk Space</b>	<ul style="list-style-type: none"> <li>200 GB (Minimum)</li> </ul>
<b>Memory</b>	<ul style="list-style-type: none"> <li>16 GB (Minimum)</li> <li>24 GB (If the number of concurrent users is higher)</li> <li>32 GB (If the database is also deployed on the same system as the ADSS Web RA)</li> </ul>
<b>Processor</b>	<ul style="list-style-type: none"> <li>A modern multi-core CPU such as Xeon E3-XXXX or E5-XXXX series is recommended</li> </ul>
<b>Processor Type</b>	<ul style="list-style-type: none"> <li>x64</li> </ul>
<b>HSM (Optional)</b>	<ul style="list-style-type: none"> <li>Thales Luna Network, PCIe, and USB</li> <li>Entrust nShield Solo XC, Connect XC, and nShield EDGE</li> <li>Utimaco CryptoServer SE Gen2</li> <li>Microsoft Azure Key Vault</li> <li>Amazon Cloud HSM</li> </ul>

### 2.2 Software Prerequisites

Component	Requirements
<b>Operating Systems</b>	<ul style="list-style-type: none"> <li>Follow this link to view details about supported OS:  <a href="https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf">https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf</a> </li> </ul>
<b>Microsoft IIS</b>	<ul style="list-style-type: none"> <li>IIS 10</li> <li>Application Development feature in IIS</li> </ul>
<b>IIS Rewrite Module</b>	<ul style="list-style-type: none"> <li>v2.1</li> </ul>
<b>.Net Framework</b>	<ul style="list-style-type: none"> <li>.Net Framework 4.8.0 or above</li> </ul>
<b>.Net Core Runtime &amp; Hosting Bundle</b>	<ul style="list-style-type: none"> <li>ASP.NET Core Runtime 9.0 or above</li> </ul>

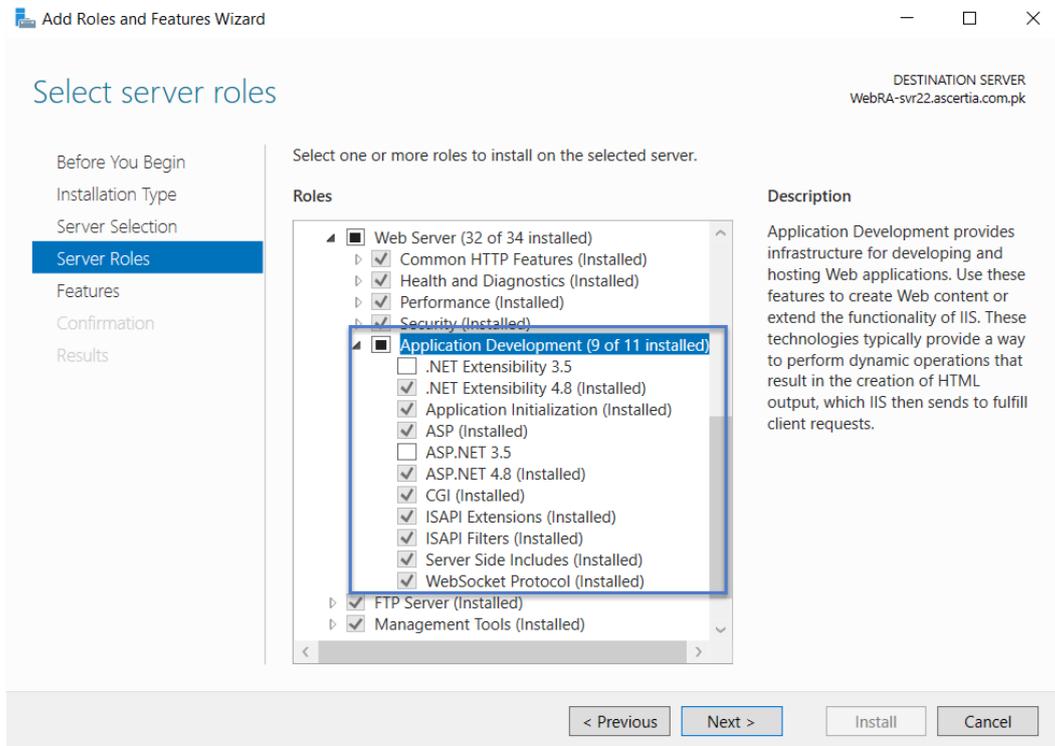
<b>Database Server</b>	<ul style="list-style-type: none"> <li>Follow this link to view details about Database Server:  <a href="https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf">https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf</a> </li> </ul>
<b>Web Brower (for end-users and administrators)</b>	<ul style="list-style-type: none"> <li>Follow this link to view details about Web Browsers:  <a href="https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf">https://manuals.ascertia.com/WebRA/ADSS-WebRA-Server-Platform-Support.pdf</a> </li> </ul>
<b>ADSS Server</b>	<p>ADSS Web RA uses ADSS Server under the hood to create and manage certificates for the end user as a CA. ADSS Server can be installed on a separate machine or on the same machine for testing and proof of concept. It is recommended to keep the ADSS installation on a separate machine for a production environment. For further requirements related to the installation of ADSS Server, please refer to the installation guide of ADSS Server.</p> <ul style="list-style-type: none"> <li>ADSS Server 6.6 or above</li> </ul>
<b>DMZ Proxy Systems</b>	<p>A DMZ proxy server is recommended to provide enhanced security for ADSS Web RA. Supported web servers are:</p> <ul style="list-style-type: none"> <li>Windows Server + IIS, Apache or IBM HTTP Server</li> <li>Linux + Apache or IBM HTTP Server</li> </ul> <p>It is recommended to use a reasonable CPU, 4 GB RAM (Minimum), 2000 MB Disk Space for the web server machine. ADSS Web RA and ADSS Server support network proxies to allow authenticated access to external services. Certificate generation with local smartcards or USB tokens requires ADSS Server Go&gt;Sign Service.</p>

For testing and proof of concepts, ADSS Server and ADSS Web RA can be installed on the same machine along with the database server. However, for optimal performance in a production environment, it is always recommended to install them on separately dedicated machines.

The details given above are the minimum set of requirements; for higher concurrent use of the application the system requirements may vary based on the load and performance expectations.

## 2.3 Application Development feature in IIS

Enable the following features in IIS on the deployment machine:



## 2.4 Microsoft .Net Core 9.0. Runtime & Hosting Bundle

2.4.1 Download the latest version of Microsoft .Net Core i.e. Microsoft .Net Core 9.0.12. Runtime and Hosting Bundle from the following link:

[Microsoft .Net Core 9.0. Runtime & Hosting Bundle](#)

2.4.2 Download the Hosting Bundle installer.

9.0.12  
[Release notes](#) Latest release date January 20, 2026

Build apps - SDK

### SDK 9.0.310

OS	Installers	Binaries
Linux	<a href="#">Package manager instructions</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS	<a href="#">Arm64</a>   <a href="#">x64</a>	<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>   <a href="#">winget instructions</a>	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>
All	<a href="#">dotnet-install scripts</a>	

**Visual Studio support**  
 Visual Studio 2022 (v17.14)  
**Included in**  
 Visual Studio 17.14.25  
**Included runtimes**  
 .NET Runtime 9.0.12  
 ASP.NET Core Runtime 9.0.12  
 .NET Desktop Runtime 9.0.12  
**Language support**  
 C# 13.0  
 F# 9.0  
 Visual Basic 17.13

### SDK 9.0.309

OS	Installers	Binaries
Linux	<a href="#">Package manager instructions</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS	<a href="#">Arm64</a>   <a href="#">x64</a>	<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>   <a href="#">winget instructions</a>	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>
All	<a href="#">dotnet-install scripts</a>	

Run apps - Runtime

### ASP.NET Core Runtime 9.0.12

The ASP.NET Core Runtime enables you to run existing web/server applications. **On Windows, we recommend installing the Hosting Bundle, which includes the .NET Runtime and IIS support.**

**IIS runtime support (ASP.NET Core Module v2)**  
 19.0.25343.12

OS	Installers	Binaries
Linux	<a href="#">Package manager instructions</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS		<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>   <a href="#">Hosting Bundle</a>   <a href="#">winget instructions</a>	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>

### .NET Desktop Runtime 9.0.12

The .NET Desktop Runtime enables you to run existing Windows desktop applications. **This release includes the .NET Runtime; you don't need to install it separately.**

OS	Installers	Binaries
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>   <a href="#">winget instructions</a>	

### .NET Runtime 9.0.12

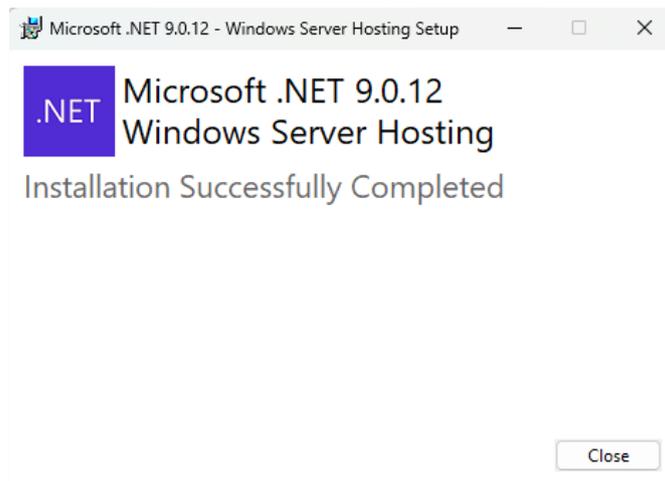
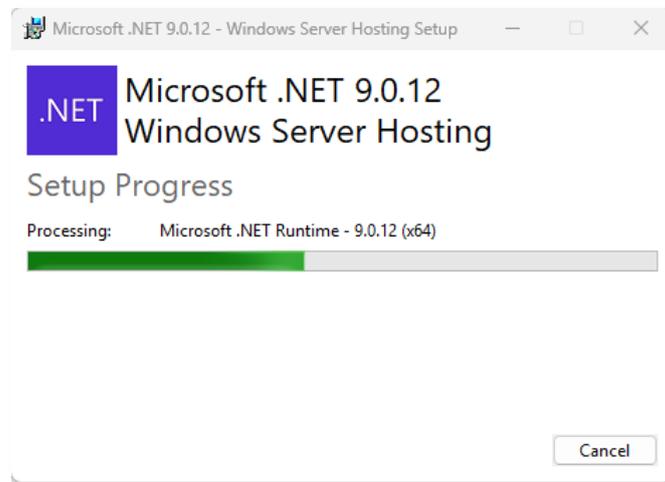
The .NET Runtime contains just the components needed to run a console app. Typically, you'd also install either the ASP.NET Core Runtime or .NET Desktop Runtime.

OS	Installers	Binaries
Linux	<a href="#">Package manager instructions</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS	<a href="#">Arm64</a>   <a href="#">x64</a>	<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>

2.4.1. Once downloaded, execute the installer by executing **dotnet-hosting-9.0.12-win.exe**



**2.4.2.** The setup will begin and take a few minutes to complete.



**2.4.3.** Once the installation process is complete, click **Close**.

**2.4.4.** To test if the installation was correct and components are reachable, run command line and type the following command:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.17763.4252]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ >dotnet

Usage: dotnet [options]
Usage: dotnet [path-to-application]

Options:
  -h|--help          Display help.
  --info             Display .NET information.
  --list-sdks        Display the installed SDKs.
  --list-runtimes    Display the installed runtimes.

path-to-application:
  The path to an application .dll file to execute.

