



## Knowledge blog



# Setting up IIS as Reverse Proxy for the .Net Core Kestrel Web Server

## IIS as Reverse Proxy Kestrel behind



We often need to set IIS as Reverse Proxy server especially when you host some self-hosted web application in your server. You may not want them directly appear your web application instead you want to route through your existing IIS web server. And this technique is even useful when you want to setup a Load balancing or failover for your web application. Generally IIS can be setup as a reverse proxy using ARR (Application Request Routing) and URL Rewrite modules.

This is just not about running your .Net Core Kestrel behind the IIS but you can use this technique for all other web applications or services.

### Step 1:

Download ARR (Application Request Routing) from below URL.

<https://www.iis.net/downloads/microsoft/application-request-routing>

## Step 2:

Stop WAS and WMSVC processes by typing. This step is necessary because ARR installs a Hotfix.

Open your elevated command prompt using Run As Administrator and type below commands.

```
net stop was /y
```

Copy

```
net stop wmsvc /y
```

Copy

You may not be having WMSVC (Web Management Service) if just ignore the error saying "Service name invalid"

## Step 3:

Run your ARR installer exe you just downloaded.

Accept the End User License Agreement (EULA) and complete your installation.

## Step 4:

Start back your WAS and WMSVC processes by typing.

```
net start was
```

Copy

```
net start wmsvc
```

Copy

## Step 5:

Download URL Rewrite module from below link and run the installer.

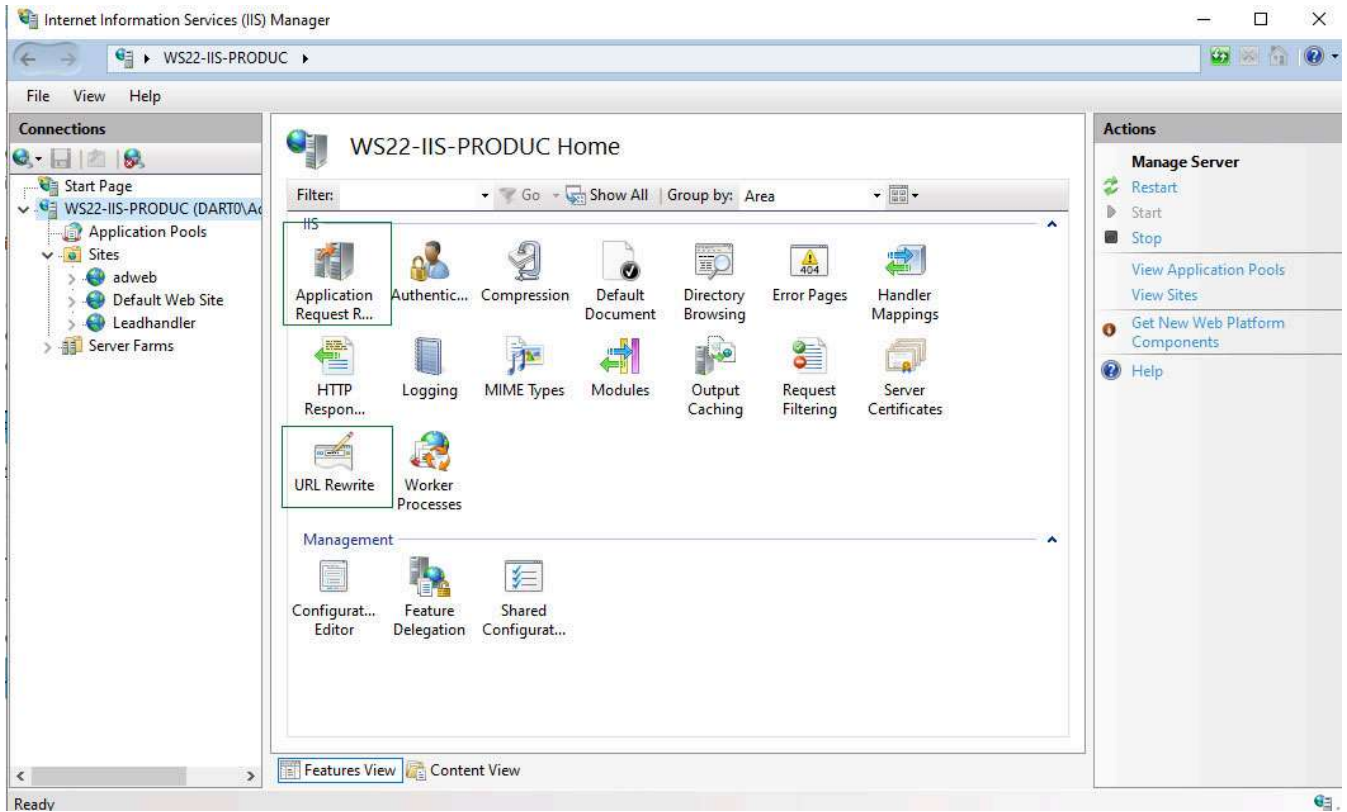
<https://www.iis.net/downloads/microsoft/url-rewrite>

