

# Visual Studio 2017/IIS Express: Unable to connect to the configured development Web server

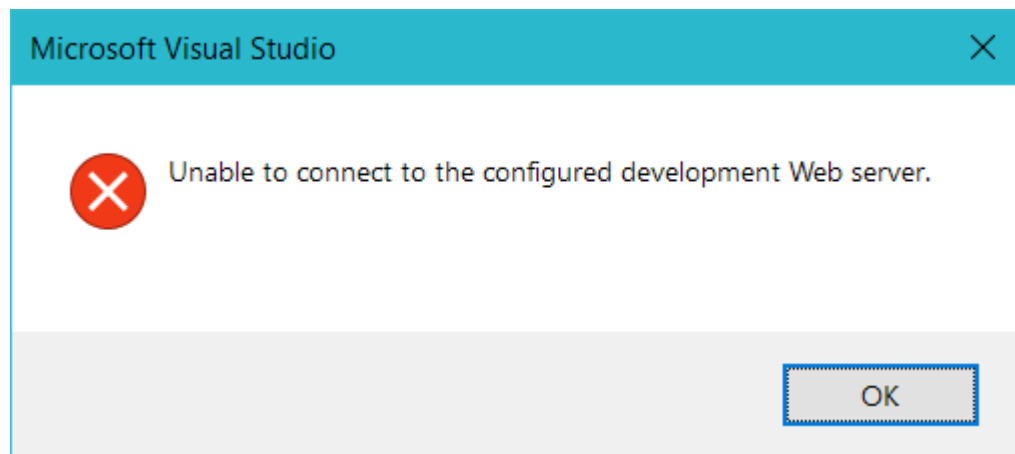
Asked 3 months ago Viewed 1k times

- ▲ *saving work in progress*
- 6 When trying to run a web-site from Visual Studio 2019 (or Visual Studio 2017, or Visual Studio 2015), i get the error:

▼

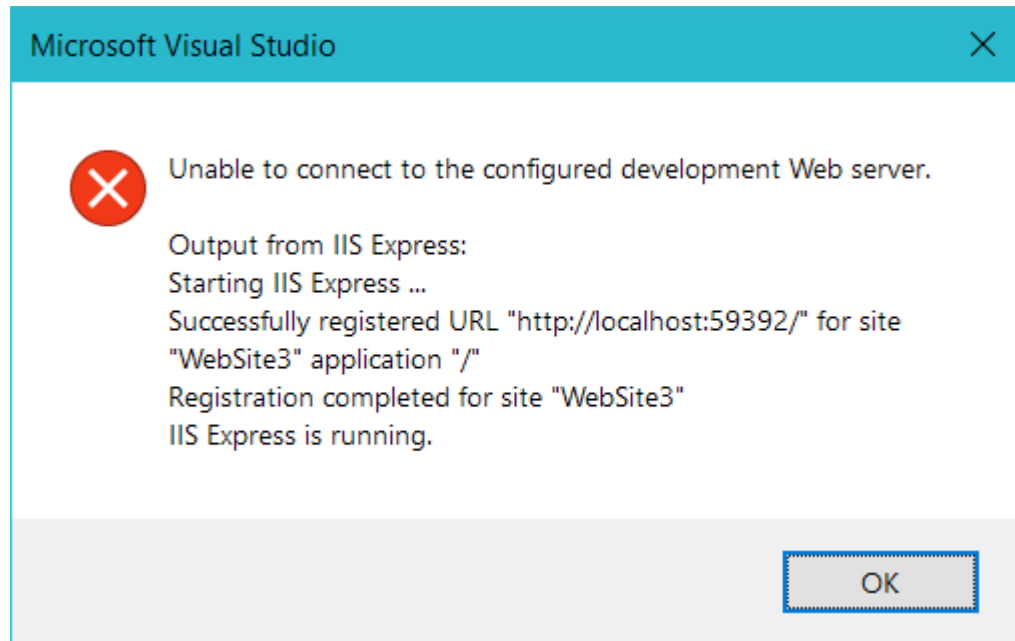
- **Visual Studio 2015:**

★  
1



Unable to connect to the configured development Web server.

- **Visual Studio 2017:**



Unable to connect to the configured development Web server.

