Configure load balancer or reverse proxy

URL: https://docs.grassfish.com/docs/configure-load-balancer-or-reverse-proxy

Archiviert am: 2025-07-17 18:30:56

IXM One supports API access through load balancer and/or proxy setups that help manage server systems.

Load balancers distribute large amounts of traffic across a pool of available servers. A load balancer allows an application to scale beyond a single server, making processing more efficient. By configuration, a load balancer can also serve as a reverse proxy. A reverse proxy takes a request from a client, forwards it to an available server, and returns the server response to the client.

Seamlessly using a load balancer and/or reverse proxy with IXM One requires additional configuration of Internet Information Services (IIS) for Windows Server. Follow the instructions in this article to configure your system.

Prerequisites

Before you begin configuration, make sure you meet the following prerequisites:

- You are using IXM One version 11.16.0 or later.

- The URL Rewrite module is installed on each web server in your system. For more information, see the Microsoft documentation: https://www.iis.net/downloads/microsoft/url-rewrite

> ### Note
>
> *Apply the following settings only if IIS is behind a load balancer.*

Add allowed server variables

You must add allowed server variables to the URL Rewrite module. To do so, perform the following steps:

1. Open IIS Manager.

2. Under **Connections**, select your server.

3. Open **URL Rewrite**.

4. Under **Actions**, select **View Server Variables**.

5. Click **Add** and add the following variables:

   ◦ HTTP_HOST

   ◦ HTTPS

   ◦ REMOTE_ADDR

- REMOTE_HOST

- SERVER_NAME

- SERVER_PORT
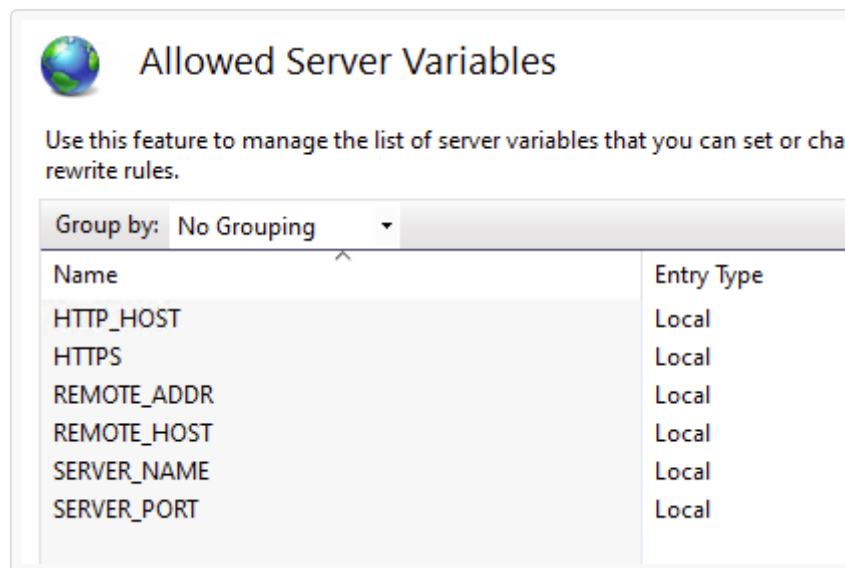
Set environment variable

To set the environment variable, perform the following steps:

1. Go to IXM_BASE_DIR\www.

2. Open the web.config file.

3. Add the environment variable ASPNETCORE_FORWARDEDHEADERS_ENABLED and set it to **true**.

4. Save your changes and close the file.

```
<configuration>
    <system.webServer>
        <aspNetCore>
            <environmentVariables>
                <environmentVariable name="ASPNETCORE_FORWARDEDHEADERS_ENABLED" value="true" />
            </environmentVariables>
        </aspNetCore>
    </system.webServer>
</configuration>
```

Add rewrite rules

There are two scenarios for the IIS binding:

- **The binding is different from the MainServerURL**

We recommend using this solution because performance is better when SSL is managed solely by the reverse proxy.

