

تصویر امنیتی و یا کپچا برای تشخیص و احراز انسان بودن استفاده کننده استفاده میشود و بصورت تصویری که استخراج نوشته‌های درون آن برای روبات‌ها بسیار سخت و یا نشدنی است ایجاد میشود و دارای انواع و اقسام متفاوتی است. در این میان برای استفاده از این امکان نمونه‌هایی در زبانهای مختلف تهیه شده که بسته به سلیقه و نیاز مورد استفاده قرار گرفته شده است. در این مقاله قصد داریم با بررسی تصویر امنیتی که در وبلاگ کنونی استفاده شده آنرا تا حدودی بازسازی کنیم.

تصویر آشنای کاربران سایت برای ورود به قسمت مدیریت که در صفحه مورد نظر از تصویر امنیتی استفاده شده است:



با توجه به آدرس لینک تصویر مشخص است که برای تولید این تصویر از هندلر استفاده شده و با توجه به پارامترهایی که به آن داده شده تصویر مورد نظر را ایجاد میکند و با فرمت مشخصی بر میگردداند:

```
&foreColor=%231B0172
&fontSize=12
&fontName=Tahoma
```

پارامترهای پاس داده شده به ترتیب ذیل برای اهدافی قرار داده شده اند:

نام	مورد استفاده	مقدار
text	متنی که بصورت کد شده حاوی متن مورد استفاده تصویر امنیتی است	H2yL5iOXIuu0dsBvu%2F405AnXkeacis3dMQ%2FHXA8h4A%2BjwDM0f7%2FRZMHvBG5pXYJ
foreColor	رنگ مربوط به نوشته‌های تصویر	%231B0172
fontSize	سایز فونت	12
fontName	نام فونت	Tahoma

بنظر اهم پارامتر مورد نیاز همان متن است که بصورت کد شده در بالا به آن اشاره شد. برای این منظور باید کلاسی تهیه شود که حاوی متدهای:

1- انتخاب یه عدد تصادفی از بین دامنه ای که به آن داده میشود.

برای مثال یه عد از بین 100 تا 9999 که بصورت تصادفی انتخاب میشود

2- تبدیل به معادل حرفی آن

برای مثال از عدد "7062" به حروف آن یعنی "هفت هزار و شصت و دو" استخراج شود

3- کد کردن حرف تولید شده

حرف "هفت هزار و شصت و دو" را کد کرده و بصورت متنی شبیه

" H2yL5iOXIuu0dsBvu%2F405AnXkeacis3dMQ%2FHXA8h4A%2BjwDM0f7%2FRZMHvBG5pXYJ"

تبدیل کند

4- دیکد کردن و استخراج

البته این مورد با توجه به استفاده ای که ما داریم نیازی به پیاده سازی نیست

در کنار تصویر امنیتی از یک فیلد مخفی نیز برای نگهداری مقدار کد شده برای مقایسه با ورودی کاربر استفاده شده که در قسمت بعد ضمن ایجاد نمونه کاربردی بیشتر با قسمتهای مختلف آن آشنا میشویم.

نظرات خوانندگان

نویسنده: ahmadalli

تاریخ: ۲۱:۳۹ ۱۳۹۱/۰۸/۳۰

از نظر تئوری هر کدی قابل دی‌کد شدن هست. بهتر نیست برای کد اصلی از hash استفاده کنیم و hash کدی که کاربر وارد می‌کنه رو با hash اصلی مقایسه کنیم؟

