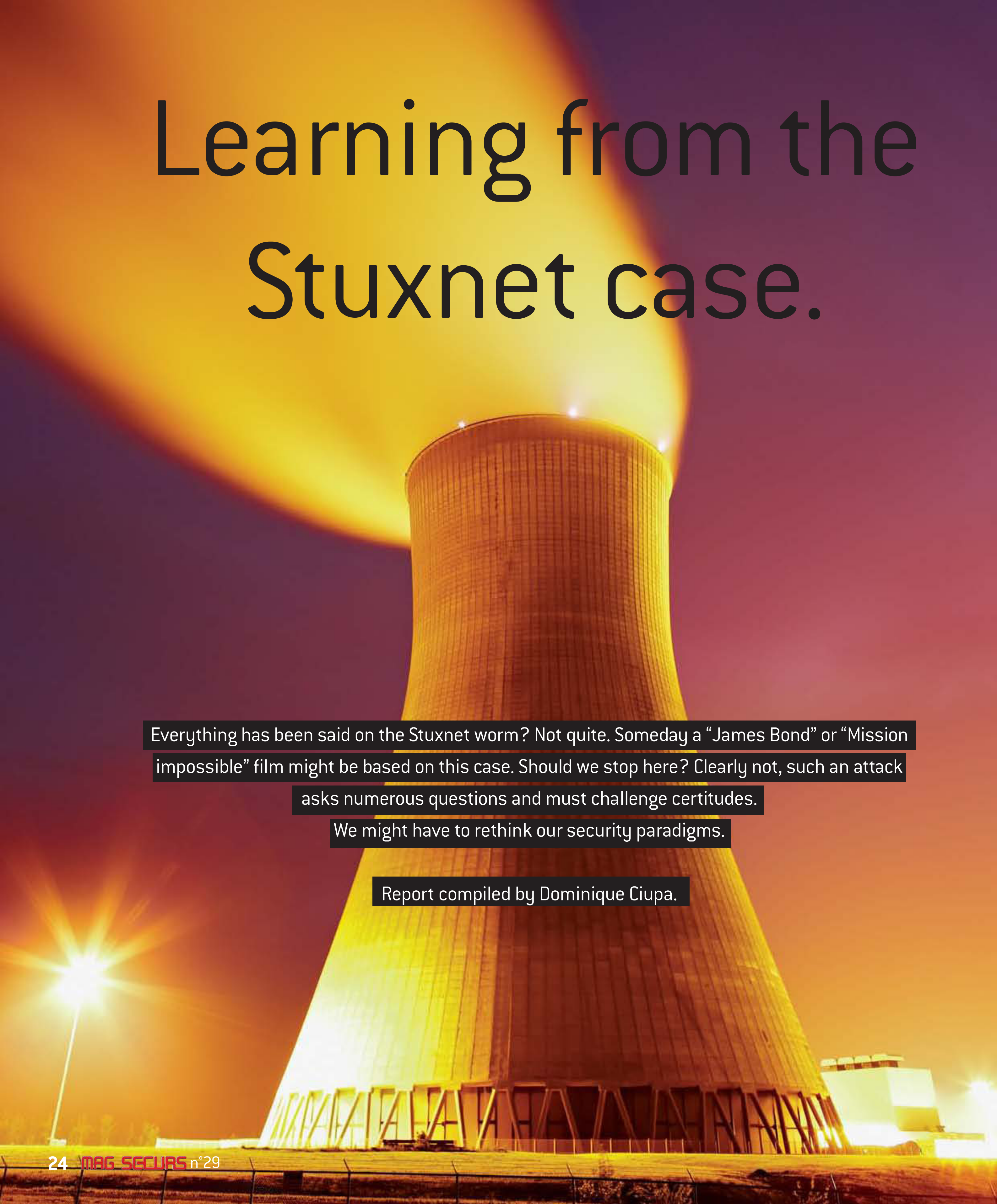
Article published in Mag-Secürs No. 29

LE MAGAZINE DE LA SÉCURITÉ INFORMATIQUE
MAG SECURS
INFORMATIQUE • RÉSEAUX • TELECOM • INTERNET
TIRER LES LEÇONS DE
L'AFFAIRE STUXNET
> TENDANCES
Chine : l'Empire du Hacking
> CERTIFICATION
Table ronde avec 10 acteurs du PCI-DSS
> FORMATION
Telecom Paris Tech



Learning from the Stuxnet case.

