

# Web2CRM Honeypot Captcha Manual

V 1.0

**How to keep those pesky robots from submitting forms into your system**



**CRM Innovation LLC 8527 Bluejacket Street Lenexa, KS 66214**

**[crminnovation.com](http://crminnovation.com)**

## Contents

Overview .....	3
How it works .....	3
Level 1 Protection – Carry Code.....	4
Level 2 Protection – Carry Code.....	4

## Overview

In the interest of continuing to improve product security while maintaining usability, Web2CRM is implementing an additional [Captcha methodology](#) known as Honeypot Captcha. The way this works, a field is added to your Web2CRM form that is invisible to website visitors, but is visible to Spam-bots and other malicious software that searches the Internet to automatically insert and submit forms.

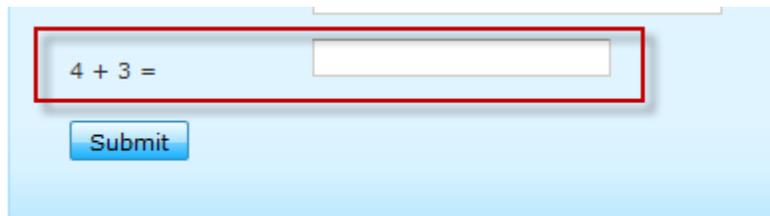
This feature offers two levels of additional protection against the robots. If you use the Hosted Form approach effective October 10, 2011 Level 2 protection is automatically included in the Hosted Forms you are current using or create in the future. You do not need to do anything to your form or URL reference to the hosted form link. Level 1 protection is included in the Carry Code option as of this date also. If you want to apply the more stringent Level 2 protection to your Carry Code then you will need to manually edit the web page code the form is placed on. This would apply to existing or new forms. Most users will not need to take the extra step of Level 2 for Carry Code unless your forms continue to being harassed by the Spam-bots.

## How it works

The Honeypot Captcha works by adding a field to your form that is invisible to a real person filling out the form. However, when a robot scans the source code of the page and fills out the form automatically it 'sees' the hidden field and tries to insert data into. When the form is submitted, the Web2CRM application inspects the form data and if it sees that there is data in the honeypot field then it knows a robot filled out the form. Therefore, it knows it is not legitimate and it will discard and not submit it to your CRM system. The robot form submitter will still be sent to the success page as if it was a legitimate submission. We want to make it think the form submission was successful even though it was not.

Since a human being will not see the field on the page, when they submit it, the honeypot field will be empty and the Web2CRM application will consider it legitimate and process it to the CRM system.

**NOTE:** If human beings are [manually or semi-manually pasting data into the forms](#) and submitting them the honeypot Captcha won't trap them. You will need to continue to rely on our simple but effective math test question.

A screenshot of a web form with a light blue background. The form contains a simple arithmetic question: "4 + 3 =". To the right of the question is an empty text input field. Below the input field is a blue "Submit" button. A red rectangular box highlights the entire form area, including the question, the input field, and the button.

Built-in Simple Arithmetic Captcha

You will still want to user in either case of applying Level 1 or 2 Honeypot protection our standard simply math Captcha.

## Level 1 Protection – Carry Code

**New and republished forms:** The Web2CRM designer will add automatically add a `div` tag with the id “TARGET\_\_” inserted between the end tag for the table and the end tag for the form in the Carry Code as illustrated below:

```
<div id="TARGET__" style="display:none; width:100%;">
  <label for="crmi_employee_target" >Business Email</label>
  <input id="crmi_employee_target" runat="server" />
</div>
```

Notice that the inline styling for the “TARGET\_\_” tag sets the display property to “none”. This is what hides the Honeypot field from view on the webpage.

**Legacy forms:** If you want to add this protection to any legacy (existing published forms on your website) without having to go to the Web2CRM designer, re-generate the Carry Code and insert it into your webpage then just copy the four lines of code above. Paste the code snippet into the existing web page code immediately above the closing table tag, which looks like this in your web page code:  
**</table>**

There is nothing else to do.

**NOTE:** do not change any of the code inside the four lines or it will break the form submission.

## Level 2 Protection – Carry Code

In the Level 1 protection the CSS styling is done inline, it may not fool more sophisticated malicious software because it can surmise that the field is invisible to humans based on the styling context. The advanced method (Level 2) requires a little more effort.

The Level 2 method for hiding the Honeypot is to use external styling via CSS to the web page. In order to accomplish this, use the following code snippet where we have removed the style attribute from the div tag shown in the previous code snippet.

```
<div id="TARGET__">
  <label for="crmi_employee_target" >Business Email</label>
  <input id="crmi_employee_target" runat="server" />
</div>
```

Next, you need to either add a reference to the Web2CRM Level 2 CSS, or modify the site’s CSS to include the same markup from the inline styling. To add a reference to the Web2CRM Level 2 CSS, add the following line into the page’s head tag:

```
<link href="https://web2crm.crminnovation.com/Level2.css" type="text/css"
rel="Stylesheet" />
```

If you elect not to utilize the CSS style that is provided on the Web2CRM application server you can modify the CSS style sheet that is being used to style the page on your site's webserver. In that case you would add the following style directive to the CSS file on your webserver.

```
#TARGET__{
    display: none;
    width: 100%;
}
```

In summary, both Level 1 and 2 protections can be implemented without recreating or republishing any of your existing forms as long as you follow the steps in this guide exactly.