

آموزش ساخت تصاویر امنیتی

تصویر امنیتی چیست ؟

احتمالا شما در هنگام دانلود فایل یا هنگام ثبت نام در برخی از سایت ها درگیر پر کردن فرم های کوچک و بزرگی شده اید. در این فرم ها احتمالا با یک تصویری که در آن چند کاراکتر و یا عدد به شکلی سوال انگیز در کنار هم قرار گرفته است برخورد کرده اید. در کنار این تصاویر از شما خواسته شده است تا کاراکترهایی را که در تصویر مشاهده می کنید را در یک TextBox وارد کنید. اما این تصاویر به چه دردی می خورند ؟

کاربرد تصاویر امنیتی ؟

تصور کنید در سایت شما بخش نظرسنجی وجود دارد و در آن هر فرد مراجعه کننده به سایت (Anonymous) می تواند نظر خود را در آن وارد کند. شما چند راه برای این کار دارید یا تعداد نظراتی را که هر فرد می تواند ثبت کند را محدود سازید و یا اینکه هیچ محدودیتی وجود نداشته باشد. در صورتی که قصد محدود سازی کاربر را داشته باشید احتمالا به سراغ متغیر Session می روید و با الگوریتم خاص خود این مهم را انجام می دهید. مثلا کاربر را مجبور می کنید تا فقط در بازه های زمانی 10 دقیقه ای امکان ثبت را داشته باشد ! سناریو ای را در نظر بگیرید که چند کاربر که از یک PC استفاده می کنند و یا کاربری قصد ارسال دو نظر را دارد، احتمالا با خواندن سناریوهائی از این دست به شما عذاب وجدان دست خواهد داد. حال فرض کنیم شما راه حل دوم را انتخاب کنید و هیچ محدودیتی برای ورود اطلاعات در نظر نگیرید. سناریوای را در نظر بگیرید که در آن هکری با نوشتن یک برنامه ساده سعی می کند به سرعت در وب سایت شما مدام نظر ثبت کند. اگر هر رکورد جدول نظر شما 2092 بایت اشغال کند (2048 بایت برای نظر) و در هر 3 ثانیه (بهترین حالت برای شما و بدترین حالت برای هکر) بتواند یک رکورد ثبت کند. اگر بانک اطلاعاتی سایت شما 10mb فضای آزاد داشته باشد در عرض 4 ساعت بانک اطلاعاتی شما پر از اطلاعات هرزه می شود. احتمالا در بهترین حالت بین 3 تا 4 روز بانک شما پر است و تا آن زمان هیچ چیزی نمی تواند در بانک اطلاعاتی شما درج شود و چندین ساعت از وقت شما صرف پاک کردن اطلاعات هرزه می شود.

نحوه ساخت تصاویر امنیتی

روشی که در این مقاله برای ساخت تصاویر امنیتی استفاده شده است بدین شرح است که از طریق ایجاد یک لینک به **HttpHandler** تصویر امنیتی را نمایش می دهیم.



همان طور که در تصویر مشاهده می شود از طریق پسوند **ashx** به **HttpHandler** تولید کننده تصویر امنیتی متصل شده و تصویر امنیتی را دریافت می کنیم. این تصویر امنیتی بر اساس مشخصات مورد نظر ما ایجاد می شوند. این مشخصات در ادامه ی پسوند **ashx** به عنوان **Query** و به صورت کاملا رمزنگاری شده تولید شده اند.

کلاس SecurityImage

این کلاس وظیفه تولید آدرس یک تصویر امنیتی را دارا می باشد. در حقیقت ما فقط از این کلاس استفاده می کنیم و خود را از درگیری با سایر مسائل رها می سازیم. به مثال زیر توجه فرمائید :

```
if (!this.IsPostBack)
{
    SecurityImage simg = new SecurityImage(SecurityLevel.Low, 300, 50);
    Session["simg_code"] = simg.Code;
    simg.Generate();
    iLow.ImageUrl = simg.SecurityImageUri;
}
```

پارامترهای ورودی سازنده کلاس Security Image سطح امنیتی کد و اندازه تصویر خروجی را تعیین می کنند. متد Generate با هر بار فراخوانی یک کد جدید تولید می کند و به طبع آن SecurityImageUri و Code نیز تغییر می کند.

این کد را در هنگام Load شدن صفحه به کار گرفته و کد امنیتی ایجاد شده را در Session نگه داری می کنیم. بعد از اینکه کاربر فرم مورد نظر را پر کرده و پست می کند. می توان مقدار تایپ شده توسط کاربر را با اصل کد امنیتی موجود در Session چک کرده و در صورت صحت آن اطلاعات را ثبت کنیم.