Accept the End User License Agreement (EULA) and complete your installation.

Now restart IIS (Type **IISReset** in command prompt).

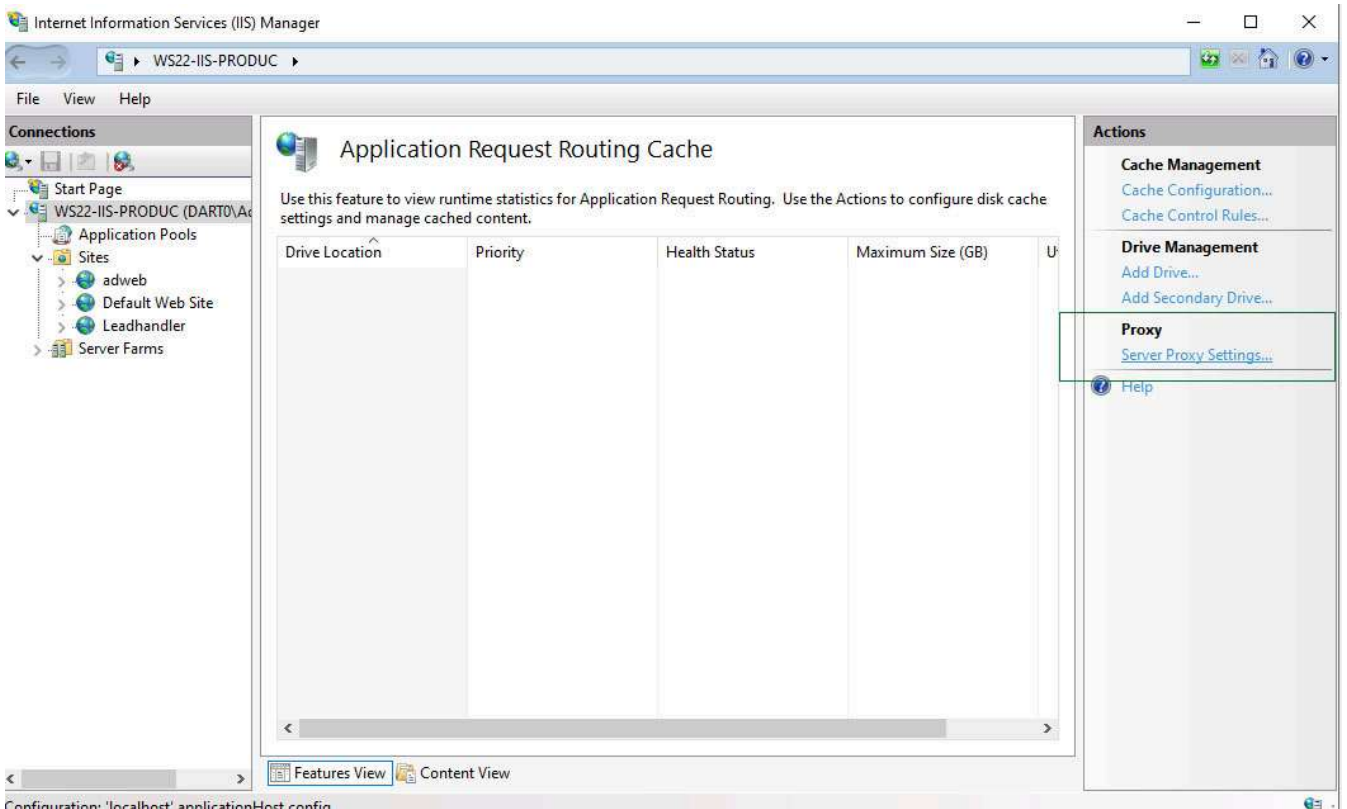
### Step 6:

Go to IIS and verify you have both ARR and URL Rewrite modules installed. Refer the screen below.

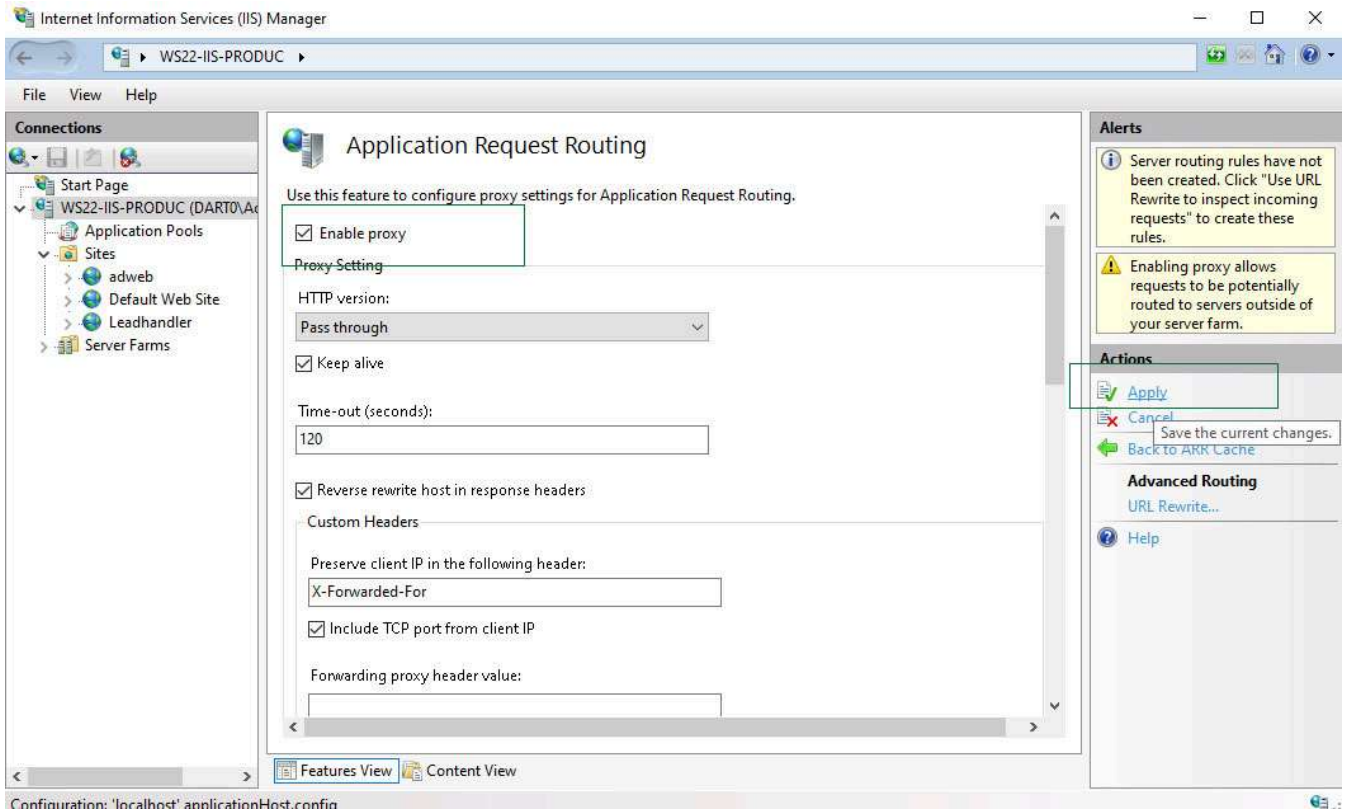


### Step 7:

Click on ARR and then click on Server Proxy Settings. Refer below image.



