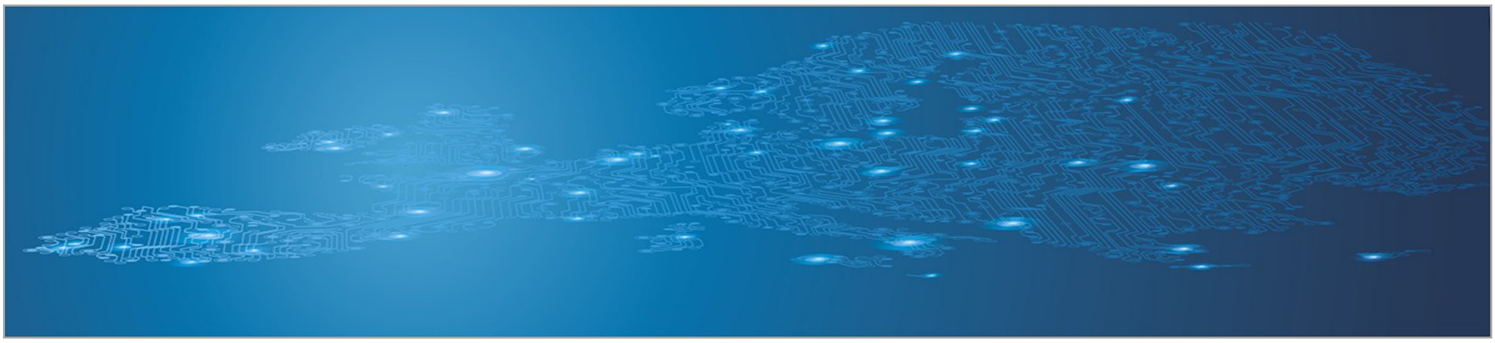

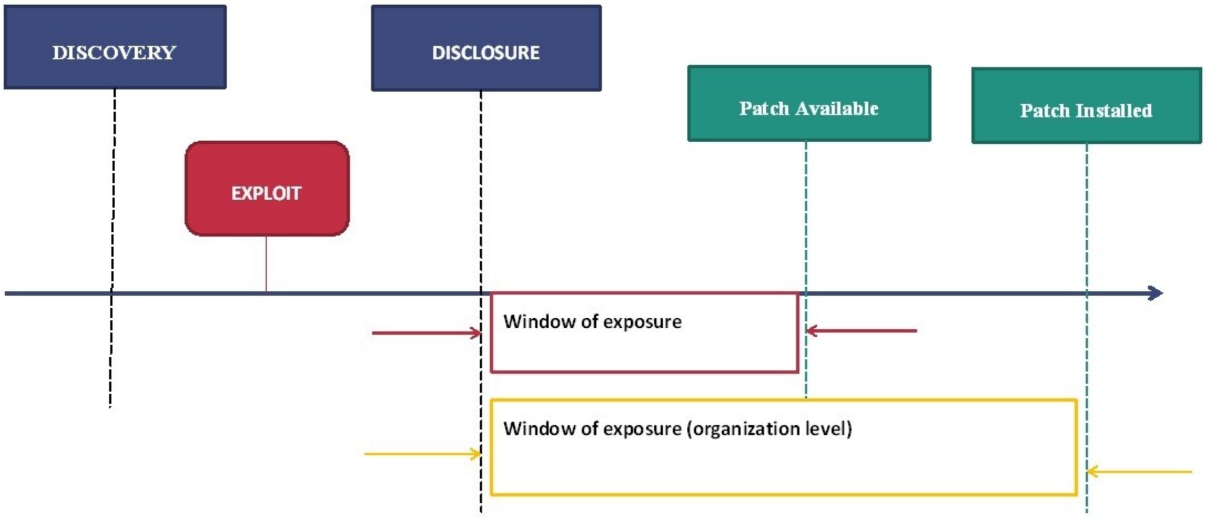

Window of exposure… a real problem for SCADA systems?

Recommendations for Europe on SCADA patching

December 2013

European Union Agency for Network and Information Security

www.enisa.europa.eu

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its member states, the private sector and Europe's citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists EU member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in EU member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Authors

- Adrian Pauna, ENISA
- Konstantinos Moulinos, ENISA

Contact

For contacting the authors please use resilience@enisa.europa.eu

For media enquires about this paper, please use press@enisa.europa.eu.

Acknowledgements

Contributors

Hans Pille (DNV KEMA)

Theo Borst (DNV KEMA)

Benessa Defend (ENCS)

Klaus Kursawe (ENCS)

Rob van Bekkum (ENCS)

Victor van der Stoep (ENCS)

Sebastian Ranft (Siemens)

Interviewees

Rick Ekkelboom (Alliander)

Egbert Battjes (ProRail)

Arnold Schuur (N.V. Nederlandse Gasunie)

Hylke Bakker (N.V. Nederlandse Gasunie)

Legal notice

Notice must be taken that this publication represents the views and interpretations of the authors and editors, unless stated otherwise. This publication should not be construed to be a legal action of ENISA or the ENISA bodies unless adopted pursuant to the Regulation (EU) No 526/2013. This publication does not necessarily represent state-of the-art and ENISA may update it from time to time.

Third-party sources are quoted as appropriate. ENISA is not responsible for the content of the external sources including external websites referenced in this publication.