C:\Users\
```

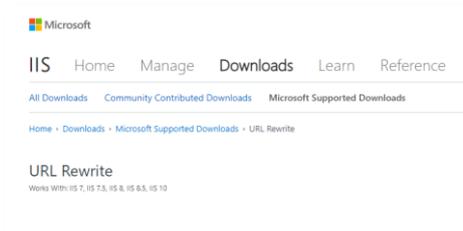
2.4.5. Now, restart your machine to apply these changes effectively.

## 2.5 Microsoft IIS URL Rewrite Module 2.1

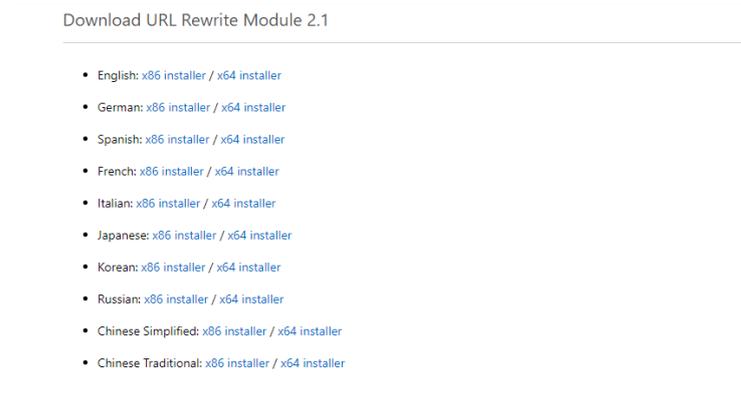
2.5.1. Download **Microsoft IIS URL rewrite module 2.1** from the following link:

### [Microsoft IIS URL Rewrite Module 2.1](#)

2.5.2. Navigating to this URL will present with the following screen:



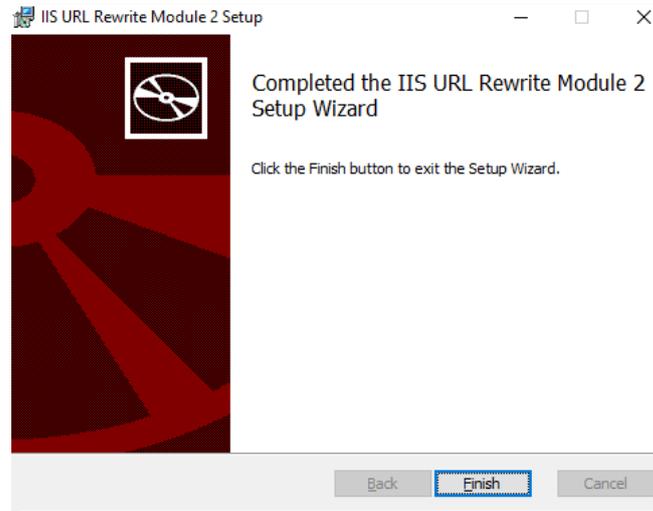
2.5.3. Scroll down to find a list of links available for download.



2.5.4. Download **x64 installer** with your preferred language. For this documentation it's **English**. Start the installation by executing the downloaded file in administrator mode.



**2.5.5.** Accept the terms in the license agreement and click **Install** to proceed, the installation will take few minutes:



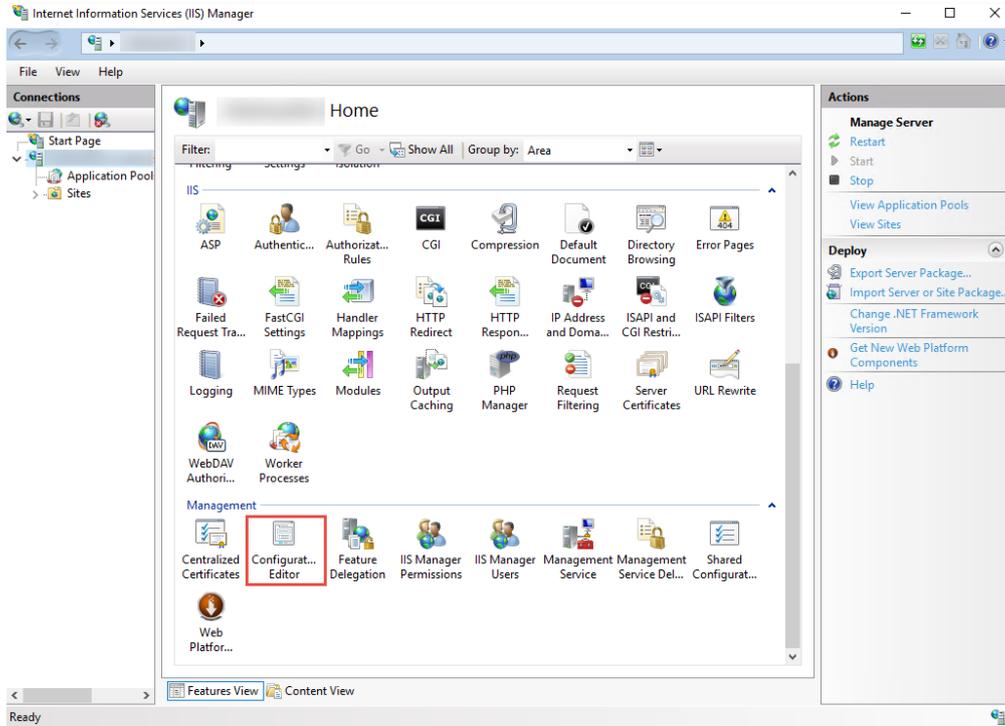
**2.5.6.** Click **Finish** once the installation process is complete.

## 2.6 Unlock system.webServer/serverRuntime section in IIS

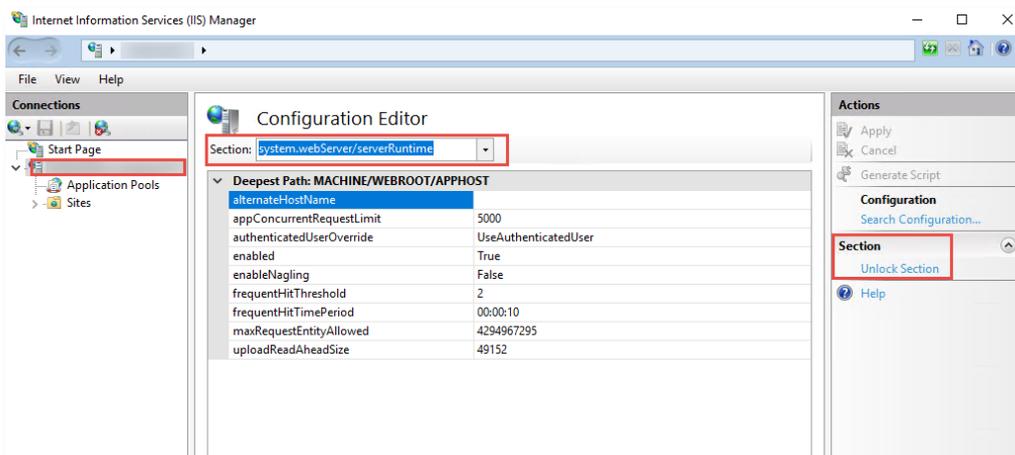
2.6.1. Launch the **IIS Manager**

2.6.2. Select **Server** from left panel

2.6.3. Open **Configuration Editor** from right pane under the Management section.



2.6.4. Unlock **system.webServer/serverRuntime** section in the Configuration Editor.



The installation process for prerequisites is complete.

## 2.7 SMTP Server

ADSS Web RA uses email as the primary notification medium. User registration, and all notifications are sent via SMTP. Hence, it is a critical part of the architecture and deployment. Details required are:

- Hostname/IP address of SMTP server
- Listening Port of SMTP server
- TLS/SSL authentication to communicate with SMTP server (if required)
- Username and password to authenticate to SMTP server (if required)
- Email from Address for notifications sent from ADSS Web RA
- Email to Address for alerts and warnings sent by ADSS Web RA
- Email Subject for alerts and warnings sent by ADSS Web RA



*If there is no alternative it is possible to still use ADSS Web RA. However, this involves copying the notification emails directly from the database and manually running the links therein. This usage is strongly discouraged in favour of a standard deployment though.*

---

## 2.8 Database

ADSS Web RA Server requires its own database. It is not required to create the schema or configure any other feature prior to the installation.

Permissions are required to allow the creation of database tables, and entry, modification, and removal of data within those tables.

### 3 Installation Modules

ADSS Web RA consists of the following modules. Note the API is the only non-mandatory ones for a working solution:

- **ADSS Web RA Admin**

Administration application that allows to manage the system wide configurations, service plans, user accounts and access controls, etc.

- **ADSS Web RA Desktop Web**

ADSS Web RA Web is used for managing certificates i.e. creation, renewal and revocation.

- **ADSS Web RA API (Restful Web Services)**

