DIRECTORATUL NAȚIONAL DE SECURITATE CIBERNETICĂ

ISACA

Effective Patch Management

Author: Liliana Apetri

Patch management is an area of systems management that involves acquiring, testing and installing multiple patches (code changes) to an administered computer system to maintain up-to-date software and often to address security risk. Patch management tasks include the following: maintain current knowledge of available patches; decide what patches are appropriate for particular systems; ensure that patches are installed properly; testing systems after installation, document all associated procedures, such as specific configurations required.

Several products are available to automate patch management tasks. Patches can be ineffective and can cause more problems than they fix. To avoid problems, patch management experts suggest that system administrators take simple steps, such as performing back-up and testing patches on non-critical systems prior to installations.

Patch Management Lifecycle and Process

Source: ***itsupportguys.com website***

- ***Step 1:*** **Discovery**

Before implementing a patch management process, any IT professional should have a comprehensive network inventory or conduct an IT assessment to understand the types of devices, hardware, systems, operating systems, OS versions, and third-party software and applications in use across his business. As businesses grow, IT resources become strained and it's not uncommon for systems to become neglected or forgotten. Spreadsheets are difficult to keep up with and so internal IT may lose track of the many systems and programs in use.

Not all applications have up-to-date vulnerability listings. This is another reason for application allowed lists and keeping an accurate inventory. System inventories are required to understand the operating systems and critical business functions supported. This includes IoT and IIoT (industrial IoT) devices. Effective patch management requires an accurate inventory. It also requires a manageable list of applications for which IT can manage risk. This is a good argument for creating application allowed lists. Placing an application on an allowed list requires IT and management approval. It also includes removing the users' ability to install any application that does not appear on the list.

- ***Step 2:*** **Categorization & Prioritize**

There is a need to segment the systems and/or users according to their risk level and priority. At the user-level, it can be prioritized the users that frequently need to share, download, or install programs. Specifically, the users that frequently need to share documents over email or online can be categorized as 'high risk' since they are more vulnerable from outside threats.

- ***Step 3:*** **Patch Management Policy Creation**

Developing patching requirements by deciding which systems, users, software needs to be patched, under what conditions and the frequency these systems/users need to be updated. The policy should include:

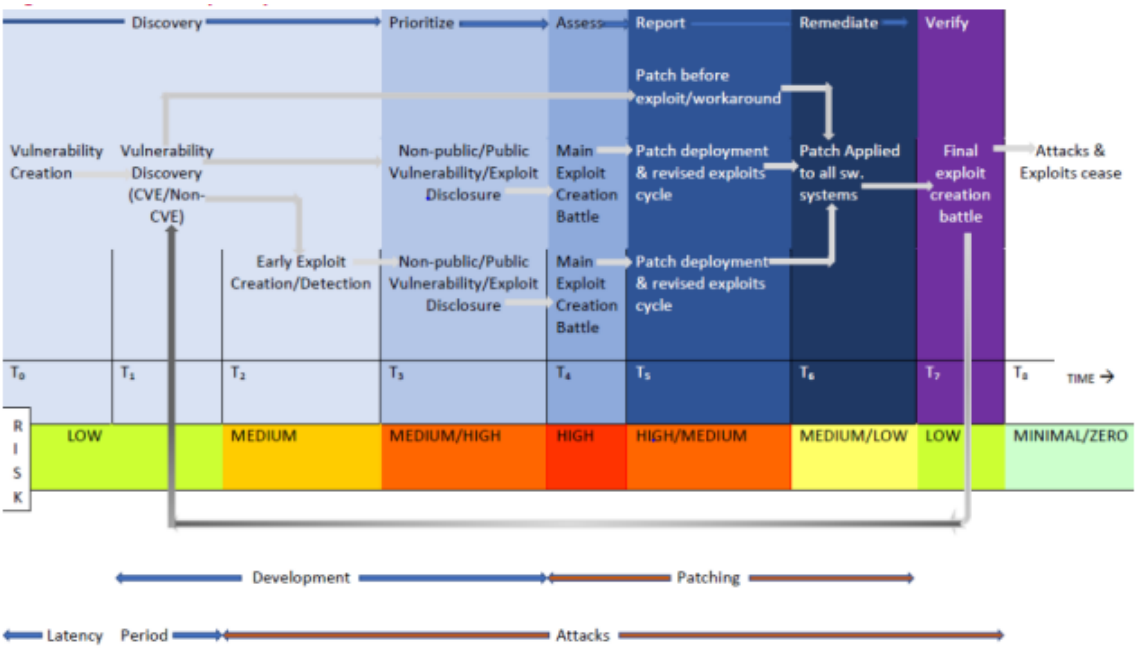
- Inventory requirements
- Responsibilities
- Patch testing
- Patching schedule

