Helena Koskenkorva

THE ROLE OF SECURITY PATCH MANAGEMENT IN VULNERABILITY MANAGEMENT

Master's thesis

Master of Engineering

Cybersecurity

2021

South-Eastern Finland
University of Applied Sciences

South-Eastern Finland
University of Applied Sciences

Author	Degree	Time
Helena Koskenkorva	Master of Engineering	November 2021

Thesis title

The role of security patch management in vulnerability management

79 pages
7 pages of appendices

Commissioned by

Global IT and business consulting services firm

Supervisor

Senior Lecturer Vesa Kankare

Abstract

The thesis was commissioned by a global IT and business consulting services firm, referred to as the commissioner in the report. The objective was to understand the role of security patch management in the vulnerability management domain and to determine possible development suggestions in the present state.

The main goal of the theory part was to discuss the concepts of risk management, vulnerability management, and security patch management to provide a deeper theoretical understanding of these concepts and create a picture of how these three concepts intertwine.

The research objective was to gain in-depth and detailed information on the studied case, to solve an identified problem while not progressing to concrete solution implementation. Thus, a case study was used as a research method. Research data was gathered through semi-structured interviews (n=8), direct observation, and document reviews (n=4).

The study showed that security patch management has a significant role in vulnerability management as it acts as a remediation plan within vulnerability management. Furthermore, having a risk-based approach to vulnerability management is strongly present. Thus, the focus should also shift towards a risk-based security patch management strategy. Therefore, the need for effective and efficient risk management becomes evident; by being the initialising and unifying force in intertwining vulnerability management and security patch management.

Keywords

Security patch management, vulnerability management, risk management, risk-based approach

CONTENTS

ABBREVIATIONS

CIS	Center for Internet Security
CM	Configuration Management
CMDB	Configuration Management Database
CPE	Common Platform Enumeration
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
CWE	Common Weakness Enumeration
ENISA	European Union Agency for Cybersecurity
EPSS	Exploit Prediction Scoring System
FIRST	Forum of Incident Response and Security Teams
HIPAA	Health Insurance Portability and Accountability Act
ICAT	Internet – Categorization of Attacks Toolkit
ISMS	Information Security Management System
ISO	International Organization for Standardization
ITIL	Information Technology Infrastructure Library
LTSC	Long-Term Servicing Channel
ML	Machine Learning
MSP	Managed Service Provider
NIST	National Institute of Standards and Technology
NVD	National Vulnerability Database
OOB	Out-of-band
OS	Operating System
OWASP	Open Web Application Security Project
PCI DSS	Payment Card Industry Data Security Standards
RBVM	Risk-Based Vulnerability Management
RDP	Remote Desktop Services
SAC	Semi-Annual Channel
SCAP	Security Content Automation Protocol
SLA	Service Level Agreement
SSU	Servicing Stack Update
WSUS	Windows Server Update Services
WU	Windows Update

1 INTRODUCTION

Vulnerability is a flaw in the system, providing a possibility for an exploit. The destructiveness of an exploit is dependent on the nature of the vulnerability. It could, for example, gain privileges, execute code, allow lateral movement, or exfiltrate data. In a worst-case scenario, it goes undetected and starts causing havoc. Here, having appropriate controls present to mitigate potential risks cannot be stressed enough. Also, today's enterprise IT infrastructures have become increasingly complex due to the expansion of the service sector from traditional infrastructure services to hybrid IT infrastructures. The hybrid IT infrastructures can have elements, for example, from on-premises, cloud, and edge, hence, simultaneously broadening the attack vector. Thus, the importance of risk management, vulnerability management, and security patch management becomes evident.

The commissioning organisation is a global IT and business consulting services firm. Furthermore, the direct commissioner is a unit offering IT outsourcing and infrastructure services in Finland, evaluating ways to improve the visibility and efficiency of vulnerability management in their IT infrastructure service operations. Vulnerability management is an utmost important subject from a service provider's perspective as it has to cover the service provider's own IT infrastructure as their client's. The commissioner examined vulnerability management from a broad perspective and identified security patch management as one sector requiring further investigation.

The focus of this thesis was on creating a deeper understanding of the role of security patch management in the vulnerability management domain, concentrating on Microsoft Windows Server Operating System (OS) security patching from a technological perspective. The view was on shifting the focus from traditional vulnerability management towards assessing the risks created by vulnerabilities, thus, explaining how risk management, vulnerability management, and security patch management intertwine to create a more holistic and in-depth approach to protect one's IT environment.

2 RESEARCH PROBLEM

2.1 Research questions

The commissioner continually examines vulnerability management from a larger perspective as the vulnerability management domain constantly changes. In the present state, the commissioner has identified several issues that require further examination. These include, for example, reacting to vulnerability scanning reports, public vulnerability announcements, performing CVE tracking to verify the correct patching status, and responding to clients' vulnerability related enquiries. Multiple specialists are using a significant number of person-hours to react to these identified issues. Thus, ways to improve the visibility and efficiency of vulnerability management are examined.

A struggle applying a remediation strategy to the increasing amount of publicly disclosed vulnerabilities is constantly present. Here, efficient security patch management policies and procedures are required to remediate known vulnerabilities. Due to this, the commissioner identified security patch management as an area for further investigation. Based on these considerations, the research problem was formulated and presented as follows: *The role of security patch management in vulnerability management requires more understanding and development.*

Following research questions were formed from the stated research problem:

- What is security patch management, and what is its role in vulnerability management?
- How do risk management, vulnerability management, and security patch management intertwine?
- How are the security patching duties divide between the service provider and their client?
- What development suggestions arise from the findings?

Finding answers to the mentioned research questions should provide a solution to the research problem.

2.2 Research objectives

Several research objectives can be identified for this thesis as multiple outputs can be received by answering the stated research questions. Firstly, a deeper theoretical understanding of the security patch management process and best practices is obtained. In addition to this, the possibility to reflect the theory against the commissioner's existing processes and procedures becomes available, enabling correlation against industry best practices.

Secondly, examining the duty division between the commissioner and their clients could lead to, for example, a revision of the service description for server management services.

Thirdly, understanding the current state of the target environment provides the ability to examine if the disclosure of development suggestions could add value to the commissioner's clients. Finally, the findings could present new study cases where the aim could be on solution implementation, for example, through a development project.

3 RESEARCH FRAMEWORK

3.1 Case study research

Case study research produces information about a present phenomenon happening in its natural operational environment. The case study uses multiple sources of evidence. It aims to provide in-depth and detailed information about the studied case and solve an identified problem. Concrete solution implementation is not part of the case study research. The research problem is mainly studied, described, and explained by asking *how* and *why* questions. The case study research most commonly progresses (Figure 1) in four phases. (Ojasalo et. al. 2015, 52-53.)

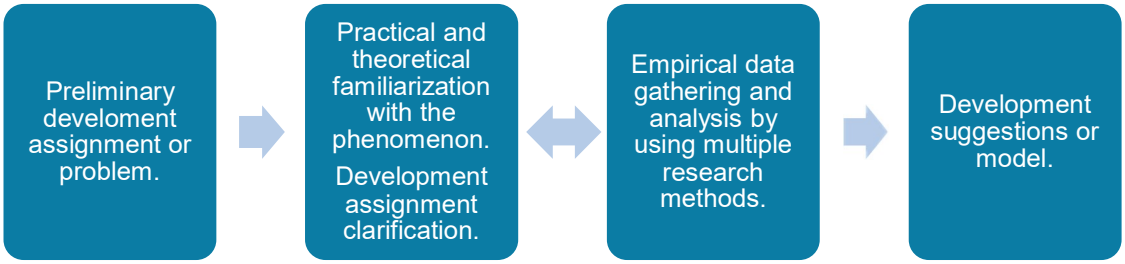

Figure 1. Case study research progress phases (Ojasalo et. al. 2015)

The preliminary development assignment or problem is often formed by a person who is, to some extent, already familiar with the phenomenon. Practical and theoretical familiarization with the phenomenon is required to understand the actual research problem. By attaining further knowledge, it is possible to make clarifications to required research questions. This will help in determining the required data gathering and analysis methods needed to solve the research problem. The progress phases are not strictly fixed. Adjustments to the research problem and/or questions can be made based on gathered research data. Finally, research data analysis will lead to development suggestions or model. (Ojasalo et. al. 2015, 53-54.)

3.2 Data gathering

In traditional qualitative research, it is possible to solve the research problem with a single qualitative research method. This is not the case with case study research because the diversity and complexity of the studied problem cannot be solved with a single research method. Instead, multiple methods are needed in data gathering and analysis to find a solution to the research problem. Hence, both qualitative and quantitative research methods are often utilised in data gathering. (Kananen 2013, 56-57.) Several data gathering methods (Figure 2) are used to create an in-depth understanding of the phenomenon (Kananen 2013, 77).

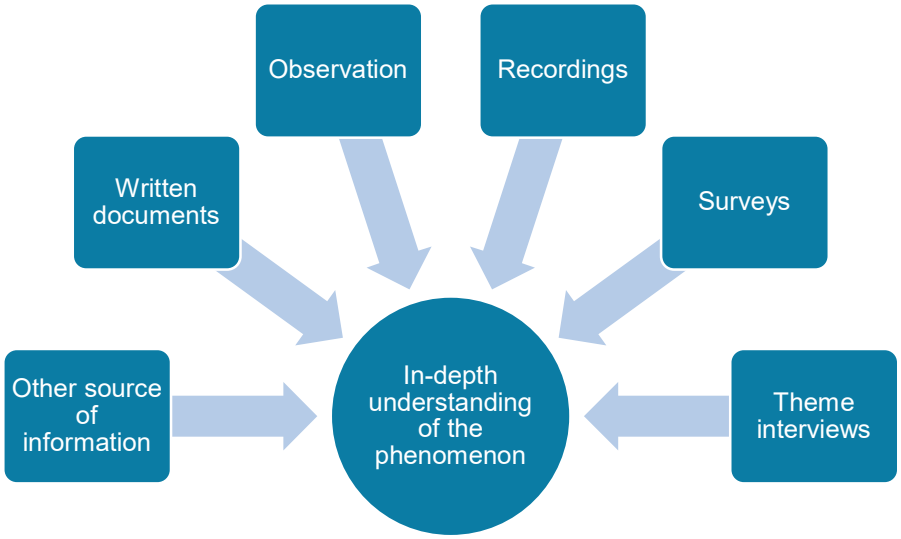

Figure 2. Data gathering methods of case study research (Kananen 2013, 77)

Kananen (2013, 80) states that in structured interviews it is always possible that the interviewer influences the research results. This happens, for example, through themes and question selection. Thus, qualitative data gathering methods used in this thesis include semi-structured interviews, direct observation, and document reviews. A thesis diary was used to document overall progress.

3.3 Research reliability

Research reliability is discussed in this chapter based on the writings of Kananen (2013, 114-122). The case study does not have reliability criteria of its own. Thus, these are conducted from either qualitative or quantitative research methods. The used criteria change accordingly if both research methods are used in the research. Confirmability, ratability, consistency of interpretation, and saturation are reliability criteria of the quantitative research methods. Documentation is the key to reliability examination, meaning that all phases of the research process are thoroughly documented.

As was presented in Figure 2, the case study has multiple data gathering methods. Using several data sources is one way to ensure research reliability. Reliability is achieved by cross-referencing data between several data sources to understand and explain the studied phenomenon and verify the validity of research results.

The qualitative research method reliability criteria are used to examine the research reliability during the thesis process phases. This will be done by using multiple sources as source material and by creating confirmability by proofreading the text and research results with the interviewees. Interview and observation notes and thesis diary create the basis of ratability through documentation.

4 RISK MANAGEMENT

The foundation of a security program framework is built by many entities. It is built with logical, administrative, and physical protection mechanisms. These mechanisms include procedures, business processes, and people. All of these elements need to work together to provide a protection level for the environment. Building this foundation from scratch and without blueprints would be an enormous task. Today there are several standards, ISO/IEC 27000 series being perhaps the most recognised one, best practices, and frameworks assisting on this daunting task. The goal is to guide how an information security management system (ISMS) should be built and maintained. (Harris & Maymi 2019, 13-17.) ISO/IEC 27000:2020 (2020, 11-12) states:

> An ISMS consists of the policies, procedures, guidelines, and associated resources and activities, collectively managed by an organisation, in the pursuit of protecting its information assets. It is a systematic approach for establishing, implementing, operating, monitoring, reviewing, maintaining, and improving an organisation's information security to achieve business objectives.

Risk management, vulnerability management, and configuration management are all controls outlined by an ISMS.

4.1 The concept of risk management

Risk can be described as a probability of a negative occurrence that is caused by external or internal vulnerabilities. These vulnerabilities may be avoided through pre-emptive actions. Identification of pre-emptive actions requires the process of risk evaluation. It covers systematic measures to identify the possible risks and to evaluate the risk probability. Furthermore, there are four general approaches to

risk. The first and perhaps the most commonly used approach is accepting the risk and the consequent losses. The second approach is to perform the necessary actions to eliminate the risk and thus avoiding the loss. The third approach is to mitigate or reduce the effect of the risk. Finally, the fourth option is to transfer the risk to another party. (Kohnke et. al. 2016, 191.)

Risk management is a formal, continuous, and complex organisational process. It ensures the security of any business operation. Furthermore, it is an information gathering and decision-making function. The specific goal is to identify, analyse, mitigate, and monitor each active and latent risks that are known to exist within the organisation. (Kohnke et. al. 2016, 183.)

ISO 31000 guides that organisation should build the foundation for managing risk by defining the principles, framework, and process as shown in Figure 3 (SFS-ISO 31000:en 2018, 5). The principle elements guide the characteristics of effective and efficient risk management. These are the foundation for managing risks. The risk management framework assists in integrating risk management into significant activities and functions. Successful framework development requires support and commitment from stakeholders, especially from top management. Systematic application of policies, procedures, and practices are involved in the risk management process. These guide the activities of communicating and consulting; establishing the context; and assessing, treating, monitoring, reviewing, recording, and reporting risk. (SFS-ISO 31000:en 2018, 7-14.)

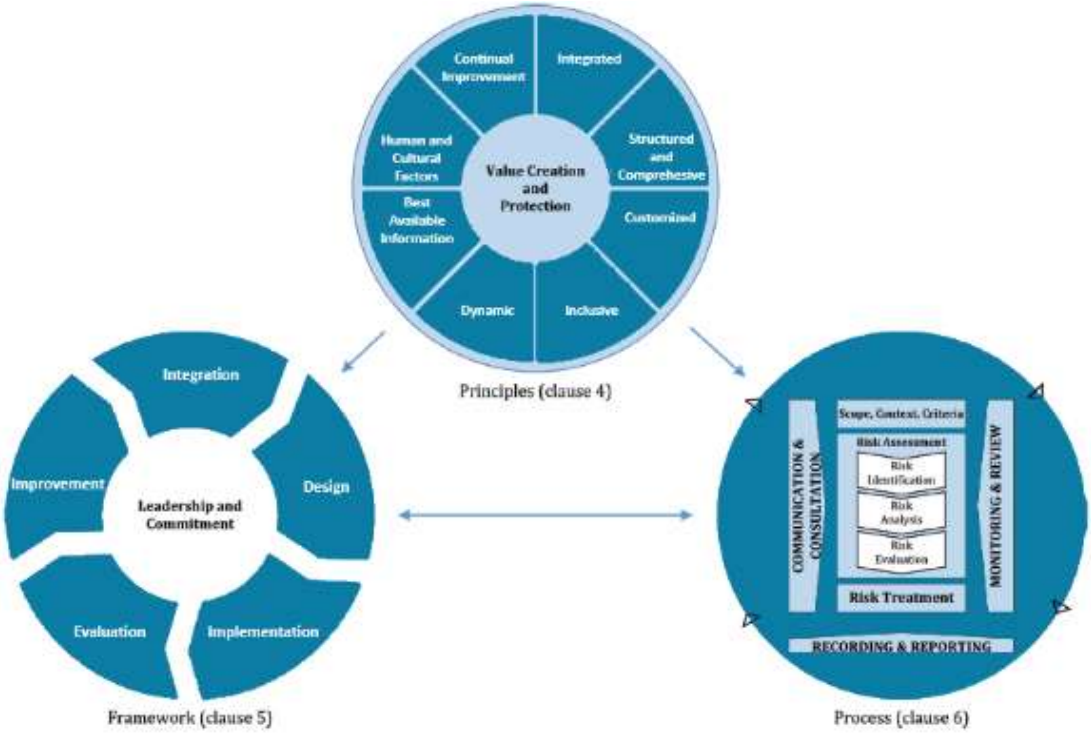

Figure 3. Principles, framework, and process for managing risk (SFS-ISO 31000:en 2018)

The risks are identified, analysed, and evaluated to create a complete picture of the required steps to mitigate the overall level of risk to an acceptable level. The organisation should be aware of several risk types from the following major categories: physical damage, human interaction, equipment malfunction, inside and outside attacks, misuse of data, loss of data, and application error. Furthermore, it is imperative to evaluate the risk against the protection level of the information. (Harris & Maymi 2019, 93.)

An organisation should ensure that all risk information obtained from the risk management activities is constantly monitored and reviewed to identify any changes in the context and maintain an overview of the complete risk picture. Monitoring should include, for example, new assets that have been included in the risk management scope, information security incidents, and new threats that could be active both outside and inside the organisation which have not been assessed. (SFS-ISO/IEC 27005:en 2018, 25-26.)

4.2 Risk management process

Berman (2014, Chapter 1) states that “a process is a set of interrelated activities designed to transform inputs into outputs.” It is a critical component of a successful security program, and it guides the use of technology. A process should not be designed to purely operate on technology. It should be designed to produce an outcome that supports the organisation’s objectives. (Foreman 2010, Chapter 6.) Risk management is a widely discussed topic, and the ways to represent the risk management process are many. However, the basic steps are all similar. This chapter explains two different risk management process approaches.

