Information Security – Assignment 2

Stuxnet

05/04/2019
Utrecht University

Kasper de Graaff	6281427
Stefan van der Pijl	6201202
Stan Twilt	6240399

Table of Contents

Introduction

In the years between 2000 and 2010, tension was rising globally because of Iran's nuclear efforts. It was discovered that the country had secret uranium enrichment facilities which processed uranium for usage in nuclear bombs. In the 1950s, Iran received an experimental nuclear reactor from the United States under the *Atoms for Peace* program, which would help Iran to produce nuclear energy (Weiss, 2003). This program was later discontinued by Iran's nation's leader as he was against nuclear energy, however 5 years after the program was discontinued, Ruhollah Khomeini (Iran's national leader at the time) restarted the program with the intent of ensuring the country's safety through nuclear weapons (Bruno, 2010). This led to a rising tension and negotiations between Iran and the UN which led to sanctions and a discovery of Iran's plan to build nuclear warheads. Fearing regional war, a military intervention was never done. However, a solution came in a different form: the first ever cyber weapon to be used against another country. In 2010, hundreds of centrifuges used for enriching uranium began to spin out of control, tearing themselves apart (Lindsay, 2013). A virus, later named Stuxnet, had infiltrated the computer systems of the enrichment facilities and sabotaged operations. This was a huge blow to Iran's campaign to develop nuclear weapons and after the sabotaging continuing for some time, forced Iran to the negotiating table where the country signed the *Joint Comprehensive Plan of Action* in which de-escalation of sanctions was promised as well as the dismantling of Iran's nuclear program (Martellini & Zucchetti, 2016). The virus which was responsible for the sabotaging of the nuclear facilities was first discovered later in 2010 by Sergey Ulasen (Lindsay, 2013).

Stuxnet is an incredibly complex piece of malware which allegedly required large amounts of resources to develop. The worm exploits many vulnerabilities, known but also previously unknown. By reprogramming specific controllers named *Programmable logic controller,* Stuxnet was able to alter existing code whilst hiding this from the operators of the system (Falliere, Murchu & Chien, 2011). The specifics of how this is achieved will be further explained in the sections of this paper. By altering the code meant to make the centrifuges containing uranium run smoothly, Stuxnet was able to break centrifuges by making them spin at an increasingly fast rate, which caused them to tear and break down. All this was hidden from the engineers operating the facility as Stuxnet made it seem as though all machines were giving off stable reading. The version of the worm found in the infected Iranian facilities was designed to specifically target the uranium enrichment facilities (Chen & Abu-Nimeh, 2011), most likely with intent to delay the progress towards nuclear warheads. No entity has claimed responsibility of the attack, though many believe the virus to be built by the joint cooperation of the United States and Israel. In 2013, Edward Snowden even claimed that these countries were responsible for the development of the worm (Vombatkere, 2013). Several versions of the worm have spread across the globe infecting approximately 100.000 computers in more than 100 countries worldwide, as can be seen in Figure 1. However, currently this number could be much higher as the report stating this is several years old. Presently, it is believed that these version of Stuxnet present no treat as the virus is designed to only target specific systems and only when specific conditions have been met. As the virus is extremely complex. Attempts to reverse-engineer and repurpose the worm have not been successful but version of Stuxnet which is harmful for all devices, could still surface out of nowhere.

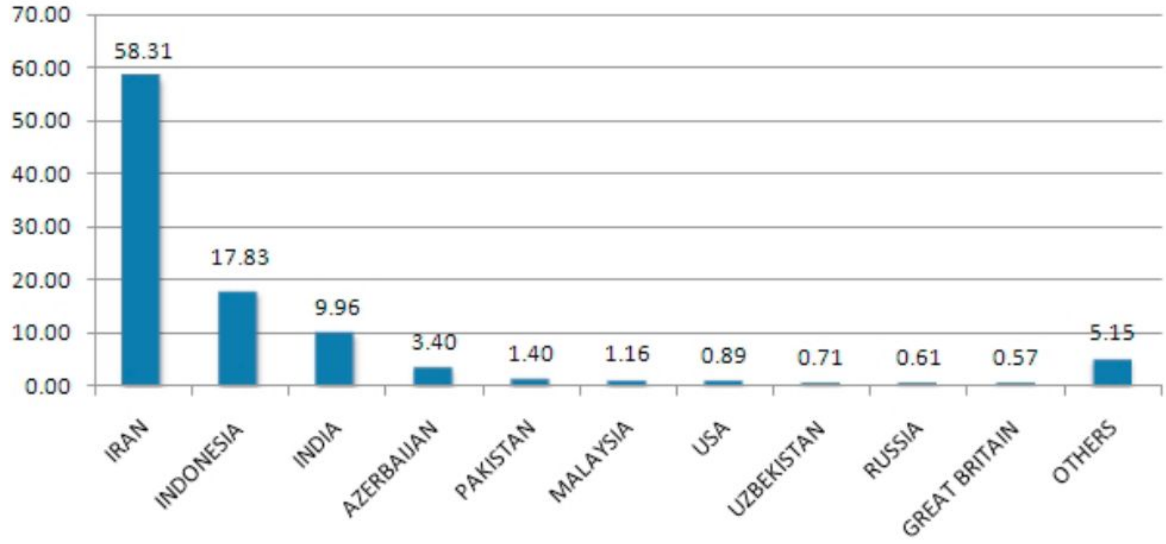

Figure 1: Geographic distributions of infections (Retrieved from Falliere, Murchu & Chien, 2011)