REST architecture API support that is used to integrate ADSS Web RA functionality within your own portal. The API uses JWT to implement authentication and authorization. There is a separate API Guide that provides full details of the REST architecture implementation.

- **ADSS Web RA Device**

ADSS Web RA Device is used to manage device enrolment for certificate creation, renewal and revocation.

- **ADSS Web RA SSL Device**

ADSS Web RA SSL Device is used to manage device enrolment over SSL for certificate creation, renewal and revocation e.g. EST Protocol

- **Windows Enrolment**

ADSS Web RA Windows Enrolment is used to manage certificate renewal or auto-enrolment on a Windows machine.

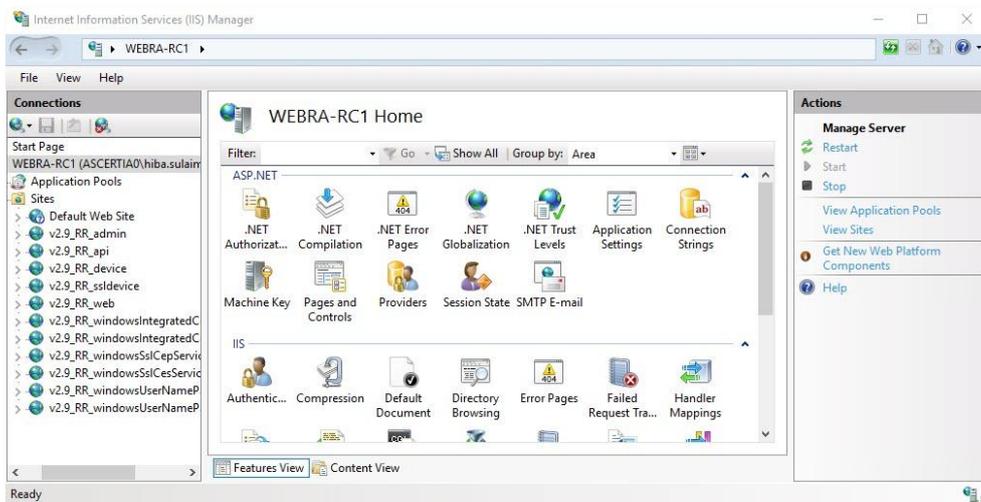
## 4 ADSS Web RA Installation on Windows Server

### 4.1 Regular Release Installation

**Note:** If you are upgrading from v2.9 to v2.9.10, ensure that your v2.9 deployment is functioning properly by accessing it in a browser.

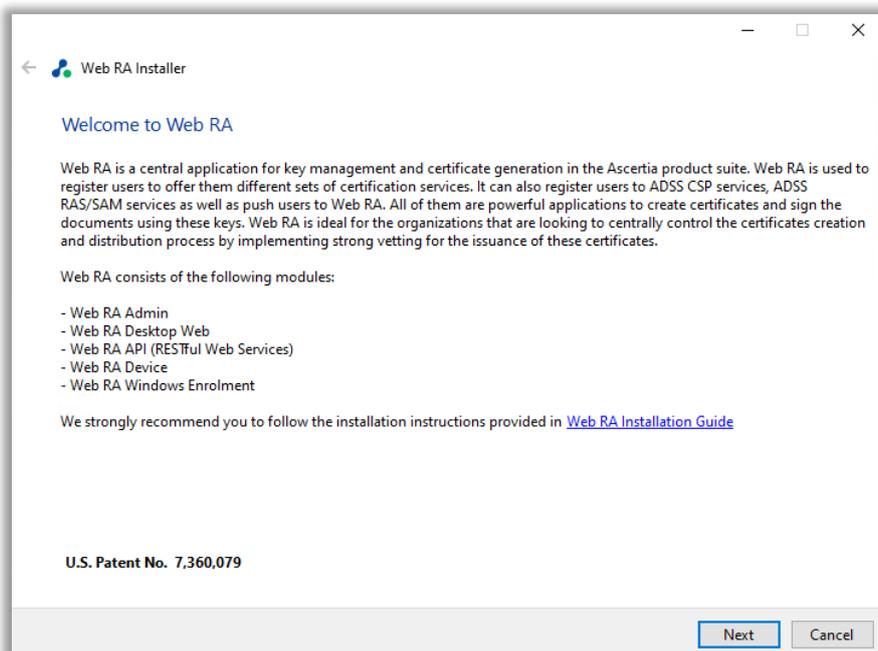
Follow the instructions below to install ADSS Web RA's regular release. Before starting the installation make sure that you have taken a backup of the Web RA database and have stopped the IIS Server.

To stop the IIS Server, launch the IIS Manager and click Stop under the Manage Server action.

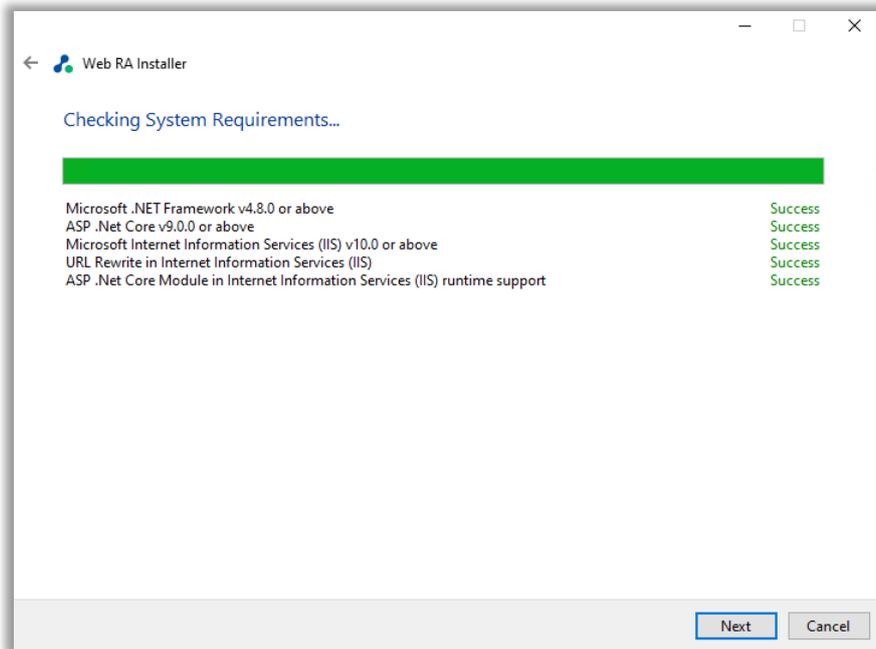


4.1.1 Launch the installer by right-clicking the file name [Web RA Regular Release Installation Directory]/setup/install.bat and select Run as administrator. Follow the installation wizard as described below:

The Welcome screen will appear:

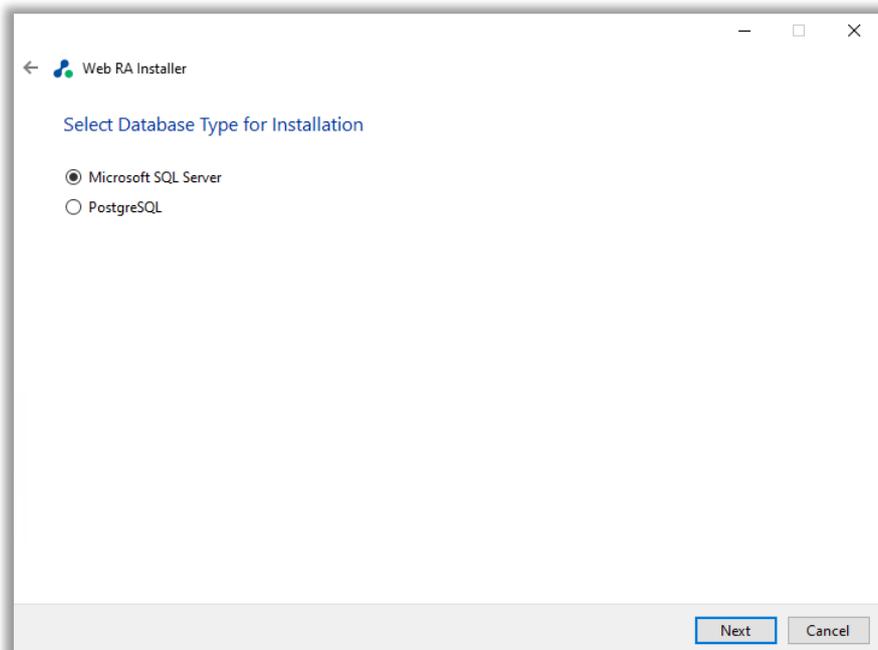


4.1.2 Click the Next button to continue. The system requirements screen will appear next to validate if all the required prerequisites are installed.

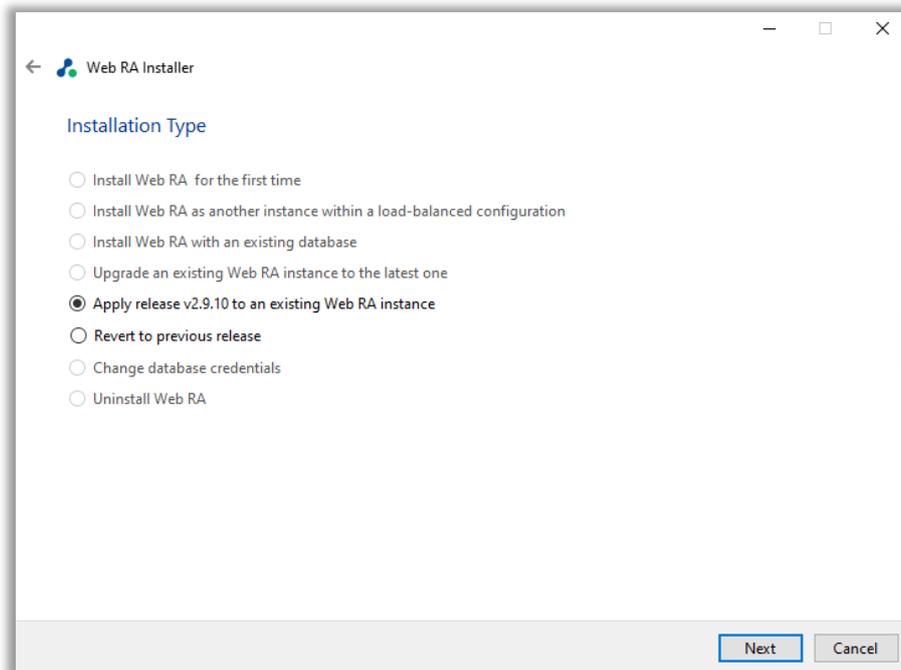


4.1.3 Click the Next button to continue to the database type screen. Select the database type -- **Microsoft SQL Server** or **PostgreSQL** – that was used in your previous Web RA installation.

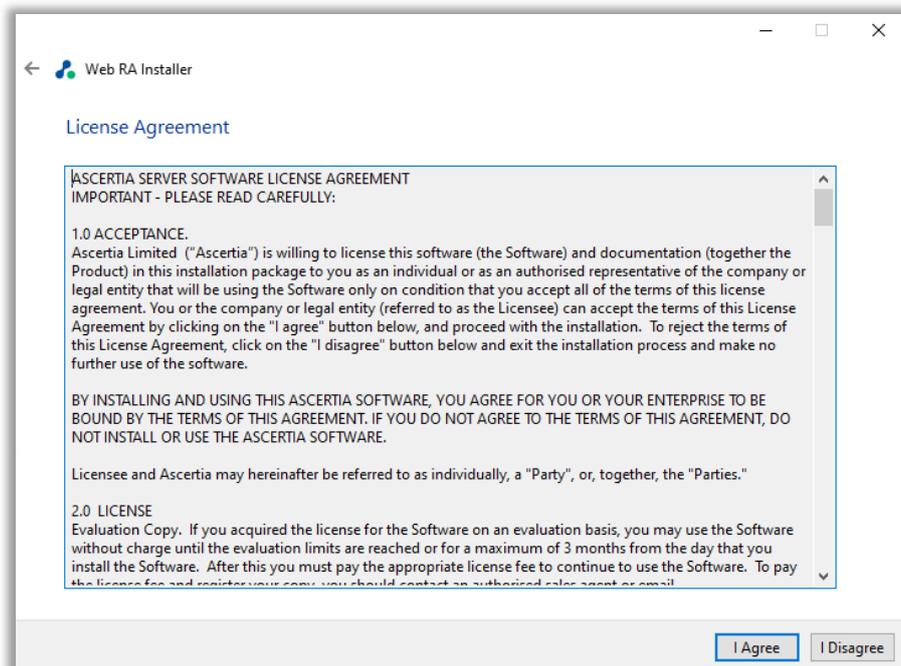
**Note:** To apply release v2.9.8 or above to an existing Web RA instance, Microsoft SQL Server 2022 or later is required.



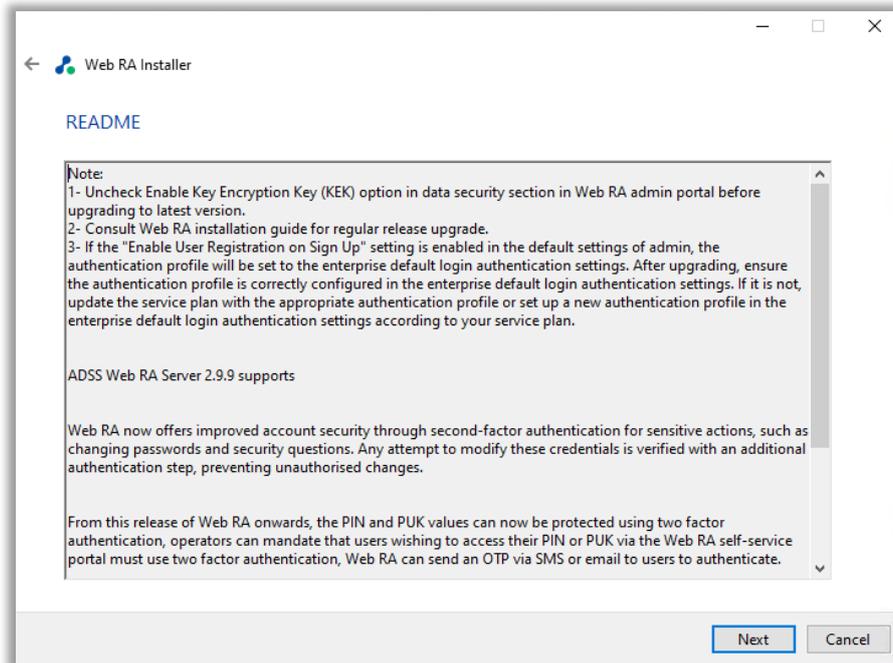
4.1.4 Click the Next button to continue. The 'Installation Type' screen will appear.



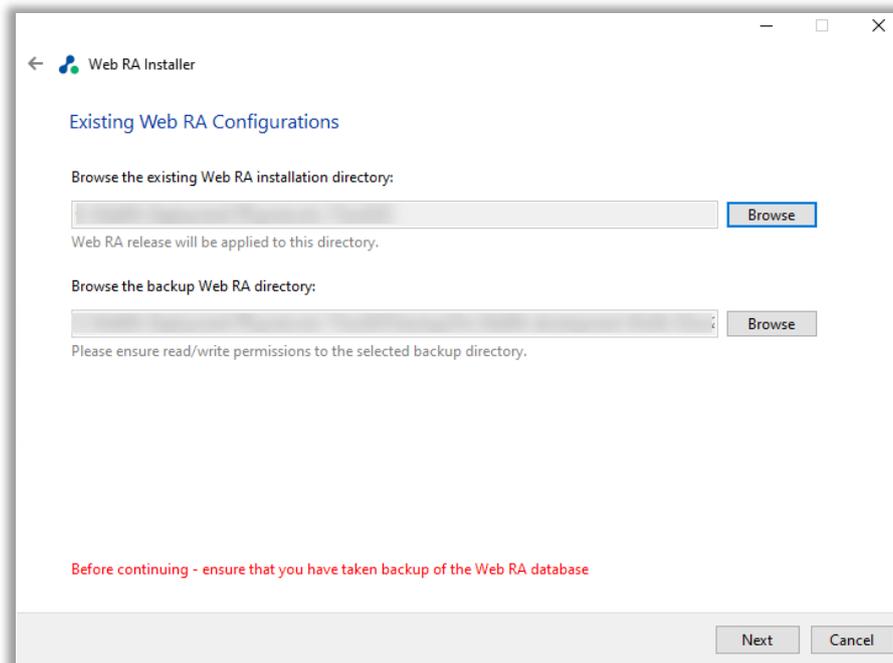
4.1.5 Click the Next button to view and accept the 'License Agreement'.



#### 4.1.6 Click 'I Agree' to proceed to the 'Read Me' screen.



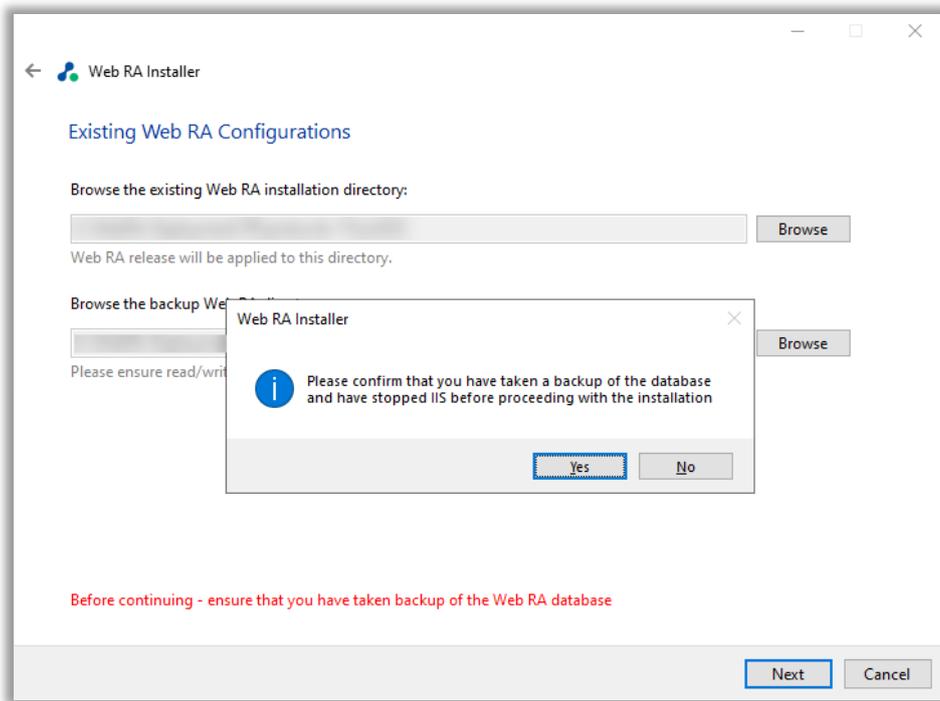
#### 4.1.7 Click the 'Next' button to provide the 'Existing and Backup' Web RA directory addresses.



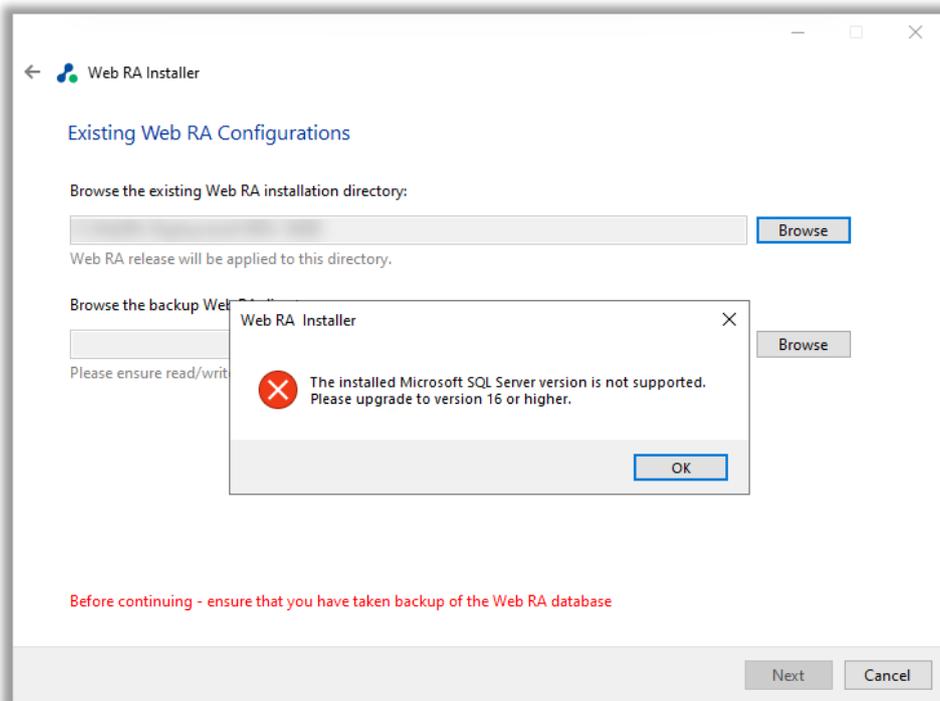
Click the Browse button against the existing Web RA installation directory. Then click the Browse button against the backup Web RA directory, to browse to the addresses for the respective directories.

By default, when the existing Web RA installation directory address is selected, the installer will automatically create a backup Web RA folder and select it as backup directory. However, if the user wants to change the backup directory, they can click "Browse" and manually select the backup directory.

