Hochschule Neu-Ulm
University of Applied Sciences

Master Thesis,

in the Master's Program

Digital Transformation and Global Entrepreneurship (M.Sc.)

at the University of Applied Sciences Neu-Ulm

Topic:

Assessing the Impact of Unpatched Software on Cybersecurity Vulnerabilities in an e-Assessment Platform During Digital Transformation (Software Installation & Upgrades): A Case Study of "Campus Ready GmbH - Startup"

Degree Program: Digital Transformation and Global Entrepreneurship (M.Sc.)

1st Examiner: Professor Dr. Klaus Lang

2st Examiner: Professor Dr. Daniel Schallmo

Author: Ömer Kücükacar

Enrolment number: 328984, Heinz-Rühmann-Str. 9 · 89231 Neu-Ulm

The topic received: 02.10.2024

Date of submission: 25.02.2025

2025

Hochschule Neu-Ulm
University of Applied Sciences

ABSTRACT

The rapid digitalization of universities has instigated the increased adoption of e-Assessment platforms for universities' online admission and academic evaluations. However, cybersecurity vulnerabilities, particularly those related to unpatched software during the process of upgrades and installations, considerably endanger such platforms. This case study investigates the contribution of unpatched software to the cybersecurity vulnerabilities of the Campus Ready e-Assessment platform during software installation and upgrades. Using a case study design, the research utilized quantitative methods such as vulnerability scanning, penetration testing and statistical analysis in order to measure security vulnerabilities resulting from unpatched software modules.

Research shows that there are higher chances of cyber-attacks like data breaches, malware and unauthorized access if software is unpatched. Some of the most critical vulnerabilities discovered include misconfigurations, weak authentication mechanisms and exposure of sensitive information because of outdated security patches. Against these threats, the research suggests a formalized cybersecurity framework based on the ISO/IEC 27001:2022 standards specifications, this framework highlights proactive patch management and vulnerability and configuration management controls to enhance security with actionable suggestions to education institutions, enterprises and digital platform providers on how to enhance cybersecurity controls, safeguard sensitive information and preserve stakeholder trust.

Keywords: Cybersecurity, e-Assessment, unpatched software, vulnerability management, patch management, ISO/IEC 27001, digital transformation, educational technology, software security.

Hochschule Neu-Ulm
University of Applied Sciences

Acknowledgment

This success is a landmark in my academic and professional career. It has been a challenge but one which I have relished and I sincerely appreciate all the people who have directed and supported me in this pursuit.

Above all, I would like to thank God, whose wisdom and grace directed and guided me every step of the way. Only through His blessings was this success attainable.

I am also profoundly grateful to the Hochschule Neu-Ulm: University of Applied Sciences for offering me a satisfying study experience and a friendly examination atmosphere. The college has influenced my thinking regarding digital transformation and global entrepreneurship and I am grateful for the learning and insight acquired through the studies.

I am most thankful to "My Mentor", my principal supervisor, for his direction, motivation and foresight during this research. His expertise has made an enormous difference to the course of my research and its culmination. Special gratitude is extended to "Campus Ready, whose support was greatly beneficial to this project. I am particularly grateful to Head of Operations at Campus Ready, for the time, effort and experience contribution. His contributions were critical to the success of this research and I appreciate him immensely for being willing to facilitate this study and make it a reality and his commitment to building an innovative test environment.".

I would also like to thank my friends and classmates for their unwavering support, motivation and cooperation which have made my university life a memorable experience.

Finally, I am eternally grateful to my family, especially my parents, for their unconditional love, support and belief in me. Without their encouragement and sacrifices, I would not have been able to achieve this milestone. This accomplishment is as much theirs as it is mine.

To everyone who has contributed to this journey in any way, I extend my heartfelt thanks and appreciation.

Hochschule Neu-Ulm
University of Applied Sciences

TABLE OF CONTENTS

List Of Figures

List Of Tables

List of Abbreviations & Acronyms

DoS	Denial of Service
CTO	Chief Executive Officer
PII	Personally Identifying Information
MBSA	Microsoft Baseline Security Analyzer
ZAP	Zed Attack Proxy
SSL	Secure Sockets Layer
OWASP	The Open Worldwide Application Security Project
HTTP	Hypertext Transfer Protocol
XSS	Cross-Site-Scripting
SQL	Structured Query Language
XaaS	Anything as a Service
RPA	Robotic Process Automation
REST-based API	Representational State Transfer (REST) Application Programming Interface (API)
CVSS	Common Vulnerability Scoring System
WFH	Work From Home
WLAN	Wireless Local Area Network
AI	Artificial Intelligence
LLM	Large Language Models
SDG	The Sustainable Development Goals
AR	Augmented Reality
PLG	Personalized Assessment & Gamification
CIA	The Central Intelligence Agency
ISO/IEC	International Organization for Standardization (ISO) International Electrotechnical Commission (IEC)
IT	Information Technology
RQ1, RQ2	Number of Research Questions: 1 and 2
H1, H2	Hypothesis: Number 1 & 2
CTO	Chief Technology Officer
v4.0	Version 4.0
UN	United Nations
IP	Internet Protocol address
SCCM	System Center Configuration Manager
MECM	Microsoft Endpoint Configuration Manager
SLAs	Service Level Agreements
ISO	Information Security Officer
CMDBS	Configuration Management Databases
IAC	Infrastructure as Code
CIS	Critical Security Controls
SHH	Secure Shell
CI	Configuration Item
CISA	Cybersecurity and Infrastructure Security Agency
GDPR	General Data Protection Regulation

1. INTRODUCTION

This introduction section provides background and context to the research topic which is focusing on the security vulnerabilities in Campus Ready e-Assessment platform during its (Software upgrades/installations), specifically related to unpatched software. The research problem identifies the critical issues associated with cybersecurity risks in this platform. Additionally, the study's objectives and research questions clarify the focus of the research. The significance of the study emphasizes the importance of addressing cybersecurity challenges for Campus Ready also for other parties like educational institutions and platforms. Lastly, the structure of the thesis outlines the roadmap for the research to guide readers through the study's methodology, findings, implementation, validation of research questions & hypothesis, limitations & further research and conclusions.

1.1 Background and Context

Digital platforms like e-Assessment platforms are essential in today's business and education/universities as they significantly enhance user engagement, scalability and assessment delivery efficiency. By leveraging cutting-edge digital technologies like artificial intelligence (AI) and machine learning (ML), these platforms enable innovative solutions like testing strategies that cater to diverse educational needs to reduce dependence on traditional, resource-intensive methods such as in-person interviews and automating aptitude tests through scalable cloud-based infrastructures. E-Assessment platforms also promote global initiatives such as inclusion and innovation in order to offer wider access to higher education and empowering universities to evaluate a larger pool of applicants more effectively and fairly (Rathore, 2023).

In addition, technological advancements such as cloud computing, AI, big data analytics and machine learning play a significant role in the development of more dynamic and interactive e-Assessment environments. Since these technologies allow platforms to deliver real-time, on-demand, internationally accessible content, breaking down traditional barriers to assessment and improving the accessibility of e-Assessment for people from all backgrounds (Mukul & Büyüközkan, 2023).

As education undergoes a digital transformation (Developing digital education assets), e-Assessment platforms have become integral in streamlining academic operations and addressing inefficiencies, such as those in university admissions. Campus Ready, an example

of such innovation, undergoing a developing software (digital education assets) to streamline the assessment process for international college applicants in Europe (Sembey et al., 2024).

However, cybersecurity threats loom large due to their reliance on web-based infrastructures and the integration of emerging technologies like cloud computing. These platforms handle vast amounts of sensitive digital data to make them prime targets for cyberattacks and exposing vulnerabilities that could compromise user privacy and system functionality (Wei, 2023).

Also, e-Assessment platforms can face significant cybersecurity challenges, particularly with unpatched software vulnerabilities, which remain a persistent concern for general digital platforms managing sensitive data in an ever-evolving technical environment, therefore this issue is not a problem for e-Assessment platforms also for all businesses who is developing a software for their business success (Rathore, 2023).

Unpatched software refers to systems or applications mean that have not received the latest security updates to leave them open to exploitation. Therefore, Hackers can target these vulnerabilities to insert malicious code or gain unauthorized access to confidential information. For platforms like Campus Ready, which interact with institutional systems and store personal data from applicants, these vulnerabilities can lead to data theft, service disruptions and diminished trust from stakeholders. Additionally, given the reliance on advanced digital infrastructure and AI models, addressing these vulnerabilities is crucial, as any breach could disrupt operations or compromise sensitive applicant data (Akacha & Awad, 2023, p.1-10).