Development of Stuxnet is believed to have been started in 2006, when the United States, specifically NSA, began searching for an option regarding the Iran nuclear problem, in which the could delay the nuclear program enough to force them to negotiate without entering an all out war. Under codename *Olympic games* development started on a piece of malware capable of inflicting physical harm to the Iranian networks (Lindsay, 2013). The development was completed in several stages. During the first stage, a version of Stuxnet would infiltrate the Iranian systems and try to map the whole network. After security details and other relevant data was sent back to the NSA, a more active version of Stuxnet was inserted into Iranian system where it would remain dormant before finally going active. During the time the worm was dormant, it collected data about the readings given off by the uranium centrifuges when operations were normal. This data was sent to the engineers while the virus went active (Karnouskos, 2012). This way, it seemed as though the centrifuges were all operating normally while in reality, they were tearing themselves apart because Stuxnet altered their operating code and made them go haywire.

As stated above, the worm was first discovered by Sergey Ulasen who at first thought the malfunctions caused by the worm were actually just Windows misconfigurations. After looking into the case more and working with his partner in Iran, he later figured out the true nature of the worm (Kaspersky, 2018). He concluded a rootkit was involved as well as that the malware was using zero-day vulnerabilities, which are vulnerabilities not yet found and patched, to spread. After his company spread a report on the virus, the malware started to gain attention. Before Stuxnet itself was discovered and named, older versions of the worm were already found and reported about (Kaspersky, 2018). In 2009, the earliest version of Stuxnet was detected. This version however, did not have signed driver files as opposed to the version Sergey found, which had driver files signed with a valid Realtek Semiconductor Corps certificate (explained in *Background*) and a valid JMicron certificate (Falliere, Murchu

& Chien, 2011). After the news about Stuxnet spread, companies began to monitor the worm and report about it while Microsoft patched several vulnerabilities. It became apparent that, due to the nature and complexity of the worm, the worm had to have been developed by a huge and influential party, possibly even a nation. The theory of a nation creating the virus gained even more credibility when it was found that the worm only became active when situated on a programmable logic controller (also, explained in *Background*) found in Iranian facilities (Kaspersky, 2018). And later, as mentioned above, Snowden claimed the United States and Israel were behind Stuxnet.

Background

In this section, information is given about some basic aspects regarding malware and vulnerabilities. This will provide a clearer picture of the environment in which Stuxnet was developed and deployed. Stuxnet as a piece of malware behaves like a worm. This means the malware spreads by making copies of itself and spreading these in a computer network (Rieck, et al., 2008). This works well in the targeted networks of the Iranian uranium enrichment facilities as these networks are large and separated from other networks via an air gap. This means the computer systems are completely closed of from other networks such as the internet. Therefore, the operators behind Stuxnet would need to physically get a copy of the worm in the network, for example through an infected USB drive (Lüders, 2011). The worm also used several zero-day vulnerabilities to access and spread through the computer networks. These vulnerabilities are called this way because the vulnerabilities have never before been noticed. This makes these vulnerabilities very powerful and expensive. Stuxnet exploiting several of these indicates that a lot of resources went into developing the worm (Falliere, Murchu & Chien, 2011).

Stuxnet has a rootkit component of which the purpose will be described in the section *Description*, however what is a rootkit? A rootkit is an, often malicious, piece of code which is able to access areas of computer software which is not otherwise accessible (Davis, Bodmer & LeMasters, 2009). The code often tries to hide its or other code's presence from the user and Antivirus software. This can be done because the code has administrator access which allows it to have full control of a computer of system. Because of this, rootkit malware is very hard to detect and remove (Davis, Bodmer & LeMasters, 2009). Stuxnet has both user-mode and kernel-mode rootkit functionality (Falliere, Murchu & Chien, 2011). These modes represent levels of access the code has. In user-mode, code is separated from hardware by system API's which makes crashes at this level recoverable. In, kernel-mode, executing code is able to directly access hardware and memory. This level of access is only given to the most trusted functions as crashes at this level will stop the entire computer. Stuxnet files were signed by a certificate of a trusted company, first a certificate from Realtek Semiconductor Corps and later when this certificate was removed, a certificate from JMicron (Falliere, Murchu & Chien, 2011). These certificates are used to give a piece of software credibility by showing that a company trusts this software. Having the drivers signed by certificates showed all other software that Stuxnet was a very 'trustworthy' piece of software, after all the drivers were signed by certificates of trusted companies.

