CYLUS

Patch Management

in Rail Technology Environments

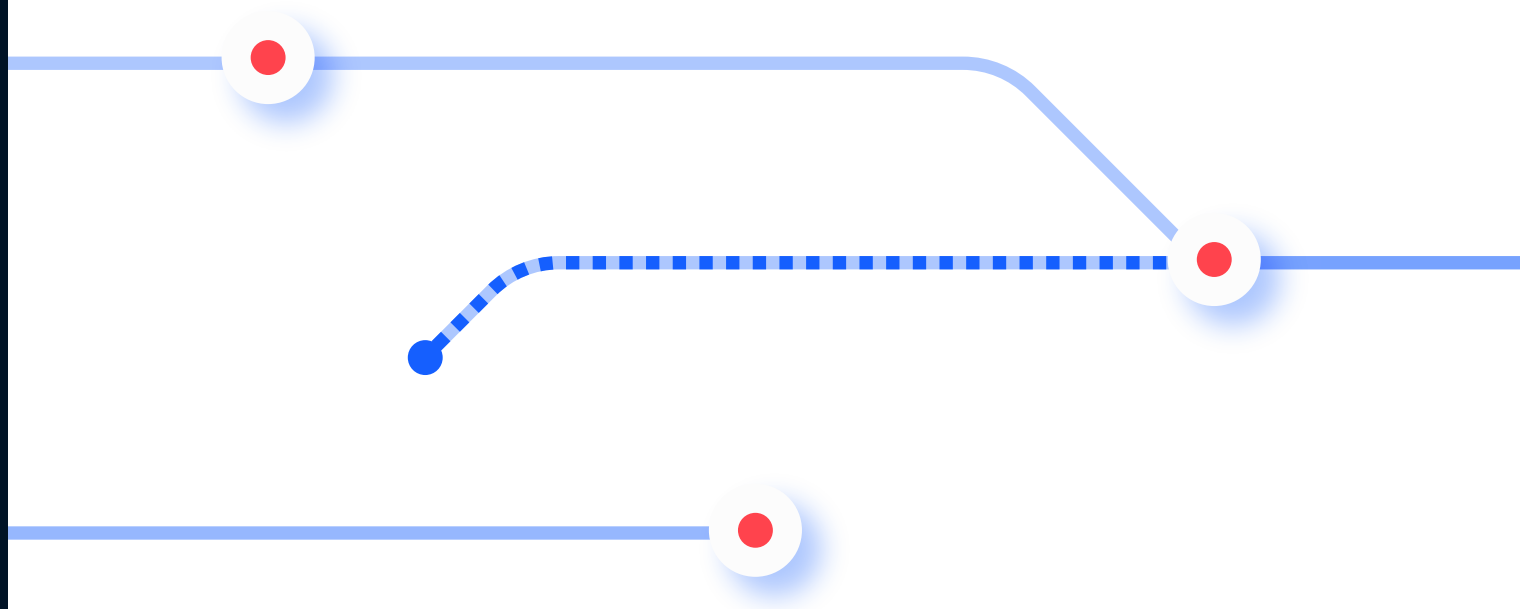
Introduction

In late 2022, the Transportation Security Administration (TSA) issued its third Security Directive (numbered 1580/82-2022-01) to improve cybersecurity and defend critical infrastructures. In pursuit of greater cyber resiliency, the directive emphasizes the importance of patch management in Rail Technology, aligning with the European Union Commission's Cyber Resiliency Act.

With rail systems having a lifespan of up to 30 years, many systems in use today may have never received security patches. However, modern rail standards and best practices, such as CENELEC TS 50701 and the upcoming IEC 63452, highlight the importance of patch management in building a strong cybersecurity strategy.

It's essential to note that there is no one-size-fits-all approach to patch management in the Rail Technology environment. Each rail operator must assess its risk tolerance and regulatory requirements to determine the best method for their specific needs.

This paper will provide an overview of the key considerations for implementing an effective patch management strategy.



Why is Patching Needed?

Patching is a matter that is very different from the typical mentality of a rail operator. Historically, when rail systems were deployed, you could (potentially) not patch them at all, and they would still operate on day ten thousand just as they did on day one. Likewise, the safety analysis would typically remain the same on day ten thousand, and all the safety risks would be mitigated during the system's deployment. Patches would only apply when functional bugs were identified, and operators would roll them out carefully and selectively. However, in recent years, two significant trends have made security patching a crucial aspect of Rail Technology.

The connection between safety and security has become widely recognized in the rail industry

First, the connection between safety and security has become widely recognized in the rail industry, leading to the understanding that a system cannot be considered safe if it is not secure. This means that security must be managed throughout a system's lifespan, including addressing vulnerabilities and applying security patches.

The second trend is the increasing use of Commercial Off-The-Shelf (COTS) components in the rail industry. Due to practical and economic reasons, Rail Technology suppliers have started relying on elements from the public domain, leading to a rise in the number of vulnerabilities affecting these systems. In the past, the railway being a tight community caused fewer vulnerabilities to be published. With the increased use of COTS, vulnerabilities published daily could affect the most critical rail systems and should be considered during patch management processes.

Now that we've gone through the what and the why, we will introduce the how. We will go through the steps of implementing an effective patch management strategy.

Keep Up to Date on Vulnerabilities

The first step is staying informed. Keeping up with new vulnerabilities is crucial when building your patch management program. Knowing potential security risks as soon as they appear allows you to proactively evaluate and respond to them and protect your systems against attacks.

If you have existing systems without such contractual obligations, don't worry; you can stay ahead of vulnerabilities with the right tools.

For new rail systems, it's best to include a requirement for vulnerability management from your suppliers. This guarantees you'll receive notifications about new vulnerabilities affecting your equipment and demonstrates your suppliers' commitment to analyzing and creating vendor-approved patches to address these vulnerabilities.

If you have existing systems without such contractual obligations, don't worry; you can stay ahead of vulnerabilities with the right tools. By relying on trusted sources such as the NVD (National Vulnerability Database) and ICS-CERT (by CISA), you'll have access to the latest information about potential security risks and be able to check if they apply to your systems.

Maintain an Inventory of Assets and Software

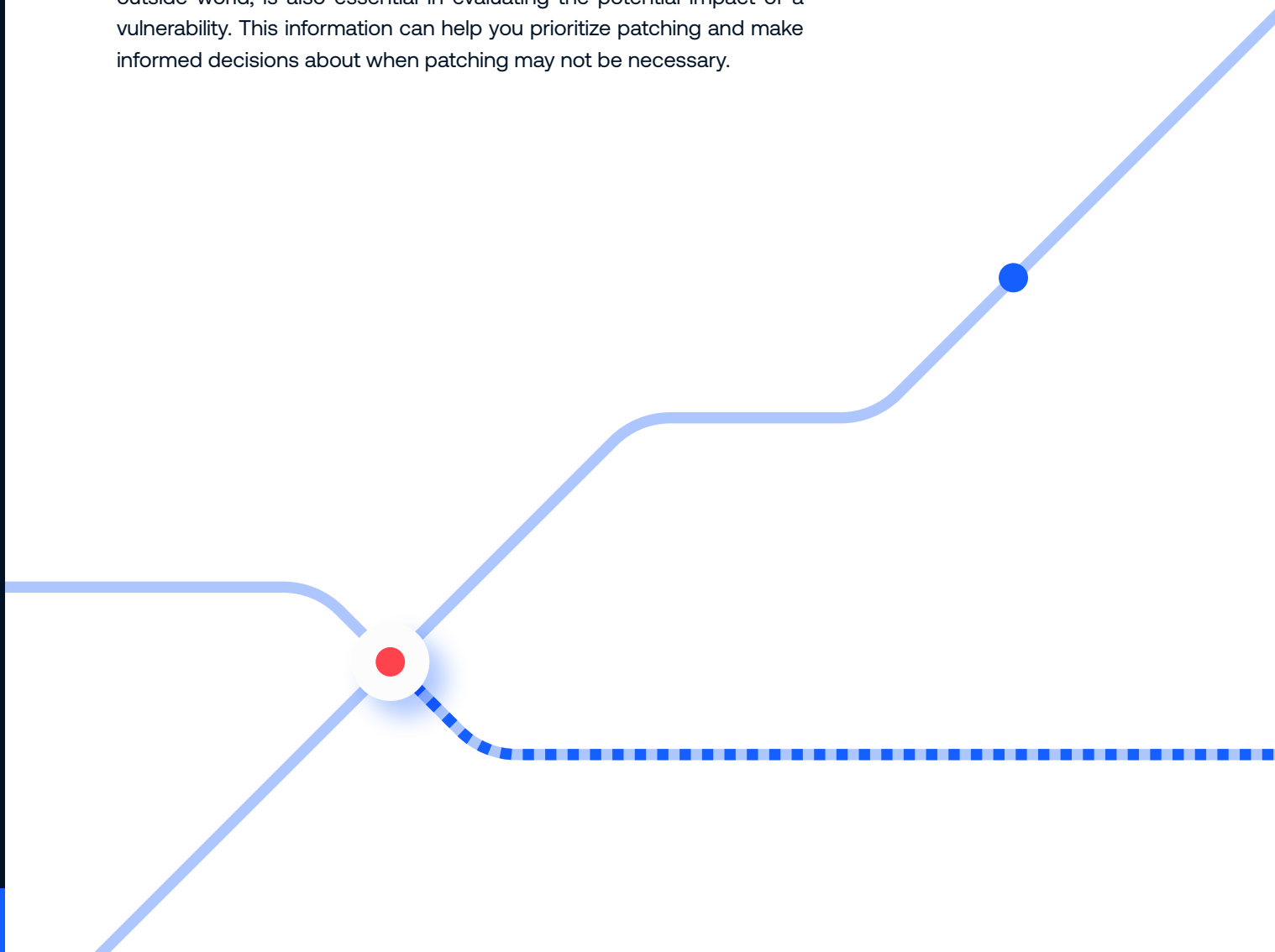
Being informed of the vulnerabilities in specific libraries isn't always enough. The more significant challenge is understanding whether they apply to your system. A Rail Technology network might consist of hundreds of thousands of assets that are only sometimes accurately documented. To understand whether a vulnerability applies to your network, you have to know (1) what assets you have and (2) what software they run (or - their Software Bill of Materials, AKA SBOM).

A Rail Technology network might consist of hundreds of thousands of assets that are only sometimes accurately documented.

A proper asset inventory should include a list of all the assets in your network and a complete understanding of each asset's available type and operational context. This includes information about each asset's role in the rail environment and details about its connectivity and SIL (Safety Integrity Level) ratings.

Knowing the functional type of an asset is crucial for evaluating the potential implications of a vulnerability. For example, if a remote code execution vulnerability is discovered on an interlocking system with a valid attack vector, it could expose the entire system to a safety risk. On the other hand, if the same vulnerability were discovered on a less critical asset, the impact would likely be much less severe.

The operational context of an asset, including information about its safety ratings, connectivity to wireless systems, and connection to the outside world, is also essential in evaluating the potential impact of a vulnerability. This information can help you prioritize patching and make informed decisions about when patching may not be necessary.



Prioritize Effectively

Now, we have an inventory of assets and software and receive notifications about vulnerabilities. What next? Remember that the rail system's status quo used to be: “Don’t patch.”

The result of tracking vulnerabilities on assets, combined with the reasonably outdated software installed on them and the pace of new vulnerabilities reported, might be overwhelming. Therefore, as a railway operator, you need to define a clear patching prioritization strategy.

According to the database, some of the default approaches include taking the default CVSS score and only patching vulnerabilities above a specific score. Such an approach might be suboptimal, as the default scoring does not necessarily reflect the importance of a particular vulnerability in your environment.

We suggest considering additional parameters and incorporating them into your patching prioritization strategy. Such parameters include:

1. **“Crown Jewels”** - Based on your asset inventory, you can choose assets that meet specific criteria and define them as the “crown jewels” essential to your security posture.

2. **(Adjusted) CVSS Analysis** - Vulnerabilities are reported with a CVSS score, often calculated based on CVSS 3.1. This formula is public, and if you look at it, you can see that some parameters are dynamic. Specifically, it includes the “Environmental Score” of the vulnerability, which is a mix of the requirements of the environment, and the ease of exploitation of this environment. In our case, the “Environment” under consideration is the Rail Technology system, and none of the public sources currently evaluates vulnerabilities based on it. **Therefore, adjusting the base CVSS analysis to what is in your environment is highly recommended.**

3. **Known Exploited Vulnerabilities (KEV) is highly recommended** - CISA maintains a Known Exploited Vulnerabilities catalog, which also includes a list of vulnerabilities that have been exploited in the wild, meaning that threat actors have compromised them as part of attack campaigns. This information might be helpful, as it might affect the ranking of vulnerabilities.

4. **Alternative Mitigations** - There might be ways to mitigate vulnerabilities that do not require patching. As patching procedures might be costly and impractical in an isolated and not always accessible environment such as Rail Technology, the alternatives should be evaluated too. We elaborate on this topic in this blog's “Evaluate Alternative Mitigations” section.

5. **Properly Manage Fail-Safe Behavior** - This one is tricky and unique to rail but has to be considered. In traditional IT and software, a category of vulnerabilities triggers fail-safe behavior. In rail, on the other hand, fail-safe behavior was created by design. In the IT world, for example, the system’s default will trigger a stop in an abnormal situation. On the other hand, stopping is a safe situation in rain operations and is not considered abnormal. When rail operators evaluate vulnerabilities, one criterion of vulnerability impact is that impact on availability. This means that a vulnerability might be considered severe if, for instance, it causes a train to stop. If an operator immediately patches it, it might be a redundant effort, as, by definition, the system already contains many conditions that cause the train to stop, and that is just another one. Therefore, when you evaluate the impact of a vulnerability on your environment, compare it to normal operations and system conditions and not just to the base CVSS score that exists online.

As a railway operator, you need to define a clear patching prioritization strategy.

Evaluate Alternative Mitigations

This section is essential, as it might be sufficient and cost-effective to mitigate vulnerabilities with alternative methods or so-called compensating controls.

1. **Tighten Segmentation** - Tightening and/or monitoring the segmentation of vulnerable assets might significantly reduce the risk of exploitation of the vulnerability in question. Segmentation usually includes a definition of and implementation of stricter firewall rules that will reduce an adversary's potential access to the vulnerable system.

2. **Restrict Access to Humans and Applications** - Whenever possible and relevant, restricting access to the vulnerable asset by restricting human and application access is preferred. These actions are effective, as they help reduce the vulnerability's threat surface.

Note that both of the above solutions require changes to the solutions themselves. An additional and effective mitigation strategy is the detection of exploitation attempts. By monitoring exploitation attempts over the network and responding promptly and effectively, a rail operator can also reduce the risks of exploitation arising from unpatched assets.

These actions are effective, as they help reduce the vulnerability's threat surface.

Patch What is Needed

After getting notified and analyzing a vulnerability, ensure you patch the system based on the standard patching procedures. Such procedures include phases: installation on the test environment, comparative patch testing, regression tests, rollback tests, and documentation. TS 50701 offers guidance on implementing patching while ensuring operational requirements are met. It is often necessary to use vendor-approved patches for known vulnerabilities, as the vendors have already tested those to ensure the entire system remains operational.

After getting notified and analyzing a vulnerability, ensure you patch the system based on the standard patching procedures

Final Thoughts

Patch Management of Rail Technology environments is a new process to be introduced to a rail operator's “must have” processes. It should be implemented carefully and in a way that will not exhaust resources. Due to the scale of the process and the requirement to make informed decisions in a timely manner, using tools and automation is recommended.

Solutions like CylusOne can help you build and manage an inventory, receive real-time vulnerability notifications, prioritization, and, most importantly, infuse the crucial rail context into the entire process to ensure you don’t waste your human resources on an infinite chase against vulnerabilities.

Solutions like CylusOne can help you build and manage an inventory, receive real-time vulnerability notifications, prioritization, and, most importantly, infuse the crucial rail context into the entire process