

This site uses cookies to help deliver services. By using this site, you agree to the use of cookies. [Learn more](#)

Allen Conway's .NET Weblog

Exploring All Things .NET And Beyond...

Tuesday, Create A Self-Signed SSL Certificate Using IIS 7

If you are a .NET developer that creates IIS or self-hosted WCF services, then you will probably have the need at some point to secure the transport with a SSL certificate if using a http binding type. If you have a WCF service hosted by IIS, applying a SSL certificate is a bit more trivial because the endpoint configuration does not dictate the URL. The virtual directory in IIS will create the URL for your endpoint. However, if you are hosting your WCF service in a Windows Service, you dictate the endpoint and applying the SSL certificate is a little more involved. Because of this you may want to create a self-signed SSL certificate while still in development to ensure that your 'https' endpoint works correctly. With IIS websites, legacy .asmx services, or WCF hosted services, applying a SSL certificate happens after the fact via IIS and the initial testing with a SSL certificate may not even be desired. Regardless of your situation, the following tutorial shown you a simple procedure to create a self-signed certificate on your local machine.

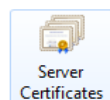
So what is a self-signed SSL certificate you may ask. A 'CA' or Certificate Authority is a trusted provided to generate a SSL certificate. Your local machine is a CA, but unfortunately and as expected the CA on your machine is not trusted (as should be) by any outside party, so any SSL certificate generated locally is good and trusted just there: locally! To get a SSL certificate generated by a trusted CA, you need to go to a commercial provider like 'GoDaddy' or 'Verisign' and purchase a SSL certificate. These Certificate Authorities are trusted on the internet and are able to provide SSL certificates with a set expiration time (i.e. 2 years out). Once applied, you can view the SSL certificate information of a secure site by pressing the secure lock icon in most browsers next to the URL, and will see who issued the SSL certificate, its expiration, and other public details like the public key.

If you happen to be on an Active Directory domain doing 'intranet' or internal software development, you may have a CA on the domain that will issue certificates which will be trusted within the domain. This is the way to go so one does not have to buy a GoDaddy or Verisign SSL certificate for every internal WCF service or hosted ASP.NET site. Check with your server folks (unless that's you!) to see if there is a CA that issues SSL certificates trusted by all on the domain.

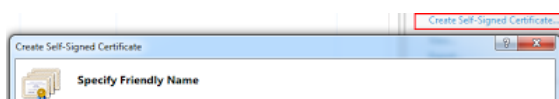
If you don't have IIS7, generating a SSL certificate is still possible. You just do the similar steps under the 'Directory Security' tab in IIS for a given site. Using IIS to create the certificate does not mean we have to host our service in IIS. It just has a convenient 'wizard' style interface to generate certificates and place them in the proper 'stores'. You can manually decide which stores your certificate is placed in and trusted by using the Certificate Manager MMC snap-in. That is really off topic for this post, but good to see how local and purchased certificates are managed. The snap-in is not under the administrator tools by default so look to the following link if interested in adding or accessing this MMC utility:

How to Add Certificate Manager to Microsoft Management Console

To begin a new certificate request, open IIS7 and click on the root element which is your machine or server node. Locate the 'Server Certificates' icon and double click it:



On the right-hand side of the screen select the 'Create Self-Signed Certificate' link which will display the following dialog:



About Me



Allen Conway

I am a Magenit Senior Consultant that is an advocate of Microsoft .NET technologies and the Microsoft platform both professionally and personally. I enjoy the challenge and creativity behind software engineering, and hope during this process to extract some of my thoughts and ideas in order to give back to others in the community here on this Blog.

[VIEW MY COMPLETE PROFILE](#)

Awards



Microsoft is a registered trademark of Microsoft Corporation in the United States and/or other countries.

Disclaimer

This weblog does not represent the thoughts, intentions, plans or strategies of my employer. It is solely my opinion.

