

# Re: Preventing Request.Form abuse

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2006-10/msg03865.html>

---

- *From:* "Juan T. Llibre" <[nomailreplies@xxxxxxxxxxxxx](mailto:nomailreplies@xxxxxxxxxxxxx)>
  - *Date:* Tue, 24 Oct 2006 21:08:48 -0400
- 

re:

I'm still waiting for people to find holes in the suggestion though – Juans a likely candidate for sinking my idea.....lol

<lol>

I've been mulling it over, trying to think of a failsafe method...and drawing a blank.

re:

A list of forms that are only subject to postback on submission is easy to create and could reside in web.config (or anywhere cachable) – crude, but we can think of another way later.

That could get cumbersome.

re:

A begin request intercepted in an ihttpmodule could verify the ispostback property of any request.

Remember that the legal use of the form also involves a postback ( does it, Mark? ), so you can't block on the basis of the request being a postback.

re:

If its not a postback form, and is in the list of forms that require postback then dump the request and return a redirect to some random fictitious URL. It wont even touch the actual form being requested.

If what I suspect is true, the reverse procedure would work.

Re: Preventing Request.Form abuse

As far as I can determine, the crux is that they aren't using postback, but posting directly to the form.

( Am I right in assuming that ? ) I need a reality check ... ;-)

If that is so, we shouldn't be thinking about payback (as tempting as it is), but about a solid defense, preferably one which is simple to implement.

So, if the request being a postback is a requirement, would checking for IsPostBack and, if it isn't a Postback, clearing all the fields accomplish what we want ?

If the hackers/spammers are \*not\* using Postback (I am assuming that...) that should work.

The key is whether they \*are\* POSTing without having requested the page. If not, then I need to think about this some more.

re:

On detecting an attempt to use a postback it would actually be quite easy to also block their IP real time in the filter

Wouldn't blocking postbacks also block the intended legal use of the feedback form ?

Juan T. Llibre, asp.net MVP  
asp.net faq : <http://asp.net.do/faq/>  
foros de asp.net, en español : <http://asp.net.do/foros/>

=====  
"John Timney (MVP)" <x\_john@xxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:cZ-dnVXshMe0FqPYnZ2dnUVZ8smdnZ2d@xxxxxxxxxxxxxxxxxxxx](mailto:news:cZ-dnVXshMe0FqPYnZ2dnUVZ8smdnZ2d@xxxxxxxxxxxxxxxxxxxx)

I think I would redirect them to a large video file on one of the online video places which may well crash their program with the size of the response. That said, its not fair to send them to someone else server and use their bandwidth, hence the suggestion of the fictitious URL.

On detecting an attempt to use a postback it would actually be quite easy to also block their IP real time in the filter, so any future request from them was always dropped or always resulted in a large video being sent as the response. It would be a one hit system.

I've done most of what we're discussing in the past on net 1.1, but not for this reason so the code should be very easy to put together.....I'm still waiting for people to find holes in the suggestion though – Juans a likely candidate for sinking my idea.....lol

--  
--  
Regards

Re: Preventing Request.Form abuse

John Timney (MVP)  
VISIT MY WEBSITE:  
<http://www.johntimney.com>  
<http://www.johntimney.com/blog>

"Mark Rae" <mark@xxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:edz\\$3M79GHA.3348@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](news:edz$3M79GHA.3348@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

"John Timney (MVP)" <x\_john@xxxxxxxxxxxxxxxxxxxx> wrote in  
message  
<news:9cCdnSxkqMv0H6PYnZ2dnUVZ8qudnZ2d@xxxxxxxxxxxxxxxxxxxx>

John,

A list of forms that are only subject to postback on  
submission is easy to create and could  
reside in web.config (or anywhere cachable) – crude, but we  
can think of another way later. A  
begin request intercepted in an ihttpmodule could verify the  
ispostback property of any request.  
If its not a postback form, and is in the list of forms that  
require postback then dump the  
request and return a redirect to some random fictitious URL.  
It wont even touch the actual form  
being requested.

I like it!

If we were to use a real rather than a fictitious URL for the redirect, do you  
think that would  
be a good thing or a bad thing? I guess it would be a bad thing because (I  
suppose) it would look  
to the target URL that the posting was coming from our IP address rather  
than the spammer's IP  
address...

Being based in the UK, I think I would find it rather satisfying if the  
spammers suddenly found  
themselves trying to post here: <http://www.met.police.uk/computercrime/>

:–)

Re: Preventing Request.Form abuse