This publication is intended for information purposes only. It must be accessible free of charge. Neither ENISA nor any person acting on its behalf is responsible for the use that might be made of the information contained in this publication.

Copyright Notice

ISBN 978-92-9204-070-3 doi:10.2824/25757

Table of Contents

1. Introduction

Much of Europe's critical infrastructure which resides in sectors such as energy, transportation,water supply is largely managed and controlled by SCADA (Supervisory Control and Data Acquisition) systems, a subgroup of Industrial Control Systems (ICS). In the last decade SCADA technology has passed through a transformation, from isolated and proprietary systems into open architectures and standard technologies that are highly interconnected with other corporate networks and the Internet.

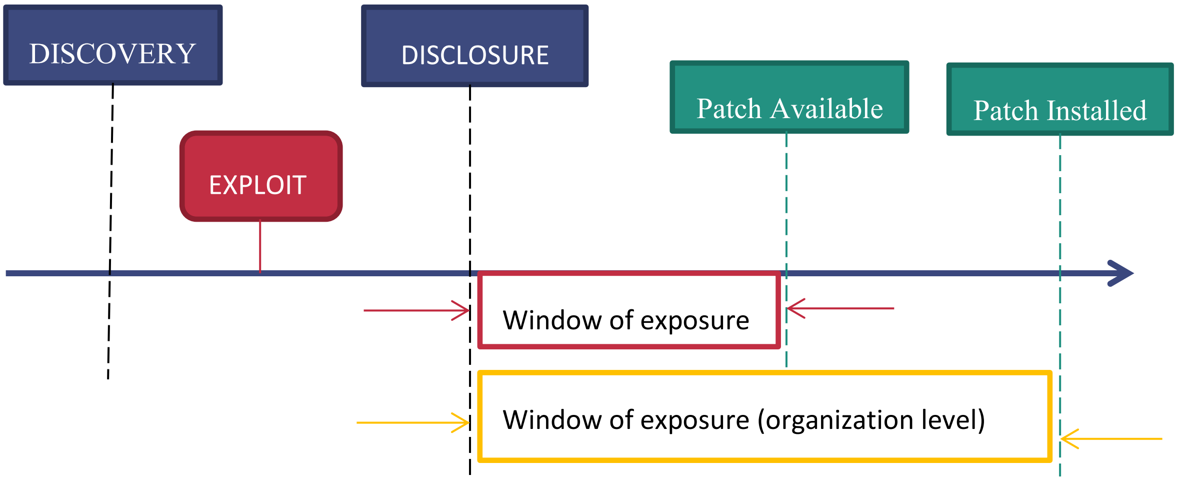

Figure 1. Window of exposure1

A consequence of this transformation is the increased vulnerability to outside attacks. One way to enhance the security of SCADA is through the application of patches.

Two of the key important issues with patching, at the moment are the failure rate of patches[2] and the lack of patches[3] for SCADA suystems.

Application of patches could have a significant effect on the operational behaviour of SCADA systems. When a patch is not tested thoroughly it can introduce unknowns[4] into the system, which is not acceptable for an environment utilizing SCADA. SCADA systems are usually deployed to stay operable for a longer time than regular IT systems. During this time patches are required to correct security and functionality problems in software and firmware.

From a security perspective, patches are important because they mitigate software flaw vulnerabilities, applying patches reduces the opportunity for exploitation. Patches can also be used to add new features to or improve on existing features of software and firmware. However, from a safety point of view, patches and software updates can also be a risk, as they might inadvertently change the behavior of a component in a way that endangers the process stability.

[1] Honeywell (2012), *Industrial Control System Cyber Security*. Retrieved from: https://www.honeywellprocess.com/library/news-and-events/presentations/HUGAP-IndustrialCyberSecurity.pdf

[2] "In 2011, ICS-CERT saw a 60% failure rate in patches fixing the reported vulnerability in control system products."(Kevin Hemsley –ICS-CERT)

[3] Less than 50% of the 364 public vulnerabilities recorded at ICS-CERT had patches available at that time." (SCADA Security Scientific Symposium (S4) in January 2012, Sean McBride)

[4] An interview with Joe Weis , security expert for the industrial critical infrastructures, in which he mentions several cases of patches that created problems: http://news.cnet.com/8301-27080_3-20004505-245.html

Regarding the above stated facts , the issue of window of exposure to vulnerabilities, comes in to the big picture of SCADA security.

In fact , the big question is if Europe can afford having critical infrastructures that use unpatched SCADA systems and for how long?

As ENISA stated in its 2011 report[5], there is a big requirement on " the research in the area of Patching and updating equipment without disruption of service and tools", therefore this white paper tries to address the issues from the perspective of "patching or not patching" SCADA systems.

Technical background: Window of exposure to vulnerabilities for a SCADA system .

The ***Window of exposure*** *is considered to be the time between the moment a vulnerability is disclosed and the moment a patch is available.*

From the perspective of an organization the moment a window of exposure is closed, is considered to be the moment all the affected systems have been patched.

Example:

After the first identification of Stuxnet, in mid-June 2010, the experts presented the vulnerability exploited in the targeted PLC device.

Those versions of the controllers used in SCADA (Supervisory Control and Data Acquisition) systems allowed DLL (Dynamic-Link Library) files to be loaded into the devices without validation.

In 2012 the producer of the PLCs released a patch so to prevent that from happening.