In addition to these, the policy should include vulnerability scanning expectations and approaches; and requirements for managing unpatched systems.

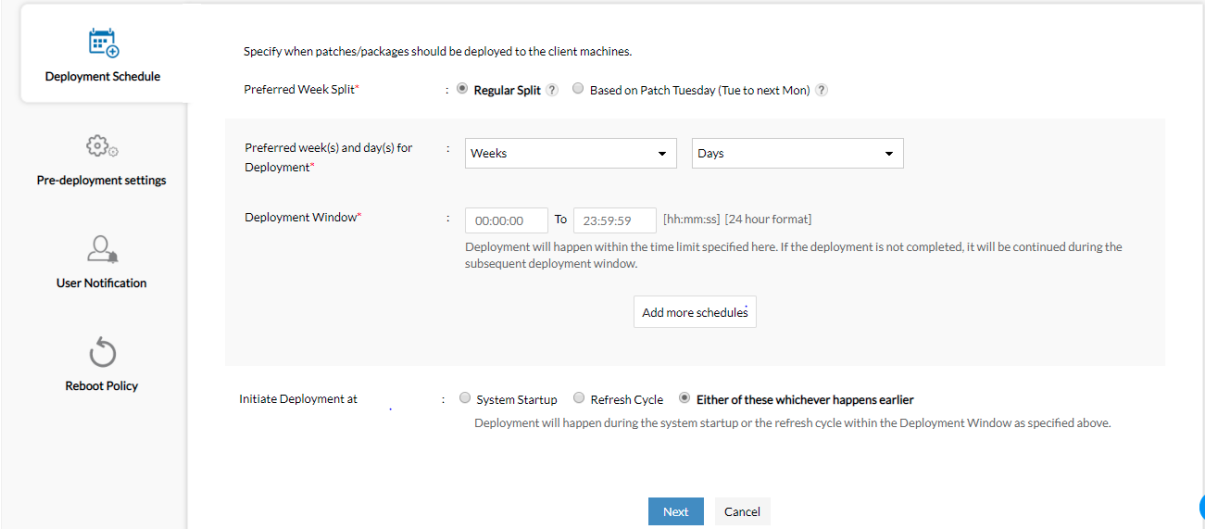
- ***Step 4:*** **Monitoring for New Patches and Vulnerabilities**

The Common Vulnerabilities and Exposures (CVE) defines a vulnerability as: a weakness in the computational logic (e.g., code) found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability. Mitigation of the vulnerabilities in this context typically involves coding changes but could also include specification changes or even specification deprecations (e.g., removal of affected protocols or functionality in their entirety)."