Stuxnet was the first piece of malware to have a Programmable Logic Controller rootkit (Falliere, Murchu & Chien, 2011). This meant the target of Stuxnet was finding these so-called PLC's and hiding themselves in their environment. Stuxnet specifically searching for and then targeting PLC's, leads software security companies to believe that Stuxnet had a very clear purpose to inflict damage on these systems. Because of this, Stuxnet is called the first ever cyberweapon. The PLC's Stuxnet was targeting were responsible for the smooth operation of the centrifuges which were enriching uranium. But what is a PLC exactly? A PLC is an industrial computer which is in charge of running industrial processes like manufacturing. These computers have been developed carefully to ensure they can withstand extreme conditions. PLC's are vital for many operations and must be fully functional at all times.

PLC's often operate in combination with SCADA (supervisory control and data acquisition) systems (Srivastava, et al., 2012). These systems are developed by Siemens and are used to monitor and control physical processes which occur in some industrial practices. In the Iranian nuclear enrichment facilities, these SCADA's were able to program the PLC's which were linked to the centrifuges. Stuxnet sought out these SCADA's and infected them. Via these systems, Stuxnet was able to infect the PLC's which the SCADA's were in control of, and were thus able to gain control over the full functionality of the centrifuges.

To give some more context into the operations that were done in the Iranian nuclear enrichment facilities, we will shortly analyse how the centrifuges were able to 'enrich' the uranium. Uranium has three naturally occurring isotopes of which two are important in the current context; U-235 and U-238. Of all naturally occurring uranium, only .7% is u-235. However, to create a nuclear warhead, approximately 90% of the used uranium needs to be isotope U-235. The increase the percentage of U-235 present and decrease the amount of U-238 present, centrifuges are used. These centrifuges use the difference in mass the two isotopes have to split the uranium. U-238 has a slightly higher mass and is pushed to the edge of the spinning centrifuge while U-235 remains close to the middle. The U-235 is then siphoned out of the centrifuges and this process is repeated until the percentage of U-235 present in the uranium is high enough. The centrifuges doing this are spinning at a very high frequency and are engineered in such a way that fluctuations of in the spinning frequency could severely damage the centrifuges (Villani, 1979).

Related work

“Malware such as Stuxnet can affect critical physical infrastructures that are controlled by software, which implies that threats might extend to real lives.”(Chen & Abu-Nimeh,2011)

Stuxnet as Pioneer

Stuxnet is believed to be the first real cyberwarfare weapon (Falliere, Murchu & Chien, 2011). Stuxnet was not the first targeted attack against industrial systems, but the first one to receive worldwide attention due to its unique purpose as a cyber weapon (Bencsáth et al, 2012b). Chen & Abu-Nimeh (2011), argue that malware that is similar to Stuxnet might be used as a first strike weapon. This means that it can be used to disable or considerably weaken the adversary before an engagement in the open takes place. According to Chen & Abu-Nimeh (2011), Stuxnet served as an eye opener for many security researchers. Where previously it was thought that malware was only a threat to computers and computer networks it is now evident that it can also affect physical objects, in Stuxnet’s case, the Iranian uranium centrifuges. Farwell & Rohozinski (2011) argue that Stuxnet is not as revolutionary as it is claimed to be. They state that it is not as stealthy as the average advanced malware that criminals use due to the fact that it uses a DNS based command-and-control network. In addition to this the “Air gap” jumping is dubbed as old news since it was already done a few times before. Although the techniques used may not have been revolutionary according to some experts like Farwell & Rohozinski, the use and combining of the techniques are regarded as such by other experts like Faliere et al and Chen & Abu-Nimeh. Chen & Abu-Nimeh (2011) state that Stuxnet might not only be a pioneer in the field of altering physical constructions with the use of malware, but also showed that isolating a network from the internet is not an effective defence. In addition to this, Denning (2012) states that Stuxnet will most likely accelerate and serve as a building block or blueprint for the development of new cyber-weapons that target ICS devices.

*“Stuxnet could be a forbearer of the way nation-states use cyber-warfare,
offering militaries a weapon that may be morally superior to a kinetic one, such as a bomb,
when it incurs less harm and risk than the kinetic weapon while achieving the same
objective.” (Denning, 2012)*

Stuxnet as Blueprint

Due to the success of Stuxnet in taking out physical infrastructure and infecting and monitoring a big network without being noticed for some time, it served as an example of how to create malicious software with the intend to destroy physical infrastructure as well as an example of how to use a cyber weapon with the intent to weaken a country, group or individual. A number of other malware were found which share a few characteristics with Stuxnet. Some of these malware are explained below.