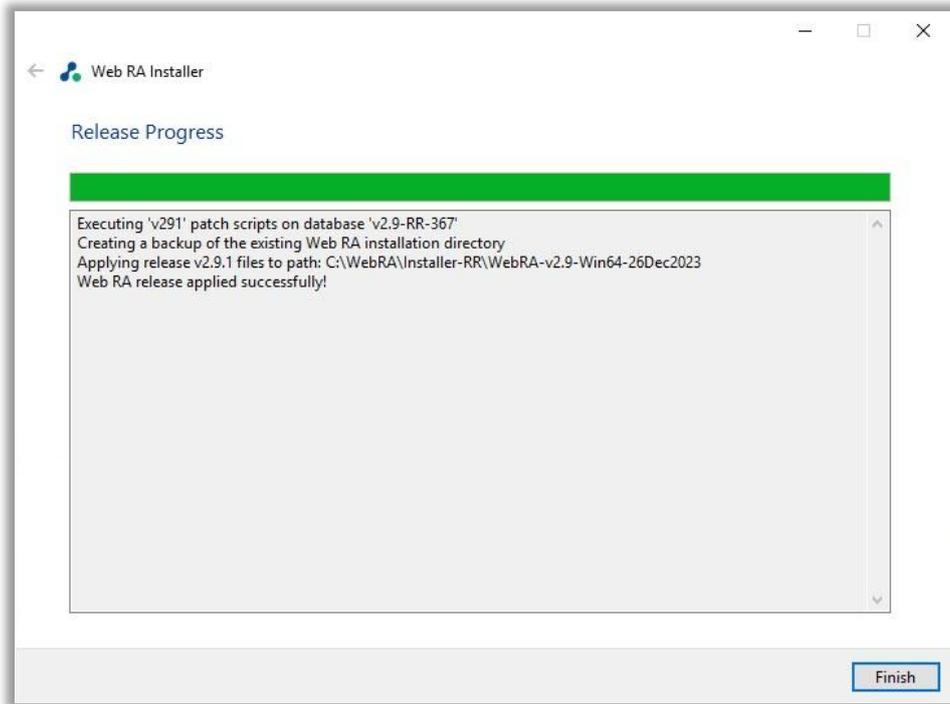
After selecting the directories, when you click 'Next' a confirmation dialog will appear. Click the 'Yes' button to confirm that you have taken a backup of the database and have stopped the IIS before proceeding with the installation:



**Note:** If the installed Microsoft SQL version is below 16, the installer will display an error dialog, prompting you to upgrade to version 16 or higher.



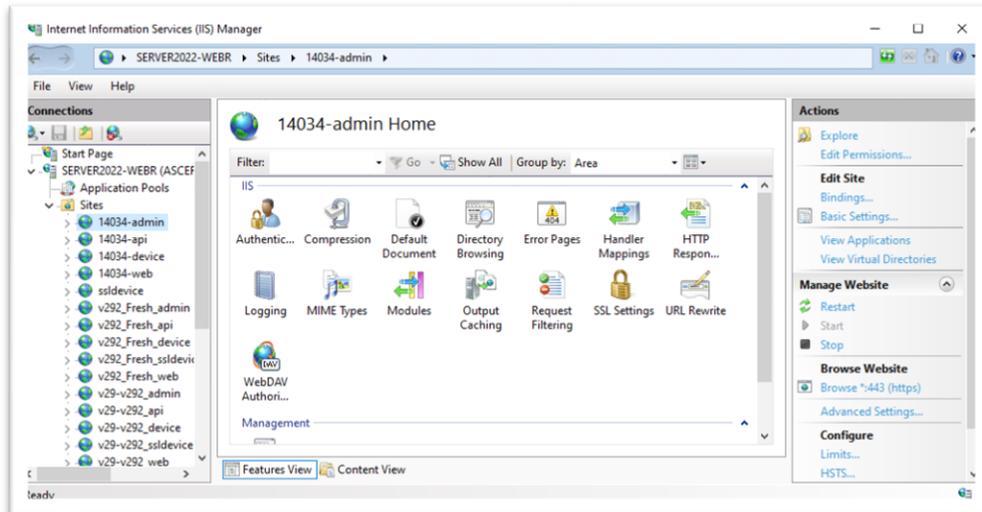
4.1.8 Click the 'Finish' button to complete the installation process.



## 4.2 Uninstalling Regular Release

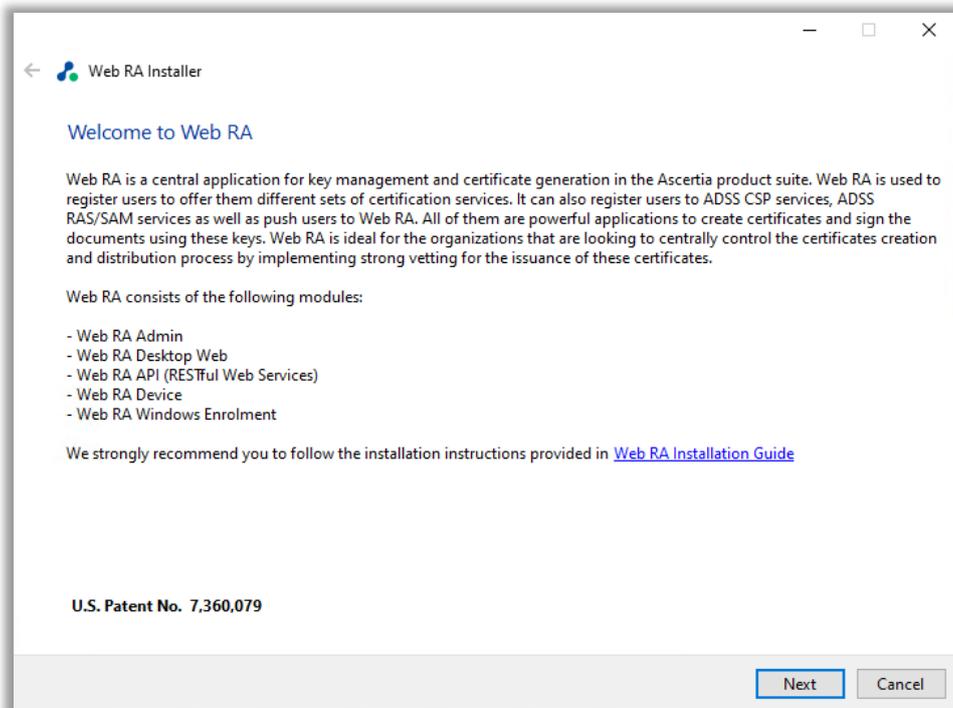
Follow the instructions below to uninstall ADSS Web RA's regular release. Before starting the uninstallation make sure that you have taken a backup of the Web RA database and have stopped the IIS Server.

To stop the IIS Server, launch the IIS Server and click Stop under the Manage Server action.

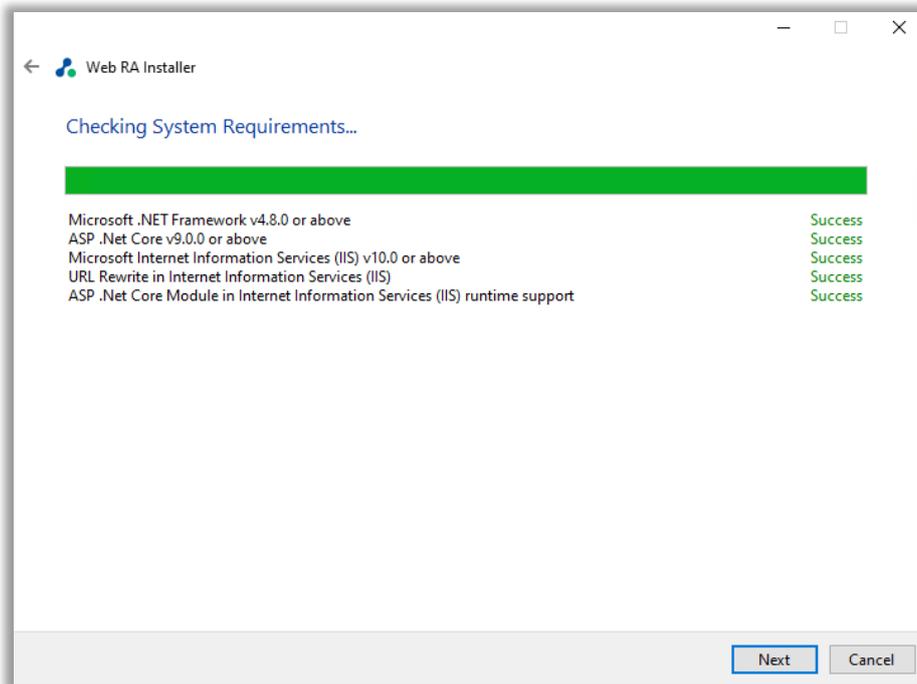


4.2.1 Launch the installer by right-clicking the file name [Web RA Regular Release Installation Directory]/setup/install.bat and select Run as administrator. Follow the installation wizard as described below:

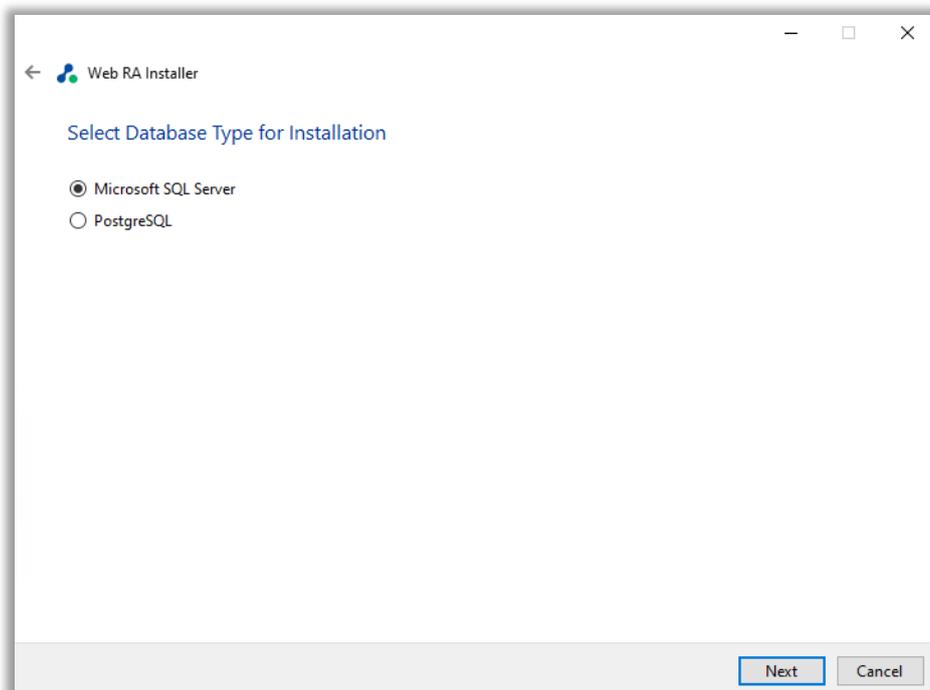
The Welcome screen will appear:



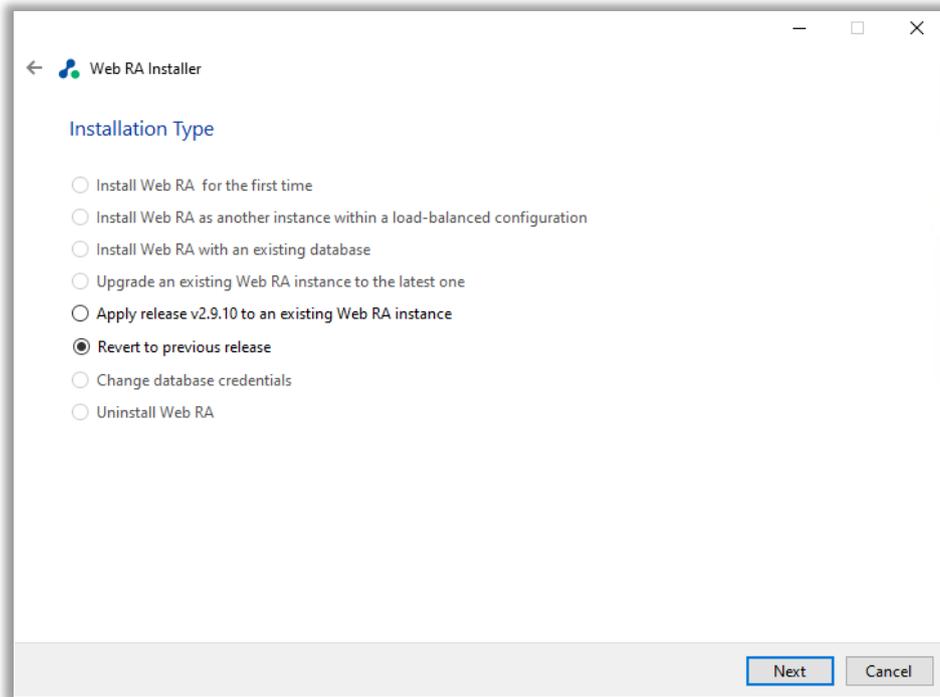
4.2.2 Click the Next button to continue. The system requirements screen will appear next to validate if all the required prerequisites are installed.



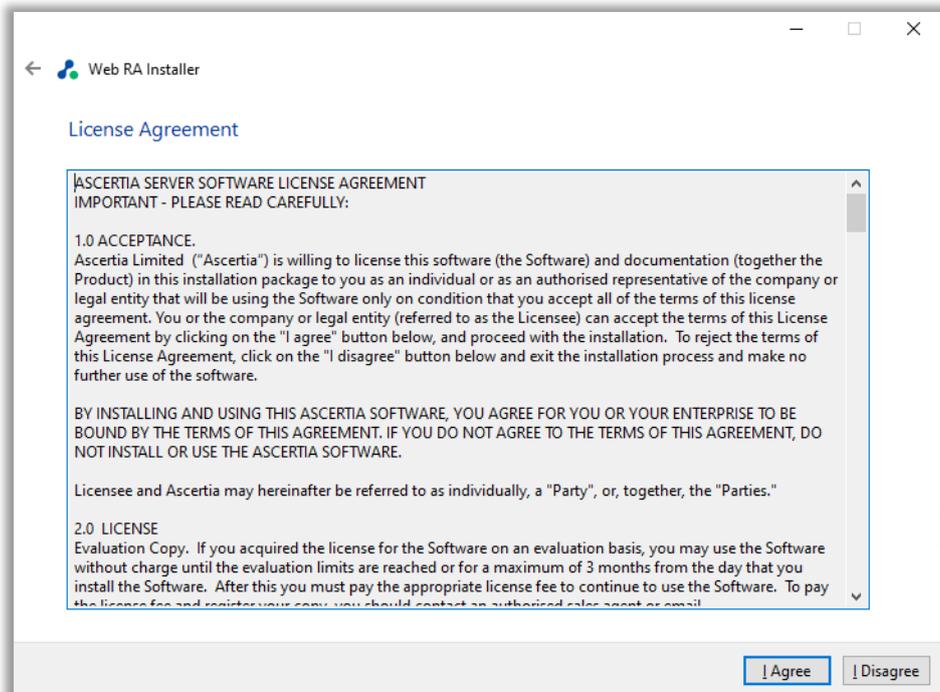
4.2.3 Click the Next button to continue to the database type screen. Select the database type -- **Microsoft SQL Server** or **PostgreSQL** – that was used in your previous Web RA installation.



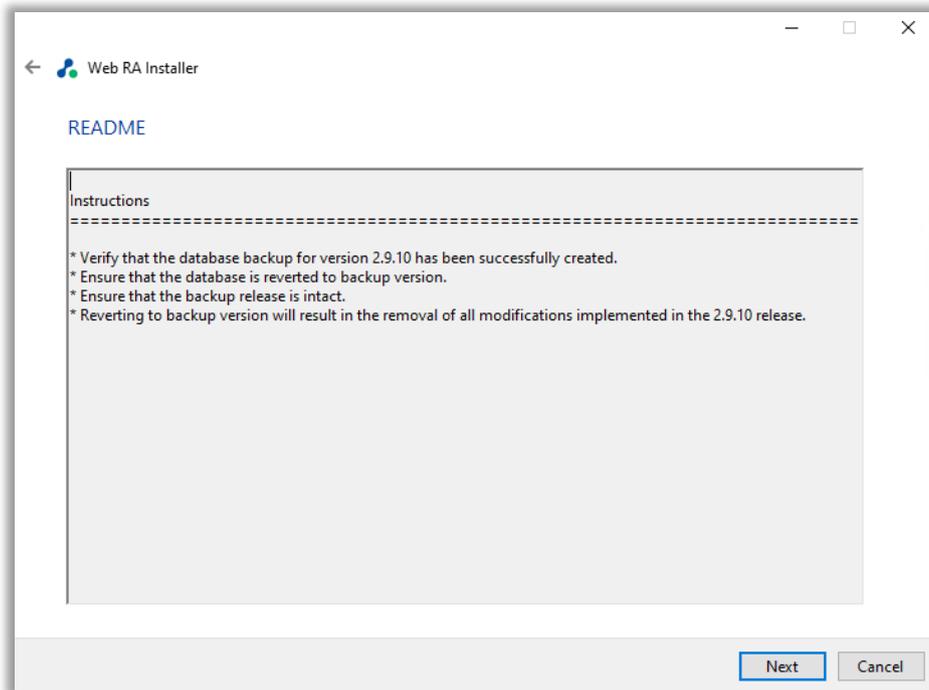
4.2.4 Click the 'Next' button to continue to the 'Installation Type' screen. Select the "Revert to previous release" option.



4.2.5 Then, click 'Next' button to view and accept the 'License Agreement'.

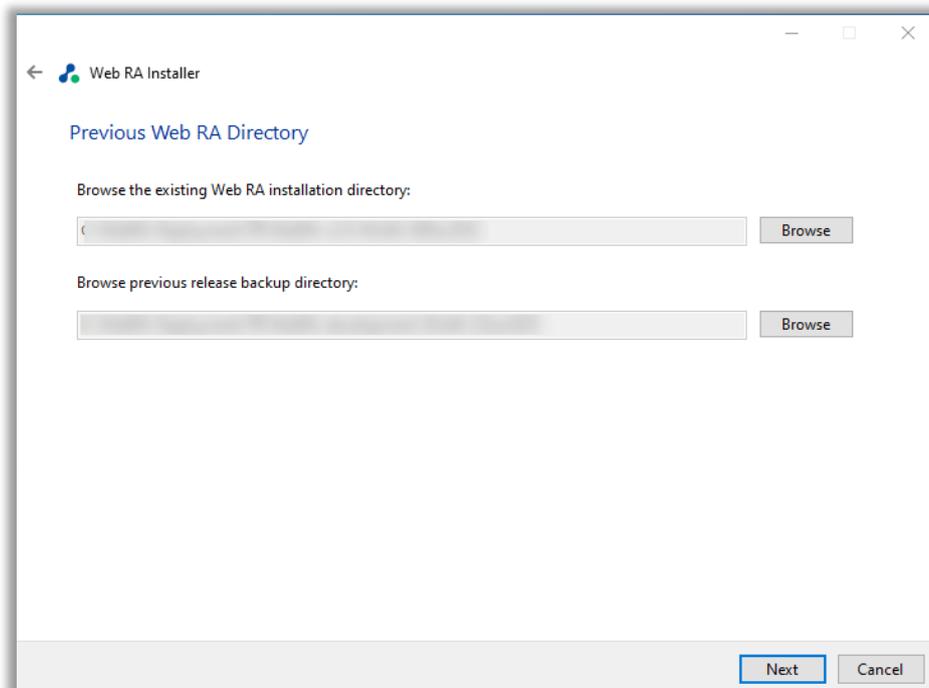


4.2.6 Click 'I Agree' to proceed to the 'Read Me' screen.

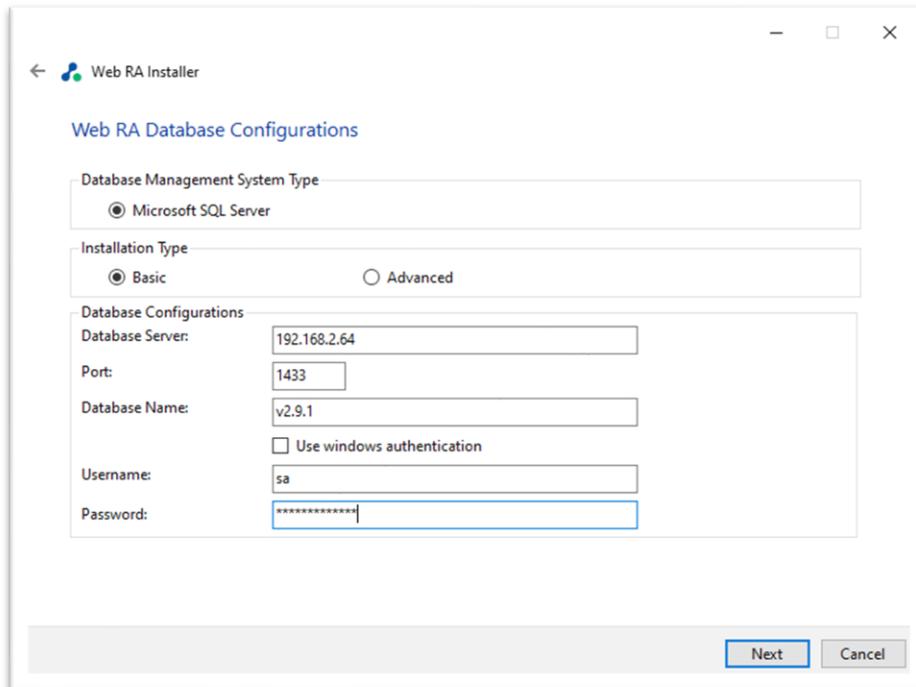


4.2.7 Click the 'Next' button to provide:

1. The existing Web RA installation directory.
2. Previous release backup directory, which will be set automatically. You also have the option to browse and select your own path.



4.2.8 Click the 'Next' button to view database details:

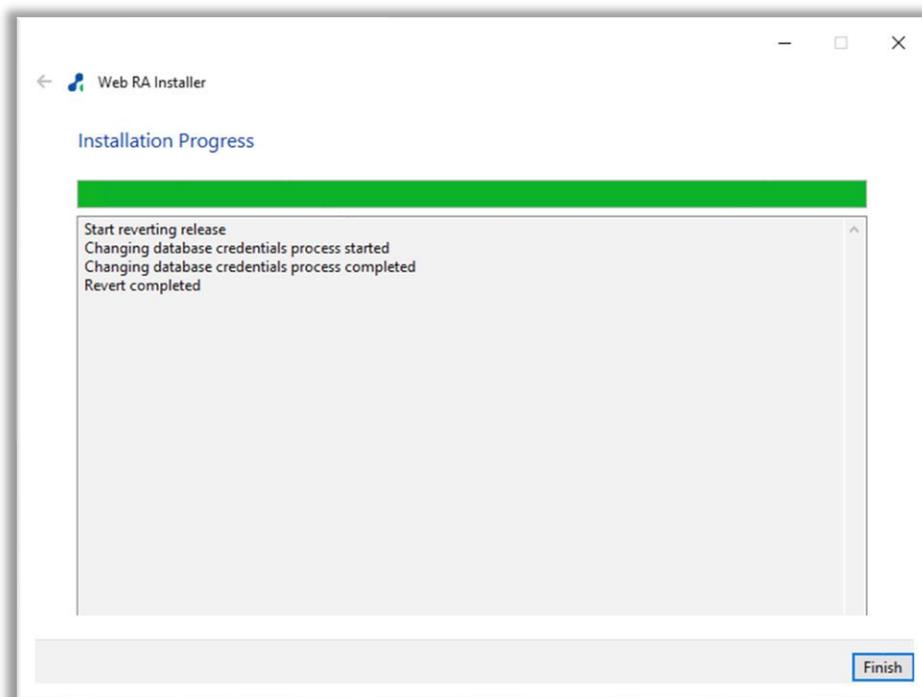


The screenshot shows the 'Web RA Database Configurations' window. It contains the following fields and options:

- Database Management System Type:**  Microsoft SQL Server
- Installation Type:**  Basic,  Advanced
- Database Configurations:**
  - Database Server: 192.168.2.64
  - Port: 1433
  - Database Name: v2.9.1
  - Use windows authentication
  - Username: sa
  - Password: [masked]

Buttons: Next, Cancel

4.2.9 Click 'Finish' to complete the installation process.



The screenshot shows the 'Web RA Installation Progress' window. It displays a green progress bar and the following log messages:

```
Start reverting release
Changing database credentials process started
Changing database credentials process completed
Revert completed
```

Button: Finish

## 5 ADSS Web RA Installation on Linux System

### 5.1 Prerequisites for Linux Installation

#### 5.1.1 Install and Setup .Net Runtime 9

Source: [Install .NET on RHEL and CentOS Stream - .NET | Microsoft Learn](#)

The ASP.NET Core Runtime allows you to run .Net applications that do not include the runtime. The following command installs the ASP.NET Core Runtime, which is the most compatible runtime for .NET.

#### Installation

In your terminal, run the following command:

```
Bash: sudo dnf install aspnetcore-runtime-9.0
```

Verify Installation by running the following command:

```
Bash: dotnet --info
```

#### 5.1.2 Install and Setup Nginx

Source: [nginx: Linux packages](#)

- First, start by ensuring your system is up-to-date.

```
Bash: sudo dnf clean all
Bash: sudo dnf update
Bash: sudo dnf groupinstall "Development Tools"
```

- **Installing Nginx on AlmaLinux 9.**

By default, Nginx is available on the AlmaLinux 9 base repository. Simply install the Nginx package by using the `dnf` command:

```
Bash: sudo dnf install nginx
```

- After the installation is complete, start the service of the Nginx server. Then, enable it so that it starts itself automatically with the system reboot:

```
Bash: sudo systemctl restart nginx
Bash: sudo systemctl status nginx
Bash: sudo systemctl enable nginx
```

- Configure Firewall.

```
Bash: sudo firewall-cmd --permanent --add-service=http
Bash: sudo firewall-cmd --permanent --add-service=https
Bash: sudo firewall-cmd --reload
```

- Accessing Nginx Web Interface

**i** Once the installation is successful, verify that the webserver is running and accessible by entering your server's IP address in a browser: `http://your-server-ip-address`. If you see this page, it means that your Nginx web server is correctly installed and is running on AlmaLinux 9.

### 5.1.3 Install CIFS Utilities

The “cifs-utils” package is required for mounting shared folders using the CIFS (Common Internet File System) protocol. During installation, the absence of this package may cause interruptions or mounting errors.

To install "cifs-utils", run the following command:

```
sudo apt install cifs-utils
```

Ensure this package is installed on the Linux system before proceeding with the Web RA installation.

#### 5.1.4 Java JRE Installation [Required for Certificate Signing Request (CSR) Verification]

**Need to install the Java JRE latest version on Linux machine for CSR policy verifications during certificate creation.**

The Java Runtime Environment (JRE) is required to support CSR policy checks when creating certificates through Web RA. Ensure OpenJDK 17 JRE is installed and properly configured on your Linux machine.

##### **On Ubuntu:**

1. Update your package list:

```
sudo apt update
```

2. Install OpenJDK 17 JRE:

```
sudo apt install openjdk-17-jre
```

3. Verify installation:

```
java -version
```

- Expected output should look like this:

```
openjdk version "17.0.x" ...
```

- Optional (if full JDK is needed):

```
sudo apt install openjdk-17-jdk
```

##### **On AlmaLinux:**

1. Update your package list:

```
sudo dnf update -y
```

2. Install OpenJDK 17 JRE:

```
sudo dnf install java-17-openjdk -y
```

3. Verify installation:

```
java -version
```

- Expected output should look like this:

```
openjdk version "17.0.x" ...
```

- Optional (if multiple Java versions are installed):

```
sudo alternatives --config java
```

You will be prompted to select the default version.

## 5.2 Pre-Installation Steps

### 5.2.1 Access the Root Directory

On a Linux machine, the **root directory** (/) is the highest-level directory that contains all system files and user directories.

### 5.2.2 Locate the `/var` Folder

- The `/var` directory is used to store variable data such as logs, cache, and web files.
- Navigate to this directory inside the root folder (`/var`).

### 5.2.3 Check for the `www` Folder

- Inside `/var`, check for the **www** folder.
- Some Linux distributions automatically create this folder, but in some cases, you might need to create it manually.

#### If the `www` folder is not present:

- Create a new `www` folder inside `/var`.
- Ensure appropriate permissions are set so that the installation can proceed without issues.

### 5.2.4 Place the Installation Package

- Copy the extracted WebRA installation package into the `/var/www/` directory.

### 5.2.5 Access the Installation Folder

- In the extracted package, navigate to the `LinuxFresh` folder.
- Then, go to `/var/www/LinuxFresh/setup/bin/` to access the `install.json` file.
- Each parameter in the `install.json` file must be correctly configured before proceeding with the installation.

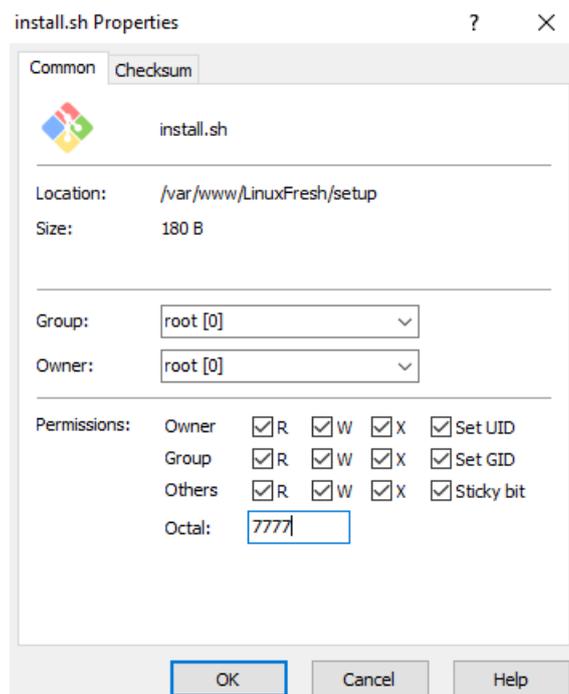
### 5.2.6 Set Execution Permissions for the Installation Script

Before starting the installation, the install.sh file must have execution permissions enabled.

Name	Size	Changed	Rights	Owner
..		3/11/2025 1:06:44 PM	rwxr-sr-x	root
bin		3/11/2025 1:04:26 PM	rwxr-sr-x	root
cert-linting		3/11/2025 1:04:26 PM	rwxr-sr-x	root
certs		3/11/2025 1:04:28 PM	rwxr-sr-x	root
db-scripts		3/11/2025 1:35:49 PM	rwxr-sr-x	root
executable		3/11/2025 1:05:01 PM	rwxr-sr-x	root
license		3/11/2025 1:05:01 PM	rwxr-sr-x	root
logo		3/11/2025 1:05:01 PM	rwxr-sr-x	root
third-party		3/11/2025 1:05:02 PM	rwxr-sr-x	root
install.log	5 KB	3/11/2025 1:41:42 PM	rw-r--r--	root
install.sh	1 KB	3/11/2025 1:34:43 PM	rwsrwsrwt	root

To grant execution permissions:

- Locate the install.sh file inside **/var/www/LinuxFresh/setup/**.
- Right-click the file and select Properties.
- Go to the Permissions section.
- Grant permissions and click Ok.



## 5.3 Configuring Installation Parameters in install.json file

The install.json file contains all the required settings for the Web RA installation. The operator must define these configurations correctly before proceeding with the installation. The installation process reads this file to determine how the setup should be performed.

Each parameter in install.json must be configured according to your system requirements. The following sections explain each parameter in detail:

### 5.3.1 Set Agreement Parameter

- The LicenseAgreement parameter must be set to true if you want to include an agreement confirmation step in the installation process. This confirms acceptance of Ascertia's licensing terms and conditions.
- Possible values: true or false.
- If set to false, the installation will proceed without an explicit agreement confirmation.

```
{
  "Agreement": {
    "LicenseAgreement": true,
    "comment": "possible values are TRUE/FALSE"
  },
}
```

### 5.3.2 Installation Modes

This section defines the type of installation to be performed. Choosing the correct mode is essential for a successful setup.

Possible values:

- **REGULAR\_RELEASE:** Installs a regular update package.
- **UNINSTALL\_REGULAR\_RELEASE:** Removes a previously installed update.



*1. Database and SMTP configuration details will not appear in the "install.json" file after the installation of Web RA is complete on the Linux machine.*

*2. If you are installing a regular release update, make sure to use the same site names and port numbers that were used during the original installation.*

```
},
  "InstallationMode": {
    "Type": "",
    "comment": "Possible values are
FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE/REGULAR_RELEASE/UNINSTALL_REGULAR_RELEASE/CHANGE_DB_CREDENTIALS/UNINSTALL"
  },
  "ExistingInstallation": {
    "BackupDirectory": ""
  }
}
```

### 5.3.2.1 Installing Regular Release

When installing a regular release update, set the “Type” value under “InstallationMode” to: “REGULAR\_RELEASE”.