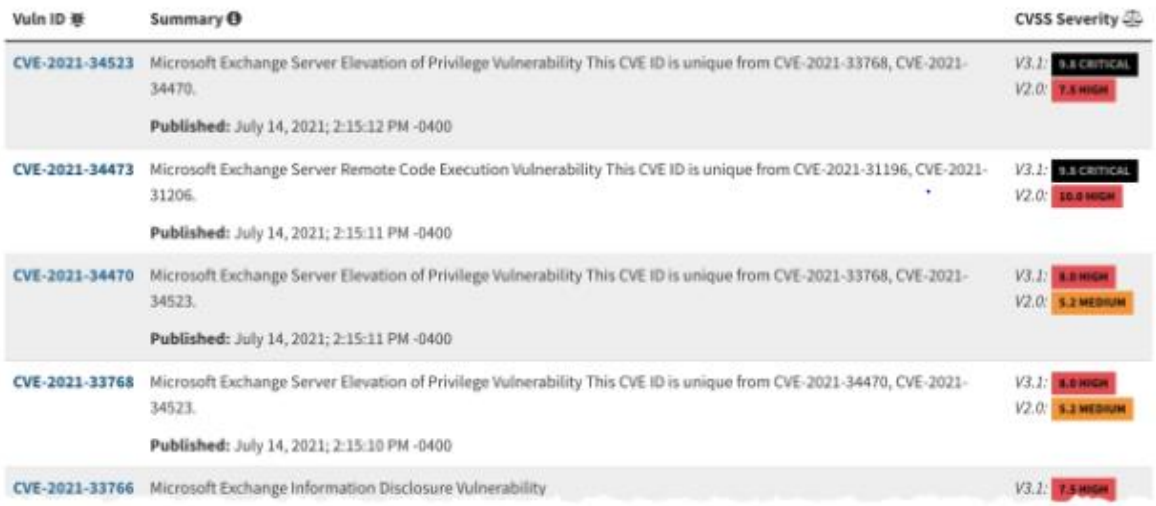
Below is presented the vulnerability management. Mapping the vulnerability lifecycle identifies significant milestones and events that define risk transitioning boundaries. The significance of risks increases as vulnerabilities trigger the creation of the associated exploits and decrease when the patches become available.



Modern businesses utilize a range of systems, software, and digital products, each with their own patch release and vulnerability disclosure schedules. An example related to the disclosure schedule is presented in the figure below.



Most mainstream vendors routinely announce security patches for their products. Organizations must regularly check (daily is best) for patches reported by vendors or by other sources. Organizations can often sign up with vendors for email patch or vulnerability notification. Another approach is to use the National Vulnerability Database (NVD). A security team can search for each application allowed to run on their networks to identify reported vulnerabilities and whether a patch is available.



Vuln ID	Summary	CVSS Severity
CVE-2021-34523	Microsoft Exchange Server Elevation of Privilege Vulnerability This CVE ID is unique from CVE-2021-33768, CVE-2021-34470. **Published:** July 14, 2021; 2:15:12 PM -0400	V3.1: 9.8 CRITICAL V2.0: 7.5 HIGH
CVE-2021-34473	Microsoft Exchange Server Remote Code Execution Vulnerability This CVE ID is unique from CVE-2021-31196, CVE-2021-31206. **Published:** July 14, 2021; 2:15:11 PM -0400	V3.1: 9.8 CRITICAL V2.0: 10.0 HIGH
CVE-2021-34470	Microsoft Exchange Server Elevation of Privilege Vulnerability This CVE ID is unique from CVE-2021-33768, CVE-2021-34523. **Published:** July 14, 2021; 2:15:11 PM -0400	V3.1: 8.8 HIGH V2.0: 5.2 MEDIUM
CVE-2021-33768	Microsoft Exchange Server Elevation of Privilege Vulnerability This CVE ID is unique from CVE-2021-34470, CVE-2021-34523. **Published:** July 14, 2021; 2:15:10 PM -0400	V3.1: 8.8 HIGH V2.0: 5.2 MEDIUM
CVE-2021-33766	Microsoft Exchange Information Disclosure Vulnerability	V3.1: 7.5 HIGH

Figure 1 on NVD Microsoft Exchange Search

Figure below is a partial list of 2021 Exchange vulnerabilities. If a patch is available, it is listed in the Hyperlink section.

Figure 2 Vulnerability Availability

Recommendations related to vulnerability management:

Security and risk management leaders responsible for vulnerability management should:

Address the lack of visibility on infrastructure and assets by working with I&O teams, application owners and other asset owners to close any infrastructure blind spots and to have up-to-date knowledge of assets.

Scan at an optimal frequency that is rational and in sync with the remediation cycle. This frequency should be derived from the organization's risk tolerance, compliance mandates and the number of other asset classes, such as critical infrastructure.

Implement a risk-based vulnerability approach that strategizes the teams' effort and enables them to treat vulnerabilities that are relevant, exploitable and possess significant business risk.

Not assume that patching is the only treatment option. Leveraging compensating controls and risk acceptance is not ideal, but it may be an appropriate alternative. Incorporate network security controls, configuration changes and system hardening, as well as network segmentation, to mitigate the risk of exceptions.

