The Case of the Modified Binaries

leviathansecurity.com/blog/the-case-of-the-modified-binaries

Josh Pitts October 23, 2014

Summary

After creating and using a new exitmap module, I found downloaded binaries being patched through a Tor exit node in Russia. Tor is a wonderful tool for protecting the identity of journalists, their sources, and even regular users around the world; however, anonymity does not guarantee security.

Background

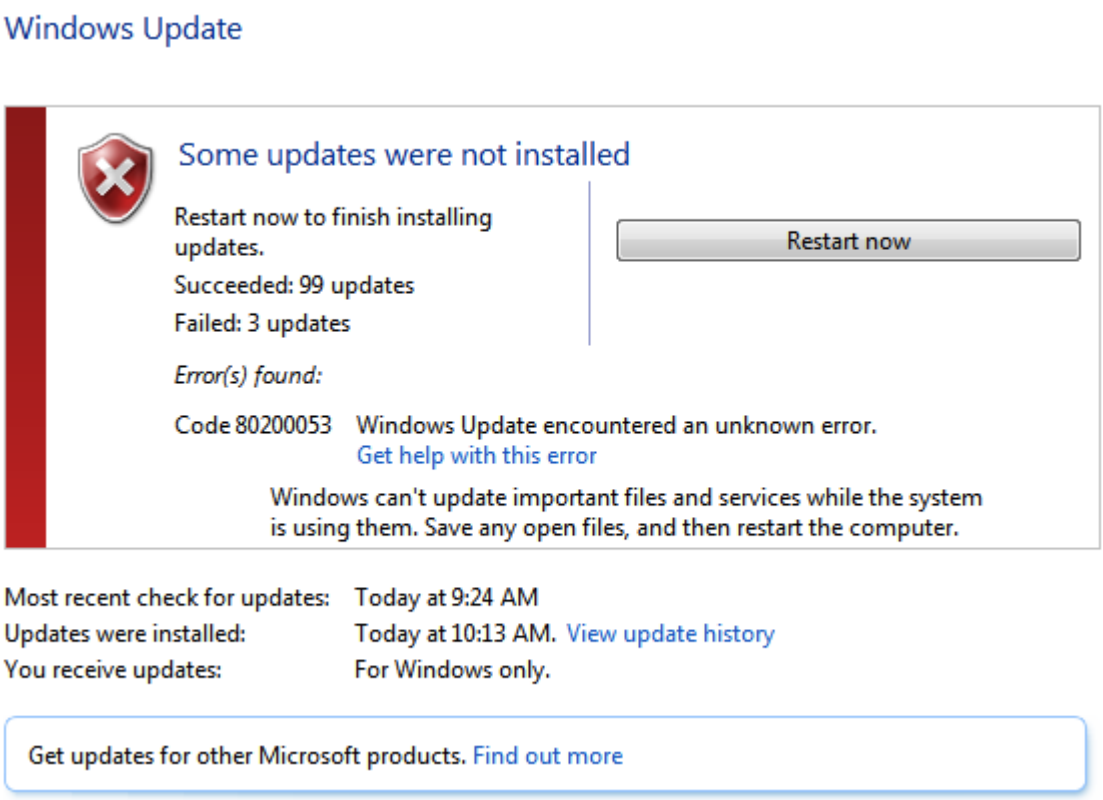
At DerbyCon this year I gave a presentation of my binary patching framework, BDF. Many binaries are hosted without any transport layer security encryption. Some binaries are signed to prevent modification, but most are not. During that presentation, I talked about the MITM patching of binaries during download, and showed how easy it was using BDFProxy. I also mentioned that similar techniques are probably already in use on the Internet.

I had only circumstantial evidence until recently.

Circumstantial Evidence

Microsoft Updates Error

I tested BDFProxy against a number of binaries and update processes, including Microsoft Windows Automatic updates. The good news is that if an entity is actively patching Windows PE files for Windows Update, the update verification process detects it, and you will receive error code 0x80200053.

