# IEC 62443-2-3 "Patch Management"

**© *Rob Hulsebos***

Version 16/9/2018

## Introduction

Patching software is one of the most effective methods to be protected against all sorts of malware. Even though it will not protect against ***all*** malware, this should not be a reason for doing nothing. It is so important that the IEC-62443 devotes a whole document (2-3) to it.

All of us have experience with patching, either with your PC / laptop, tablet or mobile telephone. You probably have been bothered when your PC/laptop wants to install software at just the wrong moment, and/or the amount of time this takes. Now multiply this by, say, hundred (or more) for the effort needed to keep all equipment in an IACS-environment up-to-date, on a monthly basis, for all sorts of devices, from various vendors, who all have their own patch procedures and release moments… and you get an idea of the effort.

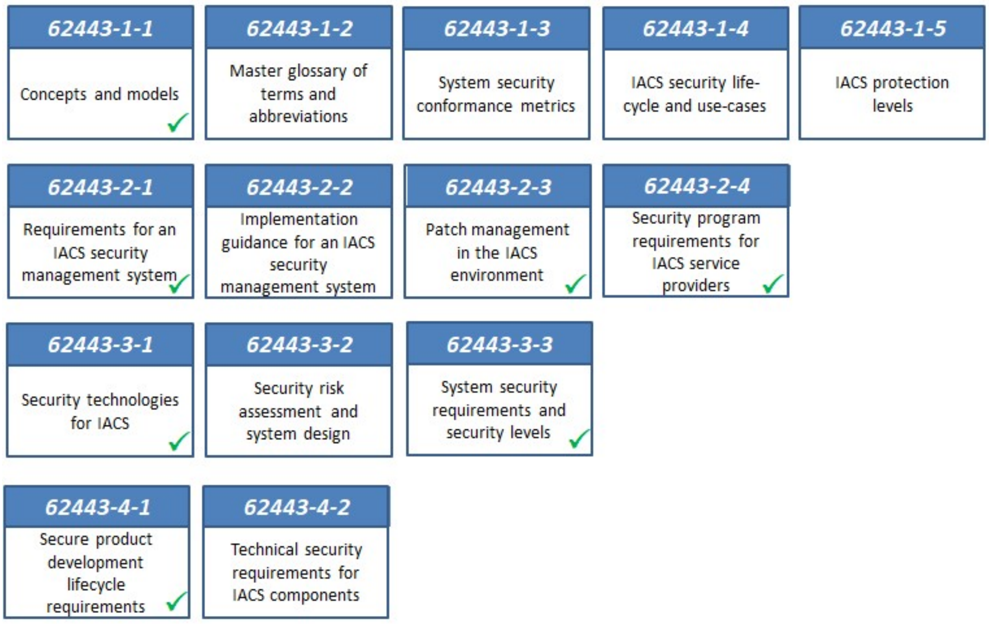
It is here where the IEC-62443 wants to help you establish a patch management process with associated procedures, causing patches to be installed on all appropriate devices, with no inadvertent consequences for the application, without forgetting a device.

In this article we'll look at what is described in the IEC-62443, and how patching is done in real-life (based on experiences of the author).

*The IEC document discusses both IACS-**owners** as IACS-**vendors** In this document, we will only discuss the processes and procedures for-IACS owners.*

## 1 What is IEC TR 62443-2-3 ?

The IEC-62443 "Security for industrial automation and control systems" consists of a group of document for asset owners, system integrators and vendors. An asset owner is responsible for the patch management of his equipment, which is described in the 62443-2-3, "Patch management in the IACS environment". Of course the help of IACS-vendors is needed, which is also described in this document.



| | | | | |
|---|---|---|---|---|
| 62443-1-1<br>Concepts and models ✓ | 62443-1-2<br>Master glossary of terms and abbreviations | 62443-1-3<br>System security conformance metrics | 62443-1-4<br>IACS security life-cycle and use-cases | 62443-1-5<br>IACS protection levels |
| 62443-2-1<br>Requirements for an IACS security management system ✓ | 62443-2-2<br>Implementation guidance for an IACS security management system | 62443-2-3<br>Patch management in the IACS environment ✓ | 62443-2-4<br>Security program requirements for IACS service providers ✓ | |
| 62443-3-1<br>Security technologies for IACS ✓ | 62443-3-2<br>Security risk assessment and system design | 62443-3-3<br>System security requirements and security levels ✓ | | |
| 62443-4-1<br>Secure product development lifecycle requirements ✓ | 62443-4-2<br>Technical security requirements for IACS components | | | |

The 2-3 is not a standard but a “TR”: a “Technical Report”. According to the IEC: “Technical Reports are entirely informative in nature and shall not contain matter implying that they are normative”. The reason is probably practical: a highly technical procedure like patch management cannot be written down in a standard. And if it could, it would probably be outdated within a few years.

The 2-3 contains (beginning in chapter 4) just a few pages stating the usefulness of patching, the patch lifecycle states, and the requirements for asset owners and IACS product suppliers. In chapter 7 and appendix A description is given of the data model to be used for exchanging patch information, a so-called “VPC” (Vendor/Patch Compatibility), in XML format.

In appendix B the IACS asset owner is given guidance on the patching process: information gathering, project planning and implementation, monitoring and evaluation, testing, and deployment. Appendix B describes the IACS product supplier’s tasks: discovery of vulnerabilities, development, verification, distribution and communication / outreach.

Because the appendices B and C are informative, they do not describe the procedures that an organisation that implements IEC-62443 **must** follow. However, a lot of experience is written down in these chapters, from (large) companies that have implemented and execute patch management processes. This can be used as a starting point for the implementation in your own company.

