

Esta breve guía muestra el paso a paso de cómo preparar la entrega y cómo deployar el sistema el día de la defensa en los laboratorios de la universidad. **Es fundamental ir con tiempo antes de la hora marcada de la defensa para desplegar a tiempo, sin generar demoras.**

Primera parte - Previo a la entrega del obligatorio

Al entregar el obligatorio deberán entregar una carpeta con la aplicación lista para ser desplegada en el servidor de aplicaciones IIS. **Es de suma importancia que estos pasos los hagan previos a entregar.**

1 - Compilar la aplicación en Release

Pararse sobre la carpeta de la solución o el proyecto de WebApi (dependerá de cómo tengan creado el proyecto de inicio) y ejecutar el comando:

- `dotnet build -c Release`

2 - Crear una publicación de nuestro sistema

Esta carpeta se generará con todo lo necesario para desplegar nuestra aplicación en el servidor de aplicaciones IIS.

Nuevamente sobre la carpeta de la solución o el proyecto de WebApi y ejecutar el comando:

- `dotnet publish -c Release`

Si vamos a la ruta “[WebApiProject]\bin\Release\netcoreapp2.1\” encontraremos una carpeta llamada “**publish**” y dentro estarán los archivos de nuestra aplicación así como también las bibliotecas de clases de otras dependencias como EFCore.

3 - Quitando WebDAV

Por defecto en los laboratorios no podremos ejecutar peticiones del tipo **PUT ni DELETE**. Para ello debemos quitar el módulo de WebDAV de nuestra aplicación.

Dentro de la carpeta “**publish**” generada, se encuentra un archivo llamado “**web.config**”, el contenido es similar a este:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <configuration>
3   <system.webServer>
4     <handlers>
5       <add name="aspNetCore" path="*" verb="*" modules="AspNetCoreModule" resourceType="Unspecified" />
6     </handlers>
7     <aspNetCore processPath="dotnet" arguments=".\AGT.WebApi.dll" stdoutLogEnabled="false" stdoutLogFile=".\logs\stdout" />
8   </system.webServer>
9 </configuration>
10 <!--ProjectGuid: E3772596-65C4-4544-8C10-66158F624620-->
```

Debemos agregar la siguiente configuración:

```
<modules runAllManagedModulesForAllRequests="false">
  <remove name="WebDAVModule" />
</modules>
```

Resultando en un archivo similar al siguiente:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <system.webServer>
    <modules runAllManagedModulesForAllRequests="false">
      <remove name="WebDAVModule" />
    </modules>
    <handlers>
      <add name="aspNetCore" path="*" verb="*" modules="AspNetCoreModule" resourceType="Unspecified" />
    </handlers>
    <aspNetCore processPath="dotnet" arguments=".\\AGT.WebApi.dll" stdoutLogEnabled="false" stdoutLogFile=".\logs\stdout" />
  </system.webServer>
</configuration>
<!--ProjectGuid: E3772596-65C4-4544-8C10-66158F624620-->
```

4 - Cambiando el connection string para la base de datos

Si lo desean, pueden entregar la aplicación ya configurada con el motor de base de datos del laboratorio. Igualmente deben validar el día de la defensa que el nombre del motor sea el correcto al de la máquina del laboratorio, si no es correcto deberán cambiar la configuración el día de la defensa siguiendo estos mismos pasos.

Para cambiar el connection string, deben ir nuevamente a la carpeta “**publish**” y buscar el archivo “**appsettings.json**”, nótese que cómo creamos la publicación en modo release, no usamos el archivo de desarrollo. El archivo debería quedar similar al siguiente:

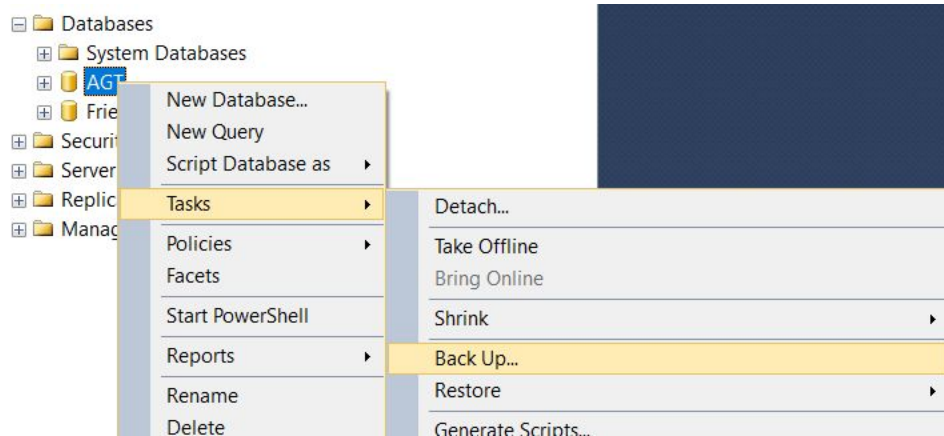
```
{
  "Logging": {
    "LogLevel": {
      "Default": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "AGTDatabase": "Server=\\.\\SQLSERVER_R14;Database=AGT;Trusted_Connection=True;Integrated Security=True;"
  }
}
```

5 - Exportando la base de datos

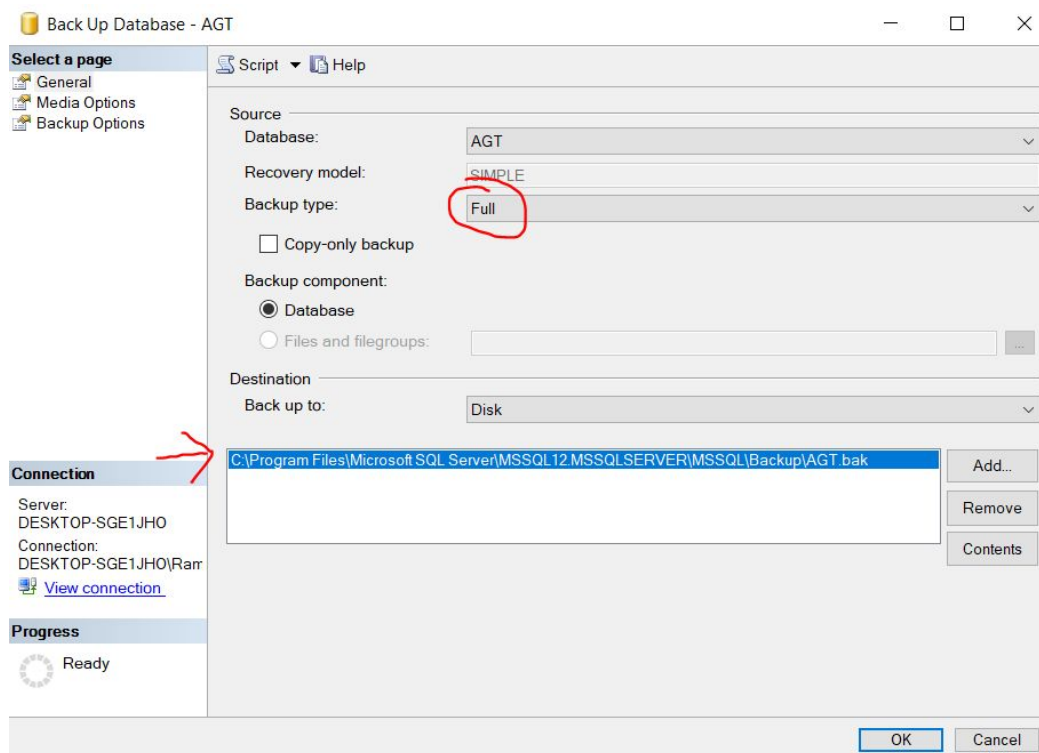
Deberán entregar los backups de la base de datos para luego desplegarla el día de la defensa. Esta guía no cubre cómo generar una migración con EFCore ni cómo impactarla en la base. Se asume que ya tienen la base creada en una instancia de **SQL Server 2014**. **Es de suma importancia que la versión de SQL usada sea 2014 de lo contrario no podrán levantar el backup en el laboratorio.**

Pasos:

- A. Abrir SQL Server Management Studio
- B. Conectarse a la instancia donde se encuentre la base
- C. Generar un Back up



- D. Prestar atención al directorio donde se genera ya que SQL Server sólo tiene permisos de lectura sobre ciertos directorios



6 - Carpeta de entrega

Una vez realizados los pasos anteriores, en la raíz de su repositorio crear una carpeta llamada Entrega o de forma similar. Dentro colocar el **.bak** de la base, la carpeta **publish** con su aplicación y la **colección de test de postman exportada**.