Output from IIS Express:

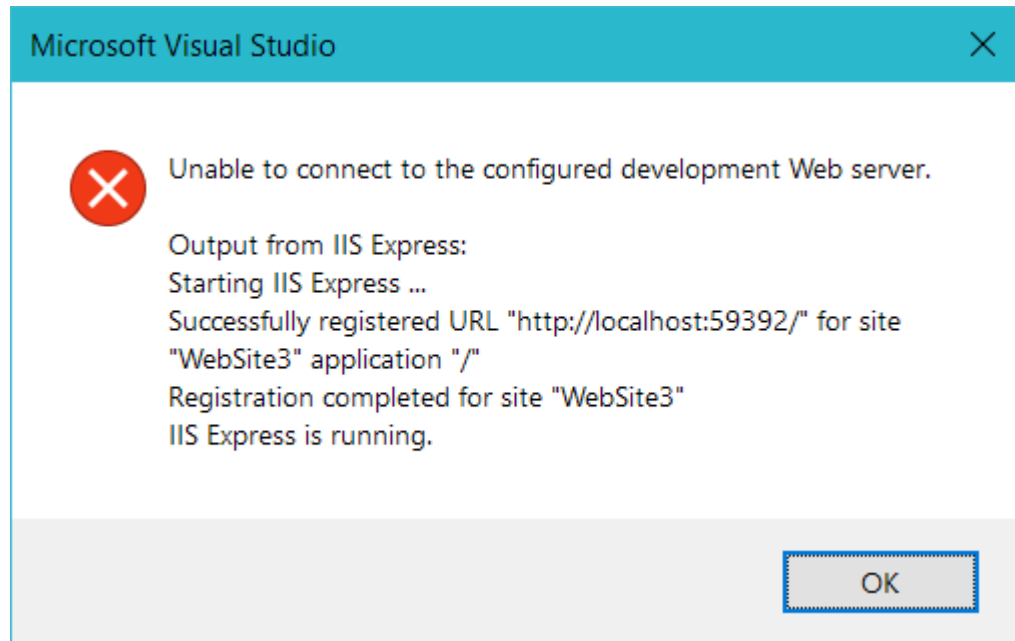
Starting IIS Express...

Successfully registered URL "<http://localhost:59392/>" for site "WebSite3" in application "/"

Registration completed for site "WebSite3"

IIS Express is running.