• **The IIS binding matches the MainServerURL**

This solution is simpler because you don't have to change the binding when adding a reverse proxy to an existing setup. However, it's slower due to the SSL overhead on the web server.

Select your scenario and follow the matching instructions below.

IIS binding is different from MainServerURL

In this scenario, your external URL and internal IIS binding are different. For example: https://external.domain and http://localhost.
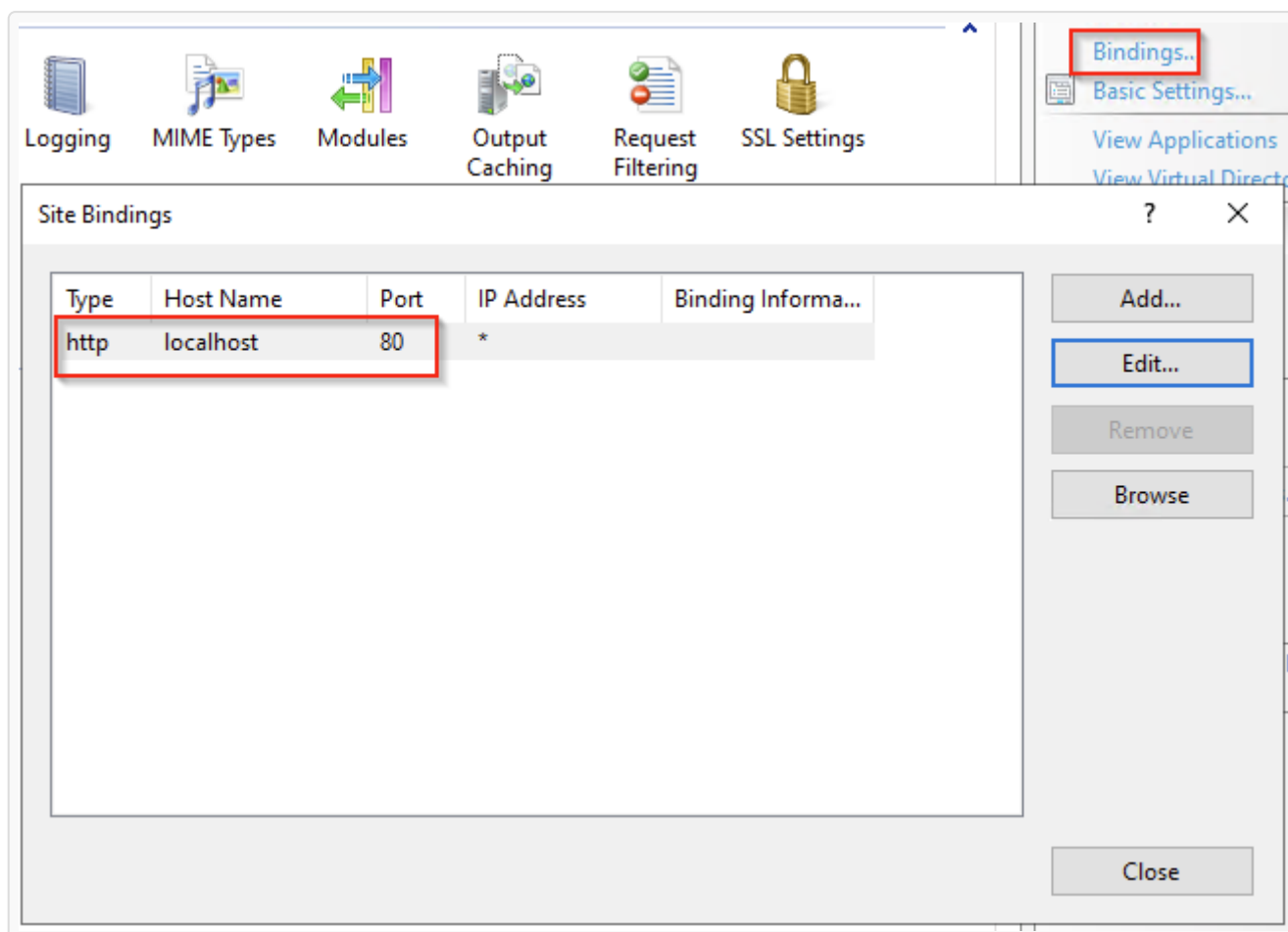
To add the required rewrite rules, perform the following steps:

1. Go to IXM_BASE_DIR\www.

2. Open the web.config file.

3. Add the following code for your rewrite rules:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
```

```
        <system.webServer>
            <rewrite>
                <rules>
                    <!-- add rules here -->
                </rules>
            </rewrite>
        </system.webServer>
    </configuration>
```

4. Add the following rewrite rules:

```
 <rule name="X-Forwarded-Host">
     <match url=".*" />
     <serverVariables>
         <set name="HTTP_HOST" value="{C:0}" />
         <set name="SERVER_NAME" value="{C:1}" />
         <set name="SERVER_PORT" value="{C:3}" />
     </serverVariables>
     <action type="None" />
     <conditions>
         <add input="{REMOTE_ADDR}" pattern="^192\.168\.175\.1$" />
         <add input="{HTTP_X_FORWARDED_HOST}" pattern="^([\w\-\.]+)(\:(\d+))?$" />
     </conditions>
 </rule>
 <rule name="X-Forwarded-Proto-Https">
     <match url=".*" />
     <serverVariables>
         <set name="HTTPS" value="on" />
     </serverVariables>
     <action type="None" />
     <conditions>
         <add input="{REMOTE_ADDR}" pattern="^192\.168\.175\.1$" />
         <add input="{HTTP_X_FORWARDED_Proto}" pattern="^https$" />
     </conditions>
 </rule>
 <rule name="X-Forwarded-Proto-Http">
     <match url=".*" />
     <serverVariables>
         <set name="HTTPS" value="off" />
     </serverVariables>
     <action type="None" />
     <conditions>
         <add input="{REMOTE_ADDR}" pattern="^192\.168\.175\.1$" />
         <add input="{HTTP_X_FORWARDED_Proto}" pattern="^http$" />
     </conditions>
 </rule>
 <rule name="X-Forwarded-For">
     <match url=".*" />
     <serverVariables>
         <set name="REMOTE_ADDR" value="{C:0}" />
         <set name="REMOTE_HOST" value="{C:0}" />
     </serverVariables>
     <action type="None" />
     <conditions>
