

[← Blog posts](#)

## Episerver Forms: Avoiding spam with a honeypot

C# TIPS AND TRICKS CMS OPTIMIZEZY (EPISERVER)



Alan Thraen | 1 Years Ago | PDF |



Spam comes in many forms and can be really annoying. Often, when you put a form on your website it will be found by spam bots that will post lots of spam in the form. A common defense is CAPTCHA's, but they are annoying the real users and typically not WCAG compliant. Here, I'm showing a simple Episerver implementation of another approach that works wonders for me - the honeypot.

It's amazing how much spam can be aggregated by something as simple as a form online. I used to have daily spam-submissions - just to my regular little contact form on this website and it quickly reached a point where I would risk missing actual incoming leads by simply dismissing them as spam.

Now, I didn't really want to implement a captcha, as I've never been a big fan - and I've seen a fair share of terrible captchas (like those with completely incomprehensible computer generated handwriting you have to type out) - so I was looking for alternatives, and it seems like honeypots is a good alternative.

The basic idea with a honeypot is to add honeypot form field to your form. A field that in html (to the bots) looks like a regular form field that should be filled out - but then is hidden to the real web visitors using CSS or javascript. Then - it's just a matter of checking when a form is submitted if the field is filled out - and if it is, you caught the spam and can delete it. If it's not filled out, you can assume it is very likely legit (or really clever spam).

Form elements

<input type="checkbox"/>	Name
<input type="checkbox"/>	Company
<input type="checkbox"/>	Email
<input type="checkbox"/>	Phone
<input type="checkbox"/>	Message
<input type="checkbox"/>	HoneyPotElementBlock
<input type="checkbox"/>	GDPR Text
<input checked="" type="checkbox"/>	Consent
<input checked="" type="checkbox"/>	Send

You can drop form elements here or [create a new element](#).

It's implemented as a new ElementBlock for Episerver Forms that you simply drag onto your form. Along with it is an actor that will disregard the submission if the field is filled out. It's pretty rough around the edges right now, but seems to work well on this site. Improvement suggestions are appreciated!

```

1 using EPIServer.Forms.Core.PostSubmissionActor;
2 using EPIServer.Forms.Core.PostSubmissionActor.Internal;
3 using EPIServer.Forms.Helpers.Internal;
4 using System;
5 using System.Collections.Generic;
6 using System.Linq;
7 using System.Web;
8
9 namespace Demo.Web.Business.Forms
10 {
11     public class HoneyPotActor : PostSubmissionActorBase, ISyncOrderedSubmissionActor
12     {
13         public int Order => 1;
14
15         public override object Run(object input)
16         {
17             var formContainerBlock = this.FomIdentity.GetFormBlock();
18             var allFormElements = formContainerBlock.Form.Steps.SelectMany(st => st.Elements).Where(elem => elem.SourceContent.Is HoneyPot
19             if (allFormElements.Any())
20             {
21                 foreach (var fe in allFormElements)
22                 {
23                     if (string.IsNullOrEmpty(this.SubmissionData.Data[fe.ElementName]?.ToString()))
24                     {
25                         var result = new SubmissionActorResult();
26                         result.CancelSubmit = true;
27                         result.ErrorMessage = "Spam bot detected";
28                         return result;
29                     }
30                     else
31                     {
32                         //Remove data for HoneyPotElement from final data
33                         this.SubmissionData.Data.Remove(fe.ElementName);
34                     }
35                 }
36             }
37
38             return "";
39         }
40     }
41 }

```

HoneyPotActor.cs hosted with ❤️ by GitHub

[view raw](#)

```

1 using EPIServer.DataAnnotations;
2 using EPIServer.Forms.Core;
3 using System;
4 using System.Collections.Generic;
5 using System.Linq;
6 using System.Web;
7
8 namespace Demo.Web.Business.Forms
9 {
10     [ContentType(DisplayName = "HoneyPot Spam Avoidance", GUID = "B8248208-75D5-461A-6F5C-62898697B010", Description = "")]
11     public class HoneyPotElementBlock : ElementBlockBase
12     {
13     }
14 }
15 }

```

HoneyPotElementBlock.cs hosted with ❤️ by GitHub

[view raw](#)

```

1 @model Demo.Web.Business.Forms.HoneyPotElementBlock
2 <style>
3     .ohnohoney {
4         opacity: 0;
5         position: absolute;
6         top: 0;
7         left: 0;
8         height: 0;
9         width: 0;
10        z-index: -1;
11        tabindex: -1;
12    }
13 </style>
14 <label class="ohnohoney" for="@ModelFormElement.Guid"></label>
15 <input type="text" name="@ModelFormElement.ElementName" autocomplete="off" placeholder="Your name" id="@ModelFormElement.Guid" class="ohnoh

```

HoneyPotElementBlock.cshtml hosted with ❤️ by GitHub

[view raw](#)

[C#](#) [TIPS AND TRICKS](#) [CMS](#) [OPTIMIZEZY \(EPISERVER\)](#)

Post Comments 0