# 2 Vulnerabilities and patching

## Discovered vulnerabilities

Many researchers and hackers are looking into the firmware of all sorts of devices in the hope of finding a new vulnerability. When a vulnerability is found, the following can happen:

1 It may be misused, in malware, to attack equipment.

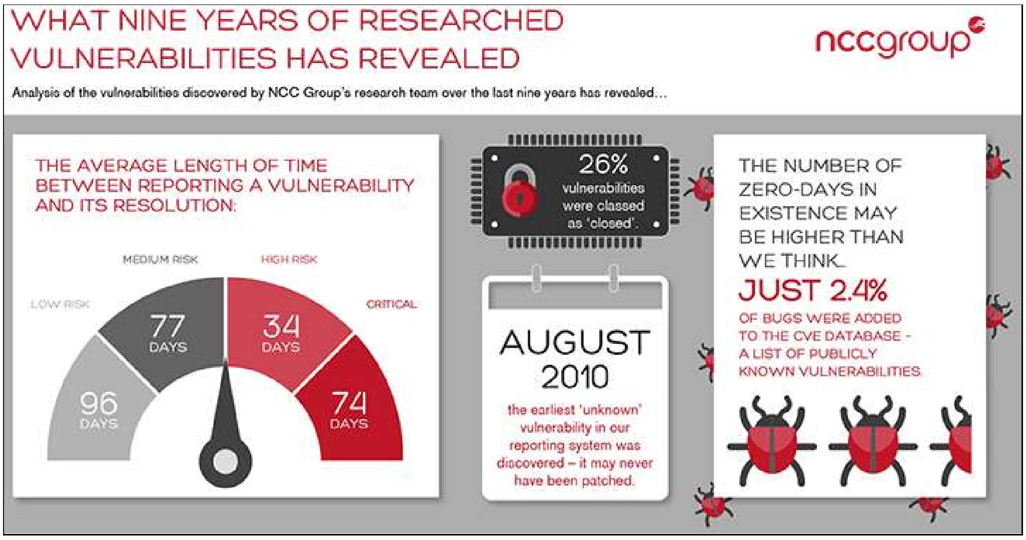
2 It can be sold to a company that buys these “zero days” for re-selling them to others.

3 It can be reported to the vendor of the device. The vendor can either use this information to create a patch, or ignore the information (do nothing). The time a vendor needs to fix a vulnerability is usually several months (see NCCGroup statistics below).

4 It can be reported to the CVE or ICS-CERT, who publishes it, and reports it to the vendor (see [3]). Statistics data from NCCGroup (see below) show that only 1 in 25 vulnerabilities is ever published in the CVE database.

5 It is disclosed in a publication or a talk at a security conference. Responsible researchers give the vendor of the device a few months to fix the vulnerability (see [3]), but if the vendor is slow or does nothing, the publication / talk proceeds anyway.

In [3], the vendor can fix the vulnerability in his software, and release a "security patch". A patch is just that (small) piece of software, usually one file, that replaces the same file that contains the software with the vulnerable part. This way it is not necessary to re-install a complete software package, which may take a *lot* of time and often requires all configuration data to be re-entered.



A patch can also be issued by a vendor to fix a functional problem in software (for example, software that doesn't know[1] about 29 days in February in leap years).

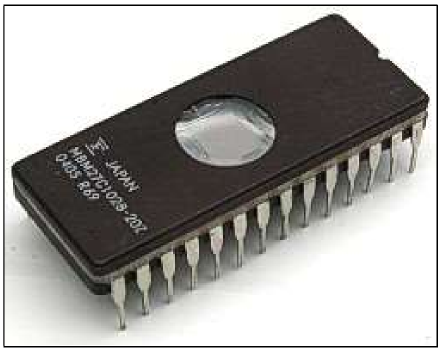
*In the remainder of this article, we will only discuss patches related to cybersecurity vulnerabilities. We will continue to use the word "patch", even though some vendors may call it an "update", "hotfix", "service pack", or "fix".*

# An era where patching was not possible

When I started in the software business, beginning of the 80's, there was no internet and not even Ethernet. Flash memory also didn't exist yet. Hard disks were 10 Mbyte in size, and 360 Kbyte or 720 Kbyte floppies were main transfer medium. Hacking didn't exist and updates were only for functional problems, and very limited in number.

For embedded devices, the only way to install new firmware was to manually remove the EEPROM with the firmware from its socket (on the PCB) and insert a new one. Devices had either to be sent back to the manufacturer, or a service engineer had to visit the customer to execute the exchange procedure.

[1] After 2018 years, you'd expect we know now what is special in a leap year. But for programmers, this seems to be difficult (see https://codeofmatt.com/2016/02/29/list-of-2016-leap-day-bugs/ for an overview).

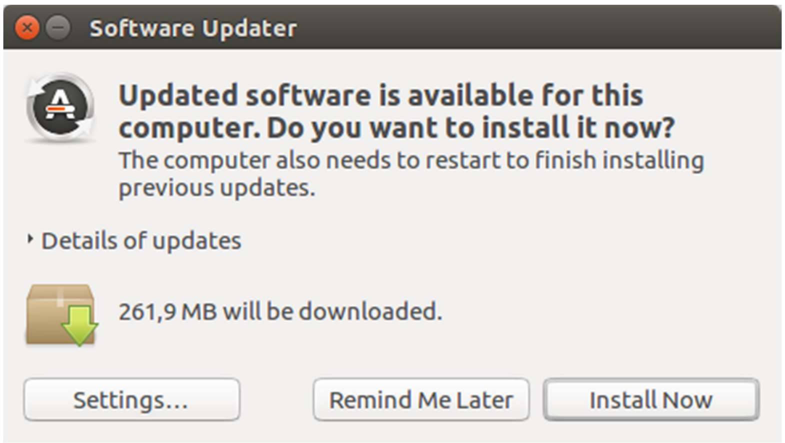


As a software engineer, writing software for such devices was often tedious and slow. If you modified the software, it had to be burned into an EEPROM, the EEPROM inserted in the PCB, the PCB inserted in the device and all cables connected, and could only then be tested. In case of a bug, the EEPROM had to be erased, while in the meantime you modified the software, compiled it, burned into the EEPROM, etc. etc. It was a slow procedure, so you took care to properly test the software to avoid having to update all customers *again.*

Nowadays, upgrading software and installing patches is much easier. I wonder whether this had the consequence that attention to testing of software has decreased, because “we promised to release this software tomorrow” followed by “we’ll just release a new patch next week”. It increases the burden on the customers of such devices, unfortunately.

# Automatic patching

Some vendors have added an automated patch installation feature to their software. For example, Microsoft does this Windows, Apple for IOS, Google in the Chrome browser, etc. These systems regularly ‘call home’ via their internet connection, determine if new patches are available, and if so, download them in the background. Then, at a certain moment, the patches need to be installed. This can be done after you have given permission, or automatically during the night. Usually a reboot is also necessary, and this, as you have probably experienced, always happens at the wrong moment.