Segunda parte - El día de la defensa

El día de la defensa deberán desplegar la aplicación en una máquina del laboratorio (máquina 1) y deberán abrir un cliente de postman-rest en otra máquina (máquina 2).

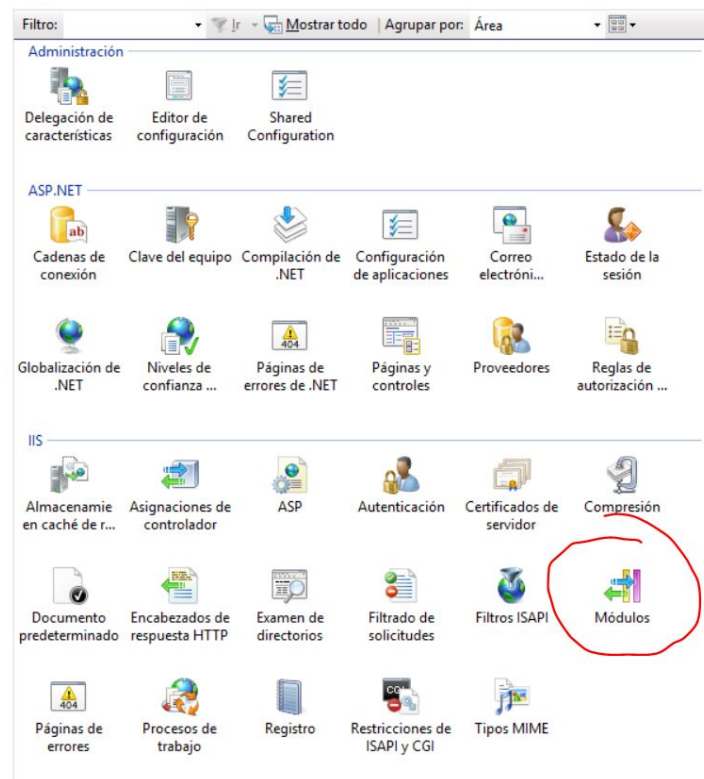
1 - Verificando que la máquina 1 tenga lo necesario:

La aplicación que vamos a desplegar en IIS es una aplicación de .NET Core. La aplicación no utiliza el CLR que por defecto utilizan las demás aplicaciones de IIS, por lo cual debemos validar que esté efectivamente instalado el runtime de .NET Core en la máquina.


A - Para ello abrimos IIS, una forma rápida de hacerlo:

- Tecla inicio + R
- Escribir "inetmgr" y darle enter

B - Una vez abierto IIS, en la página principal, ir a la sección de módulos o modules:



C - Si tenemos instalado el Runtime de .Net Core debería figurar el siguiente módulo en la lista:

 **Módulos**

Utilice esta característica para configurar los módulos de código nativo y administrado que procesan solicitudes realizadas al servidor web.

Agrupar por: Sin agrupar			
Nombre	Código	Tipo de módulo	Tipo de entrada
AnonymousAuthenticationModule	%windir%\System32\inetsrv\authanon.dll	Nativo	Local
AnonymousIdentification	System.Web.Security.AnonymousIdentificationModule	Administrado	Local
AspNetCoreModule	%SystemRoot%\system32\inetsrv\aspnetcore.dll	Nativo	Local

D - Si no está instalado, debemos bajarlo e instalarlo desde el siguiente link:

(BAJAR RUNTIME Y NO EL SDK)

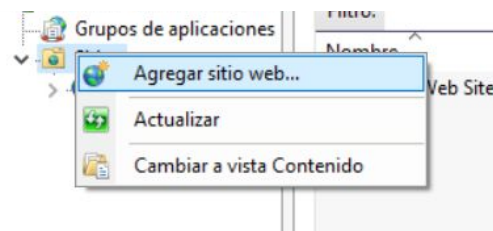
<https://www.microsoft.com/net/download>

2 - Desplegando la aplicación en la máquina 1:

A - Copiar los archivos

Tomar la carpeta **“publish”** ubicada en la carpeta de entregables generada en la Parte 1 y pegarla en **“C:\inetpub\wwwroot”**. Esta carpeta es la carpeta por defecto del IIS, los usuarios de IIS tiene permisos necesarios sobre dicho directorio. Si ya existe una carpeta publish en dicho directorio, pueden renombrar su carpeta antes de pegarla.