Firstly, NIST special publication 800-39 (2011, 6-7) describes four components that are required when forming a risk management process. The first component addresses how organisations *frame* risk or establish a risk context. This requires the identification of risk assumptions, risk constraints, risk tolerance, and priorities and trade-offs.

The second component addresses how organisations *assess* risk. This is done within the context of the organisational risk frame. This requires the identification of threats, internal and external vulnerabilities, the harm caused by potential threats exploiting vulnerabilities, and harm likelihood. The risk assessment component is supported by identifying the tools, techniques, and methodologies that are used to assess risks; the assumptions related to risk assessments; the constraints that may affect risk assessments; roles and responsibilities; how risk assessment information is collected, processed, and communicated; how risk assessments are conducted; the frequency of risk assessments; and how threat information is obtained. (Joint task force transformation initiative 2011, 6-7.)

Risk *response* is the third component of the risk management process. The purpose is to develop alternative courses of action for responding to risk, determining appropriate courses of actions consistent with risk tolerance, and implementing risk responses. The risk responses can, for example, be accepting,

avoiding, mitigating, sharing, or transferring risk. (Joint task force transformation initiative 2011, 6-7.)

The fourth component addresses how risk is *monitored* over time. Risk monitoring is used to verify that planned risk response measures are implemented, and derived information security requirements are satisfied; determine the ongoing effectiveness of risk response measures; and identify risk-impacting changes to information systems and the environments in which the systems operate. (Joint task force transformation initiative 2011, 6-7.)

Figure 4 illustrates the four components of the risk management process and how the flow of information and communication flow between these components. It presents risk *framing* as the component which informs all the sequential step-by-step activities, moving from risk *assessment* to risk *response* to risk *monitoring*. An efficient risk management process requires bidirectional and flexible information and communication flows and execution order between the four components. This is required as the risk management process is a dynamic and developing discipline by nature. (Joint task force transformation initiative 2011, 8.)

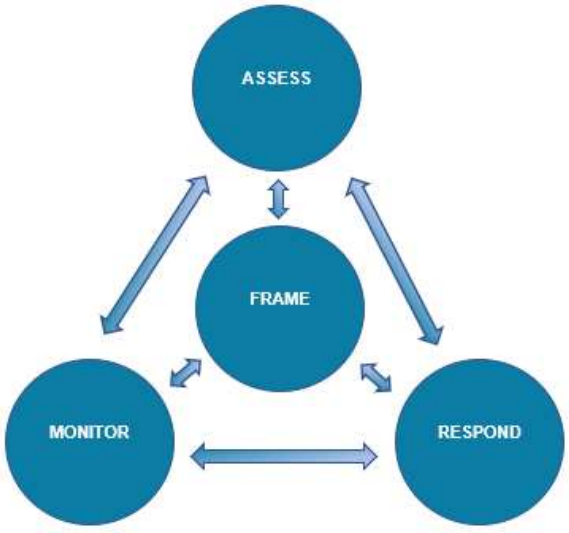

Figure 4. Risk management process and the information and communication flow among components (Joint task force transformation initiative 2011)

Secondly, Kohnke et. al. (2011, 182-183) states that risk identification is the groundwork for risk management process creation. Here possible risks are

uncovered, recognized, and described. Thorough risk/threat assessment ensures that all risks in the organisation's risk environment are correctly identified and categorised. After completing initial identification and characterisation, the risk management process involves five generic steps shown in Figure 5.

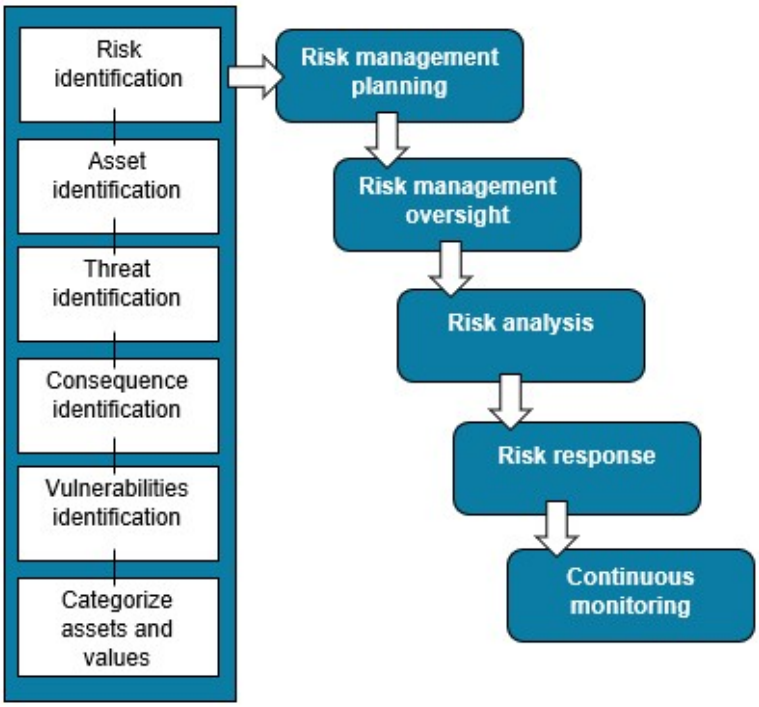

Figure 5. Five elements of the risk management process (Kohnke et. al. 2016)

Risk management planning is the first step to be taken when an organisation wants to ensure that the risk management process will support the organisation's business operations. The risk management plan is a high-level document that shapes the risk management process. It provides information that guides the strategic decision making about risk. It ensures that an accepted and systematic set of policies and procedures are in place to handle known risks. It details the overall approach that will be employed to control the risks worth addressing. After formal and detailed risk management planning, the organisation has created a framework of detailed policies and procedures which comprise the risk management process. (Kohnke et. al. 2016, 183.)

Risk management process oversight is the formal oversight process that ensures the sustainment of the security strategy. This oversight process monitors and reports the organisation's risk situation. Newly appeared risks should also be

distinguished. The threat environment should be assessed regularly to ensure that the current risk mitigation schema is appropriate and successfully mitigating threats. (Kohnke et. al. 2016, 185.)

Risk analysis is the information gathering function. It identifies and evaluates each relevant risk. The risk is evaluated to determine the risk magnitude which is the combination of likelihood and consequence. The controls that are needed to respond to the risks are itemised. Risk analysis is one of the most important phases of risk management since systematic risk analysis can direct the prioritisation of the steps that the organisation will deploy to do risk management. (Kohnke et. al. 2016, 185.)

During *risk response*, the risk is treated by creating risk mitigation strategies, preventive, and contingency plans. The targets of the risk response have to be precisely established. Finally, the existing threat environment should be under *continuous monitoring*. This is necessary to identify and mitigate any new threats that might arise. (Kohnke et. al. 2016, 185-186.)

To implement a comprehensive risk management process, the process should be integrated throughout the organisation. NIST special publication 800-39 (2011, 9-11) addresses this issue with a three-tiered approach to address risk as shown in Figure 6. The purpose is to achieve seamless use of the risk management process across the three tiers. Here, the focus is on continuous improvement in the risk-related activities, and effective inter-tier and intra-tier communication among all stakeholders.

Tier 1 is the organisational perspective that implements the first component of risk management. The risk management activities include the establishment and implementation of a risk executive; the establishment of the risk management strategy; the development and execution of organisation-wide investment strategies for information resources and information security. (Joint task force transformation initiative 2011, 9.)

Tier 2 includes mission/business process risk management activities include defining the mission/business processes; prioritising the mission/business processes; defining the types and criticality/sensitivity of the information and the information flows; establishing an enterprise architecture with embedded information security architecture. (Joint task force transformation initiative 2011, 10.)

Tier 3 incorporates information system risk management activities that include categorisation of organisational information systems; allocating security controls to organisational information systems and the environments; managing the selection, implementation, assessment, authorisation, and ongoing monitoring of allocated security controls. (Joint task force transformation initiative 2011, 10.)

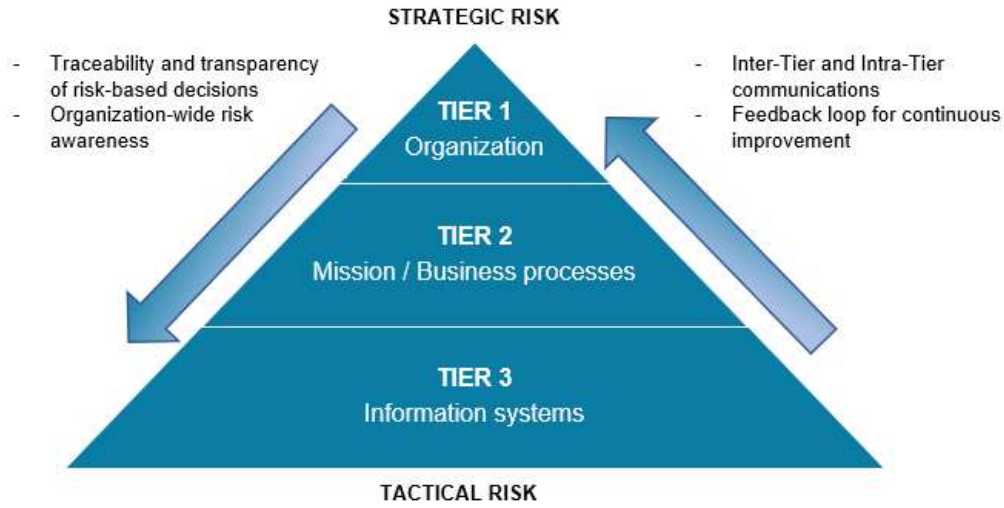

Figure 6. Multitiered organisation-wide risk management (Joint task force transformation initiative 2011)

The risk management process can be carried out in two different ways, by choosing an ad hoc or coordinated approach to risk management. Ad hoc risk management, which many organisations choose to use, is the starting point for a new or undocumented repeat process where security controls are created to fulfil specific security requirements. Despite being a cost-efficient approach, it most certainly results in flawed protection since the organisation is reacting to events rather than deploying coordinated protection. The deployment of a coordinated set of controls is a difficult undertaking which is why a coordinated approach to risk management is many times disregarded even though it offers better security. (Kohnke et al. 2016, 197.)

5 VULNERABILITY MANAGEMENT

It is essential to understand the different security concepts and their relationship before diving deeper into the vulnerability management concept. *A threat agent* is an entity that takes advantage of a vulnerability that gives rise to a threat. *A threat* is any potential danger that can breach the security of a system with the exploitation of a vulnerability. *A vulnerability* is a weakness in an information system, hardware, system security procedure, internal controls, or human weakness that allows a threat source to compromise its confidentiality, integrity, and availability. Exploit of such vulnerability leads to *risk.* Risk ties the vulnerability, threat, and likelihood of exploitation to the resulting business impact and hence can cause damage to a business *asset*. A vulnerability exposes to possible damages, and *exposure* is the instance exposed to losses. Potential risks can be mitigated by implementing *safeguards*. Countermeasure is used to eliminate a vulnerability or reduce the likelihood of a threat agent being able to exploit a vulnerability. Figure 7 illustrates the relationship between these security concepts. (Harris & Maymi 2019, 6-8.)

Figure 7. The relationship between different security concepts (Harris & Maymi 2019, illustration Jönsas 2021)

One should accept the fact that a vulnerability-free information system does not exist. Information system vulnerabilities exist in software, processes, and people. (Harris & Maymi 2019, 990-991.) Vulnerabilities related to corporate strategy, economics, and the environment do also exist. These require the attention of a risk manager. Hence, vulnerability management is part of the risk management process. (Foreman 2010, Chapter 1.)

5.1 The concept of vulnerability management

Vulnerability management as an IT discipline is rather immature. The first vulnerability scanners were released in the late 90s and early 2000s (Horev 2019). At that time, in the year 2000, 1020 disclosed security vulnerabilities were reported. In recent years, the number of known vulnerabilities has exploded, and the year 2017 became a milestone with nearly 15000 disclosed vulnerabilities. (CVE details n.d.) From a vulnerability management perspective, the year 2017 was challenging as WannaCry and NotPetya caused significant damage to some of the biggest organisations in the world. Both attacks utilised the EternalBlue exploit. Equifax suffered a highly publicised breach when attackers compromised Apache Struts vulnerability on a public-facing web server. It is noteworthy that the exploits occurred while a patch was released for both vulnerabilities. (Williams 2019, 2.)

It is still typical that people equate vulnerability management as a periodical vulnerability scanning against their information systems. This is far from the truth, as, according to Harris and Maymi (2019, 990), vulnerability management is “the cyclical process of identifying vulnerabilities, determining the risks they pose to the organisation, and applying security controls that bring those risks to acceptable levels.” Vulnerability management must be considered as a critical component of a holistic information security program (Williams 2019, 1).

A vulnerability management program can be implemented, for example, by utilizing the Threat and Vulnerability Maturity Model, which is a combination of asset analysis, vulnerability scanning, patch management, process implementation, and metrics. The model consists of the following six levels:

- Level 0: *non-existent,* in which no real strategy for tackling vulnerabilities exists.
- Level 1: *scanning,* in which vulnerability assessment solution is in place and run regularly.
- Level 2: *assessment and compliance,* in which a structured strategy to handle emerging vulnerabilities exists, and a vulnerability management program is built with the help of compliance requirements.
- Level 3: *analysis and prioritisation,* in which published vulnerabilities are analysed, and the prioritisation is determined by risk.
- Level 4: *attack management,* in which vulnerabilities and patching are considered as a complete ecosystem where risk is assessed holistically.
  Level 5: *business-risk management,* in which a fully developed management program, that takes the entire environment into account, exists.

By progressively advancing from one level to another, the organisation can enable an understanding of how adversaries act, what vulnerabilities exist within the organisation, how this combination exposes risks to critical assets, and how these risks can be managed and mitigated. (Core Security n.d.)

Another approach could be to follow The Open Web Application Security Project (OWASP) Vulnerability Management Guide which focuses on building vulnerability management in more manageable repeatable parts. Detection, reporting, and remediation form three cycles, and each cycle is a domain that comprises four main processes. Detection cycle processes are scope, tools, run tests, and confirm findings. Reporting cycle processes are asset groups, metrics, audit trail, and reports. Remediation cycle processes are prioritise, remediate, investigate false positives, and control exceptions. The cyclical nature implies continuous process improvement; a single process feeds into other processes, and all tasks are interconnected across the three domains. (The OWASP® Foundation 2020, 3, 17.)

Digital transformation constantly changes the threat landscape. Thus, vulnerability management must adapt accordingly. The CIS Controls™ are a prioritised set of actions that collectively form defence-in-depth best practices. These practices mitigate the most common attacks against information systems and networks. Continuous vulnerability management is the third CIS Control. It

contains the following sub-controls that are the specific actions to be taken to implement this control (Center for Internet Security 2019, 1, 16):

- running automated vulnerability scanning tools
- performing authenticated vulnerability scanning
- protecting dedicated assessment accounts
- deploying automated OS patch management tools
- deploying automated software patch management tools
- comparing back-to-back vulnerability scans
- utilising a risk-rating process

SANS has created a framework outlining five focus areas as being part of a successful vulnerability management program. These five focus areas form the PIACT process illustrated in Figure 8. *Prepare* (P) focuses on defining, building, and continuously improving the program. *Identify* (I) focuses on finding the vulnerabilities that are present within the organisation's operating environment. Vulnerability identification can be automated, manual, or externally performed. The identified vulnerabilities need to be analysed and prioritised. This is done in the *analyze* (A) phase that consists of two sub-areas, prioritisation, and root cause analysis. The analysis results should be presented to all stakeholder groups to create an understanding of the corresponding risks and treatment options. *Communicate* (C) phase contains two sub-groups which are metrics & reporting and alerting. Implementation, testing, and monitoring solutions to address the vulnerabilities are done during the *treat* (T) phase. This phase consists of three sub-areas: change management, patch management, and configuration management. (Risto 2020.)

Figure 8. PIACT process (Risto 2020)

Depending on the organisation, for example, ISO/IEC 27000 series, NIST special publication 800-53 controls, and regulations like Payment Card Industry (PCI) Data Security Standard (DSS), and the Health Insurance Portability and Accountability Act (HIPAA) could influence on the vulnerability management considerations. Hence, vulnerability management needs to be understood and carefully planned both operationally and legally. (Harris & Maymi 2019, 15; Foreman 2010, Chapter 1.)

Furthermore, as stated in ISO/IEC 27002:en (2017, 53) “a current and complete inventory of assets is a prerequisite for effective technical vulnerability management.” The software vendor, version numbers, current state of deployment, and the person(s) responsible for the software are examples of the specific information needed to support technical vulnerability management.

5.1.1 Common Vulnerabilities and Exposures (CVE®)

According to Foreman (2010, Chapter 4) “the main purpose of the CVE is to provide a cross-platform standard for identification of vulnerabilities.” CVE was launched in September 1999 by a non-profit organisation, the MITRE corporation. Nowadays, CVE is the industry standard for vulnerability and exposure identifiers, providing reference points for data exchange so that cybersecurity products and services can speak with each other. Industry-endorsed by the CVE numbering authorities (CNAs), CVE board, and numerous organisations include CVE entries in their products, services, vendor alerts, and security advisories. (CVE 2019a.)

CVE provides a standardised identifier for a given vulnerability or exposure. CVE Entry (Figure 9) is comprised of an identification number, a description, and at least one public reference. By knowing the common identifier, it is possible to accurately access information about the issue across multiple information sources as CVE is designed to allow vulnerability databases and other capabilities to be linked together and to facilitate the comparison of security tools and services. (CVE 2019a.)

