

# Configure Windows for Local Webhook Testing Using ngrok



by Devin Rader (<https://www.twilio.com/blog/author/devinrader>) on March 14, 2014 (<https://www.twilio.com/blog/2014/03/configure-windows-for-local-webhook-testing-using-ngrok.html>)

[Like](#) [Tweet](#) [Follow @twilio](#)

As a developer running Windows discovering [ngrok](http://ngrok.com) (<http://ngrok.com>) was like discovering [manna from heaven](http://www.youtube.com/watch?v=g3DLxR2fOdM&t=0m12s) (<http://www.youtube.com/watch?v=g3DLxR2fOdM&t=0m12s>). Exposing my local web server to the public internet, simplified debugging and testing of services like [Twilio](https://www.twilio.com/try-twilio) (<https://www.twilio.com/try-twilio>) and a tool that did not make me install and configure other frameworks or language runtimes just to get started? It all had me [smiling](http://www.memecenter.com/fun/16054/na) (<http://www.memecenter.com/fun/16054/na>).

However, as I've shown ngrok to other developers running Windows I've also learned that how you run ngrok on Windows depends on your specific system and certain configurations require a few extra steps to make ngrok work.

In this post I'll show you which configurations I've found need a bit more love and how you can set them up to use ngrok to expose your local web server to the internet.

If you're confused by what this whole ngrok thing is and why it just might be better than [sliced bread](http://www.quickmeme.com/meme/3pm9uj) (<http://www.quickmeme.com/meme/3pm9uj>), head over to my friend Kevins [intro to ngrok](https://www.twilio.com/blog/2013/10/test-your-webhooks-locally-with-ngrok.html) (<https://www.twilio.com/blog/2013/10/test-your-webhooks-locally-with-ngrok.html>) blog post to learn all about how ngrok can be a valuable addition to your toolbox.

## The Changing World of Microsoft Web Servers

Over the past decade, Microsoft created a bunch of ways to run web applications on your local machine. Depending on your version of Windows and version of Visual Studio, you might be using the Visual Studio Web Development Server, IIS Express or even full blown IIS.

Here is a quick rundown of some of the different combinations that you can have installed:

- VS2005 – VS2010
- Visual Studio Web Development Server
- VS2010 SP1
  - Visual Studio Web Development Server – default

- IIS Express – optional
- VS2012 / VS2013 / VS2015
- IIS Express – default
- Visual Studio Web Development Server – optional

On top of those options if you have chosen to use the Web Sites project type in Visual Studio, you could also be running your site on a local IIS server.

Needless to say, it can be a bit confusing and depending on what you have installed it will change what you need to do to use ngrok. ngrok and the Visual Studio Web Development Server

If you are using the Visual Studio Web Development Server, using ngrok is simple. Simply start ngrok from a command line using the following command:

```
ngrok [port]
```

where the port argument is the port number that your website is using. For example if when you run your website the URL is `http://localhost:147114`, then you would start ngrok with this command:

```
ngrok 147114
```

Now anyone on the internet should be able to load your website using your ngrok URL:

```
http://6e5c9d59.ngrok.com/
```

If you start your website in debug mode and have breakpoints set, users (or services like Twilio) loading this URL will cause those breakpoints to be hit allowing you to debug your application in real time.

## ngrok and IIS or IIS Express

If however you are using IIS or IIS Express as your local web development server then you need to add the additional parameter “-host-header”:

```
ngrok 147114 -host-header="localhost:147114"
```

This will fake IIS Express or IIS into thinking the client is accessing via the localhost domain on your project's port number. Without this, you need to do a lot more configuration tweaking. Read on if you're curious.

### Server Bindings

Starting with IIS 7 and IIS Express 7.5, a new file named `applicationhost.config` is used as the root application configuration file. It uses a simple XML schema to define all sites, applications, virtual directories and application pools, as well as global defaults for the web server (similar to `machine.config` and the root `web.config` for .NET Framework settings).

You can find the config file for IIS Express here:

```
%USERPROFILE%\My Documents\IISExpress\config\applicationhost.config
```

and the config file for IIS here:

```
%WINDIR%\system32\inetsrv\config
```

When using Visual Studio 2012 to create a new website, or when changing an existing site in Visual Studio 2010 SP1 to use IIS Express, Visual Studio will add a new element to the `applicationhosts.config` file.