The urge to remain safe while updating software becomes stronger, as scalability and functionality tend to take priority at the expense of robust security. Despite how much e-Assessment platforms contribute to greater efficiency and access, they expose institutions to threats typical in modern digital settings. These threats, if not managed, can overshadow the benefits and compromise universities in their efforts to adopt such platforms. Therefore, cybersecurity must be included in the digital transformation processes (Software Installation & Upgrades) to allow platforms like Campus Ready to operate securely while encouraging innovation (Dissanayake et al., 2021).

Therefore, to avoid these cybersecurity problems and allow e-Assessment platforms like Campus Ready to operate securely while encouraging innovation, the implementation of robust cybersecurity frameworks is needed. ISO 9001:2015 Quality Management (Bouchetara et al., 2022) and ISO/IEC 27001:2022 Information Security Management Systems provide

elaborate guidelines that can be used to minimize risks, improve patch management and improve vulnerability, change and configuration management measures. By adhering to these universally accepted standards, Campus Ready is able to institutionalize best practices in quality assurance and cybersecurity in order to secure the platform as reliable and compliant to universal standards. The incorporation of these frameworks will not only enhance the security position of the platform but also ease its expansion and the reputation of the platform in the competitive education technology market, while safeguarding sensitive applicant details and evading potential cyberattacks (Bouchetara et al., 2022; Proença & Borbinha, 2018; ISO 27001:2022, 2023).

1.2 Relevance of the Topic

In recent years, cybersecurity has emerged as a crucial element of all-encompassing security plans. The swift expansion of the internet and globalization, fueled by ongoing improvements in network technologies, has given people, businesses and governments access to previously unheard-of opportunities. Information collection, communication, fundraising and public relations have all changed because of digital transformation (software upgrades & installations) which influences a variety of stakeholders, including students, hackers, security forces and adversaries (Fertig et al., 2019).

With the advent of web-based systems which offer increased accessibility, interaction and efficiency and the digital revolution has completely changed the assessment scene. Users can interact with academic systems at any time and from any location thanks to cloud-based e-Assessment solutions like Campus Ready which allow for both live and asynchronous assessments. These platforms use artificial intelligence (AI) and ML (machine Learning) techniques to improve their capabilities by delivering sophisticated features including tracking user progress, automatically analyzing assessments and instantly providing feedback or results. Traditional evaluation techniques have been drastically redesigned by the incorporation of digital technology, becoming more user-centric, scalable and adaptive (Alasa et al., 2022, pp. 1336-1350).

In addition, despite e-Assessment platforms revolutionary potential, they are vulnerable to system vulnerabilities that, if unpatched, it could be used by hostile actors to interfere with business operations or obtain unauthorized access to vital information. These vulnerabilities not only endanger the platforms but also the digital infrastructures they are a part of, such as the government or other vital sectors. The assessment industry which includes educational institutions their staff and the digital platforms that enable these operations and it is regarded as an essential part of the country’s infrastructure (Maatuk et al., 2021, pp. 20-24).

Examining in depth how software upgrades impact on their system vulnerabilities has become essential due to the increasing dependence on e-Assessment platforms. In order to keep these systems safe, dependable and resistant to changing cyberthreats, the cybersecurity measures are required. In order to safeguard sensitive assessment data and maintain user confidence and facilitate the broad adoption and success of e-Assessment systems, these issues must be resolved. If this is not done, the platforms might be left vulnerable to cyberattacks on bigger and more important networks due to their lack of security (Maatuk et al., 2021, pp.23-26).

1.3 Research Problem

The rapid adoption of digital technologies in university admissions, particularly through undergoing developing a software, has significantly enhanced accessibility, scalability and efficiency in academic assessments. However, this revolution online has been accompanied by a deluge of cybersecurity challenges, the majority of which are related to unpatched software vulnerabilities. These vulnerabilities, typically resulting from neglected security patches or delayed updates, expose digital systems to cyberattacks on sensitive applicant data, such as integrating academic transcripts and personal information (Al-Ansi et al., 2023, pp. 1-8). While applications like Campus Ready become more pivotal to the admission process, software patch vulnerabilities not being addressed can pose risks po tentially seriously undermining their security, performance and users' trust (Rathore, 2023).

Despite having software patches available, organizations like e- Assessment websites are still struggling with effective patch management. Detection, release and deployment of patches in a timely manner remain major challenges lead to systems vulnerable to cyberattacks (Pulliainen, 2016, pp. 1-5). Sophisticated technologies such as cloud computing, AI and machine learning complicate security measures are causing loopholes in the platforms' security architectures. These weaknesses compromise the integrity of the e-Assessment process and result in data loss, service interruptions and loss of stakeholder confidence (Al-Ansi et al., 2023, pp. 3-10).

Moreover, there are no studies on how unpatched software vulnerabilities affect e-Assessment platform security in online university admissions (Ching-Huang et al., 2008, p. 1251). This inability to understand is critical since the increasing sophistication of digital infrastructures in e-Assessmenplatforms makes it increasingly difficult to monitor and respond to vulnerabilities in real-time. With the national security concerns posed by the increasing interconnection of digital systems, there is more need for sophisticated research and

techniques to safeguard these platforms than ever (GowherHassan & Hassan, 2023, pp. 1-3).

Hence, the research problem is to understand how unpatched software vulnerabilities and rapid digital transition (Software Installation & Upgrades) of university admissions can threaten the security, operation and credibility of e-Assessment platform. This study will explore the impact of unpatched software on e-Assessment platform security and suggest solutions in the form of patch management systems and enhanced cybersecurity regulations to effectively mitigate these threats.

Unpatched Software: A Gateway to Cyberattacks

Unpatched software remains the most significant web-based system risk like Campus Ready, as it provides an open window to online attacks. Delays or neglection of patching software is what makes an attacker utilize established vulnerabilities in the system, creating opportunities for executing various forms of malicious attacks. These include data-compromising malware infections to Denial of Service (DoS) attacks that disrupt platform operations to render it inaccessible to users (Auoissa & Auoissa, 2025). Such security threats to the unpatched software go beyond the direct security breach; they can have the extended impact on the loss of user trust, reputation loss, and legal charges resulting from failure to comply with data protection standards (Antonio, 2017, pp. 1–5).

In order to address these challenges, organization should adopt a proactive stance on cybersecurity. This includes implementing regular software updates and patches to ensure that known vulnerabilities are fixed before they can be exploited. Additionally, fostering a culture of cybersecurity within the organization with a dedicated team to monitor and address vulnerabilities, is crucial. An organization-wide commitment to cybersecurity, coupled with automated tools and robust patch management processes which will significantly reduce the likelihood of successful cyberattacks. Through these measures, e-Assessment platforms like Campus Ready can prevent data breaches to minimize downtime and maintain the integrity and security of their operations (Akacha & Awad, 2023, pp. 1-27).

Unpatched software often serves as a gateway for various types of cyberattacks, such as including: Cyberattacks frequently exploit unpatched software allowing malicious entities such as viruses and Trojan horses to infiltrate online platforms. These breaches can lead to data theft, file corruption or even unauthorized modifications to academic records (Habib et al., 2020, pp. 1-3).

Several factors contribute to delays in patching such as including human errors, complex IT systems and limited resources. Addressing these challenges requires a proactive security policy, automated and methodical patch management and robust organizational cybersecurity measures to mitigate the risk of ransomware and data breaches (Dissanayake et al., 2022, pp. 1-12).

While many websites have made significant strides in features, flexibility and accessibility, progress comes with security vulnerabilities. New features and third-party functionality make systems more complex and are likely to leave them vulnerable to malicious attacks. This is the justification for having strict security controls against intrusions on the basis of unpatched elements or insecure interfacing with external systems (Schmidt & Tang, 2020; Kepuska & Tomasevic, 2024, pp. 1-15).

In addition to unpatched software, web platform security vulnerabilities frequently result from "system misconfigurations" and human error. Misconfigurations such as servers configured improperly to expose private data or firewalls permitting unauthorized traffic to create holes for attackers. These vulnerabilities usually stem from insufficient knowledge or technical skills (Mell et al., 2005, pp. 2-7). These weaknesses are all the more severe since such networks are likely to be connected with upper-level networks and increasing the scale of attacks. This speaks volumes for the importance of robust defenses (Ching-Huang et al., 2008, p. 1252; Kepuska & Tomasevic, 2024, pp. 2-19).

Security Misconfiguration Attack – An Example: Scenario: Security Misconfiguration Attack on Internal Network

One of the examples of Security Misconfiguration Attacks starts with an attacker who gets into an internal network because of poor outside defenses. While inside the network, the attacker scans the network for devices that have default passwords. Taking advantage of this weakness, the attacker is able to get into several devices without needing to penetrate other security protection. From there, the attacker is able to reach compromised devices and internal resources such as databases and file servers, leading to unauthorized data exfiltration or system compromise. This attack highlights the tremendous risk of failing to modify default configurations is facilitating attackers to exploit these configurations easily (OWASP, 2021).