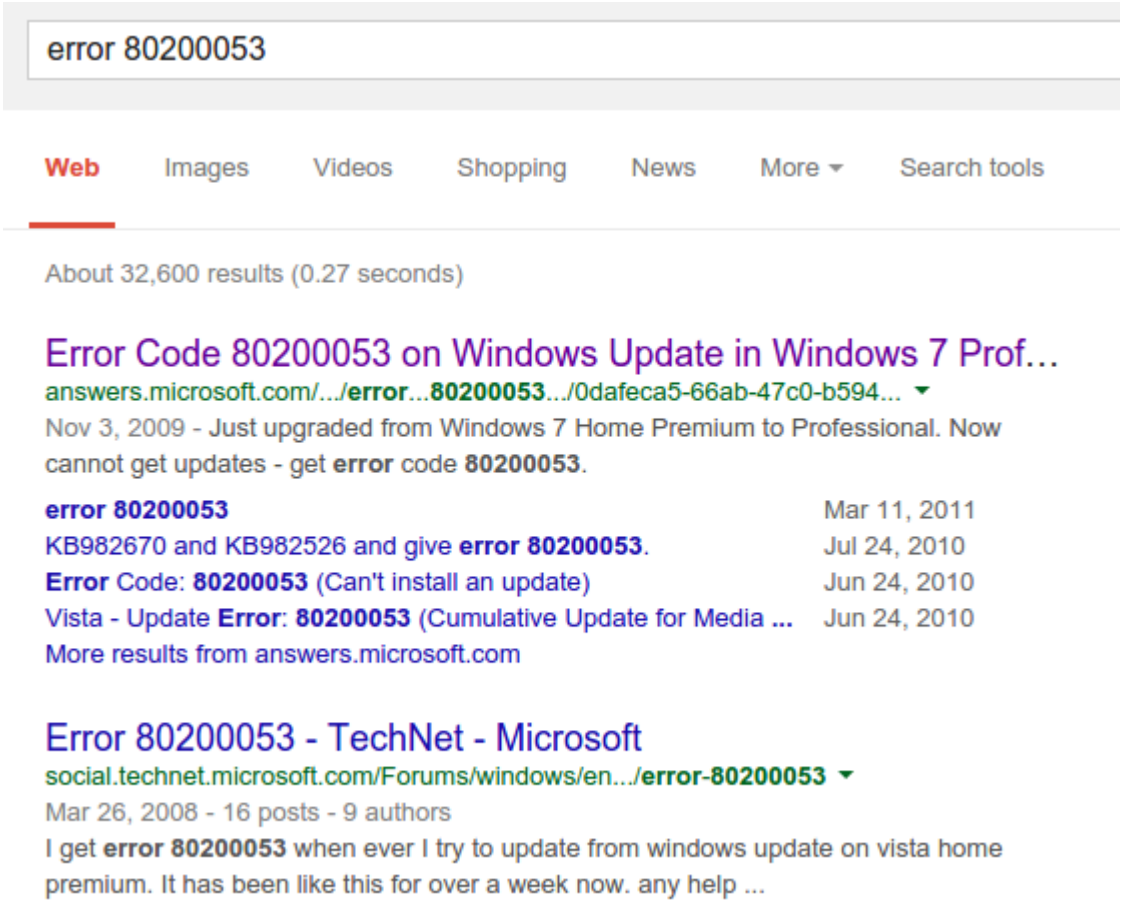


failedSigs

This error code indicates a failed signature verification for the downloaded binary. Windows Update produces this error code for three root causes:

1. The file was truncated during download. Very possible.
2. The file was patched during download. Improbable.
3. MS certificate verification is broken. Very improbable.

If you Google the error code, the official Microsoft response is troublesome.