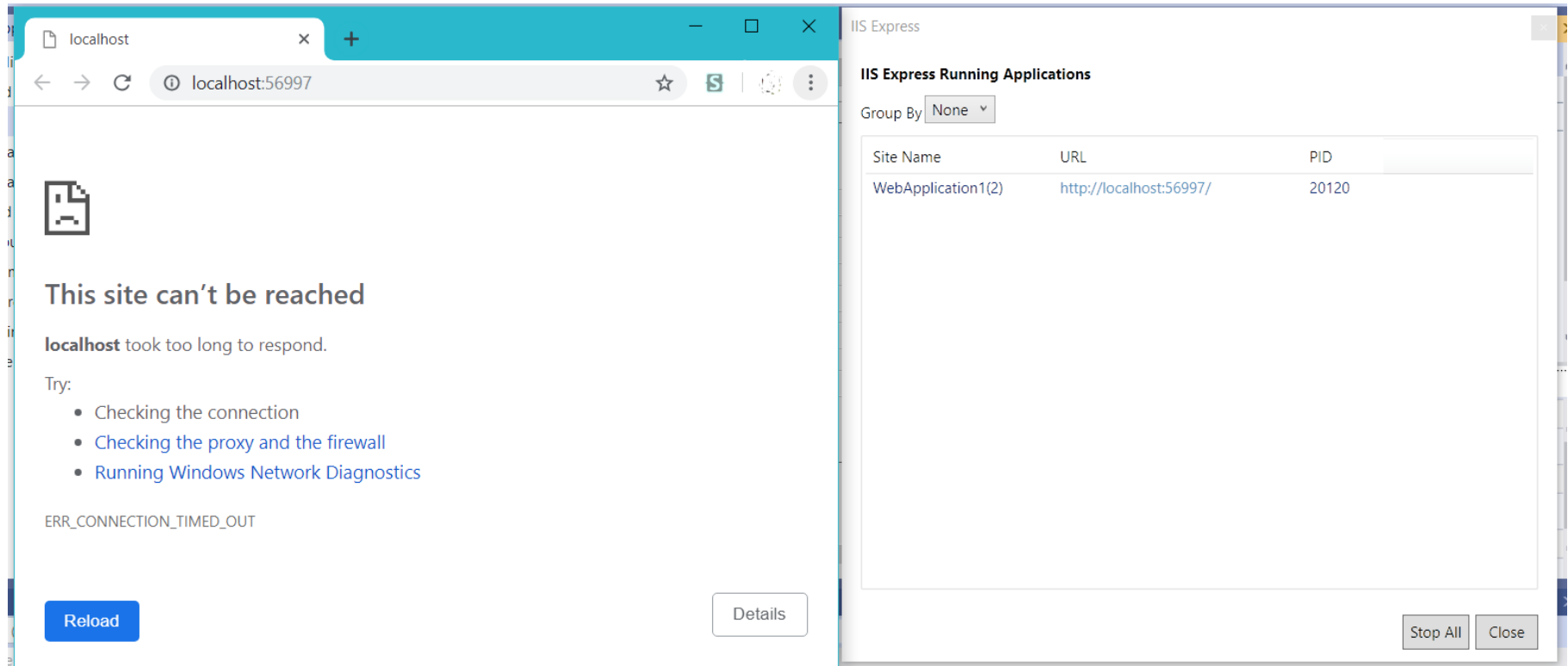
- **Visual Studio 2019:**



Unable to connect to the configured development Web server.

Output from IIS Express: Starting IIS Express ... Successfully registered URL "<http://localhost:59392/>" for site "WebSite3" application "/" Registration completed for site "WebSite3" IIS Express is running.

IISExpress **is** in fact running, and listening, but nothing actually works:



## What have you tried - [everything](#)

Things i've tried (from every other question on Stackoverflow):

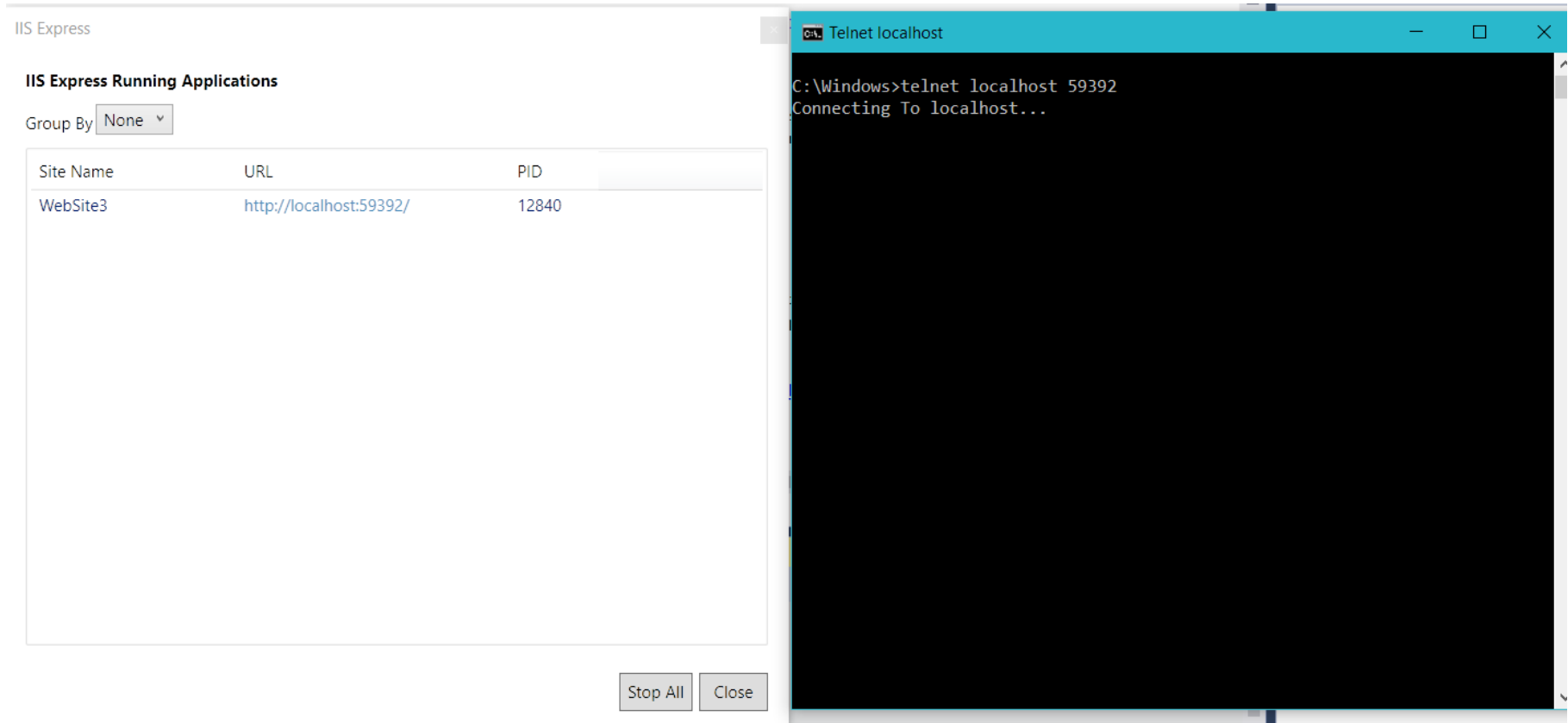
- `netsh http add urlacl url=http://localhost:56997/ user=everyone`
- run Visual Studio as an Administrator
- restart Windows
- delete the hidden `.vs` folder
- run the web-site from a different folder
- turn off the Windows Firewall
- change the **port** of the web-site
- change the **port** of the web-site and click **Create Virtual Directory**
- delete the **IISExpress** folder from my **Documents** folder
- install Visual Studio fresh (i installed 2019)