نویسنده: مهدی پایروند

تاریخ: ۲۳:۴۴ ۱۳۹۱/۰۸/۳۰

با توجه به [قسمت دوم](#) از قسمت دیکد کردن استفاده نشده ولی چون از این نوع رمزنگرای استفاده شده در مواقعی ممکن است از دیکد کردن نیز استفاده کرد ولی به هرحال این رمزگذاری رو میشه بدخواه تغییر داد.


```

        {
            s = s + Dahyek[d12 - 10];
        }
        else
        {
            var d2 = d12 / 10;
            if (d2 != 0)
                s = s + Dahgan[d2] + " و ";
            var d1 = d12 % 10;
            if (d1 != 0)
                s = s + Yakan[d1] + " و ";
            s = s.Substring(0, s.Length - 3);
        }
        return s;
    }

    public static string ConvertIntNumberToFarsiAlphabetic(string snum)
    {
        var stotal = "";
        if (snum == "0") return Yakan[0];

        snum = snum.PadLeft(((snum.Length - 1) / 3 + 1) * 3, '0');
        var l = snum.Length / 3 - 1;
        for (var i = 0; i <= l; i++)
        {
            var b = int.Parse(snum.Substring(i * 3, 3));
            if (b != 0)
                stotal = stotal + Getnum3(b) + " " + Basex[l - i] + " و ";
        }
        stotal = stotal.Substring(0, stotal.Length - 3);
        return stotal;
    }
}

```

و برای کد کردن و دیکد کردن یعنی موارد سوم و چهارم مقاله قبلی، متن بدست آمده را که بعنوان قسمتی از آدرس تصویر در ادامه آدرس هندلر معرفی شده می‌آید تبدیل به یک string بی معنی برای بازدیدکننده میکند:

```

using System;
using System.IO;
using System.Security.Cryptography;
using System.Text;

namespace PersianCaptchaHandler
{
    public class Encryptor
    {
        #region constraints
        private static string Password { get { return "Mehdi"; } }
        private static string Salt { get { return "Payervand"; } }
        #endregion

        public static string Encrypt(string clearText)
        {
            // Turn text to bytes
            var clearBytes = Encoding.Unicode.GetBytes(clearText);

            var pdb = new PasswordDeriveBytes(Password, Encoding.Unicode.GetBytes(Salt));

            var ms = new MemoryStream();
            var alg = Rijndael.Create();

            alg.Key = pdb.GetBytes(32);
            alg.IV = pdb.GetBytes(16);

            var cs = new CryptoStream(ms, alg.CreateEncryptor(), CryptoStreamMode.Write);

            cs.Write(clearBytes, 0, clearBytes.Length);
            cs.Close();

            var encryptedData = ms.ToArray();

            return Convert.ToBase64String(encryptedData);
        }

        public static string Decrypt(string cipherText)
        {

```

```
// Convert text to byte
var cipherBytes = Convert.FromBase64String(cipherText);

var pdb = new PasswordDeriveBytes(Password, Encoding.Unicode.GetBytes(Salt));

var ms = new MemoryStream();
var alg = Rijndael.Create();

alg.Key = pdb.GetBytes(32);
alg.IV = pdb.GetBytes(16);

var cs = new CryptoStream(ms, alg.CreateDecryptor(), CryptoStreamMode.Write);

cs.Write(cipherBytes, 0, cipherBytes.Length);
cs.Close();

var decryptedData = ms.ToArray();

return Encoding.Unicode.GetString(decryptedData);
}
}
```

و نیز برای اعتبار سنجی عدد دریافتی از کاربر میتوان از عبارت با قاعده زیر استفاده کرد:

```
// Utils.cs
using System.Text.RegularExpressions;

namespace PersianCaptchaHandler
{
    public class Utils
    {
        private static readonly Regex NumberMatch = new Regex(@"^([0-9]*|\d*\.\d{1}?\d*)$",
        RegexOptions.Compiled);
        public static bool IsNumber(string number2Match)
        {
            return NumberMatch.IsMatch(number2Match);
        }
    }
}
```

برای استفاده نیز کافیسست که هندلر مربوط به این کتابخانه را بطریق زیر در وب کانفیگ رجیستر کرد:

```
<add verb="GET" path="/captcha/" type="PersianCaptchaHandler.CaptchaHandler, PersianCaptchaHandler,
Version=1.0.0.0, Culture=neutral" />
```