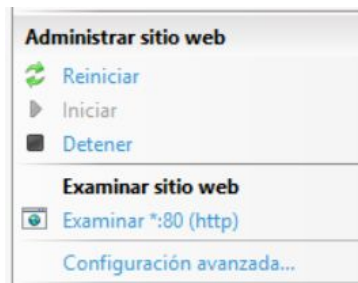
B - Volver al IIS y crear un sitio nuevo:

A screenshot of the 'Agregar sitio web' (Add new site) dialog box in IIS Manager. The dialog has several sections: 'Nombre del sitio:' (Site name) with the value 'AGT' circled in red; 'Grupo de aplicaciones:' (Application pool) with the value 'AGT' circled in red; 'Directorio de contenido' (Content directory) with 'Ruta de acceso física:' (Physical path) set to 'C:\inetpub\wwwroot\publish', where an arrow points to the text; 'Enlace' (Binding) section with 'Tipo:' (Type) set to 'http', 'Dirección IP:' (IP address) set to 'Todas las no asignadas', and 'Puerto:' (Port) set to '80', with the port field circled in red. At the bottom, there is a checkbox 'Iniciar sitio web inmediatamente' (Start site immediately) which is checked, and 'Aceptar' (OK) and 'Cancelar' (Cancel) buttons.

Prestar atención a lo marcado en rojo.

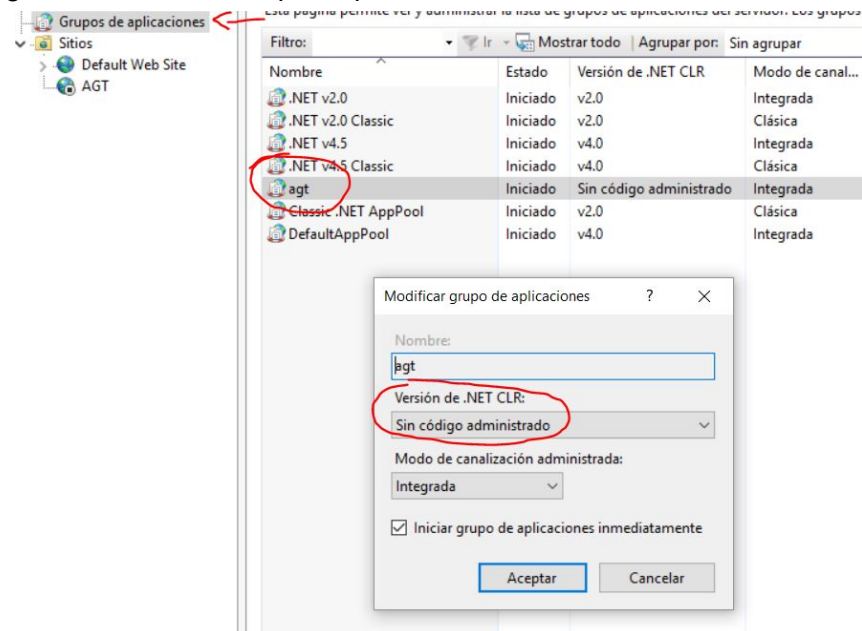
- El nombre del sitio es a elección
- Cuando creamos un nuevo sitio, IIS nos creará un nuevo POOL o grupo de aplicaciones para ese nuevo sitio, además creará un usuario de windows en este caso llamado: **"IIS APPPOOL\AGT"**
- La ruta de acceso física es la ruta a nuestra carpeta "publish" que contiene la aplicación
- El puerto es también a elección. Por defecto IIS cuenta con un sitio por defecto "Default Website" que ya corre en el puerto 80. Sólo podemos tener corriendo un único sitio por puerto. Pueden poner otro puerto o dejar el 80 y luego detener el sitio por Default.

Una vez creado nuestro sitio, pararse en el Default, y en el panel derecho detenerlo. Luego pararse en nuestro sitio recién creado e iniciarlo.



C - Quitar CLR por defecto de IIS

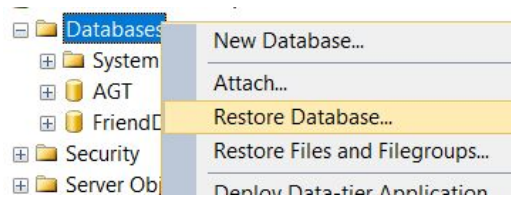
Cómo ya mencionamos, .NET Core no usa el CLR por defecto configurado en IIS. Para ello debemos configurar nuestro POOL para que no lo utilice.