مهمترین متد کلاس SecurityImage به شرح زیر است :

```
private string GenerateCode()
{
    Random r = new Random((int)DateTime.Now.Millisecond);
    int maxNumber = 0;
    int numberCodeQuantity = 0, alphabeticCodeQuantity = 0;
    string code = String.Empty;

    maxNumber = r.Next(6, 7);
    numberCodeQuantity = 1;
    if (SecurityLevel == SecurityLevel.Low)
        numberCodeQuantity = 5;
    else
        alphabeticCodeQuantity = maxNumber - numberCodeQuantity;

    for (int i = 1; i <= numberCodeQuantity; ++i)
    {
        code += ((char)r.Next(48, 57)).ToString();
    }
    for (int i = 1; i <= alphabeticCodeQuantity; ++i)
    {
        code += ((char)r.Next(65, 90)).ToString();
    }
    return code;
}
```

این متد عمل تولید کد امنیتی متناسب با سطح امنیتی تعیین شده انجام می دهد. پس از تولید کد باید اطلاعات شامل اندازه تصویر، سطح امنیتی تصویر و کد مورد نظر کد شده و به صورت یک Query درآیند.

```

public string Generate()
{
    byte[] IV = new byte[8] { 120, 34, 63, 127, 93, 240, 23, 232 };
    string cryptoKey = "GalaxyRoad2004@yahoo.com";

    _code = GenerateCode();
    byte[] codebytes = System.Text.Encoding.ASCII.GetBytes(_code);
    TripleDESCryptoServiceProvider tripleDES = new
TripleDESCryptoServiceProvider();
    MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();
    tripleDES.Key =
md5.ComputeHash(System.Text.Encoding.ASCII.GetBytes(cryptoKey));
    tripleDES.IV = IV;
    byte[] codeBuffer =
tripleDES.CreateEncryptor().TransformFinalBlock(codebytes, 0,
codebytes.Length);
    string secretCode = Convert.ToBase64String(codeBuffer, 0,
codeBuffer.Length);
    secretCode = HttpContext.Current.Server.UrlEncode(secretCode);
    string query = secretCode + "&" + Width.ToString() + "&" +
Height.ToString() + "&" + SecurityLevel.ToString();
    byte[] queryBytes = System.Text.Encoding.ASCII.GetBytes(query);
    Random r = new Random((int)DateTime.Now.Millisecond);
    _simgUri = r.Next().ToString() + ".ashx?" +
HttpContext.Current.Server.UrlEncode(Convert.ToBase64String(queryBytes,
0, queryBytes.Length));
    return _simgUri;
}

```

متد فوق این کار را برای ما انجام می دهد و اطلاعات مورد نظر را به صورت یک **Query**، رمز می کند و آن را در قالب یک **Uri** فایل تصویری **jpg** باز می گرداند.

در سمت سرور ما کلاس **SecurityImageHandler** را داریم که وظیفه ی **Handler** کردن درخواست های تصاویر امنیتی را بر عهده دارد. مهمترین متد این کلاس به شرح زیر است :

```

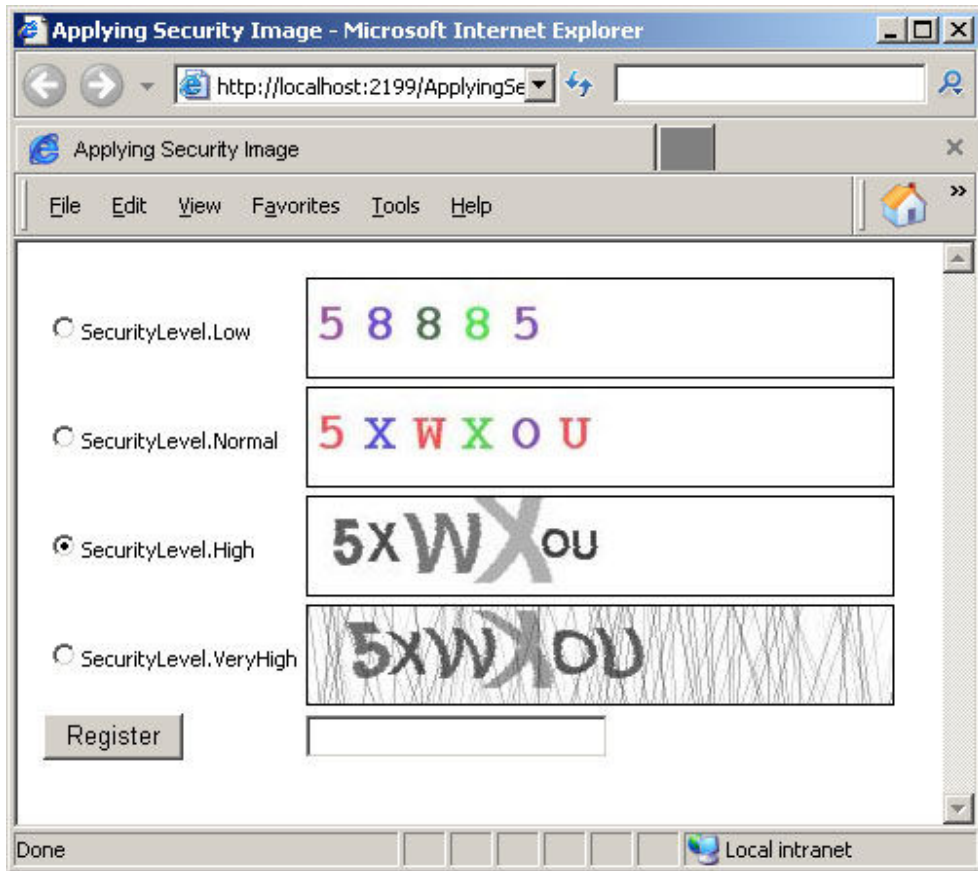
public void ProcessRequest(HttpContext context)
{
    string secretString;
    string code;
    try
    {
        secretString = context.Request.Url.Query.Substring(1);
        code = Decode(secretString);
    }
    catch (Exception)
    {
        return;
    }
    Size size = new Size(300, 50);
    Bitmap bmp = new Bitmap(size.Width, size.Height);
    Bitmap result = new Bitmap(imageWidth, imageHeight);
    Graphics g = Graphics.FromImage(bmp);
    ...