و در صفحه مورد نظرتان بطریق زیر میتوان از یک تگ تصویر برای نمایش تصویر تولیدی و از یک فیلد مخفی برای نگهداری مقدار کد شده معادل عدد تولیدی که در هنگام مقایسه با عدد ورودی کاربر مورد نیاز است استفاده شود:

```
<!-- ASPX -->
<dl>
    <dt>تصویر امنیتی</dt>
    <dd>
        <asp:Image ID="imgCaptchaText" runat="server" AlternateText="CaptchaImage" />
        <asp:HiddenField ID="hfCaptchaText" runat="server" />
        <asp:ImageButton ID="btnRefreshCaptcha" runat="server" ImageUrl="/img/refresh.png"
            OnClick="btnRefreshCaptcha_Click" />
        <br />
        <asp:TextBox ID="txtCaptcha" runat="server" AutoCompleteType="Disabled"></asp:TextBox>
    </dd>
    <dd>
        <asp:Button ID="btnSubmit" runat="server" Text="ثبت" OnClick="btnSubmit_Click" />
    </dd>
</dl>
<asp:Label ID="lblMessage" runat="server"></asp:Label>
```

در زمان لود صفحه، تصویر امنیتی مقدار دهی میشود و در زمان ورود عدد توسط کاربر با توجه به اینکه کاربر حتما باید عدد وارد کند با عبارت با قاعده این اعتبار سنجی انجام میشود:

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!Page.IsPostBack)
        SetCaptcha();
}

private void SetCaptcha()
{
    lblMessage.Text =
    txtCaptcha.Text = string.Empty;

    var newNumber =
        RandomGenerator.Next(100, 999)
        ;

    var farsiAlphabetic =
    NumberToString.ConvertIntNumberToFarsiAlphabetic(newNumber.ToString());

    hfCaptchaText.Value =
        HttpUtility
        .UrlEncode(
            Encryptor.Encrypt(
                farsiAlphabetic
            )
        );

    txtCaptcha.Text = string.Empty;
    imgCaptchaText.ImageUrl =
        "/captcha/?text=" + hfCaptchaText.Value;
}
```

و بعد از ورود عدد از سمت کاربر از متد تبدیل به حروف استفاده کرده و این مقدار تولیدی با مقدار فیلد مخفی مقایسه میشود:

```
private string GetCaptcha()
{
    var farsiAlphabetic = NumberToString.ConvertIntNumberToFarsiAlphabetic(txtCaptcha.Text);

    var encryptedString =
        HttpUtility
        .UrlEncode(
            Encryptor.Encrypt(
                farsiAlphabetic
            )
        );

    return encryptedString;
}

private bool ValidateUserInputForLogin()
{
    if (!Utils.IsNumber(txtCaptcha.Text))
    {
        lblMessage.Text = "تصویر امنیتی را بطور صحیح وارد نکرده اید";
        return false;
    }

    var strGetCaptcha =
        GetCaptcha();

    var strDecodedVAlue =
        hfCaptchaText.Value;

    if (strDecodedVAlue != strGetCaptcha)
    {
        lblMessage.Text = "کلمه امنیتی اشتباه است";
        SetCaptcha();
        return false;
    }
    return true;
}
```

```
protected void btnSubmit_Click(object sender, EventArgs e)
{
    if (!ValidateUserInputForLogin()) return;
    lblMessage.Text = "کلمه امنیتی درست است";
}