2. Target Audience

This white paper is addressed to the related community of SCADA operators and security engineers and tries to provide a small set of good practices and recommendations for policy makers and technology specialists in the sensitive domain of critical infrastructure protection.

This content will also concern asset owners and utilities interested in how to approach SCADA patching in their organization.

3. State of the art in SCADA patching

A SCADA system is composed of three parts. The field units such as RTUs (Remote Terminal Unit), PLCs (Programmable logic controllers) and IEDs (Intelligent Electronic Devices) are logic elements on the remote side that perform local logical tasks, gather measurements and send commands to the physical systems of the process control.

The other two parts are a communication system that connects those elements to each other and a master station, which includes elements like the historian and the human-machine interface (HMI). For this document, we do not explicitly cover patches to other components such as the PLC in the field. Many of the practices and observations from this document apply in this area as well, though additional challenges occur, such as delivery of the patch over a low-bandwidth communication line or rollbacks on devices with extremely limited memory.

The scope of this paper will be the Master Station which in turn consists of: SCADA software. hardware and the Human Machine Interface (HMI).

Organizations differ in the way they approach SCADA patching. While some organizations try to apply patches as quickly as possible, others are much more hesitant to apply patches that potentially change the execution environment. In some cases there is no company policy on patching, so that the decision to patch or not is made by local engineers. The policy may also vary on the nature of the process – patches on the core of highly complex and critical systems may require more thorough testing than a patch on the Human Machine Interface equipments (HMI) of a non-critical one, and some systems have safety certifications that need to be

[5] https://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services/scada-industrial-control-systems/protecting-industrial-control-systems.-recommendations-for-europe-and-member-states

re-issued once the software is changed. Often (where possible), standards are used as a guideline to construct a patch policy.

3.1 The existing standards on patch management

There are multiple standards describing controls for patch management. In the US, the NERC (North American Reliability Council) has issued critical infrastructure protection (CIP) standards to address the cyber security issues of North America's critical infrastructure. This standard does not provide technical measures but provides high-level approaches. NERC CIP-007 R3[6] specifies the necessity for a security patch management program for tracking, evaluating, testing and installing applicable cyber security software patches for all assets within the Electronic Security Perimeter(s). NERC is mandatory for most utilities in Northern America.

The German federal energy association (Bundesverband der Energie- und Wasserwirtschaft e.V.) has published a whitepaper[7] to address the cyber security of control and telecommunication systems which defines basic security measures and requirements for IT-based control, automation and telecommunication systems. Chapter 2.1.1.3 describes requirements (extending ISO/IEC 27002) for patch management of critical infrastructures:

1. The SCADA system shall allow the patching of all system components during normal system operation.
2. Installation of a patch should be possible without interruption of normal system operations and with little impact on the system's availability.
3. Preferentially, the patches will be installed on passive redundant components first. After a switch-over process (change of the active component in the redundant system) and a subsequent test the patch will be installed on the remaining components.
4. The contractor shall support a patch management process for the entire system, this process shall manage the testing, installation and documentation of security patches and system updates. In general, it is recommended that the operating staff administering the systems installs the patches and updates.
5. Installation and de-installation of patches and updates shall be authorized by the system owner and must not be performed automatically.

Further information and instructions for implementing these requirements can be found in the document 'Ausführungshinweise zur Anwendung des BDEW Whitepaper'[8] (only available in German) published in 2012.

Other documents that describe patch management are the NIST SP 800-40[9], NIST SP 800-82[10], ISA-TR62443-2-3 (which is not officially published yet but working draft is available[11]), and the document "Recommended practice for patch management of control systems" by the Department of

[6] (NERC) North American Electric Reliability Corporation (2013), *CIP-007, Systems Security Management*

[7] (BDEW) Bundesverband der Energie- und Wasserwirtschaft (2008), *Requirements for Secure Control and Telecommunication Systems*

[8] (BDEW) Bundesverband der Energie- und Wasserwirtschaft (2012), *Anforderungen an sichere Steuerungs- und Telekommunikationssysteme, Ausführungshinweise zur Anwendung des BDEW Whitepaper*

[9] (NIST) National Institute of Standards and Technology (2005), *SP 800-40, Creating a Patch and Vulnerability Management Program*

[10] (NIST) National Institute of Standards and Technology (2011), *SP 800-82, Guide to Industrial Control Systems*

[11] (ISA) International Society of Automation (2013), *ISA-TR62443-2-3 WD (TR99.02.03) - Security for Industrial automation and control systems*. Retrieved from: http://isa99.isa.org/ISA99%20Wiki/WP-2-3.aspx

Homeland Security (DHS) [12]. The ISO 27000 series also contains information on patch management of information traditional IT patching, with 27002[13] and 27019[14] adding controls for industrial control systems. Some of these documents can be used to establish a patch management policy, which will be discussed in the next chapter.

3.2 Procedural aspects of SCADA patching

3.2.1 Patch management policy

A patch management policy is an important component of an overall security management program. Separate policies should be created for patching traditional information technology (IT) networks and industrial control systems (ICSs) because IT patching typically requires relatively frequent downtime to deploy critical patches and any sudden or unexpected downtime of ICSs can have serious operational consequences.