Software Updater

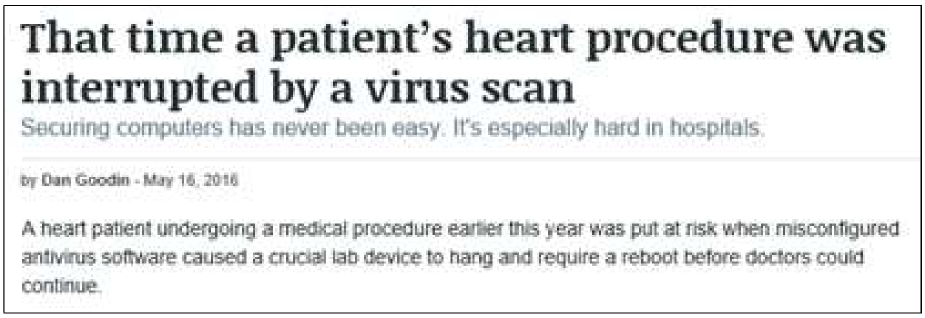
**Updated software is available for this computer. Do you want to install it now?**

The computer also needs to restart to finish installing previous updates.

‣ Details of updates

261,9 MB will be downloaded.

While the automatic patching is beneficial in a consumer environment, in an IACS environment it is **most unwanted**. The unpredictability about which patches are installed, and when, and the unscheduled automatic reboot may lead to production loss, or full disasters.

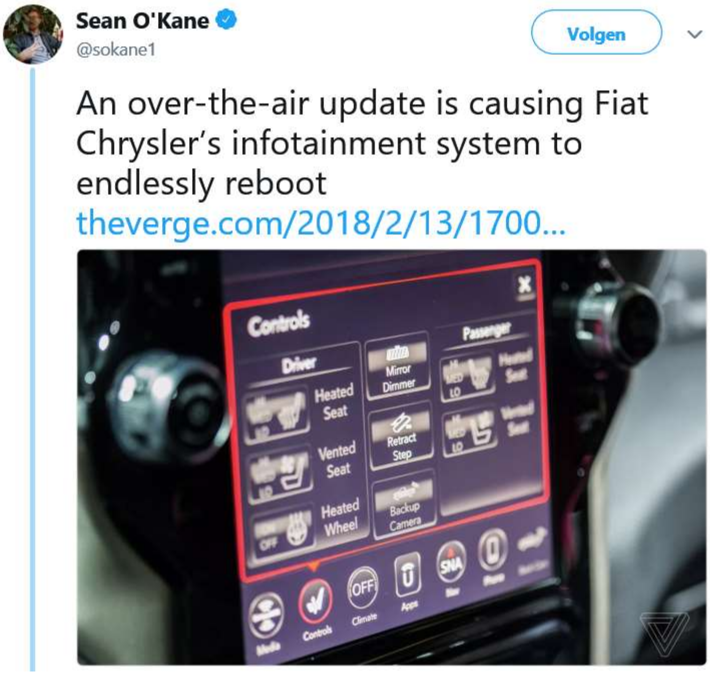

That time a patient's heart procedure was interrupted by a virus scan

Securing computers has never been easy. It's especially hard in hospitals.

by Dan Goodin - May 16, 2016

A heart patient undergoing a medical procedure earlier this year was put at risk when misconfigured antivirus software caused a crucial lab device to hang and require a reboot before doctors could continue.

Automatic patching may also cause havoc when the patch (or its installation) was not tested well. It may be rejected by the software of the device it was intended to be installed on, only to be seen as a "this patch hasn't been installed yet" and then the cycle repeats.


Sean O'Kane
@sokane1

An over-the-air update is causing Fiat Chrysler's infotainment system to endlessly reboot theverge.com/2018/2/13/1700...

Additionally, automatic patching can also work when there is an active Internet connection accessible for the IACS components. This is itself a vulnerability, which requires additional protective measures. IACS systems should be disconnected from Internet as much as possible. Thus, the advice is: no automatic updating!

## The problem with installing patches

The problem with patching is that in many cases it necessitates a reboot of the system. And while a system is doing this (stopping and restarting), it cannot run production. If you are a Window user, you've probably encountered this several times, when Windows is doing its monthly update. It may restart several times before all new software is installed. This often happens at moments when you can least use this (i.e. you turn on the PC in the morning to give a presentation, and then you'll have to wait for an hour before it is done).

The reason that Windows does this is that it cannot replace parts of its own software by a newer version if that software is actively running. So Windows installs the new software at an alternative location, then stops itself almost completely, and only then it has the possibility to replace its software by something new.

Microsoft knows this is very annoying, and has spent a lot of effort in redesigning Windows to require less reboots. But some parts still need a reboot, so you're still stuck.

Embedded devices (running their own OS) also require a reboot. This reason is that new firmware must be loaded into (flash) memory, which can only be done by a so-called "bootloader". The application firmware needs to stop, the bootloader becomes active, installs the new firmware, and then starts it. This is also a vulnerable moment for the device: if the device is switched off **before** the new firmware installation has completely finished, it cannot run – it has no valid application firmware installed in its memory anymore. That memory contains part new, part old firmware. This is the reason behind the warning that you see often "not to switch off the device".

Hopefully it can still start the bootloader if you power-cycle it; there *are* algorithms with which a bootloader can detect that a previous installation didn't complete (but the vendor must have implemented such an algorithm).

If this doesn't help: the device is "bricked" (in hacker's jargon) and probably useless. Hopefully you have a spare…

# 4 The work process

In this chapter we will discuss some of the steps described in annex B of the IEC 62443-2-3, but in more general terms and with more examples. Where possible, a reference to the applicable section of the annex is given. Note that the IEC 62443-2-3 does **not** enforce you to implement your patch management process as is described in the document, because annex B is only "informative".

The annex provides a lot of information for the IACS asset owner on how to set up an organisation with staffing and procedures to effectively implement the patch management. It defines the following major activities:

- Information gathering: asset inventory, supportability, relationships with vendors, assessment of the existing environment.
- Project planning and implementation: business case, roles and responsibilities, deployment infrastructure, back/restore infrastructure.
- Procedures and policies for patch management: monitoring for patches, evaluation, testing, deployment and change management.
- Operating a patch management system: executing the patch management process, awareness, outage scheduling, inventory maintenance, key performance indicators, auditing, and verification.

# Setting up an organisation

The IEC 62443-2-3 specifically mentions that patch management requires a business case, that can be represented to (senior) management, in order to secure staff, funding, and other resources.