Everything has been said on the Stuxnet worm? Not quite. Someday a “James Bond” or “Mission impossible” film might be based on this case. Should we stop here? Clearly not, such an attack asks numerous questions and must challenge certitudes. We might have to rethink our security paradigms.

Report compiled by Dominique Ciupa.

The Barings Bank bankruptcy, in 1995, triggered by Nick Leeson with a £ 827 millions loss was the theme of the 1999 film “Rogue Trader” with Erwan McGregor. At the beginning of 2008, we find that the “Société Générale” (a French Bank) escaped narrowly bankruptcy with the “Jérome Kerviel” case: € 50 billions exposure and almost € 5 billions loss[*]. The fall of 2008 financial turmoil gave us a completely new pace: the American “subprime” crisis involves $ 500 billions of fictitious assets! And scandals are not over: Madoff with $50 billions or even the bailout of Ireland with € 85 Billions, etcΣ

Stuxnet could well be to the Information System Security (ISS) what Nick Leeson has been to the financial system: the first episode in a long series whose effects could ultimately be devastating. The risk for industrial systems has been known for many years and is regularly revealed in major conferences like Black Hat, the RSA conference or FIC (Forum International sur la Cybercriminalité). But it was clear that this risk was considered as unconfirmed: the attack had not yet occurred... Therefore, the temptation to classify this risk as residual and acceptable without really assessing its impact is strong... The media coverage of the Stuxnet attack has at least the merit of firing of an extensive awareness campaign on safety of industrial systems.

> *The attack factors*

We must remain very cautious on what we know about the true aims of this worm. A general trend emerges, however, to assume that this malware was designed to destroy the centrifuges at the Natanz, Iran, uranium enrichment plant. Facts have indeed been reported by the IAEA on delays of this program and the Iranian government has himself acknowledged the existence of problems. Executable code to command centrifuges has been modified to change their rotation speed and destroy them. It would thus be an attack on Siemens WinCC monitoring systems, which control, on a Windows computer, the SCADA (Supervisory Control and Data Acquisition) systems.

The goal of this case was therefore to stop or at least to seriously slowdown the Iranian nuclear program. It would have also had the support of one or more countries, the United States and/or Israel, significant intellectual resources and also probably the work of traditional field agents.

This kind of attack doesn’t however look like the worms we have experienced in the early 2000 that were spreading on the internet. The equipments of a nuclear plant of enrichment plant are indeed unreachable from the Internet. Specific network are designed and totally compartmentalized. It is even common to implement “diode-firewalls” allowing measurement equipment to send data to a control room, without the possibility to send back commands to this equipment and disturb it, or even change its executable code.

Those principles are widely distributed and are specified in documents from the American Nuclear Energy Institute, including document NEI0404. The attack is said to have been conducted with a USB key. The human dimension has been employedΣ How exactly and to what level? Things aren’t clear: a local agent acting deliberately? Upstream contamination of executable programs such that authorized technicians then unknowingly compromise the equipments? Since several facilities have been compromised worldwide, the second scenario seems more plausible ...

According to the specialized Israeli military intelligence publication Debka, Iranian professor Majid Shahriari, in charge of the fight against Stuxnet, was murdered last November. The procedure was to throw explosives from a motorcycle and then to shoot from a car. Iranian government immediately accused the United States and Israel, confirming the murder of the scientist Σ

The analysis of the Stuxnet worm has been performed by many experts and we have seen very important information sharing amongst experts and anti-malware vendors. A comprehensive report has been produced by Symantec. Experts have identified the use of 4 « zero-day » exploits. The execution of an arbitrary payload, made possible by exploiting the unpatched LKN flaw, allowed to compromise the system by running malicious code from a USB key with the use of a .ink link. For the entire profession, the combination of 4 exploits represents an exceptional work, never seen up to now. Symantec explains that the motor frequency control system, between 807 Hz and 1210 Hz

[*] :Read Mag-Securs n°19 (2e trimestre 2008) – Affaire Société Générale : quelles leçons en tirer ?

however, shows that the greatest number of attacks were very clearly located in Iran, far ahea was targeted. Experts note that the attack also take advantage of the use of a default password. WinCC / PCS7 makes indeed use of a MS SQL database which requires an internal communication password. The password verification doesn't concern the system user and Siemens recommends to its customers not to change the password to prevent malfunctionsƐ

