

ASP.NET جهت مقابله با حملات XSS بطور پیش‌فرض از ورود تگ‌های HTML جلوگیری می‌کند و در صورتی که ورودی کاربر شامل این تگ‌ها باشد، `HttpRequestValidationException` صادر می‌گردد. لاگ کردن و بررسی این خطاها جهت آگاهی از وجود حمله بی اهمیت نیست. اما متأسفانه [ELMAH](#) که به عنوان معمول‌ترین ابزار ثبت خطاها کاربرد دارد این نوع Exception ها را ثبت نمی‌کند. دلیل آن هم این است که ELMAH در رویه‌های درونی خود اقدام به خواندن ورودی‌های کاربر می‌کند و در این هنگام اگر ورودی کاربر نامعتبر باشد، Exception مذکور صادر می‌شود و فرصتی برای ادامه روند و ثبت خطا باقی نمی‌ماند. به هر حال ضروری است که این نقیصه را خودمان جبران کنیم. راه حل افزودن یک فیلتر سفارشی برای ثبت خطاها به شکل زیر است (ASP.NET MVC):

```
public class ElmahRequestValidationErrorFilter : IExceptionHandler
{
    public void OnException(ExceptionContext context)
    {
        if (context.Exception is HttpRequestValidationException)
            ErrorLog.GetDefault(HttpContext.Current).Log(new Error(context.Exception));
    }
}
```

فیلتر فوق باید در `Global.asax` معرفی شود:

```
public static void RegisterGlobalFilters (GlobalFilterCollection filters)
{
    filters.Add(new ElmahRequestValidationErrorFilter());
    filters.Add(new HandleErrorAttribute());
}
```

به این ترتیب `HttpRequestValidationException` هم بعد از این در سیستم ELMAH ثبت خواهد شد.

نظرات خوانندگان

نویسنده: یاسر مرادی
تاریخ: ۱۴:۳ ۱۳۹۱/۱۱/۲۸

با سلام، در چک لیست ASP.NET MVC مورد زیر وجود دارد
آیا مورد زیر کماکان معتبر است ؟
- فیلتر پیش فرض مدیریت خطاها حذف و بجای آن از [ELMAH](#) استفاده شود.

با سپاس

نویسنده: وحید نصیری
تاریخ: ۱۴:۲۶ ۱۳۹۱/۱۱/۲۸

- به سطر HandleErrorAttribute پیش فرض نیازی نیست (البته اگر تنظیمات وب کانفیگ درستی داشته باشید). [در قسمت 16](#)
سری MVC توضیح دادم. وجود آن سبب می‌شود که ELMAH اصلاً کار نکند و خطای مدیریت نشده‌ای به آن ارجاع داده نشود (چون
قبلاً مدیریت شده).
- بله. همچنان ELMAH معتبر است. نکته فوق را هم اضافه کنید، کاملتر خواهد شد.

نویسنده: داود زینی
تاریخ: ۱۴:۲۸ ۱۳۹۱/۱۱/۲۸

سلام
این چک لیست توسط آقای نصیری تهیه شده بود. این مورد هم از نظر من معتبر است.
با تشکر