protected void btnRefreshCaptcha_Click(object sender, ImageClickEventArgs e)
{
    SetCaptcha();
}
```

در آخر این [پروژه در کدپلکسی](#) قرار داده شده، و مشتاق نظرات و پیشنهادات شما هستم و نیز [نمونه مثال بالا](#) ضمیمه شده است

نظرات خوانندگان

نویسنده: احمد احمدی
تاریخ: ۱۳:۲۸ ۱۳۹۱/۰۹/۰۱

تشکر از مطلب مفیدتون.
جناب نصیری کلاسی برای [تبدیل عدد به حروف](#) نوشته اند که استفاده از اون کلاس در این پروژه ، زیبایی کار را دو چندان می کند!

نویسنده: مهدی پایروند
تاریخ: ۱۳:۳۸ ۱۳۹۱/۰۹/۰۱

پروژه در [کدپلکس](#)

پروژه در [سایت](#)

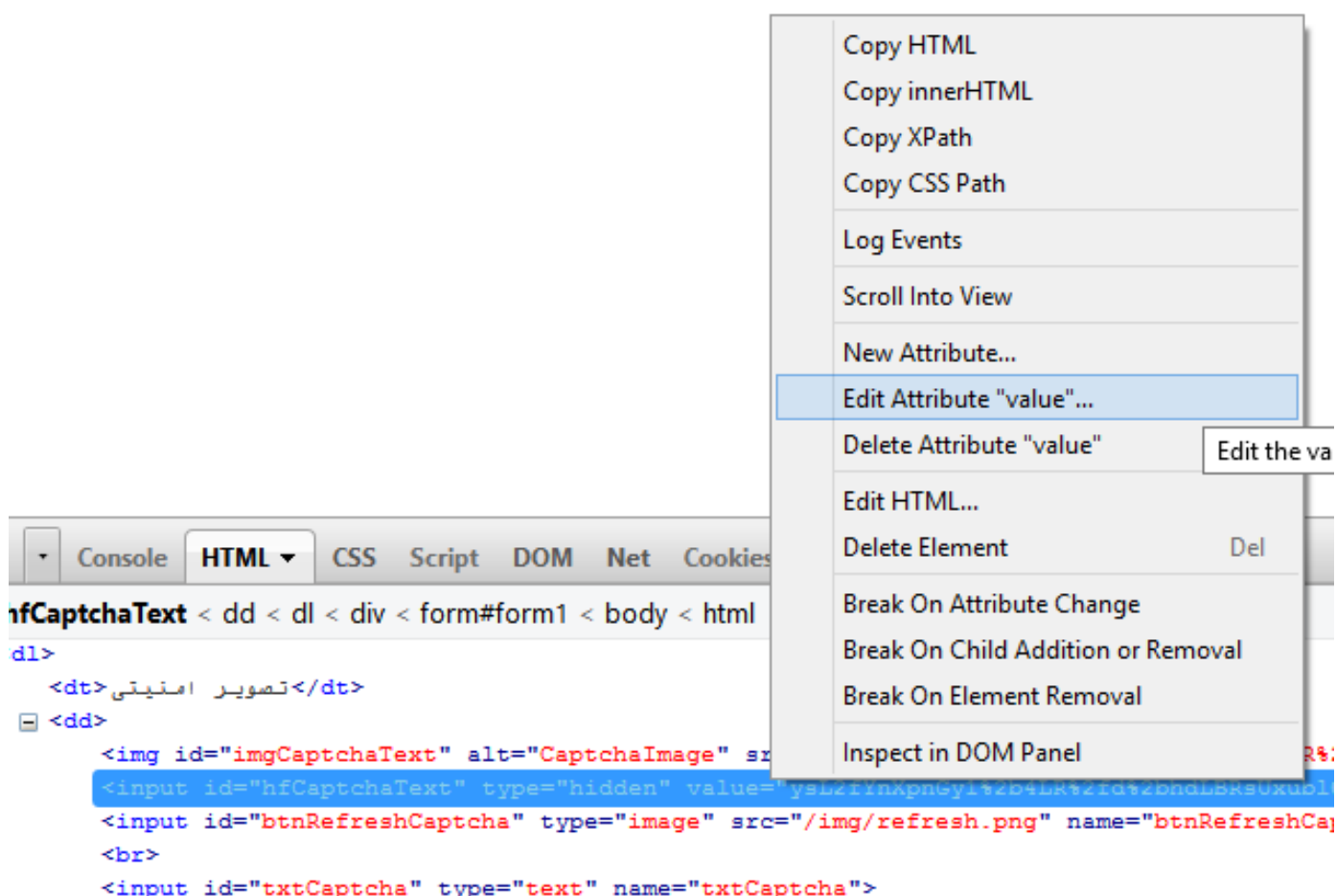
نویسنده: مهدی پایروند
تاریخ: ۱۳:۴۰ ۱۳۹۱/۰۹/۰۱

بسیار ممنون، با اجازه مهندس نصیری این کار انجام میشه.
لطفا درخواست رو در [قسمت مربوط به پروژه در سایت](#) مطرح کنید تا یک تاریخچه مطلوبی برای پروژه ایجاد بشه.

نویسنده: رحمت اله رضایی
تاریخ: ۱۱:۲۹ ۱۳۹۱/۰۹/۰۶

مشکلی که این روش دارد این است که با یک بار بارگذاری صفحه، خواندن عکس توسط انسان و یادداشت متن رمز گذاری شده می توان مقدار اولیه عکس را هر بار با ویرایش متن رمزگذاری شده به سرور ارسال کرد و تایید گرفت! مشکلی که سایت فعلی هم دارد.

برای حل این مشکل چه پیشنهادی دارید؟



نویسنده: مهدی پایروند
تاریخ: ۱۴:۱۹ ۱۳۹۱/۰۹/۰۶

ممنون از نظرتون،
موضوع مورد نظر، قسمتی است که متن تصویر برای این کپچا در آن نگهداری میشود،
1. در صورتی که بخواهیم از تواناییهای سمت سرور کمک گرفت: بنظرم برای اعتبار سنجی میتوان برای کاربران در بانک یک کد یونیک به ازای اولین هیت کاربر در سایت ایجاد کرد و این کد را بصورت رمزنگاری شده در کلاینت(کوکی) نگهداشت و مواردی از این دست را با این کد اعتبارسنجی کرد.
2. در صورتی که بخواهیم در سمت کلاینت همه چیز هندل شود : مطمئنا بهتر است که این فیلد مخفی در کنار باکس ورودی قرار داده نشود و برای نگهداری این فیلد از جاوا اسکریپت یا کوکی و از هردو کمک گرفت به این صورت که این مقدار کد شده بصورت مقداری غیر از این و با نامی بدون ربط در متغیری که از سمت سرور مقدار دهی میشود و فقط در سرور خوانده میشود تعویض کرد.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۷:۵ ۱۳۹۱/۰۹/۰۷

لطفا راهنمایی کنید چطور پروژه رو دانلود کنم؟

نویسنده: مهدی پایروند