Because patch management may cost a lot of time, may require extra equipment, and sometimes will necessitate a production stop, procedures must be in place to coordinate this with the production staff / facility management staff, etc.

Additionally, patching can often only take place outside normal office-hours. This may require extra funding for paying the overtime hours.

# *What* to patch?

Patching is the process of installing new software / firmware on a device. This is usually called: an "update". This can be anything: from a few kilobytes for small embedded devices, to many megabytes for large, complex, intelligent devices, to hundreds of megabytes for operating systems like Windows or Linux.

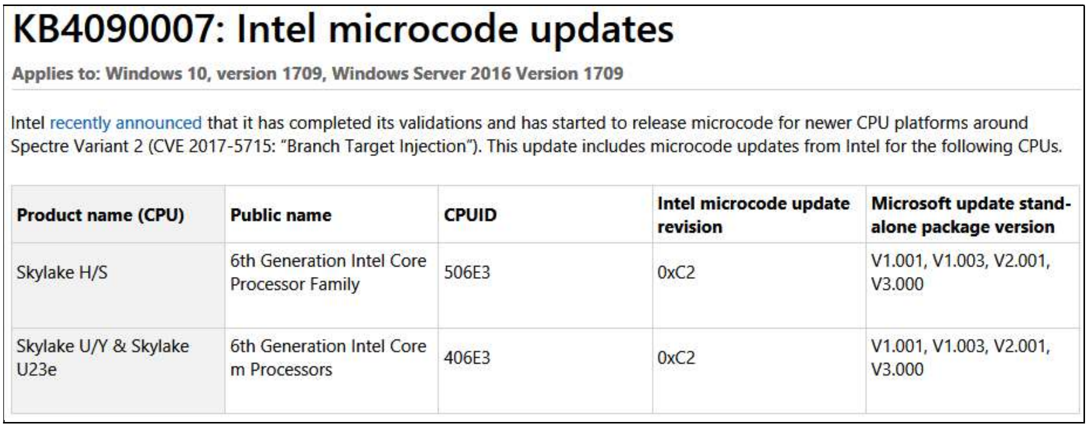
Be aware that a device can have multiple software / firmware installed. For example, for a PC you'd think that it is only Microsoft Windows that is running on it. But the PC manufacturer (not Microsoft) is responsible for the BIOS, and perhaps a few device drivers (for the hardware components), and Intel / AMD has firmware for more chips than just the CPU. A CD/DVD can have its own firmware,  just as all USB-devices, and don't forget the monitor.

Perhaps one should look suspicious at **anything** that has a power-cord (or a battery): reasonable chance that there is something in it that can (and needs to be!) updated. So, consider:

- PC's, laptops, tablets, smartphones
- Printers, scanners, NAS's
- Monitor, beamer, video-conferencing system
- Databases, (web/file/print/email)servers, backup systems
- Network: switches, routers, modems, converters, firewalls, gateways, access points, proxies …
- Building automation: smart thermostats, access control, IP camera's, …
- Industrial automation: PLC's, DCS's, historians, field devices, remote I/O, …
- Bootloaders, BIOS , PCB management, …
- Virtual machines, hypervisors, …
- Smart instruments
- Etc.

There is more equipment with firmware than you first think of!

Additionally, microcode may also need to be updated. Microcode is the software that runs *inside* a CPU (for example an Intel or AMD CPU). Microcode can (of course) also contain flaws, and may require an update. Usually one doesn't hear much about this; a notable exception are the Spectre/Ghost bugs that became known in the beginning of 2018.



KB4090007: Intel microcode updates

Applies to: Windows 10, version 1709, Windows Server 2016 Version 1709

Intel recently announced that it has completed its validations and has started to release microcode for newer CPU platforms around Spectre Variant 2 (CVE 2017-5715: "Branch Target Injection"). This update includes microcode updates from Intel for the following CPUs.

| Product name (CPU) | Public name | CPUID | Intel microcode update revision | Microsoft update stand-alone package version |
|---|---|---|---|---|
| Skylake H/S | 6th Generation Intel Core Processor Family | 506E3 | 0xC2 | V1.001, V1.003, V2.001, V3.000 |
| Skylake U/Y & Skylake U23e | 6th Generation Intel Core m Processors | 406E3 | 0xC2 | V1.001, V1.003, V2.001, V3.000 |

Finally, devices can have so-called "FPGA" (Field Programmable Gate Array), chips that can form logic circuits based on program code. A vendor can reprogram such devices to form other logic circuits (saving on changing the PCB or having to add/remove PCB connection, without having to solder on the PCB).

***Section B.3.1***
***Inventory of existing equipment***

Manually keeping inventory is a lot of work in larger installations, but may be workable in smaller systems. There exist tools to assist you, software packages listen to network traffic (completely passive) to determine the vendor, product type and software version installed. There are also tools to actively scan the network; devices found are queried to return identifying information. Both methods have their advantages and disadvantages (which is outside the scope of this document).

***Section B.3.2***
***Tools for manual and automatic scanning***

# Is there a new patch?

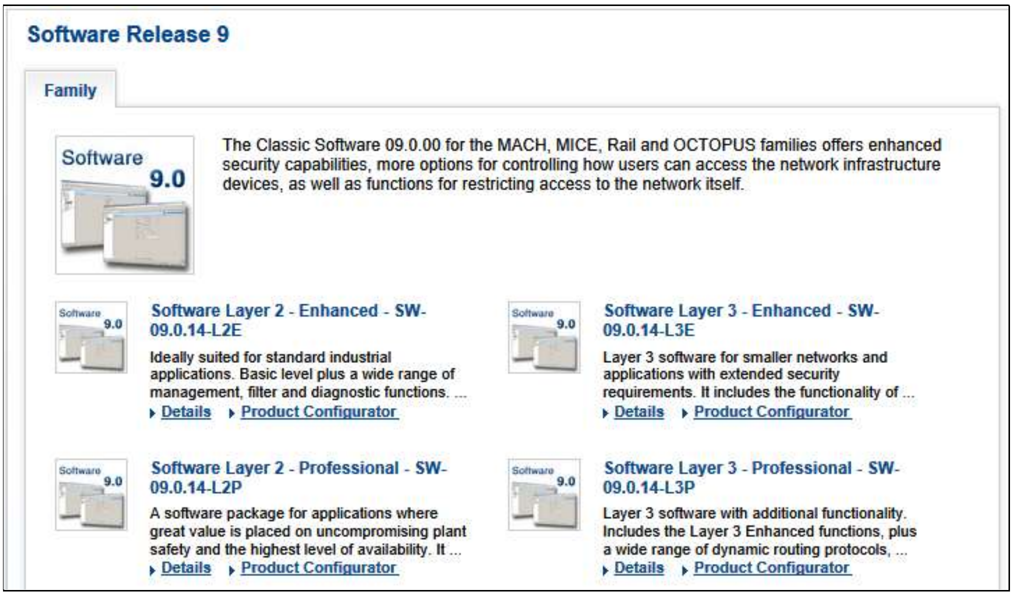
As customer, you'll have to keep track of the current patch level for all your equipment. Nowadays, many vendors have a dedicated cybersecurity webpage and/or email list. The easiest way to remain informed is to subscribe to the email list, and you'll be automatically informed when a patch becomes available.