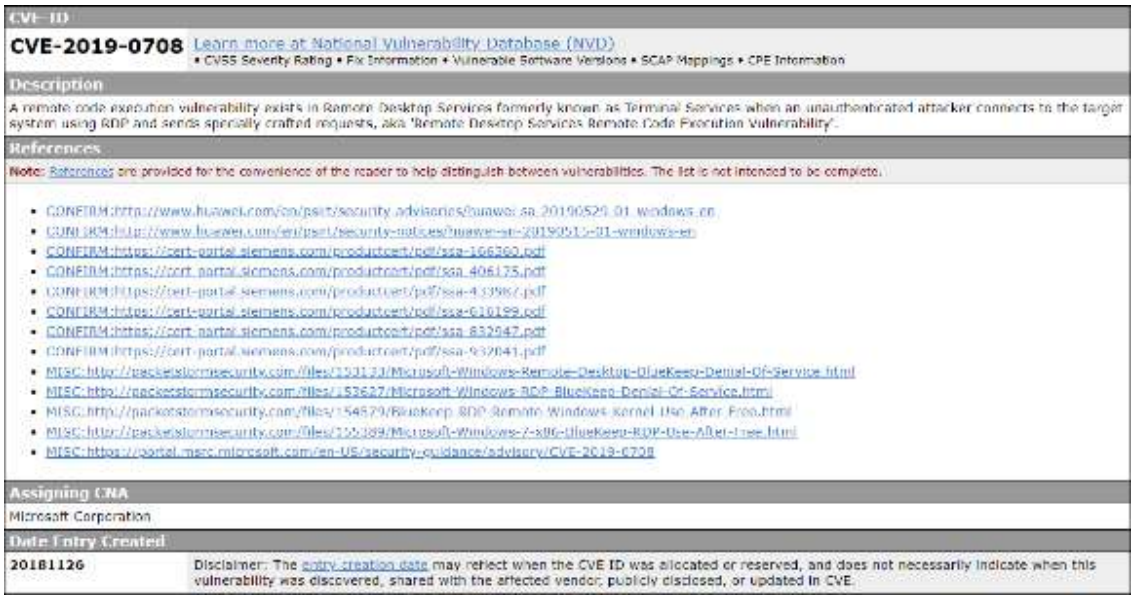

Figure 9. CVE Entry example (MITRE 2019b)

National Vulnerability Database (NVD) is a vulnerability database built upon and fully synchronised with the CVE list. As enhanced information such as risk, impact, fix information, or detailed technical information are not provided in the CVE entry, this information can be obtained from the NVD. Any updates to CVE appear immediately in NVD. (CVE 2019c.)

5.1.2 National Vulnerability Database (NVD)

NVD is a product of the National Institute of Standards and Technology (NIST). It was originally created in 2000 and at first, called the Internet – Categorization of Attacks Toolkit (ICAT). The NVD is the U.S. government repository of standards-based vulnerability management data: databases of security checklist references, security-related software flaws, misconfigurations, product names, and impact metrics. It provides advanced searching features such as by OS; vendor name, product name, and/or version number; vulnerability type, severity, related exploit range, and impact. The data is represented by using the Security Content Automation Protocol (SCAP) which is a set of standards designed to support automation of vulnerability management, compliance management, and other security functions. (NVD n.d.)

CVEs published on the CVE list are analysed by the NVD by aggregating data points from the description, references supplied, and any supplemental data publicly available at the time. Association impact metrics, vulnerability types, and applicability statements as well as other pertinent metadata are the results of the analysis. (NVD n.d.) The association impact metrics are presented by using CVSS. Vulnerability types are presented by using Common Weakness Enumeration (CWE™), which is a list of common software and hardware weakness types that have security repercussions. CWE can be used to prevent security vulnerabilities that have plagued the software and hardware industries by eliminating the most common mistakes in software and hardware deliveries. (CWE 2020.) Common Platform Enumeration (CPE) is the basis for applicability statements. It is a standardised method of describing and identifying classes of application, operating systems, and hardware devices. CPE identifiers are used to indicate systems or components subject to a particular vulnerability. (Foreman 2010, Chapter 4.)

5.1.3 Common Vulnerability Scoring System (CVSS)

Forum of Incident Response and Security Teams – FIRST (FIRST.org 2019) states that while evaluating a vulnerability, it is imperative to understand the impact a vulnerability poses to the organisation. To provide a standard framework for assessing the impact of a vulnerability and its principal characteristics, FIRST.org released the Common Vulnerability Scoring System (CVSS) in 2004. Capturing the principal vulnerability characteristic and producing a numerical score reflecting its severity can help organisations assess and prioritise their vulnerability management processes. After the initial release, CVSS has enjoyed widespread adoption and has evolved from version 1.0 to 3.1. NIST included CVSS v2.0 as part of its SCAP in 2007. In September 2007, CVSS v2.0 was adopted as part of the PCI DSS. Here, to comply with PCI DSS, merchants processing credit cards must demonstrate that none of their computing systems has a vulnerability with a CVSS score greater than or equal to 4.0. CVSS v3.0 was formally adopted as an international standard for scoring vulnerabilities (ITU-T X.1521) in March 2016. The current version 3.1 was released in June 2019 focusing on clarifying and improving the existing standard without introducing

new metrics or metric values, and without making major changes to existing formulas. CVSS v4.0 is a work in progress and is expected to incorporate larger changes to the scoring system, such as the addition of completely new metrics. (FIRST.org 2019.)

As shown in Figure 10, CVSS is composed of three metric groups, each consisting of its own set of metrics. The base metric group represents the intrinsic characteristics of a vulnerability. These characteristics are constant over time and across organisational environments. The characteristics of a vulnerability that may change over time but not across organisational environments are presented by the temporal metric group. The characteristics of a vulnerability, relevant and unique to a particular organisational environment, are represented by the environmental metric group. (FIRST.org 2019.)

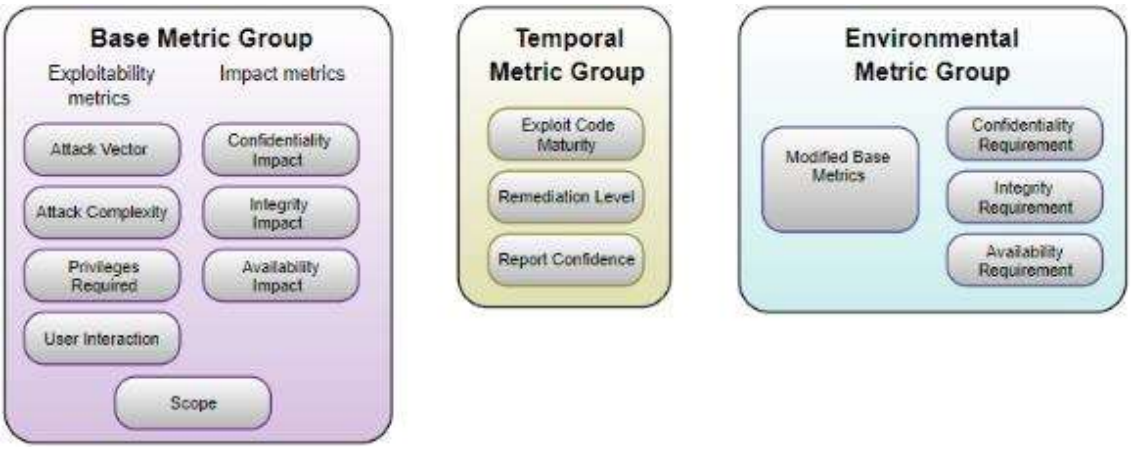

Figure 10. CVSS metric groups (FIRST.org 2019)

Base scores are usually produced by the organisation maintaining the vulnerable product, or a third party scoring on their behalf, and it is typical for only the base metrics to be published. After assigning values to the base metrics, the base equation computes a score ranging from 0.0 to 10.0. As Figure 11 illustrates, the base equation is derived from the exploitability sub-score equation and the impact sub-score equation. Scoring the temporal and environmental metrics is not required but it is recommended to more accurately reflect the relative severity posed by a vulnerability to an organisation’s environment. Generally, the base and temporal metrics are specified by vulnerability bulletin analysts, security product vendors, or application vendors. The environmental metrics are specified

by the end organisations as they possess the required information to assess the potential impact within their own computing environment. In addition to receiving a numerical CVSS score, a vector string is produced which is a textual representation of the metric values used to score the vulnerability. CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:N is an example of a produced vector string. (FIRST.org 2019.)

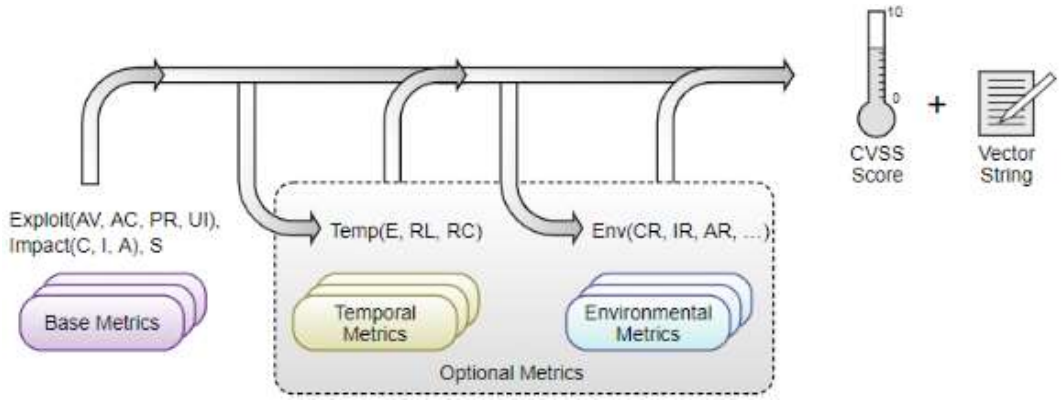

Figure 11. CVSS scoring (FIRST.org 2019)

The base metric is composed of two sets of metrics: the exploitability metrics which reflect the ease and technical means by which the vulnerability can be exploited, and the impact metrics which reflect the direct consequence of a successful exploit. Furthermore, the exploitability metrics represent characteristics of the vulnerable component when the impact metrics represent the consequence to the impacted component. Measuring whether the impact of a vulnerability in one vulnerable component impacts other resources in components beyond its security scope is captured by the scope metric which was introduced in CVSS v3.0. The current state of exploit techniques or code availability, the existence of patches or workarounds, or the confidence in the vulnerability description, are measured with the temporal metrics. The environmental metrics enable customised CVSS score presentation depending on the importance of the affected IT asset to the organisation. Detailed metric explanations are presented in Appendix 1. Scoring rubrics for the base metrics are presented in Appendix 2. (FIRST.org 2019.)

As already mentioned in Chapter 4, the risk is a potential for loss or damage if a threat exploits a vulnerability. CVSS score helps to describe the severity of an

issue. It helps to assess how quickly to react to the issue at hand. Hence, CVSS alone should not be used to assess risks posed to an organisation. Furthermore, the base score should be supplemented with a contextual analysis of the environment. A comprehensive risk assessment system should be employed that considers more factors outside the scope of CVSS, such as exposure and threat.

5.2 Vulnerability assessment

One must first have an inventory of assets to identify potential vulnerabilities and the associated risk. Thus, after mapping, categorising, and prioritising all assets, the focus shifts to continuous vulnerability assessment. Vulnerability assessment provides comprehensive knowledge on systems, services, and devices that can breach a network. Furthermore, providing a complete prioritised list of vulnerabilities requiring attention. The security risks on assets can be, for example, in the form of software vulnerabilities, missing patches, or configuration weaknesses (Haber & Hibbert 2018, Chapter 8).

According to Haber and Hibbert (2018, Introduction), “vulnerability assessment solutions test systems and network services such as NetBIOS, HTTP, FTP, DNS, POP3, SMTP, LDAP, RDP, registry, services, users and accounts, password vulnerabilities, publishing extensions, detection and audit wireless network, and much more to build a risk profile.” Furthermore, Harris and Maymi (2019, 876), state that vulnerability scanners provide capabilities to identify active hosts on the network, active and vulnerable services (ports) on hosts, applications and banner grabbing, operating systems, vulnerabilities associated with discovered operating systems and applications, and misconfigured settings. Vulnerability scanners can also be used to test for compliance with host applications’ usage/security policies and establish a foundation for penetration testing.

NIST Special Publication 800-53 discusses security and privacy controls for federal information systems and organisations. Security controls have been divided into twenty families and a two-character identifier uniquely identifies security control families. Security control family RA stands for risk assessment, and control number RA-5 refers to control named vulnerability monitoring and

scanning. The control states that organisations should monitor and scan for vulnerabilities in the system and hosted applications based on the organisation-defined frequency, and when new vulnerabilities potentially affecting the system are identified and reported; to employ vulnerability monitoring tools and techniques; to analyse vulnerability scan reports and results from vulnerability monitoring; to remediate legitimate vulnerabilities; to share information obtained from the vulnerability monitoring process and control assessments to help eliminate similar vulnerabilities in other systems, and; to employ vulnerability monitoring tool that includes the capability to readily update the vulnerabilities to be scanned. (Joint task force transformation initiative 2020, 8, 242.)

The third CIS Control, continuous vulnerability management, describes three sub-controls related to vulnerability scans. *Run automated vulnerability scanning tools* is a control that states that up-to-date Security Content Automation Protocol (SCAP) compliant scanning tool should be utilised. The tool should be used to automatically scan all systems on the network on a weekly or more frequent basis to identify all potential vulnerabilities on the organisation's systems. *Perform authenticated vulnerability scanning* is a control that states that authenticated vulnerability scanning with agents running locally on each system, or with remote scanners that are configured with elevated rights on the tested systems, should be performed. *Compare back-to-back vulnerability scans* is a control that states that the results from consecutive vulnerability scans should be regularly compared to verify that vulnerabilities have been remediated. (Center for Internet Security 2019, 16.)

Many commercial and open-source vulnerability scanning tools are available in the markets. As the purpose is not to compare different scanning tools and their capabilities, Nessus by Tenable Network Security and Nexpose Vulnerability Scanner by Rapid 7 are mentioned as examples of commercial scanning tools. Both tools also comply with Security Content Automation Protocol (SCAP). For example, The Open Web Application Security Project (OWASP) Foundation offers a comprehensive list of vulnerability scanning tools available in the market (The OWASP® Foundation n.d.).

5.3 A risk-based approach to vulnerability management

Vulnerability management should be considered as a holistic program designed to reduce overall risk (Williams 2019, 3). According to Gartner (Moore 2017), by the end of 2020, 99% of the vulnerabilities exploited will continue to be ones known by security and IT professionals at the time of the incident.

It is worth remembering that a vulnerability is only as bad as the threat exploiting it and the possible impact on the organisation. Vulnerabilities should be rated based on risk to improve the effectiveness of the organisation's vulnerability management program. Here, the focus from counting vulnerabilities should be sifted to managing risk. Furthermore, focusing on the issues that pose the greatest danger to the business. Williams (2019, 4-8) states that effective risk-based vulnerability management (RBVM) program includes the following four components:

- *Prioritising the application of patches* where the prioritisation should be on the most critical vulnerabilities. Here, the organisation must understand its network and possess a level of expert knowledge in vulnerabilities and exploit development to prioritise patching effectively.
- *Mapping security controls to assets* allows organisations to quickly understand their level of exposure when a new vulnerability is discovered.
- *Threat modelling to understand attack chains* can help the organisation to understand which systems identified in the scan report represent the highest risk. Thus, helping to plan patch efforts accordingly.
- *Applying gap analysis to prioritise new security controls*. Here, incorporating a risk-based perspective to vulnerability management allows organisations to identify consistent limitations in compensating controls that force the prioritisation of patches.

Kenna Security (2018a, 5-6) states that operationalising a risk-based approach to vulnerability management involves three steps; establishing meaningful *metrics*, *integrating risk management* into operational processes, and embracing opportunities to become *predictive*. While establishing the right metrics, the focus on risk should be both in terms of the likelihood of vulnerability exploitation and its potential business impact. Here, the progress in reducing risk across both attributes by factoring in associated assets should be tracked. Metrics can include, for example, the remediation rate of high-risk vulnerabilities and the number of these vulnerabilities in a specific environment, the median time to

remediate a high-risk issue, the median time to discover a high-risk issue, number of high-risk assets, and so worth. Integrating risk management into existing operational processes, tools, and workflows can create coherence between development teams, operations teams, and security, leading to focus on the right things. Finally, after integrating a risk-based approach to vulnerability management into core operations, the focus can be shifted from being proactive to being predictive. This could be done, for example, by utilising Exploit Prediction Scoring System (EPSS) into one's vulnerability remediation program.

5.4 Exploit Prediction Scoring System (EPSS)

The reality is that there are too many vulnerabilities to patch. An attempt can be made to patch all identified vulnerabilities to provide the greatest coverage of vulnerabilities patched, leading to an attempt to consume resources to fix vulnerabilities that pose a lower risk (Jacobs et al. 2019, 1). According to FIRST.org (n.d.a.), past research has shown that organisations can fix between 5% and 20% of known vulnerabilities per month. Also, the fact is that most vulnerabilities are never used to perform attacks. Prior studies suggest that 10-15% of all publicly known vulnerabilities have a known exploit written for them and only a small subset, 2-5% of CVE's, are exploited in the wild (Jacobs et al. 2019, 2).