You can see IISExpress creating a listening socket, and IISExpress notification area icon shows that IISExpress considers the web-site running:

The screenshot shows the Windows Task Manager window with the 'Processes' tab selected. The list of processes includes System, [System Pr..., devenv.exe, SearchUI.exe, chrome.exe, and [System Pr... The 'PID' column shows values like 4, 0, 9332, 7816, 12744, and 0. The 'Status' column shows 'Running' for most processes. The 'CPU' column shows values like 0, 0, 0, 0, 0, and 0. The 'Private Bytes' column shows values like 4, 0, 9332, 7816, 12744, and 0. The 'Working Set' column shows values like 4, 0, 9332, 7816, 12744, and 0. The 'Session ID' column shows values like 4, 0, 9332, 7816, 12744, and 0. The 'User' column shows 'obsidian' for all processes. The 'IIS Express' notification area icon is visible in the system tray, showing 'IIS Express Running Applications' with a table of running applications.

Site Name	URL	PID
WebSite1(1)	http://localhost:60810/	4380

but it just won't respond to anything:



## Bonus Chatter - Windows built-in kernel mode web server

IISExpress.exe does not open a listening socket itself.

- See: [HTTP Server Sample Application \(archive.is\)](#)

Windows comes with a built-in kernel-mode mini webserver: `http.sys`. You, and IISExpress.exe, use this web web-server by calling:

```
//Initialize HTTP Server APIs
HttpInitialize(HTTPAPI_VERSION_1_0, HTTP_INITIALIZE_SERVER, null);

//Create a Request Queue
HANDLE requestQueue;
HttpCreateHttpHandle(ref requestQueue, 0);

/*
    Add URIs to listen on. We call HttpAddUrl for each URI.
```

The URI is a fully qualified URI and must include the terminating (/) character.

The IANA port numbers state ports 49152-65535 are for dynamic/private purposes.

HttpAddUrl for localhost on a port >= 49152 works fine for non-admins.

\*/

String url = "http://localhost:80/"; //Ports 1-1024 require administrator access

/\*

You can use netsh to modify the HttpServer api ACL to grant everyone access to port 80:

```
netsh http add urlacl url=http://localhost:80/ user=EVERYONE listen=yes
delegate=no
```

But it is useful to note that WCF already has an "Everyone Allow" entry for port 80, as long as your URL starts with "/Temporary\_Listen\_Addresses/"

WCF creates URLs of the form:

```
http://+80/Temporary_Listen_Address/[random guid]/
```

\*/

```
url = "http://+:80/Temporary_Listen_Addresses/{87CB7BDF-A52D-4496-AA1D-B6F60AC2841E}"/";
//WCF style url
```

//Or we can just use one above 1024

```
url = "http://localhost:2113/";
```

## Add the URL to your request queue

//Add the url to our request queue

```
ret = HttpAddUrl(requestQueue, url, null);
```

