From: Philip Bogle
Sent: Thursday, April 24, 1997 6:27 PM
To: Michael Toutonghi
Subject: Trying to convince IE3 people to do a patch for "Safe for Scripting"

Importance: High

I spoke with Bob about the "Safe for Scripting" problems, and the fact that all of the improperly marked controls used the registry, not IObjectSafety. This suggests a fix for IE3 where you require IObjectSafety (which requires more consideration and work than the registry categories-- you actually have to read the docs and find out what "Safe for Scripting" means.)

I spoke with the IE3 people about this taking fix; apparently they'd previously rejected plans for an IE3 10, because they thought fixing MSN was enough, but will hopefully reconsider Anyway, they asked me to talk to David Cole and Chris Jones about this

-phil

-----Original Message-----
From: Philip Bogle
Sent: Thursday, April 24, 1997 6 23 PM
To: David Cole, Chris Jones, Hadi Partovi
Cc: Tony Ciccone, Robert Welland, Ben Slivka
Subject: Another ActiveX security hole
Importance: High

We desperately need to deliver a patch to IE3 that plugs the rash of security holes opened by controls incorrectly marked "Safe for Scripting". MSN was just one example, there are several others Fortunately, there's a fairly easy fix, which I urge you to take.

I realize that there are currently no plans for an IE3.10, but I urge you to reconsider.

To emphasize this point, I went yesterday to www.activex.com and looked at their *featured* control, called EasyMail There (for any hacker to read) was an explanation of the control's features, which included the ability to address messages, attach any file on the users machine, and send the message off, all without any user intervention.

For example, if an administrator on an NT machine browsed to a hostile page, that page could mail the password file to the hostile user as an attachment, to be cracked remotely using any of a number of tools availabe on the web

In a short time, I created a small demo of this control, which sends me a copy of your config.sys and autoexec bat when you visit the page. http://philbo/mailhack.htm

This example is an addition to my earlier one, which allows a hostile page to upload and execute arbitrary code on the browser's machine

nother security hole,
: using

We've got to do something about this for IE3, or we risk major embarassment. It took no cleverness at all for me to realize the control was dangerous and to create a page that exploits it, it's only a short matter of time before someone else does as well Furthermore, our accountability story falls totally to pieces. It's the page that's hostile, not the control (which is simply incompetent), so Authenticode is no use whatsoever.

The fix is fairly simple, but will require some coordination with control vendors All of the incorrectly marked controls are marked using the registry, not the IObjectSafety interface. This isn't coincidence, the registry marking is trivial and can be done without reading the documentation; whereas the interface requires that you actually read and think about the docs and what "Safe for Scripting" actually means.

So, we should get rid of the code that looks in the registry to see if controls are safe for scripting and require IObjectSafety. We should tell control vendors that they need to change their implementation if they were relying on the registry entries. To reduce the impact of this change, we can special case the CLSIDs for certain very popular controls that don't implement IObjectSafety, if those controls are in fact safe.

Ideally, we would also have some sort of review process for control authors where we would review the design of the control and make sure it is in fact safe for scripting. We wouldn't be guarding against fraud, but we would guard against basic incompetence, which is a useful thing in itself

I really think we need to take a stand and take some part in the certication of safe ActiveX controls . as I've often said, if Microsoft can't decide which ActiveX controls are safe, who can?

-phil

From: Philip Bogle
Sent: Thursday, April 10, 1997 5 42 PM
To: Ben Slivka; Robert Welland, Cornelius Willis, Bob Atkinson (Exchange)
Cc: Michael Toutonghi; Chris Jones
Subject: Another security hole, using well-intentioned third party control

Importance: High

I already mentioned the hazards of the signed Net-Installer control that you can find at www.activex com, which is incorrectly marked safe for initializing for untrusted data

I've created a nasty demo of the harm that can be done using the well-intentioned control at

file.//\\philbo\root\netinst\hack.htm

This page masquerades as the install page for the control and initializes the control with data that causes it to

- Set IE's security settings to none by munging the registry
- Install a Java class file to the trusted Java\Classes directory
- Install and run a program that causes the computer to restart

You might think that this is no different from the Internet Exploder control But there really is a fundamental difference The control author in this case had no malicious intent-- he just made an error in judgement that opened up the control to exploitation by other people Also, because of the way I've structured the page, I make it look like the control is to blame for anything bad that happens, rather than the page that misuses it Finally, the control is referenced by a well-known (and presumably trustworthy) repository of ActiveX controls.