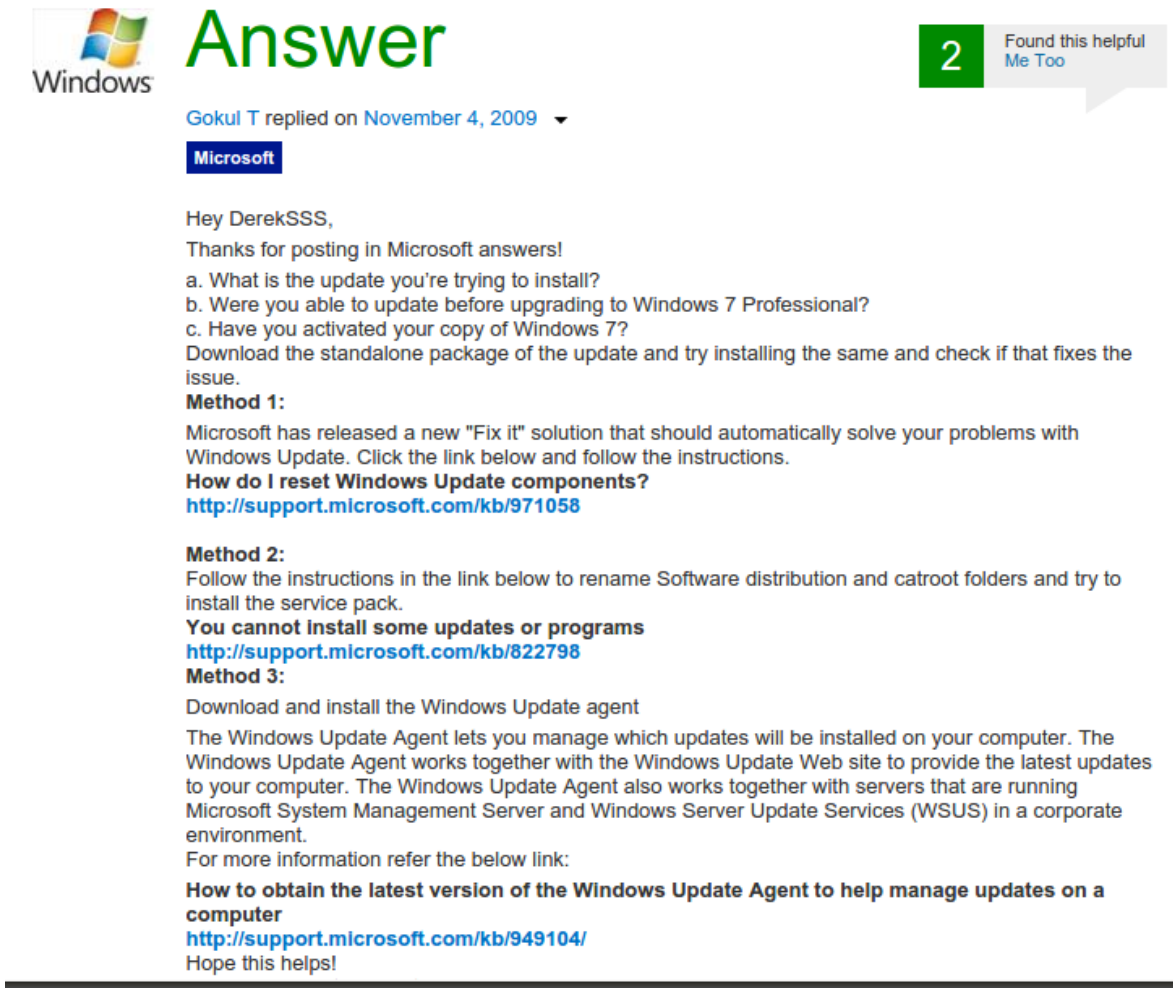


Google_error_80200053

The first link will bring you to the official Microsoft Answers website. Notice that this question has been viewed over 34,000 times.

question

If you follow the three steps from the official MS answer, two of those steps result in downloading and executing a MS 'Fixit' solution executable.



Windows

Answer

2 Found this helpful
Me Too

Gokul T replied on November 4, 2009

Microsoft

Hey DerekSSS,

Thanks for posting in Microsoft answers!

a. What is the update you're trying to install?
b. Were you able to update before upgrading to Windows 7 Professional?
c. Have you activated your copy of Windows 7?
Download the standalone package of the update and try installing the same and check if that fixes the issue.

Method 1:

Microsoft has released a new "Fix it" solution that should automatically solve your problems with Windows Update. Click the link below and follow the instructions.
How do I reset Windows Update components?
http://support.microsoft.com/kb/971058

Method 2:
Follow the instructions in the link below to rename Software distribution and catroot folders and try to install the service pack.
You cannot install some updates or programs
http://support.microsoft.com/kb/822798
Method 3:

Download and install the Windows Update agent

The Windows Update Agent lets you manage which updates will be installed on your computer. The Windows Update Agent works together with the Windows Update Web site to provide the latest updates to your computer. The Windows Update Agent also works together with servers that are running Microsoft System Management Server and Windows Server Update Services (WSUS) in a corporate environment.
For more information refer the below link:

How to obtain the latest version of the Windows Update Agent to help manage updates on a computer
http://support.microsoft.com/kb/949104/
Hope this helps!