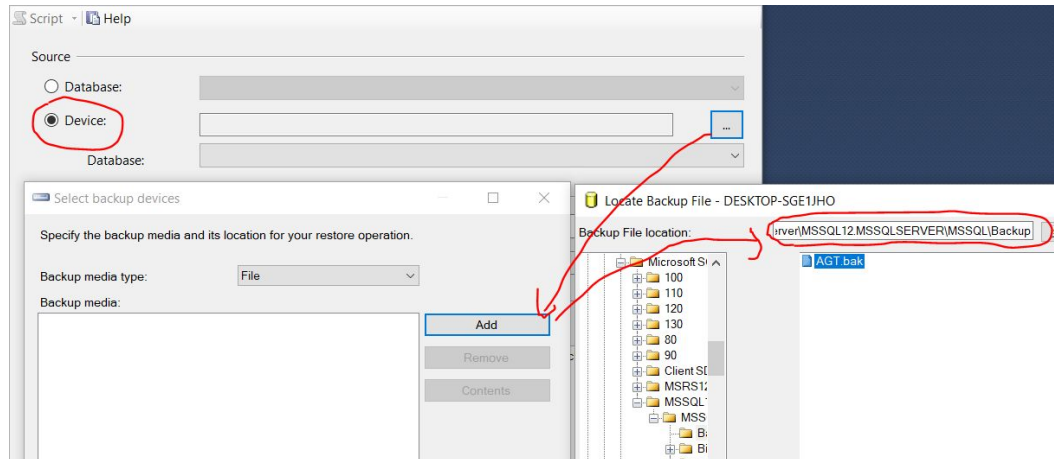
3 - Restaurar la base de datos en la máquina 1

A - Abrir SQL Server Management 2014

B - Ir al menú para restaurar la base



C - Pegar el .bak en el directorio con permisos y cargar la base

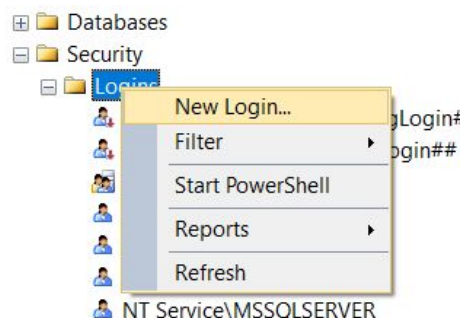


D - Validar que el connection string de nuestro "appsettings.json" de la carpeta pegada en wwwroot, sea el correcto.

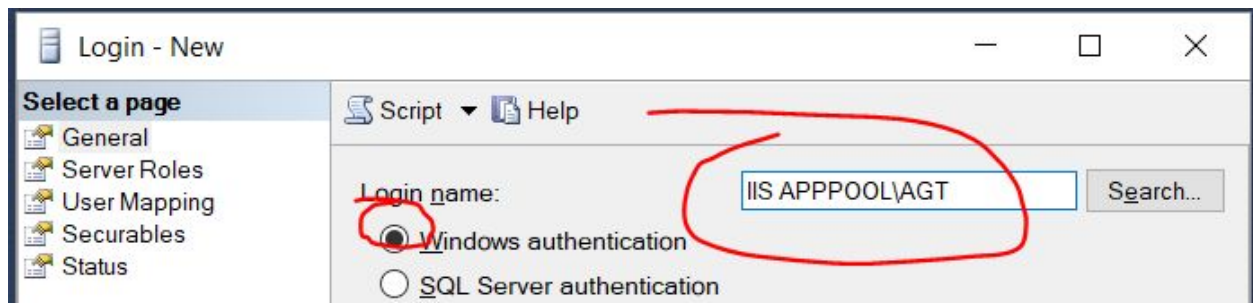
4 - Otorgar permisos al usuario del sitio de IIS sobre la base

El usuario que interactúa contra el motor de SQL Server es el usuario creado para el sitio de IIS que creamos anteriormente (IIS APPPOOL\XXXX). Debemos entonces configurar un nuevo inicio de sesión en el motor de SQL Server y otorgarle permisos para que pueda leer y escribir de a la base. No nos vamos a detener en explicar los roles y permisos de los usuarios de la base de datos, para esta defensa simplemente crearemos un usuario sysadmin.

