

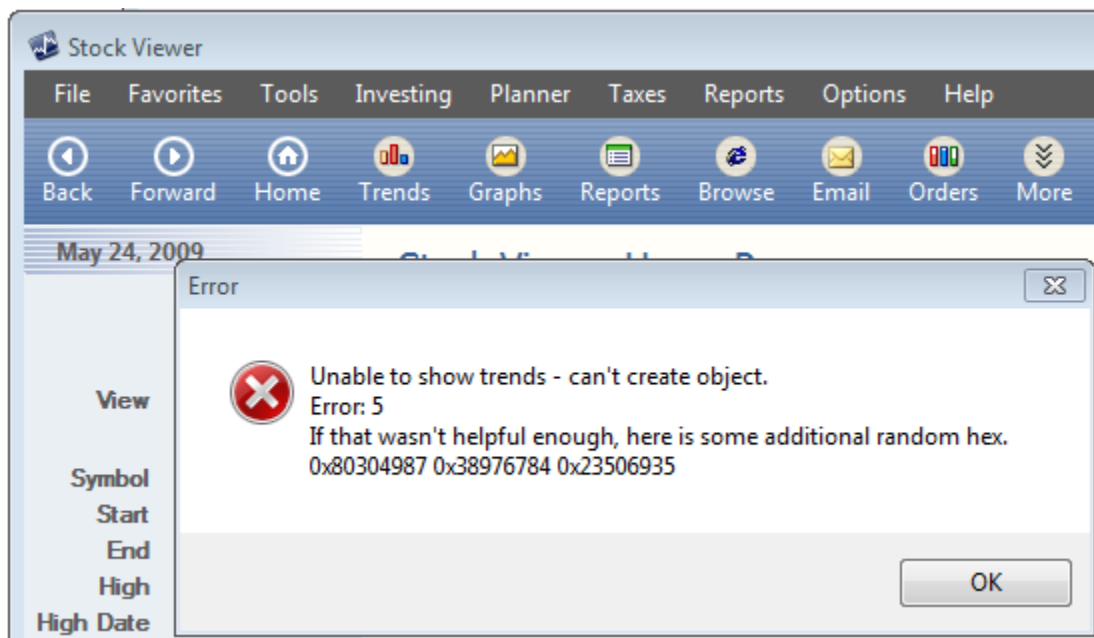
# The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)

A couple of weeks ago I participated for the first time in the keynote at Microsoft's Teched US conference to a room of over 5,000 attendees. Bill Veghte, the Senior Vice President of Windows marketing, led the keynote and gave a tour of the user-focused features of Windows 7, Iain McDonald, General Manager for Windows Server, demonstrated new functionality in Hyper-V and Windows Server 2008 R2, and I demonstrated IT Pro-oriented enhancements in Windows 7 and Microsoft Desktop Optimization Pack (MDOP).

I showed features like BitLocker To Go group policy settings, PowerShell v2's remoting capabilities, PowerShell's ability to script group policy objects, Microsoft Enterprise Desktop Virtualization (MEDV) and how the combination of App-V, roaming user profiles and folder redirection enable a replaceable PC scenario with minimal downtime. One point I reinforced was the fact that we made every effort to ensure that application-compatibility fixes (called shims) that IT Pros have developed for Vista applications work on Windows 7. I also demonstrated Windows 7's new AppLocker feature, which allows IT Pros to restrict the software that users can run on enterprise desktops with flexible rules for identifying software.

In the weeks leading up to the keynote I worked with Jason Leznek, the owner of the IT Pro portion of the keynote, to identify the features I'd showcase and to design the demos. We used dry runs to walk through the script, tweaking the demos and creating transitions, trimming content to fit the time allotted to my segment, and tightening my narration to focus on the benefits of the new technologies. For the application-compatibility demo, we decided to use a sample program used internally at Microsoft, called Stock Viewer, that's intentionality incompatible with Vista and Windows 7 in ways representative of actual line-of-business software that doesn't run without assistance on these newer operating systems. In my demo, I would launch Stock Viewer on Windows 7 and show how its trends report function fails with an obscure error message caused by incompatibility:



Then I'd show how I could deploy an application compatibility shim that enables the application to work correctly on Vista and then rerun the application successfully.