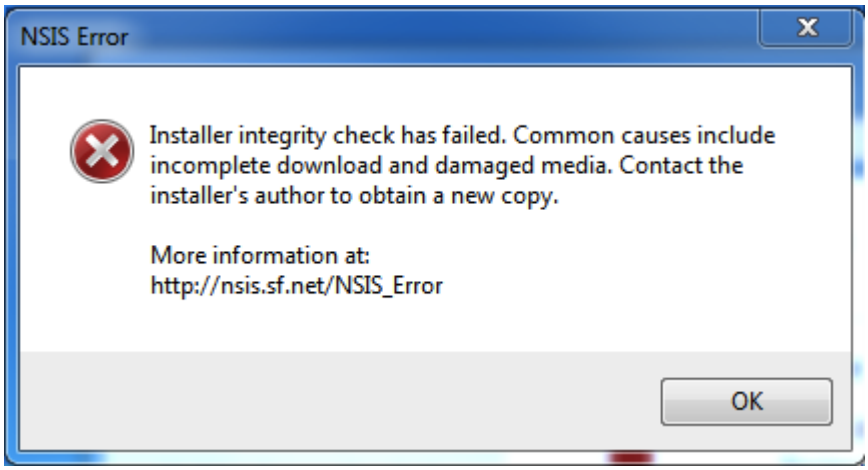
answer

If an adversary is currently patching binaries as you download them, these 'Fixit' executables will also be patched. Since the user, not the automatic update process, is initiating these downloads, these files are not automatically verified before execution as with Windows Update. In addition, these files need administrative privileges to execute, and they will execute the payload that was patched into the binary during download with those elevated privileges.

Note: a Windows Home or Enterprise user could configure AppLocker to only run signed binaries.

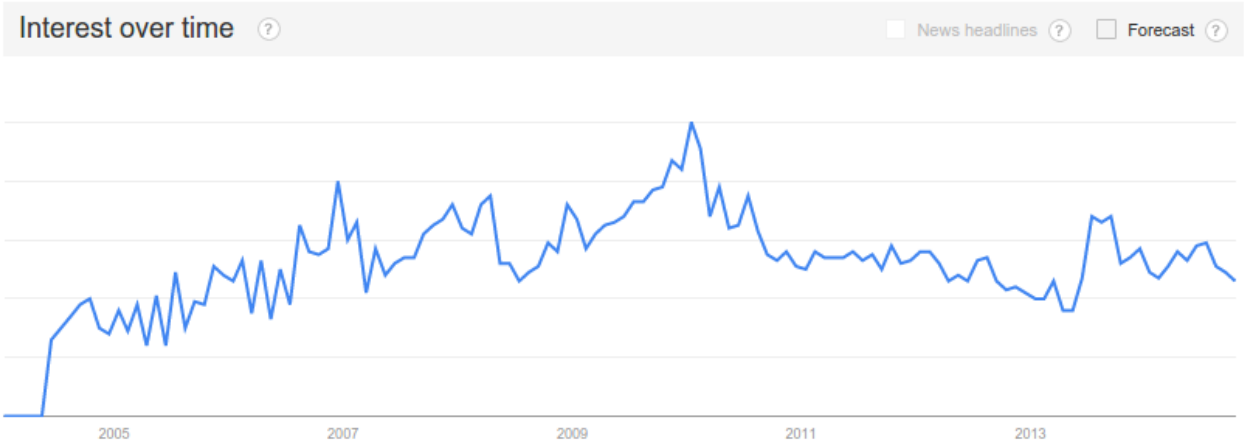
Nullsoft Scriptable Install System (NSIS) Error

NSIS provides a form of self-checking that weakly ensures that a binary was not modified after compiling. It issues the following error when the self-checking fails:



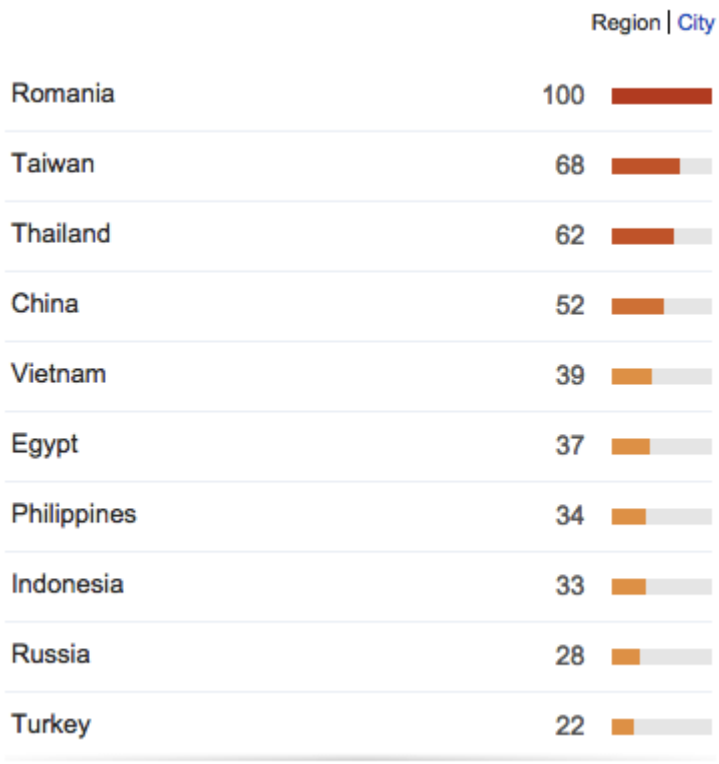
NSIScheck

Looking at Google Trends, this error message is quite common:



NSIServer

Notice the top countries where this search is originating:



topCountries

A user can receive an error code for any of the following three root causes:

1. The binary was patched. Improbable.
2. The binary was truncated due to a poor Internet connection. Very probable.
3. An actual error with the install program. Very improbable.