Duqu

Duqu is an information stealing rootkit that is most likely made by the same people that created Stuxnet (Bencsáth et al, 2012a). The malware was discovered in 2011 and was named after the property where it stores stolen data in file names starting with "DQ" (Wangen, 2015). The similarity with Stuxnet is that both malware are constructed in a modular way and both can both be remotely re-configured from a command and control server to include any kind of functionality. Because of this Duqu is extremely similar to Stuxnet in its design, complexity and implementation (Bencsáth et al, 2012a). In addition to this a lot of code from Stuxnet is believed to be reused in Duqu (Bencsáth et al, 2011). The difference between the two is that they both had different objectives, where Stuxnet was meant to ultimately destroy physical infrastructure, Duqu was meant to steal information.

Flame

Flame, also known as SKyWIper and Flamer, was discovered in 2012. The size of Flame is about 20 Mb because of this Flame is arguably one of the largest malware ever build (Wangen, 2015). Flame is also an information stealing malware. It can do this in multiple ways, much like Duqu. It is thought that flame may have been used as a probing malware before the use of Stuxnet. The idea is that the United States and Israel could already monitor the situation at the Iranian nuclear plants. This means that Flame could have been created by the same people that developed Stuxnet (Nakashima et al, 2012). Other researchers like Bencsáth et al (2012b) state that it is unlikely that it is build by the same people since it differs significantly in architecture and size from Stuxnet. In addition to this it is observed that some of Flame's modules were created using another language (LUA) than all of the Stuxnet modules.

Gauss

Gauss is another information stealing malware that is related to Stuxnet. Gauss shows many similarities with Flame since many modules share similar content and architecture. In addition to this, Gauss has similar targets as Flame and Duqu. The difference lies in the fact that were Flame and Duqu aimed at monitoring and capturing information from an infected system, Gauss also tries to capture and steal both banking and social media credentials (Bencsáth et al, 2012b). Another difference can be found in the replication of the malware, where Flame and Duqu are self-replicating, Gauss is presumed to be not self-replicating (Wangen, 2015). Another difference between Gauss and Stuxnet is that some of Gauss's modules have been encrypted in a way that it can only be decrypted on the targeted system, because of this it can currently not be analysed (Bencsáth et al, 2012b).

Description

The initial infection of Stuxnet in the nuclear facilities at Natanz and elsewhere in Iran was done by injecting an infected USB into pc's within the network of the facility. This is thought because of the fact that most nuclear facilities are air gapped (blocked from internet access) (Lüders, 2011).

Stuxnet is a sophisticated worm with a complex architecture. The main part of the malware is a large .dll file. This file consist of most of the exports and resources used by the worm. Next to this .dll file there are multiple encrypted configuration blocks (Mueller et al, 2012). Around this .dll file and the configuration blocks is a wrapper named 'stub'. When Stuxnet is executed the .dll file is extracted from the stub, the file places itself into the memory like it is a module and calls an export from the .dll file. When the .dll file is loaded for the first time Stuxnet will be installed on the computer. An export is called that is responsible for checking the operating system since Stuxnet only works on Windows, checking if the computer is already infected and controlling what the protection mechanisms of the PC are. (Falliere, Murchu & Chien, 2011) .

Once Stuxnet has successfully infiltrated the target PC it hides itself by using rootkit functionalities. A rootkit is software that enables other software to be installed in places where it is not allowed. The rootkit of Stuxnet uses stolen certificates from the companies REALTEK and JMicron. These certificates grant Stuxnet the rights to access normally protected areas of the operating system (Lüders, 2011). If this has not lead to Stuxnet having administrator rights on the system it will use a zero day exploit to try to obtain these rights. Stuxnet has two different zero day exploits to obtain administrator rights, It uses the *Task Scheduler Escalation of Privilege vulnerability* if the operating system of the machine is Windows Vista, Windows 7, or Windows Server 2008. Is this not the case then it will use the Windows *in32k.sys Local Privilege Escalation vulnerability (MS10-073)* zero day vulnerability to obtain administrator rights.

Stuxnet is highly selective about its choice of targets. It looks specifically to infiltrate Windows computers that are connected to particular programmable logic controllers (PLC) that are made by the company Siemens. Specifically, Stuxnet looks for the WinCC/PCS 7/ STEP7 SCADA software that is running on the PLC. This is supervisory control and data acquisition software (Falliere, Murchu & Chien, 2011)(Karnouskos, 2012).

Once Stuxnet has infected a computer within the target organization it will start spreading to other computers in search of a computer that has the SCADA software running. Most of the computers within nuclear facilities are air gapped. To spread to other computers Stuxnet uses the *Microsoft Windows Print Spooler Service Remote Code Execution Vulnerability (BID 43073)* zero day vulnerability. It also spreads by infecting removable drives like USB's (Lüders, 2011).