تاریخ: ۱۷:۵۶ ۱۳۹۱/۰۹/۰۷

برای دمو از این لینک میتونید استفاده کنید +
و همچنین در کدپلکس persiacaptchahandler.codeplex.com
و در همین سایت هم PersianCaptchaHandler

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۲:۴۳

بهترین راه استفاده از Expiration date هستش، به این صورت که به متن رمزنگاری شده یک تاریخ انقضای 3 دقیقه ای داد، بدون استفاده از دیتابیس و کوکی.

نویسنده: مهدی پایروند
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۲:۵۷

البته اگر از بازه عددی کوچکی استفاده بشه که ممکنه این تاریخ هم بلا استفاده بمونه، همچنین این تاریخ قرار هست کجا نگهداری بشه

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۳:۰۴

در تمام قسمت‌های رمزنگاری شده سایت جاری از یک salt با طول عمر یک روز استفاده می‌شود. این salt به کلید رمزنگاری اطلاعات اضافه می‌شود. بنابراین اطلاعات رمزنگاری شده توسط آن کلید، فقط در طی روز جاری قابل رمزگشایی خواهند بود. برای مثال لینک یک پیغام خصوصی سایت را در جایی ذخیره کنید. تا پایان روز کار می‌کند. روز بعد به صورت خودکار منقضی شده (چون salt فردا چیز دیگری است) و اطلاعات روز قبل توسط کلید جدید قابل رمزگشایی نیست. به این ترتیب شخص نیاز خواهد داشت تا لینک جدیدی را در صفحه مداخل مرتبط دریافت کند.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۳:۰۵

در واقع باید متن رو مثلا شامل دو بخش، بخش اول، و بخش دوم هم تاریخ و زمان فعلی هستش که با یه splitter از هم جدا شدن و بعد رمزنگاری کرد، و در سمت سرور ابتدا بخش دوم رو با تاریخ و زمان فعلی (submit شدن فرم) مقایسه کرد و در صورتیکه اختلاف اونها بیشتر از 3 دقیقه (یا 2 یا ...) بود captcha غیر معتبر هستش.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۳:۰۷

یک روز برای یک captcha زمان زیادی نیست؟

نویسنده: مهدی پایروند
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۳:۰۸

البته بنظرم برای نگهداری نکردن تاریخ در سرور میشه salt رو همین تاریخ امروز در نظر گرفت

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۸ ۱۳:۱۰

این لایه اول کلی هست (در تمام قسمت‌های سایت). تست کنید ببینید مطابق تصویر بالایی که ارسال شده می‌تونید در مورد کپچای سایت جواب بگیرید یا نه ...

نویسنده: کیارش سلیمان زاده