Patch management itself could prove a nightmare if managed manually without a policy in place. With a well-designed policy, patch management is much less work and the risk of making mistakes is greatly reduced. The goal of a patch management policy is to keep the security and functionality of systems regularly updated through defining processes and work methods. This ensures that systems and applications are up-to-date, known vulnerabilities are addressed and an organization is compliant with sector requirements, regulations and standards. With an effective policy in place, teams will know exactly what is expected and what they need to do.

The policy should be regularly updated, and should not only relate to the patch management of operating systems. Applications that are external from the operating system also require patching because they can also pose security risks.

There are several documents that provide guidance for those responsible for designing and implementing a patch management policy, for instance: NIST SP 800-40, NIST SP 800-82, ISA-TR62443-2-3 (working draft) and the document "Recommended practice for patch management of control systems" by the Department of Homeland Security (DHS).

Important elements of any patch management policy include[15]:

- A **configuration management program** in which is described how records of hardware and software of an organization are kept and how information is updated to this inventory.
- A **patch management plan** that includes a schedule for when patches will be applied across all systems, as well as deployment instructions, measures for progress and back-out plans in the event that a patch causes an exception or unexpected system failure.
- A **backup/archive plan** which includes backup requirements and plans describing the possibility to roll back to a previous version.
- A **detailed plan for patch testing** which documents on how patches are to be validated prior to development.

[12] (DHS) Department of Homeland Security (2008), *Recommended practice for patch management of control systems*

[13] (ISO) International Organization for Standardization (2005), *Code of practice for information security management*

[14] (ISO) International Organization for Standardization (2013), *Information security management guidelines based on ISO/IEC 27002 for process control systems specific to the energy utility industry*

[15] (DHS) Department of Homeland Security (2008), *Recommended practice for patch management of control systems*

- No matter how good an organization's systems or employees are, things can always go wrong. Therefore the policy should also include **procedures/actions for when a patch impedes system functionality** and cannot be successfully removed in a disaster recovery plan.
- An **incident response plan** that defines a scheduled discovery process to identify new vulnerabilities, patches and/or workarounds to mitigate these vulnerabilities. This plan should also include procedures to review discovered vulnerabilities and incidents/feedback on issues discovered in the patch process.
- **Documented unit patching operations** in which coordination is formalized between the operation teams responsible for operating the process. Scenarios and plans should be made to perform patches on production environment.
- A **policy on and a list of non-patchable devices**. Processors such as Intel 8088,286 and 386 are still used by many legacy control systems[16]. Even though they are adequate for the functions they have, simple security meseasures such as encryption cannot be applied.This is because this type of patches cannot be handled by them.

Patch management is generally included in various compliance regulations. Therefore teams should document their efforts to be compliant with certain regulations. This documentation can help pinpoint potential issues which allows for further refinement of the policy.

The patch management policy must list the times and limit of operations the patch management team is allowed to carry out. For example, patches that require a restart, or when it is unknown whether a restart is required, should be deployed during scheduled down time.

In addition to policies on the actual patch management, it is also a good practice to enforce policies on product development/procurement that reduce the difficulty of patching at a later stage. Of great relevance in this respect is a good documentation of the assumptions different components make about each other and the environment, so that the patch developers have guidance in developing secure patches for the whole lifetime of the system.

3.1.1 Patch management service contract

Vendors often offer their customers a service contract, in this contract an agreement is made in what way the vendor will be responsible for patch management and how the patch management will be executed. The most common contract is one in which the vendor is responsible for patching the Operating System and the SCADA application.

Other responsibilities/agreements that need to be discussed in the service contract are:

- The focus of the patches (SCADA application, Operating System, 3rd party applications).
- The distribution of the patches from supplier to the system.
- How the patch will be installed, at what level, and who will do this.
- At what level patches will be tested and by whom.
- Who will be responsible in the event of a failure.
- Until when the service contract is valid.

[16] http://energy.gov/sites/prod/files/Vulnerability%20Analysis%20of%20Energy%20Delivery%20Control%20Systems%202011.pdf

3.3Technical aspects of SCADA patching

3.3.1 Not installing patches

Some organizations deliberately choose not to install patches on (critical) SCADA system. There are multiple reasons for this:

- The application of patches could have a negative impact on the operational behavior of SCADA systems. The nature of SCADA systems means that they must be highly available and should be able to respond in a timely manner. Some extremely critical systems may have no allowed outage windows available, and can therefore not be patched, unless the patch management system allows for the patching to be done while the system remains in operation (e.g. by leveraging redundancies).
- The risk of applying a patch is considered too high compared to its benefits, i.e., the risk that something goes wrong due to the patch changing a component behavior is higher than the risk of the issue the patch is supposed to fix.
- The system has limitations that do not allow for patching, e.g., restricted CPU power or memory (in many cases, the patched system requires more memory than the original, and many devices already run on their limit when first shipped),
- The software on the SCADA system is not supported anymore by the vendor, or the vendor does not exist anymore. In this case either new SCADA systems should be deployed or organizations could develop their own patches but developing own patches for OS and SCADA applications is almost impossible, especially when the source of the OS or application is proprietary.
- Alternative controls, like operating system hardening and firewalls, have more priority in reducing the risk for a break-in.

These reasons also show the importance of early planning for patching. The ability to patch on the live system, to change communication algorithms on the fly, and sufficient resources to allow for future updates are all aspects that can be planned in during the product design phase, and therefore should be a requirement in the procurement process of critical systems.

