

## Vulnerability Advisory

<b>Name</b>	CHM File Heap Overflow
<b>Microsoft Advisory</b>	<a href="http://www.microsoft.com/technet/security/bulletin/MS04-023.msp">http://www.microsoft.com/technet/security/bulletin/MS04-023.msp</a>
<b>Date Released</b>	July 14, 2004
<b>Affected Software</b>	Microsoft Windows 98, 98SE, ME Microsoft Windows NT 4.0 Microsoft Windows 2000 Service Pack 4 Microsoft Windows XP, Microsoft Windows XP Service Pack 1 Microsoft Windows Server 2003
<b>Researcher</b>	Brett Moore <a href="mailto:brett.moore@security-assessment.com">brett.moore@security-assessment.com</a>

### Description

When thinking about buffer overflow vulnerabilities, a file can sometimes be as harmful as a packet. Even though past security issues have taught us that it is unwise to use an unvalidated value from a file/packet as a text length parameter, that is what happened here.

The HtmlHelp application (hh.exe) will read a value from a .CHM file and use this as the 'length' parameter in a REPZ MOVSD operation. By setting this to a large value, it is possible to overwrite sections of the heap with attacker supplied values.

This results in a typical win32 heap overflow landing either on the common `mov [ecx],eax / mov [eax+4],ecx` pair, or on a `call [eax+4]`. In either case the registers are under the control of the attacker leading to code execution.

### Details

When the corrupt file is opened an exception error will first occur at

```
0x78010044 REPZ MOVSD
```

The error has occurred because the destination address has reached the end of its allocated space. After clicking OK on the popup error box, execution will continue until it eventually reaches.

```
0x77fcc663 mov [ecx],eax  
0x77fcc665 mov [eax+4],ecx
```

At this time the EAX and ECX values have been filled with the data used to overwrite the heap, allowing an attacker to write an arbitrary value to a known place.

The corrupt file must be constructed in such a way to jump through some hoops first. It must pass some checks reliant on a value in the file that sets ESI.

- \* This value must be valid memory
- \* [ESI+1c] must be non null
- \* [ESI+24] must be null.

This value is simple to achieve resulting in a reliable heap exploit using any of the multiple methods now known to exploit heap overflows.

## Exploitation

Remote exploitation through Internet Explorer can be obtained through the use of the `window.showhelp()` function. Either using a public UNC share or through a 'coupled' browser exploit that saves the file to a known location before opening it. There may of course also be other ways of having a corrupt .CHM file loaded without requiring a user to download and run it, although a compiled help file may be easily accepted by a user anyway.

Automatic exploitation of browser based bugs, does not rely on an attacker sending a link, requiring the target user to click on it. Links, references and other objects can easily be opened through script code. And I am told that this can also be achieved without script code.

## Solution

Install the vendor supplied patch.

<http://www.microsoft.com/technet/security/bulletin/MS04-023.msp>

## About Security-Assessment.com

Security-Assessment.com is a leader in intrusion testing and security code review, and leads the world with SA-ISO, online ISO17799 compliance management solution. Security-Assessment.com is committed to security research and development, and its team have previously identified a number of vulnerabilities in public and private software vendors products.

For further information on this issue or any of our service offerings, contact us

Web [www.security-assessment.com](http://www.security-assessment.com)  
Email [info@security-assessment.com](mailto:info@security-assessment.com)  
Phone +649 302 5093