Stuxnet Worm Impact on Industrial Cyber-Physical System Security

Stamatis Karnouskos
SAP Research, Germany
Email: stamatis.karnouskos@sap.com

***Abstract*—Industrial systems consider only partially security, mostly relying on the basis of "isolated" networks, and controlled access environments. Monitoring and control systems such as SCADA/DCS are responsible for managing critical infrastructures operate in these environments, where a false sense of security assumptions is usually made. The Stuxnet worm attack demonstrated widely in mid 2010 that many of the security assumptions made about the operating environment, technological capabilities and potential threat risk analysis are far away from the reality and challenges modern industrial systems face. We investigate in this work the highly sophisticated aspects of Stuxnet, the impact that it may have on existing security considerations and pose some thoughts on the next generation SCADA/DCS systems from a security perspective.**

I. Introduction

Much of our critical infrastructure is controlled by cyber-physical systems responsible for monitoring and controlling various processes [1]. The Supervisory Control And Data Acquisition (SCADA) system are industrial control systems responsible for a wide range of industrial processes e.g. manufacturing, power generation, refining, as well as infrastructure e.g. water management, oil & gas pipelines, wind farms, and facilities e.g. airports, space stations, buildings etc. The importance of monitoring and control, which heavily relies on such cyber-physical systems, is paramount for European and world economies in various industrial sectors; indicatively this market is expected to grow from an estimated 275 € Bn in 2012 to 500 € Bn in 2020 [2]. As we move towards large-scale introduction of IT technologies in these sectors, and automatic management, any digital threats that may arise will have a tangible impact on the real world [3] and its processes.

The summer of 2010 was a landmark to the security of the industrial software and equipment industry. By that time it was obvious that a new computer worm called Stuxnet [4] (its name is derived from keywords in its code) was targeting highly specialized industrial systems in critical high-security infrastructures. In the months followed it was becoming clear that this was an unprecedented sophisticated attack that would have wide implications for future industrial systems. For many it was a wakeup call and increased the awareness on security which is still seen as an afterthought and add-on, and not as a continuous process that should be integrated in all operational aspects. Although attacks in IT systems are not something new, up to now it was considered highly unlikely that large scale attacks in the software side of highly specialized applications (such as that of a SCADA) were worth trying or even possible (mainly due to the very niche technology and expertise needed). Additionally it was considered that a "safe" environment (implying disconnected from the Internet and with limited personnel access) was good enough protection. All of these considerations though have been radically changed the last months due to the Stuxnet incident.

This attack comes at an extremely critical time, as modern industrial systems move towards the adoption of Internet based technologies and architectures [5]; although not necessarily connected to the Internet itself. General purpose computing systems, complex industrial applications composable of heterogeneous software and hardware components, wireless access points, abstraction of hardware and uniform access via web services etc. are on the rise. Additionally the enterprise IT systems are getting more interconnected with the industrial ones in order to make sure that events occurring on the shop-floor can be immediately communicated to the respective business processes. The IT industry is well equipped with risk analysis and security tools, however the same does not hold true for industrial systems and the risks may not be adequately assessed.

II. The Stuxnet Worm

The Stuxnet worm had as its main target industrial control systems with the goal of modifying the code running in Programmable Logic Controllers (PLCs) in order to make them deviate from their expected behavior [6], [7]. This deviation would be small and only noticeable over a longer period of time. In parallel great effort was put by the Stuxnet creators in hiding those changes from the operators, even imitating "legitimate" data. To increase the success rate a vast majority of security holes and tools was used such as rootkits (including what is now known as the first PLC rootkit), antivirus tricking, zero-day exploits, network discovery and P2P updates, process injection etc. Many of these are common on modern PCs however the sophistication of the attach was unprecedentedly well-planned and highly customized for specific industrial systems. Recent analysis [7] points out that more than 80% of the infected systems rely mainly in Iran but also in Indonesia and India. Although the main attacks were detected in mid-2010, early variants of the Stuxnet code stemming from 2009 have been found. It is believed that the development of such a highly sophisticated worm was a joint-effort with experts from different specializations and a huge investment in time and cost.