If Stuxnet reaches a computer that is connected to the internet it contacts a command and control server. There are two servers, one located in Denmark and the other in Malaysia, these servers are used to send data to the makers and controllers of Stuxnet (Lüders, 2011). Stuxnet has functionality to update itself once it discovers a newer version of itself. If a infected machine gets infected a second time with a newer version it will update itself and update Stuxnet on the connected computers (Mueller et al, 2012).

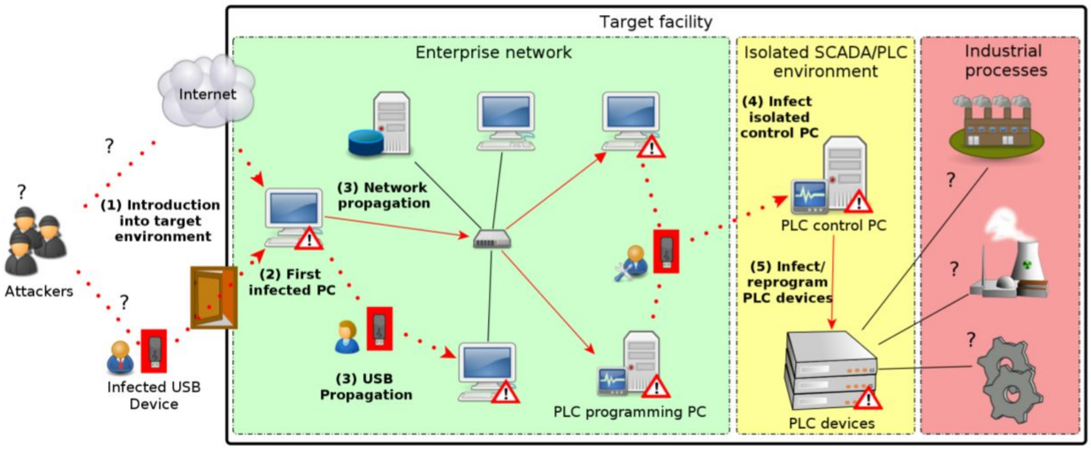

Figure 2: Infection, spreading and attack on the PLC software. (Retrieved from Brunner et al, 2010)

Upon reaching a computer that has the SCADA software running it. Stuxnet will place a copy of itself in the STEP 7 project. This will increase the attack vector if the project is copied and transferred to another pc running SCADA software. Figure 2 shows how Stuxnet spreads in search of a computer with SCADA software (Lüders, 2011) (Karnouskos, 2012).

Next Stuxnet adds a module to the s7 communication .dll files. By doing this it can act as a man in the middle and alter the information flow from the PLC to the SCADA software. This means that a operator will notice nothing wrong when Stuxnet start attacking the PLC’s. This man in the middle attack is shown in figure 3 (Karnouskos, 2012).