The worm study also shows that two certificates were stolen from JMicron and Realtek. The system indeed checks executable code authenticity from a certification authority: Verisign. But the modified executable code had original certificates and the certification authority recognized them as valid. How were the certificates stolen? Infiltration, commandos, spies, bribery, Ɛ The story doesn't tell it yet, but many films show thisƐ

Experts believe that this worm required the work of a 6 to 10 persons team for 6 months to a year. The code analysis also shows a peculiar element. It contained a file named "Myrthus", which means "myrtle" in English. However in the bible the Myrthus was a symbol of justice: "Instead of the thornbush will grow the juniper, and instead of briers the myrtle will grow. This will be for the LORD's renown, for an everlasting sign, that will endure forever." Other experts have seen an allusion to the Book of Esther, and therefore the Torah, «She was called Hadassah because the upright are called thus» Hadassah is one of the name of the Esther Queen and means myrtle. The book of Esther explains how the Queen Hadassah foiled the Persian attacks aimed at destroying the Jewish people.

Another detail from the code analysis, the worm will stop working on June 23, 2012. Experts have noted that it is exactly 100 years after the birth of Alan Turing, famous for his work on computers but also for leading a cryptanalysis team during World War II in Bletchley Park. He was able to decode German communications and played a considerable role in the Allied victory against the Nazi regime ...

Unanimity is nonetheless not reached amongst ISS experts on the planet. In Israel there are specialists who criticize a communication campaign hostile to their country and minimize the stuxnet capabilities. In France, Daniel Ventre, engineer at CNRS and director of the collection "Cyberconflits et Cybercriminalité" for the Hermès-Lavoisier publisher, is very cautious with respect to many findings that seem proved for many people. "The attack was not targeted, he says, it has affected India, Indonesia, Russia, U.S. and China! It state origin is not proven: a 10 engineers workforce during 10 month is within the reach of an enterprise or of a group of students." In its report on Stuxnet, Symantec, d of other countries ...

> A risk for the French nuclear plants?

The risk on our nuclear plants has been taken seriously by the French authorities. The IRSN (Institut de Radioprotection et de Sûreté Nucléaire) published on September 30 a research note on Stuxnet.[*]

It says that only the EPR nuclear reactor under construction at Flamanville uses a Siemens control system. Its possible sensitivity to malware such as Stuxnet must therefore be taken into account in the safety analysis. The propagation of the Stuxnet worm requires supervisory computers under Windows Operating System and using the Siemens PCS 7/WinCC line of products.

The IRSN goes on to explain the need for a comprehensive safety review, with a systematic and detailed technical analysis of systems whose dysfunctions can affect the safety of nuclear facilities. For the EPR, explains the IRSN, EDF chose the Siemens «SPPA-T2000" product, based on the "S5" technology, older and radically different from the "WinCC / S7" targeted by Stuxnet. Supervisory computers in Flamanville EPR don't use the Windows Operating System and don't use the WinCC PCS software; the Stuxnet worm has thus no influence on them. And IRSN continues by saying that safety analysis of the Siemens SPPA-T2000 platform has verified that this platform presents properties that guarantee among others, its immunity to malware, and in particular to the Stuxnet worm. The protection system of the EPR, the most important of the safety systems, is developed from another technology called Teleperm XS. This Areva technology doesn't use the pieces of software targeted by Stuxnet and its

[*] :www.irsn.fr/FR/Actualites_presse/Communiques_et_dossiers_de_presse/Pages/20100930-Ver_informatique_Stuxnet_peut_il_menacer_centrales_nucleaires_françaises.aspx

safety PLCs have no interfaces that would allow malicious software to infect them.

This is reassuringℇ or very worrisome since nothing guaranties that another malware couldn't be targeted to attack French sites. Very strict safety studies must continue to be performed.

> Deepen the principles of risk analysis and security paradigms

Above all, Stuxnet teaches us that it is necessary to deepen our principles of risk analysis.

Indeed, we are accustomed to question ourselves about the origin of the threats we face and to discard several of them to keep systems simple. Now, if we favor the scenario where Stuxnet is targeting Iranian centrifuges, we must also recognize that to achieve its contamination goal, it has spread everywhere and can still cause damages to equipment in other industries. The code used by Siemens is probably also found in numerous other equipments, as it is often practiced. Therefore, the fact that no direct enemy has been identified doesn't mean that there is no exposure to highly sophisticated attacks ... The concepts of collateral loss or damage are well known in military operations and may also exist in business or industries.