3.1.1 SCADA patching process

An ideal patch patching process includes the steps as shown in Figure 2. In practice however the process could be quite different. Due to economic reasons vendors of SCADA systems do not develop a patch upon the discovery of a new functionality or security problem, because they have to test their patches extensively before they can release them to their customers.

In some cases the patch deployment is also done by the vendor, which can make the patching process more difficult and costly.

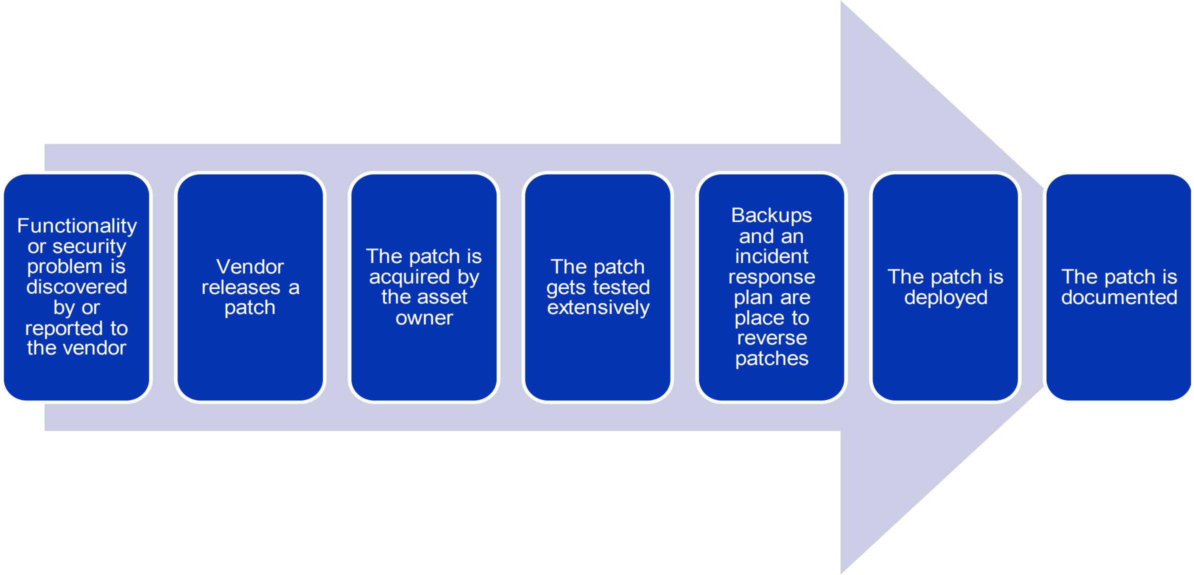
Functionality or security problem is discovered by or reported to the vendor
Vendor releases a patch
The patch is acquired by the asset owner
The patch gets tested extensively
Backups and an incident response plan are place to reverse patches
The patch is deployed
The patch is documented

Figure 2. The Ideal Patching Process

Even when testing and deployment is done by the asset owner there is still the matter of testing the patches, which can take some time in which production systems are vulnerable.

3.1.2 Testing patches

Patches should be adequately tested on an environment that closely mimics the production environment of the SCADA system on which the test will be applied. Creating such a test environment could be expensive, even when it is done virtually but it is not uncommon for patches to have an adverse effect on other software or systems. A test environment has the possibility to monitor the effect of patches without interrupting the real production process.

After a patch has shown to be working functionally it can then be deployed to production systems. Regression testing is advised[17].

Often patches are developed and thoroughly tested by the vendor, although this does not mean the patch will not break functionality in the production environment of the asset owner. Vendors test patches on a typical reference system which can differ from the environment of the asset owner because of additional software and systems being used. This can potentially affect the behaviour of the to be installed patch that was not identified by the vendor. Often the asset owner has developed a separate way to test the patches on their own environment, either through a DTAP model (Development, Testing, Acceptance and Production) in which patches are first installed on a development environment, and if they do not fail are moved to the next development phase or through installing the patches on redundant systems first before deploying them to other systems.

3.1.3 Priority based SCADA patching

Next to testing the SCADA patches there are other ways to reduce the risk of patches interfering with system availability. One way is to distribute patches on a priority basis. The priority is established by the criticality of the system being patched and the criticality of the patch[18]. An

[17] (NIST) National Institute of Standards and Technology (2011), *SP 800-82, Guide to Industrial Control Systems*
[18] Tofino Security (2013), *Making Patching Work for SCADA and ICS Security*. Retrieved from: http://www.tofinosecurity.com/blog/making-patching-work-scada-and-ics-security

inventory should be created in which all machines are prioritized and categorized into groups. These groups define when and how they are to be patched.

An example of a group is the 'Early Adaptors', who will receive a patch as soon as it is available. This group can then act as a quality assurance. Early Adaptors should not be on the production environment but typically these are lab or training systems. Another group is that of 'Business Critical' systems that are patched when the Early Adopters have been stable for a certain period of time. This time period should depend on the level of risk associated with a patch.

In the case of a widely distributed environment, patches could also be distributed depending on the geographical location of where the systems are located. Low and high priority systems could be grouped into different groups where the low priority systems are the first to be patched. This type of patch distribution allows for engineers to travel quickly between the systems in the case when errors occur.