```
{
  "Agreement": {
    "LicenseAgreement": true,
    "comment": "Possible values are True or False"
  },
  "InstallationMode": {
    "Type": "REGULAR_RELEASE",
    "comment": "Possible values are FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE/REGULAR_RELEASE/UNINSTALL_REGULAR_RELEASE/CHANGE_DB_CREDENTIALS/UNINSTALL"
  },
}
```

You must also provide the following parameter for the regular release process to complete successfully:

**ExistingWebRAPath** - This is the location where your current Web RA is installed. The system needs this path to find and remove the regular release. **For example:** existing installation directory/.

```
{
  "Agreement": {
    "LicenseAgreement": true,
    "comment": "Possible values are True or False"
  },
  "InstallationMode": {
    "Type": "REGULAR_RELEASE",
    "comment": "Possible values are FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE/REGULAR_RELEASE/UNINSTALL_REGULAR_RELEASE/CHANGE_DB_CREDENTIALS/UNINSTALL"
  },
  "ExistingInstallation": {
    "BackupDirectory": ""
  },
  "RegularInstallation": {
    "ExistingWebRAPath": "",
    "RegularBackupPath": ""
  },
  "SampleData": true,
  "comment": "Possible values are True or False",
  "DatabaseConfiguration": {
    "ReadFromEnvironment": false,
    "comment_env": "If true, database information will be read from environment variables instead of configuration file values. Expected variables: CONNECTIONSTRINGS__DBPROVIDER, CONNECTIONS",
    "connectionProviderType": "PGSQL",
    "comment": "Possible values are MSSQL, PGSQL",
    "configurationType": {
      "Type": "TYPICAL",
      "comment": "Possible values are TYPICAL and ADVANCED",
      "typicalDatabaseConfiguration": {
        "MachineName": "",
        "Port": "",
        "DatabaseName": "",
        "UserId": "",
        "Password": ""
      }
    },
    "AdvancedDatabaseConfiguration": {
      "connectionString": "",
      "comment": "Possible values are e.g. MSSQL Authentication and PGSQL Authentication",
      "authentication": "PGSQL",
      "data source=[server address];initial catalog=[database name];user id=[user_id];password=[password];MultipleActiveResultSets=True;Pooling=true;"
    }
  }
}
```

After setting the Type, save the file and close it. Then navigate to **the /var/www/LinuxFresh/setup/** folder and run the install.sh script.

**Note:** Before executing /install.sh, run the following commands:

```
dos2unix install.sh
```

```
cat -A install.sh
```

After running the above given commands, launch the /install.sh file by running the following command:

```
sudo ./install.sh
```

```
Console
Enter command: ./install.sh
Do not execute commands that require user-input or data transfer
Current directory: /var/www/WebRA-development-Lin64-23Dec2025/setup

/var/www/WebRA-development-Lin64-23Dec2025/setup$ ./install.sh
[34m[INFO] Starting ADSS Web RA Installer[0m
[34m[INFO] Checking Prerequisites ADSS Web RA[0m
[34m[INFO] Validating required system components...[0m
[34m[INFO] Verifying .NET Runtime...[0m
[37m[DEBUG] .NET Runtime 9.0 is installed.[0m
[34m[INFO] Verifying Nginx...[0m
[37m[DEBUG] Nginx is installed.[0m
[34m[INFO] Verifying cifs-utils...[0m
[37m[DEBUG] cifs-utils is installed.[0m
[34m[INFO] Verifying ADSS Web RA installer configuration file ..[0m
[37m[DEBUG] License Agreement: Agree[0m
[37m[DEBUG] Installation Mode: REGULAR_RELEASE[0m
[34m[INFO] Extracting Connection Provider type form existing installation ..[0m
[37m[DEBUG] Connection Provider type form existing installation : PGSQL[0m
[34m[INFO] Starting Web RA regular-release installation process.[0m
[37m[DEBUG] Starting release apply process. BaseVersion: 2.9.9, TargetVersion: 2.9.10[0m
[37m[DEBUG] Creating a backup of the existing Web RA installation directory[0m
[37m[DEBUG] Applying release v2.9.9 files to path: /var/www/WebRA-v2.9.7-RR-Lin64-27Jun2025[0m
[37m[DEBUG] Please wait while migrating the certificate to the new structure[0m
[37m[DEBUG] Migrating the certificate to the new structure completed[0m
[37m[DEBUG] Web RA release applied successfully[0m
```

### 5.3.2.2 Uninstalling a Regular Release

To remove a previously installed regular release update modify the **install.json** file and set the Type under "InstallationMode" to: "UNINSTALL\_REGULAR\_RELEASE"

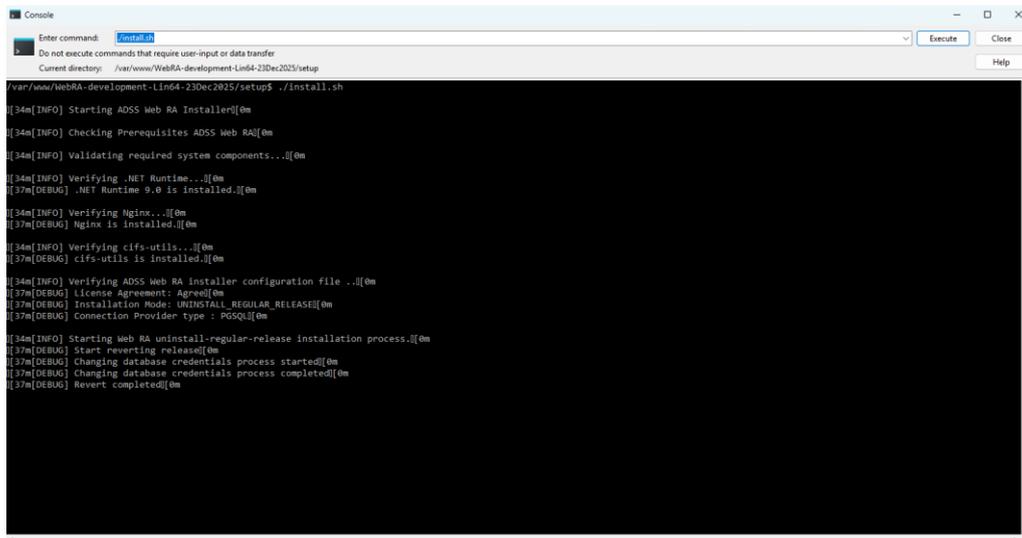
```
{
  "Agreement": {
    "LicenseAgreement": true,
    "comment": "Possible values are True or False"
  },
  "InstallationMode": {
    "Type": "UNINSTALL_REGULAR_RELEASE",
    "comment": "Possible values are FIRST_TIME/LOAD_BALANCE/UPGRADE/EXISTING_DATABASE/REGULAR_RELEASE/UNINSTALL_REGULAR_RELEASE/CHANGE_DB_CREDENTIALS/UNINSTALL"
  },
  "ExistingInstallation": {
    "BackupDirectory": ""
  },
  "RegularInstallation": {
    "ExistingWebRAPath": "",
    "RegularBackupPath": ""
  },
  "SampleData": true,
  "comment": "Possible values are True or False",
  "DatabaseConfiguration": {
    "ConnectionProviderType": "MSSQL",
    "comment": "Possible values are MSSQL, PGSQL",
    "ConfigurationType": {
      "Type": "TYPICAL",
      "comment": "Possible values are TYPICAL and ADVANCED",
      "TypicalDatabaseConfiguration": {
        "MachineName": "",
        "Port": "",
        "DatabaseName": "",
        "UserId": "",
        "Password": ""
      }
    }
  }
}
```

You must also provide the following two parameters for the uninstallation process to complete successfully:

**ExistingWebRAPath** - This is the location where your current Web RA is installed. The system needs this path to find and remove the regular release. **For example:** existing installation directory/.

**RegularBackupPath** - This is where a backup of the current Web RA will be saved before the uninstallation starts. It helps you restore things in case something goes wrong. **For example:** existing directory/backup directory/backup folder.

After setting the required values, save the file and close it. Then navigate to **the /var/www/LinuxFresh/setup/** folder and run the install.sh script.



```

Enter command: install.sh
Do not execute commands that require user-input or data transfer
Current directory: /var/www/WebRA-development-Lin64-23Dec2025/setup
/var/www/WebRA-development-Lin64-23Dec2025/setup$ ./install.sh
[34m[INFO] Starting ADSS Web RA Installer[0m
[34m[INFO] Checking Prerequisites ADSS Web RA[0m
[34m[INFO] Validating required system components...[0m
[34m[INFO] Verifying .NET Runtime...[0m
[37m[DEBUG] .NET Runtime 9.0 is installed.[0m
[34m[INFO] Verifying Nginx...[0m
[37m[DEBUG] Nginx is installed.[0m
[34m[INFO] Verifying cifs-utils...[0m
[37m[DEBUG] cifs-utils is installed.[0m
[34m[INFO] Verifying ADSS Web RA installer configuration file ...[0m
[37m[DEBUG] License Agreement: Agreed[0m
[37m[DEBUG] Installation Mode: UNINSTALL_REGULAR_RELEASE[0m
[37m[DEBUG] Connection Provider type : PGSQL[0m
[34m[INFO] Starting Web RA uninstall-regular-release installation process.[0m
[37m[DEBUG] Start reverting release[0m
[37m[DEBUG] Changing database credentials process started[0m
[37m[DEBUG] Changing database credentials process completed[0m
[37m[DEBUG] Revert completed[0m

```

### 5.3.3 Existing Installation Parameter

**BackupDirectory:** Specifies where to store backup files before upgrading or uninstalling Web RA. If left empty, no backup is created, which may lead to data loss.

```
},
"ExistingInstallation": {
  "BackupDirectory": ""
},
"RegularInstallation": {
  "ExistingWebRAPath": "",
  "RegularBackupPath": ""
```

### 5.3.4 Regular Installation Parameter

- **ExistingWebRAPath:** Specifies the file path where the currently installed Web RA instance is located. **For example:** existing installation directory/.
- **RegularBackupPath:** Directory where a backup of the current Web RA instance will be stored before installation. **For example:** existing directory/backup directory/.

```
},
"ExistingInstallation": {
  "BackupDirectory": ""
},
"RegularInstallation": {
  "ExistingWebRAPath": "",
  "RegularBackupPath": ""
```

### 5.3.5 Sample Data

- If set to **True**, the installation will include sample data to help with testing and initial configuration. The following items will be included in the sample data:
  - Default ADSS Connector
  - Default SMTP Connector
  - Default ADSS Service Profile
  - Default Subscriber Agreement
  - Default Vetting Form
  - Default Service Plan
  - Default Authentication Profile
- If set to **False**, the installation will proceed without adding sample data and you will have to create everything by scratch.

#### 5.3.5.1 Database Configuration

- **ConnectionProviderType:**

Defines the type of database server that Web RA will connect to.

- **Possible values:**
  - **MSSQL** — Use Microsoft SQL Server as the database.
  - **PGSQL** — Use PostgreSQL as the database.

- **ConfigurationType:**

Specifies how the database connection will be configured during installation.

- **Possible values:**
  - **TYPICAL** — Uses default, commonly required settings with minimal manual input.
  - **ADVANCED** — Allows manual editing of the full database connection string or additional custom settings.

**Note:** You must choose either **Typical** or **Advanced** configuration. Both cannot be used at the same time.

##### 5.3.5.1.1 Typical Database Configuration

When you choose TYPICAL as the configuration type, you need to provide the following details:

<b>Machine Name</b>	The hostname or IP address of the database server that will host the Web RA database
<b>Port</b>	The port number used for connecting to the database. <ul style="list-style-type: none"> <li>• For Microsoft SQL Server the default port is 1433</li> <li>• For PostgreSQL Server the default port is 5432</li> </ul>
<b>Database Name</b>	The name of the database to be created or used by Web RA
<b>UserId</b>	The database username that Web RA will use to authenticate and connect to the database.
<b>Password</b>	The password for database authentication.

```

    },
    "SampleData": true,
    "comment": "Possible values are True or False",
    "DatabaseConfiguration": {
      "ConnectionProviderType": "",
      "comment": "Possible values are MSSQL, PGSQL",
      "ConfigurationType": {
        "Type": "TYPICAL",
        "comment": "Possible values are TYPICAL and ADVANCED",
        "TypicalDatabaseConfiguration": {
          "MachineName": "",
          "Port": "",
          "DatabaseName": "",
          "UserId": "",
          "Password": ""
        }
      }
    }
  },

```

### 5.3.6 Advanced Database Configuration

This option allows you to provide a custom connection string for full control over how Web RA connects to your database. Use this if you need to define specific connection parameters beyond what the "Typical" configuration allows.

#### Supported database types:

- MSSQL
- PGSQL

Below are example connection strings for each supported database:

#### Example for MSSQL Authentication

```
data source=[server address];initial catalog=[database name];user
id=[user_id];password=[password];MultipleActiveResultSets=True;Pooling=true;
```

#### Example for PGSQL Authentication

```
Host=[server address];Port=[server port];Database=[database
name];Username=[username];Password=[password];Pooling=true;SSL Mode=Disable;Trust Server
Certificate=true;
```

#### Note:

Make sure to replace the placeholders (e.g., [server address], [database name], [username], [password]) with the actual details for your database server.

```

    },
    "AdvancedDatabaseConfiguration": {
      "connectionString": "",
      //MSSQL Authentication
      //data source=[server address];initial catalog=[database name];user id=[user_id];password=[password];MultipleActiveResultSets=True;Pooling=true;",
      //PGSQL Authentication
      //RAEntities: "Host=[server address];Port=[server port];Database=[database name];Username=[username];Password=[password];Pooling=true;SSL Mode=Disable;Trust
Server Certificate=true;"
      "comment": "Possible values are e.g. MSSQL Authentication and PGSQL Authentication"
    }
  }

```

### 5.3.7 Custom Installation Parameter

Defines the modules to be installed and their respective configurations.

- **FullyQualifiedDomainName:** Specifies the full domain name of the server.

Each module has settings for site name, installation status, and ports.

- **AdminModule:**
  - Site name: admin
  - Install: true
  - Port: "Port Number" (default HTTPS port)
  - Application Port: "Port Number"
- **WebModule:**
  - Site name: web
  - Install: true
  - Port: "Port Number"
  - Application Port: "Port Number"
- **ApiModule:**
  - Site name: api
  - Install: true
  - Port: "Port Number"
  - Application Port: "Port Number"
- **DeviceModule** (SCEP support):
  - Site name: device
  - Install: true
  - Port: "Port Number"
  - Application Port: "Port Number"
- **HTTPSDeviceModule** (Secure communication for SCEP, CMP, ACME, EST):
  - Site name: https-device
  - Install: true
  - Port: "Port Number"
  - Application Port: "Port Number"
- **SSLDeviceModule** (EST on client authentication-based setup):
  - Site name: ssl-device
  - Install: true
  - Port: "Port Number"
  - Application Port: "Port Number"

```

},
"CustomInstallation": {
  "FullyQualifiedDomainName": "",
  "AdminModule": {
    "siteName": "admin",
    "install": true,
    "port": ████████,
    "applicationPort": ████████
  },
  "WebModule": {
    "siteName": "web",
    "install": true,
    "port": ████████,
    "applicationPort": ████████
  },
  "ApiModule": {
    "siteName": "api",
    "install": true,
    "port": ████████,
    "applicationPort": ████████
  },
},

```

```

},
//SCEP
"DeviceModule": {
  "siteName": "device",
  "install": true,
  "port": ■■■,
  "applicationPort": ■■■
},
//Install SCEP,CMP,ACME,EST
"HTTPSDeviceModule": {
  "siteName": "https-device",
  "install": true,
  "port": ■■■,
  "applicationPort": ■■■
},
//Instal EST on client Authentications based
"SSLDeviceModule": {
  "siteName": "ssl-device",
  "install": true,
  "port": ■■■,
  "applicationPort": ■■■
}
}

```

### 5.3.7.1 Port Usage Guidelines

- The same port number cannot be assigned to multiple modules. If a port is already in use, a different number must be selected for another module.
- In the application ports, if using a sequential series (e.g., **5001, 5002, 5003**), the next installation should use a different series (e.g., **4001, 4002, 4003**) to prevent conflicts.

#### Constraints

- *Windows Enrolment and Active Directory are not supported in Linux deployment.*

### 5.3.7.2 Allowing Ports on Ubuntu

If the Linux server is running **Ubuntu**, use the following command to allow a specific port:

```
sudo ufw allow <port>/tcp
```

**For example, to allow port 81:**

```
sudo ufw allow 81/tcp
```

To verify the firewall status:

```
sudo ufw status
```

### 5.3.7.3 Allowing Ports on AlmaLinux

If the server is running **AlmaLinux**, use the following command:

```
sudo firewall-cmd --permanent --add-port=<port>/tcp
```

For example, to allow **port 443:**

```
sudo firewall-cmd --permanent --add-port=443/tcp
```

After making changes, reload the firewall settings:

```
sudo firewall-cmd --reload
```

### 5.3.8 SMTP Configuration

Defines email settings for notifications:

- **Host:** SMTP server address (e.g., smtp.example.com).
- **Port:** SMTP connection port (e.g., 587 for TLS, 465 for SSL).
- **FromAddress:** Sender's email address.
- **Username** and **Password:** SMTP authentication credentials.
- **UseSsl:** Determines if SSL/TLS encryption is enabled.

**Note:** When SMTP settings are configured in the installation process, an SMTP connector is automatically created upon running the installer.