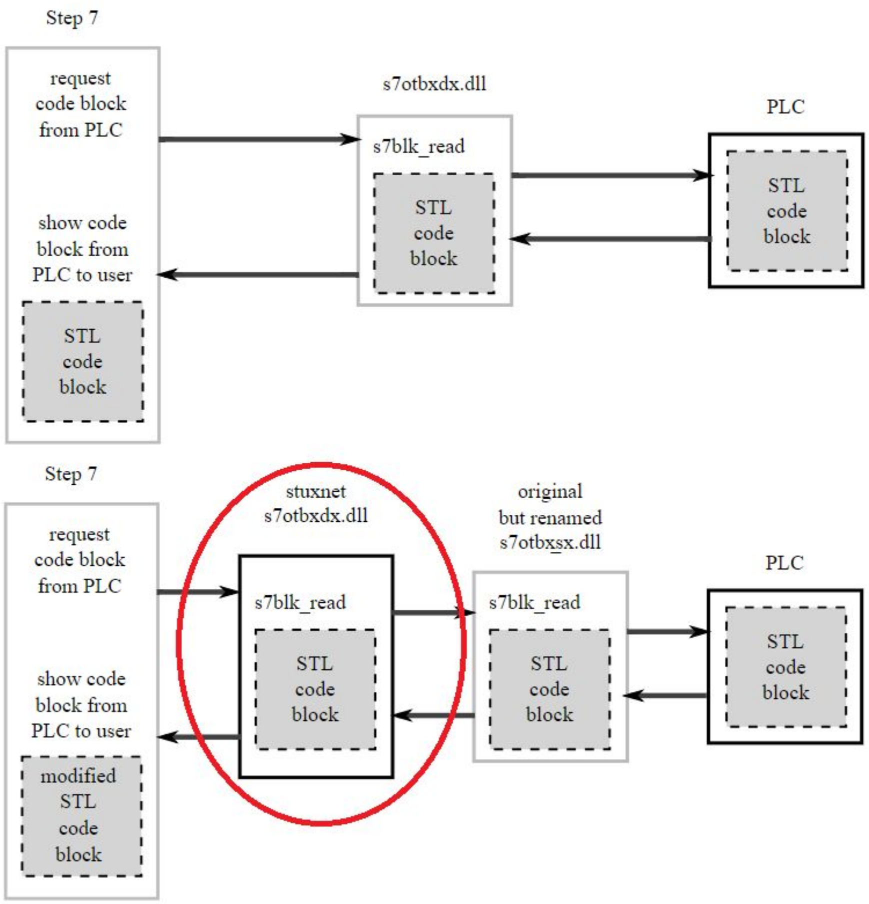

Figure 3: Altering of the Siemens s7 communication with the PLC by Stuxnet (Retrieved from Lüders, 2011)

Finally Stuxnet will, over the course of a few months, start altering the hertz at which the motors of the centrifuges are controlled. It will lower the amount of hertz from 1400 Hz to a few hundred hertz. It will repeat this process multiple times which will cause the centrifuges to wear down more quickly. The cause of this faster than normal wear down was very difficult to find for the engineered, this was due to the man in the middle controlling and altering the dataflow from the PLC to the SCADA systems (Lüders, 2011).

Mitigation

Upon discovering the presence of the worm in some of the systems that were screened by the Belarusian security company VirusblokAda, they immediately raised alarm. The international security society started exchanging all available information about Stuxnet so that more exposure was generated to help prevent further infections by the virus. As mentioned before, the power of Stuxnet to infiltrate and spread within a network was due to the fact that it contained 4 zero day exploits and used 2 certificates (Chen & Abu-Nimeh, 2011).

Upon discovering that Stuxnet operated under 2 certificates, VeriSign (a company responsible for digital certificates) revoked the Realtek certificate on the 16th and the

Jmicron Technology certificate on the 22th of June 2010. These two companies are located close to each other witch leads to the speculation that the certificates are physically stolen from these companies.(Chen & Abu-Nimeh, 2011).

Zero days

Stuxnet made use of the following 4 zero day vulnerabilities. MS10-046, MS10-061, MS08-067, MS10-073. To mitigate these vulnerabilities Microsoft issued different patches.
MS10-046 was mitigated with a Windows update on August 2 2010. On 14 September 2010 Microsoft releases a patch to mitigate the MS10-061 vulnerability that was found in August of that year. MS10-073 was a zero day vulnerability that was used for local privilege escalation, it was patched by Microsoft on 12 October 2010. The last vulnerability that was used was MS08-067. This vulnerability was already patched in 2008 by Microsoft. But because of the slow update cycle of SCADA systems it is thought that many infected machines still were damaged by this vulnerability (Falliere, Murchu & Chien, 2011)

Siemens patches

On 19 July 2010 Siemens reported that they were investigating different reports of malware targeting the SCADA systems that they developed. The SCADA systems from Siemens come with two defence systems, namely a local firewall and an intrusion detection system. The firewall's functionality was based on an IP access list. Only the IP address of remote devices that want to communicate with the SCADA software that are on the list are granted access. Any other connection is not allowed (Lüders, 2011).

The intrusion detection systems used quasi-unique numbers, computer generated hashes that change drastically with any altering of the PLC's or the hardware that is connected. These defences were found to be insufficient for the Stuxnet worm. This is due to the man in the middle functionality that it used (Lüders, 2011).

To mitigate the vulnerabilities of Stuxnet, Siemens has released a SCADA security update. This update consisted of the following three parts. A workaround for the .LNK vulnerability, improvements on the SQL authentication and scanning SCADA projects for possible infections (GOVCERT.NL, 2011). In addition to the security patches, Siemens recommends using the anti-virus products from TrendMicro, Symantec or McAfee. These antivirus programs can find Stuxnet (GOVCERT.NL, 2011).

Air gap

The nuclear facilities that were targeted by Stuxnet were air gapped. This means that the facility was not connected to the internet. This is done to protect the facility against cyber-attacks from the internet. An effect of this is that updates to the system are often performed from a USB drive that is brought from outside. It is speculated that a USB that had the task of updating the system caused the initial infection. These updates can only be performed by certified personnel, but have nevertheless led to infection. This has had the effect of prioritizing hardware safety alongside software security (Farwell & Rohozinski, 2011).

Impact

The discovery and activation of Stuxnet led to political consequences both on the short as the long term. In this chapter these effects will be addressed.

Short-term Impact (5 years after discovery)

The original target of Stuxnet, The Iranian nuclear program, suffered greatly from the effect of Stuxnet. The nuclear program was hindered badly. It is estimated that the attack set the Iranian program back by at least 2 years (Shakarian, 2011). According to Denning (2012) around 1000 of the total 9000 centrifuges were destroyed by Stuxnet. This resulted in Iran being forced to make concessions about their nuclear ambitions through signing the Joint Comprehensive Plan of Action.

