

Cognito Forms Integration

HandL UTM Grabber / Tracker How to collect and track UTM variables via Cognito Forms step by step

- [Cognito Form: Embed script implementation](#)
- [Cognito Form: Capture UTMs using Iframe](#)

Cognito Form: Embed script implementation

First, make sure all the UTM fields are created in Cognito Form. Each field should be separate and Field Name should match the name you will use later in the implementation (see the image below).

The image shows the Cognito Form configuration interface. On the left, the 'Field Settings - Textbox' panel is open, showing the following settings:

- Label:** utm campaign
- Field Name:** Utmcampaign
- Type:** Single Line (selected), Multiple Lines, Password
- Placeholder Text:** test
- Format Validation:** (empty dropdown)
- Default Value:** (empty field)

On the right, a preview of the form is shown. It includes a title 'test', a label 'testing', an 'Email' field, and a 'utm campaign' field with the placeholder text 'A test'. A context menu is visible over the 'utm campaign' field, showing options: Cut, Copy, Delete, Insert Field, Make Smaller, Make Bigger, and Justify Row. Below the form fields is a 'Submit' button and a message: 'Thank you for filling out the form. Your response has been recorded.'

Your default embed code from Cognito Form should look like this

```
<script src="https://services.cognitoforms.com/s/<YOUR FORM ID>"></script>
<script>Cognito.load("forms", { id: "2"});</script>
```

Modify the embed code from Cognito Forms like the following. Notice that, we are adding entry fields to the `Cognito.load`.

```
<script src="https://services.cognitoforms.com/s/<YOUR FORM ID>"></script>
<script>Cognito.load("forms", { id: "2",
entry: {
  ["UTMCampaign":Cookies.get("utm_campaign"),
```

```
"UTMSource":Cookies.get("utm_source") ,  
"UTMMedium":Cookies.get("utm_medium"),  
"UTMContent":Cookies.get("utm_content"),  
"UTMTerm":Cookies.get("utm_term"),  
"IP": Cookies.get("handl_ip"),    "Organic": Cookies.get("organic_source_str")  
}});</script>
```

Here we only illustrated `utm_*` and `handl_ip`. However you can use all the other parameters very similar way. See the list of all the parameters available [here](#)

Native WP Shortcodes

NOTE: Cognito does not support hidden field in the form out of the box. However you can make a regular text field hidden by creating a condition that never satisfies like this.

The screenshot displays a form configuration interface with a modal dialog titled "Visible When...". The background interface includes sections for "Placeholder Text" (with a "test" input), "Format Validation", "Default Value" (set to "="), "Number of Characters" (with "Minimum" and "Maximum" fields), "Help Text", "Show This Field" (set to "When" with a condition "Email is filled out"), "Require This Field" (with "Always", "When", and "Never" options), "Read-Only" (with "Always", "When", "Internally", and "Never" options), "Limit Quantity" (with "No Duplicates", "Specific Quantity", and "Never" options), and "Show Custom Error" (with "When" and "Never" options). The modal dialog "Visible When..." is open, showing a condition builder with the first rule: "Email" (selected from a dropdown) "contains" (selected from a dropdown) "123131312asdsad" (entered in a text field). Below this, there is a section for adding more rules with a "+" icon and the word "or". At the bottom of the modal, there are tabs for "Basic Editor" and "Advanced Editor", and buttons for "Cancel" and "Save".

Cognito Form: Capture UTMs using Iframe

First, make sure all the UTM fields are created in Cognito Form. Each field should be separate and Field Name should match the name you will use later in the implementation (see the image below).

The image shows the Cognito Form interface. On the left is the 'Field Settings - Textbox' panel. It contains the following fields: 'Label' with the value 'utm campaign', 'Field Name' with the value 'Utmcampaign', 'Type' with 'Single Line' selected, 'Placeholder Text' with the value 'test', and 'Format Validation' with a dropdown menu. On the right is a preview of the form. It has a title 'test', a label 'testing', and an 'Email' field. Below the email field is a 'utm campaign' field with the placeholder text 'A test'. A context menu is open over the 'utm campaign' field, showing options: Cut, Copy, Delete, Insert Field, Make Smaller, Make Bigger, and Justify Row. Below the form fields is a 'Submit' button. At the bottom of the preview, it says 'Thank you for filling out the form. Your response has been recorded.'

Your default iframe code from Cognito Form should look like this

```
<iframe src="https://www.cognitoforms.com/f/<YOUR FORM ID>?id=2"
style="position:relative;width:1px;min-width:100%;*width:100%;" frameborder="0"
scrolling="yes" seamless="seamless" height="323" width="100%"></iframe><script
src="https://www.cognitoforms.com/scripts/embed.js"></script>
```

Modify the iframe code from Cognito Forms like the following. Notice that, we are adding entry fields to the `Cognito.prefill`.

```
<iframe src="https://www.cognitoforms.com/f/<YOUR FORM ID>?id=2"
style="position:relative;width:1px;min-width:100%;*width:100%;" frameborder="0"
```

```
scrolling="yes" seamless="seamless" height="323" width="100%"></iframe><script
src="https://www.cognitoforms.com/scripts/embed.js"></script>
<script>
Cognito.prefill({
  "Utmcampaign": "[utm_campaign]",
    "UTMSource": "[utm_source]" ,
    "UTMMedium": "[utm_medium]",
    "UTMContent": "[utm_content]",
    "UTMTerm": "[utm_term]",
    "IP": "[handl_ip]",
    "Organic": "[organic_source_str]"
});
</script>
```

Here we only illustrated `utm_*` and `handl_ip`. However you can use all the other parameters very similar way. See the list of all the parameters available here [Native WP Shortcodes](#)

NOTE: Cognito does not support hidden field in the form out of the box. However you can make a regular text field hidden by creating a condition that never satisfies like this.

The screenshot displays the Cognito Forms editor interface. A modal window titled "Visible When..." is open, showing a condition: "Email" (selected from a dropdown) "contains" (selected from a dropdown) "123131312asdsad" (entered in a text field). The modal also includes a "+" button to add more conditions and a "Save" button. The background shows the form editor with various settings for a field, including Placeholder Text, Format Validation, Default Value, Number of Characters, Help Text, Show This Field, Require This Field, Read-Only, Limit Quantity, and Show Custom Error.