CUNY Academic Commons - Bug #8570

Anti-spam for Contact Form 7

2017-08-22 06:27 PM - Boone Gorges

Status:	Resolved	**Start date:**	2017-08-22
Priority name:	Normal	**Due date:**	
Assignee:	Raymond Hoh	**% Done:**	0%
Category name:	Spam/Spam Prevention	**Estimated time:**	0.00 hour
Target version:	1.11.11		

Description

We were informed by IT that some contact forms powered by Contact Form 7 were being used to send some spam emails to the site admins. I dug into the logs and found the following (from my email to IT):

> 158.222.111.170 - - [22/Aug/2017:13:27:04 ~~0400] "GET /contact-us/ HTTP/1.1" 200 124474 "~~" "-" 23284
> 158.222.111.170 - - [22/Aug/2017:13:27:07 ~~0400] "GET /files/wpcf7_captcha/3072338607.png HTTP/1.1" 200 1204 "~~" "-" 23284
> 158.222.111.170 - - [22/Aug/2017:13:27:20 -0400] "POST /contact-us/ HTTP/1.1" 200 124594 "http://www.google.com" "Mozilla/5.0 (IE 11.0; Windows NT 6.3; Trident/7.0; .NET4.0E; .NET4.0C; rv:11.0) like Gecko" 23380
>
> Notice the google.com referer in the third line. Looking at this, my guess is that a bot is doing the following:
>
> - In a browser, search Google for pages containing contact forms powered by this specific plugin.
> - Based on results, fetch the contact page. This is presumably **not** being done with a web browser, because all of the rest of the page assets (CSS, JS, etc) are not being loaded, and there's no user-agent recorded in the access log.
> - Parse the contact page HTML to get the URL of the CAPTCHA image, and then fetch that image.
> - Use some technique to solve the CATCHA
> - Build a POST request known to work with this plugin, including the CAPTCHA answer, and launch it from the browser. This explains why the POST request has a User-Agent as well as a Referer.

I think we can thwart this with a sort of reverse honeypot: a hidden field that has a secret token, which is rendered as part of the form but is not part of CF7. It must be part of the POST request in order for the submission to go through.

Ray, have you built this kind of thing before? If so, and you have any code, would you mind sharing? Otherwise I can try to whip something up.

Related issues:

Related to CUNY Academic Commons - Bug #9489: Email spam possibly related to ...	**Resolved**	**2018-03-26**

History

#1 - 2017-08-22 10:33 PM - Raymond Hoh

- Category name set to Spam/Spam Prevention

There's a CF7 honeypot plugin I found on the wordpress.org plugin repository:
https://wordpress.org/plugins/contact-form-7-honeypot/

Looks simple enough.

Could also hook Akismet into Contact Form 7 as per CF7's docs:
https://contactform7.com/spam-filtering-with-akismet/

Also, this blog article goes into all the options available for counteracting spam with CF7:
https://barn2.co.uk/stop-contact-form-7-spam/

I guess all those options would still require some configuring on the user's side.

Maybe there is a way to filter the CF7 form contents so the anti-spam options for the CF7 form are rendered all the time. Since we already use Akismet, we should try injecting Akismet into all our CF7 forms first. I'll take a look and see how difficult that may be.

#2 - 2017-08-23 11:08 AM - Boone Gorges

Yeah, injecting Akismet for all CF7 instances seems like it'd be the best. It appears that CF7 doesn't have a ton of filter points to make this easy. Worst case scenario, we could create our own version of wpcf7_akismet(), which would skip the wpcf7_akismet_submitted_params() check (since we'd be forcing all compatible params to use Akismet).

If you get time to look at this in the next couple days, that'd be great. Otherwise I'll see what I can rig up before the 1.11.11 release.

#3 - 2017-08-24 03:12 PM - Raymond Hoh

- Status changed from Assigned to Staged for Production Release

I've added Akismet protection to all CF7 forms in
https://github.com/cuny-academic-commons/cac/commit/60dc08b54ce0d7d39318b685dcedd728ebf62e71.