         <add input="{REMOTE_ADDR}" pattern="^192\.168\.175\.1$" />
         <add input="{HTTP_X_FORWARDED_FOR}" pattern=".+" />
     </conditions>
 </rule>
```

5. Replace each occurrence of **^192\.168\.175\.1$** with the load balancer's IP address. You must precede special characters such as a period (**.**) with a backslash (****) because this pattern is a regular expression.

6. Save your changes and close the file.

IIS binding matches MainServerURL

In this scenario, your external URL and the internal IIS binding match. For example: https://external.domain and https://external.domain.
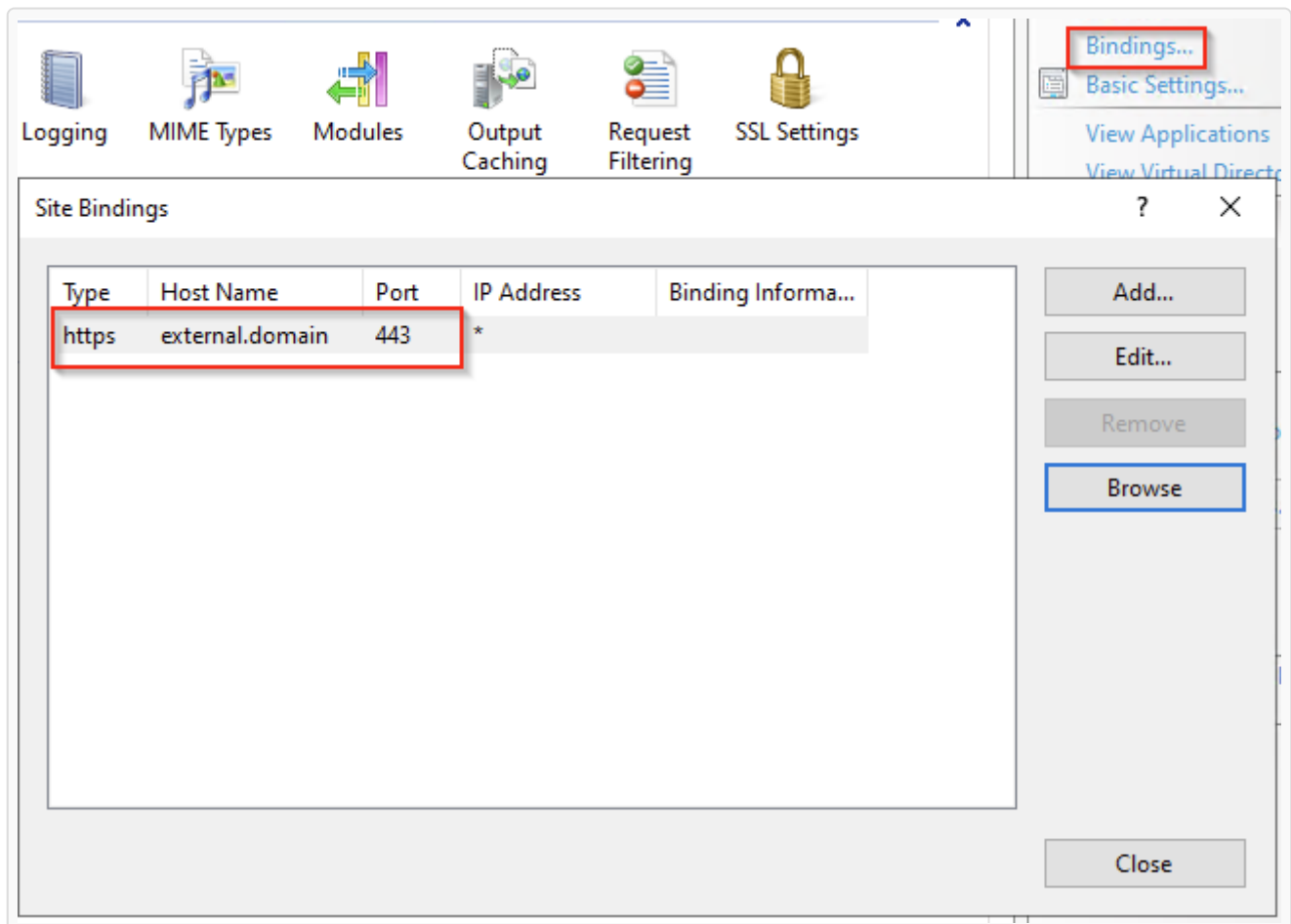
> ### *Note*
>
> *Because of the HTTPS binding, you must install the SSL certificate on the web server. HTTPS encryption creates additional CPU load on the web server.*

To add the required rewrite rule, perform the following steps:

1. Go to IXM_BASE_DIR\www.

2. Open the web.config file.

3. Add the following code for your rewrite rule:

```xml
 <?xml version="1.0" encoding="utf-8"?>
<configuration>
    <system.webServer>
        <rewrite>
            <rules>
                <!-- add rules here -->
            </rules>
        </rewrite>
    </system.webServer>
</configuration>
```

4. Add the following rewrite rule:

```xml
 <rule name="X-Forwarded-For">
    <match url=".*" />
    <serverVariables>
        <set name="REMOTE_ADDR" value="{C:0}" />
        <set name="REMOTE_HOST" value="{C:0}" />
    </serverVariables>
    <action type="None" />
    <conditions>
        <add input="{REMOTE_ADDR}" pattern="^192\.168\.175\.1$" />
        <add input="{HTTP_X_FORWARDED_FOR}" pattern=".+" />
    </conditions>
</rule>
```

5. Replace each occurrence of **^192\.168\.175\.1$** with the load balancer's IP address. You must precede special characters such as a period (**.**) with a backslash (****) because this pattern is a regular expression.

6. Save your changes and close the file.