

# [NT] Cumulative Patch for Internet Information Service (28 May 2003)

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-05/0057.html>

---

*From:* SecuriTeam ([support\\_at\\_securiteam.com](mailto:support_at_securiteam.com))

*Date:* 05/29/03

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

Date: 29 May 2003 16:04:47 +0200

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

-- promotion

In the US?

Contact Beyond Security at our new California office housewarming rates on automated network vulnerability scanning. We also welcome ISPs and other resellers!

Please contact us at: 323-882-8286 or [ussales@beyondsecurity.com](mailto:ussales@beyondsecurity.com)

-----

Cumulative Patch for Internet Information Service (28 May 2003)

---

## SUMMARY

This patch is a cumulative patch that includes the functionality of all security patches released for IIS 4.0 since Windows NT 4.0 Service Pack 6a, and all security patches released to date for IIS 5.0 since Windows 2000 Service Pack 2 and IIS 5.1.

In addition to all previously released security patches, this patch also includes fixes for the following newly discovered security vulnerabilities affecting IIS 4.0, 5.0 and 5.1:

- \* A Cross-Site Scripting (CSS) vulnerability affecting IIS 4.0, 5.0 and 5.1 involving the error message that's returned to advise that a requested URL has been redirected. An attacker who was able to lure a user into clicking a link on his or her web site could relay a request containing script to a third-party web site running IIS, thereby causing the third-party site's response (still including the script) to be sent to the user. The script would then render using the security settings of the third-party site rather than the attacker's.

## Securiteam: [NT] Cumulative Patch for Internet Information Service (28 May 2003)

\* A buffer overrun that results because IIS 5.0 does not correctly validate requests for certain types of web pages known as server side includes. An attacker would need the ability to upload a Server-side include page to a vulnerable IIS server. If the attacker then requested this page, a buffer overrun could result, which would allow the attacker to execute code of their choice on the server with user-level permissions.

\* A denial of service vulnerability that results because of a flaw in the way IIS 4.0 and 5.0 allocate memory requests when constructing headers to be returned to a web client. An attacker would need the ability to upload an ASP page to a vulnerable IIS server. This ASP page, when called by the attacker, would attempt to return an extremely large header to the calling web client. Because IIS does not limit the amount of memory that can be used in this case, this could cause IIS to fail because of running out of local memory.

\* A denial of service vulnerability that results because IIS 5.0 and 5.1 do not correctly handle an error condition when an overly long WebDAV request is passed to them. As a result, an attacker could cause IIS to fail – however, both IIS 5.0 and 5.1 will by default restart immediately after this failure.

\* There is a dependency associated with this patch – it requires the patch from Microsoft Security Bulletin MS02-050 to be installed. If this patch is installed and MS02-050 is not present, client side certificates will be rejected. This functionality can be restored by installing the MS02-050 patch.

### DETAILS

#### Affected Software:

- \* Microsoft Internet Information Server 4.0
- \* Microsoft Internet Information Services 5.0
- \* Microsoft Internet Information Services 5.1

#### Non Affected Software:

- \* Microsoft Internet Information Services 6.0

#### Mitigating factors:

##### Redirection Cross Site Scripting:

- \* IIS 6.0 is not affected.

\* The vulnerability could only be exploited if the attacker could entice another user into visiting a web page and clicking a link on it, or opening an HTML mail.

\* The target page must be an ASP page, which uses Response.Redirect to redirect the client, to a new URL that is based on the incoming URL of current request.

## Securiteam: [NT] Cumulative Patch for Internet Information Service (28 May 2003)

### Server Side Include Web Pages Buffer Overrun

\* IIS 4.0, IIS 5.1 and IIS 6.0 are not affected.

\* The IIS Lockdown tool by default disables the ssinc.dll mapping, which will block this attack.

\* By default IIS 5.0 runs under a user account and not the system account. Therefore, an attacker who successfully exploited the vulnerability would only gain user level permissions rather than administrative level permissions.

