

Post-Stuxnet *Industrial Security*

Zero-Day Discovery and Risk Containment of Industrial Malware

By Torsten Roessel, Director of Business Development, Innominate Security Technologies AG

Following its discovery in June 2010, the Stuxnet worm triggered a world-wide sensation. It was the first publicly known root-kit attack targeted at industrial plants. It has infected tens of thousands of PCs, abusing and manipulating Windows-based automation software for its own purposes—to ultimately infiltrate malicious code into the controllers of specific real-world industrial installations.

After Stuxnet, the threats from malware and insufficient IT security in automation networks forecasted by industry experts can no longer be ignored. The real danger looming out there, however, is not from Stuxnet itself, but rather from mutations likely to be created by imitators who could now circulate other arbitrary, malicious code utilizing the same basic techniques. And while Stuxnet focused on products from the Siemens SIMATIC family and on STEP 7 PLC projects with very specific properties, such mutations could affect components from other vendors as well, and turn out to be a lot less selective in their damaging impact.

Apart from the fact that PCs in industrial use often are not (and cannot be) equipped with anti-virus software, Stuxnet has also made it clear that conventional virus scanners do not provide protection against attacks of this caliber. In retrospect, the analysis of Stuxnet has shown that the worm had gone unnoticed for at least 12 months before its discovery, and had not been detected by anti-virus programs during that period for lack of any known signatures for the malware.

Damaging Impact in Four Steps

To develop protective measures against Stuxnet-like attacks, a basic understanding of the worm's activities is essential. It unfolds its damaging impact in four steps on different layers.

1. Infection of Windows PCs: The worm uses an aggressive mix of mechanisms to spread onto and contaminate both networked as well as non-networked Windows PCs (the latter via USB flash drives). For doing so, it utilizes a

total of four zero-day exploits of previously unknown vulnerabilities which exist in several generations of Windows operating systems, and have only partially been fixed by security patches to date. In addition to a number of encrypted files which the worm stores in the %SystemRoot%\inf\ directory, Stuxnet installs the two device drivers MrxNet.sys and MrxCLS.sys in %SystemRoot%\system32\drivers\. These drivers have been signed with stolen private digital keys from Realtek and JMicron and do, therefore, contain certificates which are rated as trustworthy by Windows systems.

2. Abuse and Manipulation of Automation Software: If Stuxnet

comes across installations of WinCC visualization and/or STEP 7 engineering components on an infected PC, it abuses and manipulates any found WinCC databases and STEP 7 projects to ensure its further proliferation and persistency on the PC, and to spy out the controllers referenced in those projects as potential targets for step 3.

Furthermore, Stuxnet renames the dynamic link library which is responsible for the communication between SIMATIC Manager and the projected S7 controllers (s7otbxdx.dll in the %SystemRoot%\system32\ thereby being renamed to s7otbxdx.dll) and replaces it with a wrapper DLL of its own under the original name in the same directory, effectively hiding the modification.

3. Injection of Malicious Code into Controllers: This manipulated wrapper DLL enables Stuxnet to infiltrate arbitrary malicious code into the projected PLCs, to hide those manipulations from the programming engineer, and to safeguard them from later overwriting. The precise malicious code selectively injected by Stuxnet, only into controllers and projects with very specific properties, is of remarkable sophistication and—according to the latest expert findings—is supposed to permanently manipulate frequency converters and turbine controls as inconspicuously as possible, with the goal of disrupting the controlled processes and ultimately destroying the affected equipment.

The malicious code targeted at controller models of the S7-417 series in particular, is combining denial-of-control and denial-of-view techniques into a man-in-the-middle attack in ways rarely considered until now. Under the attack, the legitimate PLC program completely loses control of the process without this being noticed by the PLC, or by the operating staff viewing their consoles in the control room. The underlying attack vector is generic and reproducible. It could be packaged into and provisioned by exploit tools such as Metasploit and then—contrary to common misconceptions—be used for attacks even by persons lacking in comprehensive insider knowledge.

4. Communication with Command and Control Servers on the Internet: From infected PCs, the worm attempts to contact its designated command and control servers on the Internet. When a connection is established, information collected from the target and its environment can be uploaded to those servers as well as new instructions and updates to the worm, and its malicious payload can be received and executed. This adds an extra dynamic depth to the worm's potential for espionage and sabotage. Combined with the worm's capabilities to spread and update itself via peer-to-peer connections and USB flash drives, all of this can have collateral effects even on systems without a network connection or Internet access.

Benefits of the Innominate mGuard Technology

The mGuard technology developed by Innominate—available in mGuard-based industrial network security appliances from multiple vendors and as an embedded technology for OEMs—comprises a set of preventive and diagnostic functions which can boost security against Stuxnet-like attacks and reduce their associated risks. Due to the diversity of proliferation paths, actively preventing 100 percent of all malware infections may not be practical. Therefore, an important aspect of protection is to discover such infections quickly and reliably rather than letting them slip through unnoticed and adversely affect facilities for a long span of time as was the case with Stuxnet.

Discover Malware on Day Zero: mGuard Integrity Monitoring

Due to the difficulties of deploying anti-virus software on industrial PCs and with the timely provision of malware signatures, alternative techniques of integrity assurance are gaining relevance and acceptance for the protection of industrial systems. The mGuard CIFS Integrity Monitoring method, for instance, provides monitoring of configurable sets of files on PCs for unexpected modifications of executable code (CIFS or Common Internet File System denoting the file sharing protocol used by Windows and other operating systems). When initialized, it computes a baseline of signatures for all monitored objects and then periodically checks them for any deviations. This process works without any external provision of virus signatures, without the risk of disrupting

operations through “false positives,” without installation of software, and with moderate load on the monitored PCs, by utilizing the processing resources of an mGuard security appliance. In this way, suspect modifications are reliably discovered and promptly reported via SNMP and e-mail to network management systems or responsible administrators.

Contain Proliferation, Prevent Virus C&C Transmissions: mGuard Firewall

When spreading across networks and vulnerabilities in operating systems, malware often takes advantage of network connections that are not at all necessary for the productive operation of a plant. The correct protection of industrial PCs and controllers or groups of such devices (security segments) with mGuard firewalls reliably blocks these unsolicited connections and restricts the proliferation of malware. In the case of Stuxnet, healthy systems could be infected by both incoming as well as outgoing connections—and vice versa, contaminated systems could spread their infection via corresponding outgoing and incoming connections. When protecting systems with firewalls, therefore, the often neglected outgoing connections (not only the incoming) should also be filtered as much as possible. In particular, this will prevent contacts via outgoing connections to command and control servers on the Internet and the associated potential for espionage and dynamic threats.

**Innominate Security Technologies AG,
a Phoenix Contact Company**

www.innominate.com

Write In **589**