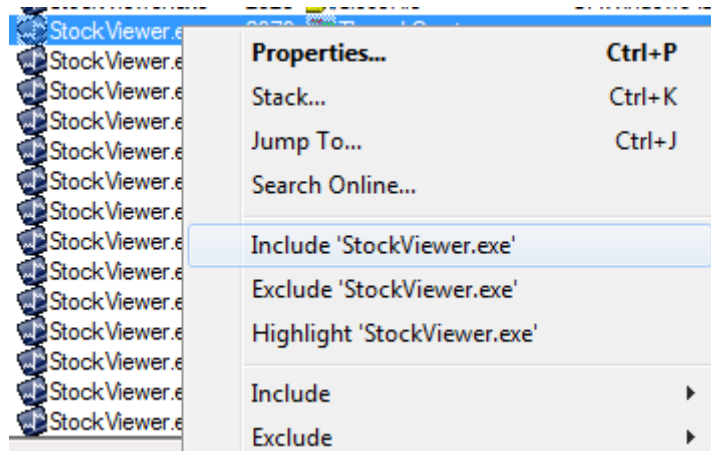
# The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)

We also wanted to show how AppLocker's rule creation wizard makes it easy to allow software to run based on the publisher or version if the software is digitally signed. Originally, we planned on showing AppLocker after the application compatibility demo and enabling Adobe Acrobat Reader, an application commonly used in enterprises. We rehearsed this flow a couple of times, but found the transitions a little awkward, so I suggested that we sign the Stock Viewer executable and move the AppLocker demo before the shim demo. I'd be able to enable Stock Viewer to run with an AppLocker rule and then show how the shim helps it run correctly, using it for both demos.

I went back to my office, signed Stock Viewer with the Sysinternals signing certificate and sent it to Jason. A few hours later he emailed me saying that something was wrong with the demo system because Stock Viewer, which had previously launched instantly, now took over a minute to start. We were counting down to TechEd and he was panicked because we needed to nail down the demos. I had heard at some point in the past that .NET does authenticode signature checks when it loads digitally signed assemblies, so my first suspicion was that it was related to that. I asked Jason to capture a Process Monitor trace and he emailed it back a few minutes later.

After opening the log, the first thing I did was filter events for StockViewer.exe by finding its first operation and right-clicking to set a quick filter:



Then I looked at the timestamps on the first item, 2:27:20, and the last item, 2:28:32, which correlated with the minute delay Jason had observed. As I scrolled through the trace I saw many references to cryptography (crypto) Registry keys and file system directories, as well as references to TCP/IP settings, but I knew that there had to be at least one major gap in the timestamps to account for the long delay. I scanned the log from the beginning and found a gap of roughly 10 seconds at 2:27:22:

2:27:21.1500506 PM	Stock Viewer.exe	2526	QueryBasicInformation...	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:21.1500415 PM	Stock Viewer.exe	2528	CloseFile	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:21.1501741 PM	Stock Viewer.exe	2528	CreateFile	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:21.1503016 PM	Stock Viewer.exe	2528	CreateFileMapping	C:\Windows\System32\vasadhlp.dll	FILE LOCKED WITH
2:27:21.1503485 PM	Stock Viewer.exe	2528	CreateFileMapping	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:21.1505176 PM	Stock Viewer.exe	2528	Load Image	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:21.1505375 PM	Stock Viewer.exe	2528	CloseFile	C:\Windows\System32\vasadhlp.dll	SUCCESS
2:27:22.0634600 PM	Stock Viewer.exe	2976	Thread Create		SUCCESS
2:27:32.3219796 PM	Stock Viewer.exe	2492	CreateFile	C:\Windows\System32\en-US\winhttp.dll.mui	SUCCESS
2:27:32.3220801 PM	Stock Viewer.exe	2492	CreateFileMapping	C:\Windows\System32\en-US\winhttp.dll.mui	FILE LOCKED WITH
2:27:32.3220976 PM	Stock Viewer.exe	2492	QueryStandardInforma...	C:\Windows\System32\en-US\winhttp.dll.mui	SUCCESS
2:27:32.3221374 PM	Stock Viewer.exe	2492	CreateFileMapping	C:\Windows\System32\en-US\winhttp.dll.mui	SUCCESS
2:27:32.3224601 PM	Stock Viewer.exe	2044	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Services\crypt32	REPARSE
2:27:32.3225041 PM	Stock Viewer.exe	2044	RegOpenKey	HKLM\System\CurrentControlSet\Services\crypt32	SUCCESS
2:27:32.3225577 PM	Stock Viewer.exe	2044	RegQueryValue	HKLM\System\CurrentControlSet\Services\crypt32\De...	NAME NOT FOUND

# The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)

The operations immediately before were references to the Rasadhlp.dll, a networking-related DLL, and a little earlier there were lots of references to Winsock registry keys, with accesses to crypto Registry keys immediately after the 10 second delay. It appeared that the system was not connected to the Internet and that the application was held up by some networking timeout of roughly 10 seconds. I looked forward in order to find the next gap and came across a 12-second interval:

2:27:32.4730150 PM	StockViewer.exe	3412	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad\{89AA6178-1F2F-4BB4-...
2:27:32.4733149 PM	StockViewer.exe	3412	RegCloseKey	HKCU\Software\Classes
2:27:32.4735418 PM	StockViewer.exe	3412	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad
2:27:32.4735745 PM	StockViewer.exe	3140	RegCloseKey	HKCU
2:27:32.4735906 PM	StockViewer.exe	3140	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:32.4741019 PM	StockViewer.exe	3412	RegQuery\Value	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters\NameSpace_Catalog5\Serial_Acc...
2:27:32.4741331 PM	StockViewer.exe	3412	RegOpenKey	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters\NameSpace_Catalog5\00000009
2:27:44.3248702 PM	StockViewer.exe	2044	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Services\crypt32
2:27:44.3249124 PM	StockViewer.exe	2044	RegOpenKey	HKLM\System\CurrentControlSet\Services\crypt32
2:27:44.3249669 PM	StockViewer.exe	2044	RegQuery\Value	HKLM\System\CurrentControlSet\Services\crypt32\DebugFlags
2:27:44.3249930 PM	StockViewer.exe	3140	Thread Exit	

Again, networking-related activity before, and crypto related activity after. The subsequent gap, also of 12-seconds, was identical:

2:27:44.4204329 PM	StockViewer.exe	2976	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad\00-1c-25...
2:27:44.4204453 PM	StockViewer.exe	2976	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad\{89AA61...
2:27:44.4206750 PM	StockViewer.exe	2976	RegCloseKey	HKCU\Software\Classes
2:27:44.4208480 PM	StockViewer.exe	2976	RegCloseKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad
2:27:44.4208788 PM	StockViewer.exe	488	RegCloseKey	HKCU
2:27:44.4208920 PM	StockViewer.exe	488	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:50.9861369 PM	StockViewer.exe	3632	Thread Exit	
2:27:56.3482524 PM	StockViewer.exe	2044	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Services\crypt32
2:27:56.3482970 PM	StockViewer.exe	2044	RegOpenKey	HKLM\System\CurrentControlSet\Services\crypt32
2:27:56.3483472 PM	StockViewer.exe	488	Thread Exit	
2:27:56.3483529 PM	StockViewer.exe	2044	RegQuery\Value	HKLM\System\CurrentControlSet\Services\crypt32\DebugFlags
2:27:56.3483770 PM	StockViewer.exe	2044	RegCloseKey	HKLM\System\CurrentControlSet\Services\crypt32