We must also accept the fact that country driven attacks, even if not fully proven yet, must now be regarded as plausible.

Many risk assessments focus on system availability. A company must produce, sell and then be paid. Substantial resources are applied to backups, fire-fighting and disaster recovery plans. In many large SME, risk analysis is often limited to this point. Information confidentiality is often taken into account because of trade issues or regulatory constraints, such as health data confidentiality or issues of national sovereignty. Again, important resources are often used: encryption, tracking logs, keys management infrastructure, etcℇ

The integrity checking of the programs and executable code doesn't give rise to a lot of concerns. Unquestionably, all companies recognize the need for anti-virusℇ although on the Macintosh and Unix / Linux this usage is still infrequent! Defense in depth is sometimes taken into account, with an anti-malware software on the workstation and a second on the enterprise gateway, or even a third on the mail server if it is hosted. It is also often customary, to ban access to some equipments by blocking ports, for instance by removing USB ports.

Then there are several code signature solutions: RSA, elliptic curves, etcℇ Those solutions require the establishment of a Public Key management Infrastructure PKI. Unfortunately we still sometimes find integrity checks based on a simple hash code. The code is send along with its MD5 or SHA1 digest: upon reception the system verifies that the code and the digest are consistent. Nothing precludes a potential attacker to modify the code and send a new digestℇ Let's be serious!

The Stuxnet case reveals a different scenario: certificate theft. The certification authority becomes useless and the PKI is destroyed. According to the Symantec report, the first evidence of a Stuxnet like attack date back late 2008. A vulnerability allowing remote code execution in a shared printer spooler was exploited in April 2009. A preliminary version of Stuxnet was discovered in June 2009. On January 25, 2010, the Stuxnet driver is signed with an apparently valid certificate own by Realtek Semiconductor Corp. On July 17, 2010, ESET identifies Stuxnet again signed with a certificate from JMicron Technology Incℇ Verisign will wait until July 22 to revoque this certificate. Briefly the PKI provided by Verisign has remained permeable for many monthsℇ

But are these scenarios integrated today in our risk analysis? Should we not change paradigms? Find another way to ensure executable code integrity? The ban of any connection on a machine doesn't always address the operational requirements. We've seen systems with USB ports blocked with resin, but there is always a time when we must update software, and then...

Claiming that a certificate will never be stolen is not very serious. Should we not go further with other security measures and greater defense in depth? ■

Le bilan Stuxnet pour les éditeurs d'antivirus

Entretien avec Pierre-Marc Bureau, chercheur-analyste chez ESET,Michel Lanaspèze, expert chez Sophos, et David Grout, expert chez McAfee.

For our three experts, the media did not publicize the Stuxnet story very early. This may be partially explained by the complexity of this threat and its history. Details on its mechanisms and its targets have trickled in publications.

Stuxnet has become newsworthy because it targets SCADA infrastructures, explains Michel Lanaspèze. Most research labs deal with more than 60000 new malware sample each day, which leaves them little time to analyze the likely intent of malicious code: their foremost mission is to detect and block threats, before analyzing and explaining the potential consequences of an attack.

David Grout adds that the strongest new elements of this attack are especially related to the fact it was carefully targeted. It belongs to a specific type of attack known as APT (Advanced Persistant Threat) that have a unique goal and are dedicated to this goal. The worm, says David Grout, used a combination of factors that leads to think that its authors had access to large resources:

- digital signatures that let them bypass applicative controls,
- the use of a large number of unknown or unpublished vulnerabilities,
- an expert level knowledge of Siemens' PLC environment,
- the need for physical access to a system to initiate the attack.

This worm is very interesting because of its complexity, he adds:

- the use of 4 zero-day exploits (e.g. ms10-046 lnk/shortcut vulnerability, ms10-061 - print spooler vulnerability)
- digitally signed and valid drivers (e.g. mrxcls.sys)
- the first PLC (Programmable Logic Controller) rootkit,
- a Windows rootkit,

- advanced techniques to avoid detection by antivirus
- propagation techniques,
- updates and mutations (for example through connections to www.todaysfutbol.com or peer to peer).

It is extremely rare, adds Pierre-Marc Bruneau, to see a software worm exploit a previously unknown vulnerability. It is the first malware to target critical infrastructures. A malicious software usually tries to spread to the largest number of systems possible. Stuxnet for its part aimed at penetrating one or several highly secured networks. Using several stolen digital certificates to spread without raising suspicion is also new. This worm was completely unknown and was propagating using new infection vectors, it was thus very difficult to detect.