## And then you setup a loop to process the requests:

```
while (true)
```

```
{
```

```
    THTTP_REQUEST_ID requestId;
```

```
    Int32 requestBufferLength = sizeof(THTTP_REQUEST) + 16384;
```

```
    PHTTP_REQUEST request = GetMemory(requestBufferLength );
```

```
    DWORD bytesRead;
```

```
    ULONG res = HttpReceiveHttpRequest(requestQueue,
```

```
        requestId,          // Req ID
```

```
        0,                  // Flags
```

```
        request,            // HTTP request buffer
```

```
        requestBufferLength, // req buffer length
```

```

        ref bytesRead,      // bytes received
        null               // LPOVERLAPPED
    );
    if (res == NO_ERROR)
    {
        res = SendHttpResponse(requestQueue, request, 451, "Totally not NSL", "I don't
know what you mean ;)");
        if (res <> NO_ERROR)
            break;

        // Reset the Request ID to handle the next request.
        requestID = 0;
    }
    else if (res == ERROR_MORE_DATA)
    {
        /*
        The input buffer was too small to hold the request headers.
        Increase the buffer size and call the API again.

        When calling the API again, handle the request that failed by passing a
RequestID.
        This RequestID is read from the old buffer.
        */
        requestId = request.RequestId;

        //Free the old buffer and allocate a new buffer.
        requestBufferLength = bytesRead;
        FreeMem(request);
        request = GetMemory(requestBufferLength);
    }
    else if ((result == ERROR_CONNECTION_INVALID) and (requestID <> 0))
    {
        /*
        The TCP connection was corrupted by the peer when attempting to handle a
request with more buffer.
        Continue to the next request.
        */
        //Got invalid connection error
        requestID := 0;
    }
    else
    {
        // Other unhandled error; stopping processing requests
        break;
    }
}

```

This is all by of way of explaining why it is **System** that is listening, and not **IISExpress.exe**.



And if you read the comments, you'll notice why trying to perform `netsh http add urlacl` is mostly cargo-cult programming; you don't need to add permissions for ports over 1024.

## Related Questions

- [How to solve "Microsoft Visual Studio \(VS\)" error "Unable to connect to the configured development Web server"](#) *(the master question that everyone links to)*
- [Unable to connect to configured development server](#) *(no answer)*
- [Unable to connect to the configured development web sever](#) *(run as administrator)*
- [Create SDDL failed, Error: 1332](#) *(if you're running Spanish Windows)*
- [unable to connect to the configured development web server](#)
- [Unable to launch the configured Visual Studio development web server](#) *(create a new project from scratch)*
- [Unable to connect to the server Web development](#) *(try to use the WebDev.WebServer.exe that no longer exists)*
- [VS2012 ASP.Net error message unable to connect to the configured development web server](#) *(turn off firewall)*
- [unable to launch the configured development web server](#) *(reinstall Visual Studio)*
- [unable to connect to the configured development web server in c# project](#) *(create a new project)*
- [Microsoft Visual Studio 2015 Community - Unable to connect to the configured development Web server' error when trying to build website](#) *(reinstall Windows 10)*
- [VS 2008 "Unable to connect to the ASP.NET Development Server"](#) *(change the port)*
- [Unable to connect the configured development Web server Visual Studio 2013](#) *(delete IISExpress folder in Documents)*
- [Web Site Administration Tool: Unable to connect to SQL Server database \(using SQL Server Development Edition\)](#) *(aspnet\_regsql.exe; which is for creating users tables in an SQL Server database - cargo-cult programming anyone?)*
- [Unable to connect to web server](#) *(no answers)*
- [Unable to start debugging on the web server. Unable to connect to the webserver](#) *(reset IIS; which doesn't apply since i don't have the IIS feature added to Windows - VS shipped it)*
- [Unable to connect to ASP.Net Development Server issue](#) *(WebDev server aka Cassini; which no longer exists)*
- [Unable to connect to Visual Studio's Localhost Web Server](#) *(try not using IISExpress - which obviously i'm not going to do)*

visual-studio-2015

visual-studio-2017

visual-studio-2019

edited Apr 23 at 14:28

asked Apr 23 at 14:21



Ian Boyd

127k

198

711

1028

Is this question similar to what you get asked at work? [Learn more](#) about asking and sharing private information with your coworkers using Stack Overflow for Teams.