XHTML

```

1 <site name="WebSite1" id="1" serverAutoStart="true">
2   <application path="/">
3     <virtualDirectory path="/" physicalPath="%IIS_SITES_HOME%\WebSite1" />
4   </application>
5   <bindings>
6     <binding protocol="http" bindingInformation="*:147114:localhost" />
7   </bindings>
8 </site>

```

The site element contains a collection of bindings which by default tell IIS Express that this website should listen for and respond to HTTP requests made to any IP address bound to this machine, at a statically defined port, with a hostname of localhost.

In order to tell IIS Express that I also want this website to listen for and respond to requests to my ngrok domain I need to add a second binding:

XHTML

```

1 <bindings>
2   <binding protocol="http" bindingInformation="*:147114:localhost" />
3   <binding protocol="http" bindingInformation="*:147114:6e5c9d59.ngrok.com" />
4 </bindings>

```

Once I add this second binding I need to make sure to shut down and restart IIS Express in order to have it reload the binding information for this website.

## http.sys Security and Url Access Control Lists

At this point, if I started ngrok and told it to forward requests to my local web app, I'd see that my site still is not working as I expect, returning an HTTP 400 Invalid Hostname response. This is because http.sys which is the Windows component that handles incoming network requests made using the HTTP protocol, is not allowing the request to my ngrok domain to be served by IIS Express. By default http.sys only allows applications that want to respond to HTTP requests to bind to ports 80 and 443. It also blocks applications from serving external traffic.

In order to get http.sys to allow IIS Express to serve the external traffic on my local port, I have two options:

1. Run IIS Express with elevated privileges which allow http.sys to accept incoming http requests from external sources. To run with IIS Express with elevated privileges you need start Visual Studio as an Administrator.
2. Explicitly tell Windows to allow incoming requests to my ngrok domain by modifying the URL Access Control List (ACL).

Since its generally considered a best practice to run Visual Studio with least privilege, I normally choose the second option.

Adding my URL to the Access Control list is simple. Pop open a new elevated privilege command prompt using the *Run As Administrator* option and run one of the following commands depending on your OS:

Vista or newer:

```
netsh http add urlacl url=http://6e5c9d59.ngrok.com:147114/ user=everyone
```

XP:

```
httpcfg set urlacl /u http://6e5c9d59.ngrok.com:147114/ /a D:(A;;GX;;;WD)
```

Note that you **must include the trailing slash** at the end of the domain in order for the command to run successfully.

Now open a browser and point it at your ngrok domain and you should see your local web server is successfully serving content.

Regardless of which option you choose keep in mind that both options open your computer to the entire internet. Its a good idea to remember when you are done testing your application to shut down IIS Express if its running with elevated privilege or to remove the URL ACL which you can do using the following command:

Vista or newer:

```
netsh http delete urlacl url=http://6e5c9d59.ngrok.com:147114/
```

XP:

```
httpcfg delete urlacl /u http://6e5c9d59.ngrok.com:147114/
```

## Wrapping It Up

Serving web content to the public internet right from my own local web server using ngrok is simply put: amazing! Even with the few extra steps I have to take to get it running on Windows, the benefits of ngrok far outweigh the drawbacks. If you want a more in depth look at all of the features available in ngrok, head over to the [ngrok \(https://ngrok.com/\)](https://ngrok.com/) website, and also check out our own [intro to ngrok \(https://www.twilio.com/blog/2013/10/test-your-webhooks-locally-with-ngrok.html\)](https://www.twilio.com/blog/2013/10/test-your-webhooks-locally-with-ngrok.html) post on the Twilio blog.

Of course you might have a slight different system configuration from those I talked about in this post. If thats the case and you find you have other steps you need take in order to get ngrok running, I'd love to hear about it so feel free to drop me an [email \(mailto:devin@twilio.com\)](mailto:devin@twilio.com) or tweet me on [twitter \(http://twitter.com/devinrader\)](http://twitter.com/devinrader).

Like

Tweet

Follow @twilio



by Devin Rader (<https://www.twilio.com/blog/author/devinrader>) [devin@twilio.com](mailto:devin@twilio.com)  
(<mailto:devin@twilio.com>) [@devinrader \(http://twitter.com/devinrader\)](http://twitter.com/devinrader)

15 Comments

Twilio Blog

1 Login ▾

♥ Recommend 3

🔗 Share

Sort by Oldest ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS

Name

**Dav.id** • 3 years ago

Excellent! was struggling with VS2013 using IISEXPRESS, and so this has explained things very clearly thank you!