```
},
"smtpConfiguration": {
  // The hostname or IP address of the SMTP server (e.g., ██████████)
  "Host": "",
  // The port number used for the SMTP connection (e.g., ████████)
  "Port": ████,
  // The email address that appears as the sender
  "FromAddress": "",
  // Default subject line for the email
  "DefaultSubject": "",
  // The default recipient email address
  "DefaultRecipient": "",
  // The username for authenticating with the SMTP server
  "Username": "",
  // The password for authenticating with the SMTP server
  "Password": "",
  // Indicates if authentication is required for the SMTP server
  "IsAuthenticationRequired": true,
  // Indicates if SSL/TLS should be used for the SMTP connection
  "UseSsl": true
}
}
```

After configuring all necessary parameters in the install.json file, launch the /install.sh file to install ADSS Web RA with the required set of configurations.

## 6 Appendix

### 6.1 Troubleshooting

6.1.1 If ADSS Web RA Admin module is installed on Windows 2012 R2, then the HTTP 403.16 error code may occur when you access the ADSS Web RA Admin console from web browser.

Follow these instructions to solve this issue:

- a. Open registry and add the key:
  - KEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL
- b. Create a new key with **Value Type: REG\_DWORD (32-bit)**
- c. Set **Value Name: ClientAuthTrustMode**
- d. Edit the field and set **Value Data: 2**

If you are interested to know more details about it, browse the Microsoft KB link:

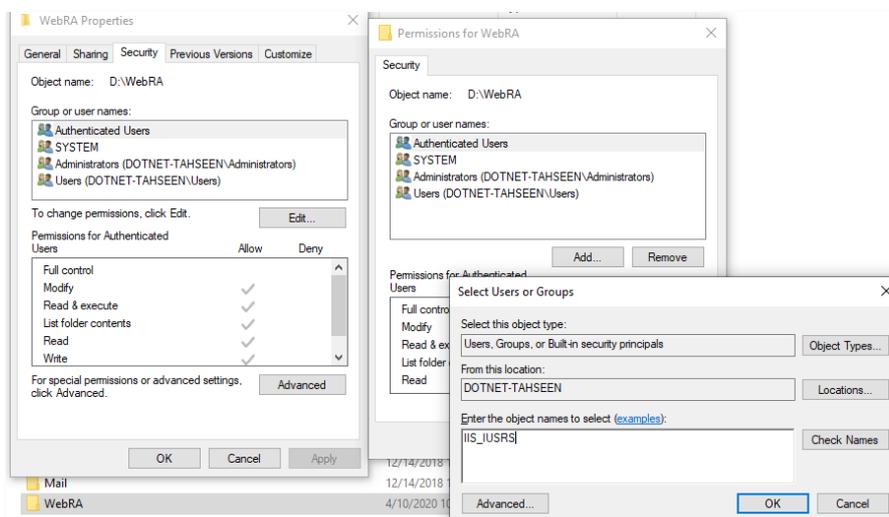
<https://support.microsoft.com/en-us/kb/2464556>.

6.1.2 If you receive the HTTP error code 500.19 whilst accessing Admin, Web or API then:

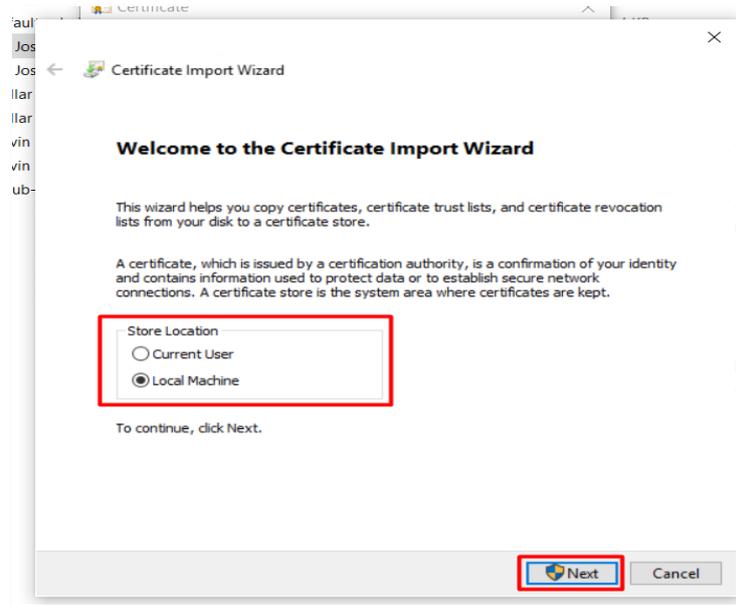
- a. Open IIS Management Console.
- b. Go to Application Pools.
- c. Select a site and click Advanced Setting.
- d. In General, make sure that Enable 32-Bit Applications is set to False.

6.1.3 If you cannot start ADSS Server from Windows Services panel on Azure, then make sure that you are not starting those services under Windows user that you have created while creating the Azure instance. You must create another Windows user with Administrative rights and start the services under that user.

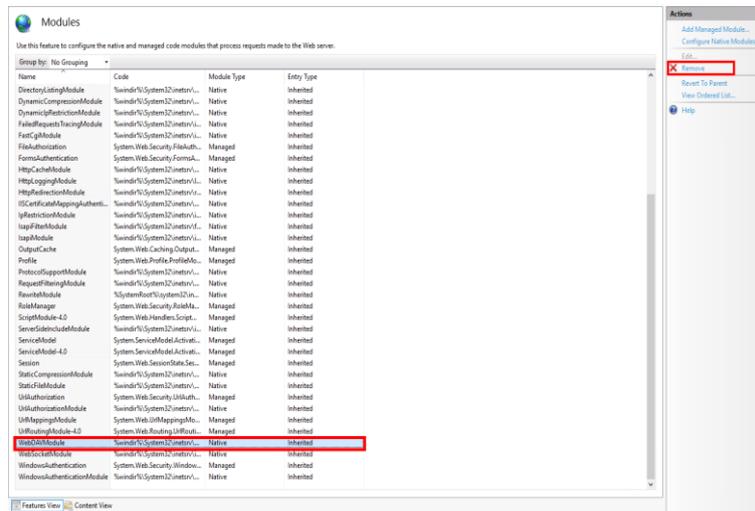
- Upon deploying to the server, you must keep in mind that the firewall and ports are open so that user can access the application from outside.
  - In **Firewall > Outbound Rules**. Open the ports if you want to 80-90, 440-450.
- Make sure the Directory has IIS permissions where code files are published.



- **Add / Install the SSL Server certificate in Microsoft Management Console** which will be imported to IIS so, connection between server and application can be established successfully.



- For API to work against all Verbs (GET,POST,DELETE,PUT etc) without 405 error, make sure WebDav Module remove against the API site.To do this click on “API” site in IIS ,select “Modules”, find the “WebDAVModule” and remove it.



## 6.2 Troubleshooting for Linux

- **Deployment Stops Unexpectedly**

If the deployment process halts during execution, it may be due to Linux security settings preventing the installation from proceeding. To resolve this issue:

1. Temporarily disable Linux security enforcement by running the following command:

```
sudo setenforce 0
```

This forces the security module to be turned off.

2. Test the nginx configuration by running the following command:

```
sudo nginx -t
```

This command tests your nginx configuration without actually starting or restarting the server.

- It checks for syntax errors in your nginx configuration files.
- Validates file paths (e.g., certs, keys, includes).
- Ensures nginx won't crash when restarted.
- It is a safe way to debug before restarting a live server.

3. Restart the nginx service to ensure proper functionality:

```
systemctl restart nginx
```

- **A Specific Web RA Service is Not Running**

If the deployment completes but a specific service (such as **Admin**, **Web**, or **API**) is not running, restart the affected service using the following command:

```
systemctl restart kestrel-webra-{service name}.service
```

Replace {service name} with the actual service name (e.g., **admin**, **web**, **api**).

- **Installation Fails Due to Spaces in Folder Name**

**Issue:**

The installation process fails or encounters errors if the folder name where the installation package is placed contains spaces.

**Solution:**

Ensure that the installation folder name does not contain spaces. Rename the folder using underscores (\_) or remove spaces before proceeding with the installation.

- **Nginx is Inactive or Not Running**

If Nginx is inactive, Web RA will not be accessible in the browser. Check the service status and restart it if necessary.

**Symptom:**

Active: inactive (dead)

**Solution:**

1. Test the nginx configuration by running the following command:

```
sudo nginx -t
```

This command tests your nginx configuration without actually starting or restarting the server.

- It checks for syntax errors in your nginx configuration files.
- Validates file paths (e.g., certs, keys, includes).
- Ensures nginx won't crash when restarted.
- It is a safe way to debug before restarting a live server.

2. Then, restart the nginx service using the following command:

```
sudo systemctl start nginx
```

- **413 Request Entity Too Large – API or File Upload Failure**

In case of executing APIs with large datasets or files, if the following error appears, apply the configuration and commands below to resolve it:

**Error:**

“413 Request Entity Too Large”

This error occurs when a client (such as a browser or an API request) tries to upload data exceeding the allowed size limit configured in nginx.

**Resolution:** Increase the `client_max_body_size` limit in nginx.

**1. Open the nginx Configuration File**

Edit your main nginx configuration file (`/etc/nginx/nginx.conf`) or the specific site configuration in `/etc/nginx/sites-available/your-site.conf`:

```
sudo nano /etc/nginx/nginx.conf
```

**2. Increase “client\_max\_body\_size”**

Add or modify this directive inside the `http`, `server`, or `location` block:

```
http {
    client_max_body_size 100M;
}
```

### 3. Test the Configuration

```
nginx -t
```

### 4. Reload nginx to apply changes

```
sudo systemctl reload nginx
```

- **License Upload Error on AlmaLinux Due to SHA-1 Restriction**

If you encounter an error while uploading the application license on an AlmaLinux machine, you need to enable the SHA-1 algorithm. Once enabled, the license upload will work successfully.

#### **How to Enable SHA-1 Algorithm on CentOS Stream 9 / AlmaLinux 9 / RockyLinux 9:**

To fix this, you need to enable the SHA-1 algorithm in your modern OS, for example in EL9 / CentOS 9. To enable it, run the following command:

```
update-crypto-policies --set DEFAULT:SHA1
```

## 6.3 Configurations used for Simple Certificate Enrollment Protocol (SCEP)

6.3.1 Make sure that following tag is added in “web.config” of web module:

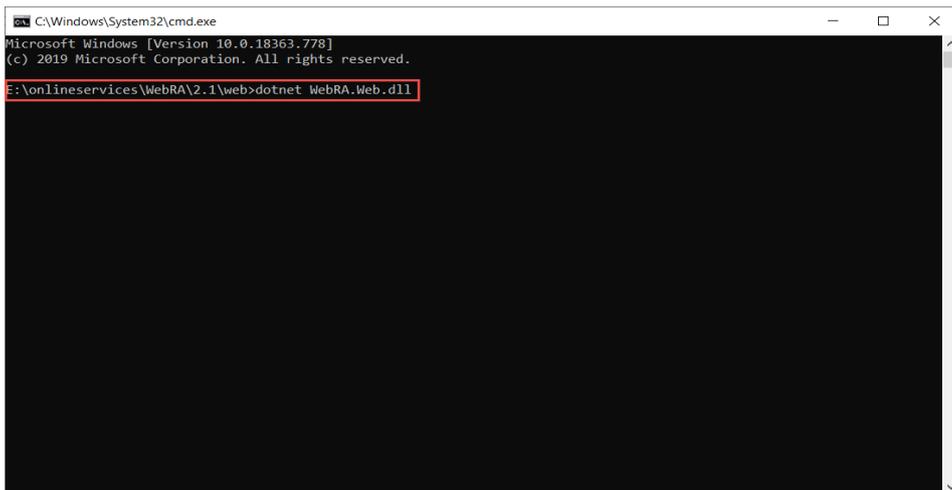
```
<security>
  <requestFiltering>
    <requestLimits maxQueryString="8192"/>
  </requestFiltering>
</security>
```

```
<configuration>
  <location path="." inheritInChildApplications="false">
    <system.webServer>
      <handlers>
        <add name="aspNetCore" path="*" verb="*" modules="AspNetCoreModuleV2" resourceType="Unspecified" />
      </handlers>
      <aspNetCore processPath="dotnet" arguments=".\WebRA.Protocol.dll" stdoutLogEnabled="true" stdoutLogFile=".\logs\stdout">
      </aspNetCore>
      <security>
        <requestFiltering>
          <requestLimits maxQueryString="8192" />
        </requestFiltering>
      </security>
    </system.webServer>
  </location>
</configuration>
<!--ProjectGuid: 31d1b205-525a-481e-bd32-4378e4f6559d-->
```

SCEP server URL that will be used for router will be:

- “[Server URL]/scep” e.g “<https://beta.web.ra.signinghub.com/scep>”
- Update URL value in Expect-CT header in “web.config” for web and admin modules according to your deployment URL. e.g. **<add name="Expect-CT" value="max-age=0, report-uri='https://adminra.signinghub.com'" />**

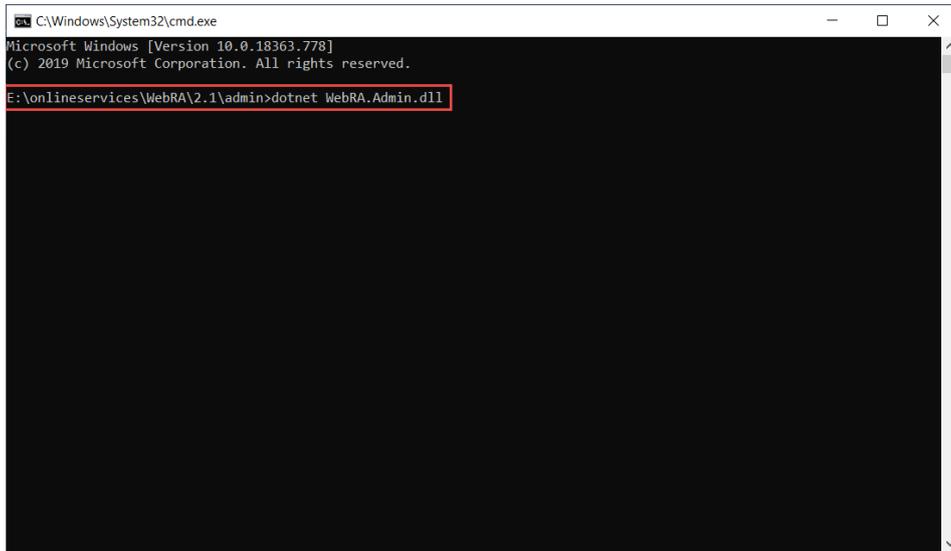
To test if the code is working properly for web, run command line in [installation-dir]/web and type following command:



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

E:\onlineservices\WebRA\2.1\web\dotnet WebRA.Web.dll
```

To test if the code is working properly for admin, run command line in [installation-dir]/admin and type following command:



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

E:\onlineservices\WebRA\2.1\admin>dotnet WebRA.Admin.dll
```

## 6.4 SSL Certificates

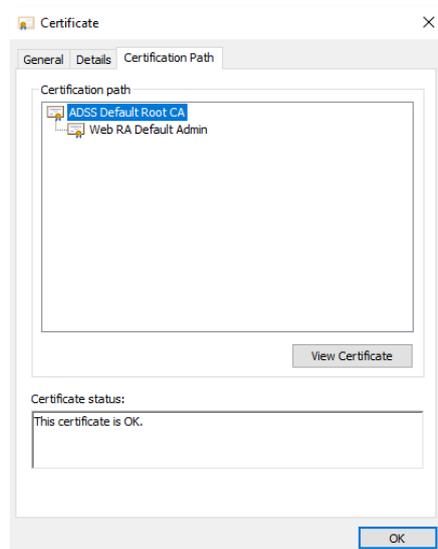
ADSS Web RA is a web application that is hosted in IIS. It is recommended to secure the communication between the server and browsers by using SSL over HTTPS. It is also recommended to use an SSL certificate issued by a well-known certificate authority (CA) e.g., Comodo, Symantec, Digicert, etc.

The Administrators portal can be accessed only via TLS client authentication. A default TLS client certificate is already packaged into ADSS Web RA.

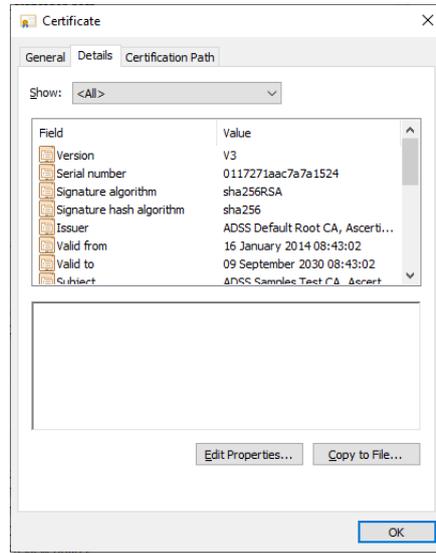
### 6.4.1 Exporting Root and Intermediate Certificates

6.4.2 In the [installation\_dir]/setup/certs directory there are two files with the name *web-ra-default-admin.cer* and *web-ra-default-admin.pfx*. TLS certificate is installed, but root certificates are not validated by the machine. To validate it, root certificate needs to be imported in the certificate store.

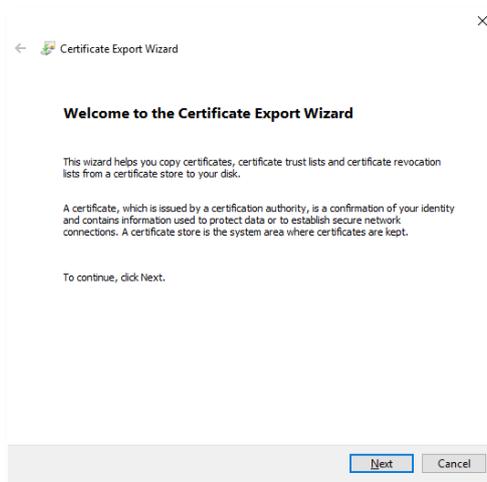
6.4.3 Double click the *web-ra-default-admin.cer* file



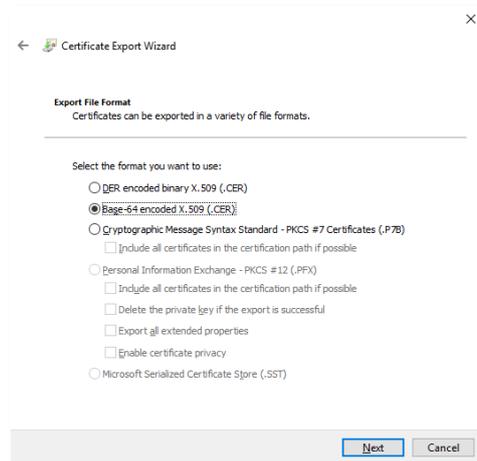
6.4.4 Select the Certification Path tab from the top. The default ADSS Web RA TLS certificate has one root certificate. Select the root certificate and click the View Certificate button. A new window will appear showing general details of the intermediate certificate.



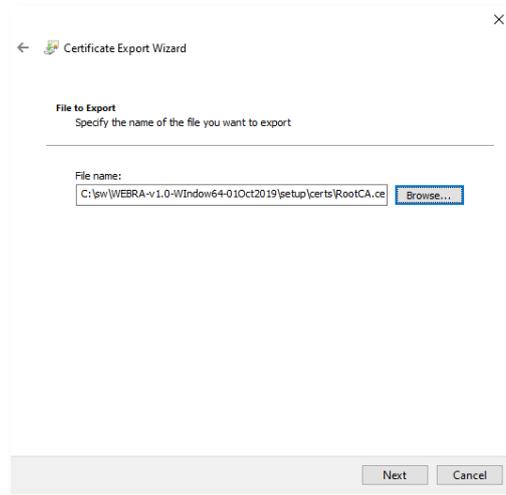
6.4.5 Select the Details tab from the top and click Copy to File. This will initiate the certificate export wizard.



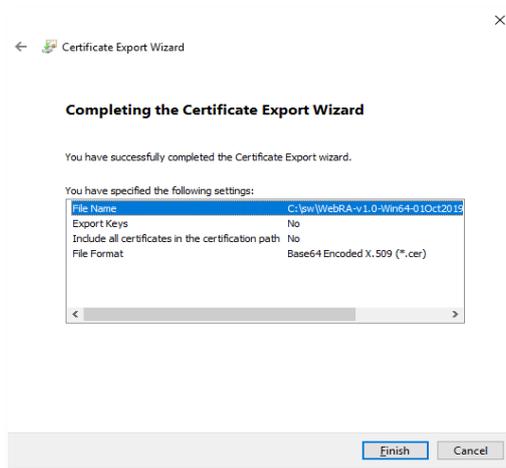
6.4.6 Click Next.



#### 6.4.7 Select the Base-64 encoded X.509 (.CER) option and click Next



#### 6.4.8 Choose a path where you want to save the certificate file for the intermediate certificate, and click Next.



#### 6.4.9 Click Finish to complete the root certificate export process.

## 6.5 SSL Configuration for Linux

After installation, SSL certificates must be configured to enable secure communication for Web RA. Follow these steps to configure SSL:

### 6.5.1 Navigate to the nginx configuration directory:

The configuration for the SSL device module is stored in different locations depending on the operating system:

- On **Ubuntu**, it is stored in the `/etc/nginx/sites-available/` directory
- On **AlmaLinux**, it is stored in the `/etc/nginx/conf.d/` directory.

Open the configuration file with a text editor:

#### On Ubuntu:

```
sudo nano /etc/nginx/sites-available/webra-ssl-device
```

#### On AlmaLinux:

```
sudo nano /etc/nginx/conf.d/webra-ssl-device.conf
```

**Note:** 'webra-ssl-device' is the file name. The file name will vary depending upon your particular configuration.

### 6.5.2 Locate the SSL Configuration Block:

Inside this file, find the section where the SSL certificate and key are defined. It should look similar to this:

```
ssl_certificate "/var/www/Linux_ED/setup/certs/EST-Server.crt";
ssl_certificate_key "/var/www/Linux_ED/setup/certs/EST-Server.key";
```

### 6.5.3 Update the Certificate Paths:

Modify these lines to point to the correct certificate and key locations:

```
ssl_certificate "/var/www/Linux_Fresh/setup/certs/EST-Server.crt";
ssl_certificate_key "/var/www/Linux_Fresh/setup/certs/EST-Server.key";
```

After updating the paths, save and exit the file.

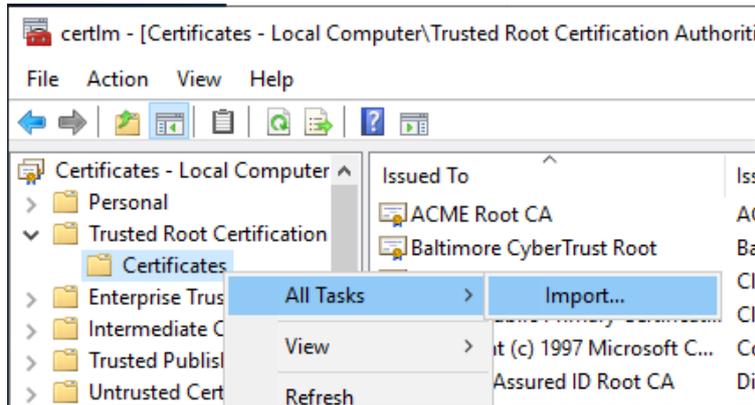
Once the configuration is updated, restart Nginx to load the new certificate. By following these steps, the Web RA module will be properly configured to use the provided SSL certificates.

## 6.6 Importing Root and Intermediate Certificates

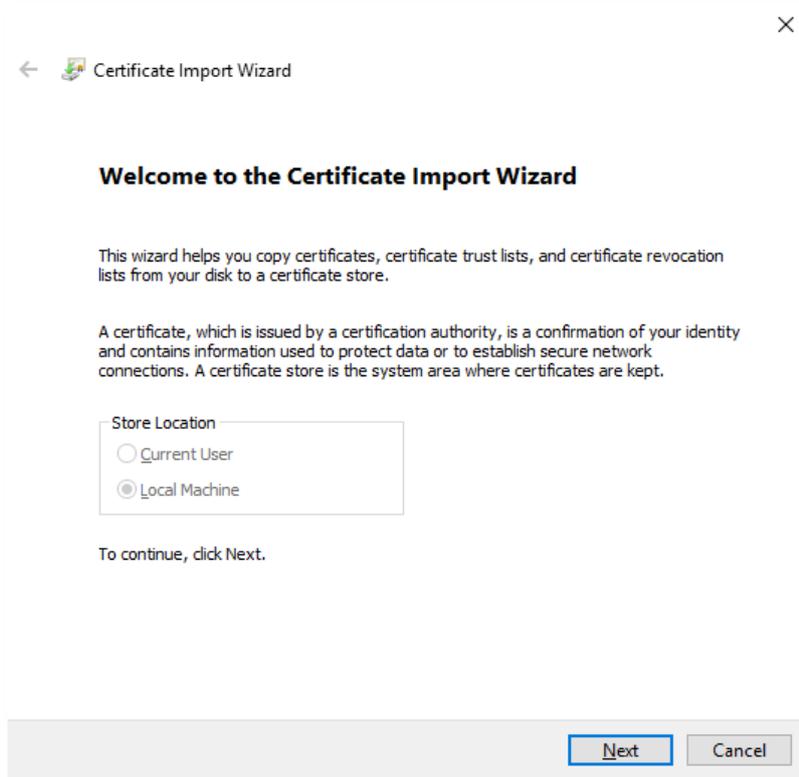
Now that we have the intermediate and root certificates exported and saved in a local file, we can import it to the certificate store.

6.6.1 Launch **certlm.msc** from the command prompt.

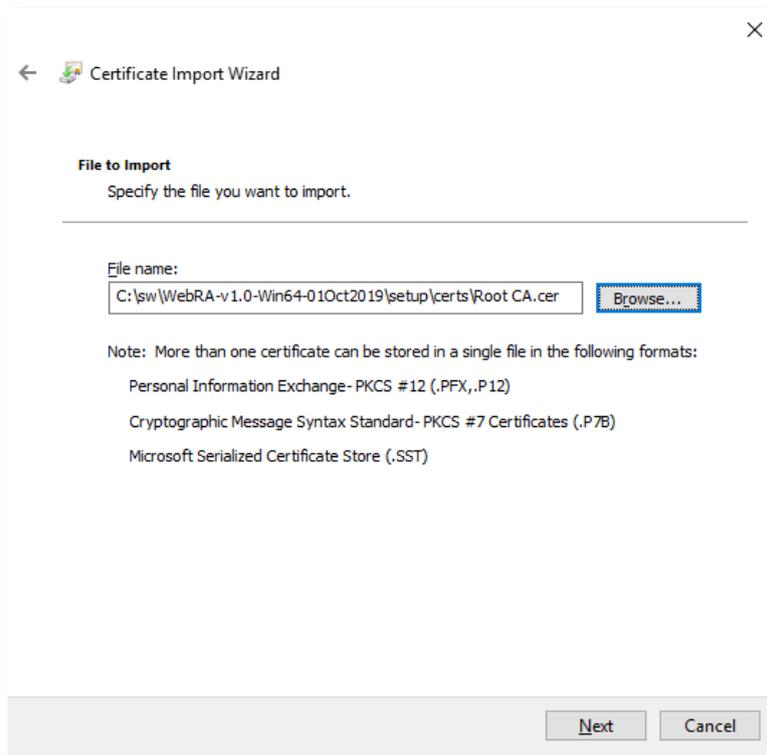
6.6.2 Expand the **Trusted Root Certification Authorities** folder from the left panel and right-click on **Certificates**. Now select **All Tasks** and then **Import...**



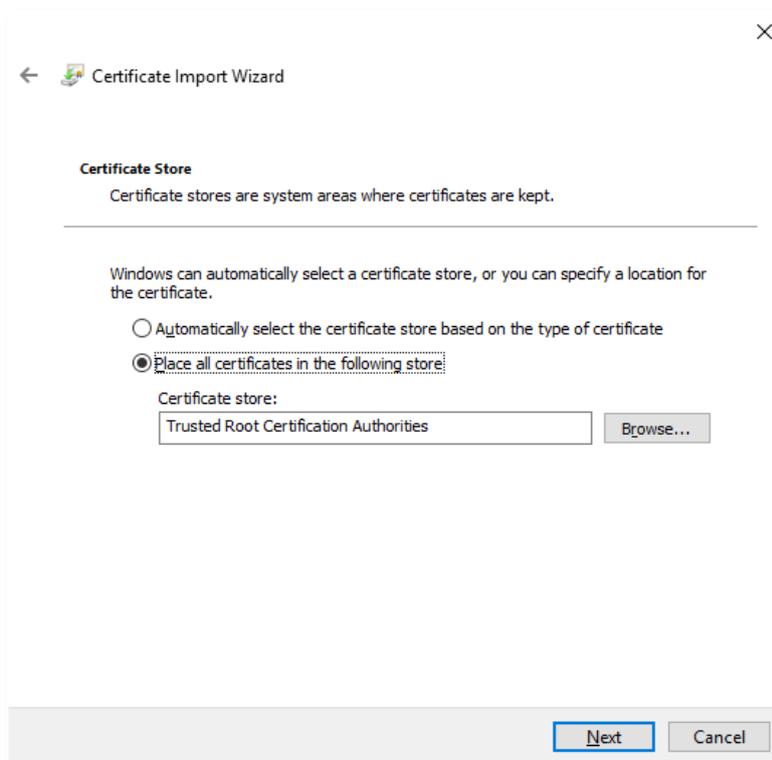
6.6.3 A certificate import wizard appears, Click **Next** to proceed.



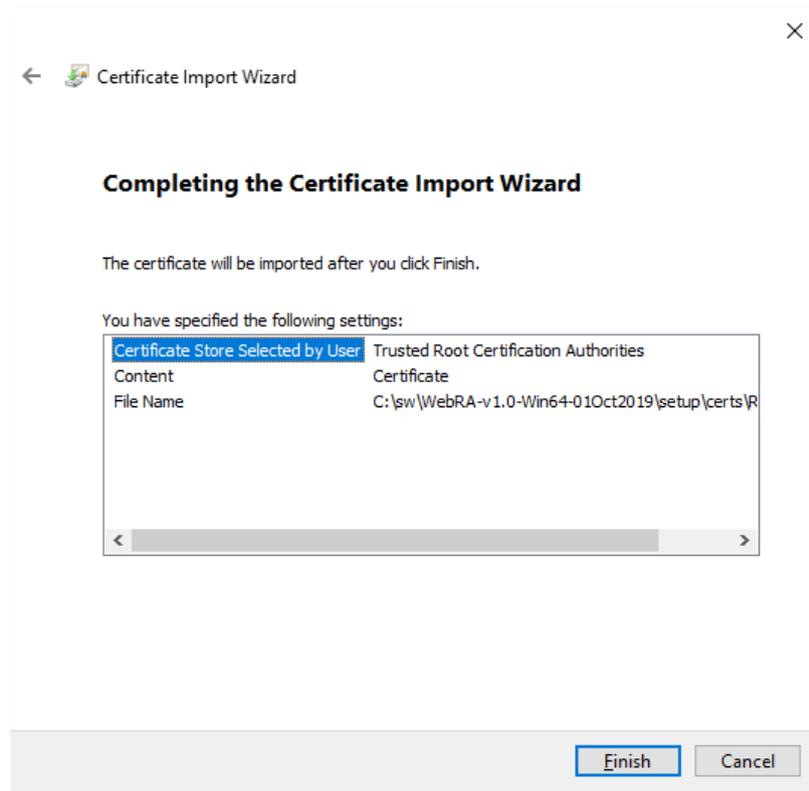
6.6.4 Browse the root certificate that we recently exported and click **Next** to proceed.



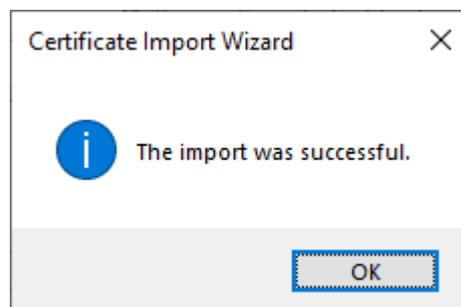
6.6.5 Click **Next** to proceed.



6.6.6 The root certificate is imported to the certificate store, click **Finish**.



6.6.7 A prompt will appear informing about the successful import of the certificate.

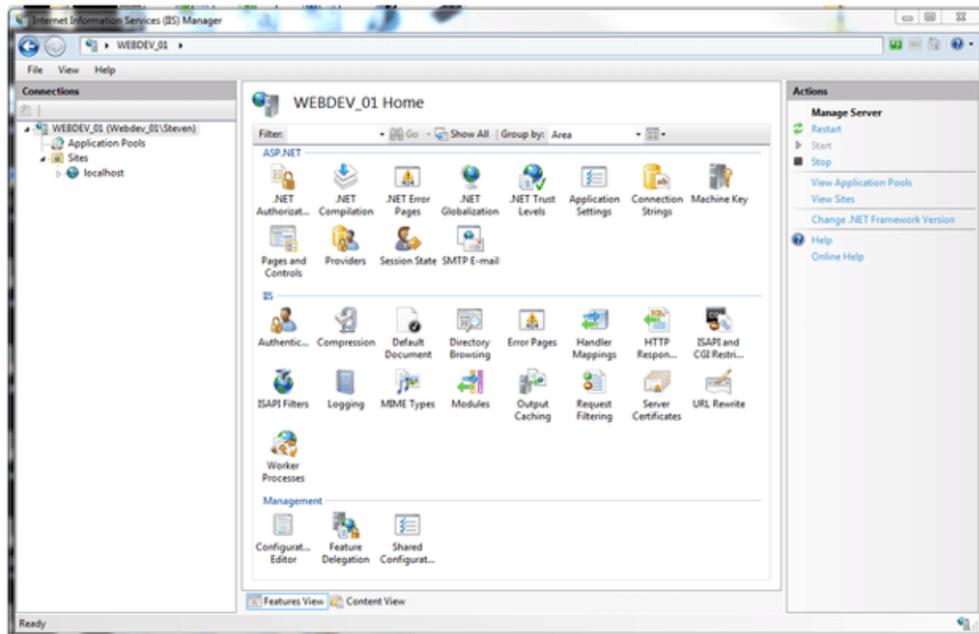


If you want to deploy the application for testing purpose you may want to use a self-signed certificate for proof of concept.

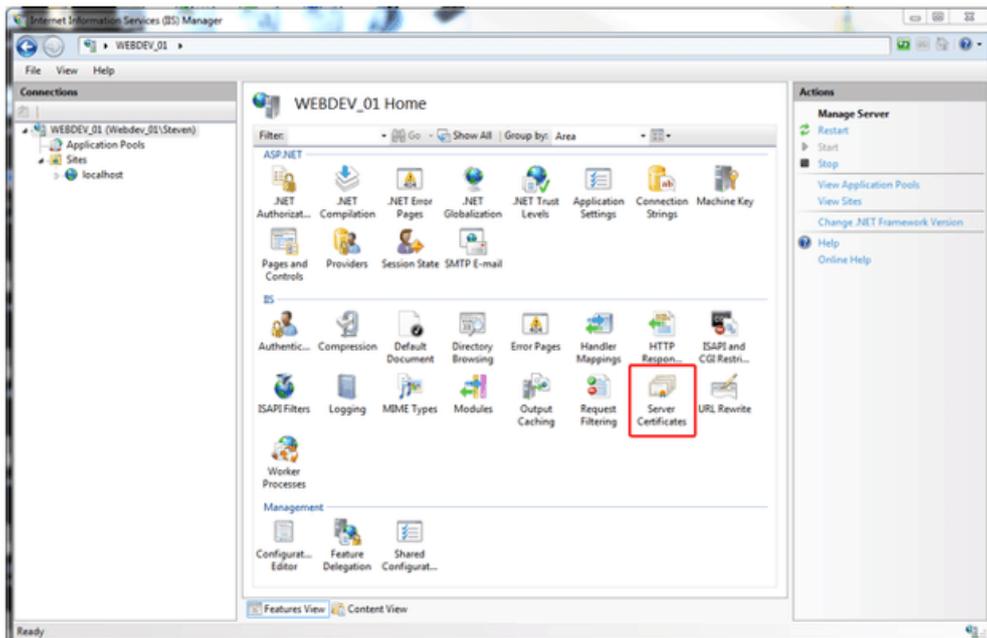
## 6.7 Generate a Self -Signed Certificate

For testing purpose or proof of concept, mostly a self-signed certificate will be required. It is easy to create a self-signed certificate with IIS.

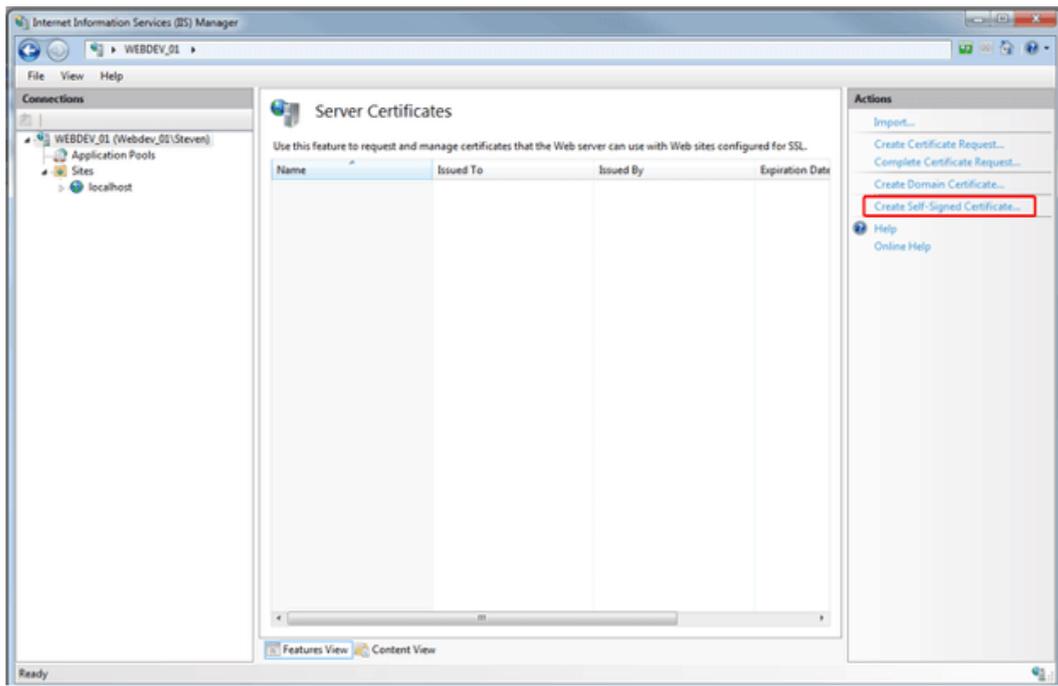
### 6.7.1 Launch the IIS Manager.



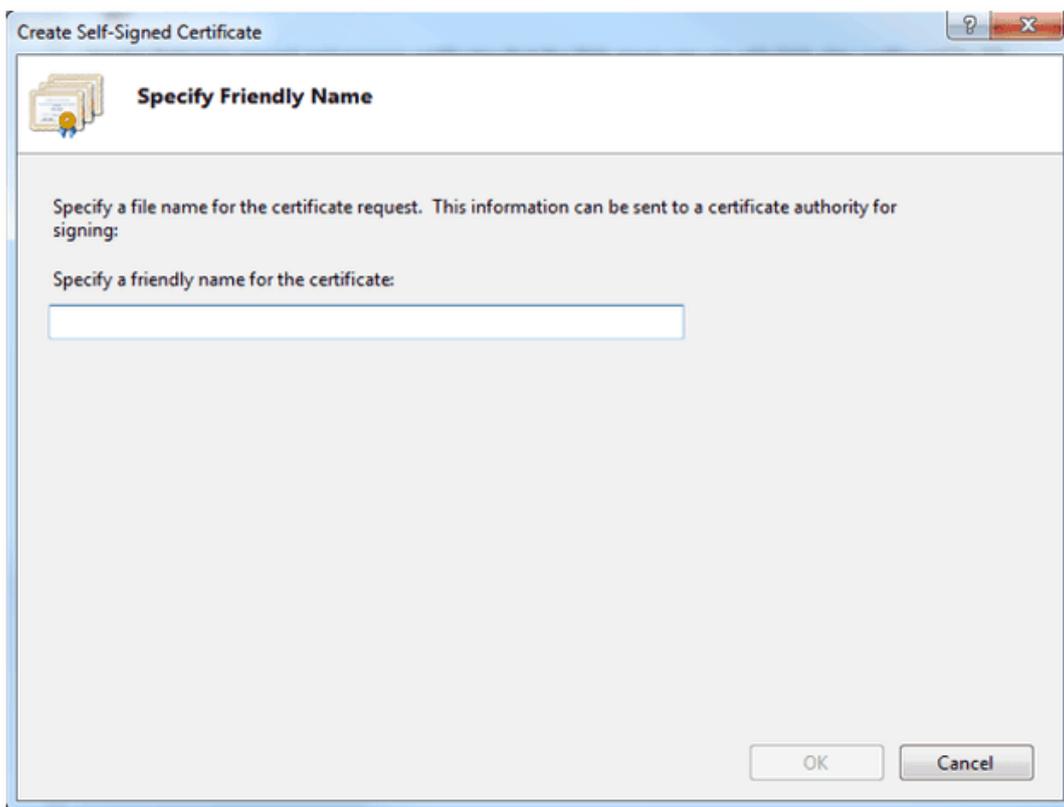
### 6.7.2 Click the **Server Name** from the **Server Connections**.



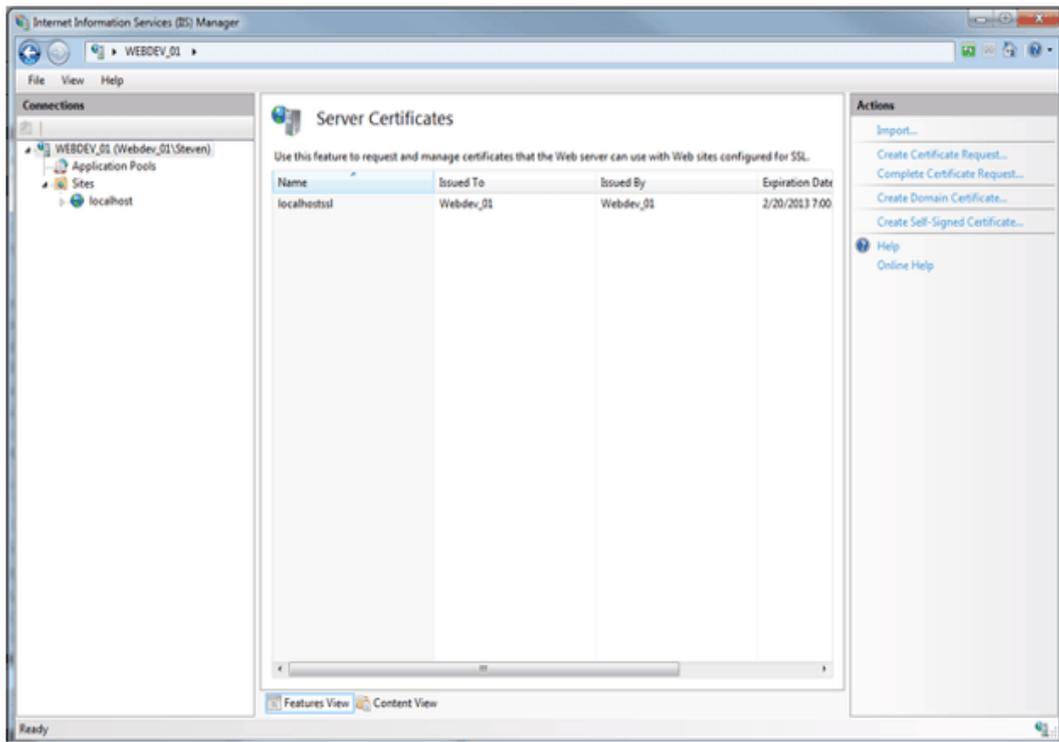
6.7.3 Double-click on **Server Certificates** from the IIS section in the middle panel.



6.7.4 Click **Create Self-Signed Certificate...** under the right Actions column.



6.7.5 Provide a meaningful name and press **OK**.

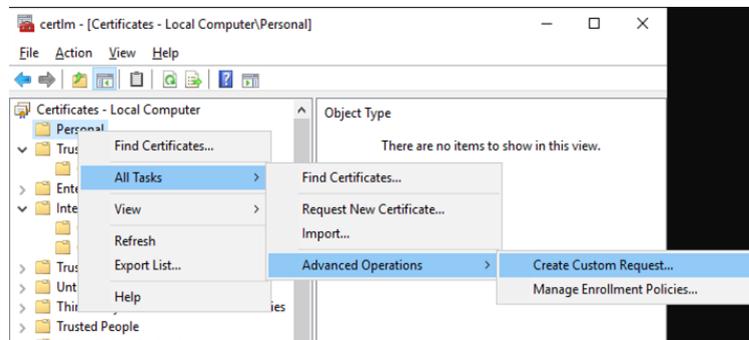


Now you have an SSL certificate that is self-signed and is valid for one year. You can select this certificate for creation of HTTPS binding for testing and proof of concept purposes.

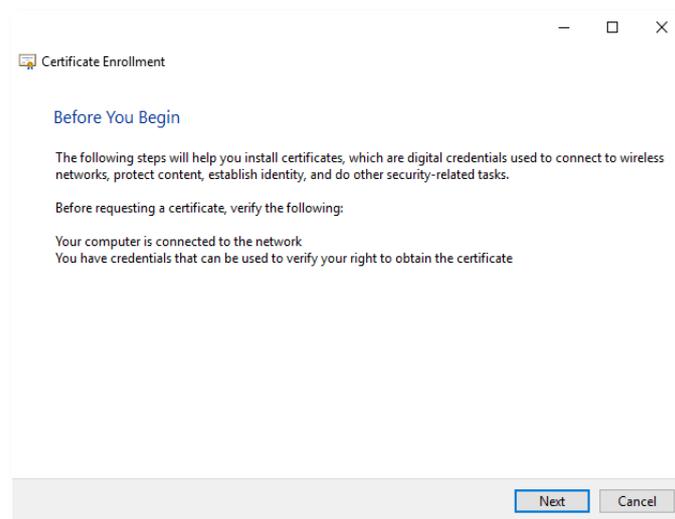
## 6.8 Generate a CSR for an SSL Certificate

To generate a self-signed SSL certificate, follow the steps given below:

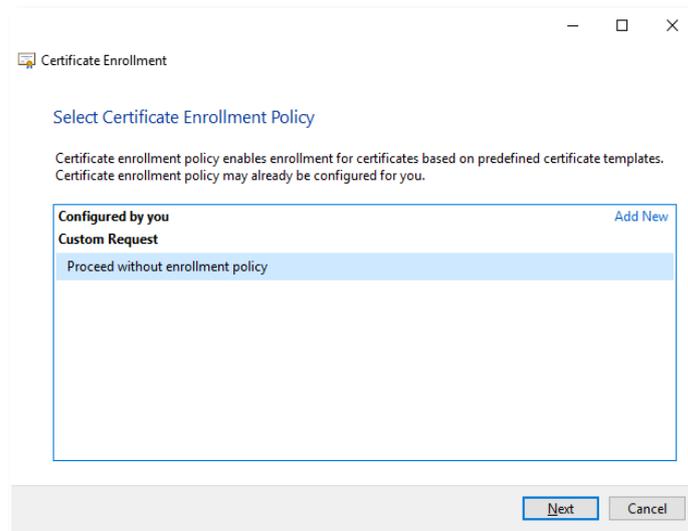
### 6.8.1 Launch **certlm.msc** from the command prompt.



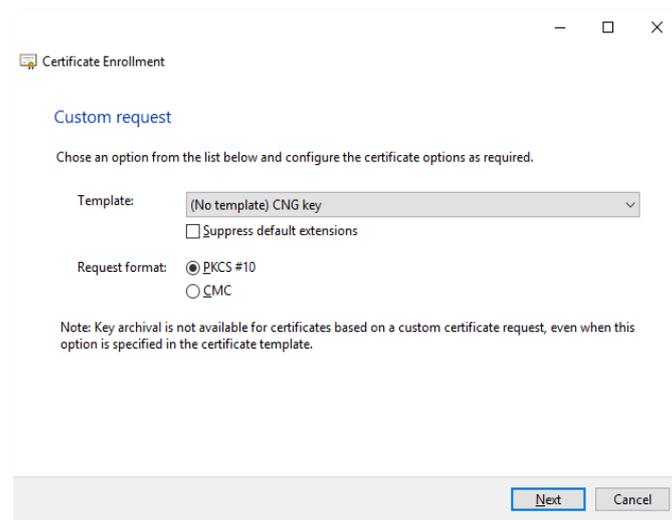
6.8.2 From the left menu, select and right-click the **Personal** folder. From the context menu, select **All Tasks > Advanced Operations > Create Custom request**. A new dialog will appear for certificate enrollment.



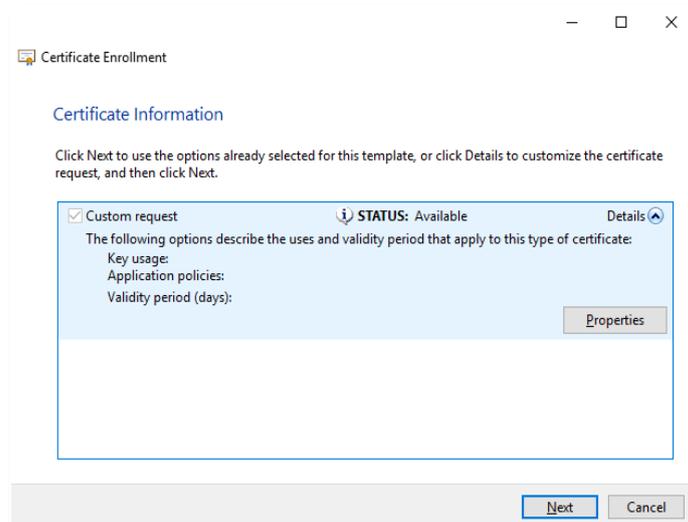
### 6.8.3 Press **Next** to proceed.



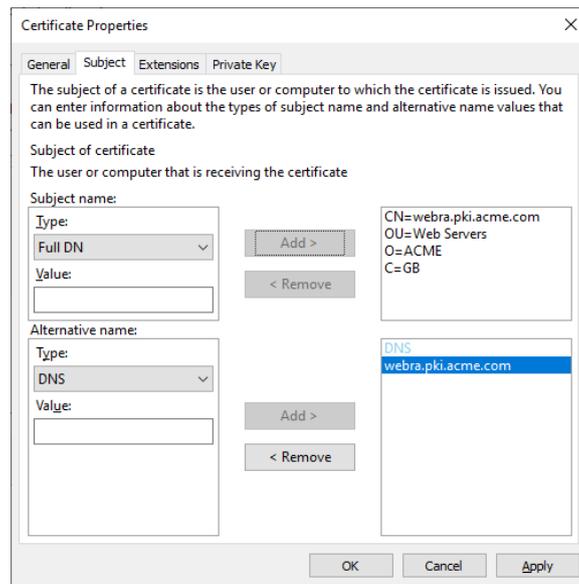
### 6.8.4 Select Proceed without enrollment policy then click **Next**.



### 6.8.5 Accept the default values and press **Next** without changing anything.

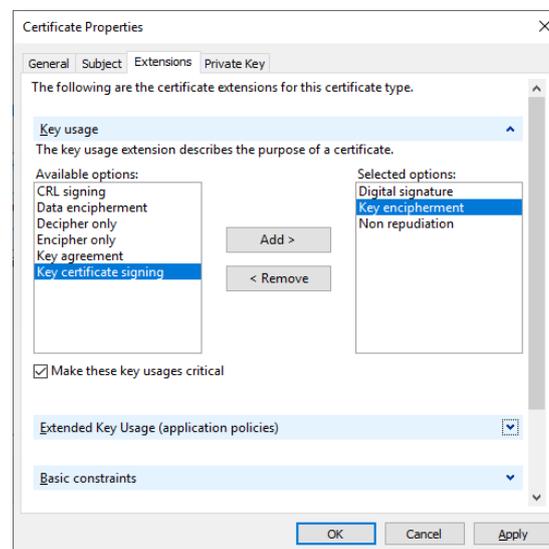


6.8.6 Click **Details** and the Properties button will appear. Click **Properties**.



6.8.7 Select the Subject tab from the top. For subject name enter CN=webra.pki.acme.com, OU=Web Servers, O=ACME, C=GB in the value and press Add >. For Alternate name enter DNS value as webra.pki.acme.com.

These values are the sample values used for certificate creation and can be replaced with the realistic data.

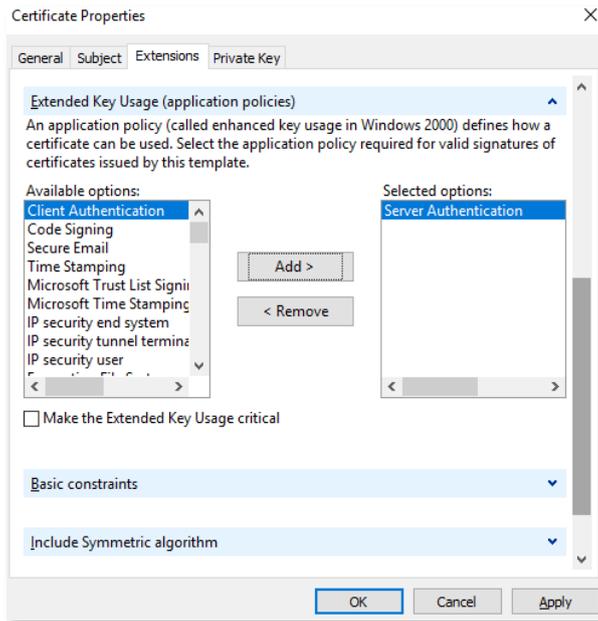


6.8.8 Select the Extensions tab from the top. Select the Key usage option from the dropdown extensions. Now from the Available options, choose the following:

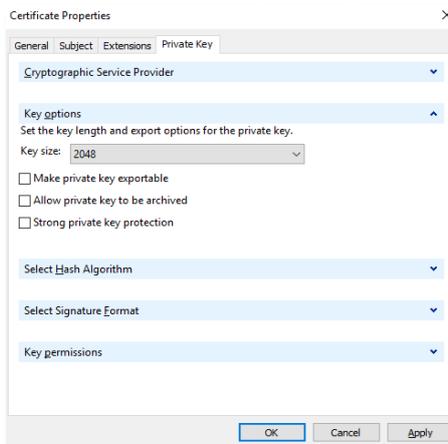
- Digital signature
- Key encipherment
- Non-repudiation

Make sure you tick the **Make these key usages critical** checkbox.

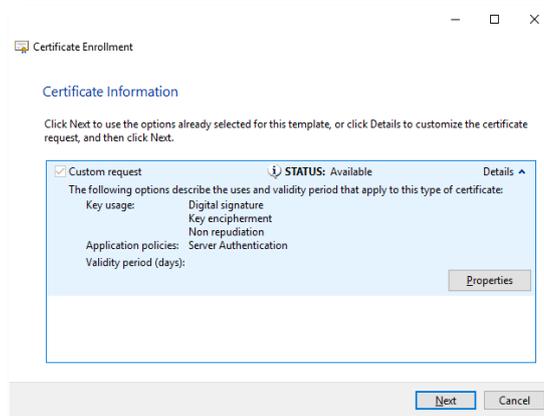
6.8.9 Now select the Extended Key Usage (application policies) from the drop down, and Server Authentication from the list.



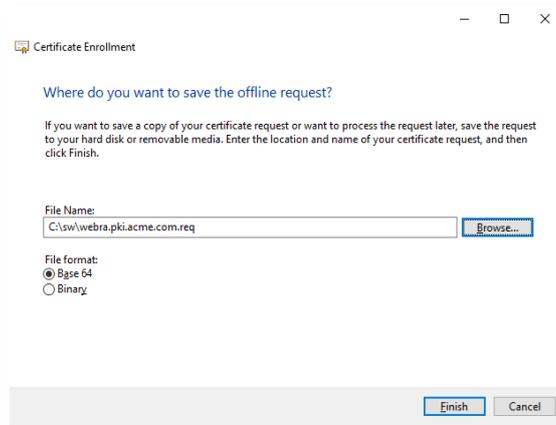
6.8.10 Select the Private Key tab from the top. Select the Cryptographic Service Provider option from the first drop down and Key options from the second drop down. Change the Key size to 2048 and click OK. The Certificate Enrollment screen will appear again.



6.8.11 Press Next to proceed.



6.8.12 Browse the location to save the request file and select the Base 64 file format. Press Finish. This request file can be submitted to any CA to create a certificate against this request. Every CA processes the request and generates a certificate as per their own policy. Once the certificate is received from a CA it can be imported into the certificates.



For further details contact us on [sales@ascertia.com](mailto:sales@ascertia.com) or visit [www.ascertia.com](http://www.ascertia.com)

\*\*\* End of Document \*\*\*