Although this resulted in the nuclear threat of Iran decreasing, it increased the stress on international relations since Iran suspected that Israel and the United states where behind the attack. This urged Iran to gather an army of IT specialists themselves to defend against further cyber-attacks spearheaded by the United States and Israel. Currently Iran is considered an emerging power in cyber warfare (Berman, 2013). With a cyber army that allegedly has the size of 120000 IT specialists, Iran claims to have the second largest cyber army in the world and with the budget for the Supreme Cyber Council ever increasing (a 600% increase between 2013 and 2015), Iran has stepped up their cyber defences to meet future attacks (Craig & Valeriano, 2016).

Ever since the discovery of the Stuxnet worm a hacker collective called “the Iranian cyber army” (not to be confused with the actual Iranian cyber army) expanded rapidly in size and in activities. This hacker collective engages in attacking internet services like Twitter and rerouting their internet traffic (Rezvaniyeh, 2010). In addition to this they take down websites which are regarded as hostile to Iran. Although the Iranian government has never recognized the group officially it is believed that the group is in fact coordinated and supported by the government (Berman, 2013).

It is also believed that Iran was behind several cyber-attacks on banks and government systems of both the United States and Israel. Iran has in many cases denied being involved in such actions. One of these alleged attacks started in late 2011 and continued in 2012, the attack was aimed at taking out the webservices of a number of American banks. The United States pointed the finger to Iran but Iran has denied any involvement (Hosseinian, 2012)(Berman, 2013).

Long-term Impact (10 years after discovery and beyond)

Cyberweapon

Stuxnet has been credited as being the first ever cyber weapon. This is mostly due to Stuxnet's ability to destroy physical infrastructure. This is seen as a huge gamechanger for waging cyber warfare. Before Stuxnet, it was believed that the reach of malware was contained within the computer or network it infected, but now it became ever more clear that malware could modify or damage machinery or other physical infrastructure. Stuxnet served as an eye opener for the security industry because now malware was not bound to manipulating or destroying computers or stealing information but could now also manipulate or destroy infrastructure (Chen & Abu-Nimeh, 2011). This development may lead to other governments or even criminal organizations developing similar tools to take out their opponents' infrastructure or machinery.

Political

Due to the long lasting conflict between the United States and Israel on one hand and Iran on the other side about Iran's nuclear ambitions it becomes clear that this development lead to an even more hostile attitude from Iran to the United States and Israel. In the larger scheme of things the Stuxnet attack may have further destabilized the situation in the middle east. In addition to this the Stuxnet incident increased the pace in which the cyber arms race takes place. As said in the short term impact section of the paper, Iran invests heavily in its own cyber force to meet future attacks. This will likely continue as it is probable that another attack will take place if Iran continues to be vague about their nuclear activities.

Cyberweapon

In the wake of the success of Stuxnet, perhaps other nations, states or even criminal organisations might try to replicate the functionality that Stuxnet contained. Being able to control SCADA appliances is a very strong weapon since a lot of automated entities, e.g. railway systems, machinery and oil pipelines, depend on SCADA to operate. Stuxnet paved the way for other organisations to do exactly that. Some experts state that Stuxnet is the first of many attacks that will take place in cyberspace it is thought to mark the beginning of the Cyberwar era (Farwell & Rohozinski, 2011).

"For cyber war, the future is now"(Farwell & Rohozinski, 2011)

References

1. Bencsáth, B., Pék, G., Buttyán, L., & Félegyházi, M. (2012, April). Duqu: Analysis, detection, and lessons learned. In *ACM European Workshop on System Security (EuroSec)* (Vol. 2012).

2. Bencsáth, B., Pék, G., Buttyán, L., & Félegyházi, M. (2011). Duqu: A Stuxnet-like malware found in the wild. *CrySyS Lab Technical Report*, *14*, 1-60.

3. Bencsáth, B., Pék, G., Buttyán, L., & Felegyhazi, M. (2012). The cousins of Stuxnet: Duqu, flame, and gauss. *Future Internet*, *4*(4), 971-1003.

4. Berman, I. (2013). The Iranian Cyber Threat, Revisited. Statement before the US House of Representatives Committee on Homeland Security Subcommittee on Cybersecurity, Infrastructure Protection, and Security Technologies, 2.

5. Brunner, M., Hofinger, H., Krauß, C., Roblee, C., Schoo, P., & Todt, S. (2010). Infiltrating critical infrastructures with next-generation attacks. Fraunhofer Institute for Secure Information Technology (SIT), Munich.

6. Bruno, G. (2010). Iran’s nuclear program. *Council on Foreign Relations*, *10*.

7. Chen, T., & Abu-Nimeh, S. (2011). Lessons from Stuxnet. *Computer*, *44*(4), 91-93.

8. Craig, A., & Valeriano, B. (2016, May). Conceptualising cyber arms races. In 2016 8th International Conference on Cyber Conflict (CyCon) (pp. 141-158). IEEE.