1 ^ | v • Reply • Share ›

**Kinzo** • 3 years ago

You are THE MAN!!!

1 ^ | v • Reply • Share ›

**Alex** • 3 years ago

Thanks a lot!!!

^ | v • Reply • Share ›

**Sean** • 3 years ago

For some reason, manually adding the binding in the applicationhost.config file caused 503 service unavailable errors for me. I added the binding in IIS Manager. inetmgr -&gt; Default Web Site -&gt; Bindings -&gt; Set hostname and port and I was up and running.

^ | v • Reply • Share ›

**Devin Rader** → Sean • 3 years ago

@Sean interesting. Was this in order to configure IIS Express or full strength IIS?

^ | v • Reply • Share ›

**Michael** • 2 years ago

Great post!

The URL change all the time that I enters to ngrok.exe so how I can define in iisexpress and http.sys to accept all subdomains of ngrok? This possible?

&lt;binding protocol="http" bindinginformation="\*:6964:\*.ngrok.io"/&gt;

and in Cmd:

netsh http add urlacl url=http://\*.ngrok.io:6964/ user=everyone

So any you don't need to define this everytime that I open ngrok.exe

^ | v • Reply • Share ›

**Devin Rader** → Michael • 2 years ago

Hey @Michael. I don't beleive this is posible because IIS does not (as I understand anyway) support wildcard host names:

<http://stackoverflow.com/qa...>

If you subscribe to ngrok, which I highly encourage, you get domain reservations which mean the domain won't change on each restart, plus you get a bunch of other awesome features.

^ | v • Reply • Share ›

**Liminal18** • 2 years ago

just so you know in windows 10 open netsh then typr http then add aclurl url=http://+:LocalHostPortHere/YourSubDomain.ngrok.io user= \Everyone

^ | v • Reply • Share ›

**Chandrasekhar Prathap** • 2 years ago

What an awesome article, super stuff mate..

^ | v • Reply • Share ›

**freddy** • 2 years ago

So hard for new people knows about ngrok espeacially in windows

^ | v • Reply • Share ›

**Phil Leggetter** • 2 years agoThere's simple around now. Via: <http://stackoverflow.com/a/...>

Run with the -host-header flag e.g.

ngrok http -host-header="localhost:8000" 8000

Where you change 8000 to your host in both the -host-header option and the port parameter.


2 ^ | v • Reply • Share ›

**Marcos Placona** → Phil Leggetter • 2 years ago

You're right. We've been using that lately and meant to get this post updated to say that. Great times we're living!

You're right. We've been using that lately and meant to get this post updated to say that. Great times we're living!

2 ^ | v • Reply • Share ›




**Santosh** → Phil Leggetter • a year ago

Yes. This is super useful Though, when I am using asp.net mvc's social authentication module, the return URL becomes http://localhost/signin-facebook for example. Any idea how to fix that to point to my ngrok subdomain?

I went with the solution of editing the config file and adding to the urlacl. Even, after that I had to run VS 2015 as Admin to be able to serve my application on the ngrok subdomain.


^ | v • Reply • Share ›




**jxy** • 7 months ago

forget the keyword "http"

^ | v • Reply • Share ›



**Rabia A** • 2 months ago



Attached is the link I got after I installed ngrok however when I type as the document suggests ngrok 8080 in command prompt it gives the following error message:

'ngrok' is not recognized as an internal or external command, operable program or batch file.

Am I missing something in here.

^ | v • Reply • Share ›

Search ...

Power modern communications. Build the next generation of voice and SMS applications.

**Start Building For Free (<http://twilio.com/try-twilio>)**

Posts By Stack

- .NET (<https://www.twilio.com/blog/tag/net>)
- Arduino (<https://www.twilio.com/blog/tag/arduino>)
- Java (<https://www.twilio.com/blog/tag/java>)
- JavaScript (<https://www.twilio.com/blog/tag/javascript>)
- PHP (<https://www.twilio.com/blog/tag/php>)
- Python (<https://www.twilio.com/blog/tag/python>)
- Ruby (<https://www.twilio.com/blog/tag/ruby>)
- Swift (<https://www.twilio.com/blog/tag/swift>)