عنوان: **تشخیص نوع فایل با استفاده از محتوای فایل**
نویسنده: سالمیان
تاریخ: ۱۳۹۴/۰۴/۲۴ ۰:۵۵
آدرس: www.dotnettips.info
گروه‌ها: MVC, Security

بی‌شک اگر در سایت خود بخشی را برای دریافت فایل‌های کاربر قرار داده باشید یکی از دغدغه‌های شما اعمال فیلتر و محدودیت روی نوع فایل‌های آپلود شده توسط کاربران خواهد بود. ممکن است سیاست شما پذیرای فایل هایی با پسوند خاص (برای مثال فقط عکس) باشد ولی **هیچ تضمینی وجود ندارد که فایلی با پسوند مورد نظر شما محتوایی مشابه با پسوند خود داشته باشد** .
اگر بخواهیم دقیق‌تر به این موضوع نگاه کنیم فرض می‌کنیم شما در وب سایت خود قسمتی برای آپلود عکس‌های کاربر قرار داده باشید و با مکانیزمی مانند این روش صحت نوع فایل‌های آپلود شده توسط کاربر را بررسی کنید.
اگر شخصی به قصد تخریب و هدر دادن فضای ذخیره سازی شما فایلی با محتوایی غیر از عکس (برای مثال یک فایل اجرایی) را تغییر پسوند داده و به جای عکس آپلود کند چه اتفاقی می‌افتد؟!
دو دلیل مهم برای چک کردن محتوای فایل خواهیم داشت:
**1- جلوگیری از اتلاف حافظه ذخیره سازی** (ممکن است شخص مهاجم هزاران فایل چند مگابایتی با محتوایی غیر از پسوند فایل بر روی سرور قرار دهد)
**2- جلوگیری از اختلال در نمایش فایل‌های آپلود شده** (بی شک عدم نمایش عکس در سایت چهره خوبی نخواهد داشت و ممکن است اعتبار سایت را زیر سوال ببرد)

همیشه برای چک کردن نوع فایل باید به دونکته توجه داشت: یکی آنکه پسوند فایل را حتما چک کنیم تا مطابق فیلتر و سیاست مورد انتظار ما باشد ، دوم اینکه به روشی که در ادامه توضیح خواهیم داد، با استفاده از محتوای باینری فایل، MimeType آنرا تشخیص دهیم.
برای اینکار ما از یک تابع API استفاده میکنیم که با استفاده از 255 بایت ابتدایی محتوای فایل، نوع فایل را مشخص می‌کند. این تابع API که FindMimeFromData نام دارد، در فایل urlmon.dll قرار دارد و گویا توسط مرورگر IE برای چک کردن نوع فایل، بر اساس محتوای آن مورد استفاده قرار می‌گیرد. ما نیز از این تابع در دات نت بهره گرفته و با پاس دادن آرایه‌ای از بایتها به آن نوع فایل خود را مشخص می‌کنیم.
کلاسی برای اینکار تهیه شده که در زیر مشاهده میکنید:

```
using System;
using System.Runtime.InteropServices;
using System.Reflection;

namespace Parsnet.Core
{

    public class MimeTypeDetector
    {
        [DllImport(@"urlmon.dll", CharSet = CharSet.Auto)]
        private extern static System.UInt32 FindMimeFromData(
            System.UInt32 pBC,
            [MarshalAs(UnmanagedType.LPStr)] System.String pwzUrl,
            [MarshalAs(UnmanagedType.LPArray)] byte[] pBuffer,
            System.UInt32 cbSize,
            [MarshalAs(UnmanagedType.LPStr)] System.String pwzMimeProposed,
            System.UInt32 dwMimeFlags,
            out System.UInt32 ppwzMimeOut,
            System.UInt32 dwReserverd
        );

        public string GetMimeType(byte[] content)
        {
            var result = "unknown/unknown";

            try
            {
                byte[] buffer = new byte[256];
                var length = (content.Length > 256) ? 256 : content.Length;
                Array.Copy(content, buffer, length);

                System.UInt32 mimetype;
                FindMimeFromData(0, null, buffer, 256, null, 0, out mimetype, 0);
```

```
                System.IntPtr mimeTypePtr = new IntPtr(mimetype);
                result = Marshal.PtrToStringUni(mimeTypePtr);
                Marshal.FreeCoTaskMem(mimeTypePtr);
            }
            catch (Exception ex)
            {
                //Log.WriteError(MethodInfo.GetCurrentMethod(), ex);
            }


            return result;
        }

    }
}
```