\* An attacker must have the ability to upload files to the IIS Server.

### ASP Headers Denial of Service

\* An attacker must have the ability to upload files to the IIS server.

\* IIS 5.0 will automatically restart after failing.

\* IIS 5.1 and IIS 6.0 are not affected.

### WebDAV Denial of Service

\* IIS 6.0 is not affected.

\* IIS 5.0 and 5.1 will restart automatically after this failure.

\* The IIS Lockdown tool disables WebDAV by default, which will block this attack.

### What versions of Windows does Internet Information Services 6 ship with?

IIS 6 ships with Windows Server 2003. It is not affected by any of the vulnerabilities described in this security bulletin.

### Patch availability:

#### Download locations for this patch

\* IIS 4.0:

<http://microsoft.com/downloads/details.aspx?FamilyId=1DBC1914-98E9-4DED-ADBF-E9B374A1F79D&displaylang=en>  
All

\* IIS 5.0:

<http://microsoft.com/downloads/details.aspx?FamilyId=2F5D9852-4ADD-44F8-8715-AC3D7D7D94BF&displaylang=en>  
All

\* IIS 5.1:

<http://microsoft.com/downloads/details.aspx?FamilyId=77CFE3EF-C5C5-401C-BC12-9F08154A5007&displaylang=en>  
32-bit Edition,  
<http://microsoft.com/downloads/details.aspx?FamilyId=86F4407E-B9BF-4490-9421-008407578D11&displaylang=en>  
64-bit Edition

### Redirection Cross Site Scripting

What's the scope of this vulnerability?

This cross-site scripting vulnerability could allow an attacker to send a request to an affected server that would cause a web page containing script to be sent to another user. The script would execute within the user's browser as though it had come from the third-party site. This would let it run using the security settings appropriate to the third-part web site, as well as allowing the attacker to access any data belonging to the site. The vulnerability could only be exploited if the user opened an HTML mail or visited a malicious user's web site – the code could not be "injected" into an existing session.

What is redirection?

Redirection happens when a web browser makes a request for a web page that does not exist and the web server redirects the browser to another page such as a generic error page or the website's homepage. For example, the webpage <http://microsoft.com/xp> does not exist, but instead of providing an error, the web server redirects the browser to a page that suggests pages that the user may have been looking for as well as providing a site map. This process is redirection.

What's Cross-Site Scripting?

CSS is a security vulnerability that potentially enables a malicious user to "inject" code into a user's session with a web site. Unlike most security vulnerabilities, CSS does not apply to any single vendor's products – instead, it can affect any software that runs on a web server and does not follow defensive programming practices

How does CSS work?

A good description is available in the form of an executive summary and a FAQ. However, at a high level of detail, here is how CSS works. Suppose Web Site A offers a search feature that lets a user type a word or phrase to search for. If the user typed "banana" in as the search phrase, the site would search for the phrase, then generate a web page saying, "I'm sorry, but I can't find the word 'banana'". It would send the web page to his browser, which would then parse the page and display it. Now suppose that, instead of entering "banana" as the search phrase, the user entered something like "banana < SCRIPT> Alert('Hello'); </ SCRIPT>". If the search feature were written to blindly use whatever search phrase it's provided, it would search for the entire string, and create a web page saying "I'm sorry, but I can't find the word "banana < SCRIPT> Alert('Hello'); < /SCRIPT>". However, all of the text beginning with < SCRIPT> and ending with </ SCRIPT> is actually program code, so when the page was processed, a dialogue box would be displayed by the user's browser, saying "Hello".

So far, this example has only shown how a user could "relay" code off a web server and make it run on his own machine. That is not a security vulnerability. However, it is possible for a malicious web site operator to invoke this vulnerability to run on the computer of a user who visits his site. If Web Site B were operated by a malicious user who was able to entice the user into visiting it and clicking a hyperlink, Web Site B could go to Web Site A, fill in the search page with malicious script, and

submit it on behalf of the user. The resulting page would return to the user (since the user, having clicked on the hyperlink, was ultimately the requester), and process on the user's machine.