LIVE! 360



[Twitter](#)



[Follow Me On Twitter](#)

[Contact Me](#)

[RSS Link](#)



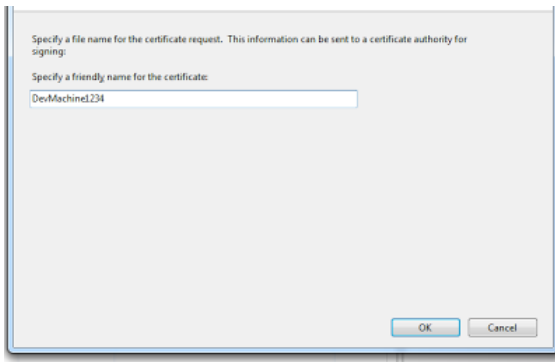
[Click here for the RSS link](#)

[Tags](#)

[.asmx](#)
[.ASMX Services](#)
[.NET](#)

[Blog Archive](#)

► 2014 (5)
► 2013 (26)
► 2012 (26)
▼ 2011 (31)
► December (1)
► November (2)
► October (2)



This is the important part which is dictating the friendly-name of your certificate. For local WCF development you really have (2) choices: name the certificate 'localhost' or the name of your machine. I recommend the name of your machine as it is more explicit. So in the example below my machine name is 'DevMachine1234'. The name is important for hosting WCF services and applying a SSL certificate to the exposed endpoint. If the SSL name does not match the domain of the hosted service it will not work. In the case of local development, name the certificate the same name as your machine.

After completing the request you will see the SSL certificate has been generated by the local machines CA, the friendly name, and the certificate hash.



The hash value will be important in the next post about applying this self-signed certificate to a port number that is dictated in the WCF configuration for a service hosted by a Windows Service. If you are applying the SSL certificate to a IIS hosted service or site, all you have to do is select it from the dropdown when configuring the 'https' binding in IIS7.

Posted By Allen Conway On 6/28/2011 10:22:00 PM
Labels: ASP.NET, Security, SSL, WCF

Related Postings:

- [AspConf - A Free Virtual Conference On ASP.NET](#)
- [How To Handle OnChange / SelectedIndexChanged Type Event In JavaScript For The AJAX ComboBox](#)
- [One Of A Thousand Possible Reasons For The "Invalid Postback Or Callback Argument" Exception](#)
- [Specifying Document Compatibility Modes For ASP.NET Intranet Sites Using IE8](#)
- [September MSDN Webcast Training For Newbie ASP.NET Developers](#)
- [Help! My ASP.NET Page Is Generating A JavaScript "Object Expected" Error Now That I Am Using JQuery, Plus A Little On URLs In ASP.NET](#)
- [Using Basic Authentication In REST Based Services Hosted In IIS](#)
- [RESTful Services: Authenticating Clients Using Basic Authentication](#)
- [Creating A WCF RESTful Service And Secure It Using HTTPS Over SSL](#)
- [Create A Self-Signed SSL Certificate Using IIS 7](#)
- [How To: Set Up BitLocker Full Disk Encryption + Pre-Boot Pin In Windows 7 Ultimate](#)
- [Applying And Using A SSL Certificate With A Self-Hosted WCF Service](#)
- [Tip On Preventing Web Discover Service Error](#)
- [Download FTP Files Using FTP Over SSL \(SFTP\) In .NET](#)
- [Visual Studio LIVE! Day 1 - WCF And Web API](#)
- [Exposing Multiple Binding Types For The Same Service Class In WCF](#)
- [How To Serialize A SyndicationFeed Object To Be Returned From WCF](#)
- [How To Fix When A WCF Service Does Not Debug Via The "WCF Test Client" Tool](#)
- [Fixing The "Attempting To Access A Service In A Cross-domain Way..." Error When Consuming A WCF Service Running Via A VS.NET 'Localhost' Binding](#)
- [HTTP Error 405 When Making A PUT Or DELETE Call To Web API Service](#)

7 Comments:

Tim said...

- 101 Topics
- ADO.NET
- Advice
- Agile
- AJAX
- Android
- Architecture
- Architecture
- ASP.NET
- asynchronous
- Asynchronous Programming
- Awards
- Azure
- Basic Authentication
- Best Practices
- Bindings
- BitLocker
- Book Review
- Books
- Building
- Business Logic
- C#
- Caching
- Career
- Cassini
- Code Camp
- code refactoring
- Computer Hardware
- Conference
- Conversions
- Custom Built PC
- DDD
- Debate
- Debugging
- Decompiling
- Delete Files
- Design
- Design Patterns
- DI
- DropDownList
- Editorial
- Encryption
- Entity Framework
- Enums
- Errors
- Exceptions
- Expression Blend
- Expression Encoder
- Freebies
- FTP
- Future Predictions
- History
- HTML
- HTML5
- IIS
- IIS 7.0
- IIS 7.5
- Internet Explorer
- September (6)
- August (2)
- July (3)
- ▼ June (2)
 - Create A Self-Signed SSL Certificate Using IIS 7
 - My New Computer: A Developers Dream
- May (1)
- April (3)
- March (2)
- February (3)
- January (4)
- 2010 (42)
- 2009 (13)

