

# [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-03/0083.html>

---

*From:* [support@securiteam.com](mailto:support@securiteam.com)

*Date:* 03/18/02

From: [support@securiteam.com](mailto:support@securiteam.com)

To: [list@securiteam.com](mailto:list@securiteam.com)

Date: Mon, 18 Mar 2002 13:45:50 +0100 (CET)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> -- Know that you're safe.

-----

VBScript Handling in IE can Allow Web Pages to Read Local Files

---

## SUMMARY

Frames are used in Internet Explorer to provide for a fuller browsing experience. By design, scripts in the frame of one site or domain should be prohibited from accessing the content of frames in another site or domain. However, a flaw exists in how VBScript is handled in IE relating to validating cross-domain access. This flaw can allow scripts of one domain to access the contents of another domain in a frame.

A malicious user could exploit this vulnerability by using scripting to extract the contents of frames in other domains, then sending that content back to their web site. This would enable the attacker to view files on the user's local machine or capture the contents of third-party web sites the user visited after leaving the attacker's site. The latter scenario could, in the worst case, enable the attacker to learn personal information like user names, passwords, or credit card information.

In both cases, the user would either have to go to a site under the attacker's control or view an HTML email sent by the attacker. In addition, the attacker would have to know the exact name and location of any files on the user's system. Further, the attacker could only gain access to files that can be displayed in a browser window, such as text

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

files, HTML files, or image files.

### DETAILS

Vulnerable systems:

- Microsoft Internet Explorer 5.01
- Microsoft Internet Explorer 5.5
- Microsoft Internet Explorer 6.0

Mitigating factors:

- The vulnerability could only be used to view files. It could not be used to create, delete, modify or execute them.
- The vulnerability would only allow an attacker to read files that can be opened in a browser window, such as image files, HTML files and text files. Other file types, such as binary files, executable files, Word documents, and so forth, could not be read.
- The attacker would need to specify the exact name and location of the file in order to read it.
- The email-borne attack scenario would be blocked if the user were using any of the following: Outlook 98 or 2000 with the Outlook Email Security Update installed; Outlook 2002; or Outlook Express 6.

Patch availability

Download locations for this patch

<<http://www.microsoft.com/windows/ie/downloads/critical/q318089/default.asp>>  
<http://www.microsoft.com/windows/ie/downloads/critical/q318089/default.asp>  
<<http://www.microsoft.com/Windowsupdate>>  
<http://www.microsoft.com/Windowsupdate>

What's the scope of the vulnerability?

This is an information disclosure vulnerability. It could allow a malicious web site operator to view files on the local computer of a visiting user. In addition, it could allow a malicious site operator to collect information from a user's browsing session after he had left the malicious site. This information could then be passed back to the malicious site and could include personal information such as usernames, passwords, or credit card information.

In both cases, the malicious user would have to entice the intended victim to a web site under her control. To read information on the user's local machine, the malicious site operator would have to know the exact name and location of any file on the user's computer. The vulnerability would not allow an attacker to add, change or delete files on the user's computer.

What causes the vulnerability?

The vulnerability results because of a flaw in the handling of scripts across domains within frames. The flaw allows script to violate IE's Cross-Domain Security Model in a way that would enable a web site to read

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

data in a frame belonging to another domain.

What are scripts?

Scripts are used to allow web developers to manipulate the items on a web page. Common uses of scripts on a web page are validating user input, working with controls on a page, and communicating with the user.

By default, Internet Explorer supports two scripting languages VBScript and JScript. Web site developers can use either of these programming languages on their sites.

In addition, developers and site operators can choose to support other third-party scripting languages. These must be installed on the client system to run successfully, however.

What are frames?

A frame is a sub-window of the main browser window. For example, you can use frames to divide the browser window into a table of contents on the left hand side, and a page display on the right hand side.

From the perspective of the software, however, each frame is a separate window and is independent of any other windows. This means, for example, that three frames in a browser can show content from three different sites, content from three different parts of the same site, or some combination of content from the same and different sites.

What is IE's Cross-Domain Security Model?

Since each frame is actually an independent window, the concept of "domains" was developed to allow frames that part of the same web site to be treated as a logical whole. For example, if a browser displays a page from [www.microsoft.com](http://www.microsoft.com) in one frame, and a page from [www.microsoft.com/security](http://www.microsoft.com/security) in another frame, they are reckoned as part of the same domain. Alternately, if a browser displays a page from [www.microsoft.com](http://www.microsoft.com) in one frame, and a page from another web site in another frame, they would be reckoned as being in different domains. This domain is then used as a security boundary, isolating content from unrelated sites from each other, and grouping content from the same site together.

This domain security model is used to enforce security on scripting within frames. By design, scripts should be able to execute on frames within the same domain. This allows, for example, a click button in a table of contents frame to manipulate the display text from the same site in another frame. Additionally, by design, scripts should not be able to manipulate content in frames from other domains.

What's wrong with how script is handles across domains?

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

There is a flaw in how domain boundaries are calculated. Because of this flaw, frames that are in different domains can be incorrectly reckoned to be part of the same domain. This could make it possible for scripts to take actions on frames outside of their domain.

How could an attacker exploit this vulnerability?

An attacker could attempt to exploit this vulnerability by constructing a web page that would exploit the vulnerability. The attacker could then either post this web page on a server under their control or send it via email to the user.

