



How Changes in MS Internet Explorer
Version 8 Will Affect Web Pages
Containing Input type=file

1 Document History

Date	Version	Change Description	
8/27/2010	1.0	Initial Entry	
8/30/2010	2.0	Add JavaScript info	
8/31/2010	3.0	Includes feedback	

2 Table of Contents

1	Document History	2
2	Table of Contents	3
3	Introduction	4
3.1	Problem Solved	4
3.2	Audience	4
3.3	Assumptions	4
3.4	Scope	4
4	Basics	4
4.1	Example web page	4
4.2	What is the problem?	10
5	Conclusion	11

3 Introduction

This document describes problems that applications including html forms that include input type=file will encounter after transitioning to Microsoft Internet Explorer version 8.

3.1 Problem Solved

Due to the security enhancements added to Microsoft Internet Explorer version 8 (MS IE8), certain features that developers have become accustomed to, when using prior version of MS IE8 will no longer work when the browser used is upgraded to MS IE8. There are no known solutions to this change in behavior. Instead, this document's purpose is to make developers aware of this change.

3.2 Audience

This paper is intended for software engineers developing web-based applications that include a page that allows users to upload files into their target device. Unless your application includes this feature, this paper provides no information of value.

3.3 Assumptions

The issues described in this document represent changes in the behavior of Microsoft's Internet Explorer V8 as opposed to that same behavior in the V7 version. In addition, this document describes how those changes might affect Digi NET+OS AWS-based applications. There is no remediation described in this document as the change in behavior is outside the scope of Digi International. Our assumption is that you need to be informed of these changes, and you will make changes to your application or documentation as needed.

3.4 Scope

- The sole intent of this paper is to describe a change in the behavior of MS IE8 that will affect applications that allow users to upload files into a target device.

4 Basics

This section describes an example web page that might be included in an application affected by this change in MS IE8. It demonstrates the behavior before and after the change.

4.1 Example web page

The following code includes a simple html file that might allow a user to upload a file into a target device:

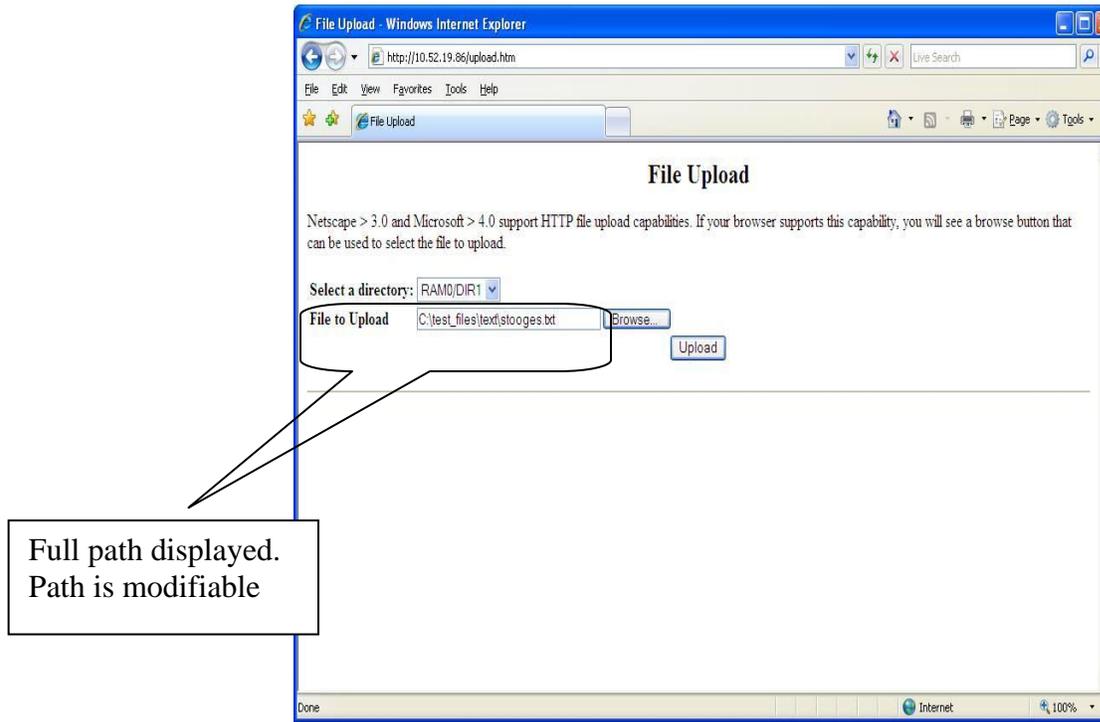
```
<html>  
<head>
```

MS IE8 Effects on File Upload Forms Applications

```
<title>Example web page including file upload</title>
</head>
<body>
<form method=post enctype="multipart/form-data">
<p>
File to upload:
<p>
<input type="file" name="file" size="32" maxlength="32">
<input type="SUBMIT" name="Submit" value="Upload">
</form>
</body>
</html>
```

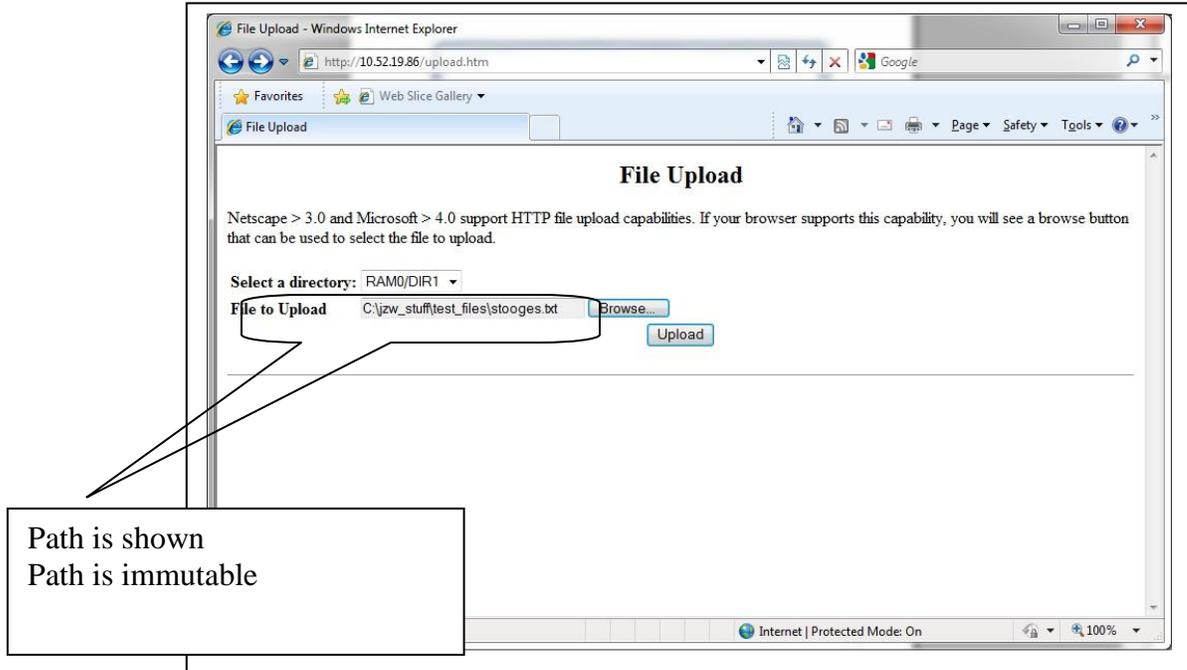
The slide on the next page displays what a user would see when attaching to a NET+OS AWS-developed application that includes a web page that allows file download.