Prepare and present operational and executive metrics that measure performance, prompt actions and convey the value delivered by the vulnerability management capability-e.g. 1. operational outcomes-coverage-time open for exploitation: Average days to patch critical systems with critical patches, Percentage of systems patched in X days, Percentage unable to be patched , Quarterly costs to patch systems

2 Effectiveness Outcomes. Incidents and costs related to incidents: Number of security incidents caused by exploited vulnerabilities, Quarterly costs related to incidents

3.Business Context and Decisions: Investments to support faster or slower patch times

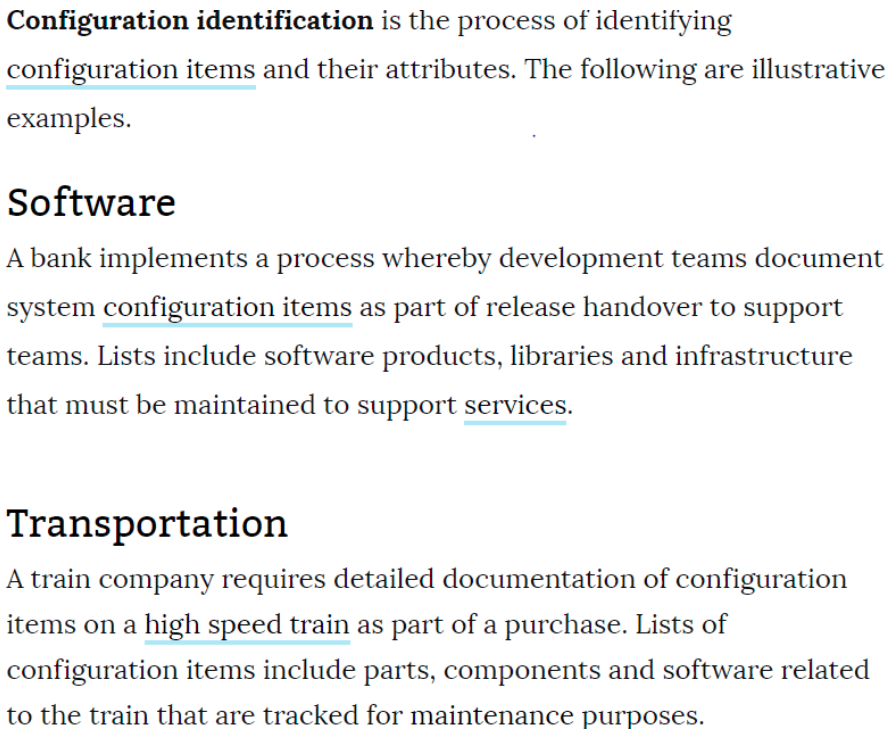
Credibility and defensibility of current outcomes to regulators, shareholders and investors

- ***Step 5:*** **Patch Testing**

Before rolling out patches, especially on mission-critical elements like business servers, it should be necessary to create a non-production test environment, deploy the patch, and monitor for incompatibility or performance issues. If creating a test environment is not possible, it can be used the testing patches on a small segment (two users) to assess if any adverse effects occur.

- ***Step 6:*** **Configuration Management**

After the testing phase, it should be documented the intended changes and results. In this way the IT will quickly identify and troubleshoot unintended changes. Two examples are presented in the figure below.



Source: https://simplicable.com/

• ***Step 7: Patch Roll Out***

Roll out the patch while logging manually or automatically when a patch is applied and the system patched; automated systems can scan managed systems to determine which systems were or were not successfully patched. An example related to the patching status in presented in the figure below.