Since each CF7 form can be different, based on the CF7 Akismet docs, I've only injected the akismet:author_email attribute since there is always going to be an [email] tag. We can't really determine what is going to be the author name or URL field, so I left those out.

I tested locally and Akismet protection worked.

#4 - 2017-08-24 10:26 PM - Boone Gorges

Cool, thanks, Ray!

> Since each CF7 form can be different, based on the CF7 Akismet docs, I've only injected the akismet:author_email attribute since there is always going to be an [email] tag. We can't really determine what is going to be the author name or URL field, so I left those out.

Does this mean that all items will be sent to Akismet, because CF7 only needs to have a single Akismet-specified field to send the entire submission through Akismet? Or does it mean that only the email field is sent to Akismet?

#5 - 2017-08-25 11:17 AM - Raymond Hoh

> Does this mean that all items will be sent to Akismet, because CF7 only needs to have a single Akismet-specified field to send the entire submission through Akismet?

Yes, this looks to be the case even though we only use the one akismet:author_email attribute for CF7.

The following was the query sent to Akismet as a querystring, I just formatted it a bit better for viewing here:

```
comment_author:

comment_author_email:
    test@test.com

comment_author_url:

comment_content:
    Test

    Testing Akismet

    What?

blog:
    http://localhost/

blog_lang:
    en_US

blog_charset:
    UTF-8

user_ip:
    ::1

user_agent:
    Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/57.0.2987.133 Safari/537
.36

referrer:
    http://localhost/hello-world/

comment_type:
    contact-form

SERVER_SOFTWARE:
    Apache
```

```
REQUEST_URI:
    /wp-json/contact-form-7/v1/contact-forms/1256/feedback

REDIRECT_STATUS:
    200

HTTP_HOST:
    localhost

HTTP_CONNECTION:
    keep-alive

CONTENT_LENGTH:
    1010

HTTP_ACCEPT:
    application/json, text/javascript, */*; q=0.01

HTTP_ORIGIN:
    http://localhost

HTTP_X_REQUESTED_WITH:
    XMLHttpRequest

HTTP_USER_AGENT:
    Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/57.0.2987.133 Safari/537
.36

CONTENT_TYPE:
    multipart/form-data; boundary=----WebKitFormBoundaryLEkr88OVybKHK0uL

HTTP_REFERER:
    http://localhost/hello-world/

HTTP_ACCEPT_ENCODING:
    gzip, deflate, br

HTTP_ACCEPT_LANGUAGE:
    en-US,en;q=0.8

SERVER_SIGNATURE:

SERVER_NAME:
    localhost

SERVER_ADDR:
    ::1

SERVER_PORT:
    80

REMOTE_ADDR:
    ::1

DOCUMENT_ROOT:
    /www

REQUEST_SCHEME:
    http

CONTEXT_PREFIX:

CONTEXT_DOCUMENT_ROOT:
    /www

SERVER_ADMIN:
    admin@localhost

SCRIPT_FILENAME:
    /index.php

REMOTE_PORT:
    64594
```

```
REDIRECT_URL:
    /wp-json/contact-form-7/v1/contact-forms/1256/feedback

GATEWAY_INTERFACE:
    CGI/1.1

SERVER_PROTOCOL:
    HTTP/1.1

REQUEST_METHOD:
    POST

QUERY_STRING:

SCRIPT_NAME:
    /index.php

PHP_SELF:
    /index.php

REQUEST_TIME_FLOAT:
    1503601753.147

REQUEST_TIME:
    1503601753
```

#6 - 2017-08-25 03:59 PM - Boone Gorges

Gotcha. Thank you, sir!

#7 - 2017-09-01 03:50 PM - Boone Gorges

- Status changed from Staged for Production Release to Resolved

Deployed.

#8 - 2018-03-26 05:13 PM - Raymond Hoh

- Related to Bug #9489: Email spam possibly related to CAC form added