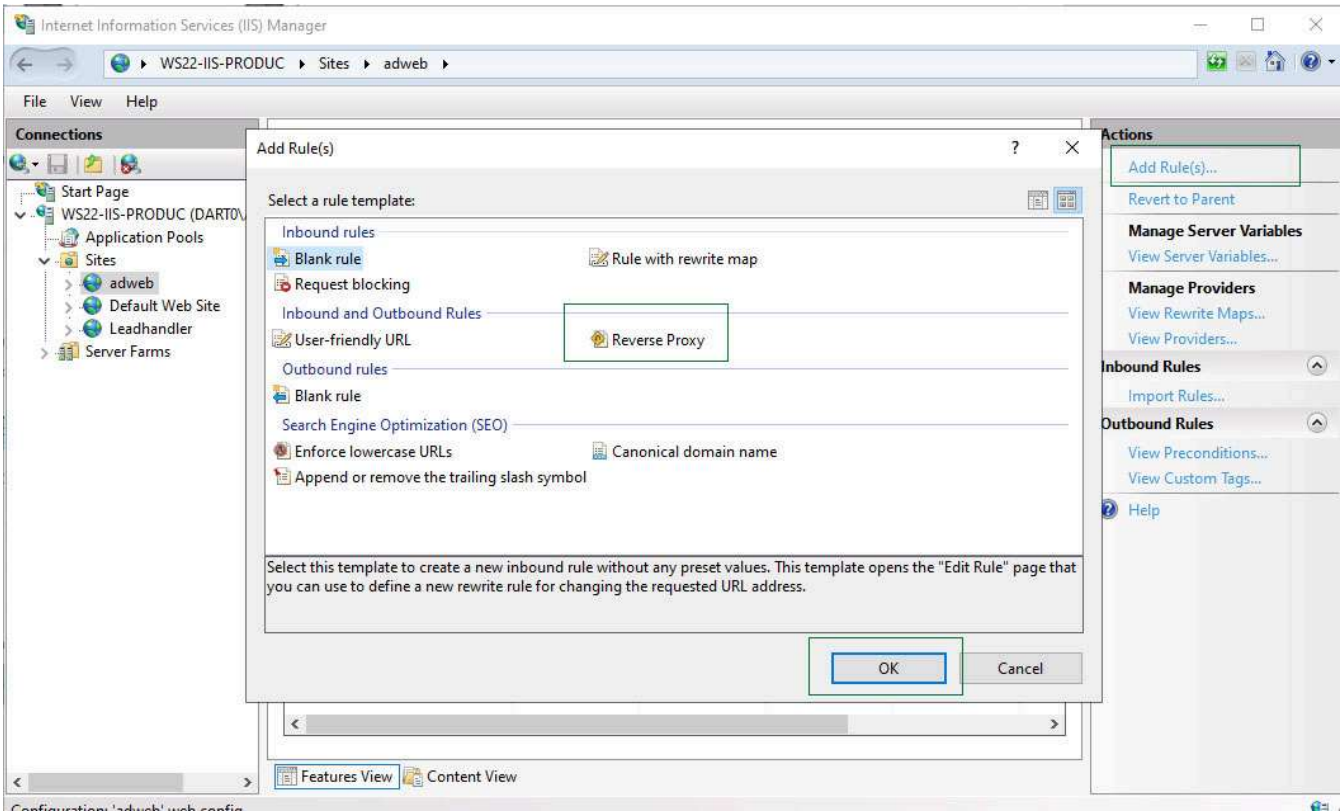
## Check on Enable Proxy and then Click Apply.



That's all with ARR. We will need to work with URL Rewriting from now.