تنها نکته قابل توجه در این کد این‌است که در بدنه‌ی متد GetMimeType اگر طول آرایه از 256 بایت بیشتر شود، فقط از 256 بایت ابتدایی آن استفاده میکنیم. در غیر اینصورت کل بایت‌های آرایه، برای چک کردن محتوای فایل به کار گرفته میشوند. برای افزایش کارآیی، بهتر است هنگام فراخوانی این تابع، فقط 256 بایت را به آن ارسال کنید. بخصوص اگر حجم فایلی که باید چک شود زیاد باشد.

نحوه‌ی استفاده از کلاس فوق هم بسیار ساده است. برای مثال هنگام آپلود فایل در MVC می‌توانیم از آن استفاده کنیم:

```
[HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Upload(HttpPostedFileBase file)
        {
            //ابتدا با مکانیزم مورد نظر خود پسوند فایل را چک میکنیم اگر پسوند معتبری بود بعد محتوای فایل را
چک میکنیم

            var data = new byte[256];
            file.InputStream.Read(data, 0, 256);
            var detector = new Parsnet.Core.MimeTypeDetector();
            var mimeType = detector.GetMimeType(data);
            if (CheckWhiteList(mimeType) == true)
            {
                file.SaveAs("yourpath");
            }
            else
            {
                ModelState.AddModelError("InvalidFileContent", "فایل بارگزاری شده مورد پذیرش نیست.");
            }
            return View();
        }
```

همیشه از یک مکانیزم دیگر برای اعتبارسنجی پسوند فایل استفاده کنید. برای اینکار می‌توان با تلفیق روش فوق و این روش یک کنترل اعتبارسنجی خوب ایجاد کرد که در مطالب بعدی حتما به آن خواهم پرداخت.

## نظرات خوانندگان

نویسنده: احمد رجبی
تاریخ: ۱۳۹۴/۰۴/۲۴ ۱۱:۱۳

نکته: لیست MIME Type هایی که توسط این تابع پشتیبانی می‌شوند را میتوانید در اینجا ببینید.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۴/۰۴/۲۴ ۱۱:۲۸

### یک نکته‌ی تکمیلی

در ASP.NET **4.5** به صورت توکار فضای نام System.Web.MimeMapping و متد GetMimeMapping ، اضافه شده‌اند (البته به نظر فقط بر اساس پسوند فایل کار می‌کند).