4 Challlanges related to SCADA patching

The integration of SCADA technology into highly interconnected corporate networks and the Internet has been accompanied by the awareness that these systems needs to be secured better. One way to do this is through efficient patch management. Patch management has already been defined to a great extent in the IT domain but patching SCADA systems, partially due to the criticality and complexity of the systems and the processes, is quite different. In this chapter an overview is given of open issues related to SCADA patching and SCADA patch management.

A. Procedural challenges:

- **Appropriate boundaries for the service agreement** - should be defined between the vendor and the customer. This can be done in the patch management service contract or some other service agreement. Responsibilities, in case of failure, should be clearly defined. For instance, although a vendor will thoroughly test a patch on a laboratory environment it cannot be guaranteed that this patch will not break system functionality when placed on the production environment of a customer, especially when a vendor is not able to test a patch the system of a customer. However, even if a patch is tested on a system similar to the one utilized by the customer it is still difficult to identify the party responsible once the patch breaks the system.
- **Vulnerabilities are rated with the use of the classic IT scoring method** CVSS (Common Vulnerability Scoring System). These scorings may not be suitable for control systems like SCADA. ICS-CERT recommends that control systems owners and operators customize the CVSS score to their local environment[19], but this can lead to different parties using different scorings. Furthermore, some experts state that the ICS-CERT's vulnerability reporting is not addressing the underlying issue – "that the most serious vulnerabilities in control systems are deliberate design features, not bugs"[20], referring to design features like default credentials and the lack of encryption. Another issue with vulnerability reports sometimes include exploit code, or information on how to make an exploit. Combined with the lengthy patch deployment interval in SCADA systems this could lead to attackers having an advantage on an unpatched system.

[19] (NIST) National Institute of Standards and Technology (n.d.), *Common Vulnerability Scoring System Version 2 Calculator*. Retrieved from: http://nvd.nist.gov/cvss.cfm?calculator&adv&version=2

[20] K.J. Higgings (2013), *ICS-CERT, SCADA Patching Under The Microscope*.
Retrieved from: http://www.darkreading.com/vulnerability/ics-cert-scada-patching-under-the-micros/240150763

- **Patch confidentiality** - The most critical time period for any security vulnerability is the time between the vulnerability being known to potential attackers, and the time the patch to fix this vulnerability is deployed. If patches need to be extensively tested in the field, this requires that the patch itself is known to a comparatively large number of people long before it can be applied. This allows a potential attacker to reverse-engineer the vulnerability from the patch, and use it on the still unprotected systems. There are some techniques to resolve these issues on IT systems, but it is unclear if they can be directly translated to the SCADA setting. Compensating controls (for instance: application whitelisting and intrusion detection) can be utilized to reduce the risks during this window of vulnerability.
- **Vulnerability discovery** - Not all vulnerabilities, issues and patches are communicated through the same channel. An organization should therefore maintain relations with all the suppliers relevant to its systems. These relationships can vary, from weekly or monthly calls to just subscriptions to a vendor's security announcement list. Without a patch management service contract organizations are often not aware of new vulnerabilities and patches related to the system.

B. Technical challanges

- **Transferring and obtaining patches -** Another open issue on finding the most suitable way of distributing patches, which can be big packages with more than two gigabyte of data, within an organization or between vendor and customer. There are different ways to do this (media like DVD, internet portals with access control for customers, secure file transfer implementations etc.) but there is no standardised guideline. Preferably a dedicated patch management system is used for obtaining and applying software patches. There are however very few automated patch deployment solutions for SCADA systems. A patch management system could also introduce significant risks; they could be infected or used as a centralized attack vector to industrial systems. Adequate sandboxing of patch management systems is paramount[21].
- **Patch deployment intervals -** Ideally an organization would deploy patches as soon as they come available, however this is often not possible because of the complexity of the process in which SCADA systems are incorporated and because the systems often need to be operable at any given moment. Furthermore patches need to be tested thoroughly before they can be applied to production environment, which can take days or even weeks, during which a system is vulnerable. Ideally alternative controls should be used during the window of exposure for preventing a vulnerability to be exploited. For instance, when a webserver vulnerability has been discovered the organization could, if possible, block unwanted traffic to the webserver or disable the webserver all together. Some organizations rely on the vendor to patch their SCADA systems, which is agreed upon in the service contract. Decisions on patch deployment intervals are discussed between vendor and customer and are related to the size of an ICS solution, the criticality of installing new patches and the size of an organization. Experience from vendors shows that patching deployment intervals could range anywhere from every six months to a year. Deploying patches on a more regular interval is often too expensive for vendors because of the test processes and the complexity of deploying patches.
- **Legacy systems** - Another issue is that many ICS utilize older versions of operating systems that are no longer supported by the vendor. Consequently, available patches may not be

[21] E.D. Knapp (2011), *Industrial Network Security, Securing Critical Infrastructure Networks for Smart Grid, SCADA and Other Industral Control Systems*, Syngress

applicable[22]. This could be the case when the vendor has ceased to exist. In this case the systems should be replaced but this could turn out to be a costly undertaking. Often an organization will adapt the "if it isn't broken, don't fix it" mentality, especially when there is no patch management policy in place. These legacy systems often contain outdated SCADA system but also outdated Operating Systems which are equally as problematic. For instance, Microsoft will end support for Windows XP in 2014, which means that even though vulnerabilities and exploits might be discovered, no new patches are developed and distributed anymore. Furthermore, developing own patches for OS and SCADA applications is almost impossible, especially when the source of the OS or application is proprietary.