It is now known that the target was solely the Siemens SCADA systems targeting very specific industrial processes. Stuxnet infects project files of the Siemens WinCC/PC S7 SCADA control software and intercepts the communication between the WinCC running in Windows and the attached PLC devices when the two are connected via a data cable (widely known as "man-in-the-middle" attack). The original infection of the Windows computer may be done via simply plugging in a USB flash drive or from the internal network if an infected machine exists.

Stuxnet focused on identifying specific slave variable-frequency drives attached to the Siemens S7-300 system. Furthermore it has been reported that it would only attack specific provider of those PLC systems i.e. coming from Vacon (Finnish vendor) and Fararo Paya (Iran). However in order to have a more specialized target, it monitors the frequency of the attached motors, and only attacks systems that spin between a specific range. Then it installs malware on the PLC that monitors the Profibus of the system and under certain conditions it periodically modifies that frequency, which results in that the connected motors change their rotational speed. Additionally it has installed the first known industrial rootkit which fakes industrial process control sensor signals, hence no alarms or shutdown is done due to abnormal behavior! This slowly deviating behavior in combination with the projection of "legitimate" data results in difficulty to assess what is malfunctioning and to pinpoint the faults before it is too late.

In order to demonstrate the sophistication of this effort, we would like to point out that Stuxnet:

- Utilized zero-day exploits i.e. security holes that the software developers were unaware of.
- Its code was obfuscated and difficult to reveal its functionality. Even today we do not understand it in its hole.
- A custom encryption algorithm was used for its configuration data.
- It took advantage of the private network (not connected in the Internet) to automatically update itself once a new copy of it was discovered. Hence an infected machine with newer Stuxnet version in the network would result in all existing Stuxnet installations to be upgraded to that version.
- It utilized peer-to-peer networks to dynamically discover and communicate (update) with all Stuxnet installations.
- All of the actions were done in memory and there fore no disk evidence (files) exists.
- It kept an infection counter.
- Had a highly modular architecture.
- Was masking under legal programs.
- Deployed anti-virus detection mechanisms.
- Could detect Internet connectivity and only then would attempt to connect to its Internet hosted Command & Control center.
- Elevated privileges (via specific exploits) in an unpatched machine in order to have the necessary execution rights
- Would infect in a very specific way only targeted systems (highly target-customizable).
- Had strict self-scalability control i.e. it would contain safeguards to prevent infected computers spreading the worm to more than three others.
- Had an un-install mechanism which removed itself (self-lifecycle management). It was programmed to erase itself on 24-June-2012.
- Contains, among other things, code for a man-in-the-middle attack that fakes industrial process control sensor signals; hence processes and tools relying on the data it generates would falsely depict further "normal" values and functionality that did not mirror the actual real world.
- Deployed legitimate digitally signed device drivers (with stolen private keys of two certificates that were stolen from separate companies)
- Had external websites configured as command and control (C&C) servers. This would enable various monitoring and control activities (if Internet was available) including industrial espionage by uploading information (originating internal connections to external servers are usually "acceptable" flows by firewalls)

An in depth analysis as well as concrete technical details of the aforementioned issues is available [6], [7].

III. Lessons Learned and Discussion

As one can derive from the characteristics that Stuxnet possesses, it is a highly sophisticated worm with a very specific strategic goals. Its detection, analysis and measured effects are reshaping the industry, as security awareness has increased. What is worth noticing is that some features of it partially fall under the good engineering practices for modern application and system development. For instance modular design and self-evolution (even by incremental updates) in the network are desired features for larger systems.

To realize much of its functionality, Stuxnet relied on a number of existing vulnerabilities, some of which dated two years back. Updates should have been applied during that time. "Dont touch a running system" is not applicable when it comes to security. Continuous security updates should be done (after testing) considering the context of the operating machine as well as its role in total within the plant. Of course if the systems are not connected to the Internet update site of the manufacturer, then an Intranet update server needs to be installed. Understandably this creates some overhead, plus experts needs to apply updates to critical software applications.