If an outside party discovers this control, they can use it to support the argument that the responsibility that ActiveX forces on a control author is too high-- they'll say everyday developers shouldn't have to make decisions that could undermine the security of the whole system, that responsibility should be centralized in a sandbox system. I can't say I entirely disagree with them.

Also, this case raises troubling questions about how we fix the issue. Since the author had no ill-intent, it will look bad for Microsoft to lean on him with the lawyers. I forsee many similar situations in the future if we don't have a better policy for certifying or sandboxing controls.

-phil

---Original Message---
From: Philip Bogle
Sent: Thursday, April 10, 1997 10 29 AM
To: Philip Bogle
Subject: FW Serious security holes in ActiveX controls for MSN (and elsewhere)
Importance: High

From: Philip Bogle[FAX:+1 (206) 881-8095]
Sent: Thursday, April 10, 1997 6 20 AM
To: Robert Welland, Ben Slivka, Chris Jones
Cc: Michael Toutonghi, Cornelius Willis, Brad Schick, Larry Sullivan
Subject: Serious security holes in ActiveX controls for MSN (and elsewhere)
Importance: High

After hearing about the security hole in the Norton navigator control, I decided to review the security of other ActiveX controls likely to be preinstalled on users machine

What I found is pretty shocking-- developers often sign and publish inherently unsafe or incorrectly marked controls, and Microsoft is one of the worst offenders.

Consider MSN, for example. It preinstalls several powerful and unsafe controls, nonetheless marked "Safe for Scripting" I created several web pages (attached below) that use these controls to munge the registry and accomplish other evils . Regardless of security settings, the MSN user gets absolutely no warning or chance to reject the controls before the damage is done (because MSN preinstalls them.)

Even without scripting, many controls are inherently dangerous In a brief search of www.activex com, I found three signed controls with the ability to run arbitrary unsigned code on the users machine without requiring any scripting and without any warnings to users. I have attached descriptions of these controls below

Norton was not an isolated incident. We really have a general problem-- ActiveX control developers are behaving extremely naively with respect to security They are creating signed controls with automated capabilities far too dangerous for general internet use, or marking controls "Safe for Scripting" that clearly shouldn't be, despite the pains we took to explain the meaning of that concepts At this rate, ActiveX is going to gain an extremely bad reputation

Below are the descriptions of the insecure controls .

MSN Registry Control
The web page below uses the "MSN Registry Control" (!!!) to inspect and modify keys in the various MSN subtrees of the registry (It doesn"t allow arbitrary registry munging, but even the current capability is enough to cause serious trouble) The example I created displays and/or defetes the MSN registry entnes that list the URLs that the user has typed in
<< File· hack-msn >>
A script using the control could do many more evil things render components of MSN unusable by blowing away the appropnate registry entries, substitute a bogus stock server for the real one, fill up the users hard drive by creating registry keys ad infinitum, etc.

MSN Mail Control
This page uses the MSN mail control to obtain the user ID (which it could send back to the server) and to launch mail

<< File hack-mail >>
Active Data Connecter (possibly dangerous)
The Active Data Connector is marked "Safe for Scripting", and appears to allow a script on an untrusted page to connect to a database behind a firewall (as long as its not password protected) and siphon data out of it I didn't come up with an example page, but someone who knows more about the ADC should think carefully about this

Below are controls that are dangerous even without scripting, which I found on www.activex com.

DataRamp Assistant
"The DataRamp Assistant makes it easy to embed application references in World Wide Web pages. When a user clicks on a Web page, the DataRamp Assistant ActiveX control automatically downloads the application definition files, configures the required data source, and launches the application"

http //www activex com/PC/Result/TitleDetail/0,16,0-15818,00 html

Net-Installer
This controls allows programs to download and run automatically on the users machine

http //www twenty.com/Pages/NI/NIDTK.shtml

App-launcher
This control supports the ability to launch arbitrary apps without user intervention.

http //www activex com/PC/Result/Download/0,27,0-22413,00 html