C. Legal challenges :

- **International business -** Most SCADA vendors serve a worldwide market. As such, they have to face legal issues arising from customs regulations of each country which might affect the ability to deliver software to the respective countries. In terms of SCADA patch management this can mean that a provider of SCADA patch management would not be allowed to deliver the required patches to a country of its customer, although this is not an issue within the European Union (EU). It can be a problem for any provider of SCADA patch management that does not reside in the EU.
- **Use of open source software (OSS) -** For vendors, there is also an important legal issue originating from the use of OSS in SCADA applications. OSS has to be cleared internally for use with the vendors own solution so that the vendor is able to use OSS without any legal claim. Another important open issue is the process of developing new patches when possible legal issues could arise in regards to OSS. It might happen that the vendor has to develop two critical patches OSS in a short time frame. Since the patches are critical they have to be quickly tested and released to customers. However, the patches have to be cleared first by the legal department before they can be distributed. This slows down development and distribution.
- **Vendor warranty -** Another issue is vendor warranty – an asset owner can lose its warranty when patching their system. Arrangements should be made with vendors to address this issue before deployment.
- **Asset management -** Asset management is an important part of patch management. Asset management is defined as the process whereby a large organization collects and maintains a comprehensive list of the items it owns such as hardware and software. This data is used in connection with the financial aspects of ownership such as calculating the total cost of ownership, depreciation, licensing, maintenance, and insurance. Asset management in the field of SCADA is much different from existing solutions for IT infrastructures. First of all there is no defined asset classification to be used by different vendors of SCADA solutions. Each vendor is inventing its own classification for the used assets. So the question arising from that is if it would be an improvement for the transparency towards the customers if there would be a standardized asset classification used by all vendors of SCADA solutions. Furthermore, there are not a lot of discovery tools (scanning for used hardware and software independent from the used platform) for complex and distributed systems used in SCADA solutions on the market. The usage of these tools is often prohibited due to possible technical issues that the scanning might cause. This leads to the open issue to what extent it is possible to automate the process of asset discovery and asset documentation.
- **Procurement and design for patch ability -** A number of patching issues would become substantially easier if systems are designed with updatability in mind. This does, however,

[22] (NIST) National Institute of Standards and Technology (2011), *SP 800-82, Guide to Industrial Control Systems*

require additional resources and thus makes the system more expensive during purchasing. Next to identifying, developing and integrating the measures into the system and device design it is therefore important that procurement departments learn how to add the proper requirements into tenders, and how to measure the level to which the requirements are met.

5 Good practices and recommendations

This section contains a list of good practices and recommendations related to SCADA patching.

A. Compensating controls :

Installing and distributing patches on a regular basis, is difficult for organizations and vendors because of the procedural and technical issues related to it. Patching should not be seen as a single method of defense, a good practice is to increase defense in depth (DiD) through the use of compensating controls. The term defence in depth is a term that refers to a strategy where multiple layers of defense are used to prevent attacks.

Important elements of a defense in depth strategy for SCADA systems are:

a. **Create awareness and understanding** in the organizations as to what failure of the SCADA systems could mean to the operations of the organization and what the policies and best practices are regarding patch management and security as a whole. A training program should be set up with job relevant information on how to apply security and how to respond to security threatening situations.

b. **Hardening the SCADA systems**, hardening the system means removing unnecessary features and locking down the functionality of the various components of the SCADA system. Microsoft Windows for instance contains a lot of different applications and services that are not necessary for operations, and they should be removed or disabled.

c. **Firewalls should be configured** in a way that only allows connections between trusted machines to trusted ports. Other ports should be closed when not in use. Firewalls should also implement an alarm-reporting mechanism to alert any time that abnormal behavior is detected. Intrusion Detection Systems (IDS) could also be used to detect this behavior. Where applicable, one-way communication diodes can provide an even higher level of protection. Deep Packet Inspection (DPI) Firewalls could also be used to check traffic for strangely formatted messages or unusual behaviours.

d. **Increase defense in depth through network segmentation**. Network segmentation is a complex measure but the basics are straightforward. Sets of equipment should be identified based on trust and similarity and placed into different zones. The next step is to identify what communications need to pass between the zones. At the points where zones communicate access controls like firewalls should be placed.

e. **Conducting regular risk and security assessments** to reduce potential security risks. Risks that cannot be eliminated should be reduced and the residual risk controlled.

f. **Application White Listing (AWL)** to compensate for malware code injection and execution by defining the allowed applications on a system, and restricting all other applications from running.

Between the time that a vulnerability is discovered and/or published other controls could also be used to temporarily mitigate the vulnerability. The settings or configuration of a system could be changed (temporarily) to block known attack vectors. These changes will not correct the underlying vulnerability but will reduce the risk of having these vulnerabilities exploited. Vendors of backbone telecommunications equipment often suggest configuration changes to their clients[23]. Microsoft also offers this service, included in most Security Bulletins is a section called “Workarounds” which describes how the system could be altered to reduce the risk of possible exploitation. Not a lot of SCADA vendors offer a likewise strategy, but there are numerous possibilities. For instance, if a vulnerability has been found in the webserver, and the webserver is not being used by any business critical operations, the webserver could be temporarily disabled or special rules could be deployed on the firewall or IDS to detect malicious behaviour.