After the file was submitted to antivirus software vendors, says Pierre-Marc Bruneau, a trigger has been added and instances of Stuxnet are now detected like any other piece of malware. When it installs on a system, Stuxnet uses the same vector as other malwares, namely a set of Portable Executable (PE) files.

Michel Lanaspèze agrees with this analysis. «Antimalware are very efficient to detect and block known malware.They are also efficient, but less so, to block unknown malware.» As soon as Stuxnet was identified, most anti malware software vendors have promptly updated their solutions to block this new threat and prevent the infection from spreading.

«Practically all anti malware solutions use technique that go well beyond the classical signature to prevent infection by unknown malware.» says Michel Lanaspèze. For example, techniques of behavioral protection, HIPS, etc. These techniques are always being developed and their improvement will allow to minimize the impact of «zero day» attacks. We must however keep in mind that a 100% effective protection will probably remain an inaccessible Graal and that the response to such attacks must thus be seeked in complementary protection techniques (network access control, intrusion detection, vulnerability management, etc.) and the ability to react quickly and efficiently to new kinds of attack. For Pierre-Marc Bruneau, better collaboration between security vendors and user communities would certainly allow a potentially malicious file to be submitted for analysis to anti malware vendors as soon as it is identified. This collaboration would favor the identification of threats that would thus be detected faster.

Furthermore, several solutions can be considered to secure operating systems; lets not forget that without the unknown vulnerabilities it exploited, Stuxnet would not have escaped detection for so long.

But Pierre-Marc Bruneau also sets limits to what an antivirus can do: «Our antivirus must not be in charge of verifying digital signatures.» This task should be left to the operating system. In the case of Stuxnet, the largest breach was that code signing certificates were stolen from JMicron and Realtek and these companies did not signal the theft. This omission put thousands of users that trust their certificates in danger. For David Grout, preventing such attacks depends on the combination of whitelist application filtering, antivirus, and antirootkit but also physical access control.

«Today, there are two main kinds of attacks: worms and viruses for ... and a new generation of malware targetting particular assets to which Stuxnet, Zeus, and Aurora belong, explains David Grout.

Many companies view antivirus software as commodity tools, I believe they are wrong; The data of an enterprise have more than ever a high value: competition for patents, tecnological advantage, profit...»

«To conclude, says David Grout, it is necessary to properly assess the criticality of the target to be protected to provide the right levels and means of protection. Even an operating system that is relatively unknown or an application that is less under attack might attract a well versed public.»

«Finally, this affair shows that questioning the security of certificates is justified, says Michel Lanaspèze, since Stuxnet seems to have been digitally signed with certificates it was not authorized to used.»

«It seems clear that in the future, validating the integrity of a system will rely in part on a hardware component, says Pierre-Marc Bruneau. However, I am not qualified to envision how these mechanisms might be deployed. Defense in depth, privilege separation, critical system isolation, access control are well known solutions that provide an effective protection to IT systems.» ■

Validy: a Paradigm Switch to Ensure Code Integrity

During the Forum International de la Cybercriminalité, late march 2010, Mag Securs met with Validy. We already knew this company and had looked at their technology in 2005.

Our discussions in may and june have touched on the possibility of ensuring executable code integrity.

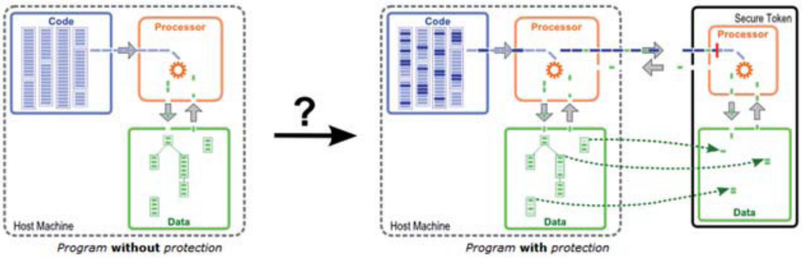
Validy Net Inc. was founded in 1998 by a French team in the state of Oregon, USA. The founders had met at X Pôle, Ecole Polytechnique's startup incubator in Palaiseau where they were working for Hyperparallel Technologies, a predecessor of the HPC Project and of present-day super computers. Gilles Sgro comes from the world of IT Systems management. Jean-Christophe Cuenod is a graduate from Ecole Normale Supérieure (1981), majoring in physics. Christophe Vedel holds a PhD in Computer Science and graduated from Ecole Polytechnique in 1989.