Unfortunately in long-lived industrial infrastructures (e.g. lifetimes 10+ years), updates are not as often due to the fear of unwanted side-effects. However these poorly defended, poorly patched and poorly regulated systems such as the PLCs in the Stuxnet case, will be the first ones that will be used as Trojan horses. A risk analysis should be done considering the whole infrastructure lifecycle. Concerns about single points of failure are valid however they should be assessed against fixing issues a posteriori.

Apart from the vulnerabilities in the Siemens Simatic S7 PLC itself [8], default hard-coded access accounts and passwords existed in the Siemens products; however it would be naive to expect these to stay secret or assume no use since these would operate in isolated environments. Security negligence over easiness of functionality may have catastrophic results.

Using configuration for malicious behavior detection e.g. via anti-virus programs may be a good start for existing well known attack signatures but the reliance on a single detection mechanism is not a guarantee especially for non-considered attacks. Heuristics for estimating behavior deviation may provide hints, which should be assessed and analyzed in conjunction with other metrics.

In the case of Stuxnet, removable media acted as propagation framework. Although basic checks are at the operating system side, for critical systems a complete physical access framework as well as IT policy needs to be in place. This may call for trusted hardware as well as avoidance of any carrier of potential malicious code. Apart from USB stick based attacks [9], [10], several other exist for other hardware parts e.g. the Ethernet card [11], the battery [12] etc.

Additionally, apart from accessing the host for one-time operations, such attacks can be used for industrial espionage e.g. use unintended USB channels to create bidirectional communication with external entities [13]. Since most of the devices in the future industrial infrastructures are expected to be an amalgamation of hardware and software, one has to consider also the possibility that the software part may be compromised or even the hardware itself may be partially designed to enable a multitude of attacks [14].

Stuxnet installed two kernel drivers that were digitally signed by valid certificates that were stolen from two different issuing companies. Real time online validation of certificates e.g. such as those offered by the Online Certificate Status Protocol (OCSP) must be in place and immediately propagated to the respective systems as this minimizes the window of error (of course only after these have been revoked by the issuer). Critical infrastructure devices such as the PLCs were not connected to the Internet or even to the internal network. However they would be connected e.g. for reprogramming with a laptop. That would be enough to propagate the malware. Not connecting a system to a network does not give a guarantee that it will be safe [15]. Opportunistic connections e.g. for reconfiguration or maintenance or even functionality assessment would be enough to introduce malicious code. This holds especially true for more modern devices which may also feature wireless (even short-range) protocols but not have them deactivated.

Security clearance on people does not imply security on their accompanying assets. In the Stuxnet case, a trustworthy employee with an unknowingly rootkited laptop or an infected USB flash drive would be enough to spread the malware. This could be for instance a contractor assigned to do the maintenance on the facility.

Considering that there was very good chance that no Internet connectivity would be available (only access to the internal network), Stuxnet developers put all of its logic in the code without the need of any external communication. As such the Stuxnet was an autonomous goal-oriented intelligent piece of software capable of spreading, communicating, targeting and self-updating; remarkable features that would enable it to survive in a stealth and persistent way in large scale systems.

Stuxnet was impersonating the normal behavior of the PLC. Any network management system or control room operator would probably not see a rogue PLC as the signals were faked. As such the physical process and the reported by the PLC behavior would mismatch. For industrial systems information from multiple sources would need to be correlated and possibly collected by multiple independent devices. This might have an effect assuming that no collaboration and common fake reporting of those is in place. Collaboration of machines [16] is a key feature for emerging systems and impersonation efforts like the ones Stuxnet realized might create problematic behaviors that are very difficult to pinpoint and correct.

