

Re: MTOM vs HttpRequest for file upload

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices/2006-10/msg00188.htm>

- *From:* "russell.lane" <russell.lane@xxxxxxxxxxxxxx>
 - *Date:* Fri, 27 Oct 2006 08:30:08 -0400
-

Steven –

Many thanks for this reply, and for your follow-up to me.

We are authenticating the client to the service layer using integrated Windows authentication. This is primarily for tracking who is doing what, i.e., auditing, and less for security per se. We have other controls on who can run the app itself, so the back end trusts the client to not permit illegal uploads. If you can get to the service at all through the app, the service assumes that you are a legitimate user.

The upload service, via its app pool identity, owns the permission to read and write to the backing filesystem, the user does not.

As it turns out, I also overestimated the size of the files. They are binary *.wav files ranging in size from the low K's to hundreds of K's of bytes.

From your comments so far my intuition is that HTTP POST will be the better choice. Does that seem correct?

Many thanks, again, for your comments, they were very helpful.

Best –

R

"Steven Cheng[MSFT]" <stcheng@xxxxxxxxxxxxxxxxxxxxxx> wrote in message news:fYbXQdz9GHA.1984@xxxxxxxxxxxxxxxxxxxxxx

Hello Russell,

As for the file uploading from your client winform application to the server service, do you need to do additional authentication against the client before upload or at each file uploading?

Re: MTOM vs HttpRequest for file upload

Regardless of any other requirement, from performance perspective, use HTTP

POST to upload file will be more efficient because the file content is directly transferred as binary content in the http post request. While using WSE MTOM, it still use some particular encoding schema to convert the file content thus it will add some additional overhead. However, comparing to directly transfer byte[] in webservice method, using MTOM will be much more efficient.

You can consider which approach to use according to your scenario. If there is any other questions on this, please feel free to post here.

Sincerely,

Steven Cheng

Microsoft MSDN Online Support Lead

This posting is provided "AS IS" with no warranties, and confers no rig