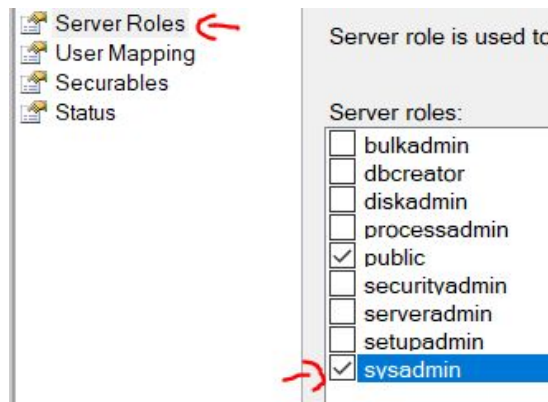
A - Ir a la carpeta de Security y crear un nuevo inicio de sesión



B - El Login Name corresponde a "IIS APPPOOL\[NOMBRE DEL GRUPO DE APLICACIONES DE IIS]" por ejemplo: el sitio que creamos se llama AGT y el grupo de aplicaciones se llama igual. Resultando en lo siguiente:



C - A este nuevo usuario debemos asignarle el ROL de deseado, en nuestro casos sysadmin. Una vez hecho esto, darle OK.



5 - Máquina 2.

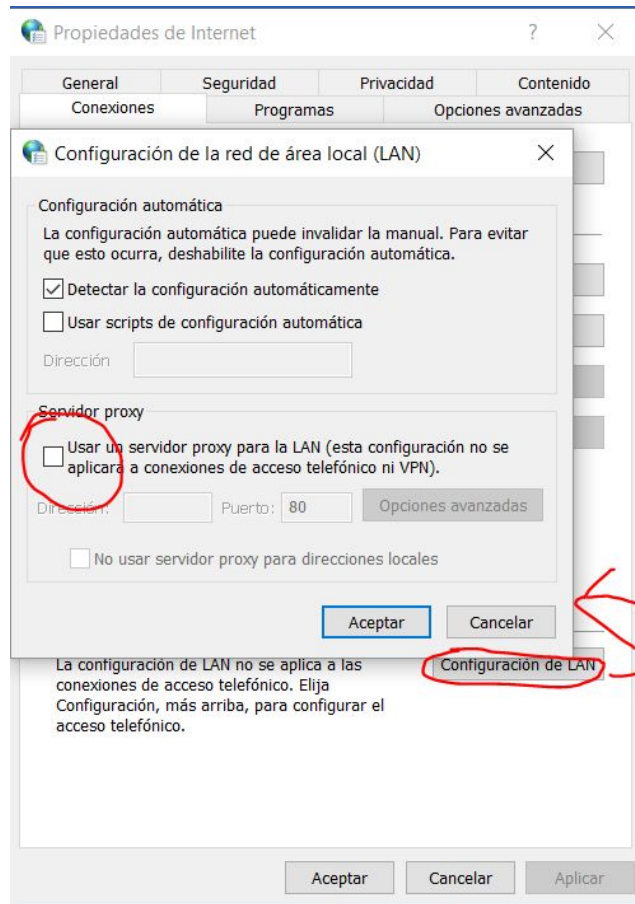
Si seguimos todos los pasos correctamente, nuestra aplicación ya debería estar funcionando en la misma máquina 1. Pero la api la vamos a consumir desde un postman en la máquina 2.

A - Para ello necesitamos obtener la ip de la máquina 1:

- Tecla inicio + R
- cmd
- Ejecutar comando "ipconfig" y obtener la ip de la máquina 1

B - En la máquina 2, abrir postman y cargar las colecciones de tests entregadas. Configurar en las variables de entorno o globales, la ip de la máquina 1 y el puerto en donde se encuentra nuestro sitio. Si usaron el puerto 80 no es necesario especificar el puerto.

C - Se debe bajar el proxy de la máquina 2 para que pueda conectarse a la máquina 1. Hay varias formas de hacer esto, desde el panel de control o desde chrome. Asegurarse de desmarcar la opción de proxy.



Esta opción vuelve a marcarse periódicamente, si la defensa tarda mucho, puede que en la mitad de la misma la configuración se reinicie.

6 - Antes de llamar a los docentes para la ejecución de la defensa, asegurarse que la api funcione correctamente desde la máquina 2. Probar peticiones GET, POST, PUT, [PATCH] y DELETE.