Server Name	Remediation Plan	Computer Group	Operating System	Patching Status	Server Compliance
Win2k12R2-07	None	Pre-Prod EMEA	Windows Server 2012 R2		62% / 0%
Win2k12R2-06	None	Production US	Windows Server 2012 R2		59% / 0%
Win2k12R2-05	None	Production EMEA	Windows Server 2012 R2		61% / 0%
Win2k12R2-04	None	Pre-Prod US	Windows Server 2012 R2		62% / 0%
Win2k12R2-01	None	Production EMEA	Windows Server 2012 R2		61% / 0%
win-20i15j64jft	None	Pre-Prod EMEA	Windows Server 2012 R2		62% / 0%
RND-RH7-07	None	Redhat	Red Hat Enterprise Linux 7.x		100% / -
RND-RH7-05	None	Redhat	Red Hat Enterprise Linux 7.x		100% / -
RND-RH7-03	None	Redhat	Red Hat Enterprise Linux 7.x		100% / -
RND-RH7-01	None	Redhat	Red Hat Enterprise Linux 7.x		100% / -

Source: https://jetpatch.com/

• ***Step 8:*** **Patch Auditing**

Post-patch rollout is a moment to identify any failed or pending patches. These should be monitored for incompatibility issues. It can be useful to reach out two a few tech-savvy end users that can help provide feedback if needed. See the example above.

• ***Step 9:*** **Reporting**

Each business unit has stakeholders, IT is no different. A monthly patch compliance report should be shared with the C-Suite and executives when needed. This will ensure everyone understands the importance of patch management.

Example: Report on compliance & vulnerability requirements

Compliance regulations have many requirements to ensure financial, health, or other personal data is secure in networks and systems. Failing to comply or demonstrate compliance

can mean serious fines, angry customers and lost business. It can be used automated, formatted reports auditors need to demonstrate compliance for the multiple requirements in PCI DSS, HIPAA, SOX, GLBA, PSN, and CoCo regulations.

- ***Step 10:*** **Review, Optimize, and Repeat**

As with most business processes – periodically review, update and repeat steps one through 9. It should be identified the systems that have reached their End-of-Life (EOL), outdated hardware/machines, review policies quarterly, and revise as needed to ensure the effectiveness of the patch management policy.

Patch Vulnerabilities by The Numbers

According to Forrester's State of Application Security Report, application vulnerabilities are the most common external attack method, making patch management critical to the company's overall security. In fact, according to the Ponemon Institute, 57% of cyberattack victims report that their breaches could have been prevented by installing an available patch and even more chilling, 34% of those victims knew of the vulnerability, but hadn't taken action.

Info Security Magazine reported that more than 18,000 Common Vulnerabilities and Exposures (CVEs) were published last year alone. This number also reflects a 6% rise in CVEs reported in 2020 compared to 2019. As technology continues to move more quickly, it's expected so will potential vulnerabilities. With so many gaps in such a dynamic landscape, it can feel impossible to stay on top of making sure the company applications are safe from attacks.

Beyond the sheer volume of constant new vulnerabilities, patching can be time consuming. A patch needs be tested first to make sure it 1. Works to eliminate the vulnerability and 2. Doesn't adversely affect your installed software. In fact, 74% of companies say they simply can't patch fast enough. In fact, the average time to patch is 102 days according to Ponemon.

Recommendations

- IT management needs to define policies that governs the patch management activities within the organization including who, how and when patches are tested and applied into production systems -***Patch Management policy***

- IT needs to know every asset in its environment in order to identify which patches are needed when vendors make them available- ***Asset's inventory***

- A procedure and a lab environment are required to test patches before applying it into the production environment- ***Patch testing***.

-The complexities of the modern IT stack, with its numerous points of integration, customized pieces, add-ons, etc. that are often spread among multiple locations as well as mobile endpoints, make patching more complicated. Access to the infrastructure component map is required to properly manage the patch testing and installation processes- ***Structure and planning***

- A typical IT department has many workers who apply patches as part of their portfolio of responsibilities; as a result, patch management can become a task done by many but owned by no one. It is difficult for an enterprise to have a strong patch management process without clear accountability- ***Ownership and accountability***

- A strong patch management discipline should include a way to identify and document patches as they are released by vendors, when they are scheduled to be tested and deployed in the enterprise, and when the patches have been completed- ***Document***.

Source:

IT Support guys website

CISA manual

ENISA -Effective Patch Management

The Importance of Patching and Best Practices for 2021 | CNP Technologies: Data, Voice and Security Services

GFI.com website

https://www.toolbox.com/it-security

https://simplicable.com/

https://jetpatch.com/

ENISA-State of vulnerabilities

Gartner-The Essential Elements of Effective Vulnerability Management