# EasyHook: Down & Dirty with Microsoft Windows

By Aaron Grothe/NEbraskaCERT

03/17/2010

# LibSafe

Has anybody heard of it?

Very interesting little library for Linux written by Avaya

Did function call interceptions to protect certain basic strings functions such as strcpy

Went from sliced bread to ignored in about 6 months

# LibSafe

Has anybody heard of it?

Very interesting little library for Linux written by Avaya

Did function call interceptions to protect certain basic strings functions such as strcpy

Went from sliced bread to ignored in about 6 months

# LibSafe (Cont)

I thought you were going to talk about Microsoft Windows?

Avaya also wrote a version for Microsoft Windows using a product from Microsoft called Detours

Do a bing search and you'll be able to find the presentation, the white paper is a little harder to find

# Detours

At least we're talking about Windows now :-)
Detours is a Microsoft research project available at their site (guess the URL)
http://research.microsoft.com

# Detours

A Couple of issues with Detours

- Last Release was in 2006
- Only handles unmanaged code – not CLR aware
- 2 Versions Express/Research and Commercial

# Detours

What does Detours do for you?

Quite simply detours lets you interject your code into a function and force it to be called.

Pop out to a section from the Usenix paper

# Detours

Microsoft is one of the biggest users of Detours

- They use it to intercept registry calls for the Windows Vista Virtual Registry system
- Also used by AppFix to allow older progams to run on newer versions of windows

# Detours

Ok.  That sounds interesting why does it matter?

“Innovative systems research hinges on the ability to easily instrument and extend existing operating systems and application functionality”

Hunt & Brubacher – Detours: Binary Interception of Win 32 Functions – Presented at Usenix 1999

# Other Techniques

Some of the other techniques in the past have been as simple as rewriting binaries in memory – more difficult with authenticode and other techniques Others have been to recompile the program with various hooks in it as well

# EasyHook

So what is EasyHook?

EasyHook is a cleanroom implementation of a Detours like system.  EasyHook uses chunks of the Microsoft CLR to simplify a lot of the system.  No Assembly required (Pause for laughter about Pun or explain Pun when no laughter).

# EasyHook

How about a Demo or Two?

Ok.  Coming Up.  We'll show filemon and procmon tools that use EasyHook to implement systems that are similar to the classic Sysinternals commands.

# EasyHook

Lets take a bit of a look at how a project looks

# EasyHook

Doesn't this demo just do what the old filemon from Sysinternals did?

EasyHook's programs are just examples of what the system can do.  It does it in a cleaner implementation than the SysInternals stuff

# EasyHook

So what can EasyHook do?

- It will allow you to intercept/modify calls to the Windows API
- You can do additional checks/verification as part of the calls

# EasyHook

So I can do a chroot/sandbox with EasyHook?

EasyHook's documentation recommends against this.  One of the ways of trying to bypass something like EasyHook is to load the DLL and then do jumps to the various addresses to bypass EasyHook Still it is a direction that is being explored by projects like AppStract

# EasyHook

Where are you with EasyHook?

- Still figuring it out.  Microsoft Windows really isn't the place I spend most of my time.
- I'm still used to having the code and being able to modify that

# EasyHook

What I'd like to do with EasyHook

- Write a new version of libsafe and see what percentage of active bugs it would prevent
- Write a container for Adobe Acrobat to be able to contain all calls to the filesystem or to be able to notify people before it goes rogue
- That or a wrapper for Firefox to create a virtual conatiner/filesystem would be really cool

# EasyHook

Some of the limitations of EasyHook

- It can be bypassed or subverted
- It can generate some issues with Windows Defender
- There is more than one way to do a lot of things in Microsoft Windows – use an unmonitored call
- Hard to do things “System Wide”

# Patchguard

Patchguard and its API are the documented way of extending the Windows Kernel in 64 bit systems

It can do this systemwide and it can also be setup to be very difficult to bypass

Documentation and access to the API are being worked on to this date as part of the resolution of issues with the EU

AV vendors and other groups use these APIs on newer systems

# 5 Things I've Learned

#1. Microsoft Visual C++ Express is a very nice IDE, which is quite a bit different from GCC
#2. EasyHook is a truly amazing little piece of software
#3. If you want to do experimentation on an program without having source code access you can take a good shot at it with EasyHook
#4. codeplex is a nice site
#5. First rule of Patchguard is you don't talk about Patchguard

# Summary

So what can I do with this stuff?

- You can write interesting tools
- You have additional options to get more information about what is happening on your system

# References from the Talk

EasyHook
Microsoft Detours
Patchguard & Patchguard API
Wininternals & Sysinternals
AppStract