Why would an attacker be able to exploit this via HTML email?

HTML mails are essentially web pages that are sent by mail. By creating a web page that exploits the vulnerability, and then sending it as an HTML mail, an attacker could mount essentially the same attack as by a web site. If scripting were enabled for HTML Email, when the mail was opened, either by double-clicking the message or viewing it in a preview pane, the script would execute.

I'm using one of the email products you listed above. Does this mean I don't need the patch? The Outlook Email Security Update, Outlook 2002, and Outlook Express 6 will protect you against the mail-borne attack scenario. However, we still recommend that you install the patch, to ensure that you're protected against the web-based scenario.

What could this vulnerability enable an attacker to do?

This vulnerability could enable an attacker to manipulate the contents of other frames in other domains and attempt to pull information from another site's frame into its own.

In plain terms, this means that they could view an HTML page, either on a web site or mailed to them. It could then read information from other frames, including one opened on the user's local computer and send that information back to their site. This would happen until the user either closed the browser or the HTML email.

Could this vulnerability be exploited accidentally?

No. The steps that a web site would need to take in order to exploit this vulnerability are extremely unlikely to be useful for any legitimate purpose.

How likely am I to be affected by this vulnerability?

It depends in large part on your browsing habits. Since exploiting this vulnerability requires that the attacker lure the potential victim to a website under their control, users who visit familiar, professionally-operated sites most likely face less risk than those hot

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

regularly go to unknown web sites.

Security Zones are a very good way to manage risk based on browsing habits, and we recommend that customers consider using them regularly to differentiate between well-known, trusted sites, and unknown, untrusted sites.

This sounds a lot like a variant of the Frame Domain Verification vulnerability, is it the same thing?

It is similar, but slightly different. The important difference between these two issues is in the location of the flaw that causes each. The "Frame Domain Verification" vulnerability is caused by a flaw in IE. In contrast, this vulnerability is a result of how VBScript is handled in IE.

You said it's a result of how VBScript is handled. Does this mean that JScript is not affected by this vulnerability?

Correct. JScript is not affected by this vulnerability.

What about third-party scripting languages, are those vulnerable too?

Possibly. IE does support the use of Python, Perl and other third-party scripting languages, if such languages have been installed by the user. Depending on how those languages are implemented, and how they handle certain domain security checks, they could be affected.

Does the patch address the vulnerability for third-party scripting languages?

No. An architectural change is being made in a future service pack of IE that will ensure that this cannot be an issue for third-party scripting languages.

How do I know what version of VBScript I have?

VBScript.dll file ships with two software products, Internet Explorer and Microsoft Windows Script.

IE 6.0: Any customer running IE 6.0, regardless of platform, will have Windows Script 5.6 installed by default. IE 6.0 ships with Windows Script 5.6.

IE 5.5: Any customer running IE 5.5, regardless of platform, will have Windows Script 5.5 installed by default. IE 5.5 ships with Windows Script 5.5.

IE 5.01: Any customer running IE 5.01, regardless of platform, will have Windows Script 5.1 installed by default.

Customers who have not upgraded their versions of Internet Explorer to 6.0 or 5.5 are most likely running the following versions of Windows Script:

Windows 2000: Windows Script 5.1

Win ME: Windows Script 5.5

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

How can I be sure of the version I'm running?

You can verify the version of VBScript you're running by checking the version of VBScript.dll, that resides in directory System32 of the Windows directory, by right clicking on it in Explorer and choosing "Properties".

I've upgraded from the default version of VBScript. What patch do I apply?

If you've upgraded from the default version of VBScript, you should apply the patch version that corresponds to your installed version.

Customers with VBScript 5.6 should install the patch available for IE 6.0.

Customers with VBScript 5.5 should install the patch available for IE 5.5.

Customers with VBScript 5.1 should install the patch available for IE 5.01.

I'm confused. Shouldn't I install a patch based on my version of IE?

In almost all cases, you will want to install a patch based on the version of IE. VBScript ships with IE and the versions of VBScript and IE, by default, are related.

However, if you've upgraded your version of VBScript manually, the versions of VBScript and IE no longer match.

Since most customers do not upgrade VBScript manually, we have labelled the patches based on the default IE version to make it easier for most customers to identify the patch they need to apply.

What does the patch do?

The patch corrects the vulnerability by instituting domain verification handling for VBScript

### ADDITIONAL INFORMATION

Microsoft thanks Zentai Peter Aron, Ivy Hungary Ltd ( <<http://w3.ivy.hu/>> <http://w3.ivy.hu/>) for reporting this issue to us and working with us to protect customers.

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

[list-unsubscribe@securiteam.com](mailto:list-unsubscribe@securiteam.com)

In order to subscribe to the mailing list, simply forward this email to: [list-subscribe@securiteam.com](mailto:list-subscribe@securiteam.com)

=====

=====

## Securiteam: [NT] VBScript Handling in IE can Allow Web Pages to Read Local Files

### DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

---

- **Previous message:** [support@securiteam.com](mailto:support@securiteam.com): "[NT] BitVise WinSSH Denial of Service"
- **Messages sorted by:** [\[ date \]](#) [\[ thread \]](#) [\[ subject \]](#) [\[ author \]](#) [\[ attachment \]](#)