Network policies must be fine-grained per device and network. Today it is common at least for IT networks that Intranet initiated connections may be allowed to external servers; but not the opposite. This eliminates only part of the potential attack scenarios. In the Stuxnet case all communication was issued from the Intranet towards the extranet. Data was passed as a parameter to an external request, that would be valid e.g. for accessing a web page. Apart from the obvious permission to contact only specific web sites (e.g. contact only a trusted maintenance web site of the provider), additional checks should be done by stateful firewalls and Intrusion Detection Systems (IDS) [17] on the information exchanged and raise flags on detection of misuse. This of course assumes that the the "trusted" provider site we communicate with is not compromised nor that a copycat exists (e.g with stolen valid certificates and DNS redirection).

The bottom line is that security is a multi-angled process where a vulnerability [18] and risk analysis may dictate what is the acceptable level. Solutions focusing asymmetrically on some aspects may give a false sense of safeness and security; which will be shattered by the reality as in the case of Stuxnet.

IV. Considerations for Next Generation of SCADA/DCS

The emerging industrial infrastructure will be a system of systems that heavily relies on individual monitoring and control entities such as the SCADA/DCS systems; the later will undergo some significant changes in the next years [5]. Not only they will be empowered with more processing and communication capabilities, but they will enhance their functionality as part of a collaborative system of systems rather than act in standalone fashion. To this end mainly trends in the current IT industry will play a key role such as (i) the focus on information driven interaction rather than device driven integration, (ii) tighter integration with enterprise systems and realization of distributed business processes, (iii) cross-layer cooperation among systems horizontally and vertically, (iv) virtualization and cloud-computing, (v) wide availability of multi-core systems and GPU computing (vi) existence of SOA-ready devices. These developmental trends point out that the risk area for future SCADA/DCS systems increases rapidly. Hence apart from being vulnerable to well known attacks [19], one has to consider even more complex and combined attacks against these these critical systems.

The aforementioned trends may result to more sophisticated plant infrastructures that will ease management and boost integration – however without the appropriate security considerations the Aeolus' bag will be opened. For instance if each device exposes its functionality as a web service [20], [21] and makes it network-wide accessible, then worms like the Stuxnet may be more successful in taking advantage of security bugs and directly acquire valuable information via monitoring as well as manage the respective devices via the offered interfaces.

It is expected for instance that the HMI is no longer attached to a static location, but accessible anywhere any time from mobile devices [22]. Additionally their functionality will not be monolithic, but composed as a mash-up application from various services [21] hosted on-device and in-network (e.g. in cloud). This may give new communication channels to worms attach-to and interact-with; especially if the connectivity with the outside world over (mobile) Internet is available.

It is envisioned that both monitoring of data as well as the control of the linked processes is done in a collaborative manner [23]. Additionally in-cloud powerful services may deliver high performance analytics on the monitored data and hosted decision support systems may analyze real time data coming not only from the shop-floor but also interconnected business processes. Maliciously interacting with any of these and impersonating the monitored data may result not only on misconfiguration of devices or misbehavior of a single process, but may have enterprise-wide propagated effects. For instance false data may be delivered to the enterprise system from a single plant which in turn is assessed with the result that the enterprise system communicates unrealistic goals (based on acquired "trusted" data) to the rest of world-wide distributed factories (in order to keep up with its internal goals). As such false information or deviating behavior may result to a chain reaction and with global effects.

The future "Perfect Plant-Wide System" [24] will be able to seamlessly collaborate and enable monitoring and control information flow in a cross-layer way [25]. The different systems will be part of a SCADA/DCS ecosystem, where components can be dynamically added or removed and dynamic discovery enables the on-demand information combination and collaboration [20]. All current and future systems will be able to share information in a timely and open manner, enabling an enterprise-wide system of systems [26] that will dynamically evolve. As all systems will be more "fluid" and loosely coupled, we expect an easy upgradeable infrastructure that can co-evolute with the emerging business needs. In this highly complex ecosystem of devices, systems, processes and people, security and trust have to be considered holistically and not at standalone or isolated level. We need cross-domain consideration that will efficiently tackle rapidly emerging risks ranging from simple single device attacks to Advanced Persistent Threats (APTs) such as Operation Shady RAT [27].