تاریخ: ۱۳:۱۸ ۱۳۹۱/۰۹/۰۸

تو لایه‌های بعدی چطور تست میکنید؟
طریقه ایجاد salt رو میشه بیشتر توضیح بدید؟
چیزی رو در دیتابیس هم ذخیره میکنید؟

نویسنده: وحید نصیری
تاریخ: ۱۳:۲۱ ۱۳۹۱/۰۹/۰۸

salt مورد استفاده تاریخ ساده روز سیستم است؛ در بانک اطلاعاتی هم ذخیره نمیشود.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳:۴۴ ۱۳۹۱/۰۹/۰۸

ممنون،
"تست کنید ببینید مطابق تصویر بالایی که ارسال شده میتونید در مورد کپچای سایت جواب بگیرید یا نه..."
بله تست کردم.
بعد از اینکه مشخص شد salt متعلق به امروز هستش، چطور تست میکنید که آیا مطابق با تصویر بالا، متن کپی شده یا اصلی هستش؟

نویسنده: وحید نصیری
تاریخ: ۱۳:۴۷ ۱۳۹۱/۰۹/۰۸

در مرحله بعد از یک کش 5 دقیقه‌ای با کلیدی بر اساس IP شخص استفاده میکنم.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳:۵۴ ۱۳۹۱/۰۹/۰۸

مگه IP نسبت به ISP برای بعضی‌ها یکسان نیست؟
در این صورت مشکلی پیش نمیداد؟

نویسنده: وحید نصیری
تاریخ: ۱۳:۵۸ ۱۳۹۱/۰۹/۰۸

تا الان که کسی مشکلی گزارش نکرده.
همچنین مقدار ذخیره شده در این کش با طول عمر کوتاه 5 دقیقه‌ای به ازای هر بار ریفرش صفحه متفاوت خواهد بود.

نویسنده: رحمت اله رضایی
تاریخ: ۱۴:۱۱ ۱۳۹۱/۰۹/۰۸

هفته پیش کپچای سایت این مشکل را داشت ولی الان برطرف شده.

نویسنده: وحید نصیری
تاریخ: ۱۶:۱۲ ۱۳۹۱/۰۹/۰۸

گیرم اصلاً این کپچا نباشد. با anti forgery-token می‌خواهید چکار کنید؟ با مازول آنتی داس چطور و خیلی مسایل دیگر.
خلاصه صرف اینکه در داخل یک مرورگر به کمک یک افزونه برای مدت کوتاهی توانستید اطلاعاتی را شبیه سازی کنید به این معنا نیست که قابلیت استفاده وسیع و خارج از مرورگر را هم به سادگی دارا است (مثلاً پروت فورس پسورد کاربران). تا حد شبیه سازی کامل رفتار یک مرورگر باید پیش برید (و ... اصلاً کار ساده‌ای نیست).

نویسنده: رحمت اله رضایی

تاریخ: ۲۱:۵۴ ۱۳۹۱/۰۹/۰۸

هدف من از طرح این مساله این بود که نمایش تصویر امنیتی بدون در نظر گرفتن این مورد هیچ فایده ای برای سایت نخواهد داشت و کار به ماژول آنتی داس و مشکلات مربوط به آن کشیده می شود (یک لایه عقب تر). در واقع کپچا یک تصویر تزئینی می شود.

دستکاری اطلاعات ارسالی به سرور، فقط در اینجا مطرح نیست. عمده مورد استفاده آن در ویرایش idها (مثلا در DropDownList ها) ست. (ثبت نام مسکن برای شهری که هنوز ثبت نام آن فعال نشده، خرید بلیت جایگاه ویژه برای یک کنسرت، پرداخت بیمه با دستکاری هزینه و ...)

