STUXNET: THE CURIOUS INCIDENT OF THE SECOND CERTIFICATE

“Is there any point to which you would wish to draw my attention?”

“To the curious incident of the dog in the night-time.”

“The dog did nothing in the night-time.”

“That was the curious incident,” remarked Sherlock Holmes.

Arthur Conan Doyle (Silver Blaze)

[From ew: William Ockham, who knows a whole lot more about coding than I, shared some interesting thoughts with me about the Stuxnet virus. I asked him to share those thoughts it into a post. Thanks to him for doing so!]

The key to unraveling the mystery of Stuxnet is understanding the meaning of a seemingly purposeless act by the attackers behind the malware. Stuxnet was first reported on June 17, 2010 by VirusBlokAda, an anti-virus company in Belarus. On June 24, VirusBlokAda noticed that two of the Stuxnet components, Windows drivers named MrxCls.sys and MrxNet.sys, were signed using the digital signature from a certificate issued to Realtek Semiconductor. VirusBlokAda immediately notified Realtek and on July 16, VeriSign revoked the Realtek certificate. The very next day, a new Stuxnet driver named jmidebs.sys appeared, but this one was signed with a certificate from JMicron Technology. This new Stuxnet driver had been compiled on July 14. On July 22, five days after the new driver was first reported, VeriSign revoked the JMicron certificate.

The question I want to explore is why the attackers rolled out a new version of their driver signed with the second certificate. This is a key question because this is the one action that we know the attackers took deliberately after the malware became public. It’s an action that they took at a time when there was a lot of information asymmetry in their favor. They knew exactly what they were up to and the rest of us had no clue. They knew that Stuxnet had been in the wild for more than a year, that it had already achieved its primary goal, and that it wasn’t a direct threat to any of the computers it was infecting in July 2010. Rolling out the new driver incurred a substantial cost, and not just in monetary terms. Taking this action gave away a lot of information. Understanding why they released a driver signed with a second certificate will help explain a lot of other curious things in the Stuxnet saga.

It’s easy to see why they signed their drivers the first time. Code signing is designed to prove that a piece of software comes from a known entity (using public key infrastructure) and that the software hasn’t been altered. A software developer obtains a digital certificate from a “trusted authority”. When the software is compiled, the certificate containing the developer’s unique private key is used to “sign” the code which attaches a hash to the software. When the code is executed, this hash can be used to verify with great certainty that the code was signed with that particular certificate and hasn’t changed since it was signed. Because drivers have very privileged access to the host operating system, the most recent releases of Microsoft Windows (Vista, Win7, Win2008, and Win2008 R2) won’t allow the silent installation of unsigned drivers. The Stuxnet attackers put a lot of effort into developing a completely silent infection process. Stuxnet checked which Windows version it was running on and which anti-virus software (if any) was running and tailored its infection process accordingly. The entire purpose of the Windows components of Stuxnet was to seek out installations of a specific industrial control system and infect that. To achieve that purpose, the Windows components were carefully designed to give infected users no sign that they were under attack.

The revocation of the first certificate by VeriSign didn’t change any of that. Windows will happily and silently install drivers with revoked signatures. Believe it or not, there are actually good reasons for Windows to install drivers with revoked signatures. For example, Realtek is an important manufacturer of various components for PCs. If Windows refused to install their drivers after the certificate was withdrawn, there would be a whole lot of unhappy customers.

The release of a Stuxnet driver signed with a new certificate was very curious for several reasons. As Symantec recently reported [link to large pdf], no one has recovered the delivery mechanism (the Trojan dropper, in antivirus lingo) for this driver. We don’t actually know how the driver showed up on the two machines (one in Kazakhstan and one in Russia) where it was found on July 17, 2010. This is significant because the driver is compiled into the Trojan dropper as resource. Without a new dropper, there’s no way for that version of the virus to have infected additional computers. And there is no evidence that I’m aware of that Stuxnet with the new driver ever spread to any other machines.

The release of the newly signed driver did exactly one thing: Increase publicity about Stuxnet. The inescapable conclusion is that the Stuxnet attackers wanted to make headlines in July 2010. As Holmes says in *Silver Blaze*, “one true inference invariably suggests others”. From this one inference, we can begin to understand the most puzzling parts of the Stuxnet project. Who would publicize their secret cyber attack on an enemy? Why were there clues to the identity of the attackers left in the code? Why did the last version of Stuxnet use multiple 0-day exploits? Why did the attackers only take minimal steps to hide the true nature of the code? The answer to these questions is relatively simple. The Stuxnet project was never intended to stay secret forever. If it had been, there would never have been a new Stuxnet driver in July 2010. That driver helps put all the other pieces in context: the clues left inside the code ("myrtus", "guava", and using May 9, 1979 as a magic value); the aspects of the code that have led various experts to label Stuxnet as amateurish, lame, and low quality; even the leak campaign by the U.S. and Israeli governments to unofficially take credit for Stuxnet. Rather than being mistakes, these were elements of the larger Stuxnet project.

Stuxnet was more than a cyber attack. It was a multi-pronged project. The design of the code supports the overall mission. The mission included a publicity campaign, or as the military and intelligence folks style it, a PSYchological OPeration (PSYOP). Unlike a typical malware attack, Stuxnet had (at least) two distinct phases. Phase 1 required a stealthy cyber attack against the Iranian nuclear program. Phase 2 required that the effects of that cyber attack become widely known while giving the perpetrators plausible deniability. That may seem a little strange at first, but if you put yourself in the shoes of the attackers, the strategy is more than plausible.

In fact, the attackers have explained it all. Take a look back at the story told in the New York Times article on January 15, 2011. According to the NYT, the Stuxnet project started as an alternative to an Israeli airstrike:

> Two years ago, when Israel still thought its only solution was a military one and approached Mr. Bush for the bunker-busting bombs and other equipment it believed it would need for an air attack, its officials told the White House that such a strike would set back Iran's programs by roughly three years. Its request was turned down.

Couple that statement with the reason the article appeared when it did:

> In recent days, American officials who spoke on the condition of anonymity have said in interviews that they believe Iran's setbacks have been underreported.

Imagine that you're an American policymaker who has to choose between launching a cyber attack and allowing a close ally to launch an actual military attack. If you choose the cyber attack option, how will anyone know that you've succeeded? If no one knows that you've successfully delayed the Iranian nuclear program, you'll be vulnerable to right-wing attacks for not doing enough to stop Iran and the pressure to bomb-bomb-bomb of Iran will grow. There's another reason to publicize the attack. If you're a superpower who starts a cyber war, you have to realize that your country contains a lot of very soft targets. You would want to make a big splash with this malware so that your industrial base starts to take the cyber war seriously. So, from the very beginning, the project included planning for the inevitable discovery and understanding of the Stuxnet malware. Just like the spread of the malware itself, the psyop will be impossible to directly control, but easy enough to steer in the appropriate direction. The attackers likely didn't know it would be Symantec and Ralph Langner who would start to unravel the exact nature of the Stuxnet malware, but they knew someone would. And they knew they would be able to get the New York Times to print the story they wanted to get out (I'm not demeaning the work of the reporters on this story, but I would hope they realize that there is a reason they aren't being investigated for publishing a story about our efforts to undermine Iran's nuclear program and James Risen was).