V. Conclusion

The problem is that Stuxnet successfully demonstrated the feasibility of a very targeted and highly sophisticated cyber-warfare [28] attack. However Stuxnet's design and architecture are not domain-specific and it can be used as a tool for APTs such as Operation Shady RAT [27]. Hence with some modifications it could be tailored as a platform for attacking other systems e.g. in the automobile or power plants. Its highly sophisticated actions may prevent detection until it is too late. In the hands of criminally inclined groups it may be a very effective cyber weapon with significant impact. The fear that we may have seen only a successful capability demonstration in 2010, is strengthened by the distribution of modern SCADA and PLC systems over the world, the majority of which rely on Europe, Japan and the US. Hence it is imperative to invest on the security as a process [29] by looking holistically the emergent cyber-physical system of systems infrastructures.

Acknowledgment

The authors would like to thank the partners of European Commission funded projects IMC-AESOP (www.imc-aesop.eu) and Cooperating Objects Network of Excellence (www.cooperating-objects.eu) for the fruitful discussions.

References

[1] S. Karnouskos, "Cyber-Physical Systems in the SmartGrid," in *IEEE 9th International Conference on Industrial Informatics (INDIN), Lisbon, Portugal*, 26-29 Jul. 2011.

[2] European Commission DG Information Society and Media, "Monitoring and control: today's market, its evolution till 2020 and the impact of ICT on these," Oct. 2008, workshop presentation. [Online]. Available: http://www.decision.eu/smart/SMART_9Oct_v2.pdf

[3] T. Chen and S. Abu-Nimeh, "Lessons from stuxnet," *Computer*, vol. 44, no. 4, pp. 91–93, 2011.

[4] "Stuxnet." [Online]. Available: https://secure.wikimedia.org/wikipedia/en/wiki/Stuxnet#cite_note-5

[5] S. Karnouskos and A. W. Colombo, "Architecting the next generation of service-based SCADA/DCS system of systems," in *37th Annual Conference of the IEEE Industrial Electronics Society (IECON 2011), Melbourne, Australia.*, 7–10 Nov 2011. [Online]. Available: http://sites.google.com/site/karnouskos/files/2011_IECON_ngSCADA.pdf

[6] N. Falliere, L. O. Murchu, and E. Chien, "W32.stuxnet dossier," Symantec, Tech. Rep., Feb. 2011. [Online]. Available: http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf

[7] A. Matrosov, E. Rodionov, D. Harley, and J. Malcho, "Stuxnet under the microscope," eset, Tech. Rep., 2010. [Online]. Available: http://www.eset.com/resources/white-papers/Stuxnet_Under_the_Microscope.pdf

[8] D. Beresford, "Exploiting Siemens Simatic S7 PLCs," in *Black Hat USA+2011, Las Vegas, NV, USA*, 3–4 Aug. 2011. [Online]. Available: https://media.blackhat.com/bh-us-11/Beresford/BH_US11_Beresford_S7_PLCs_WP.pdf

[9] J. Clark, S. Leblanc, and S. Knight, "Compromise through usb-based hardware trojan horse device," *Future Gener. Comput. Syst.*, vol. 27, pp. 555–563, May 2011. [Online]. Available: http://dx.doi.org/10.1016/j.future.2010.04.008

[10] A. Davis, "USB – undermining security barriers," in *Black Hat USA+2011, Las Vegas, NV, USA*, 3–4 Aug. 2011. [Online]. Available: https://media.blackhat.com/bh-us-11/Davis/BH_US_11-Davis_USB_WP.pdf

[11] L. Duflot, Y.-A. Perez, G. Valadon, and O. Levillain, “Can you still trust your network card?” CanSecWest 2010, Vancouver, Canada, 22–26 Mar. 2010. [Online]. Available: http://www.ssi.gouv.fr/IMG/pdf/csw-trustnetworkcard.pdf

[12] C. Miller, “Battery firmware hacking,” in *Black Hat USA+2011, Las Vegas, NV, USA*, 3–4 Aug. 2011. [Online]. Available: https://media.blackhat.com/bh-us-11/Miller/BH_US_11_Miller_Battery_Firmware_Public_WP.pdf