What could the script do on the user's machine?

The script from Web Site B (the attacker's site) would run on the user's machine as though it had come from Web Site A. In practical terms, this would mean two things: It would run using the security settings on the user's machine that were appropriate to Web Site A.

The script from Web Site B would be able to access cookies and any other data on the user's system that belonged to Web Site A.

What causes the vulnerability?

The vulnerability results because the ASP function responsible for redirection displays the URL in HTML text without proper encoding.

What's wrong with IIS Redirection?

The ASP function responsible for redirection does not correctly encode the URL for displaying in HTML text. As a result, it is possible to embed script in a redirection request and cause this to be returned to the web browser.

What would this vulnerability enable an attacker to do?

The vulnerability would allow an attacker who operated a web site and was able to lure another user into clicking a link on it to carry out a cross-site scripting attack via another web site that was running IIS. As discussed above, this would enable the attacker to run script in the user's browser using the security settings of the other web site (the one running IIS), and to access cookies and other data belonging to it. However, most browsers will automatically follow the redirection response header and skip the HTML text. The client is not vulnerable in this case.

How could an attacker exploit this vulnerability?

To exploit the vulnerability an attacker would need to host a webpage and lure a user to click on a link in that page, or would need to send the user a URL in an e-mail. This URL would need to contain script. It would also need to point to a web page on the vulnerable IIS Server that did not exist – when the IIS redirection function handled the redirection of the non-existent page, it would pass the attacker's script back to the browser.

You said that the point of the attack would be for the attacker to get script running in the user's browser using the security settings of my web site. What specific capabilities would the attacker gain by doing this? It would vary from site to site, based on what Security Zone the attacker's site and yours were placed in.

If they were both in the same zone (and by default, all web sites reside in the Internet Zone unless the user moves them), they would both be subject to exactly the same security restrictions and the attacker would

gain nothing through the vulnerability.

If the user had put the attacker's site into a more restricted zone than yours, the attacker would gain the ability for his script to do anything on the user's computer that script from your site could do.

If the user had put your site into a more restricted zone than yours, the attacker would actually lose capabilities through the attack.

It is important to note, however, that regardless of the security settings, the attacker's script would always be able to access cookies and any other data on the user's system belonging to the third-party site. This is because, as far as the browser can tell, the attacker is the third-party site.

How does the patch eliminate the vulnerability?

The patch eliminates the vulnerability by ensuring script is not passed during an IIS redirection request.

SSINC Buffer Overrun

What's the scope of this vulnerability?

This is a buffer overrun vulnerability. It could allow an attacker to execute code of their choice with user-level permissions on the IIS Server.

What causes the vulnerability?

The vulnerability results because IIS does not correctly check parameters when responding to requests for Server-side Include (SSINC) pages such as shtml, .stm and .shtm files.

What's wrong with the way IIS responds to requests for static web pages?

There is a flaw in the component responsible for serving static web pages. The component does not correctly validate requests passed to it and as a result, a buffer-overrun condition occurs when overly long requests are passed to it.

What could this vulnerability enable an attacker to do?

This vulnerability could enable an attacker to execute code of their choice with user-level privileges on the IIS Server. However to do so, an attacker would need to be able to first upload SSINC web pages to the IIS Server.

Does the IIS Lockdown Tool block this attack?

Yes – By default, the IIS Lockdown tool will remove the SSINC script map.

What is the significance of an attacker only gaining user-level permissions from this attack?

By default, the affected component operates under a user account and not the system account. This user account has far less privileges on the server than the system account – for example, a user account cannot add or remove other user accounts or restart services.

How could an attacker exploit this vulnerability?