In fact, the next few gaps looked virtually identical. In each case there was a reference to HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections immediately before the pause, so I set a filter for that path and RegOpenKey and sure enough, could easily see six gaps of exactly 12-seconds each:

2:27:21.0764339 PM	StockViewer.exe	2492	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:21.1212746 PM	StockViewer.exe	2492	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:32.3539308 PM	StockViewer.exe	3140	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:32.4215307 PM	StockViewer.exe	3140	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:44.3402017 PM	StockViewer.exe	488	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:44.3970143 PM	StockViewer.exe	488	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:56.4164436 PM	StockViewer.exe	1840	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:27:56.4816304 PM	StockViewer.exe	1840	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:28:08.3923185 PM	StockViewer.exe	3800	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:28:08.4287594 PM	StockViewer.exe	3800	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:28:20.3860900 PM	StockViewer.exe	2448	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections
2:28:20.4268362 PM	StockViewer.exe	2448	RegOpenKey	HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Connections

The sum of the gaps – 12 times 6 – equaled the delay Jason was seeing. Next, I wanted to verify that the repeated attempts to access the network were caused by signing verification so I started looking at the stacks of various events by selecting them and typing Ctrl+K to open the stack properties dialog. The stack for events related to the Internet connection settings revealed that crypto was the reason:

## The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)

U 9	KernelBase.dll	RegCloseKey + 0x7d
U 10	winhttp.dll	CRegBlob::~CRegBlob + 0x17
U 11	winhttp.dll	WinHttpGetIEProxyConfigForCurrentUser + 0xc9
U 12	cryptnet.dll	InetGetProxy + 0xcf
U 13	cryptnet.dll	InetSendAuthenticatedRequestAndReceiveResponse + 0...
U 14	cryptnet.dll	InetSendReceiveUrlRequest + 0x312
U 15	cryptnet.dll	CInetSynchronousRetriever::RetrieveObjectByUrl + 0x5f
U 16	cryptnet.dll	InetRetrieveEncodedObject + 0x64
U 17	cryptnet.dll	CObjectRetrievalManager::RetrieveObjectByUrl + 0xbb
U 18	cryptnet.dll	CryptRetrieveObjectByUrlWithTimeoutThreadProc + 0x67
U 19	kemsel32.dll	BaseThreadInitThunk + 0xe
U 20	ntdll.dll	__RtlUserThreadStart + 0x70
U 21	ntdll.dll	__RtlUserThreadStart + 0x1b

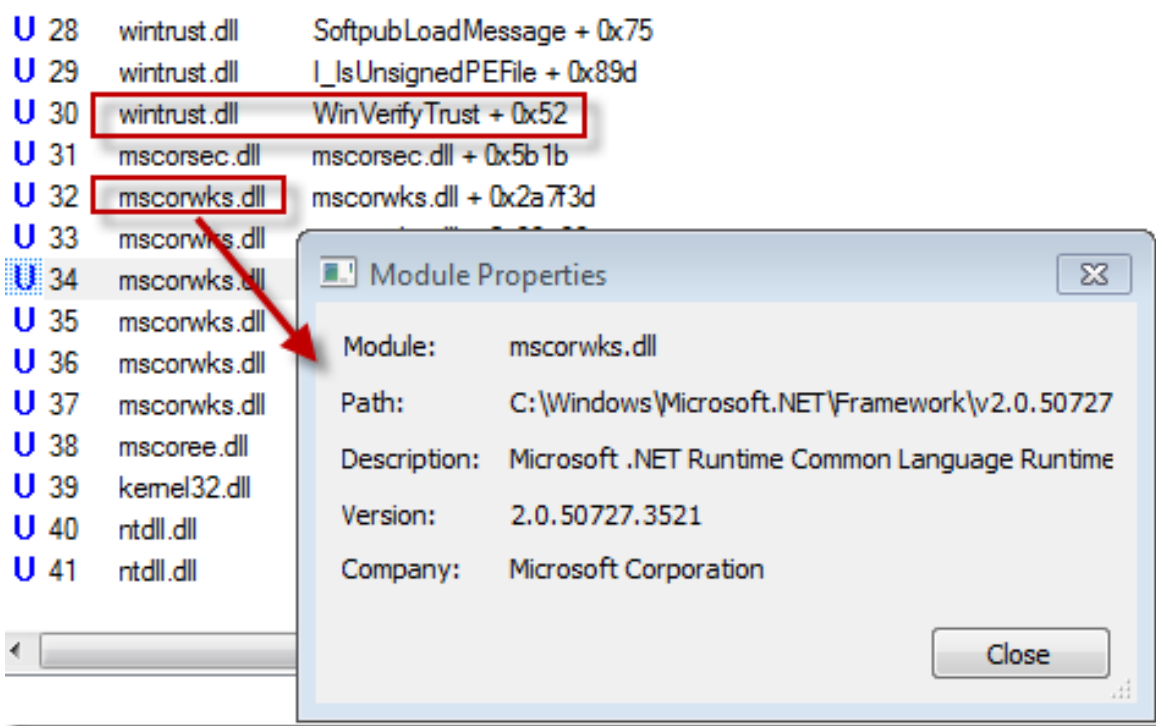
One final piece of evidence I wanted to check for was that .NET was ultimately responsible for these checks. I rescanned the log and I saw events in the trace that confirmed that Stock Viewer is a .NET application:

Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\System32\Wldap32.dll
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\System32\Wldap32.dll
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\Microsoft.NET\Framework\v2.0.50727\mscorlib.dll
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\Microsoft.NET\Framework\v2.0.50727\mscorlib.dll
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\winsxs\x86_microsoft.vc80.ct_1fc8b3b9a1e18e3b_8.0.50727.3521_none_d0... FI
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\winsxs\x86_microsoft.vc80.ct_1fc8b3b9a1e18e3b_8.0.50727.3521_none_d0... S
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\assembly\NativeImages_v2.0.50727_32\mscorlib\3ff595610f8be0c50733e48... FI
Stock Viewer.exe	2044	CreateFileMapping	C:\Windows\assembly\NativeImages_v2.0.50727_32\mscorlib\3ff595610f8be0c50733e48... S

I also looked at the stacks of some of the early events referencing crypto Registry keys and saw that it was the .NET runtime performing the call to WinVerifyTrust, the Windows function for checking the digital signature on a file, that started the cascade of attempted Internet accesses:

# The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)



Confident now that the cause of the startup delay was due to .NET seeing that Stockviewer.exe was signed and then checking to see if the signing certificate had been revoked, I entered Web searches looking for a way to make .NET to skip the check, since I knew that the keynote machines probably wouldn't be connected to the Internet during the actual keynote. After a couple of minutes of reading through articles by others with similar experiences, I found this KB article:



The article describes exactly the symptoms we were seeing and notes that .NET 2.0, which is the version of .NET I could see Stock Viewer was using based on the paths of the .NET DLLs it accessed during the trace, supports a way to turn off its obligatory checking of assembly digital signatures: create a configuration file in the executable's directory with the same name as the executable except with ".config" appended (e.g. StockViewer.exe.config) containing the following XML:

# The Case of the Slow Keynote Demo

Mark Russinovich  
(From Mark Russinovich Blog)

```
<?xml version="1.0" encoding="utf-8"?>  
<configuration>  
  <runtime>  
    <generatePublisherEvidence enabled="false"/>  
  </runtime>  
</configuration>
```

A total of about 15 minutes since I had received Jason's email, I sent him a reply explaining my conclusion with the configuration file attached. Shortly after, he wrote back confirming the delays were gone and expressing amazement that I had figured out the problem and solution so quickly. It might have seemed like magic to him, but I had simply used basic Process Monitor troubleshooting techniques and the Web to solve the case. Needless to say, the revised demo flow and transition between AppLocker and application compatibility came off great.