[13] J. Clark, S. Leblanc, and S. Knight, “Hardware trojan horse device based on unintended usb channels,” in *Proceedings of the 2009 Third International Conference on Network and System Security*, ser. NSS ’09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1109/NSS.2009.48

[14] S. T. King, J. Tucek, A. Cozzie, C. Grier, W. Jiang, and Y. Zhou, “Designing and implementing malicious hardware,” in *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*. Berkeley, CA, USA: USENIX Association, 2008, pp. 5:1–5:8. [Online]. Available: http://dl.acm.org/citation.cfm?id=1387709.1387714

[15] M. Amanullah, A. Kalam, and A. Zayegh, “Network security vulnerabilities in scada and ems,” in *Transmission and Distribution Conference and Exhibition: Asia and Pacific, 2005 IEEE/PES*. IEEE, 2005, pp. 1–6.

[16] P. J. Marrón, S. Karnouskos, D. Minder, and A. Ollero, Eds., *The emerging domain of Cooperating Objects*. Springer, 2011. [Online]. Available: http://www.springer.com/engineering/signals/book/978-3-642-16945-8

[17] R. Barbosa and A. Pras, “Intrusion detection in scada networks,” *Mechanisms for Autonomous Management of Networks and Services*, pp. 163–166, 2010.

[18] N. Cai, J. Wang, and X. Yu, “SCADA system security: Complexity, history and new developments,” in *6th IEEE International Conference on Industrial Informatics (INDIN), Daejeon, Korea*, 13–16 Jul. 2008, pp. 569–574.

[19] W. Gao, T. Morris, B. Reaves, and D. Richey, “On scada control system command and response injection and intrusion detection,” in *eCrime Researchers Summit (eCrime), 2010*, oct. 2010, pp. 1 –9.

[20] S. Karnouskos, D. Savio, P. Spiess, D. Guinard, V. Trifa, and O. Baecker, “Real World Service Interaction with Enterprise Systems in Dynamic Manufacturing Environments,” in *Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management*, L. Benyoucef and B. Grabot, Eds. Springer, 2010, no. ISBN 978-1-84996-118-9, (in press).

[21] D. Idoughi, M. Kerkar, and C. Kolski, “Towards new web services based supervisory systems in complex industrial organizations: Basic principles and case study,” *Comput. Ind.*, vol. 61, pp. 235–249, April 2010.

[22] J. Song, S. Jung, and S. Kim, “Study on the future internet system through analysis of scada systems,” *Communication and Networking*, pp. 10–14, 2010.

[23] Z. Vale, H. Morais, M. Silva, and C. Ramos, “Towards a future scada,” in *Power & Energy Society General Meeting, 2009. PES’09. IEEE*. IEEE, 2009, pp. 1–7.

[24] P. Kennedy, V. Bapat, and P. Kurchina, *In Pursuit of the Perfect Plant*. Evolved Technologist, 2008.

[25] A. W. Colombo and S. Karnouskos, “Towards the factory of the future: A service-oriented cross-layer infrastructure,” in *ICT Shaping the World: A Scientific View*. European Telecommunications Standards Institute (ETSI), John Wiley and Sons, 2009, vol. 65-81.

[26] M. Jamshidi, Ed., *Systems of Systems Engineering: Principles and Applications*. CRC Press, Nov. 2008.

[27] D. Alperovitch, “Revealed: Operation Shady RAT,” McAfee Labs, Tech. Rep., Aug. 2011. [Online]. Available: http://www.mcafee.com/us/resources/white-papers/wp-operation-shady-rat.pdf

[28] J. Farwell and R. Rohozinski, “Stuxnet and the future of cyber war,” *Survival*, vol. 53, no. 1, pp. 23–40, 2011.

[29] K. Stouffer, J. Falco, and K. Scarfone, “Guide to industrial control systems (ics) security,” National Institute of Standards and Technology (NIST), U.S. Department of Commerce, Tech. Rep. NIST Special Publication 800-82, Jun. 2011. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-82/SP800-82-final.pdf