نویسنده: غلامرضا ربال
تاریخ: ۱۳۹۴/۰۴/۲۴ ۱۶:۳۰

لیست MimeTypeها به همراه Extension

```
private static readonly Dictionary<string, string> SpecialMIMETypes = new Dictionary<string, string>
        {
            {"ai", "application/postscript"},
            {"aif", "audio/x-aiff"},
            {"aifc", "audio/x-aiff"},
            {"aiff", "audio/x-aiff"},
            {"asc", "text/plain"},
            {"atom", "application/atom+xml"},
            {"au", "audio/basic"},
            {"avi", "video/x-msvideo"},
            {"bcpio", "application/x-bcpio"},
            {"bin", "application/octet-stream"},
            {"bmp", "image/bmp"},
            {"cdf", "application/x-netcdf"},
            {"cgm", "image/cgm"},
            {"class", "application/octet-stream"},
            {"cpio", "application/x-cpio"},
            {"cpt", "application/mac-compactpro"},
            {"csh", "application/x-csh"},
            {"css", "text/css"},
            {"dcr", "application/x-director"},
            {"dif", "video/x-dv"},
            {"dir", "application/x-director"},
            {"djv", "image/vnd.djvu"},
            {"djvu", "image/vnd.djvu"},
            {"dll", "application/octet-stream"},
            {"dmg", "application/octet-stream"},
            {"dms", "application/octet-stream"},
            {"doc", "application/msword"},
            {"docx","application/vnd.openxmlformats-officedocument.wordprocessingml.document"},
            {"dotx", "application/vnd.openxmlformats-officedocument.wordprocessingml.template"},
            {"docm","application/vnd.ms-word.document.macroEnabled.12"},
            {"dotm","application/vnd.ms-word.template.macroEnabled.12"},
            {"dtd", "application/xml-dtd"},
            {"dv", "video/x-dv"},
            {"dvi", "application/x-dvi"},
            {"dxr", "application/x-director"},
            {"eps", "application/postscript"},
            {"etx", "text/x-setext"},
            {"exe", "application/octet-stream"},
            {"ez", "application/andrew-inset"},
            {"gif", "image/gif"},
            {"gram", "application/srgs"},
            {"grxml", "application/srgs+xml"},
            {"gtar", "application/x-gtar"},
            {"hdf", "application/x-hdf"},
            {"hqx", "application/mac-binhex40"},
            {"htc", "text/x-component"},
            {"htm", "text/html"},
            {"html", "text/html"},
            {"ice", "x-conference/x-cooltalk"},
```

```
            {"ico", "image/x-icon"},
            {"ics", "text/calendar"},
            {"ief", "image/ief"},
            {"ifb", "text/calendar"},
            {"iges", "model/iges"},
            {"igs", "model/iges"},
            {"jnlp", "application/x-java-jnlp-file"},
            {"jp2", "image/jp2"},
            {"jpe", "image/jpeg"},
            {"jpeg", "image/jpeg"},
            {"jpg", "image/jpeg"},
            {"js", "application/x-javascript"},
            {"kar", "audio/midi"},
            {"latex", "application/x-latex"},
            {"lha", "application/octet-stream"},
            {"lzh", "application/octet-stream"},
            {"m3u", "audio/x-mpegurl"},
            {"m4a", "audio/mp4a-latm"},
            {"m4b", "audio/mp4a-latm"},
            {"m4p", "audio/mp4a-latm"},
            {"m4u", "video/vnd.mpegurl"},
            {"m4v", "video/x-m4v"},
            {"mac", "image/x-macpaint"},
            {"man", "application/x-troff-man"},
            {"mathml", "application/mathml+xml"},
            {"me", "application/x-troff-me"},
            {"mesh", "model/mesh"},
            {"mid", "audio/midi"},
            {"midi", "audio/midi"},
            {"mif", "application/vnd.mif"},
            {"mov", "video/quicktime"},
            {"movie", "video/x-sgi-movie"},
            {"mp2", "audio/mpeg"},
            {"mp3", "audio/mpeg"},
            {"mp4", "video/mp4"},
            {"mpe", "video/mpeg"},
            {"mpeg", "video/mpeg"},
            {"mpg", "video/mpeg"},
            {"mpga", "audio/mpeg"},
            {"ms", "application/x-troff-ms"},
            {"msh", "model/mesh"},
            {"mxu", "video/vnd.mpegurl"},
            {"nc", "application/x-netcdf"},
            {"oda", "application/oda"},
            {"ogg", "application/ogg"},
            {"pbm", "image/x-portable-bitmap"},
            {"pct", "image/pict"},
            {"pdb", "chemical/x-pdb"},
            {"pdf", "application/pdf"},
            {"pgm", "image/x-portable-graymap"},