```

```
result.Save(context.Response.OutputStream,  
System.Drawing.Imaging.ImageFormat.Jpeg);  
}
```

برای اختصار از آوردن کدهای تولید تصویر خودداری کردم. قبل از تولید تصویر کد امنیتی باید **Query** ارسال شده **Decode** گردد که این عمل توسط متد زیر صورت می گیرد :

```
private string Decode(string secretCode)  
{  
    byte[] queryBytes =  
Convert.FromBase64String(HttpContext.Current.Server.UrlDecode(secretCode));  
    string query = System.Text.Encoding.ASCII.GetString(queryBytes, 0,  
queryBytes.Length);  
    string[] parameters = query.Split('&');  
    int imageWidth = int.Parse(parameters[1]);  
    int imageHeight = int.Parse(parameters[2]);  
    int ParseSecurityLevel(parameters[3]);  
    byte[] codeBuffer =  
Convert.FromBase64String(HttpContext.Current.Server.UrlDecode(parameters[0]));  
    byte[] IV = new byte[8] { 120, 34, 63, 127, 93, 240, 23, 232 };  
    string cryptoKey = "GalaxyRoad2004@yahoo.com";  
    TripleDESCryptoServiceProvider tripleDES = new  
TripleDESCryptoServiceProvider();  
    MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider();  
  
    tripleDES.Key =  
md5.ComputeHash(System.Text.Encoding.ASCII.GetBytes(cryptoKey));  
    tripleDES.IV = IV;  
    byte[] codebytes =  
tripleDES.CreateDecryptor().TransformFinalBlock(codeBuffer, 0,  
codeBuffer.Length);  
    return System.Text.Encoding.ASCII.GetString(codebytes, 0,  
codebytes.Length);  
}
```



نحوه استفاده از SecurityImage تهیه شده در این بخش :

کد زیر را در PageLoad قرار دهید :

```

if (!this.IsPostBack)
{
    SecurityImage simg = new SecurityImage(SecurityLevel.Low, 300, 50);
    Session["simg_code"] = simg.Code;
    simg.Generate();
    iLow.ImageUrl = simg.SecurityImageUri;
}

```

کد زیر را در هم برای چک کردن اینکه آیا کاربر کد را صحیح وارد کرده است یا خیر استفاده نمایید :

```

protected void Check_Click(object sender, EventArgs e)
{
    bool correct = false;
    if (Session["simg_code"]
    ].ToString().ToLower().Equals(tbCode.Text.ToLower()))
        correct = true;

    if (correct) // Success

```

```
        Do some thing
    Else // Failed
        Response.Write("Failed.");
}
```

در فایل `web.config` مابین تگ های `<system.web>` کد زیر را درج نمائید :

```
<httpHandlers>
  <add path="*.ashx" verb="*"
type="Farahy.Security.SecurityImageHttpHandler, SecurityImage"/>
</httpHandlers>
```

و در نهایت مطمئن شوید که فایل `securityimage.dll` را در شاخه `bin` وب اپلیکیشن کپی کرده اید.

موفق باشید

سید محمد رضا فراحی (manager)