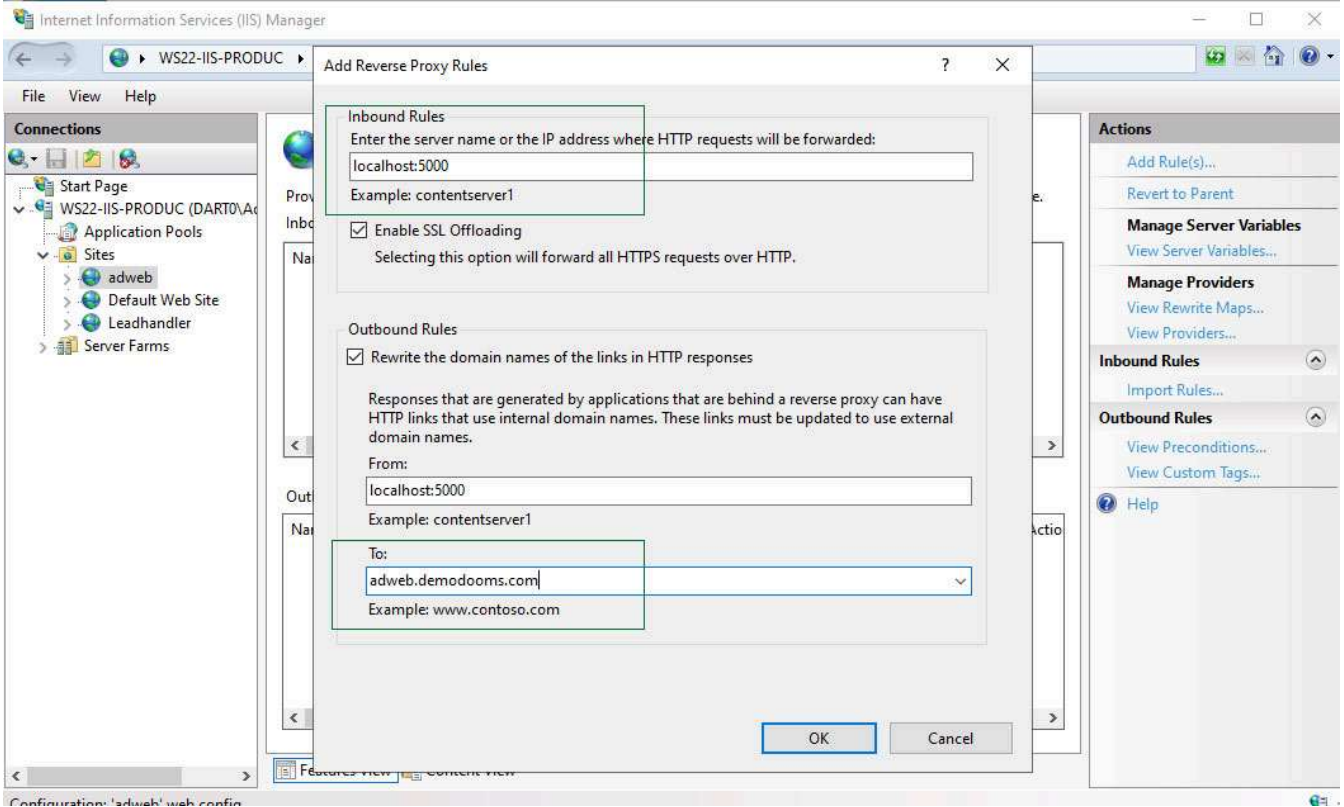
### Step 8:

Select the web site you want to work as Reverse Proxy. Then double click the module URL Rewrite. Now click on **Add Rule** from the right side menu and then click on the **Reverse Proxy** and Click **OK**. Refer the image below.



In the **Inbound Rules** add the URL for which you actually want to redirect the proxy. In my case I am running the AD Web self-hosted application behind this proxy.

**Outbound Rule** is optional but it is useful to add your public URL so the other web services you connect will know where the request is coming from.



As it, go ahead and test your web URL in browser.

Looking for a Web or Mobile development or consultation? Our experts can help you >>