B. Establishing a patch management program and service contract:

a. **Asset owners should establish a patch management program**. Without an established patch management policy, the process of applying patches could prove to be a very difficult undertaking. Even when the decision is made not to patch SCADA systems, due to more focus on compensating controls or because of the criticality of the systems, it should be formulated in a policy.

   b. **Asset owners should have a well-designed policy[24] in place so to reduce the effort of patch management and the risk of making mistakes**. Such a policy will define responsibilities and will help teams to understand what is expected and what they need to do. Important elements of a patch management program include[25]: asset / configuration management, patch management plan, backup/archive plan, plan for testing patches, an incident response plan and plans for documenting patches.

   c. **Asset owners should also establish a patch management service contract -** A patch management service contract helps to define on the responsibilities of both the vendor and the customer in the patch management process. An agreement should be made on the focus of the patches (SCADA application, OS, 3rd party application), patch deployment intervals, how the patches will be distributed, installed and tested, whose responsibility it is when something goes wrong and until when the service contract is valid.

C. Testing patches :

Testing patches before deployment is especially important for SCADA systems since unexpected behavior could potentially harm the operational functionality and could pose serious problems for asset owners.

   a. **Asset owners should always conduct their own tests.** This can be done virtually or by maintaining separate systems to test on. While patches for SCADA systems are often thoroughly tested by their vendors, there is still no guarantee that they will not disturb operations in a production environment. The environment an asset owner maintains can be very different from the environment that the vendor uses to test their patches and it is too costly to recreate a likewise environment.

[23] Tofino Security (2013), *Solving the SCADA/ICS Security Patch Problem*

[24] There are several documents that provide guidance for those responsible for designing and implementing a patch management policy, for instance: NIST SP 800-40, NIST SP 800-82, ISA-TR62443-2-3 (IEC/TR 62443-2-3) and the document “Recommended practice for patch management of control systems” by the Department of Homeland Security (DHS). In Europe the BDEW has published a whitepaper24 which describes requirements for patch management (extending ISO/IEC 27002).

[25] (DHS) Department of Homeland Security (2008), *Recommended practice for patch management of control systems*

b. **The test environment should closely simulate the operational environment**. A common way to do is to apply the DTAP street concept, which includes systems for Development, Test, Acceptance and Production.

c. **Redundant systems could be used to deploy the patch on**, which are then put in to production. During the evaluation whether the patch works as expected additional operational staff should be available to give their support to any potential issues caused by the patch. If the patch fails the organization can revert to the earlier system.

d. **Certified systems should be re-certified after a patch is applied.** This is an extremely important point. Technically speaking, any system that has been certified from a security perspective (e.g. a cryptographic device that is FIPS 140-2 certified) should be re-certified following a software patch. This is something that the industry tends to ignore because it costs time and money.

D. Distributing patches :

It is preferred to dedicate a patch management system for obtaining and applying software patches. The sandboxing of such a system is paramount, as a patch management system could introduce significant risks[26], attackers could potentially use it to distribute unwanted software.

The following best practices may be considered:

a. **Locate the patch management within an enclave that already has open Internet access**, such as the business network. If the patch management systems needs to be located in SCADA or DCS networks (e.g., if the business network is geographically separate), create a unique enclave for patch management with true air gap boundaries.

b. **The patch management system is responsible for downloading and testing patches**, configuration files, upgrades, and other third-party material; testing it for malware; and then archiving the validated files to read-only media (preventing any subsequent infection or manipulation).

c. **If required, implement two instances of the patch management system**: one to retrieve patches in isolation and one to distribute the validated patches after they have been hand carried across a true air gap.

d. **Evaluate patches and updates in a test environment in order to asses the risk of deployment**. “Early Adopter” machines could also be used to test patches before they are deployed to “Business Critical” machines.

e. Utilize digital signatures on patches or do hash verification where possible/feasible

E. Patch scheduling:

a. **Patch scheduling and deployment can be done after a patch has been tested thoroughly** and when the patch is approved for deployment.

b. **Depending on the chosen distribution method the approval of production managers is necessary** before deployment can be executed.

c. **Preferably the deployment is incorporated into regular maintenance schedules**, or when systems are less critical for operations, for instance during night time or during a period of low production.

[26] E.D. Knapp (2011), *Industrial Network Security, Securing Critical Infrastructure Networks for Smart Grid, SCADA and Other Industral Control Systems*, Syngress

6 Conclusions

Although patch management should not be seen as a silver bullet to resolve the security issues of SCADA systems it is nevertheless important that organizations establish a patch management policy. Without a well-designed patch management policy organizations might be vulnerable for attacks from the outside and the process of patching itself could prove to be a difficult undertaking. In the United States patch management is required for organizations that have to comply with the NERC CIP standard. While NERC CIP is mandatory for most utilities in Northern America there is no such universally mandatory standard in Europe. Therefore many different approaches are taken by asset owners within the European Union to handle patch management. The European Union or the Member States could increase the awareness of patches and patch management through enforcing that the issue of patch management should be taken into consideration when new requirements for devices are established.

TP-01-13-815-EN-N

ENISA
European Union Agency for Network and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office
1 Vass. Sofias & Meg. Alexandrou
Marousi 151 24, Athens, Greece

ISBN 978-92-9204-070-3
doi: 10.2824/25757

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu