

[NT] 28 March 2002 Cumulative Patch for Internet Explorer

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-03/0131.html>

From: support@securiteam.com

Date: 03/31/02

From: support@securiteam.com

To: list@securiteam.com

Date: Sun, 31 Mar 2002 18:33:50 +0200 (CEST)

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.

28 March 2002 Cumulative Patch for Internet Explorer

SUMMARY

This is a cumulative patch that includes the functionality of all previously released patches for IE 5.01, 5.5 and IE 6. In addition, it eliminates the following two newly discovered vulnerabilities:

* A vulnerability in the zone determination function that could allow a script embedded in a cookie to be run in the Local Computer zone. While HTML scripts can be stored in cookies, they should be handled in the same zone as the hosting site associated with them, in most cases the Internet zone. An attacker could place script in a cookie that would be saved to the user's hard disk. When the cookie was opened by the site, the script would then run in the Local Computer zone, allowing it to run with fewer restrictions than it would otherwise have.

* A vulnerability in the handling of object tags that could allow an attacker to invoke an executable already present on the user's machine. A malicious user could create HTML web page that includes this object tag and cause a local program to run on the victim's machine.

DETAILS

Securiteam: [NT] 28 March 2002 Cumulative Patch for Internet Explorer

Affected Software:

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

Mitigating factors:

Cookie-based Script Execution:

* The script would run with the same rights as the user. The specific privileges the attacker could gain through this vulnerability would therefore depend on the privileges accorded to the user. Any limitations on a user's account, such as those applied through Group Policies, would also limit the actions of any script executed by this vulnerability.

Local Executable Invocation via Object tag:

* The vulnerability would not enable the attacker to pass any parameters to the program. Microsoft is not aware of any programs installed by default in any version of Windows that, when called with no parameters, could be used to compromise the system.

* An attacker could only execute a file on the victim's local machine.

The vulnerability could not be used to execute a program on a remote share or web site.

* The vulnerability would not provide any way for an attacker to put a program of his choice onto another user's system.

* An attacker would need to know the name and location of any executable on the system to successfully invoke it.

* Outlook 98 and 2000 (after installing the Outlook Email Security Update), Outlook 2002, and Outlook Express 6 all open HTML mail in the Restricted Sites Zone. As a result, customers using these products would not be at risk from email-borne attacks.

Patch availability:

Download locations for this patch:

<http://www.microsoft.com/windows/ie/downloads/critical/Q319182/default.asp>
<http://www.microsoft.com/windows/ie/downloads/critical/Q319182/default.asp>

What vulnerabilities are eliminated by this patch?

This is a cumulative patch that, when applied, eliminates all previously addressed security vulnerabilities affecting Internet Explorer 5.01, 5.5 and 6.0. In addition to eliminating all previously discussed vulnerabilities versions, it also eliminates two new ones:

* A vulnerability that could allow an attacker to cause script embedded in a cookie to execute in the Local Computer Zone.

* A vulnerability that could allow an attacker to invoke an executable already present on the user's system.

I notice that this particular IE cumulative patch supports IE 5.01 SP2 on Windows NT 4.0 SP6a. The other cumulative patch didn't support this version of IE, why is Microsoft supporting it now?

Based on feedback from customers and their needs, Microsoft will provide support for IE 5.01 on Windows NT 4.0 for this release of the IE

cumulative patch. This support is slated to be discontinued in June 2002.

Cookie-based Script Execution (CVE-CAN-2002-0078)

What's the scope of first vulnerability?

This is an elevation of privilege vulnerability. An attacker who was able to successfully exploit this vulnerability would be able to cause HTML scripts on a web site to execute as if they were run locally on the user's system. This could allow the scripts to run outside of the constraints usually imposed on web site scripts. The scripts could then take any action on the system as if they were the user.

The attacker's actions would be limited by any restrictions that govern the user's actions. Thus, in an environment where accounts adhere to the rule of least privilege, the attacker might be significantly limited in the actions his program could take.

What causes the vulnerability?