نویسنده: داود حنیفی
تاریخ: ۹:۵۱ ۱۳۹۱/۰۹/۲۱

با سلام
ببخشید این مورد تو vs2012 چطور کار میکنه؟
تست من تو vs2012 جواب نداد.
با تشکر از زحماتون.

نویسنده: مهدی پایروند
تاریخ: ۹:۴۶ ۱۳۹۱/۰۹/۲۱

من این [مثال](#) رو با vs2012 باز کردم مشکلی نداشت

نویسنده: داود حنیفی
تاریخ: ۱۰:۳۴ ۱۳۹۱/۰۹/۲۱

ببخشید من اشتباه کردم. منظورم با (.net 4.5) بود.

نویسنده: مهدی پایروند
تاریخ: ۱۰:۵۰ ۱۳۹۱/۰۹/۲۱

با اون دات نت 4.5 هم مشکلی نداشت
اگه کارتون با نمونه پروژه تبدیل شده راه میوفته من اونو تو این لینک گذاشتم
[PersianCaptcha_2012-12-11_10-51-20.rar](#)

نویسنده: مسعود
تاریخ: ۱۶:۱۹ ۱۳۹۱/۱۱/۲۶

سلام
لازم است در ابتدا بابت این پروژه تشکر کنم.
من مشکلی با این کپچا دارم و آن این است که روی محیط تست خودم کار می کند اما زمانی که آن را پابلیش می کنم و روی سرور اصلی قرار می دهم، تصویر کپچا نمایش داده نمی شود.
لازم به ذکر است که از پروتوکل https استفاده می کنم.

با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۶:۵۴ ۱۳۹۱/۱۱/۲۶

یک حدس از راه دور:

تنظیم پیش فرض این پروژه در وب کانفیگ مربوط به IIS6 است (system.web/httpHandlers). برای IIS7 تنظیم جداگانه‌ای (system.webServer/handlers) نیاز دارد.

نویسنده: مسعود
تاریخ: ۱۷:۲۳ ۱۳۹۱/۱۱/۲۶

سلام

ممنون از پاسختون.

اما IIS من ۷ است و موردی رو هم که فرمودید پیش از این چک کردم ولی تغییری حاصل نشده بود.

نویسنده: مسعود
تاریخ: ۱۷:۳۲ ۱۳۹۱/۱۱/۲۶

با راه اندازی مجدد IIS مشکل رفع شد.

ممنون

نویسنده: مرتضی مختاری
تاریخ: ۱۶:۱۶ ۱۳۹۲/۱۰/۲۲

سلام؛ بنده آخر متوجه نشدم که از کپچا به این صورت استفاده کنیم یا نه؟

استفاده از expiration time به نظر من جواب نمیده چون توسط ربات تویه عرض یک دقیقه میشه هزار تا comment ثبت کرد. فکر کنم استفاده از captcha به همین روش و استفاده از anti forgery-token هیچ مشکل امنیتی نداشته باشه. شما چطوری مشکل این روش رو برطرف کردید؟ با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۶:۲۱ ۱۳۹۲/۱۰/۲۲

نمونه بهبود یافته مطلب جاری در اینجا « [نحوه ایجاد یک تصویر امنیتی \(Captcha\) با حروف فارسی در ASP.Net MVC](#) »