Separating the signal from the noise is what matters and should be addressed. This is far from being an easy task to perform. Figure 12 presents the overlap between all CVEs rated as CVSS 7 and above. According to CVE details (n.d.), this would mean an overwhelming amount of 44,107 published vulnerabilities. Patching all CVSS 7+ CVEs would be a daunting task. Furthermore, to measure the quality of this approach, the vulnerabilities known to have been exploited in the wild should be tracked. By doing so, vulnerability remediation decisions can be divided into four categories (Figure 12). *True positives* are correctly prioritised vulnerabilities as these count as vulnerabilities exploited in the wild. *False positives* are incorrectly prioritised as these vulnerabilities were not exploited. Hence, counting as wasted resources. *False negatives* are incorrectly delayed as these vulnerabilities were not prioritised but were observed as exploited in the

wild. *True negatives* are correctly delayed as these vulnerabilities were not prioritised nor exploited. Furthermore, the efficiency of the remediation effort for CVSS 7+ is about 5-7% and remediation effort coverage is about 45-65%. (Jacobs & Roytman 2019, 15-17; FIRST.org n.d.a.)

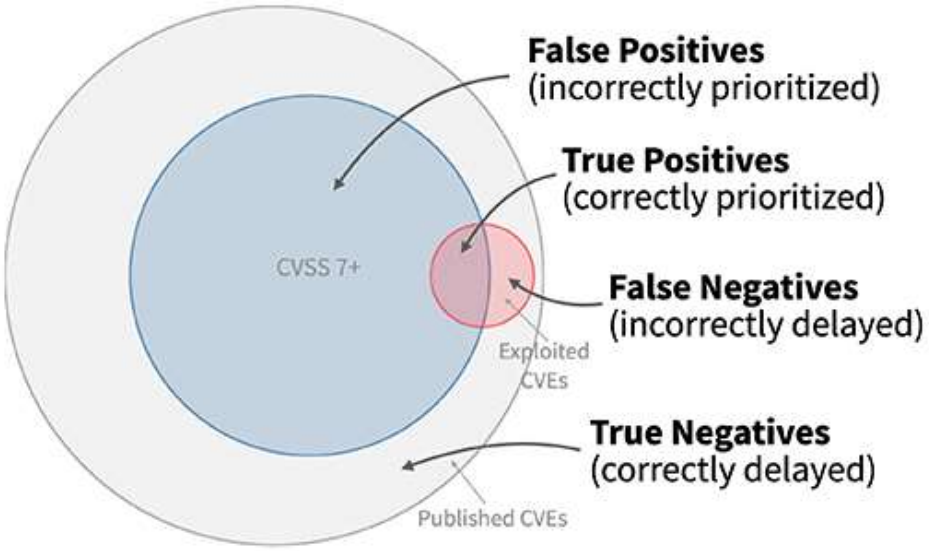

Figure 12. Vulnerability remediation decision categories (Jacobs & Roytman 2019)

Exploit Prediction Scoring System (EPSS) was firstly introduced at BlackHat 2019 conference. It is an open data-driven framework for assessing vulnerability threat. EPSS is used to predict the probability of vulnerability exploitation within the first twelve months after public disclosure. The model builds on previous research, Improving Vulnerability Remediation Through Better Exploit Prediction by Jacobs et al. (2019). Here, machine learning (ML) was used to create exploit prediction models. The research data relates to vulnerabilities published by MITRE's Common Vulnerability Enumeration (CVE) and NIST's National Vulnerability Database (NVD), in a two-year window between June 1, 2016, and June 1, 2018. Exploit code was extracted from Exploit DB. Weaponized exploits were found from Rapid 7's Metasploit framework, D2 Security's Elliot Framework, and the Canvas Exploitation Framework. Information on whether the vulnerability was exploited in the wild comes from closed sources, especially from Fortinet, Proofpoint, AlienVault, and Grey Noise. (Jacobs et al. 2019, 3-5.)

The estimating model is based on 16 specific variables. Seven related to the software vendor (VENDOR), including Microsoft, IBM, Adobe, HP, Apache,

Google, and Apple. Two related to exploit code (EXPLOIT), whether exploit code had been published and whether exploit code had already been weaponized. Six related to properties of the vulnerability and impact (TAG), including memory corruption, code execution, local, remote, and web. The final variable is a count of references (REF) in a published CVE. Each variable carries either a positive (more likely) or negative (less likely) weight towards predictability. The descriptive statistics of each variable formulate the estimating model to be as shown in Equation 1. (Jacobs et al. 2019, 9-10.)

$$\Pr[\text{exploitation}]_i = f(\alpha_0 + \alpha_v VENDOR_i + \alpha_I EXPLOIT_i + \alpha_\Sigma TAG_i + \alpha_R REF_i + \varepsilon_i) \quad (1)$$

Here, VENDOR is a vector of binary variables related to the most frequently exploited software vendors. EXPLOIT is a vector of binary variables related to the exploit code. TAG is a vector of variables related to the characteristics of the vulnerability. REF is the log value of one plus the count of vendor references of the published vulnerability (CVE). The random error term, ε, is assumed to be independent of the observed covariates. The regression results of estimating Equation 1 are presented in Table 1. (Jacobs et al. 2019, 9-10.)

Table 1. Regression results (Jacobs et al. 2019)

Variable	LogOdds (weight)	Standard error
VENDOR: Microsoft	2.44***	0.111
VENDOR: IBM	2.07***	0.138
EXPLOIT: Weaponized	2.00***	0.164
VENDOR: Adobe	1.91***	0.136
VENDOR: HP	1.62***	0.213
EXPLOIT: Published	1.50***	0.091
VENDOR: Apache	1.10***	0.231
REF: Count	1.01***	0.078
TAG: Code execution	0.57***	0.096
TAG: Remote	0.23**	0.089
TAG: Denial of service	0.22*	0.098
TAG: Web	0.06	0.091
TAG: Memory corruption	-0.20	0.126
TAG: Local	-0.63***	0.143
VENDOR: Google	-0.89**	0.280

VENDOR: Apple	-1.92***	0.399
(Intercept)	-6.18	0.143

*Significance of p-value: ***<0.001, **<0.01, *<0.05*

The regression results presented in Table 1, allows computation of log odds (Equation 2), which is a cumulative sum of the observations about a vulnerability multiplied by the coefficients from the model. Here, each variable on the right-hand side of Equation 2, are encoded as 1 or 0 depending on if the attribute is present (1) or not (0) in the vulnerability. The reference count is an exception as being a continuous variable transformed by adding one and taking the log. (Jacobs et al. 2019, 19.)

$$\begin{aligned}\text{LogOdds} = -6.18 &+ \\ 2.44 &* \text{vend:microsoft} + \\ 2.07 &* \text{vend:ibm} + \\ 2.00 &* \text{exp:weaponized} + \\ 1.91 &* \text{vend:adobe} + \\ 1.62 &* \text{vend:hp} + \\ 1.50 &* \text{exp:poc code} + \\ 1.10 &* \text{vend:apache} + \\ 1.01 &* \text{log(ref:count + 1)} + \\ 0.57 &* \text{tag:code execution} + \\ 0.23 &* \text{tag:remote} + \\ 0.22 &* \text{tag:denial of service} + \\ 0.06 &* \text{tag:web} + \\ -0.20 &* \text{tag:memory corruption} + \\ -0.63 &* \text{tag:local} + \\ -0.89 &* \text{vend:google} + \\ -1.92 &* \text{vend:apple}\end{aligned} \tag{2}$$

Finally, the LogOdds value is converted into the estimated probability exploitation by using Equation 3.

$$\Pr[\text{exploitation}] = 1/(1+e^{-\text{LogOdds}}) \tag{3}$$

As already stated, EPSS is an open data-driven framework for assessing vulnerability threats. Designed for easy implementation. EPSS can be used, for example, in a spreadsheet by using the above-mentioned data and equations.

Another option is to use the Exploit Prediction Scoring System calculator (Figure 13), provided by Kenna Security (n.d.), to support one's vulnerability remediation actions.

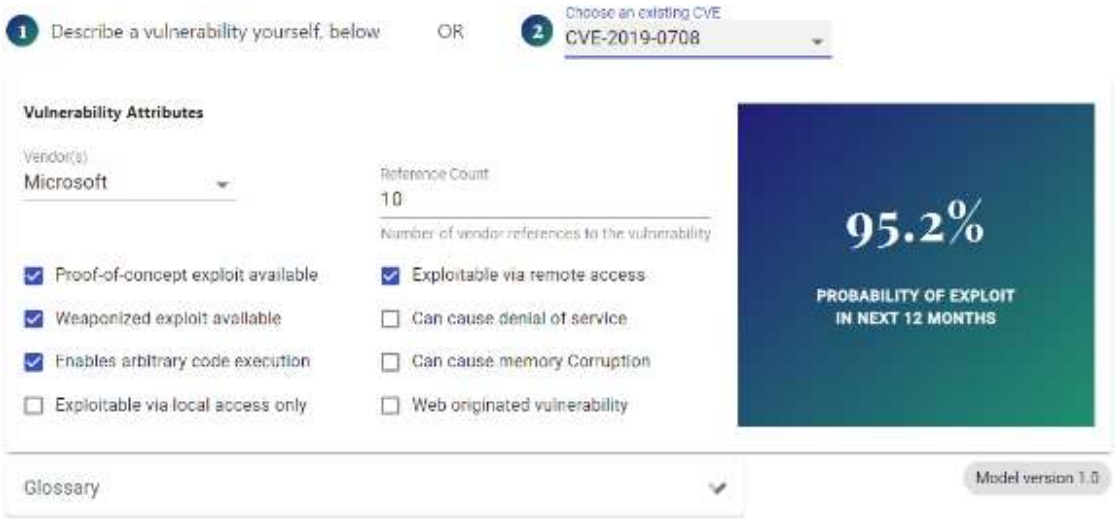

Figure 13. Exploit Prediction Scoring System calculator (Kenna Security n.d.)

The CVE example, CVE-2019-0708, used in the calculator is a Remote Desktop Services Remote Code Execution Vulnerability, also known as BlueKeep (Microsoft 2019a). The estimating model predicts an approximately 95% probability of this vulnerability being exploited within 12 months of being published. At this point, it can be stated that the prediction was a correct one.

6 SECURITY PATCH MANAGEMENT

Nicastro (2011, Chapter 5), states that "security management is the policies, processes, procedures, and technologies instituted to protect the confidentiality, integrity, and availability of the IT infrastructure, which includes all of the organization's assets." The defined level of security is established and maintained through security operations. These include administration, maintenance, and operations of the security measures implemented to support the security management processes. Strategic, tactical, and operational are the three functional areas of security management. Firstly, the corporate security policy is established within the strategic security management. Secondly, security engineering tasks, for example, determining whether security technology should be implemented, are performed within the tactical security management. Thirdly, carrying out day-to-day tasks of administering, maintaining, and operating the

security measures are done within operational security management, also known as security operations. (Nicastro 2011, Chapter 5.)

Security patch management is an output of the security management process (Figure 14). The stand on patch management is defined through the strategic aspect of security management. The strategic group is responsible for patch management policy establishment. The tactical aspect is responsible for patch management process establishment. The implementation aspects, like patch management process facilitation and patch deployment, are performed within the operational security management. (Nicastro 2011, Chapter 5.)

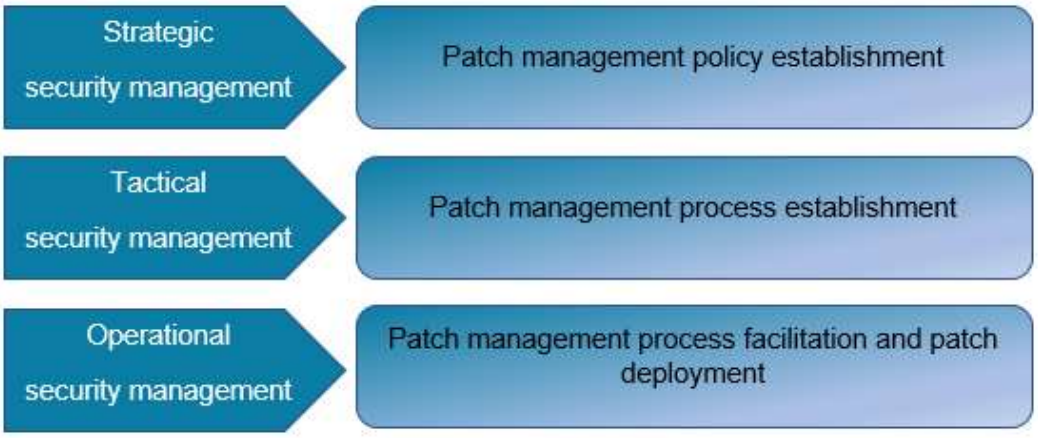

Figure 14. Relation between security patch management and security management process

Security patch management is a remediation plan within the vulnerability management process. While vulnerability management covers a broad spectrum of tasks and activities, patch management deals with applying patches to vulnerable systems. To embed security patch management into a vulnerability management program, one should consider the implementation of asset management establishment, vulnerability prioritisation, vulnerability remediation to reduce risk, measuring the success of vulnerability management program, and development of partnership and support. (Nicastro 2011, Chapter 6.)

Patch management is a subset of change management, configuration management, and release management processes. The change management process assures that standard methods and procedures are used to handle the lifecycle of all changes. It ensures that all changes to service assets and

configuration items are recorded in the configuration management system. Thus, reducing the risks and disruption to the business. (Axelos 2011, 12.) Johnson et al. (2011, 4) explain that “configuration management comprises a collection of activities focused on establishing and maintaining the integrity of products and systems, through control of the processes for initialising, changing, and monitoring the configurations of those products and systems.” Finally, the release management process is responsible for planning, scheduling, and controlling the build. It aims to protect the integrity of existing services. (Axelos 2011, 47.)

6.1 The concept of patch management

According to Souppaya and Scarfone (2013, 2), patch management is “the process for identifying, acquiring, installing, and verifying patches for products and systems.” The purpose of patches is to correct security and functionality problems in software and firmware which, if left unpatched, could put an information system at risk of exploitation. Furthermore, by not eliminating these software flaws, the attack surface is cumulatively increased with every emerging vulnerability. (Souppaya & Scarfone 2013, 2.) Hence, it is important to recognise patch management as one element of a multilayered defence system used to protect critical assets. It is a technical control type used to reduce the risk an organisation faces. (Harris & Maymi 2019, 8.) Patching and security vulnerability management go hand in hand as the goal of vulnerability management is to minimize the risk introduced by vulnerabilities. Patching activity is a step of managing security vulnerabilities and thus a means to an end of minimising these risks. (Williams 2019, 3.)

Various security compliance frameworks, mandates, and other policies place patch management requirements for organisations. In NIST Special Publication 800-53, security control family SI stands for system and information integrity. SI-2 flaw remediation includes installing security-relevant software and firmware patches, testing patches before installation, and incorporating patches into configuration management processes. (Joint task force transformation initiative 2020, 333.) PCI DSS requirement 6: develop and maintain secure systems and applications states that all system components and software should be protected

from known vulnerabilities by installing applicable vendor-supplied security patches. Furthermore, critical security patches should be installed within one month of release. (PCI security standards council 2018, 54.)

6.2 Patch management process and best practices

Nicastro (2011, Chapter 9) states that “a patch management process is a best practice that should be employed in any organisation, regardless of size, to govern how to respond to security-related vulnerabilities.” Furthermore, according to Nicastro (2011, Chapter 1), the goals behind implementing a patch management process are many:

- positioning patch management process within larger problem space, vulnerability management
- improving the protection from current vulnerabilities and the threat of exploitation before a patch is deployed
- improving the dissemination of information to the stakeholders
- record keeping formalisation in the form of tracking and reporting
- introducing automated discipline once a process is in place
- allowing faster remediation rate and effective prioritisation to release security vulnerabilities with a reduced number of resources
- improving accountability for the roles responsible for security and systems

Figure 15 displays a high-level walkthrough by describing the patch management process with seven key activities. Firstly, the analysis phase includes the monitoring and impact assessment activities. Here, security intelligence sources are monitored for security vulnerabilities, and impact assessment is performed on new security vulnerability findings. Secondly, the remediation phase includes developing and testing activities. Here, the emphasis lay on an action plan development used to address the patch and mitigate the risks within the organisation. Testing activities are also included in this phase. Thirdly, updating the operational environment phase includes implementation, documentation, and integration activities. Here, the technical remediation strategy is implemented on all affected hosts, the vulnerability life cycle is documented, and the patch or configuration changes are integrated into a related application or system baseline and standard build. Lastly, the tracking patches phase includes status reporting activities. Here, the importance is on ensuring that all vulnerable systems have

been patched appropriately. It is also worth mentioning that all the activities are subject to the status reporting requirements. (Nicastro 2011, Chapter 9.)

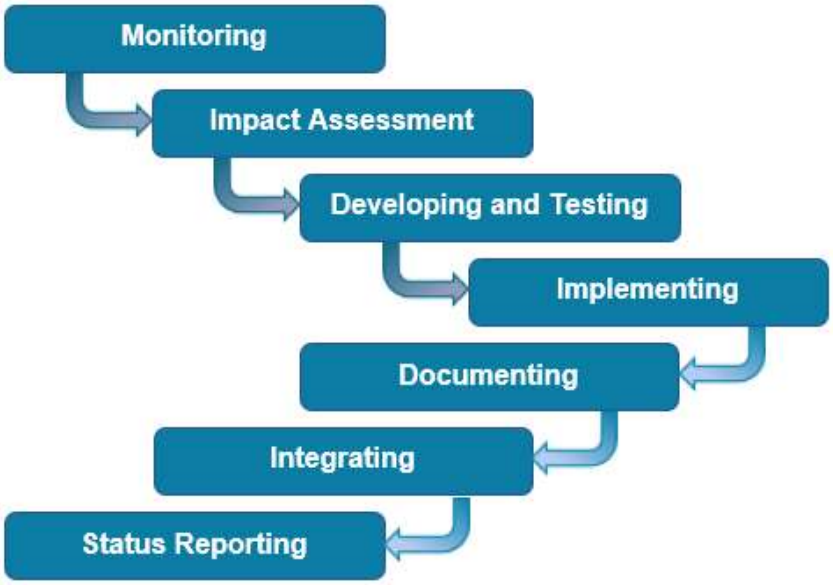

Figure 15. High-level patch management flow (Nicastro 2011)

Patches are published, for example, by OS vendors, application vendors and network equipment vendors. Hence, having even a decentralised/unmanaged patching model in place is better than having none. However, centralised patch management for security operations can be considered a best practice. (Harris & Maymi 2019, 994-995.) It is vital to keep in mind that patch management is a process, not a technology. Thus, the technology aspect should not be the driving force when considering patch management best practices. Technology is there to support different phases of the process. In small companies, where the number of endpoints is low, manual patching can be an option, but as the number of patchable endpoints increases, automatic patching comes into question.

As stated in Chapter 5.1, continuous vulnerability management is the third CIS Control. Here, two sub-controls *deploy automated operating system patch management tools* and *deploy automated software patch management tools* can be considered as patch management best practices. Deploying an automated operating system and software patch management tools ensures that the operating systems and third-party software on all systems are running the most recent security updates provided by the software vendors. (Center for Internet Security 2019, 16.)

6.3 Patch prioritisation

As stated by European Union Agency for Cybersecurity (2019, 9), “the significance of risks increases as vulnerabilities trigger the creation of the associated exploits and decrease when the patches become available.” The number of devices in use, operating systems, and software combined with the complexity of today’s interconnected software systems and their countless configurations, creates a scenario where many organisations are unable to patch all vulnerabilities discovered by vulnerability scans. (Williams 2019, 5.) Thus, organisations should not try to patch everything but instead shift the focus on exploitable vulnerabilities (Panetta 2020).

According to Kenna Security (2018b, 2-4), there are four key factors that security organisations should consider when identifying and prioritising vulnerabilities. Vulnerabilities that allow *remote code execution* as it enables attackers to access a computing device anywhere in the world. Vulnerabilities that have *an exploit published in a widely used toolkit,* for example, Metasploit or Cobalt Strike, should be at the top of the list of vulnerabilities to patch or mitigate. Vulnerabilities that have *network accessibility* should not be left unnoticed when determining the severity of a threat and the likelihood of exploitation. Vulnerabilities included in the *exploit database* are more likely to emerge as a significant, broad-based threat.

Williams (2019, 5) presents the following patch prioritisation criteria to consider while prioritising patches:

- criticality of the data processed on the vulnerable asset
- possibility to use the vulnerable asset to pivot to sensitive data
- presence of compensating controls to prevent exploitation or mitigate an attack
- vulnerability present in a default configuration
- vulnerability actively exploited in the wild
- exploit for the vulnerability publicly available
- exploit for the vulnerability available for private sale
- the difficulty of exploiting the vulnerability
- vulnerability used to exploit other organisations in the same vertical

Furthermore, Nicastro (2011, Chapter 8) describes an expanded patch priority list that categorises vulnerabilities based on four priorities:

- *Critical priority*: Vulnerabilities for applications or operating systems or Internet-facing hosts; vulnerabilities that will allow self-propagation without user interaction; vulnerabilities that may allow for internal or external compromise of critical hosts; a vulnerability that may allow for widespread impact with an available exploit in the wild.
- *Urgent priority*: Vulnerabilities that may allow for exploiting weaknesses in commonly used and necessary network services but that is normally blocked by a firewall without adversely affecting end-users; widespread application or operating system vulnerability that requires user interaction to initiate; a vulnerability that may allow for widespread impact but for which no exploit currently exists.
- *Low priority*: a vulnerability that may affect a minority of hosts or applications, and exploit success is low or does not currently exist; the impact of the vulnerability is low to medium, due to other mitigating factors that decrease the likelihood of exploitation.
- *Maintenance*: required to fix an uncommonly used service; the impact of the vulnerability would have minimum effect on host or application functionality; upgrades to application or operating system functionality that are not essential to usability.

The ability to consider the mentioned prioritisation criteria and patch priority list, rely on having a solid inventory of software, assets, data, and compensating controls present on the network. Furthermore, the organisation must possess a level of expert knowledge in vulnerabilities and exploit development. (Williams 2019, 5.)

The patch prioritisation process should be established to evaluate acceptable timeframes for testing and deploying patches. The process presence is vital to keep up with the increase in patch release frequencies and the decrease in time before exploits are available. The release schedule based on security priority could be, for example, *critical priority* vulnerability is recommended to be patched within 48 hours or two weeks at the latest. *Urgent priority* vulnerability within two weeks or four weeks at the latest. *Low priority* vulnerability between one to two months. *Maintenance priority* vulnerability on a regularly scheduled basis, for example, quarterly or when a service pack or update rollup becomes available. (Nicastro 2011, Chapter 8.)

6.4 Patching Microsoft Windows Server operating system

Patch management has existed from the day the first operating system was released. Previously, till March 2017, Microsoft published information about vulnerabilities and security updates on the Microsoft Security Bulletin website. Security bulletin website included security bulletins addressing security vulnerabilities, describing their remediation, and providing links to the applicable updates. Security bulletin ID number, for example, MS17-010, was used as a pivot point. Security Update Guide replaced the Bulletin websites in April 2017. Security Update Guide pivots on vulnerability ID numbers (CVE), for example, CVE-2019-0708, and KB article ID, for example, KB4499180. Figure 16 illustrates the change from bulleting websites to Security Update Guide. (Kakiuchi 2017.) In September 2020, Microsoft published an update to the Security Update Guide to provide a more intuitive user experience, filtering, and customisation abilities (Microsoft 2020).

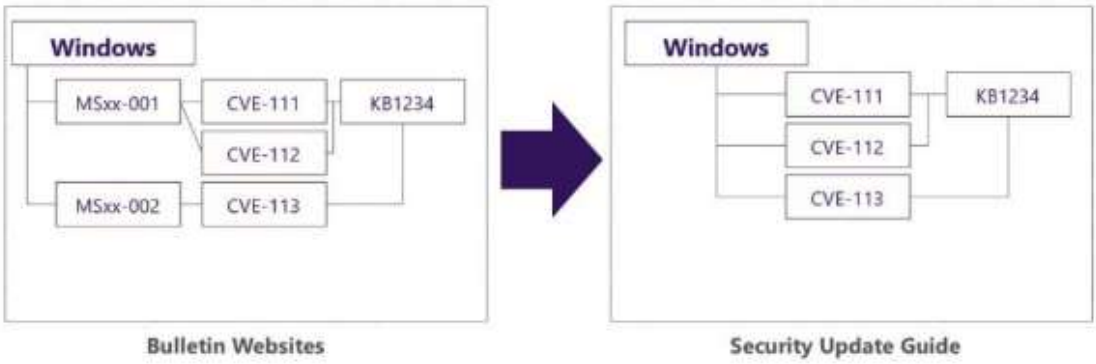

Figure 16. Change from Bulletin Websites to Security Update Guide (Kakiuchi 2017)

Since October 2003, Microsoft has scheduled the release of security updates on the second Tuesday of each month at 10 a.m. Pacific Standard Time (PST). Update Tuesday (“B” release), unofficially referred to as Patch Tuesday, includes both security and non-security fixes. These are the primary and most important updates of all the monthly update events. Microsoft can also publish out-of-band (OOB) security updates or quality fixes at any time through the month based on the urgency. Microsoft recommends a monthly patching schedule but with OOB patches recommendation is to install the patches without delay. There are also “C” and “D” releases that occur during the third and fourth weeks of the month. These are preview releases containing only nonsecurity updates. Preview

updates are intended to provide visibility and testing ability to the planned nonsecurity fixes targeted for the next month's Update Tuesday ("B") release. (Arban 2017; Wilcox 2018; Branscombe 2019.)

Furthermore, one should take time while choosing which server edition is chosen. Opting for the Windows Server Core version reduces the potential attack surface as the Desktop Experience with user interface elements and graphical management tools are not included (Microsoft 2019b). Thus, reducing the number of required patches. Fewer patches equal less downtime. One should also consider avoiding multi-purpose servers as the attack surface increases with each installed service and application.

6.4.1 Microsoft security update severity rating system

Microsoft (n.d.a.) states that attacks resulting from the exploitation of previously unknown vulnerabilities are rare. Instead, vulnerabilities that have a patch released but not applied are the ones being exploited. The severity of vulnerabilities varies, and the risk associated with each patchable vulnerability should be understood. Thus, Microsoft has published a severity rating system (Table 2) that rates each vulnerability according to the worst theoretical outcome if a vulnerability would be exploited.

Table 2. Microsoft security update severity rating system (Microsoft n.d.a.)

Rating	Description
Critical	A vulnerability whose exploitation could allow code execution ***without*** user interaction. These scenarios include self-propagating malware (e.g. network worms), or unavoidable common use scenarios where code execution occurs without warnings or prompts. This could mean browsing a web page or opening an email. Microsoft recommends that customers apply Critical updates immediately.
Important	A vulnerability whose exploitation could result in compromise of the confidentiality, integrity, or availability of user data, or of the integrity or availability of processing resources. These scenarios include common use scenarios where a client is compromised ***with*** warnings or prompts regardless of the prompt's provenance, quality, or usability. Sequences of user actions that do not generate prompts or warnings are also covered. Microsoft recommends that customers apply Important updates at the earliest opportunity.

Moderate	The impact of the vulnerability is mitigated to a significant degree by factors such as authentication requirements or applicability only to non-default configurations. Microsoft recommends that customers consider applying the security update.
Low	The impact of the vulnerability is comprehensively mitigated by the characteristics of the affected component. Microsoft recommends that customers evaluate whether to apply the security update to the affected systems.

Furthermore, Microsoft evaluates the potential exploitability of each vulnerability associated with a Microsoft security update. The exploitability information is published as part of the monthly Microsoft security update details. The purpose of the exploitability index is to help in evaluating the risk of a vulnerability. The exploitability index uses one of four values (0-3) to communicate the likelihood of a vulnerability being exploited. The values are as follows: 0 – Exploitation detected; 1 – Exploitation more likely; 2 – Exploitation less likely; and 3 – Exploitation unlikely. (Microsoft n.d.b.)

Figure 17 illustrates a security update guide example of a vulnerability with a critical rating. CVE-2019-0708, Remote Desktop (RDP) Services Remote Code Execution Vulnerability, also known as BlueKeep, is a security vulnerability where an unauthenticated attacker connects to the target system using RDP and sends specially crafted requests. This security vulnerability was fixed with a windows security patch by correcting how Remote Desktop Services handles connection requests. (Microsoft 2019a.)

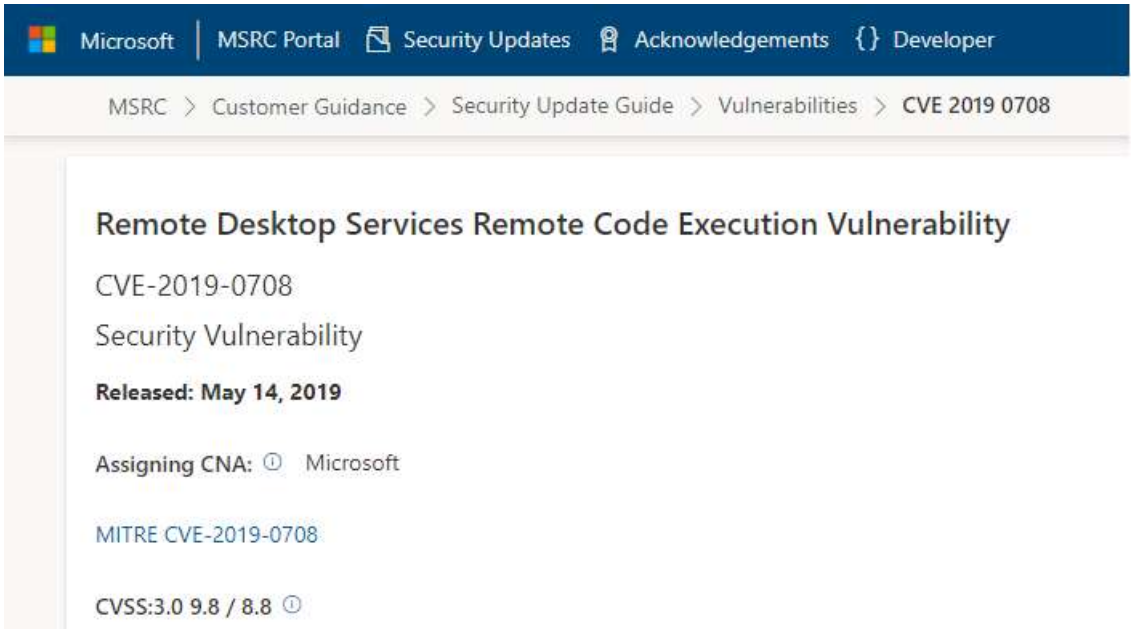

Figure 17. Vulnerability example CVE-2019-0708 (Microsoft 2019)

CVE-2019-0708 is a good example of a security vulnerability where a three-phased approach could be initiated. Firstly, the vulnerability could be mitigated by disabling Remote Desktop Services if they are not required. Secondly, workarounds could be used to mitigate the risk by enabling Network Level Authentication (NLA) or by blocking TCP port 3389 at the enterprise perimeter firewall. Thirdly, by removing the underlying vulnerability with a security patch. (Microsoft 2019a.)

6.4.2 Security updates for Windows Server

A security update is a widely released fix for security-related vulnerabilities which are rated by their severity. Since October 2016, Microsoft shifted from publishing individual patches to a rollup model for Windows Server 2008 R2 SP1 (end of support), Windows Server 2012, and Windows Server 2012 R2. The rollup model precludes the possibility of rollback an individual patch in case of compatibility issues. Instead, rollback can only be performed to the previous month's cumulative update. Table 3 explains the difference between the different cumulative updates applying to the mentioned Windows Server operating systems. (Cheng 2016.)

Table 3. Security update definitions (Cheng 2016)

Update name	Details
Security-only Quality update (Security-only update)	Release date: 2nd Tuesday of the month. New security fixes for the month. Available from Windows Server Update Services (WSUS) and Windows Update Catalog.
Security Monthly Quality Rollup (Monthly Rollup)	Release date: 2nd Tuesday of the month. New security fixes for the month, nonsecurity fixes from the latest Preview Rollup and fixes from all previous monthly rollups. Available from Windows Update (WU), Windows Server Update Services (WSUS) and Windows Update Catalog.
Preview of Monthly Quality Rollup (Preview of Monthly rollup)	Release date: 3rd Tuesday of the month. Nonsecurity updates for the month and all previous monthly rollups. Available from Windows Update (WU), Windows Server Update Services (WSUS) and Windows Update Catalog.

Windows Server 2016, the successor to Windows Server 2012 R2, includes several new features to its core foundation. At the same time, bringing some changes to the monthly cumulative update model. According to Microsoft (Christensen 2017), the changes in Windows Server 2016 simplify and streamline patching. This is achieved with update consolidation, predictable cadence, and proactive patch discovery. With Windows Server 2016, applying also to Windows Server 2019, a single cumulative update package is released during Update Tuesday (“B” release). The cumulative package includes all previous security and nonsecurity fixes. Thus, removing the possibility to install security-only updates. (Christensen 2017.)

While considering Windows Server operating system patching, installation of servicing stack updates (SSU) should be kept in mind. The servicing stack is the code that installs other operating system updates and contains the component-based servicing stack (CBS) which is a key component for several elements of Windows deployment such as DISM and SFC. The reliability of the update process is improved with SSUs. Microsoft has categorized SSUs as security updates and having a critical severity rating. This approach is an attempt to ensure that the latest SSU is applied. (Microsoft 2020.)

7 RESEARCH DATA

In this thesis, the semi-structured interviews and document reviews acted as the prime data gathering methods. However, as the researcher works for the commissioner, the researcher was partly interconnected to the studied phenomenon and research problem. At times, the researcher was involved with cases that directly concerned security patch management. Thus, providing the possibility to observe the phenomenon happening in the present operational environment. The following chapters discuss the research phase of the thesis process.

7.1 Interviewee selection

The direct commissioner of this thesis is a unit offering IT outsourcing and infrastructure services in Finland. Thus, defining the interviewee selection to this specific business unit. However, to understand the present state of the studied phenomenon, interviewing specialists from different organisational groups was required.

The interview topics focused on vulnerability and patch level reporting, asset and configuration management, operational Windows Server security path management, application discovery and dependency mapping, and change management. These topics were identified as vital to be able to reflect the theory to the present state of the operational environment. Also, the documents selected for a review guided the interviewee selection.

Thus, based on these conditions, seven specialists (n=7) were selected as interviewees. As the commissioner is not identified in this report, neither are the identities of the interviewees, nor their work history or titles. However, all interviewees have multiple years of work experience in IT outsourcing and infrastructure services. Furthermore, the chosen interviewees were technical specialists directly working with the interview topics. For referral purposes, the following describes a label for the interviewees and the topic division between them:

- Intvw1 and Intvw2: Application discovery and dependency mapping.
- Intvw3 and Intvw4: Vulnerability and patch level reporting.
- Intvw5: Operational Windows Server security patch management.
- Intvw6: Asset and configuration management.
- Intvw7: Change Management.

7.2 Semi-structured interviews

In structured interviews, the possibility of the interviewer influencing the research results is present (Chapter 3.2). Thus, the semi-structured interview method was chosen to achieve a more open and truthful discussion with the interviewees. Therefore, strictly lead interviews with predefined questions were not used. This decision was also supported by the fact that the chosen interviewees were technical specialists on the topics. Thus, the researcher did not possess adequate knowledge to opt for predefined questions. Also, as the interviewees represented different organisational groups, utilising the same question series did not suit the purpose.

The interviews (n=8) took place during January and February 2020. The interviewee working with operational Windows Server security patch management was interviewed twice during this time frame. The following describes the themes discussed in the interviews:

1. Application discovery and dependency mapping:
    - Visibility on assets and configuration items.
    - Visibility on services and applications.
    - Issues in the present state.
2. Vulnerability and patch level reporting:
    - Present state of Windows OS vulnerability reporting.
    - Present state of Windows OS patch level reporting.
    - Visibility and accessibility of reports.
    - Ability to correlate vulnerabilities to existing patch levels.
    - Issues in the present state.
3. Operational Windows Server security patch management:
    - Patching tools.
    - Patch levels.
    - Day-to-day operations.
    - Issues in the present state.
4. Asset and configuration management:
    - Used asset and configuration management system.
    - Visibility on assets and configuration items.
    - Visibility on services and applications.

- Visibility on patch levels.
- Information flow from other systems.
- Issues in the present state.

5. Change Management:
   - Stand on vulnerability management and security patches.
   - Issues in the present state.

Live interviews were in question. The interviews were not recorded; interview and observation notes were used for data recording. The average duration of an interview was 60 minutes. The emphasis was on understanding the present state and issues in the target environment. Stated issues had an important role in answering the research question *what development suggestions do arise from the findings*.

7.3 Document reviews

Document reviews were carried out on four documents (n=4). Documents included a global patch and vulnerability management standard, local patch office description, service description for server management services, and patch management section from one client agreement. For referring purposes, the documents are labelled as follows:

- Doc1: Global patch and vulnerability management standard.
- Doc2: Local patch office description.
- Doc3: Service description for server management services.
- Doc4: Patch management section from a client agreement.

Doc1 and Doc2 were chosen for a review to reflect obtained theoretical knowledge on vulnerability management and security patch management against these documents. Doc3 and Doc4 were explicitly chosen to answer the research question; *how are the security patching duties divided between the service provider and their client*.

8 RESEARCH RESULTS AND CONCLUSIONS

The research problem was formulated and presented as *the role of security patch management in vulnerability management requires more understanding and development*. The theoretical aspect was highly present while trying to understand the studied case. Thus, the stated research problem was such that

the answers relied heavily on existing theoretical knowledge. Especially the ability to answer the research questions *what is security patch management, and what is its role in vulnerability management* and *how do risk management, vulnerability management, and security patch management intertwine* required theoretical examination of the phenomena. An in-depth theoretical understanding enabled the possibility to formulate generic answers to the research questions. Thus, providing the ability to reflect the gained information towards the target environment.

Two research questions focused solely on examining the present state of the target environment; *how are the security patching duties divided between the service provider and their client*, and *what development suggestions do arise from the findings*. The following highlights the main findings and development suggestions:

- Risk management, vulnerability management, and security patch management are all information security management system (ISMS) controls.
- Having a risk-based approach to vulnerability management is strongly present. Therefore, the focus should also shift towards a risk-based security patch management strategy.
- Evaluating the implementation of EPSS to support vulnerability remediation prioritisation decisions.
- Stressing the importance of security patch management and emphasising automatic patching methods to the clients should be a constantly occurring practice.
- Evaluating including a production review plan into operational processes to ensure CMDB data accuracy.
- Examining the presence and requirement for service asset and configuration management process and release management process descriptions.
- Evaluating the client contracts against the global patch and vulnerability management standard.
- Establishing the ability for configuration item patch prioritisation.
- Evaluating the establishment of a virtual patch management team.

The following chapters discuss each research question in more detail.

8.1 The role of security patch management in vulnerability management

In a perfect digital world, vulnerabilities would not exist. Thus, making security patch management obsolete. The reality is quite the opposite. The increasing

volume of publicly disclosed vulnerabilities steers towards an approach where patching each vulnerability is not an option. Furthermore, as stated, the typical exploits occur on vulnerabilities that already have a patch released. For example, WannaCry and NotPetya attacks utilised the EternalBlue exploit. While the exploits occurred, Microsoft had already published a security patch to address the vulnerability in their Server Message Block (SMB) protocol implementation, CVE-2017-0144 (Chapter 5.1). Therefore, confirming the research question of *what is security patch management, and what is its role in vulnerability management* relevant.

Based on theoretical findings, one should not make the mistake of considering security patch management and vulnerability management as the same. These processes have a compatible relationship, and both are needed to improve the overall security. The short answer to *what is security patch management* is that it is “the process for identifying, acquiring, installing, and verifying patches for products and systems.” (Souppaya & Scarfone 2013, 2). Furthermore, *what is its role in vulnerability management* can be answered by stating that security patch management is one vital part of a comprehensive vulnerability management program by being a remediation plan within the vulnerability management process.

As mentioned above, the industry has witnessed the havoc caused by cyber-attacks. Nevertheless, the importance of security patch management has not been fully concretised. Supported by the theoretical findings, one significant factor could be the pure volumes of publicly disclosed vulnerabilities. Here, the organisations might concentrate purely on the amounts, not on the vulnerabilities that matter. Thus, also exhausting their resources while doing so. The finding supports the fact that the focus should shift towards a risk-based security patch management strategy.

8.2 Intertwining the concepts

Due to the increasing complexity of IT environments, the threat landscape is continuously expanding. Therefore, organisations should not consider risk

management, vulnerability management, and security patch management as separate entities. Each entity supports the actions of protecting the organisation's assets. Thus, one of the research questions aimed to provide an answer to *how do risk management, vulnerability management, and security patch management intertwine*. This subchapter discusses the theoretical findings.

Risk management, vulnerability management, and security patch management are information security management system (ISMS) controls. Thus, ensuring the confidentiality, integrity, and availability of information assets. While examining the relationship between different security concepts (Figure 7), the connection between the controls is present: an exploit of a vulnerability leads to risk, and the potential risk gets mitigated by implementing safeguards, for example, a security patch.

Risk management is part of all organisational activities. One could say that it could be seen as the initialising and unifying force in intertwining the concepts. In the risk management process, risk assessment covers the risk identification, analysis, and evaluation phases. During risk identification, the organisation identifies possible risks that might prevent them from achieving their objectives. Thus, intertwining vulnerability management into this phase as vulnerability management is the process of identifying vulnerabilities, and as was stated, vulnerabilities might pose a risk to the organisation. Risk treatment defines how the organisation addresses an identified risk. The security patch management process includes the activities to eliminate acknowledged vulnerabilities, thus, reducing the risk of exploitation. Therefore, by doing so, intertwining security patch management to the risk treatment phase of the risk management process. Figure 18 illustrates how the concepts intertwine.

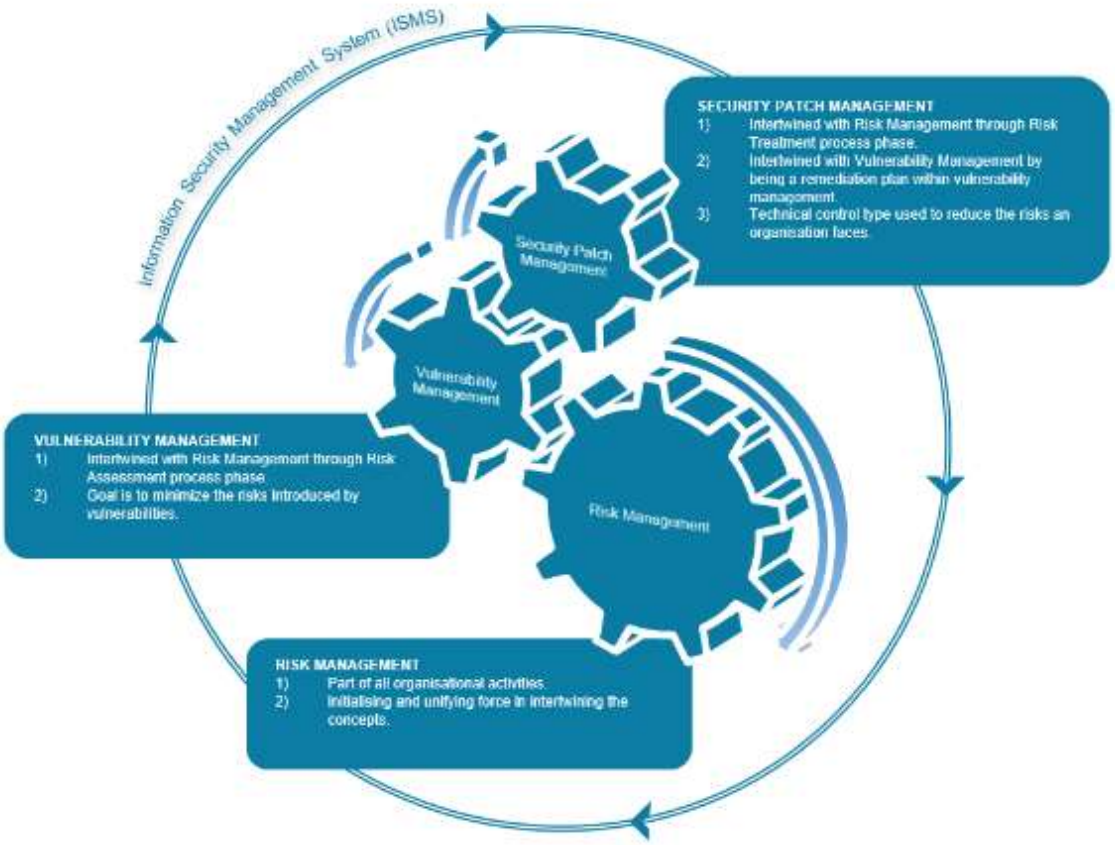

Figure 18. Intertwining the concepts

The findings from the literature review clearly stated that a need of having a risk-based approach to vulnerability management is strongly present. The organisations are required to adopt a new course of action from counting and remediating every emerging vulnerability to a risk-based prioritisation of vulnerabilities. The volumes of publicly disclosed vulnerabilities are so high that a vulnerability management program can only remain effective if vulnerabilities are rated based on risk. Thus, also these findings bind the concepts together as one step of implementing a risk-based approach to vulnerability management suggests integrating risk management into existing operational processes. As moving towards risk-based vulnerability management, the organisation is simultaneously shifting to risk-based security patch management. Adopting a risk-based approach is supported by the fact that patching each publicly disclosed vulnerability is not an option. Assessing vulnerabilities and patching demand based only on CVSS scores does not correctly evaluate the risk of potential exploitation.

8.3 Dividing security patch management duties

In 2019, the IT industry, especially the technology service providers, experienced a shocking reality check. The hacking campaign Cloud Hopper became public knowledge, while origin going back at least to early 2014. At least twelve cloud providers were affected by the hacking campaign. The cloud providers were used as pivot points to enter their clients' networks. To gain access to the cloud, APT10 (short for Advanced Persistent Threat) sent phishing emails to administrators with high-level access or cracked in through contractors' systems. (Barry & Volz 2010.)

The above-mentioned demonstrates that installing a patch is not insurance against all risks as the vulnerabilities also lie beyond technology, for example, in people and processes. In this case, the exploitation occurred through the service providers vulnerabilities, but this could easily be the case the other way around. Understanding the different scenarios on how patch management duties divide between the service provider and their client cannot be stressed enough. Thus, *how are the security patching duties divided between the service provider and their client* was one of the research questions. Neither parties, the service provider from the host perspective nor the client from the guest perspective, can undermine the importance of security patch levels.

Nowadays, holding all the strings has become even more challenging as the IT environments have become more complex. While doing so, the responsibilities are also divided between multiple operators. Here, it is vital to have an unambiguous duty division between the service provider and the client. Furthermore, in a multi-operator environment, the interconnectedness of the responsibilities should be clearly stated and documented. In this thesis, the concentration was on examining the duty division between the commissioner and their clients; more closely defining two specific duty division scenarios: traditional datacenter service and Software as a Service (SaaS). By focusing on these two scenarios, it was possible to demonstrate the considerable differences from the perspective of both parties.

Firstly, let us examine the findings concerning the traditional datacenter service scenario. Addressing security updates of an operating system scratches the surface of required patching actions. The hardware computing and networking layers, the services, the application runtime layer, and applications should also be kept up to date (Figure 19). Based on the findings of Doc3, while datacenter services are in question, the commissioner is responsible for the hardware computing, hardware networking, and OS level layer security patching. Furthermore, if the commissioner manages the service running on the server, patching is performed by the commissioner. Everything above that should fall into client responsibilities. As stated by Intvw7, the commissioner could perform the patching operation through the change management process, the client acting as the initiator. Here, the commissioner could examine does the present state of the operational environment correlate to the responsibility division described in Doc3. Also, as discussed with Intervw5, emphasis should be to examine the realisation of application runtime layer patching.

Secondly, when Software as a Service (SaaS) is in question, the duty division is quite simple. The service provider is responsible for the whole technological stack required to offer the service. Here, the client can concentrate solely on using the service while the service provider holds the obligation to maintain the service secure and up-to-date. Figure 19 illustrates the duty division between these two scenarios, the red line outlining the service provider duty layer.

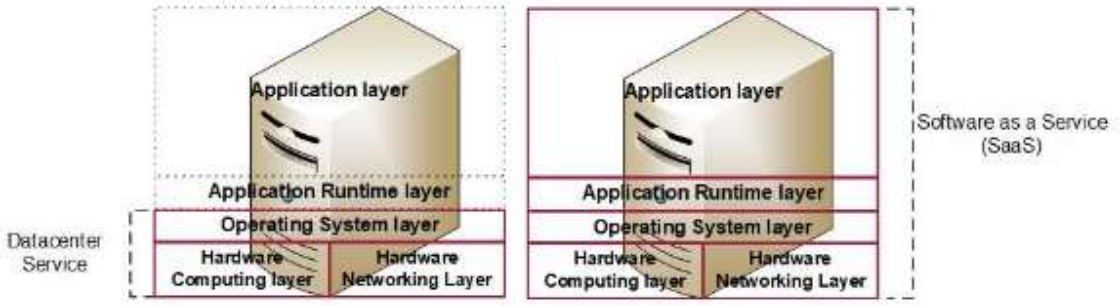

Figure 19. Security patch management duty division

Furthermore, Table 4 presents the duty division between the service provider and the client in a RACI matrix. RACI matrix is a responsibility chart in which R stands for Responsible, A for Accountable, C for Consulted and I for Informed.

Table 4. Duty division RACI matrix

Duty division RACI matrix	Datacenter Service		SaaS	
	Service Provider	Client	Service Provider	Client
Application layer	I/C/(R)	A/R	A/R	I
Application Runtime layer	I/C/(R)	A/R	A/R	I
Operating System layer	A/R	I	A/R	I
Hardware Computing layer	A/R	I	A/R	I
Hardware Networking layer	A/R	I	A/R	I

The deployment of guest servers occurs based on a need. Clients require servers to support their business operations. Thus, making the client the owner of the configuration item in question. Here, the service provider cannot solely dictate how the server is controlled and relies on the cooperation between the client to maintain the security posture of the server. As confirmed by the Intvw5, the operational aspect of security patch management is in place and functioning in day-to-day server management operations. Automation is in use, as it should be, to perform patching during agreed patching schedules. Hence, the commissioner is following the sub-control of the third CIS control (Chapter 6.2). However, as agreed with Intvw5, stressing the importance of security patch management and emphasising automatic patching methods to the clients should be constantly occurring actions. The client must be fully aware of their security patch management duties, as an outdated server is a security risk for both parties.

8.4 Development suggestions

The stated research problem was such by nature that it relied heavily on existing theoretical knowledge. Reflecting the theoretical knowledge against the target environment and adding value to the research from the commissioner's perspective, finding answers to the research question of *what development suggestions do arise from the findings* became vital. The following subchapters present the research findings and arisen development suggestions.

8.4.1 Assets and configuration items

Protecting IT environments against vulnerabilities is possible only if one knows what hardware and components are in the server room. Furthermore, one must be aware of what operating systems and software are running on the servers. To maintain this knowledge, one must have an asset register and configuration management database in place.

As mentioned by Intvw6, human interaction is present in the accuracy of configuration item data. Thus, the possibility for human errors is actively present. Intvw1 and Intvw2 both mentioned that efficient vulnerability management depends on accurate asset and configuration item data. In addition, as Intvw6 commented, also other factors like the production status of configuration items and invoicing depend on valid inventory. Intvw7 pointed out that the risk for inaccurate configuration item information is particularly present with project implementations, for example, client onboardings. Here, the commissioner could consider including a production review plan into their operational processes to ensure data accuracy. Furthermore, as Intvw6 suggested, the commissioner should have a process in place to regularly scan the shared and dedicated client environments for existing assets and configuration items and validate the information against the asset register and configuration management database.

During the research phase, the researcher did not manage to locate service asset and configuration management process and release management process descriptions that could have supported the research. Thus, suggesting examination of the presence and requirement for these documents. The fact that patch management is a subset of change, configuration, and release management processes supports the suggestion.

8.4.2 Exploit Prediction Scoring System (EPSS)

As theoretical findings stated, the CVSS score is composed of three metric groups. However, it is typical for only the base metrics to be published. Thus, leaving the temporal and environmental metrics out of the equation. A more

accurate evaluation of the impact on the computing environment would require presence from each metric group value. Thus, alongside the CVSS score, additional risk assessment considerations should be present.

EPSS estimates the probability of exploitation activity, helping in vulnerability remediation by guiding prioritisation decisions. FIRST has initiated Exploit Prediction Scoring System (EPSS) as one of their Special Interest Groups (SIGs). Furthermore, as EPSS SIG is one of the Standards Group, it is intended of being developed as a standard for internal use or external publication. (FIRST.org n.d.b.) Here, the commissioner could consider using EPSS globally if considerations of shifting towards a risk-based approach to vulnerability management exist. Regardless of the global considerations, the commissioner could examine EPSS implementation from the perspective of add-value to their local clients.

8.4.3 Security patch management

A review on Doc1 confirmed that the commissioner has a global patch and vulnerability management standard in place. Therefore, recognising the two as separate entities, working seamlessly together to provide asset security. Here, the possibility to locally influence the mandates from global standards is limited. However, based on Doc1 and Doc4, the commissioner should evaluate their client contracts against the global patch and vulnerability management standard, as it takes a stand on assets owned or managed by the commissioner used in internal, shared environments, or client-dedicated environments.

Efficient security patch management, especially patch prioritisation, depends on accurate CMDB data. The theoretical findings discussed in Chapter 6.3 support the statement that the ability to extract configuration items based on patch priority should be present. Therefore, the commissioner could assess the value of keeping client-specific configuration item prioritisation records. Thus, providing the ability for more accurate patch prioritisation as part of vulnerability remediation actions. The configuration item patch priority could, for example, be a

combination of factors such as accessibility to attackers, business criticality, and value.

The discussion with Intvw5 supported the researcher's existing knowledge that operational issues with security patch management occur, for example, with project implementations. Intvw5 emphasised that issues exist particularly with client onboardings, where as-is configuration item transitions to the service providers maintenance occur. Thus, the commissioner should closely examine the possible vulnerability risks and financial implications of as-is configuration item transitions. Production -status for an as-is configuration item should not be allowed before a production review. Production review could, for example, include stating the current patch level and verifying the functionality of patching methods. Thus, providing an understanding of the possible risks related to the configuration item and the actions required to reduce these risks.

From the operational security patch management perspective, the research implies a few development suggestions to consider. Firstly, as discussed with Intvw5, the client should provide a valid justification for not opting for the monthly automatic patching method. Choosing this option should also reflect possible Service Level Agreements (SLAs) related to security patch management.

Secondly, as stated by Intvw5, the commissioner should require the client to confirm two maintenance windows for each configuration item, primary and secondary. By doing so, allowing the ability to use the secondary maintenance window, for example, to distribute critical out-of-band patches or investigate patching issues.

Thirdly, the commissioner could examine the ability of the server automation system to automatically generate a ticket to the ITSM tool if patching fails for a configuration item, thus, providing better traceability and visibility of the security patch management operations. Moreover, providing data for other supporting processes, for example, problem management. The third suggestion arises from the researcher's knowledge of the commissioner's operational environment.

Fourthly, as discussed with Intvw3 and Intvw4, the commissioner could examine the present state of security patch management reporting. The ability to easily access and extract the patch level of a configuration item should exist, for example, from a reporting portal. Improving the security patch management reporting abilities could provide value internally and in client intercommunications, especially if there is a client requirement for security patch management reporting.

Finally, as the complexities of IT environments increase, so does the need for efficient patch management to provide a better security posture. Therefore, a seamless security patch management function is required to cover both the end devices that use the services to the servers providing the services. By expanding the ideology described in Doc2, the commissioner could, for example, evaluate the establishment of a virtual patch management team that could estimate the impact of announced patches on existing services. Here, this could be supported, for example, by having an internal patch advisory procedure in place. Furthermore, this might also allow a better ability to react to possible issues caused by the installed patches.

9 DISCUSSION

In this thesis, qualitative research method reliability criteria were used to examine the research reliability. While building up the theoretical framework, multiple sources were used as source material to verify the coherency of the information. Furthermore, attention was paid to the fact that the information obtained from the different sources supported each other. Thus, not providing contradictory statements. The research result reliability and confirmability realisation were enabled by report proofreading performed by one of the interviewees, the manager responsible for Windows Server platform operations, and the commissioner's thesis contact person.

While starting this project, the topic presented itself as an intriguing one. At early stages, it became evident that security patch management and vulnerability management require aspects from risk management, thus, presenting the

requirement for having a risk-based approach to vulnerability management. The focus should be on the issues that have a real potential to be realised. Thus, posing a negative impact on the business. The ability to make decisions based on risk assessment is vital. Here, the implementation of EPSS could support vulnerability remediation prioritisation decisions. Therefore, the first suggestion for a research topic arises; implementing EPSS into corporate vulnerability management program.

While considering the theoretical viewpoint, risk management, vulnerability management and security patch management are easily intertwined. Intertwining these concepts in practice is a whole other story. Without a doubt, the corporate security posture would improve if defined in the ISMS and the operational activities. Here, it is worth keeping in mind that the positive impact on improved security can with ease go in vain if not aligned with efficient hardware, OS, and software life cycle management. None of these ISMS controls can provide more reliable security if the IT environment is outdated. Therefore, one emphasis should be on the existence of effective life cycle management.

One might consider patch management a complex subject to approach as it is easily considered only from the operational aspect. As stated earlier, patch management is a process, not a technology. Throwing tools at patch management is not the answer. Every process, fulfilling a need and having a real purpose, can succeed if carefully planned, designed, defined, and documented. While not forgetting stating accountability between stakeholders and putting the process into practice. Yet, the operational aspects can be complex and cause headaches. Organisations can ease their pain by having a managed service provider (MSP) providing the service for them. However, by doing so, the organisation is not discharged from patch management related responsibilities. Perhaps the task to keep an IT environment up to date from a patching perspective might be easier if the security operations of the whole IT infrastructure would be self-administered or outsourced only to one service provider. The complexity and challenges surely increase while operating in a multi-supplier IT environment.

The reporting aspects need to be considered while examining security patch management. Is it enough to focus on reporting the installed patches rather than invariance of the installed patches? From a vulnerability management perspective, the invariance of an installed patch should be of interest. Here, the second suggestion for a research topic arises; to investigate ways to verify the invariance of installed patches after environmental changes. These are, for example, changes to the registry or Dynamic-link libraries (DLL).

This thesis has dealt with the tip of an iceberg of operational patch management, as the concentration was only on the Windows Server OS patching. While going through the theoretical source material, I could not shake off the feeling that even Microsoft appears slightly lost with their servicing cadence and quality. I have to wonder where the balance should lie when considering the reliability, stability, and quality of security-related patch releases, releasing technological enhancements, and new features. One could say that the present state of Microsofts' patching practices could be a risk for the confidentiality, integrity, and availability of IT environments. Thus, I can fully understand the frustration amongst IT administrators responsible for Windows Server OS patching. The cold sweat and worries caused by the approaching Patch Tuesday are well-grounded.

REFERENCES

Andrew, D. 2020. The ultimate guide to vulnerability scanning. WWW article. Updated 30 March 2020. Available at: https://www.intruder.io/guides/the-ultimate-guide-to-vulnerability-scanning [Accessed 12 May 2021].

Arban, M. 2017. Windows server patching: best practices. WWW article. Updated 22 March 2020. Available at: https://social.technet.microsoft.com/wiki/contents/articles/43406.windows-server-patching-best-practices.aspx [Accessed 10 January 2021].

AXELOS Limited. 2011. ITIL® glossary and abbreviations. PDF document. Available at: https://www.axelos.com/glossaries-of-terms [Accessed 21 May 2021].

Barry, R. & Volz, D. 2019. Ghosts in the clouds: inside China's major corporate hack. *The Wall Street Journal*. Available at: https://www.wsj.com/articles/ghosts-in-the-clouds-inside-chinas-major-corporate-hack-11577729061 [Accessed 20 May 2021].

Berman, P. 2014. Successful business process management: what you need to know to get results. Ebook. New York: AMACOM. Available at: cgi.percipio.com [Accessed 1 June 2020].

Bhajanka, P. & Lawson, C. 2018. Implement a risk-based approach to vulnerability management. PDF document. Available at: https://www.gartner.com/en/documents/3887782/implement-a-risk-based-approach-to-vulnerability-managem [Accessed 5 July 2020].

Branscombe, M. 2019. Everything you need to know about Windows updates. WWW article. Updated 11 March 2019. Available at: https://www.techrepublic.com/article/everything-you-need-to-know-about-windows-updates/ [Accessed 10 February 2021].

Center for Internet Security. 2019. CIS Controls™ version 7.1. PDF document. Available at: https://www.cisecurity.org/controls/ [Accessed 7 July 2020].

Cheng, H. 2016. Configuration Manager and simplified windows servicing on down level operating systems. WWW article. Updated 11 July 2018. Available at: https://techcommunity.microsoft.com/t5/configuration-manager-archive/configuration-manager-and-simplified-windows-servicing-on-down/ba-p/274056 [Accessed 10 February 2021].

Christensen, E. 2017. Patching with Windows Server 2016. Blog. Updated 27 June 2017. Available at: https://docs.microsoft.com/en-us/archive/blogs/mu/patching-with-windows-server-2016 [Accessed 11 February 2021].

Core Security. n.d. How mature is your vulnerability management program? Blog. Available at: https://www.coresecurity.com/blog/how-mature-your-vulnerability-management-program [Accessed 8 September 2020].

CVE. 2019a. About CVE. The MITRE Corporation. WWW document. Updated 6 November 2019. Available at: https://cve.mitre.org/about/index.html [Accessed 26 October 2020].

CVE. 2019b. Search CVE list. The MITRE Corporation. WWW document. Updated 4 January 2019. Available at: https://cve.mitre.org/cve/search_cve_list.html [Accessed 26 October 2020].

CVE. 2019c. CVE and NVD relationship. The MITRE Corporation. WWW document. Updated 2 August 2019. Available at: https://cve.mitre.org/about/cve_and_nvd_relationship.html [Accessed 26 October 2020].

CVE details. n.d. The MITRE Corporation. WWW document. Available at: https://www.cvedetails.com [Accessed 6 July 2020].

CWE. 2020. About CWE. The MITRE Corporation. WWW document. Updated 19 August 2020. Available at: https://cwe.mitre.org/about/index.html [Accessed 27 October 2020].

ENISA. 2019. Sate of vulnerabilities 2018/2019 – analysis of events in the life of vulnerabilities. PDF document. Published 14 January 2019. Available at: https://www.enisa.europa.eu/publications/technical-reports-on-cybersecurity-situation-the-state-of-cyber-security-vulnerabilities [Accessed 2 June 2020].

FIRST.org. n.d.a. The EPSS Model. WWW document. Available at: https://www.first.org/epss/model [Accessed 12 February 2021].

FIRST.org. n.d.b. Special interest groups (SIGs). WWW document. Available at: https://www.first.org/global/sigs/ [Accessed 3 July 2021].

FIRST.org. 2019. Common Vulnerability Scoring System (CVSS-SIG). WWW document. Available at: https://www.first.org/cvss/v3-1/ [Accessed 27 October 2020].

Foreman, P. 2010. Vulnerability management. Ebook. Boca Raton: CRC Press. Available at: cgi.percipio.com [Accessed 6 July 2020].

Haber, M. & Hibbert, B. 2018. Asset attack vectors: building effective vulnerability management strategies to protect organizations. Ebook. New York: Apress Media. Available at: cgi.percipio.com [Accessed 2 June 2020].

Harris, S. & Maymi, F. 2019. All-in-one CISSP exam guide. 8th edition. New York: McGraw-Hill Education.

Horev, R. 2019. A history of vulnerability management. Blog. Updated 14 March 2019. Available at: https://blog.vulcan.io/a-history-of-vulnerability-management [Accessed 6 July 2020].

Jacobs, J. & Roytman, M. 2019. Predictive vulnerability scoring system. PDF document. Available at: https://i.blackhat.com/USA-19/Thursday/us-19-Roytman-Jacobs-Predictive-Vulnerability-Scoring-System.pdf [Accessed 16 February 2020].

Jacobs, J., Romanosky, S., Adjerid, I. & Baker, W. 2019. Improving vulnerability remediation through better exploit prediction. PDF document. Available at: https://www.ftc.gov/system/files/documents/public_events/1415032/privacycon2019_sasha_romanosky.pdf [Accessed 16 February 2020].

Jacobs, J., Romanosky, S., Edwards, B., Roytman, M. & Adjerid, I. 2019. Exploit Prediction Scoring System (EPSS). PDF document. Available at: https://arxiv.org/ftp/arxiv/papers/1908/1908.04856.pdf [Accessed 16 February 2020].

Johnson, A., Dempsey, K., Ross, R., Gupta, S. & Bailey, D. 2011. NIST special publication 800-128. Guide for security-focused configuration management of information systems. United States of America: U.S. Department of Commerce. PDF document. Updated 10 October 2019. Available at: https://csrc.nist.gov/publications/detail/sp/800-128/final [Accessed 11 April 2021].

Joint task force transformation initiative. 2011. NIST special publication 800-39. Managing information security risk. Organization, mission and, information system view. United States of America: U.S. Department of Commerce. PDF document. Available at: https://csrc.nist.gov/publications/detail/sp/800-39/final [Accessed 30 June 2020].

Joint task force transformation initiative. 2020. NIST special publication 800-53. Security and privacy controls for federal information systems and organizations.

Revision 5. United States of America: U.S. Department of Commerce. PDF document. Available at: https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final [Accessed 11 May 2021].

Kakiuchi, Y. 2017. Get started with security update guide – new portal for security updates. Blog. Updated 13 April 2017. Available at: https://docs.microsoft.com/fi-fi/archive/blogs/yurikasensei/get-started-with-security-update-guide-new-portal-for-security-updates [Accessed 10 February 2021].

Kananen, J. 2013. Case-tutkimus opinnäytetyönä. Jyväskylä: JAMK University of Applied Sciences.

Kenna Security. n.d. Exploit Prediction Scoring System calculator, model version 1.0. WWW document. Available at: https://www.kennaresearch.com/tools/epss-calculator [Accessed 12 Fegruary 2021].

Kenna Security. 2018a. How to implement a risk-based approach to vulnerability management. PDF document. Available at: https://www.kennasecurity.com/wp-content/uploads/2020/03/how-to-implement-a-risk-based-approach-to-vulnerability-management.pdf [Accessed 5 July 2020].

Kenna Security. 2018b. How to manage vulnerabilities based on risk, rather than popularity. PDF document. Available at: https://www.kennasecurity.com/resources/beyond-the-hype-how-to-manage-vulnerabilities-based-on-risk-rather-than-popularity/ [Accessed 5 July 2020].

Kohnke, A., Shoemaker, D. & Sigler, K. 2016. The complete guide to Cybersecurity risks and controls. Ebook. Boca Raton: CRC Press. Available at: cgi.percipio.com [Accessed 29 June 2020].

Microsoft. n.d.a. Security update severity rating system. WWW document. Available at: https://www.microsoft.com/en-us/msrc/security-update-severity-rating-system [Accessed 10 February 2021].

Microsoft. n.d.b. Microsoft exploitability index. WWW document. Available at: https://www.microsoft.com/en-us/msrc/exploitability-index?rtc=1 [Accessed 10 February 2021].

Microsoft. 2019a. CVE-2019-0708. Remote Desktop Services remote code execution vulnerability. WWW document. Available at: https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2019-0708 [Accessed 10 February 2021].

Microsoft. 2019b. Install server core. WWW document. Updated 21 May 2019. Available at: https://docs.microsoft.com/en-us/windows-server/get-started/getting-started-with-server-core [Accessed 14 February 2021].

Microsoft. 2020. Servicing stack updates. WWW document. Updated 11 April 2020. Available at: https://docs.microsoft.com/en-us/windows/deployment/update/servicing-stack-updates [Accessed 11 February 2021].

Microsoft. 2021. Security Update Guide, vulnerabilities. WWW document. Available at: https://msrc.microsoft.com/update-guide/vulnerability [Accessed 13 February 2021].

Moore, S. 2017. Focus on the biggest security threats, not the most publicized. WWW document. Updated 2 November 2017. Available at: https://www.gartner.com/smarterwithgartner/focus-on-the-biggest-security-threats-not-the-most-publicized/ [Accessed 5 July 2020].

Nicastro, F. 2011. Security patch management. 2nd edition. Ebook. Boca Raton: CRC Press. Available at: cgi.percipio.com [Accessed 1 June 2020].

NVD. n.d. National Institute of Standards and Technology (NIST). WWW document. Available at: https://nvd.nist.gov/ [Accessed 27 October 2020].

Ojasalo, K., Moilanen, T. & Ritalahti, J. 2015. Kehittämistyön menetelmät. Uudenlaista osaamista liiketoimintaan. 4th edition. Ebook. Helsinki: Sanoma Pro Oy. Available at: ellibslibrary.com [Accessed 16 February 2020].

The OWASP® Foundation. n.d. Vulnerability scanning tools. WWW documents. Available at: https://owasp.org/www-community/Vulnerability_Scanning_Tools [Accessed 11 May 2021].

The OWASP® Foundation. 2020. OWASP vulnerability management guide (OVMG). Updated 1 June 2020. PDF document. Available at: https://owasp.org/www-project-vulnerability-management-guide/OWASP-Vuln-Mgm-Guide-Jun05-2020.pdf [Accessed 12 May 2021].

Panetta, K. 2020. Gartner top 10 security projects for 2020-2021. Blog. Updated 15 September 2020. Available at: https://www.gartner.com/smarterwithgartner/gartner-top-security-projects-for-2020-2021/ [Accessed 30 October 2020].

Risto, J. 2020. Vulnerability Management Maturity Model. Security Boulevard. Blog. Updated 5 July 2020. Available at: https://securityboulevard.com/2020/07/vulnerability-management-maturity-model/ [Accessed 30 October 2020].

SFS-ISO/IEC 27000:en. 2020. Information technology. Security techniques. Information management systems. Overview and vocabulary (ISO/IEC 27000:2018).

SFS-ISO/IEC 27002:en. 2017. Information technology. Security techniques. Code of practice for information security controls (ISO/IEC 27002:2013, Cor 1:2014 and Cor 2:2015).

SFS-ISO/IEC 27005:en. 2018. Information technology. Security techniques. Information security risk management.

SFS-ISO 31000:en. 2018. Risk management. Guidelines.

Souppaya, M. & Scarfone, K. 2013. NIST special publication 800-40. Guide to enterprise patch management technologies. Revision 3. United States of America: U.S. Department of Commerce. PDF document. Available at: https://csrc.nist.gov/publications/detail/sp/800-40/rev-3/final [Accessed 1 February 2020].

Wilcox, J. 2018. Windows 10 update servicing cadence. Blog. Updated 8 January 2018. Available at: https://techcommunity.microsoft.com/t5/windows-it-pro-blog/windows-10-update-servicing-cadence/ba-p/222376 [Accessed 11 February 2021].

Williams, J. 2019. Why your vulnerability management strategy is not working – and what to do about it. SANS Institute. PDF document. Available at: https://www.sans.org/reading-room/ [Accessed 7 July 2020].

LIST OF FIGURES

LIST OF TABLES

CVSS metric groups and metric values

Table 5. CVSS metric groups and metric values (FIRST.org 2019)

Metric group	Metric name	Description	Metric values
Base	Attack vector (AV)	An indicator of the level of access required to exploit the vulnerability.	• Network (N) – Remotely exploitable vulnerability, exploitable at the protocol level one or more hops away up to and including entire Internet. • Adjacent (A) – Attack is limited at the protocol level to a logically adjacent topology. • Local (L) – Vulnerability is not exploitable over network. Read/write/execute capabilities are required to perform an attack. • Physical (P) – Physical interaction is required to perform an attack.
	Attack complexity (AC)	Describes the conditions beyond the attacker's control that must exist to exploit the vulnerability.	• Low (L) - No specific pre-conditions are required to exploit the vulnerable component. • High (H) – Effort is required as a successful attack depends on conditions beyond the attacker's control.
	Privileges required (PR)	Describes the level of privileges required before exploiting the vulnerability.	• None (N) – No privilege or special access required to carry out an attack. • Low (L) – Basic user level privileges required. • High (H) – Administrative or similar privileges required.
	User interaction (UI)	Describes whether the vulnerability can be exploited solely at the will of the attacker.	• None (N) – User interaction not required to carry out an attack. • Required (R) – User interaction required.
	Scope (S)	Describes whether a vulnerability in one vulnerable component impacts resources in components beyond its security scope.	• Unchanged (U) – Affects only resources managed by the same security authority. • Changed (U) – Affects resources beyond the security scope managed by the security authority of the vulnerable component.

Appendix 1/2

Metric group	Metric name	Description	Metric values
Base	Confidentiality (C)	Measures the impact to the confidentiality of the resources managed by a software component due to a successfully exploited vulnerability.	• High (H) – Total loss of confidentiality, resulting in all resources within the impacted component divulged. • Low (L) – Some loss of confidentiality. • None (N) – No loss of confidentiality.
	Integrity (I)	Measures the impact to integrity of a successfully exploited vulnerability.	• High (H) – Total loss of integrity, or a complete loss of protection. • Low (L) – Data modification is possible but does not have a direct, serious impact on the impacted component. • None (N) – No loss of integrity.
	Availability (A)	Measures the impact to the availability of the impacted component.	• High (H) – Total loss of availability, access to resources are fully denied. • Low (L) – Performance is reduced, or interruptions can occur in resource availability. • None (N) – No impact to availability.
Temporal	Exploit code maturity (E)	The likelihood of the vulnerability being attacked. Typically based on the current state of exploit techniques, exploit code availability, or active “in-the-wild” exploitation.	• Not defined (X) – Insufficient information to choose one of the other values. Has no impact on the overall temporal score. • High (H) – Functional autonomous code exists, or no exploit is required, and details are widely available. • Functional (F) – Functional exploit code available. • Proof-of-concept (P) – Proof-of-concept code available. Attack demonstration not practical for most systems. • Unproven (U) – No exploit code available, or an exploit is theoretical.

Appendix 1/3

Metric group	Metric name	Description	Metric values
Temporal	Remediation level (RL)	Describes the remediation level of a vulnerability which acts as an important factor for prioritisation.	• Not defined (X) - Insufficient information to choose one of the other values. Has no impact on the overall temporal score. • Unavailable (U) – No solution available. • Workaround (W) – Unofficial, non-vendor solution available. • Temporary fix (T) – An official but temporary fix available. • Official fix (O) – Complete vendor solution available.
	Report confidence (RC)	Describes the degree of confidence in the existence of the vulnerability and the credibility of the known technical details.	• Not defined (X) - Insufficient information to choose one of the other values. Has no impact on the overall temporal score. • Confirmed (C) – Detailed reports exists, or functional reproduction is possible. • Reasonable (R) – Significant details are published. Full confidence in the root cause does not exists, or access to source code not available to fully confirm all of the interactions. • Unknown (U) – Reports of impacts are present. Cause of vulnerability is unknown, or reports may differ on the cause or impacts of the vulnerability.

Appendix 1/4

Metric group	Metric name	Description	Metric values
Environmental	Modified base metrics	Enables the analyst to override individual base metrics based on the specific characteristics of an organisation's environment. • Modified attack vector (MAV) • Modified attack complexity (MAC) • Modified privileges required (MPR) • Modified user interaction (MUI) • Modified scope (MS) • Modified confidentiality (MC) • Modified integrity (MI) • Modified availability (MA)	• The same values as the corresponding base metric, as well as not defined (the default).
	Security requirements: • Confidentiality requirement (CR) • Integrity requirement (IR) • Availability requirement (AR)	Enables the analyst to customise the CVSS score depending on the importance of the affected IT asset to the organisation. Measured in terms of confidentiality, integrity, and availability.	• Not defined (X) – insufficient information to choose one of the other values. Has no impact on the overall environmental score. • High (H) – Loss is likely to have catastrophic adverse effect on the organisation or individuals associated with the organisation. • Medium (M) – Loss is likely to have a serious adverse effect on the organisation or individuals associated with the organisation. • Low (L) - Loss is likely to have only a limited adverse effect on the organisation or individuals associated with the organisation.

CVSS scoring rubrics for the Base metric group

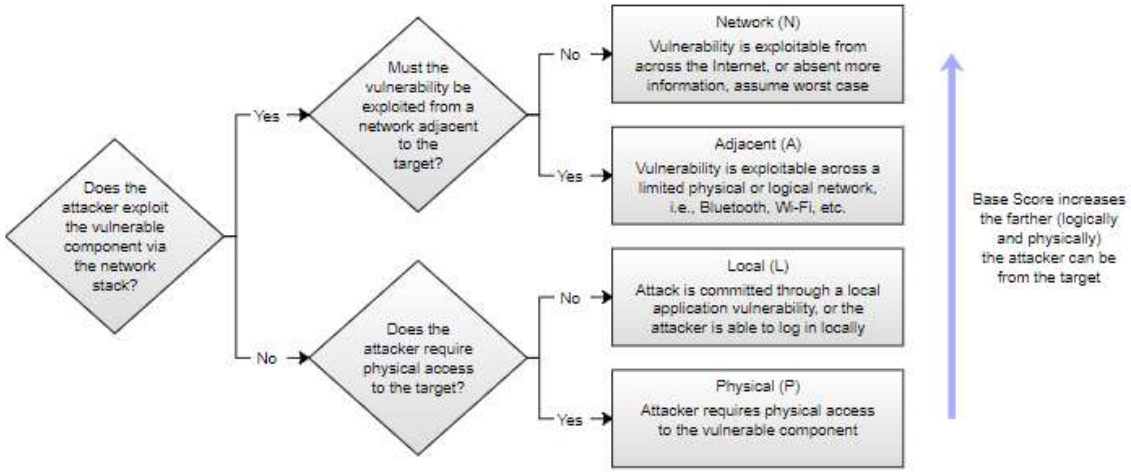

Figure 20. Exploitability metrics: Attack Vector rubric (FIRST.org 2019)

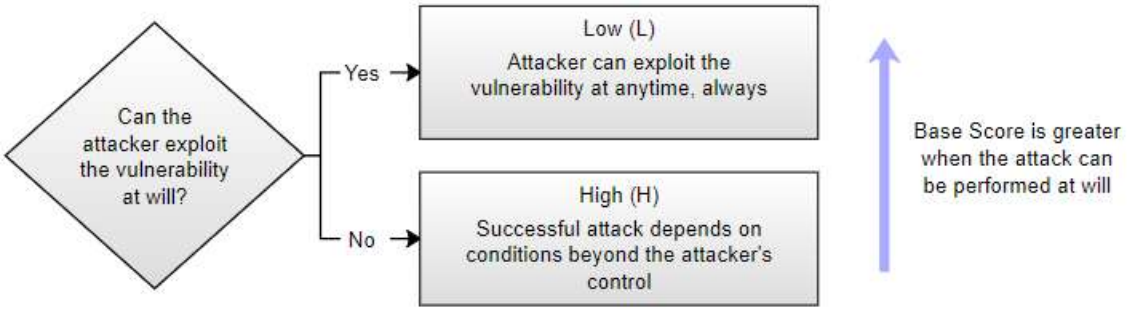

Figure 21. Exploitability metrics: Attack Complexity rubric (FIRST.org 2019)

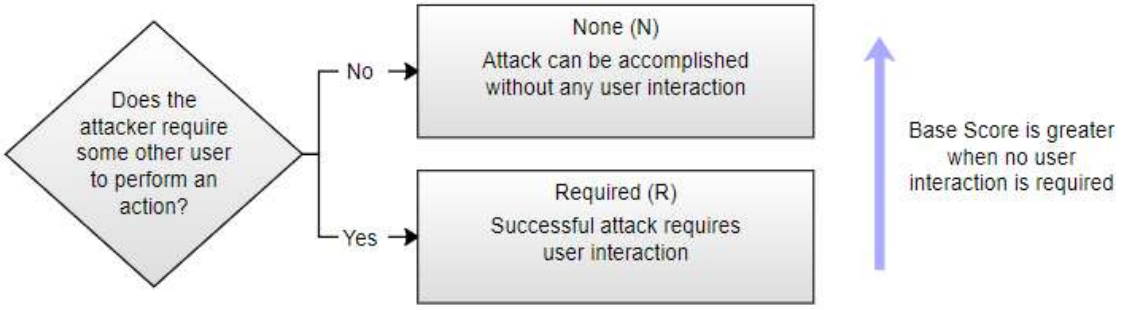

Figure 22. Exploitability metrics: User Interaction rubric (FIRST.org 2019)

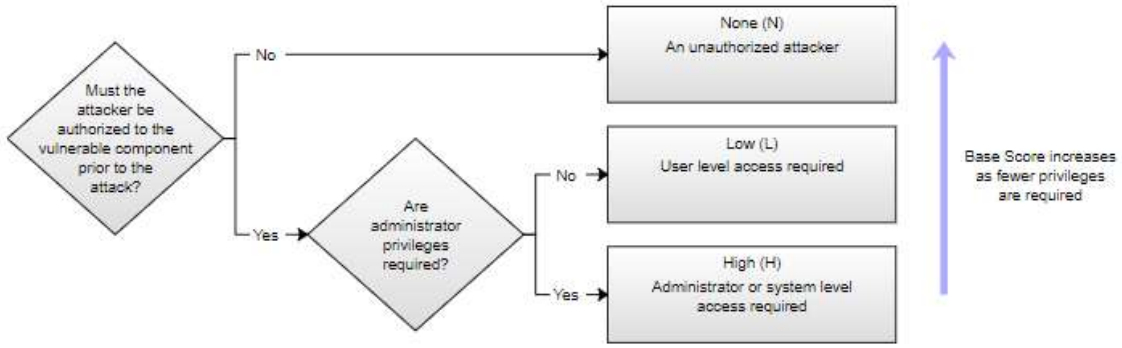

Figure 23. Exploitability metrics: Privileges Required rubric (FIRST.org 2019)

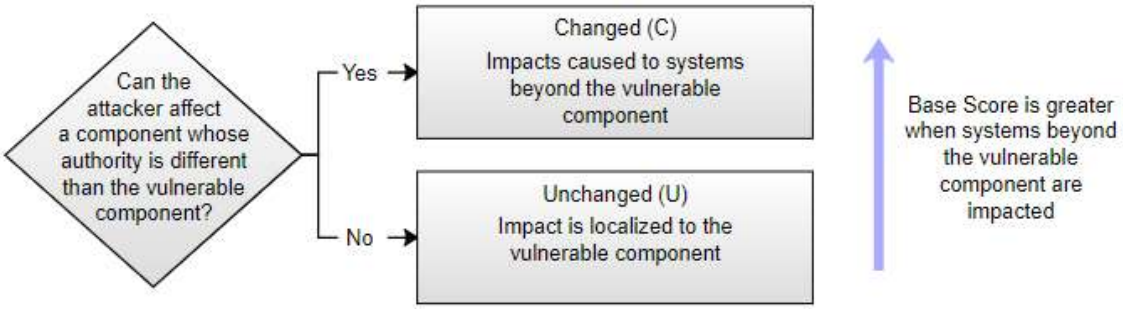

Figure 24. Scope rubric (FIRST.org 2019)

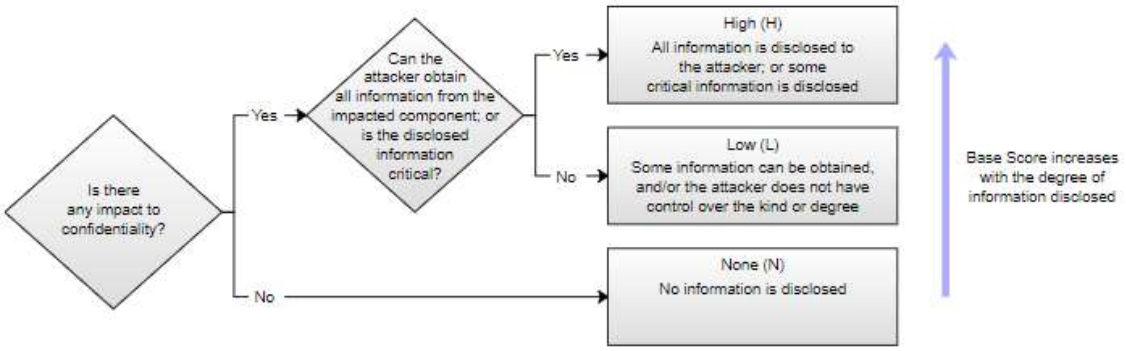

Figure 25. Impact metrics: Confidentiality Impact rubric (FIRST.org 2019)

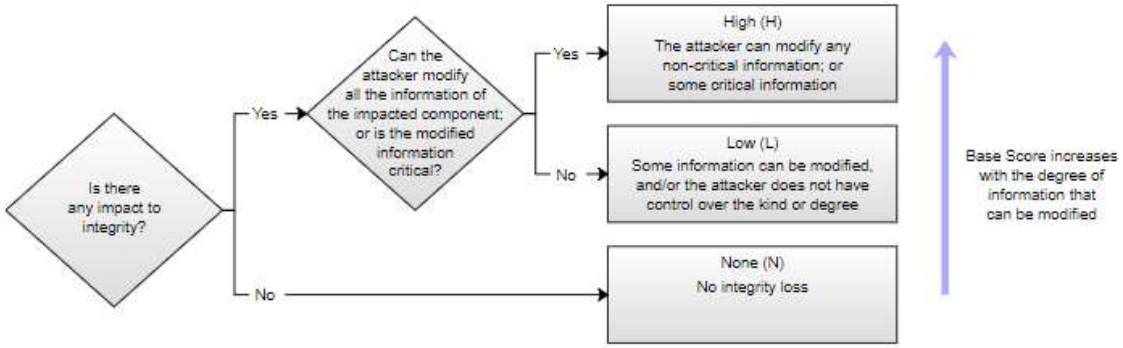

Figure 26. Impact metrics: Integrity Impact rubric (FIRST.org 2019)

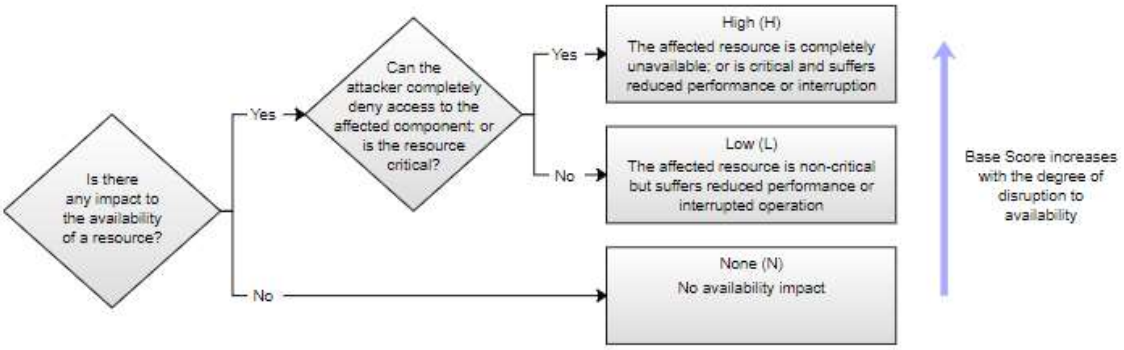

Figure 27. Impact metrics: Availability Impact rubric (FIRST.org 2019)