More and more vendors have a monthly release cycle, meaning that all new vulnerabilities will be published on a predetermined day of the month. However, an "out of band" patch may be published in case of dangerous vulnerabilities that should be patched as quickly as possible.

***Section B.3.3***
***IACS product supplier contact and relationship building***

Some vendors just mention the existence of a new patch on their website, but do not inform customers. In such cases, you should regularly visit the the webpage to check for changes. It is best to do this at a regular moment, i.e. weekly, bi-weekly or monthly.

***Section B.3.4***
***Supportability and product supplier product lifecycle***



Software Release 9

The Classic Software 09.0.00 for the MACH, MICE, Rail and OCTOPUS families offers enhanced security capabilities, more options for controlling how users can access the network infrastructure devices, as well as functions for restricting access to the network itself.

Software Layer 2 - Enhanced - SW-09.0.14-L2E

Ideally suited for standard industrial applications. Basic level plus a wide range of management, filter and diagnostic functions. ...

▸ Details ▸ Product Configurator

Software Layer 3 - Enhanced - SW-09.0.14-L3E

Layer 3 software for smaller networks and applications with extended security requirements. It includes the functionality of ...

▸ Details ▸ Product Configurator

Software Layer 2 - Professional - SW-09.0.14-L2P

A software package for applications where great value is placed on uncompromising plant safety and the highest level of availability. It ...

▸ Details ▸ Product Configurator

Software Layer 3 - Professional - SW-09.0.14-L3P

Layer 3 software with additional functionality. Includes the Layer 3 Enhanced functions, plus a wide range of dynamic routing protocols, ...

▸ Details ▸ Product Configurator

# When there are *no* patches

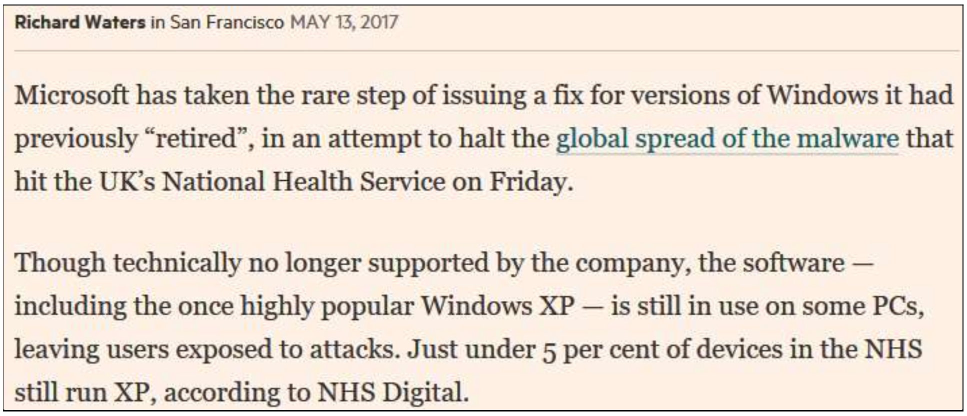
Patching software is extra work for vendors, usually the customer does not have to pay for it. But many vendors stop releasing patches for products they deem ‘old’. But: how old is old? Mobile phones two years. But in industrial automation, systems may run for 10 or 20 years. Support usually has stopped a long time before that (with some exceptions).

Many vendors announce at a certain moment that a product is no longer sold (end-of-sales), and usually it remains supported for a few more years. During this time patches for security issues can be released. Then, the end-of-support period ends, and any newly discovered vulnerabilities will not be patched anymore. How long the support period lasts, may differ per vendor.

IEC 62443

***Section B.3.4***
***Supportability and product supplier product lifecycle***

*But to everything there is an exception – even after Microsoft formally stopped Windows/XP support, it still released a few patches to fix a few serious bugs that couldn’t be ignored. But you cannot count on this to happen, it is at the vendor’s discretion.*

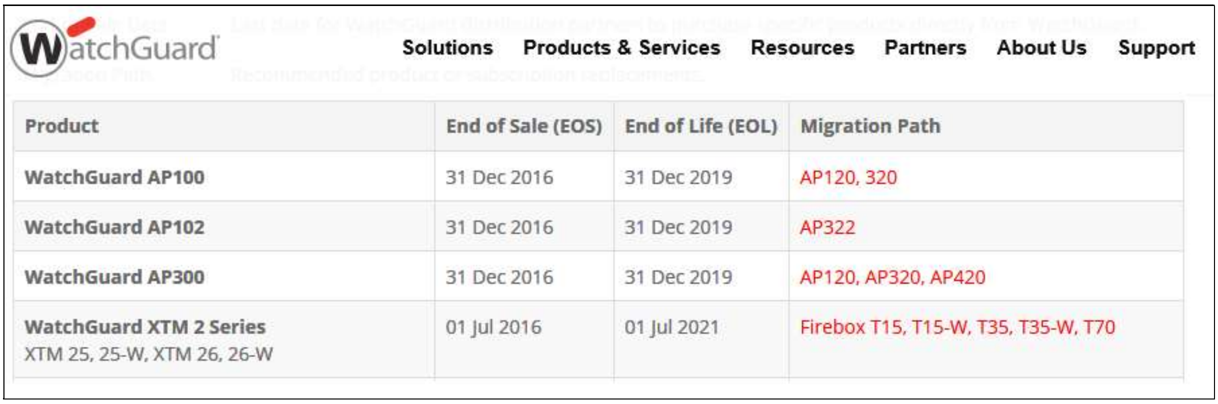


**Richard Waters** in San Francisco MAY 13, 2017

Microsoft has taken the rare step of issuing a fix for versions of Windows it had previously “retired”, in an attempt to halt the global spread of the malware that hit the UK’s National Health Service on Friday.

Though technically no longer supported by the company, the software — including the once highly popular Windows XP — is still in use on some PCs, leaving users exposed to attacks. Just under 5 per cent of devices in the NHS still run XP, according to NHS Digital.

The decision of a vendor to stop supporting a product is sometimes a technical choice, because it is no longer possible to generate (edit, compile, test, deliver) a new firmware – simply because the equipment that is needed for this is itself too old! It could also be that the electronics components are no longer available, so the product must be redesigned (and then also given a modern look & feel). But it can also be a commercial decision – hopefully you buy new hardware!



| Product | End of Sale (EOS) | End of Life (EOL) | Migration Path |
|---|---|---|---|
| **WatchGuard AP100** | 31 Dec 2016 | 31 Dec 2019 | AP120, 320 |
| **WatchGuard AP102** | 31 Dec 2016 | 31 Dec 2019 | AP322 |
| **WatchGuard AP300** | 31 Dec 2016 | 31 Dec 2019 | AP120, AP320, AP420 |
| **WatchGuard XTM 2 Series**<br>XTM 25, 25-W, XTM 26, 26-W | 01 Jul 2016 | 01 Jul 2021 | Firebox T15, T15-W, T35, T35-W, T70 |

In case a vendor stops the support of this product, so a vulnerability cannot be patched, and the product is not replaced by a newer version, then mitigating measures must be taken, for example (but not limited to):

- Extra firewall rules
- “Virtual Patching”
- Air gapping

***Section B.6.8***
***Risk mitigation***

# When you’re not *allowed* to patch

Some vendors do not allow their customers to patch software, for example process control software vendors do not want you to install new Windows updates. The reason is that the vendor cannot guarantee the correct operation of their products anymore.

*Why not? In most cases a patch does not have any side-effects, but sometimes they are. For example, when a software function erroneously relied upon an unknown side-effect. When the application unknowingly relies on this side-effect too, it will not work anymore after the patch is installed. The vendor must now also modify his own software (and perhaps issue an update…).*

The vendor first tests the patch, and only when it has been determined that his application software continues to work as intended, may the customer install the patch.

***Section B.5.3***
***Determining patch applicability***

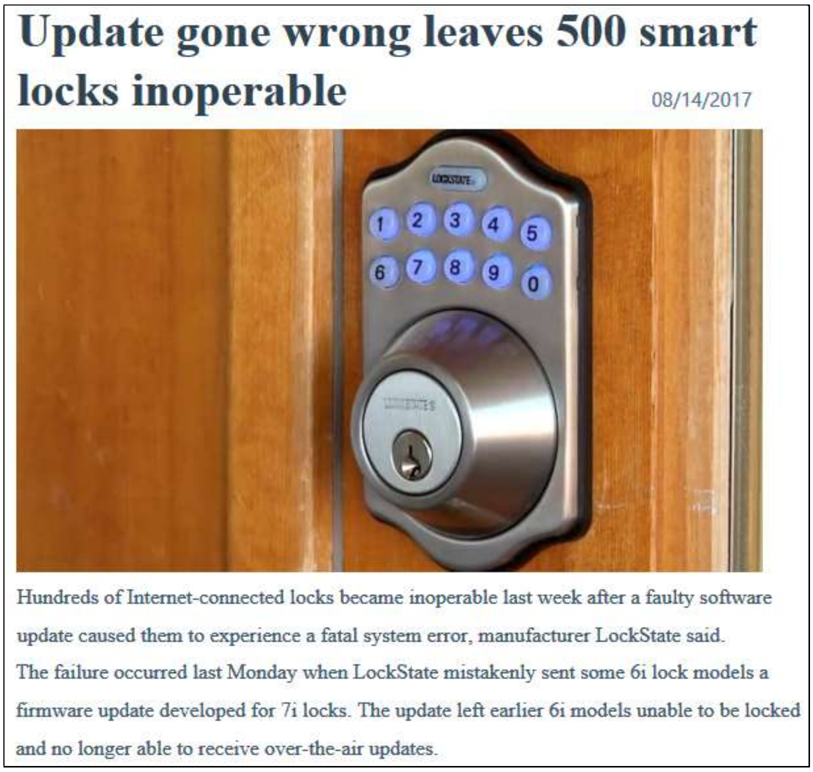
Of course this takes some time (even months), an in the meantime the customer(s) remain vulnerable and should implement compensating measures.

# Selecting the right patch

Windows has many patches each month, for the base operating system and for all other Microsoft products (i.e., Office). It is not necessary to patch software that you don’t have. If you are going to install such a patch anyway, it will be a waste of time because it will refuse to install (after a while).

Patches for embedded devices are more dangerous: usually they must match **exactly** with the hardware version they are meant for. Installing a wrong patch may (at worst) brick your device (= making it completely useless).

Some vendors have different sets of software for the same hardware, depending on licenses that determine which functionality is unlocked. Selecting a patch for a software for which you have no license(s) might result in a non-functional device.



**Update gone wrong leaves 500 smart locks inoperable**

08/14/2017

Hundreds of Internet-connected locks became inoperable last week after a faulty software update caused them to experience a fatal system error, manufacturer LockState said.

The failure occurred last Monday when LockState mistakenly sent some 6i lock models a firmware update developed for 7i locks. The update left earlier 6i models unable to be locked and no longer able to receive over-the-air updates.

## Does the patch cost anything?

Usually patches are free. Remember it is a fault of the vendor, that is now fixed, why should you pay for it?

## Is the patch valid?

After having decided that a patch must be done, the next step is to acquire the necessary file(s). Nowadays, this means: download it via internet, usually from the website of the vendor of the device, or via a reseller.

Beware of other website that also seem to offer the file(s) for the patch! The risk is: are you *sure* that the file(s) is/are **identical** to the originals? Are you sure nothing has been added, for example malware?

And are you sure that no transmission errors (however unlikely) happened while you downloaded the patch? Or that the USB-drive / harddisk the file was stored on did not have any data corruption errors? Data corruption in a patch file may be undetectable, and lead to very strange run-time problems.

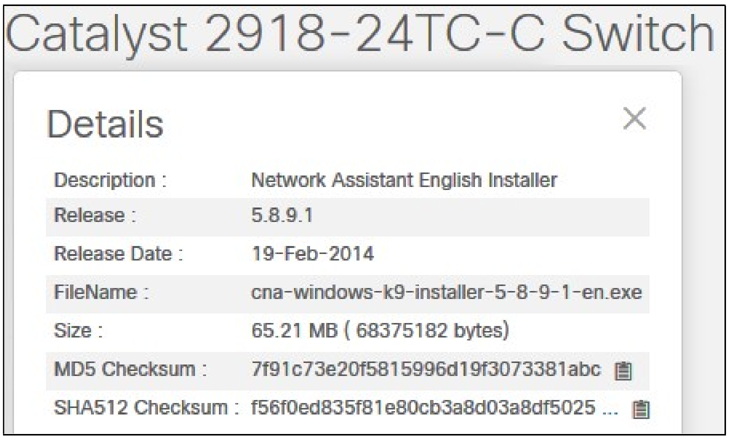
So the advice is thus to check the validity of a patch fileset immediately after it has been downloaded. But, without further information, you cannot verify the correctness of a patch file. The vendor must assist here, this can be done in different ways:

- The installation software on the device checks the patch before installing it. Usually this is done via a "certificate", a digital signature that has been added to the patch. Only the vendor knows the secret key with which the certificate can be signed. When the patch is installed, the device checks that it has the expected signature. If not, installation of the patch is refused.

- The vendor publishes the size of the file with the patch. A very simple procedure, which gives not much certainty about the file that you have. Better is:

- The vendor calculates a "hash" (sometimes called: "checksum"), and mentions this on his website. After you have downloaded the files, you should also calculate the hash, and check that both values are the same. If not, the downloaded files are not the same as the vendor has originally made.

  Various algorithms for calculating hashes exist, some well-knowns are "MD5", "SHA-1, "SHA-2" and "SHA-256". The outcome of a hash algorithm is always a large (hexadecimal) number, which is unique to the contents of a file. If only a *single* bit of the file changes, the calculated hash will be completely different.



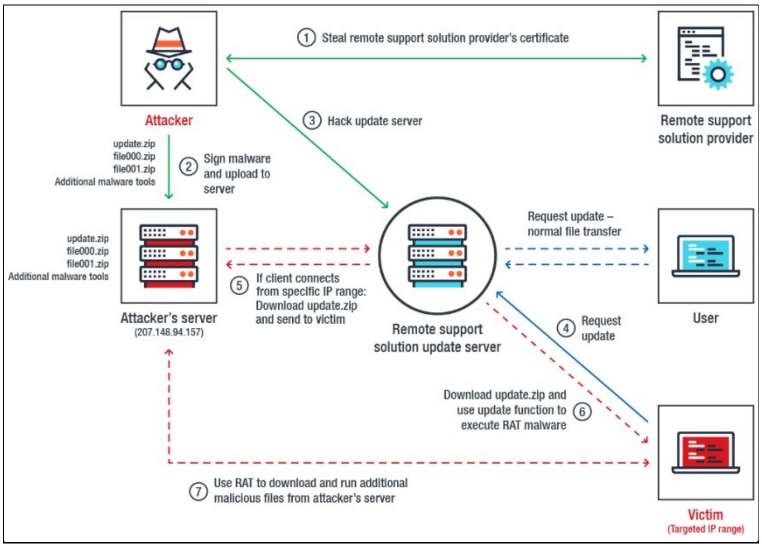
Catalyst 2918-24TC-C Switch

Details

| | |
|---|---|
| Description : | Network Assistant English Installer |
| Release : | 5.8.9.1 |
| Release Date : | 19-Feb-2014 |
| FileName : | cna-windows-k9-installer-5-8-9-1-en.exe |
| Size : | 65.21 MB ( 68375182 bytes) |
| MD5 Checksum : | 7f91c73e20f5815996d19f3073381abc |
| SHA512 Checksum : | f56f0ed835f81e80cb3a8d03a8df5025 ... |

*Example of how Cisco publishes a size, and two hash-values, for a file.*

It should be noted that even with the measures that a vendor takes, there is still no 100% guarantee that the patch that you download is the original one. After all, if a hacker could modify the software, why did he not change the hash/checksum/etc. values on the vendor's website to match the new patch software as well?


1 Steal remote support solution provider's certificate
Attacker
Remote support solution provider
3 Hack update server
update.zip
file000.zip
file001.zip
Additional malware tools
2 Sign malware and upload to server
update.zip
file000.zip
file001.zip
Additional malware tools
Request update – normal file transfer
5 If client connects from specific IP range: Download update.zip and send to victim
Attacker's server
(207.148.94.157)
Remote support solution update server
4 Request update
User
Download update.zip and use update function to execute RAT malware 6
7 Use RAT to download and run additional malicious files from attacker's server
Victim
(Targeted IP range)

*Infected update servers can install malware. Source: Trend Micro.*

Additional measures are necessary on the vendor's part, where the vendor must create a fingerprint of his software within a secure perimeter in his company, and then store this fingerprint in a secure repository kept. Anyone who want to install a patch must verify its fingerprint against the one kept in the secure repository; if they match then the user can be assured he has an unmodified patch.

IEC 62443

***Section B.6.3***
***Determining patch file authenticity***

# Make a full backup

Before a patch is installed, it is advised to make a full backup of the device. This eliminates any uncertainty about the age of the last full backup, and backups have their use in equipment failure/disaster recovery as well (which falls outside the scope of this document).

Not all devices can have their software backed up (usually embedded devices). With such equipment the 'backup' can be the latest firmware installed on it and (if available) the set of patches installed on it, and a document describing all the configuration settings. When you buy such a device, it is recommended to immediately download the latest firmware from the vendor's website, even if the device has this firmware already installed. You may need the files later, and it is not guaranteed that internet and/or the vendor's website are available when you urgently need them!

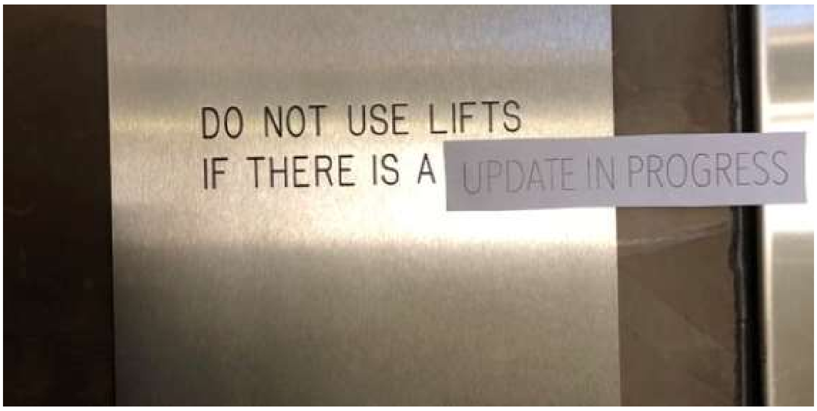
IEC 62443

***Section B.4.5***
***Backup and restoration infrastructure***

# The right moment for installing

Installing a patch almost always means: a device is no longer available for production purposes. This means that in an IACS environment the moment at which a patch can be installed is often only when there is a production stop (perhaps in a weekend, perhaps only once per year). And then: all devices that have patches waiting for them must be processed, so you'll be very busy.

This means that the investigative work discussed above must have been done earlier. During a production stop, there is often little time available. Make sure that enough time is left for testing purposes.



I have experienced that in IACS installations with redundant networks and redundant devices another procedure is followed: for each of the devices to be patched, the one that is in "standby" mode is taken out, and then patched. Later, when all "standby" devices have been patched, the system is switched over; the standby devices are now active and the others now in standby, so they can be patched now.

Although it works, you should remember that the redundancy is there for a reason, and that was not: patching. For the time needed to install all patches, there is no redundancy. Although the likelihood that something happens just in this period is probably small, the consequences may be large.

# Equipment needed

On a PC, no extra equipment is needed to install a patch – only some disk-space. But for embedded devices, installing a patch often requires extra equipment. Some examples:

- A USB-stick (thumbdrive) in the right filesystem format (FAT, FAT32, NTFS, etc.) with the files needed to install the patch. When the USB-stick is inserted in a USB-port of the device, and the device is then rebooted (power-cycled, or by pressing a reset-button) it recognized the files on the USB-stick, and will start installing the patch.
  Note that older devices are often not capable of recognizing the most modern USB-sticks (for example because they are too large). Retain 'old' USB-sticks for future use!

- A laptop with a serial cable, and a special software install program for the device to be updated. The other end of the cable must be plugged into the device to be updated, which usually requires a reboot as well.
  Note that the software for older devices usually needs older Windows versions which may be no longer in use (or allowed) in an IT-environment. A virtual machine with the older Windows version may sometimes help, but not always. So if the IT-department wants to replace that old laptop by a more modern version, it might be worthwhile to keep it, despite that fancy new Windows version and despite the IT-department wishing to destroy it.

- An SD or PCMCIA card. The latter is the professional form of the "SD" card that you use in your mobile phone or digital camera. The advantage of these cards is that you can prepare the firmware installation in advance (i.e. on a PC with a PCMCIA-card reader), and installing the new firmware is nothing more than exchanging the existing card with the new card. You then automatically have the 'old' card as the backup. A disadvantage of this method is that PCMCIA cards are not so cheap as SD-cards are.

*Cisco warns that a memory card should be large enough to contain a new software version.*

# Notification of parties

Before a device is going to be patched, which probably results in some downtime, the production staff must be informed. Or, vice-versa: the production staff lets you know when there is time to install the patch(es). The larger the organisation, the more important it is to have a good coordination procedure, as production always has priority and installation time is sometimes limited to weekends or holiday periods, sometimes only once per year.

# Executing the installation procedure

The installation procedure is normally to be done as documented by the vendor of the device. The exact procedure may be different per vendor and even for devices of the same vendor.

# Clean-up of the device

After the patch has been verified to work, remove all files copied (added) to the device. Probably it doesn't do any harm to leave them, but when the next (or next.. next) patch is installed, the continuous adding of files may cause a disk to fill up, and the other colleague is puzzled why *his* patch doesn't install.

Also, take care to remove any USB-stick that has been used. If it remains plugged in, the device might try to reinstall the same patch over and again every time it reboots.

When it was necessary to remove the device from its standard location, now place it back and insert all cables again. This is a critical step because often cables are put back in the wrong place!

# Testing the correct installation

After a patch has been installed, it is usually necessary to power-cycle / reboot / restart the device and/or the software. When it starts again, it is a (albeit simple) indication that the device still works. But it does not mean that the patch has installed correctly, perhaps something went wrong and the original software / firmware is still present.

Usually it not much work to determine the actual software/firmware version that is running on a device, this should match with the documentation of the patch.

The next verification step is usually the restart of the application software of the system / machine / installation.

# Administration

After a patch's installation has been verified, update your administration.

# Does it solve the vulnerability?

A patch is made to solve a vulnerability in software. But the patch is itself software, so can have mistakes too. In 2011, ICS-CERT reported that there was a 60% failure rate in fixing vulnerabilities. Although there are no figures available for later years, a lot of vendors have taken up the process of patching their software in a more professional way, so probably the failure rate is less.

Nevertheless, in many cases it is almost impossible for an IACS asset owner to test whether a vulnerability has really been solved, and one simply relies on the vendor.

# Reverting a patch

Sometimes a patch does not work as intended (i.e. due to unexpected side-effects), and one would like to revert to the old firmware. This isn't always possible! The reason is usually that the vendor has implemented a new mechanism for installing firmware, which is not capable of recognizing / handling / going back to old(er) firmware versions.

In case you *are* able to revert, the next step is: do you *possess* the old firmware files? If you've installed that old firmware earlier, then the answer is likely to be: yes. But if you have never installed firmware before (= the device still has its factory settings), you first have to get the old firmware. Some vendors put all firmware versions ever released on their website, while others don't, or remove the oldest versions after a while.

And even if they do, they must be accessible via internet at the time you urgently need them. My advice is to do this as soon as you first install a device: look up what firmware version is in the device, download the files from the vendor's website, and archive it for future use.

# A better way of patching

Of course, it would be best of no patching is necessary at all. But writing flawless software is not possible. And even if it were, advances in cryptography (finding flaws in cryptographic algorithms), advances in cpu powers, flaws in hardware (i.e. Spectre), etc. might still necessitate installation of new software.

*This article is likely not complete in describing all possible ingress policing features. If you have any suggestions, comments or additions, please do not hesitate to contact me (email: rh[at]enodenetworks.com).*