            {"pgn", "application/x-chess-pgn"},
            {"pic", "image/pict"},
            {"pict", "image/pict"},
            {"png", "image/png"},
            {"pnm", "image/x-portable-anymap"},
            {"pnt", "image/x-macpaint"},
            {"pntg", "image/x-macpaint"},
            {"ppm", "image/x-portable-pixmap"},
            {"ppt", "application/vnd.ms-powerpoint"},
            {"pptx","application/vnd.openxmlformats-officedocument.presentationml.presentation"},
            {"potx","application/vnd.openxmlformats-officedocument.presentationml.template"},
            {"ppsx","application/vnd.openxmlformats-officedocument.presentationml.slideshow"},
            {"ppam","application/vnd.ms-powerpoint.addin.macroEnabled.12"},
            {"pptm","application/vnd.ms-powerpoint.presentation.macroEnabled.12"},
            {"potm","application/vnd.ms-powerpoint.template.macroEnabled.12"},
            {"ppsm","application/vnd.ms-powerpoint.slideshow.macroEnabled.12"},
            {"ps", "application/postscript"},
            {"qt", "video/quicktime"},
            {"qti", "image/x-quicktime"},
            {"qtif", "image/x-quicktime"},
            {"ra", "audio/x-pn-realaudio"},
            {"ram", "audio/x-pn-realaudio"},
            {"ras", "image/x-cmu-raster"},
            {"rdf", "application/rdf+xml"},
            {"rgb", "image/x-rgb"},
            {"rm", "application/vnd.rn-realmedia"},
            {"roff", "application/x-troff"},
            {"rtf", "text/rtf"},
            {"rtx", "text/richtext"},
            {"sgm", "text/sgml"},
            {"sgml", "text/sgml"},
```

```
            {"sh", "application/x-sh"},
            {"shar", "application/x-shar"},
            {"silo", "model/mesh"},
            {"sit", "application/x-stuffit"},
            {"skd", "application/x-koan"},
            {"skm", "application/x-koan"},
            {"skp", "application/x-koan"},
            {"skt", "application/x-koan"},
            {"smi", "application/smil"},
            {"smil", "application/smil"},
            {"snd", "audio/basic"},
            {"so", "application/octet-stream"},
            {"spl", "application/x-futuresplash"},
            {"src", "application/x-wais-source"},
            {"sv4cpio", "application/x-sv4cpio"},
            {"sv4crc", "application/x-sv4crc"},
            {"svg", "image/svg+xml"},
            {"swf", "application/x-shockwave-flash"},
            {"t", "application/x-troff"},
            {"tar", "application/x-tar"},
            {"tcl", "application/x-tcl"},
            {"tex", "application/x-tex"},
            {"texi", "application/x-texinfo"},
            {"texinfo", "application/x-texinfo"},
            {"tif", "image/tiff"},
            {"tiff", "image/tiff"},
            {"tr", "application/x-troff"},
            {"tsv", "text/tab-separated-values"},
            {"txt", "text/plain"},
            {"ustar", "application/x-ustar"},
            {"vcd", "application/x-cdlink"},
            {"vrml", "model/vrml"},
            {"vxml", "application/voicexml+xml"},
            {"wav", "audio/x-wav"},
            {"wbmp", "image/vnd.wap.wbmp"},
            {"wbmxl", "application/vnd.wap.wbxml"},
            {"wml", "text/vnd.wap.wml"},
            {"wmlc", "application/vnd.wap.wmlc"},
            {"wmls", "text/vnd.wap.wmlscript"},
            {"wmlsc", "application/vnd.wap.wmlscriptc"},
            {"wrl", "model/vrml"},
            {"xbm", "image/x-xbitmap"},
            {"xht", "application/xhtml+xml"},
            {"xhtml", "application/xhtml+xml"},
            {"xls", "application/vnd.ms-excel"},
            {"xml", "application/xml"},
            {"xpm", "image/x-xpixmap"},
            {"xsl", "application/xml"},
            {"xlsx","application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"},
            {"xltx","application/vnd.openxmlformats-officedocument.spreadsheetml.template"},
            {"xlsm","application/vnd.ms-excel.sheet.macroEnabled.12"},
            {"xltm","application/vnd.ms-excel.template.macroEnabled.12"},
            {"xlam","application/vnd.ms-excel.addin.macroEnabled.12"},
            {"xlsb","application/vnd.ms-excel.sheet.binary.macroEnabled.12"},
            {"xslt", "application/xslt+xml"},
            {"xul", "application/vnd.mozilla.xul+xml"},
            {"xwd", "image/x-xwindowdump"},
            {"xyz", "chemical/x-xyz"},
            {"zip", "application/zip"}
        };
```