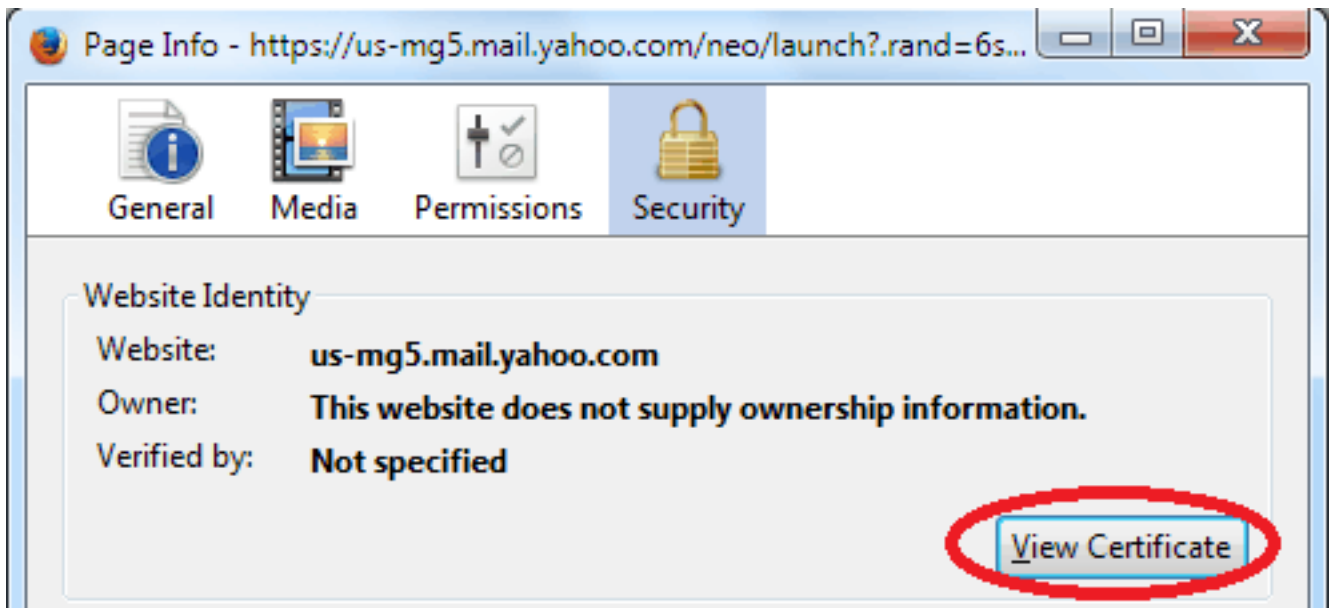


اگر به مرورگرها دقت کرده باشید، امکان نمایش SSL Server Certificate یک سایت استفاده کننده از پروتکل HTTPS را دارند. برای مثال در فایرفاکس اگر به خواص یک صفحه مراجعه کنیم، در برگه امنیت آن، امکان مشاهده جزئیات مجوز SSL سایت جاری فراهم است:



سؤال: چگونه می‌توان این مجوزها را با کدنویسی دریافت یا تعیین اعتبار کرد؟

قطعه کد زیر، نحوه دریافت مجوز SSL یک سایت را نمایش می‌دهد:

```
using System;
using System.Diagnostics;
using System.IO;
using System.Net;
using System.Security.Cryptography.X509Certificates;

namespace DownloadCerts
{
    class Program
    {
        static void Main(string[] args)
        {
            // صرفنظر از خطاهای احتمالی مجوز
            ServicePointManager.ServerCertificateValidationCallback = delegate { return true; };

            var url = "https://pdfreport.codeplex.com";
            var request = WebRequest.Create(url) as HttpWebRequest;
            request.Method = WebRequestMethods.Http.Head;
            using (var response = request.GetResponse())
            { /* در اینجا مجوز، در صورت وجود دریافت شده */ }

            if (request.ServicePoint.Certificate == null)
                return;

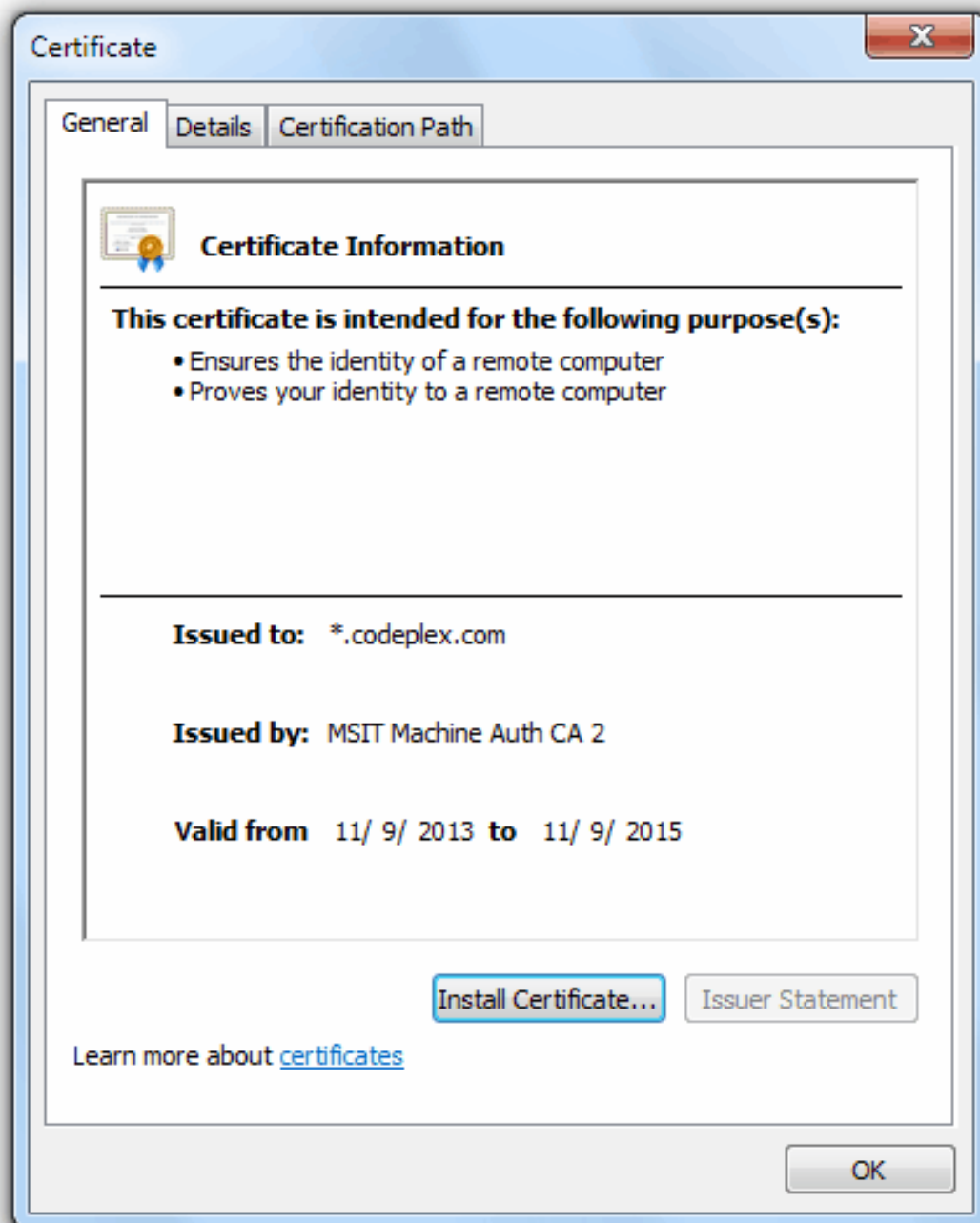
            // ذخیره سازی مجوز در فایل
            var cert = new X509Certificate2(request.ServicePoint.Certificate);
            Console.WriteLine("Expiration Date: {0}", cert.GetExpirationDateString());
            var data = cert.Export(X509ContentType.Cert);
        }
    }
}
```

```
        File.WriteAllBytes("site.cer", data);  
        Process.Start(Environment.CurrentDirectory);  
    }  
}
```

ممکن است مجوز یک سایت معتبر نباشد. کلاس `WebRequest` در حین مواجه شدن با یک چنین سایت‌هایی، یک `WebException` را صادر می‌کند. از این جهت که می‌خواهیم حتماً این مجوز را دریافت کنیم، بنابراین در ابتدای کار، `ServerCertificateValidation` را غیرفعال می‌کنیم.

سپس یک درخواست ساده را به آدرس سرور مورد نظر ارسال می‌کنیم. پس از پایان درخواست، خاصیت `request.ServicePoint.Certificate` با مجوز SSL یک سایت مقدار دهی شده است. در ادامه نحوه ذخیره سازی این مجوز را با فرمت `cer` مشاهده می‌کنید.

Name	Date modified	Type
 DownloadCerts.exe	۲۱ مهر ۱۳۹۲ ۱۲:۲۷ ق.ظ	Application
 DownloadCerts.pdb	۲۱ مهر ۱۳۹۲ ۱۲:۲۷ ق.ظ	PDB File
 DownloadCerts.vshost.exe	۲۱ مهر ۱۳۹۲ ۱۲:۲۹ ق.ظ	Application
 DownloadCerts.vshost.exe.manifest	۲۶ اسفند ۱۳۸۸ ۰۹:۳۹ ب.ظ	MANIFEST File
 site.cer	۲۱ مهر ۱۳۹۲ ۱۲:۲۹ ق.ظ	Security Certificate



نظرات خوانندگان

نویسنده: حمید حسین وند
تاریخ: ۱۶:۲۸ ۱۳۹۳/۰۱/۲۲

سلام؛ وقتی این گواهی یا certificate رو دانلود کردیم به چه دردمون میخوره؟ یعنی کاراییش برای ما چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۶:۵۸ ۱۳۹۳/۰۱/۲۲

جهت بررسی اعتبار آن می‌تواند مفید باشد. مثلا نوشتن برنامه‌ای مانند [SSL Certificate Verifier](#)

بهبود سرعت دریافت بسته‌های نیوگت

در کشور بسیاری از اوقات دسترسی به پروتکل HTTPS به کندی صورت می‌گیرد. گاهی از اوقات نیز این دسترسی غیر ممکن می‌شود تا حد دریافت چند بایت در دقیقه. همین مساله تاکنون بر روی بسیاری از مسایل دیگر نیز تاثیر گذار بوده است؛ برای مثال اگر یک مخزن کد را مثلا در CodePlex یا GitHub داشته باشید، چون تمام Commitها از طریق همین پروتکل امن صورت می‌گیرد، کار کردن با آنها بسیار مشکل خواهد شد. نمونه‌ی دیگر آن دسترسی به NuGet است. فید NuGet در VS.NET به https تنظیم شده است. اگر دسترسی به https برای شما به کندی صورت می‌گیرد فقط کافی است مسیر فید آنرا در منوی Tools، گزینه‌ی Options، ذیل قسمت Package manager یافته و به <http://nuget.org/api/v2> تغییر دهید؛ یعنی به http خالی، بجای Http s؛ تا سرعت دریافت بسته‌های NuGet مورد نظر افزایش یابند.

دریافت مستقیم بسته‌های نیوگت

برای دریافت بسته‌های نیوگت که دارای پسوند nupkg هستند، اما در اصل یک فایل zip بیشتر نیستند، الزامی به استفاده از ابزار و افزونه نیوگت در VS.NET نیست. می‌توان این بسته‌ها را به صورت مستقیم نیز دریافت کرد. برای مثال اگر آدرس بسته‌ای در سایت NuGet به صورت زیر است:

<https://www.nuget.org/packages/PropertyChanged.Fody>

برای دریافت مستقیم آن کافی است آدرس ذیل را درخواست کنید:

<https://www.nuget.org/api/v2/package/PropertyChanged.Fody/1.42.0>

یک api/v2 به این لینک اضافه می‌شود به همراه شماره نگارش مدنظر برای دریافت:

<https://www.nuget.org/api/v2/package/{packageID}/{packageVersion}>

و یا برای مثال در [سایت نیوگت](#) عضو شوید و سپس به آن لاگین کنید. به این ترتیب با مراجعه به هر کتابخانه‌ای که در آنجا آپلود شده، یک لینک download در کنار صفحه، سمت چپ ظاهر می‌شود. با کلیک بر روی آن فایل nupkg آن کتابخانه قابل دریافت خواهد بود. این فایل در حقیقت یک فایل zip است. بنابراین کار کردن با محتویات آن ساده‌است. به صورت خلاصه:

لینک اصلی کتابخانه: <https://www.nuget.org/packages/Twitter.Bootstrap.RTL.Less/3.0.0>

لینک دانلود آن: <https://www.nuget.org/api/v2/package/Twitter.Bootstrap.RTL.Less/3.0.0>

راه دیگر، ساخت دستی این آدرس است:

<https://az320820.vo.msecnd.net/packages/propertychanged.fody.1.42.0.nupkg>

که در حقیقت تشکیل شده است از:

<https://az320820.vo.msecnd.net/packages/{name}.{version}.nupkg>

اگر نام آخرین بسته ارسالی PropertyChanged.Fody 1.42.0 باشد. فقط کافی است این دو قسمت را، یعنی نام و شماره نگارش را با یک نقطه به هم متصل کنید و سپس به انتهای آن، پسوند nupkg را اضافه نمایید. این فایل در آدرس <https://az320820.vo.msecnd.net/packages> به صورت مستقیم قابل دریافت است.

اهمیت تنظیمات IE

اگر پیام قطع شدن اتصال یا مشکلات DNS را در کنسول NuGet در VS.NET دریافت می‌کنید، ابتدا سعی کنید همان روش ذکر شده

در ابتدای بحث را امتحان کنید. اگر کار نکرد احتمالا مشکل از تنظیمات IE است. برای مثال اگر بر روی تنظیمات اتصالی شما در IE یک پروکسی غیرقابل دسترسی در زمان جاری، تنظیم شده باشد، این مساله مستقیما بر روی اتصالات برنامه‌های دات نت نیز تاثیر گذار است. بنابراین ابتدا لینک `nupkg` را که ساخته‌اید یکبار با IE امتحان کنید. اگر قابل دریافت نبود یعنی تنظیمات آن به هم ریخته است و این مساله بر روی بسیاری از برنامه‌های دیگر نیز تاثیر گذار است.

نظرات خوانندگان

نویسنده: رامین علیرضایی
تاریخ: ۱۵:۹ ۱۳۹۲/۰۹/۰۸

با سلام
ممنون از مطلب خوبتون
برای TFS چه باید کرد؟ (البته bitbucket هم همین وضع رو داره)
سه روز هستش که به صورت عادی در دسترس نیست.
با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۷:۳۳ ۱۳۹۲/۰۹/۰۸

استفاده از <http://nuget.org/api/v2> در اینجا چون [اجباری](#) به استفاده از HTTPS نبوده (از طرف طراحان این سرویس)، میسر شده وگرنه این تغییر فایده‌ای نداشت.

نویسنده: آریا
تاریخ: ۸:۵۰ ۱۳۹۲/۰۹/۱۰

سلام
یه راه هم هست برای دوستانی که دسترسی مستقیم به سایت Nuget ندارن. من مدت‌ها از ساخت دستی آدرس استفاده میکردم که خوب درباره بسته‌هایی که وابستگی داشتن خیلی اذیت کننده بود چون فقط خود فایل رو میگیره. ولی جایگزین خوب این قضیه فایل اجرایی Nuget Command Line هستش [اینجا](#) و تمام و وابستگی‌ها رو هم خودش دانلود میکنه!

نویسنده: محسن خان
تاریخ: ۹:۲۵ ۱۳۹۲/۰۹/۱۰

ایمن از همون سایت دانلود می‌کنه. فرقی نداره. فایل‌ها در یک آدرس بیشتر نیست.

نویسنده: آریا
تاریخ: ۱۷:۲۷ ۱۳۹۲/۰۹/۱۰

شما وقتی این آدرس رو وارد کنید

<https://www.nuget.org/api/v2/package/PropertyChanged.Fody/1.42.0>

[فقط بسته‌ی مربوطه رو میگیرید و هیچ کدوم از وابستگی‌ها دانلود نمیشن! شما چون توی تنظیماتتون آدرس‌های مربوطه Nuget مثل](#)

[/https://www.nuget.org/api/v2](https://www.nuget.org/api/v2)

رو دارید خود Nuget تمام وابستگی‌ها رو هم براتون دانلود و نصب میکنه. من هم عرض کردم در صورتیکه از داخل VS دسترسی به سایت Nuget ندارید!

اگر دسترسی نداشته باشید بسته‌های پیش نیاز دریافت نخواهند شد , در نتیجه نصب صورت نمیگیره. پس باید تک تک آدرس‌ها رو بسازید و بسته‌های پیش نیاز رو دریافت کنید ولی بوسیله‌ی Nuget.exe میتونید همه‌ی بسته‌ها رو یکجا دریافت کنید. شاید حالت خیلی خاصی باشه ولی بعضی جاها که دسترسی به اینترنت برای سیستمی که VS روش نصبه محدوده و نیاز دارن تا بسته‌ها رو بصورت آفلاین نصب کنن این متود کارآیی زیادی داره.

نویسنده: محسن خان
تاریخ: ۱۹:۲۳ ۱۳۹۲/۰۹/۱۰

مهم نیست که وابستگی‌ها دریافت میشن یا نه. گاهی شخصی نمی‌دونه مشکلش چیه. چرا اتصال برقرار نمیشه. اصلا باید به کجا متصل بشه. الان می‌دونه که لینک نهایی‌اش کجاست و وابسته به ابزار نیست. می‌تونه خودش یک حداقل کار دیباگ رو انجام بده. می‌دونه اگر دریافتش چند بایت هست و بعد قطع میشه به چه دلیلی بوده. اگر خطای DNS دریافت می‌کنه، این به کجای سیستم ممکنه مرتبط باشه. مطلب جاری بیشتر حال و هوای دیباگ رو داره تا استفاده از ابزارهای آماده.