An attacker could seek to exploit this vulnerability by uploading a specially named SHTML web page to the IIS Server – the attacker would need explicit permissions to do this. The attacker would also need to have an understanding of the directory structure on the web server. If the attacker then requested this webpage, a buffer overrun would occur that could allow him or her to execute code of their choice in the context of the IIS User account.

What does the patch do?

The patch eliminates the vulnerability by ensuring that the affected IIS component correctly validates input passed to it.

ASP Headers Denial of Service

What's the scope of this vulnerability?

This is a denial of service vulnerability that could allow an attacker to cause IIS to fail. IIS 5.0 would restart automatically; IIS 4.0 would need to be restarted manually.

What causes the vulnerability?

The vulnerability results because the ASP function Response.AddHeader does not place a limit on the size of the header that is returned to a browser. As a result, it is possible for an maliciously crafted ASP page to generate an overly large header that exceeds the memory available to IIS, causing it to fail.

What's wrong with the way headers are generated by IIS?

The flaw is not in the way IIS actually generates headers, but in the fact that it does not place a limit on the size of the header that can be generated. As a result, it is possible to cause a header to be generated that is so large that it exhausts the memory available to IIS, causing it to fail.

What could this vulnerability allow an attacker to do?

This could allow an attacker to cause IIS to fail and therefore stop serving web pages. It is worth noting that IIS 5.0 would automatically restart, so the Denial of Service would be temporary. IIS 4.0 would require manual restarting.

How could an attacker exploit this vulnerability?

An attacker would need to be able to upload a malicious ASP file to the IIS Server – this malicious ASP would contain code that would cause an overly large header to be generated when the page was requested. If the attacker then request the page, the code would execute, which could cause IIS to fail as a result of excessive memory being required to complete the request.

It should be noted that an attacker must have permissions to upload ASP files to the server in order for him or her to carry out an attack based on this vulnerability.

Securiteam: [NT] Cumulative Patch for Internet Information Service (28 May 2003)

What does the patch do?

The patch places a restriction on the size of the return header that can be generated.

WebDAV Denial of Service

What's the scope of this vulnerability?

This is a denial of service. It could allow an attacker to cause a temporary denial of service in IIS 5.0 and 5.1.

What causes the vulnerability?

The vulnerability is caused by a flaw in the way overly long WebDAV requests that contain XML commands are handled. The flaw results because it is possible for the error handling sequence to get out of order when handling a particular type of XML error, which causes IIS to fail.

What's wrong with the WebDAV errors are handled?

It's possible for an overly long WebDAV request to cause the error handling for XML requests to get out of sequence. This causes IIS to fail, however both IIS 5.0 and 5.1 will automatically restart.

What could this vulnerability allow an attacker to do?

This could allow an attacker to cause IIS 5.0 or 5.1 to fail and therefore stop serving web pages. Both IIS 5.0 and 5.1 would automatically restart.

How could an attacker exploit this vulnerability?

An attacker could exploit this vulnerability by sending an overly long WebDAV request that contained malformed XML data to an IIS 5.0 or 5.1 web server. This could cause the error handling for the malformed XML to become out of sequence, causing IIS to fail.

What does the patch do?

The patch corrects the vulnerability by enforcing the correct error handling sequence when handling malformed XML data.

Does this patch have any dependencies on other patches?

Yes – it requires the patch from Microsoft Security Bulletin MS02-050 to be installed. If this IIS cumulative patch is installed and MS02-050 is not present, client side certificates will be rejected. This functionality can be restored by installing the MS02-050 patch either before or after installing the IIS Cumulative patch.

ADDITIONAL INFORMATION

The information has been provided by

<mailto:0\_48425\_E51E4D7D-DECD-43AE-9A29-36080E8D4C3C\_US@Newsletters.Microsoft.com>  
Microsoft Product Security.

=====

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:

Securiteam: [NT] Cumulative Patch for Internet Information Service (28 May 2003)

[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)

=====  
=====

**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.