The vulnerability results because of a flaw in how IE determines the correct security zone handling for scripts embedded in cookies. Specifically, it incorrectly treats scripts embedded in cookies as if they should be run in the Local Computer zone, rather than the same zone as the web site with which the cookie is associated.

What is a cookie?

Cookies are small data files that web servers use to store and retrieve information that is useful throughout a session. For example, suppose a web site provided a way for visitors to obtain weather reports for different parts of the country. The site might allow visitors to customize the site so that the weather report for their preferred city would be presented every time they visited the site. This customization would be accomplished by storing the users' preferences in a cookie that would be read by the site each time the user visited and be used to determine and present the right customized information.

Because each site's customization needs are different, cookies are designed to be flexible and let the site decide what kind of information is stored and the way that data is stored. For security of data each site can only access information in it's own cookie.

Why can script be embedded in a cookie?

As noted, cookies are a free-form means of data storage. By design, it is left to the web site to determine what information to store in a cookie and how to store it. Because of this, a site can choose to store any information in any way in a cookie, including HTML scripting information.

Because scripts contained in cookies are in essence an extension of the web site itself, the same security zone that applies to a web site, should also apply to its cookie.

What are security zones?

IE Security Zones are a system that divides online content into

categories, or zones based on its trustworthiness. Specific web sites can be assigned to a zone, depending on how much you trust the content of each site. The zone then restricts the capabilities of the web content, based on the zone's settings.

By default, most Internet sites are reckoned as part of the Internet zone, which has settings that prevent scripts and other active code from accessing resources on the local machine. Conversely, the Local Computer zone is a much less restricted zone that allows content to access and manipulate content on the local system. By default, files stored on the local computer are run in the Local Computer zone.

What's wrong with how IE handles script embedded in a cookie?

As we noted above, script in a cookie should be handled within the Security Zone associated with the web site that owns the cookie. However, because cookies are stored locally, IE incorrectly reckons that script within cookies should be governed by the Local Computer Zone, rather than the Internet Zone.

What would this enable an attacker to do?

An attacker could attempt to exploit this vulnerability and cause script to execute as if it were run on the local system rather than from a remote web site. Because the script would be run in the Local Computer zone, which is less restrictive than the zones that govern remote sites, the script would run with few limitations. The script would run as if it were run by the user. This could allow the script to access local resources on the user's computer. The script would then be able to take actions on the local system as if it were the user herself, including adding, changing, or deleting data or configuration information.

How might an attacker exploit this vulnerability?

An attacker seeking to exploit this vulnerability would have to build a specially formed cookie with scripting embedded within it. He would then either post it on a web site under his control or attempt to entice a user to visit his site, or send the web page as an HTML email message to the user. In both cases, once the page was loaded and read the cookie, the script would execute.

Would disabling cookies stop an attacker?

Yes, disabling cookies would stop an attempt to exploit this vulnerability by preventing the loading of cookies. Since the vulnerability requires the cookie to be loaded on the local system, this setting blocks this attack vector.

How does the patch eliminate this vulnerability?

The patch eliminates the vulnerability by having the zone determination correctly assign cookies to the same zone as the originating page. Thus, if a page is in the Internet zone, any script in a cookie offered by the site will be handled in the Internet zone.

Local Executable Invocation via Object tag (CVE-CAN-2002-007)

What's the scope of the second vulnerability?

This vulnerability could allow an attacker to invoke an executable already present on the user's machine. The attacker could create a specially formatted web page that exploits this vulnerability and either post it on a site under his control, or send it by email to the user.

The vulnerability does not provide a way for an attacker to deliver a program of his choice to the system; the program invoked must exist on the system for the attacker to invoke it. In addition, no parameters can be sent to the invoked application. Thus, the extent to which an attacker could exploit this is limited to simply invoking program already present on the system.

What causes the vulnerability?

The vulnerability results from a flaw in how IE applies security zones to objects invoked on an HTML page with the codebase property. In certain instances, IE incorrectly reckons these objects as being part of the Local Computer zone, even though the page itself is in a different zone, such as the Internet zone. Because the Local Computer zone is less restrictive than other zones, this can allow the web page to run executables on the local system without prompting.