9. Davis, M., Bodmer, S., & LeMasters, A. (2009). Hacking exposed malware and rootkits. McGraw-Hill, Inc..

10. Denning, D. E. (2012). Stuxnet: What has changed?. Future Internet, 4(3), 672-687.

11. Falliere, N., Murchu, L. O., & Chien, E. (2011). W32. Stuxnet dossier. *White paper, Symantec Corp., Security Response*, *5*(6), 29.

12. Farwell, J. P., & Rohozinski, R. (2011). Stuxnet and the future of cyber war. *Survival*, *53*(1), 23-40.

13. GOVCERT.NL. (2011). Stuxnet - een geavanceerde en gerichte aanval (Versie 2.4 – 21 januari 2011). Retrieved from https://www.ncsc.nl/binaries/content/documents/ncsc-nl/actueel/factsheets/factsheet-over-Stuxnet---een-geavanceerde-en-gerichte-aanval/1/Factsheet%2BStuxnet%2B%2B%2Been%2Bgeavanceerde%2Ben%2Bgerichte%2Baanval.pdf

14. Hosseinian, Z. (2012, September 23). Iran denies hacking into American banks. Retrieved April 4, 2019, from https://www.reuters.com/article/us-iran-cyberattacks-denial/iran-denies-hacking-into-american-banks-idUSBRE88M06O20120923

15. Kaspersky, E. (2018, March 16). The Man Who Found Stuxnet – Sergey Ulasen in the Spotlight. Retrieved April 4, 2019, from https://eugene.kaspersky.com/2011/11/02/the-man-who-found-Stuxnet-sergey-ulasen-in-the-spotlight/

16. Karnouskos, S. (2011, November). Stuxnet worm impact on industrial cyber-physical system security. In IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society (pp. 4490-4494). IEEE.

17. Lindsay, J. R. (2013). Stuxnet and the limits of cyber warfare. *Security Studies*, *22*(3), 365-404.

18. Lüders, S. (2011). Stuxnet and the impact on accelerator control systems. Proceedings of ICALEPCS2011, Grenoble, France, 1285-1288.

19. Martellini, M., & Zucchetti, M. (2016). The Iranian Nuclear Agreement: A Scientifically Reliable, Transactional and Verifiable Joint Comprehensive Plan of Action. In Nuclear Non-Proliferation in International Law-Volume III (pp. 471-488). TMC Asser Press, The Hague.

20. Mueller, P., & Yadegari, B. (2012). The Stuxnet worm. Département des sciences de linformatique, Université de lArizona. Recuperado de: https://www2. cs. arizona. edu/~ collberg/Teaching/466-566/2012/Resources/presentations/topic9-final/report. pdf.

21. Nakashima, E., Miller, G., & Tate, J. (2012). US, Israel developed Flame computer virus to slow Iranian nuclear efforts, officials say. *The Washington Post*, *19*.

22. Nicholson, A., Webber, S., Dyer, S., Patel, T., & Janicke, H. (2012). SCADA security in the light of Cyber-Warfare. *Computers & Security*, *31*(4), 418-436.

23. Shakarian, P. (2011). Stuxnet: Cyberwar revolution in military affairs. MILITARY ACADEMY WEST POINT NY.

24. Rezvaniyeh, F. (2010, February 26). Pulling the Strings of the Net: Iran's Cyber Army. Retrieved April 4, 2019, from https://www.pbs.org/wgbh/pages/frontline/tehranbureau/2010/02/pulling-the-strings-of-the-net-irans-cyber-army.html

25. Rieck, K., Holz, T., Willems, C., Düssel, P., & Laskov, P. (2008, July). Learning and classification of malware behavior. In International Conference on Detection of

Intrusions and Malware, and Vulnerability Assessment (pp. 108-125). Springer, Berlin, Heidelberg.

26. Singer, P. W. (2015). Stuxnet and its hidden lessons on the ethics of cyberweapons. *Case W. Res. J. Int'l L.*, *47*, 79.

27. Srivastava, M. D., Prerna, S. S., Sharma, S., & Tyagi, U. (2012). Smart traffic control system using PLC and SCADA. International Journal of Innovative Research in Science, Engineering and Technology, 1(2), 169-172.

28. Villani, S. (1979). Uranium enrichment.

29. Vombatkere, S. G. (2013). Edward Snowden's Wake-Up Call–Cyber Security, Surveillance and Democracy. Countercurrents. org.

30. Wangen, G. (2015). The role of malware in reported cyber espionage: a review of the impact and mechanism. *Information*, *6*(2), 183-211.

31. Weiss, L. (2003). Atoms for peace. *Bulletin of the Atomic Scientists*, *59*(6), 34-44.