We have been using self signed certs for ages but a lot of people keep telling me about the pop up boxes they are getting when they try to complete there purchase on the site. So im thinking of dumping open SSL and just [buy ssl](#) cert to stop it, though i do not know which on to buy. Do i need a wildcard ssl if im running two websites???

August 8, 2011 At 8:20 AM

Allen Conway said...

I don't commit to being a 'SSL' expert, but was rather here trying to convey how to use local *test* certificates. As far as not getting the 'prompts' you mentioned, if they are in regards to a non-matching or a non-verified SSL certificate then, yes, purchasing a valid SSL should verify your sites security and end the security prompts. As far as the wildcard SSL question, I *think* that is the answer for multiple sites as long as they have the same domian name (i.e. *.mycompanyname.com). You can check with the SSL provider for the exact details on that question.

August 15, 2011 At 2:43 PM

Vicky said...

Me and my husband are owners of a small buisness looking to set up an online environment in which our customers can feel confident in making their financial transactions with us. Thankyou for the help in this, I am still struggling to [buy ssl certs](#), but I may just purchase from godaddy as advised

October 11, 2011 At 9:05 AM

Ecommerce developer said...

Your post is truly informative for me and i am so grateful to you for sharing this informative post here. really great job done by you.

January 2, 2012 At 12:59 AM

Cheap WildCard SSL said...

Useful information for creation of self signing certificate on IIS 6. Thanks for sharing it and we really appreciate your contribution to SSL installation education. We are going to share your blog on our social.

- Thanks Cheapest SSL

March 21, 2012 At 5:04 AM

Anonymous said...

thank you!

May 30, 2012 At 9:17 PM

Jimmy Jarred said...

I really appreciate you for sharing this detail. I am struggling to obtain an ssl certificate for my website but now with the help of this article I can easily create one. Thanks.

[digital certificates](#)

January 13, 2014 At 8:41 AM

Post a Comment

Enter your comment...

Comment as:

Publish

Preview

[Newer Post](#)

[Home](#)

[Older Post](#)

[introduction](#)
[JavaScript](#)
[jQuery](#)
[Lambda Expressions](#)
[Learning](#)
[LightSwitch](#)
[LINQ](#)
[LINQPad](#)
[ListBox](#)
[Metro Apps](#)
[Microsoft](#)
[Moq](#)
[MS](#)
[MSDN](#)
[MSEvents](#)
[MVC](#)
[Natural Language Processing](#)
[Non .NET Related](#)
[NSIS](#)
[NuGet](#)
[Object Data Source](#)
[OData](#)
[ONETUG](#)
[OOP](#)
[Opinion Piece](#)
[Performance Counter](#)
[Periodic Execution](#)
[PLINQ](#)
[Project Methodology](#)
[Promotions](#)
[Prototyping](#)
[Recursion](#)
[reflection](#)
[REST](#)
[Security](#)
[Sentiment Analysis](#)
[Setup Packages](#)
[Silverlight](#)
[Smart Phones](#)
[Software Engineering](#)
[SQL](#)
[SQL2000](#)
[SQL2008](#)
[SSL](#)
[System.IO](#)
[TechEd](#)
[thoughts](#)
[Tools](#)
[TPL](#)
[Training](#)
[TSQL](#)
[Twitter API](#)
[UI Design](#)
[Unit Testing](#)
[ValueInjector](#)
[VB.NET](#)
[VB6](#)
[Visual Studio Live](#)

Visual Studio Online
VS.NET 2008
VS.NET 2010
VS.NET 2012
VS.NET 2013
VS.NET IDE
VSLive
VSLive 2013
VSS
WCF
WCF Data Services
Web API
WebForms
Windows 7
Windows 8
Windows Authentication
Windows Server
Windows Service
WiX
WMI
XML

Allen Conway's .NET Weblog