What is the Codebase property?

The CODEBASE property is an HTML standard that allows a web page to specify a location for downloading of helper applications that may not be present on the local system. For example, if a web page designer wanted to be able to invoke the Common Control dialog in his web page, to give the page the same "look and feel" of the Windows explorer, he can specify download locations for that object in case it's not present on the local system already.

What's wrong with how the Codebase property is handled by IE Zones?

When the codebase property is used in conjunction with a resource on the local computer, it is incorrectly reckoned as part of the Local Computer zone, rather than the zone of the web page. If you set the CODEBASE property to the file path of a local executable, that executable would be run in the Local Computer zone, even though the page invoking it is in the Internet zone. This means that the executable would be run without any prompting to the user. It also means that the executable could be run with the same privileges as the user.

What would this enable an attacker to do?

An attacker could use this vulnerability to invoke an executable that is already present on the local system. For example, an attacker could call cmd.exe, the command shell, and make a command window appear. However, the vulnerability would not enable the attacker to pass any parameters to the program. This turns out to be a significant mitigating item.

You said the inability to pass parameters was a significant mitigating item. Why?

Parameters are pieces of information that are provided to a program when it's executed, and tell the program what action to perform, or where and how to perform it. For example, consider the program that opens a command prompt, cmd.exe. If it is called with no parameters (e.g., "cmd.exe"), it simply opens a command prompt. However, if it is called with another program's name as a parameter (e.g., "cmd.exe ver.exe"), it will run the second program at the command prompt. In the case of our example, it would run the Ver.exe program, which shows what version of Windows you are running.

The vulnerability does let an attacker run a program, but it does not provide a way for the attacker to pass parameters to it. Thus, the attacker could run cmd.exe and open a command prompt, but could not make anything happen at the prompt. The vulnerability could only be used for destructive purposes if there was a program on the user's system that was dangerous when called without parameters. However, Microsoft has checked the list of programs that ship by default with Windows but found none that are dangerous when called without parameters.

How might an attacker exploit this vulnerability?

An attacker could create a web page that exploits the vulnerability and host it on a malicious web site. If a user visited the site and opened the web page, the page could use the vulnerability to run a program on the user's local machine. An attacker could also send the malicious web page to a user as an HTML mail. If the recipient opened the mail, it would likewise attempt to exploit the vulnerability and run a program.

Can an attacker run any program on my computer?

An attacker could run any program already present on the other user's local computer; however, he or she would need to know its name and location on the computer. Some programs have well-known and predictable names and locations, but others do not.

Could an attacker use this vulnerability to load a program on my machine from their web site or server?

No. An attacker could only use this vulnerability to invoke an executable that is already present on the user's system. It provides no way for an attacker to transfer a program of his choice to the user's system.

I heard that an attacker could forcibly log me off my computer using this vulnerability. Is this true?

Among the programs that ships with Windows is logoff.exe, which is designed to shut down the system and does not require any parameters. If an attacker exploited this vulnerability and chose to run logoff.exe, it would have the effect of shutting down the user's system. However, the user could resume normal operation by just restarting the computer, and could avoid future attacks by not returning to the web site that mounted the attack.

Can an attacker read any files through the vulnerability or see any of my personal information?

Securiteam: [NT] 28 March 2002 Cumulative Patch for Internet Explorer

Only if there were a program already present on the machine that would be able to do this – the vulnerability itself does not provide a any way to view data on the local system.

How does the patch eliminate this vulnerability?

The patch eliminates the vulnerability by handling script contained in cookies in the same zone as the originating web page.

ADDITIONAL INFORMATION

The information has been provided by

<mailto:0_28221_E51E4D7D-DECD-43AE-9A29-36080E8D4C3C_US@Newsletters.Microsoft.com>
Microsoft.

=====

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

In order to subscribe to the mailing list, simply forward this email to: 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.

-
- **Previous message:** support@securiteam.com: "[UNIX] WWWIsis Remote Command Execution and File Retrieval"
 - **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#) [\[attachment \]](#)