MS IE8 Effects on File Upload Forms Applications



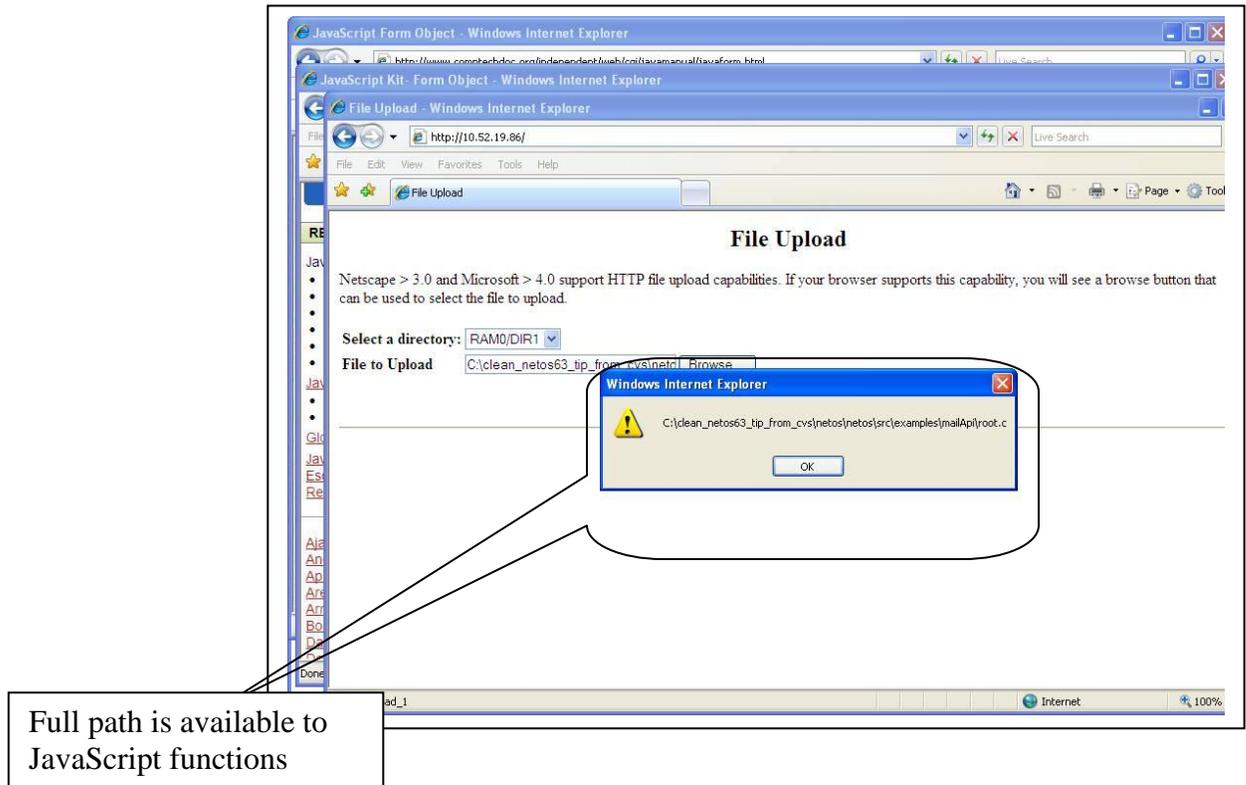
When MS Internet Explorer V7 or below is used as the web browser, there are two important user interface attributes. First, the entire path of the file to be uploaded is included in the form's file browser. Second, that path is mutable. You can place your cursor in the box and change the filename and path.

MS IE8 Effects on File Upload Forms Applications



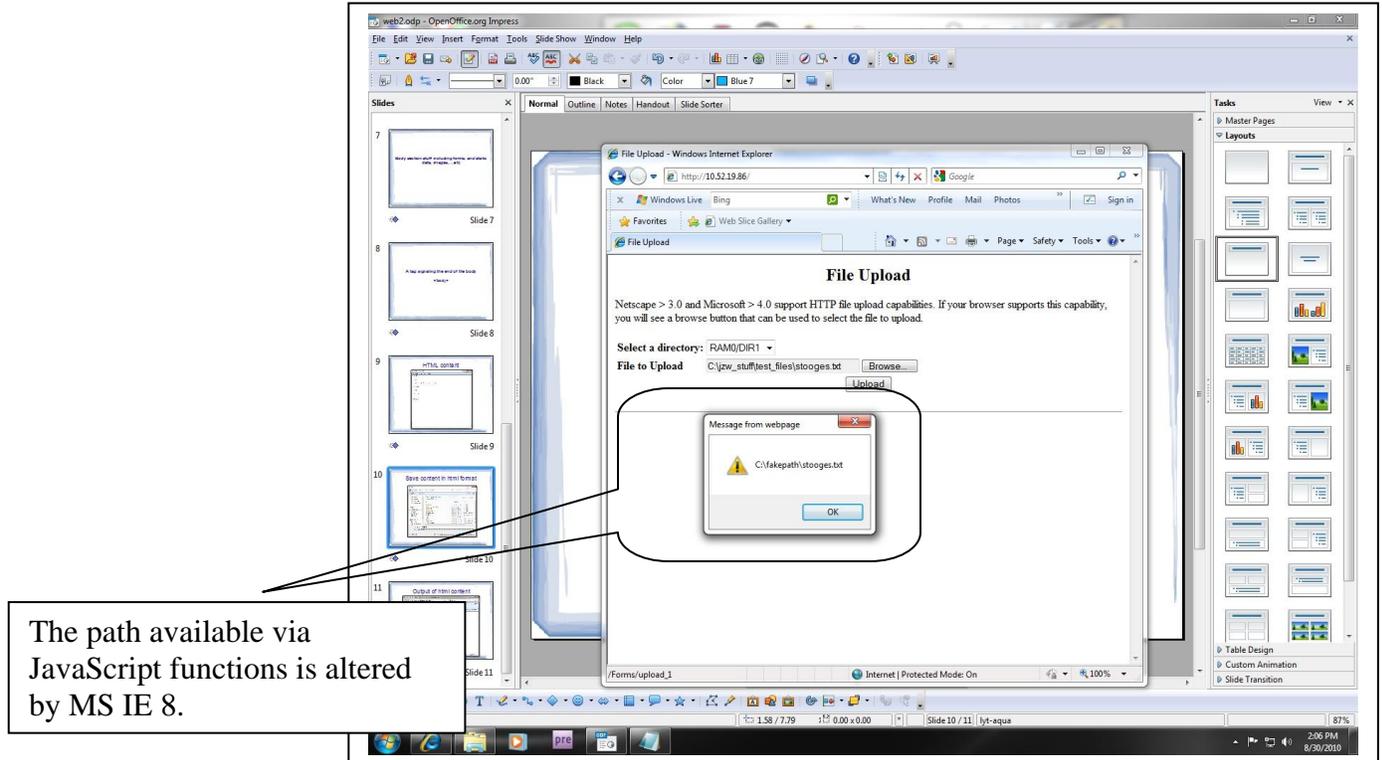
This slide shows the same operation performed using MS Internet Explorer Version 8. This slide is very similar to the slide shown for MS IE 7. The difference is that if you place your cursor in the file browser window, you are not allowed to type anything into the box. It is immutable (not changeable).

MS IE8 Effects on File Upload Forms Applications



A number of more sophisticated developers have become used to manipulating the file name associated with the form. This manipulation was performed through the use of JavaScript functions. Notice that in this screen shot, the full file name is available. This shot is the output of a JavaScript alert object.

MS IE8 Effects on File Upload Forms Applications



In this screen shot, we surfed to the same device that we did in the last screen shot. The difference is that in this shot, the browser is MS IE8 vs. MS IE 7. The big difference to notice is that the file path has been altered in the output of the alert box as opposed to that shown in the file browser object. MS IE8 has altered the path of the file, as far as JavaScript sees it. Also, the variable within JavaScript in MS IE8 is immutable. It is read only. You cannot alter it, as you might have using earlier versions of MS IE.

4.2 What is the problem?

We have received reports from customers stating that they were used to bringing up MS IE 7, surfing to their device and directly typing in the full path of a file that they wish to upload. This was possible in MS IE7. This is not possible in MS IE8. Microsoft has identified the ability to do this as a security threat. The solution to this security threat (as implemented in MS IE 8) is to disallow users from keying in the file name of files they wish to upload, through a web page. This is a feature of the browser. Digi International, NET+OS and the advanced web server have no control over the operation of the browser. Digi International knows of no workarounds for this change in behavior.

Additionally, users and developers were used to writing JavaScript functions that would manipulate the file name and/or path as entered by the user in the file browser object. This form element (the file name) is no longer alterable via JavaScript.

The bottom line is that as you and your customer's transition to MS IE8, some applications might stop working while others may work differently than they do using earlier versions of MS IE.

5 Conclusion

With the roll out of Microsoft Internet Explorer V8 (MS IE 8), some NET+OS developers developing sophisticated advanced web server applications will see features lost. These features include the ability to type file names and paths directly into the file browser of an html form, and the ability to manipulate the filename supplied by a file browser using JavaScript.

Since this loss of behavior is related to functions within MS IE8, DIGI has no control over this behavior. Developers should take the information supplied in this document into consideration before deciding on the final functionality of their web-based applications.