Validy Net Inc has invested a total of 9 millions dollars, 2 millions going to the protection of their intellectual property. Its French subsidiary, Validy SA, has applied for a dozen patents which represent including worldwide applications a portfolio of a hundred patents and patent applications. In 2010, Validy Net Inc. was a finalist of the American Security Challenge.

> Combining Hardware and Software Protection

Validy took an interest in the problem of protecting software code. To this end, they use a secure hardware component that executes a subset of an application's operations in place of the host main CPU. The host CPU can not work without obtaining the result of the operations performed by the secure component. This technology thus tackles the inherent risk of running executable code in a system. A Java bytecode recompiler has been developed by Christophe Vedel to split an executable in two parts: one for the main CPU and the other for the security coprocessor. Executable code signature techniques are being used today to reduce the risk of a system being compromised to an aceptable level.However, Jean-Christophe Cuenod told us last may (before the outbreak of Stuxnet in the media and just after our meeting at FIC 2010) that ne should not consider this residual risk as only potential: it has already happened! For embedded systems, under the potential of their user (or of an attacker, unbeknownst to the user), signature systems have loong been subverted by diverse methods.

The Xbox uses signatures to make sure that only games authorized by Microsoft can be loaded and this protection has been cracked! More recently, Apple's iPhone is designed to accept only signed applications from the Appstore.



Program **without** *protection*

Program **with** *protection*

It is however possible to jailbreak one's iPhone to let it execute any application. In both cases, software and/or hardware improvement have allowed the manufacturer to temporarily retake the advantage for new versions of their product but not to regain control of subverted machines.

> Trust can not rely on an external unverified element

Furthermore, says Jean-Christophe Cuenod, when protection is centralized (editor's note this is the case for the certification authority of a PKI), cracking it opens up the whole system «Our solution applies to each system individually, which will deter attackers». The major remaining problem, according to Jean-Christophe Cuenod is that of trusting the verification process and ultimately the certification authority. Two attacks can be devised:

- bypass the verification altogether by corrupting the verification program or its public key database,
- change the code between the time it is verified and the time it is executed.

Many classical attack methods can be applied to these tasks. Depending on the situation, one could:

- verification program V is used to verify program P. A security vulnerability is discovered in program P which must be replaced by a fixed version, signed and transmitted over the network. The attack is the following: before P is replaced by the fixed version, the attacker uses the vulnerability in P to write an exploit and take control of the machine long enough to change V into Vbad by substituting its public key. From this point, any code signed by the attacker is considered legitimate.
- The attacker has physical access to the machine. With this access, he can boot another operating system that gives him direct access to the filesystem. Through this access, he can change V into Vbad or change P directly. The difficulty of this kind of attack depends on the nature of the hardware. On a PC the operation is trivial and performed routinely using a «live CD» to change a forgotten password. On a game console, a «modchip» worth a few dollars allows the same result. On machine such as smartphones, miniaturization can be a hurdle for some hackers but will not stop a determined attacker.

These attacks have already been used with success to disable antivirus programs and will no doubt be used with the same success to disable signature verification sys

tems. Take as an example a VPN implemented using dedicated appliances. The appliances establish a secure perimeter (walled garden), but do not consider the problem of authenticating code. If within the secure perimeter, a single party becomes an attacker, intentionally or not, the appliances become useless.

To take an actual example, if an employee inside the secure perimeter wants to watch a match from the soccer world cup and plugs a 3G key into his machine, the breach in the enclosure can lead to massive compromission.

To summarize, trust can not be transfered. If you need to trust a program, you can not rely on a mechanism outside this program to guarantee its integrity.

> Verification is part of the system: a self signature without a certification authority

Validy Technology is different from all the systems I know: verification is part of the program itself. The robustness of the solution relies only on:

- the quality of the hardware implementation
- the quality of the software implementation (number of hidden variables, entropy, coverage, quality of the transform performed by the recompiler),
- the availability to the attacker of a system from which to learn.

The unique value of our solution is that its robustness does not depend on hypotheses made about external programs or mechanisms. It is very simple yet extremely important concludes Jean-Christophe Cuenod.

The effects of the Stuxnet worm started to appear in the media in july then exploded in september after our conversations in may and june with Jean-Christophe Cuenod, Gilles Sgro, and Christophe Vedel. ■