This combined circumstantial evidence left me wondering if there is an individual or group actively patching binaries on the greater Internet.

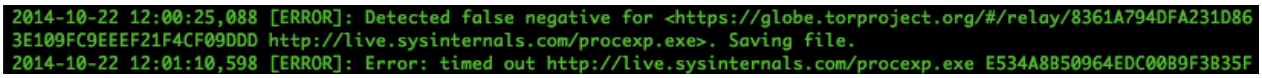
Caught Red-Handed

To have the best chance of catching modified binaries in transit over the Internet, I needed as many exit points in as many countries as possible. Using Tor would give me this access, and thus the greatest chance of finding someone conducting this malicious MITM patching activity.

After researching the available tools, I settled on exitmap. Exitmap is Python-based and allows one to write modules to check exit nodes for various modifications of traffic. Exitmap is the result of a research project called Spoiled Onions that was completed by both the PriSec group at Karlstad University and SBA Research in Austria.

I wrote a module for exitmap, named patchingCheck.py, and have submitted a pull request to the official GitHub repository. See the usage example.

Soon after building my module, I let exitmap run. It did not take long, about an hour, to catch my first malicious exit node.



falseNegative

Details from https://check.torproject.org/exit-addresses

ExitNode 8361A794DFA231D863E109FC9EEEF21F4CF09DDD

Published 2014-10-22 01:06:40

LastStatus 2014-10-22 02:02:33

ExitAddress 78.24.222.229 2014-10-22 02:08:01

This exit node was very active.

Patched Binaries by Exit Node 8361A794DFA231D863E109FC9EEEF21F4CF09DDD

Original Binary Download URL

Online Sandbox Analysis URL

http://download.microsoft.com/download/5/B/C/5BC5DBB3-652D-4DCE-B14A-475AB85EEF6E/vcredist_x86.exe

Malwr.com Analysis Results

http://download.microsoft.com/download/3/2/2/3224B87F-CFA0-4E70-BDA3-3DE650EFEBA5/vcredist_x64.exe

Malwr.com Analysis Results

http://download.tuxfamily.org/notepadplus/6.6.9/npp.6.6.9.Installer.exe

Malwr.com Analysis Results

http://downloads.malwarebytes.org/file/mbam/

Malwr.com Analysis Results

http://live.sysinternals.com/psexec.exe

Malwr.com Analysis Results

http://live.sysinternals.com/tcpview.exe

Malwr.com Analysis Results

http://nmap.org/dist/nmap-6.47-setup.exe

Over size limit for Malwr.com.

VirusTotal Results

http://www.ntcore.com/files/ExplorerSuite.exe

Malwr.com Analysis Results

http://www.spybotupdates.com/files/filealyz-2.0.5.57.exe

Malwr.com Analysis Results

http://live.sysinternals.com/procexp.exe

Malwr.com Analysis Results

Upon further inspection, the original binary is wrapped within another binary similar to the technique mentioned in the research from Flex Grobert, et al, titled "Software Distribution Malware Infection Vector" (2008). However, these malware authors solved the icon issue noted in the paper by keeping the .rsrc section intact. By using a wrapper for the original binary, the malware authors do not invoke the NSIS error and bypass simple self-checking mechanisms.

Out of over 1110 exit nodes on the Tor network, this is the only node that I found patching binaries, although this node attempts to patch just about all the binaries that I tested. The node only patched uncompressed PE files. This does not mean that other nodes on the Tor network are not patching binaries; I may not have caught them, or they may be waiting to patch only a small set of binaries.

Leviathan has notified the Tor Project of the issue.

Going Forward

Companies and developers need to make the conscious decision to host binaries via SSL/TLS, whether or not the binaries are signed. All people, but especially those in countries hostile to "Internet freedom," as well as those using Tor anywhere, should be wary of downloading binaries hosted in the clear---and all users should have a way of checking hashes and signatures out of band prior to executing the binary.