1.4 Objective of the Study

The study will examine how susceptible software supports cybersecurity threats in the Campus Ready e-Assessment system that are connecting these vulnerabilities to broader issues of digital transformation and education technology. It investigates threats like unauthorized access and system downtime are placing emphasis on robust practices to protect stakeholders, including institutions, students and policymakers (Dissanayake et al., 2021).

This study will utilize the importance of effective patching, vulnerability management, and configuration management within the process of digital software updating to identify how they can be utilized to defend against cybersecurity attacks. Using the Campus Ready e-Assessment platform as the case study, it will analyze risks that may arise from delayed or inadequate patching, such as data breaches and service disruption. Because Campus Ready is a new, under-development platform, results of vulnerability scanning and penetration testing will be utilized to assist in the determination of security vulnerabilities such as misconfiguration from unpached software. The study will also give recommendations regarding best practices that organizations can utilize to ensure systems performance and reliability during upgrades (Dissanayake et al., 2022).

1.1. Research Questions:

The thesis will help to answer the following research questions:

RQ 1: How does unpatched software during software upgrades lead to specific cyber vulnerabilities in the Campus Ready e-Assessment platform (Software), such as data breaches and malware attacks?

RQ 2: How can an effective patch, vulnerability and configuration management strategy be recommended to address and mitigate the risks associated with unpatched software in the Campus Ready e-Assessment platform, in accordance with ISO/IEC 27001:2022 guidelines?

1.2. Hypotheses

H1: Unpatched software during software upgrades significantly increases cybersecurity vulnerabilities in Campus Ready e-Assessment platform (Software) and it is making them more susceptible to attacks such as data breaches and malware infections.

H2: Implementing effective patch, vulnerability management and configuration management strategies, aligned with ISO/IEC 27001:2022 guidelines, is expected to significantly reduce

vulnerabilities and enhance the overall security risks posture of the Campus Ready e-Assessment platform.

1.5 Significance of the Study

The study is relevant in the manner in which it could be utilized in strengthening the security profile of e-Assessment systems such as Campus Ready. As there is more utilization of online platforms for undertaking assessments in educational institutions, there are higher possibilities of cyberattacks exploiting weaknesses in the platforms. Unpatched software is a critical vulnerability, providing entry points for every manner of cyber threat from data breaches, ransomware attacks, and system downtime. This research will offer keen insight into the way that unpatched software leads to such vulnerability, with practical suggestions for mitigation through proactive patch management, penetration testing and system configuration hardening (Kiennert et al., 2017).

Through its focus on the Campus Ready platform, the research will add to the body of research literature on the particular vulnerabilities of e-Assessment systems and the general cybersecurity ecosystem. It will offer institutions evidence-based guideline on how they can enhance their defenses, protect sensitive information and maintain continuity of their services. Additionally, this research will guide cybersecurity best practices for comparable platforms, providing recommendations that can be implemented by developers and administrators as a measure to protect user data and provide assurance in online examination systems (ISACA, n.d.; Souppaya et al., 2022).

During a period when cybersecurity threats are developing at a record pace, this research will be contributing to the general body of knowledge in maintaining e-Assessment systems secure so schools can remain a secure and stable place for teaching and learning for both teachers and students. These research suggestions can ultimately play a role in saving funds by preventing expensive cyber-attacks, remaining compliant with the regulators and building trust in digital education solutions (Dimitrios Sargiotis, 2024).

1.6 Structure of the Thesis

Introduction provides the research scope, aims and significance of cybersecurity in e-Assessment systems for the purpose of addressing vulnerabilities of unpatched software. Int roduction also gives the thesis structure.

Literature Review reviews existing research on cybersecurity challenges in digital platforms/e-Assessment platforms with a focus on unpatched software, vulnerabilities and mitigation strategies.

Methodology describes the research design and methods used, such including vulnerability scanning, penetration testing and risk analysis. It also outlines research design scope, data collection tools and techniques, data collection procedure, data preparation, data analysis, risk evaluation criteria, reliability and validity and strategy development.

Findings summarizes key vulnerabilities and their severity, along with recommended solutions for patch management and security improvements.

Implementation (Suggested Actionable Solutions) outlines strategies to address vulnerabilities in the Campus Ready e-Assessment platform using ISO/IEC 27001:2022 guidelines. It includes patch management, vulnerability scans, secure configurations and adherence to GDPR and ethical standards.

Validation of Research Questions & Hypothesis verifies how research questions and hypotheses align with the study findings. It is concerned with ensuring hypotheses derived are empirically based to determine the impact of unpatched software on cybersecurity vulnerabilities and efficacy of mitigation strategies based on ISO/IEC 27001:2022 guidelines.

Limitations & Future Research discusses the limitations of the study and recommends potential future research paths, such as extending analysis to other e-Assessment sites or enhancing security with automated patch management and higher-order cybersecurity systems.

Discussion & Conclusion brings together the study's purposes, methods, findings and recommendations in an attempt to emphasize the importance of proactive cybersecurity administration for e-Assessment sites.

2. LITERATURE REVIEW

Technological advancements are paramount to AI-based platforms since they enable them to manage student data and deliver course content. The fact that they rely on software, however, makes them target points for hackers and especially if the vulnerabilities go unpatched. Garg & Sengupta (2021) point out that unpatched software can result in service interruption, data theft and loss of user trust. To counter such threats, a software requires efficient patch management practices according to standards like ISO/IEC 27001:2022 (Proença & Borbinha, 2018).

2.1. E-Assessment Platforms & Cyber Security

Cybersecurity is the defense of information, networks and systems against cyber-attacks, unauthorized use and destruction. It is processes, technologies and practices implemented to guard information from malicious actions such as hacking, phishing, malware and data breaches. Cybersecurity is in place to ensure information is confidential, not damaged and accessible so cyber criminals cannot take advantage of loopholes in networks and computer systems. It is equally critical that individuals, organizations, and governments have secure space and safeguard confidential data in the increasing virtual world (Wei, 2023).

In contrast, e-Assessment platforms are some of the modern learning and delivering convenience and scalability and are digital systems installed to conduct, administer, and mark tests remotely. They allow efficient testing, grading and monitoring of performance to avail benefits such as time and space economy and global reach. However, their growing reliance on technology reflects the necessity for robust cybersecurity in ensuring the integrity of the tests and protecting them from cyber-attacks (Khlifi & El-Sabagh, 2017).

However, e-Assessment platforms face acute cybersecurity threats such as data leakage, ransomware and exploitation of unpatched vulnerabilities can compromise sensitive information and disrupt education processes. Other threats are boosted by the introduction of third-party tools and human errors. Due to this reason, active measures in the form of vulnerability scanning, penetration testing and strict security policies are essential in a bid to protect such platforms for maintaining academic integrity and secure virtual learning environments (Kiennert et al., 2017).

2.2. The Importance of Cybersecurity in E-Assessment Platforms

Cybersecurity is critical in protecting digital systems against threats that arise as a consequence of users' irresponsible behavior or system weaknesses. The threats have the potential to cause significant consequences, such as losses and reputational damage. It requires both practical and technical expertise to manage risks. Challenges during business digital transformation are limited resources, data privacy issues and talent shortages (Möller, 2023, pp. 30-45).

2.3. Unpatched Software

Dissanayake highlights that unpatched software is a major cybersecurity threat, particularly in the era of digital transformation (Software Installation & Upgrades). Untreated vulnerabilities can expose sensitive information, provide unauthorized access and disrupt key operations (Dissanayake et al., 2022). Research highlights that unpatched systems are crucial in cyberattacks that are demanding active measures for the protection of sensitive information and maintaining system integrity (Ponemon Institute, 2019). In e-Assessment settings, student records and course material leakage can lead to severe financial and reputational damages (Gebremeskel et al., 2023).

Ruan suggested that patch management is important in order to make digital assets secure. The process is one of discovery, gathering, testing and applying updates to mitigate vulnerabilities (Ruan et al., 2020). Patch management policies and automated tools make the process more efficient through reducing errors and making patch deployment standardized. The literature supports prioritizing high-risk vulnerabilities while less critical patches can be applied on a more flexible schedule (Jung et al., 2022).

Digital transformations (through software installation & updates) make patching even harder because companies are combining new technology with existing systems. Such complexity usually leaves unpatched software that is causing mounting cybersecurity risks. Alsubaie & Emam (2018) found that the attackers can exploit the vulnerabilities to have unauthorized access or disrupt operations and particularly during software upgrades. Therefore, Alsubaie & Emam says that having a strong patch management system is crucial to overcoming such challenges (Alsubaie & Emam, 2018).

According to ISACA information security framework, effective patch management is compliant with standards like ISO/IEC 27001:2022 which offers structured means of minimizing risk. The standard encourages proactive vulnerability management, ongoing monitoring and

prioritization based on the level of risk. With adherence to such standards, organizations are in a position to enhance their security position and reduce risk likelihoods of unpatched software (ISO/IEC, 2022).

Theoretical frameworks like the vulnerability management lifecycle and risk management frameworks provide conceptual foundations for understanding how patch management mitigates exploitation risk. The vulnerability management lifecycle highlights stages from vulnerability discovery to remediation and validation (Scarfone & Mell, 2010). Furthermore, risk management frameworks highlight the importance of taking into account and managing risks in an organized way to safeguard assets and maintain operating integrity (Dempsey et al., 2011).

Although literatures provide information on patch management, there are still areas that are not addressed. Not much is known about the effectiveness of patch prioritization techniques and automation of patch management, especially for e-Assessment platforms undergoing digital upgrades. Further research is needed to explore these areas and particularly in practice alignment with ISO/IEC 27001:2022 to develop effective measures for vulnerability management in e-Assessment platforms (Dempsey et al., 2011).

2.4. Risks Posed by Unpatched Software

Unpatched systems pose a significant threat to e-Assessment applications. The attackers will utilize outdated systems for unauthorized access to sensitive information such as grades and students' information and can generate the risk of data breach, as well as serious reputational loss (Habib et al., 2020, pp. 1-5; Ching-Huang et al., 2008, p. 1252). Unpatched devices are at risk of several forms of cyberattacks, such as having malware infections, data breaches and ransomware, which can greatly compromise the security of the platform (Dennis, 2018).

2.5. The Vulnerability Lifecycle and Patch Management

Vulnerability lifecycle, or the vulnerability "window of vulnerability" highlights the importance of patching in a timely fashion in order to limit exposure to cyberattacks. The lifecycle begins when the flaw is first introduced and reports of the risk rise as the vulnerability is found and made public. The gap between discovery of the vulnerability and patch installation is a vulnerable window and timely patching is paramount to remain one step ahead of exploitation (Brykczynski & Small, 2021, p. 51).

2.6. Challenges of Patching Systems

One of the key challenges of system security assurance is timely and consistent patching, particularly in sophisticated digital environments with legacy systems. Insufficient resources, limited IT staff and concerns about system incompatibility are typically reasons for patching delays which is leaving the platform more vulnerable to cyber-attacks (Craig S. Wright, 2008, pp. 471-472). Also, the apprehension of downtime due to system failures or compatibility issues is common and holds up critical updates (Dissanayake et al., 2022, pp. 11-12).

2.7. Organizational and Human Factors in Cybersecurity

Unpatched software tends to be that way due to a combination of organizational inefficiencies, human error and technical problems. Lack of awareness, poor training and ineffective patch management policies are some of the reasons that lead to delayed or unsuccessful application of patches (Dissanayake et al., 2022, pp. 1-12). Complexity in IT system management also contributes to the challenge in patching, especially in environments with old or unsupported legacy systems (Pulliainen, 2016, pp. 4-25).

2.8. Automating Patch Management and Prioritization

Automating patch management is possible to speed up the process and install updates in a timely and periodic manner. Automation, however, has its own drawbacks, and one of them is the risk of administrators installing updates that might compromise system functionality. Prioritization of the patches based on the severity of the vulnerabilities they are meant to close is also needed so that the most serious threats are addressed first (Tom et al., 2008, pp. 6-14). The period between the release of a patch and its deployment that has been referred to as the "patch gap" is a window of exposure during which systems are susceptible to attacks (Dennis, 2018).

2.9. Configuration Management and Security

Correct configuration management is important to ensure the security of e-Assessment platforms. It provides that systems and components are securely configured to avoid unauthorized access. But security settings tend to conflict with the working requirements of the platform which is turning out to be difficult to implement security and functionality in balance. This is an issue which is very cutting in multi-IT infrastructure or legacy environment configurations (Pulliainen, 2016, pp. 61-65).

2.10. Proactive Security Strategies

Given the increase in cyberattacks and IT sophistication, proactive security features need to be deployed. These include monitoring in real time, continuous vulnerability scan and adherence to industrial standards like ISO/IEC 27001:2022. Such initiatives keep organizations ahead of the emerging threats and make the platform resilient to possible cyberattacks (Kersten & Schröder, 2023).

3. METHODOLOGY

This chapter defines the research design and methodology utilized in this case study of the Campus Ready e-Assessment system. It starts with giving an overall research design overview and explaining the rationale for following a "Quantitative Research" design (Mohajan, 2021). The approach has been utilized with the expectation that objective and measurable data shall be gathered with statistical analysis of determining the platform's security vulnerabilities conclusively. The aim is to present an evidence-based and systematic report on the security status of the platform using measurable risk indicators.

Beneath this case study, "Correlation Heatmap Analysis" is employed in order to discover correlations among numerical vulnerability dataset variables to identify patterns and high correlations (Rabzelj et al., 2020).. The method assists in identifying how various indicators, including risk classes and severity scores, are related to one another and can be employed to detect redundant variables in an attempt to reduce the complexity of the analysis (Rabzelj et al., 2020). Additionally, Root Cause Analysis (RCA) is applied to investigate the root true causes of security vulnerabilities within the Campus Ready (Zhou et al., 2007).

RCA extends beyond symptoms since it determines root causes, i.e., resource constraint or process inefficiencies, or misconfigurations that are yielding information on long-term solutions. All these methods in combination, thus, can provide a comprehensive, evidence-based examination in an attempt to systematically determine vulnerabilities, their cause and possible remedies, all within the context of this case study of the Campus Ready platform (Zhou et al., 2007).

3.1. Research Context

Impact of unpatched software on cybersecurity in Campus Ready e-Assessment platform during digital transformation (software installation & upgrades)

Campus Ready is one area of focus for this study and it is an AI-powered e-Assessment solution designed to automate university entrance testing. Its platform is riddled with instant cybersecurity risks, and the problem involves unpached software that can reveal wide vulnerabilities. Campus Ready is a startup company developing its services based on sensitive academic data, all under the strain of maintaining data intact and within stringent cybersecurity requirements (Gebremeskel et al., 2023; Dissanayake et al., 2021). This research setting is significant in the sense that it portrays the fine line between adopting the newest AI technologies and staying abreast with the risks associated with cybersecurity, especially software installations & updates. System updates always reveal new security vulnerabilities so much so that identifying and handling probable threats in real time becomes indispensable (Gebremeskel et al., 2023; Dissanayake et al., 2021).

Through examining such vulnerabilities in the Campus Ready platform, the research seeks to provide actionable understanding of how its security system can be enhanced. Not only will the findings improve the security posture of the platform, but also user trust and confidence. In general, this research helps Campus Ready to keep expanding and developing in the education technology space while meeting high cybersecurity standards (Gebremeskel et al., 2023; Dissanayake et al., 2021).

3.2. Research Design

This research is founded on a case study of the Campus Ready e-Assessment platform. "Quantitative Research" study design with the aim of giving a general evaluation of the current security state of the Campus Ready e-Assessment platform. Quantitative Research study design will be selected since it enables detailed investigation of the Campus Ready e-Assessment platform's singular infrastructure and vulnerabilities. Quantitative methodology is also most suitable for this research with statistical analysis and graphs because it emphasizes the gathering and analysis of numerical data to enable simple, objective and quantifiable conclusions. With the use of statistical software, the methodology enables researchers to discover trends, patterns and relationships in the data hence most suitable to examine such as vulnerability numbers, severity scores or risk scores (Mohajan, 2021).

Besides, quantitative research enables hypothesis testing and generalizability of findings which are extremely important when developing actionable recommendations and making

meaningful conclusions. Its systematic methodological process guarantees reliability and validity and hence is the most desirable option when dealing with numeric data and evidence-based conclusions are needed (Mohajan, 2021).

The study will utilize a quantitative method in examining the data gathered with the aim of comprehending the frequency and effect of unpatched software vulnerabilities on the security posture of the e-Assessment platform. The first step "Descriptive statistics" will be the first process in data analysis to summarize the data and provide an insight into its central tendency and distribution (Lim, 2024). Use of graphs and charts, such as bar graphs, pie charts or histograms, also enhances better clarity and visual representation of complex data and simpler communication of findings (Lim, 2024; Mohajan, 2021).

This methodology facilitates a comprehensive comprehension of the existing vulnerabilities and possible hazards linked to software that is not patched and hence aiding in the formulation of targeted and implementable suggestions. There are six main phases to the research process (Souppaya et al., 2022):

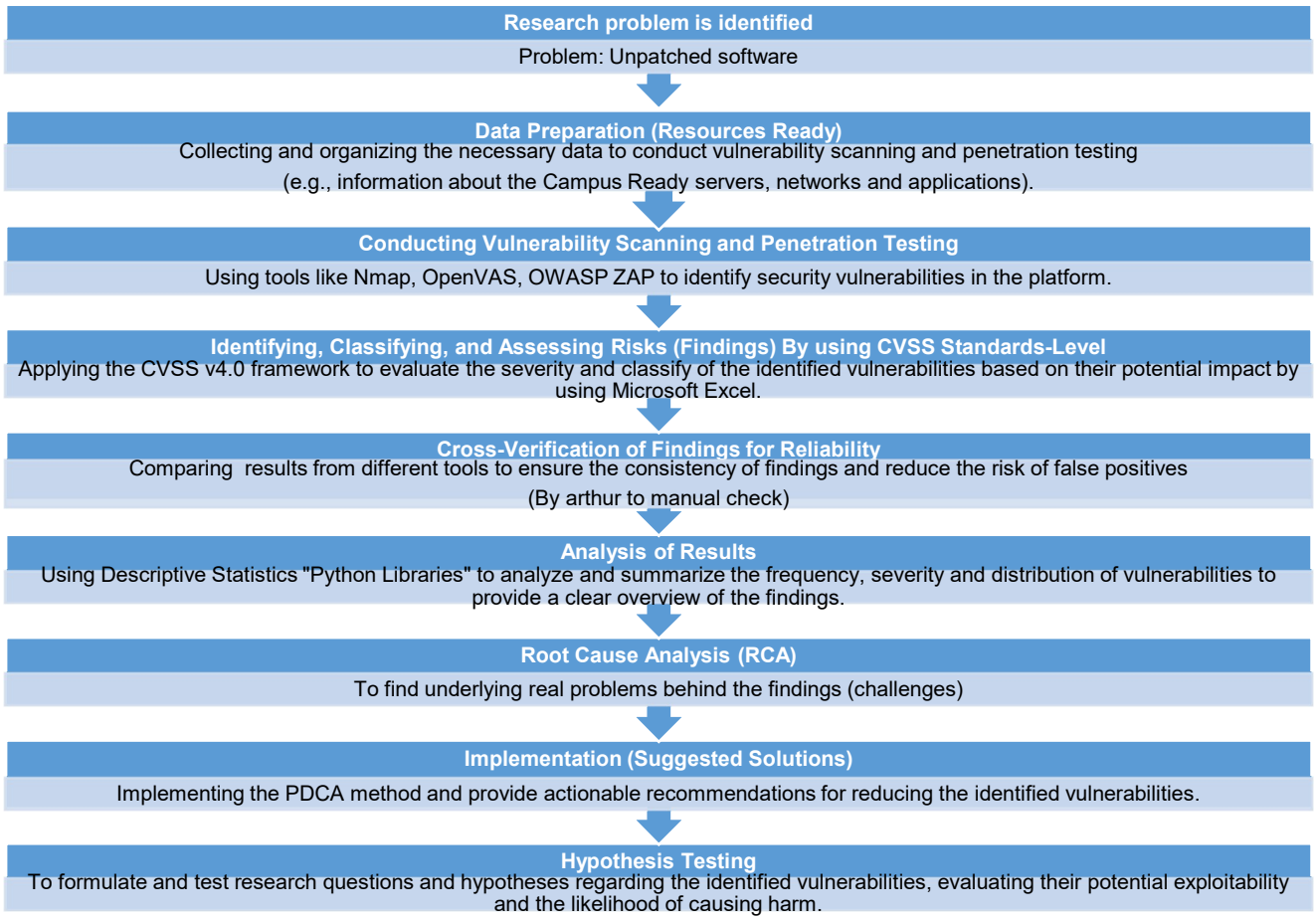


Figure 1: Processes of Research Design

Source (s): Arthur`s own work

3.2.1. Scope: Case Study: Campus Ready GmbH – Startup

The study will analyze the entire Campus Ready e-Assessment platform. Under-going Campus Ready e-Assessment platform: https://campus-ready.com/

3.2.2. Campus Ready GmbH Overview

Campus Ready is a start-up based at the University of Applied Sciences HNU in Neu-Ulm, Germany, is an innovative digital e-Assessment platform designed to streamline the admissions process for German and European universities and they are focusing on international students from non-EU countries. Traditional admissions methods, reliant on time-consuming in-person interviews, often leave study spaces unfilled despite high demand. To address this, Campus Ready offers a scalable and customizable e-Assessment platform featuring AI-driven aptitude tests that efficiently evaluate applications in order to handle thousands of candidates simultaneously. The platform stands out by tailoring exams to each applicant's academic background, ensuring fair evaluations aligned with diverse assessment systems (Provided by Campus Ready, n.d.).

In addition, Campus Ready provides preparatory materials to support students, promoting equitable access to higher education. Campus Readys mission aligns with the UN's Sustainable Development Goals, particularly SDG 4 (Quality Education), SDG 9 (Innovation)and SDG 10 (Reduced Inequalities). Their future plans include expanding to other European countries, such as France and Spain while continuing to enhance inclusivity and innovation in admissions (Provided by Campus Ready, n.d.).

Campus Ready's Departments and Positions:
The organizational structure of Campus Ready comprises four key departments, each playing a crucial role in the platform's success. The Design Department focuses on user-centered design with the Product Designer enhancing user experience. The Marketing and Sales Department is led by the Marketing and Sales Manager who drives strategic outreach, marketing campaigns and user engagement. In the Product Development Department, the Product Manager oversees the product lifecycle to ensure alignment with user needs and market trends. Lastly, the Technical Operations Department is managed by the Technical Manager who ensures the platform's technical efficiency and robustness (Campus Ready, n.d.).

3.3. Data Collection Tools and Techniques

Comprehensive vulnerability evaluations utilizing industry standard cybersecurity tools will be used to collect data by using penetration testing and vulnerability scanning techniques. Since Vulnerability scanning is an automated process that identifies known vulnerabilities in a system, such as outdated software or misconfigurations, and provides a list of issues ranked by severity for remediation. It is non-intrusive, cost-effective, and conducted regularly as part of routine security maintenance. In contrast, penetration testing is a manual and more in-depth process where ethical hackers simulate real-world attacks to exploit vulnerabilities, uncover complex risks and assess the actual impact of a breach. While vulnerability scanning focuses on identifying potential issues, penetration testing evaluates how attackers could exploit those vulnerabilities in practice to provide a deeper understanding of security risks (Abdulghaffar et al., 2023).

Utilizing a variety of methods will guarantee a comprehensive assessment of the Campus Ready e-Assessment platform and addressing vulnerabilities related to the application also those at the infrastructure level. The studys chosen instruments are as showed in the table 1 at below (Mell et al., 2005):

Table 1: Data Collection Tools & Techniques

Tools	Phase	Purposes	Key Features / Functions
Nmap	Asset Discovery	Identifying network assets and mapping network topology	Performs host discovery, service detection, and port scanning. Identifies misconfigurations and vulnerabilities in networked systems. Offers scripting for automated scan etc.
OpenVAS	Vulnerability Assessment	Detecting and assessing vulnerabilities across networked systems	Conducts comprehensive scans for vulnerabilities, provides detailed reporting and prioritization, integrates with third-party tools for risk assessment etc.
OWASP ZAP	Penetration testing Threat modelling	Dynamic application security testing tool used for finding security vulnerabilities in web applications	Acts as a proxy server, manages traffic including HTTPS, operates in daemon mode, supports REST-based API etc.
ISO/IEC 27001:2022	risk assessment	International standard for information security management systems (ISMS)	Provides a structured framework for defining, assessing and reduce information security risks. Aligns security practices with business objectives.

Source (s): (Shahid et al., 2022).

Four different cybersecurity tools Nmap, OpenVAS, OWASP ZAP are used in the research methodology to find and examine vulnerabilities in the e Campus Ready e-Assessment platform (Shahid et al., 2022).

The goal of vulnerability management is to locate and fix IT system flaws Campus Ready's e-Assessment platform. To check for known vulnerabilities in the system and administrators need to utilize tool like OpenVAS (Ching-Huang, et al.,2008, pp.1252-1253) & (Pulliainen, 2016, pp.5-20).

The platform's server and web infrastructure will be thoroughly scanned using OpenVAS to identify known vulnerabilities, misconfigurations and unpatched software components (Shahid et al., 2022). OpenVAS produces detailed reports that classify vulnerabilities by priority with a focus on unpatched software vulnerabilities that are catastrophic security risks. OpenVAS is an open-source, full-featured vulnerability scanning tool focused on network security scanning. OpenVAS boasts an impressive list of features, headed by a regularly updated feed of vulnerability tests so the latest threats can be added. With its inclusion within the platform ecosystem, OpenVAS is capable of conducting thorough scans and delivering actionable intelligence to further enhance the security position of the platform (Greenbone OpenVAS, n.d.; Sharma et al., 2024).

The technique can be utilized to identify a range of security vulnerabilities within the operating system or services running on the host. OpenVAS employs a modular architecture and taps into a massive repository of network vulnerability tests (NVTs) that are regularly updated to counter emerging threats. NVTs can also be edited and tailored for the target environment to ensure accurate detection and vulnerability analysis. Scan configuration, scanning results, and triggering some of the mitigation processes can be performed by administrators via the Greenbone Security Assistant in order to secure the system effectively (Greenbone OpenVAS, n.d.; Sharma et al., 2024).

Web application layer will be tested using "OWASP ZAP" which will detect vulnerabilities such as SQL Injection and Cross-Site Scripting (XSS). These tools will generate reports on vulnerabilities that contain information on inputs that can potentially be used and types of attacks that can potentially be conducted via buggy configurations or unpatched software. Apache License controls usage of ZAP (Zed Attack Proxy), a web application security scanner by scanning them upon running them. As a proxy server, it gives the user the ability to filter

any traffic, including HTTPS encrypted communication. It can also be executed in daemon mode, which is managed by a REST-based API (Shahid et al., 2022, pp.9; Abdulghaffar et al., 2023).

Lastly, Nmap (Network Mapper) is an open-source security tool for scanning networks and testing their security. Gordon Lyon developed Nmap, which is widely employed to scan network devices, host or service availability monitoring and management. Nmap can scan large networks but also supports scanning on a single host. The utility supports OS identification, service version identification and scriptable interaction with the target using the Nmap Scripting Engine (NSE). Nmap's flexibility and capacity to scan for vulnerabilities in network setup make it a necessary tool for administrators wanting to increase security. It supports various operating systems, such as including Linux, Unix, Windows and macOS, and comes pre-installed in most penetration testing distributions (Nmap, 2024; Kang et al., 2021).

Nmap's scanning and testing tool usage, including OWASP ZAP and OpenVAS, presents valuable advantage to Campus Ready. By detecting and patching vulnerabilities, the software fortifies the security position of the platform overall to provide effective defenses against future cyber attacks. This proactive approach will enable industry standards like ISO/IEC 27001 compliance to fulfill regulatory needs for maintaining credibility (Tenable, 2023; Flavio Rodrigues Pereira et al., 2023; Greeshma Swapnika Manubolu., 2024). Ensuring sensitive applicant and institution data protects user trust and stakeholder trust, crucial for an education platform handling sensitive education data.

Secondly, earlier detection of vulnerabilities and mitigation cut the cost of repairing security post-deployment, driving up operational costs. Finally, they enable the platform to scale effectively enough that it can support growing numbers of users without compromising performance or security (Flavio Rodrigues Pereira et al., 2023; Greeshma Swapnika Manubolu., 2024). Microsoft Excel for data structuring and cleaning and Python libraries such as Pandas and Matplotlib will be utilized. Since they can facilitate efficient data visualization and statistical analysis to assist in proper classification and prioritization of vulnerabilities (Khadka, 2019).

3.4. Data Collection Procedure

In this phase, security testing using the selected tools will be conducted to search for vulnerabilities. Manual penetration testing along with automated vulnerability scanning will be performed. The tools discussed in the previous phase i.e. "Data Collection Tools and Techniques "will be used to scan for known vulnerabilities, misconfigurations and security weaknesses, with manual testing thereafter to exploit and ascertain defences' effectiveness.

There are five steps that will be included in the data collection process to make it a comprehensive and systematic process of discovering and reporting vulnerabilities in the Campus Ready e-Assessment system:

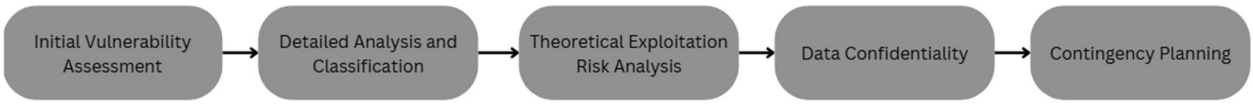


Figure 2: Data Collection Procedure Steps

Source (s): Arthur`s own work

1.Initial Vulnerability Assessment

Using scanners Nmap, OWASP ZAP and OpenVAS a tentative initial sweep of the infrastructure and application aspects of the Campus Ready e-Assessment system will be performed during this phase. The primary aim is to create a baseline of vulnerabilities with special interest in those caused by unpatched software. Each weakness found will be recorded along with details about how severe it is and which affected components and solutions are proposed (Shahid et al., 2022; (Kotronoulas et al., 2023).

In addition to simplification and conformity facilitation while doing data analysis, the acquired scan and test tool data (i.e., Nmap, OWASP ZAP, and OpenVAS) will be stored methodically inside an Excel spreadsheet. Since the structured data will facilitate easier structuring, visualization, and prioritization of vulnerabilities (Shahid et al., 2022; (Kotronoulas et al., 2023).

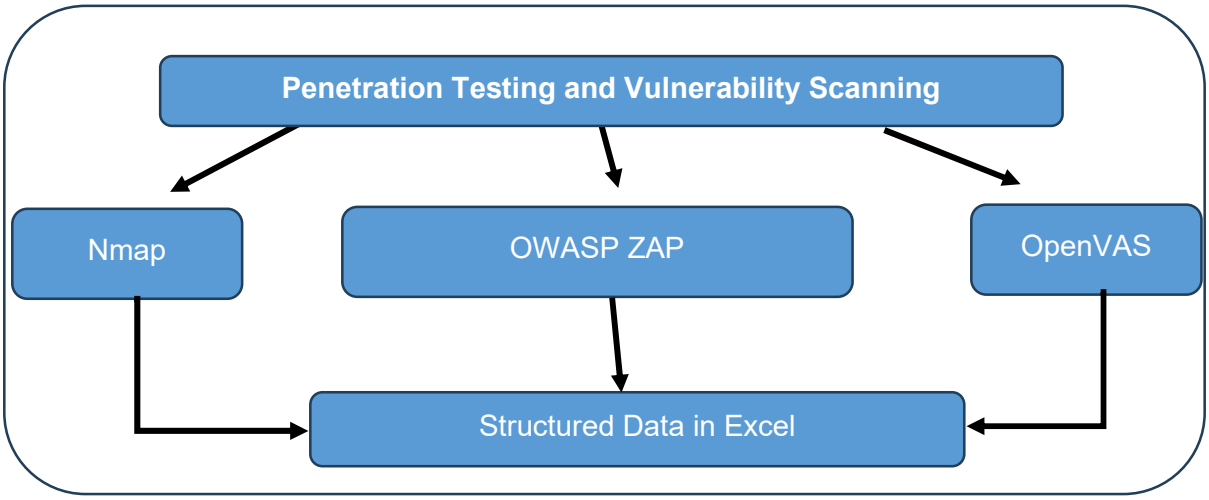


Figure 3: Data Collection Method

Source (s): Arthur`s own work

2.Detailed Analysis and Classification

Following the initial assessment, the vulnerabilities will undergo a more detailed classification. Each vulnerability will be categorized based on severity and potential impact. The platforms' software and system configurations will be scrutinized to determine whether they are vulnerable to known threats, particularly those caused by unpatched software. The analysis will also focus on identifying patterns across the platforms various components (e.g., web applications, network infrastructure and databases).

3.Theoretical Exploitation Risk Analysis

A non-intrusive risk analysis will be carried out using OpenVAS to simulate potential exploits based on the vulnerabilities discovered. This analysis will help assess the practical risks posed by unpatched vulnerabilities, without actual exploitation. The goal is to determine the potential harm that could arise from these vulnerabilities if they were exploited by a cyber attacker. This theoretical risk analysis will help prioritize which vulnerabilities need to be addressed first (Shahid et al., 2022).

4.Data Confidentiality

During data collection, all data collected during vulnerability assessments will be securely stored in encrypted formats. Access to the data will be restricted to authorized personnel involved in the study. To ensure compliance with data protection regulations, such as GDPR

and any personally identifiable information (PII) encountered during the assessments will be excluded from analysis (Dimitrios Sargiotis, 2024).

5.Contingency Planning

In order to address false positives or tool-specific limitations, the results from each scanning tool will be cross verified using manual work in an Excel sheet (By Arthur). Discrepancies will be manually analysed to determine the validity of the identified vulnerabilities. Thus, this will ensure that only accurate, actionable findings will inform the recommendations.

3.5. Data Preparation for Analysis

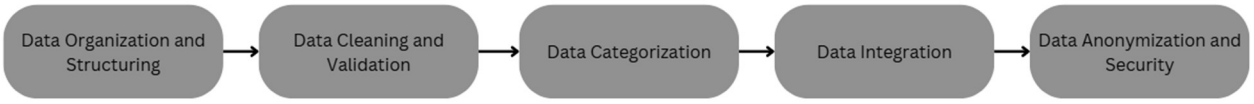


Figure 4: Data Preparation Steps

Source (s): Arthur`s own work

This phase involves gathering information on the system architecture of the Campus Ready platform, determining the scope of testing, and setting up the necessary tools and frameworks to ensure that the testing phase is aligned with the platform's specific configurations and infrastructure.

After data collection, the data will be prepared by taken these steps at below “Figure:6 Data Preparation Steps” to next data analysis step:

Data Organization and Structuring

All vulnerability data from the various tools (Nmap, OpenVAS, OWASP ZAP) will be organized into a structured format (Shahid et al., 2022, pp. 9–10). This may involve categorizing vulnerabilities based on their type (e.g., network vulnerabilities, application vulnerabilities), severity (e.g., critical, high, medium and low) and affected components (e.g., server, application layer, database) (NVD – NIST Home, n.d.) & (Mell et al., 2005, p. 3-2). The collected data will be thoroughly structured remove any inconsistencies or erroneous results by using Microsoft Excel (Kotronoulas et al., 2023).

Data Cleaning and Validation

The collected data will be thoroughly cleaned manually to remove any inconsistencies or erroneous results by using Microsoft Excel. For instance, any false positives or incomplete data will be flagged and verified using additional manual checks. Vulnerabilities identified by Arthur for cross-check to ensure the accuracy (Ridzuan & Zainon, 2019). The first phase of

analysis involves importing the dataset into a Pandas DataFrame which facilitates efficient data manipulation and organization. During this stage, the dataset is thoroughly cleaned to address any missing or inconsistent values to ensure the data is accurate and ready for analysis. Since, cleaning the data is a crucial step to avoid errors in subsequent analyses and to maintain the integrity of results (Setiyanto & Setiawan, 2022).

Data Categorization

After cleaning, the vulnerabilities will be categorized based on their severity and risk level. Each vulnerability will be classified according to the Common Vulnerability Scoring System (CVSS v4.0) to prioritize them, “Table 2: CVSS v4.0 (Version 4.0) Ratings”. Critical and high-severity vulnerabilities will be identified for immediate attention while medium and low-severity vulnerabilities will be handled in the long term NVD – NIST Home, n.d.) & (Mell et al., 2005, p. 3-2).

The table at below provides a detailed „Structure of Data Categorization” type of cyber security vulnerabilities from an unpatched software of Campus Ready will be identified during a security scan and testing. Each row represents a unique vulnerability in servers, networks, websites and applications. Campus Ready e-Assessment platform itself quietly new online platform and it is still undergoing development process; therefore, the results of Vulnerability scans and penetration testing will be enough to define the cyber security vulnerabilities such as misconfigurations caused by unpatched software. Therefore, only those type of vulnerabilities mentioned in the below **Table 1: Structure of Data Categorization for Data Analysis** going to be analysed to detecting an unpatched software: is a breakdown of each column in the table:

Vul. ID: A unique identifier assigned to each vulnerability, used for tracking purposes.
Type of Vulnerability: A brief description of the specific vulnerability discovered. This could range from missing security headers to insecure cookie settings.
Severity Score: A numerical score representing the severity of the vulnerability. This score typically follows a range based on the CVSS (Common Vulnerability Scoring System). Higher scores indicate more critical vulnerabilities on Campus Ready e-Assessment Platform.
Severity Level: Categorizes the severity of the vulnerability into levels: None, Low, Medium, High or Critical. These categories assist in prioritizing vulnerabilities according to their potential impact on the Campus Ready e-Assessment Platform.
Affected Software/System: Identifies the software, system, or component where the vulnerability was found. For example, "Web Application" or "Web Server".

Impact Scope: Defines the potential scope of the vulnerability's impact, such as whether it affects the entire system, a specific component, or a limited feature.

Patch Availability (Y/N): Indicates whether a patch is available for the identified vulnerability. "Available" means a fix has been identified and can be implemented.

Discovery Date: The date when the vulnerability was first discovered during the scan and testing.

Detection Source: The tool used to identify the vulnerability. Common tools include OWASP ZAP, OpenVAS and Nmap.

Table 1: Structure of Data Categorization for Data Analysis

Vul. ID	Type of Vulnerability	Severity Score	Severity Level	Affected Software/System	Impact Scope	Patch Possible?	Discovery Date	Detection Source
1	Absence of Anti-CSRF Tokens	0-10	X	Web Application	Potential CSRF attacks	YES/NO	X	OWASP ZAP
2	Missing Anti-clickjacking Header	0-10	X	Web Application	UI redirection vulnerability	YES/NO	X	OWASP ZAP
3	Content Security Policy (CSP) Header Not Set	0-10	X	Web Application	Cross-site scripting risks	YES/NO	X	OWASP ZAP
4	X-Content-Type-Options Header Missing	0-10	X	Web Application	Risk of MIME-type sniffing	YES/NO	X	Nmap
5	Cross-Domain JavaScript Source File Inclusion	0-10	X	Web Application	Potential data exposure	YES/NO	X	OWASP ZAP
6	Server Leaks Version Information via 'Server' Header	0-10	X	Web Server	Information disclosure	YES/NO	X	OWASP ZAP
7	Strict-Transport-Security Header Not Set	0-10	X	Web Application	Risk of downgrade attacks	YES/NO	X	Nmap
8	TCP Timestamps Information Disclosure	0-10	X	Network Infrastructure	Uptime information leakage	YES/NO	X	OpenVAS
9	Weak MAC Algorithm(s) Supported (SSH)	0-10	X	SSH Server	Potential data integrity issue	YES/NO	X	OWASP ZAP
10	Open TCP Port: 22	0-10	X	Web Server	Unsecured access point	YES/NO	X	Nmap
11	Open TCP Port: 443	0-10	X	Web Server	Potential unauthorized access	YES/NO	X	Nmap
12	Open TCP Port: 80	0-10	X	Web Server	Potential unauthorized access	YES/NO	X	Nmap
13	Insecure Cookie Setting: Missing Secure Flag	0-10	X	Web Application	Risk of session hijacking	YES/NO	X	OWASP ZAP
14	Missing security header: Referrer-Policy	0-10	X	Web Application	Information leakage via referrers	YES/NO	X	Nmap
15	Server Software and Technology Found	0-10	X	Web Server	Information disclosure	YES/NO	X	OpenVAS
16	Security.txt file is missing	0-10	X	Web Server	Missing security contact details	YES/NO	X	OWASP ZAP
17	HTTP OPTIONS Enabled	0-10	X	Web Server	Potential method discovery attack	YES/NO	X	Nmap

Source (s): Arthur`s own work

The Structure of Data Categorization for Data Analysis will be crucial for effectively analyzing the vulnerabilities in the Campus Ready e-Assessment platform, especially concerning unpatched software. By grouping information into clearly defined categories, such as Vul. ID, Type of Vulnerability, Severity Score, Severity Level, Affected Software/System, Impact Scope, Patch Availability, Discovery Date, Detection Source and Risk Score, vulnerability analysis of unpatched software becomes more organized and insightful. This categorization will allow for an organized approach to understanding and remediating vulnerabilities, with it being easy to prioritize which vulnerabilities are most critical based on their severity, impact, available patches and risk score. For instance, the categorization of vulnerabilities based on severity score and level allows one to ascertain which issues need to be addressed immediately, while information on affected software/system and impact scope helps to understand where security must be strengthened.

The Risk Score, as a product of impact and severity, helps to prioritize the vulnerabilities by their potential risk, so that the team can address high-risk vulnerabilities first. Additionally, patch availability being incorporated signifies that the analysis also considers whether remediation possibilities exist. Tracking the discovery date and detection source also helps a timeline of when and by what means vulnerabilities are discovered which can prove helpful for making updates and improvements in a timely manner. This structured approach not only keeps the analysis in order but also helps to make effective decisions for enhancing Campus Ready's cybersecurity posture and remediating the vulnerabilities effectively.

Table 2: An example of Structure of Data Categorization for Data Analysis:

Vul. ID	Type of Vulnerability	Severity Score	Severity Level	Affected Software/ System	Impact Scope	Patch Possible?	Discovery Date	Detection Source
1	Code Injection	8,5	High	Internal CRM	Integrity	Yes	2024-12-01	OpenVAS

Source (s): Arthur`s own work

Data Integration

Data from various tools (mentioned in the Data Collection Tools and Techniques part) and stages of the process will be integrated into a single database like Microsoft Excel for easier access and structure of the data which is already mentioned the the chapter “Initial Vulnerability Assessment”. Tools like Python (with libraries such as Pandas and Matplotlib) will be used for data visualization, statistical analysis and identification of patterns across the vulnerabilities (Khadka, 2019).

Data Anonymization and Security

To keep the data safe and private, any sensitive information (such as IP addresses, usernames or other personally identifiable details) will be anonymized. Only the necessary metadata related to the vulnerabilities will be retained for analysis (Dimitrios Sargiotis, 2024).

3.6. Data Analysis

Descriptive Statistics

The study will utilize a quantitative methodology to examine the gathered data with the aim of comprehending the frequency and consequences of unpatched software vulnerabilities on the security posture of the e-Assessment platform. The first step "Descriptive statistics" will be the first step in data analysis to summarize the data and provide information on its central tendencies and distribution (Lim, 2024).

Severity and Impact Analysis

A severity and impact analysis will then prioritize vulnerabilities according to their potential impact. Further, the analysis will rate vulnerabilities according to their severity levels based on the Common Vulnerability Scoring System (CVSS v4.0). High and critical vulnerabilities will be dealt with on priority basis first as these are the biggest threats to the security of the platform. Medium and low vulnerabilities will be documented with the suggestions for mitigation longer term compass (NVD – NIST Home, n.d.) & (Mell et al., 2005, p. 3-2).

Penetration Testing and Vulnerability Scanning

This statistical summary will give the overall security stance of the platform, identifying the weakest points. After that, a "severity and impact analysis" will determine each vulnerability's potential impact on the platform's functionality and data integrity. It will prioritize vulnerabilities by criticality and impacted component so the most severe risks are addressed first. Network scanning and vulnerability testing will also be conducted using Nmap, a software designed to discover open ports, detect running services, and map network configurations. By simulating real-world attack scenarios, Nmap helps to evaluate real-world threats associated with the identified vulnerabilities. The step gives deeper insight into system weaknesses, enabling the platform to prioritize remediation effectively (Nmap, 2024; Kang et al., 2021).

Data Visualization

Afterward, data analysis will be done using Python libraries such as Pandas and Matplotlib (Khadka, 2019). These libraries will enable proper data manipulation, statistical analysis and visualization to ensure accurate classification and prioritization of vulnerabilities. For rendering the data more comprehensible, visualization techniques with Matplotlib and Seaborn such as bar and pie charts will be employed to represent categorical data such as types of vulnerabilities and their severity levels, while heatmaps will be employed in representing relationships among over two variables. Visualizations will help in presenting complex datasets in a simple way and in an understandable manner and in allowing deeper insights into the vulnerabilities (Khadka, 2019).

They will be listed all vulnerabilities found and classified them according to severity (for example: critical, high, medium, low and none) as in Detailed Analysis and Classification with Python libraries as at shown at below/next page Figure 5 and to illustrate how these vulnerabilities are spread throughout the platforms many components such as the application layer, network layer or database (Khadka, 2019; NVD – NIST Home, n.d.).

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Create a DataFrame for the given data
data = {
    "Vul. ID": [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18],
    "Type of Vulnerability": [
        "Absence of Anti-CSRF Tokens", "Missing Anti-clickjacking Header", "Content Security Policy (CSP) Header Not Set",
        "X-Content-Type-Options Header Missing", "Cross-Domain JavaScript Source File Inclusion",
        "Server Leaks Version Information via 'Server' Header", "Strict-Transport-Security Header Not Set",
        "TCP Timestamps Information Disclosure", "Weak MAC Algorithm(s) Supported (SSH)",
        "Open TCP Port: 22", "Open TCP Port: 443", "Open TCP Port: 80",
        "Insecure Cookie Setting: Missing Secure Flag", "Missing security header: Referrer-Policy",
        "Server Software and Technology Found", "Security.txt file is missing",
        "HTTP OPTIONS Enabled", "Cookie without SameSite Attribute"],
    "Severity Score": [5, 5, 7.5, 2, 2.5, 3, 5.5, 2.6, 2.6, 5, 3.5, 3, 2.5, 5, 2.5, 5, 5, 2],
    "Severity Level": ["Medium", "Medium", "High", "Low", "Low", "Low", "Medium", "Low", "Low", "Medium", "Low",
                       "Low", "Low", "Medium", "Low", "Medium", "Medium", "Low"],
    "Affected Software/ System": [
        "Web Application", "Web Application", "Web Application", "Web Application", "Web Application",
        "Web Server", "Web Application", "Network Infrastructure", "SSH Server",
        "Web Server", "Web Server", "Web Server", "Web Application", "Web Application",
        "Web Server", "Web Server", "Web Server", "Web Server"],
    "Impact Scope": [
        "Potential CSRF attacks", "UI redirection vulnerability", "Cross-site scripting risks",
        "Risk of MIME-type sniffing", "Potential data exposure", "Information disclosure",
        "Risk of downgrade attacks", "Uptime information leakage", "Potential data integrity issue",
        "Unsecured access point", "Potential unauthorized access", "Potential unauthorized access",
        "Risk of session hijacking", "Information leakage via referrers", "Information disclosure",
        "Missing security contact details", "Potential method discovery attack",
        "A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' r
    ],
    "Patch Possible?": ["YES"] * 18,
    "Discovery Date": ["20.01.2025"] * 18,
    "Detection Source": [
        "OWASP ZAP", "OWASP ZAP", "OWASP ZAP", "Nmap", "OWASP ZAP", "OWASP ZAP", "Nmap",
        "OpenVAS", "OWASP ZAP", "Nmap", "Nmap", "Nmap", "OWASP ZAP", "Nmap", "OpenVAS",
        "OWASP ZAP", "Nmap", "OWASP ZAP"]}

df = pd.DataFrame(data)

# Display the first few rows of the dataset
print("Preview of the data:")
print(df.head())

# Example visualization 1: Bar plot of vulnerabilities by Severity Level
sns.countplot(data=df, x='Severity Level', palette='viridis')
plt.title("Count of Vulnerabilities by Severity Level")
plt.xlabel("Severity Level")
plt.ylabel("Count")
plt.show()

# Example visualization 2: Bar plot of vulnerabilities by Detection Source
sns.countplot(data=df, y='Detection Source', palette='coolwarm')
plt.title("Count of Vulnerabilities by Detection Source")
plt.xlabel("Count")
plt.ylabel("Detection Source")
plt.show()

# Example visualization 3: Severity Score distribution
sns.histplot(df['Severity Score'], kde=True, bins=10, color='blue')
plt.title("Distribution of Severity Scores")
plt.xlabel("Severity Score")
plt.ylabel("Frequency")
plt.show()

# Example visualization 4: Box plot of Severity Score by Severity Level
sns.boxplot(data=df, x='Severity Level', y='Severity Score', palette='Set2')
plt.title("Severity Score by Severity Level")
plt.xlabel("Severity Level")
plt.ylabel("Severity Score")
plt.show()

# Example visualization 5: Correlation heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap="coolwarm", fmt='.2f')
plt.title("Correlation Heatmap of Numeric Data")
plt.show()
```

Figure 5: Data Visualizations (Structure and Method) in Python for Data Analysis

Source (s): Arthur`s own work

Root Cause Analysis

The next step will involve a "root cause analysis" will find underlying real problems behind the findings, such as lack of budget of resources and investigate the reasons why updates were not implemented to go deeper into the causes of these vulnerabilities. The purpose of this analysis is to identify any systemic weaknesses in the Campus Ready regarding patch management procedures that gave rise to the vulnerabilities (Zhou et al., 2007).

Figure 6: Root Cause Analysis Steps

Source (s): (Zhou et al., 2007).

Correlation Heatmap Analysis

A Correlation Heatmap will be conducted using Python Libraries like Pandas and Matplot in an attempt to provide a visual representation of the numerical variable associations in a dataset to identify patterns and relationships easily (Khadka, 2019). It indicates strong positive or negative correlations that can help identify the ways variables influence one another. For instance, in a vulnerabilities data set, a high correlation between "Severity Score" and "Patch Possible?" can reveal patterns in vulnerabilities being prioritized or segmented. Also, it can help identify redundancies by displaying highly correlated variables that contain duplicative information, thereby simplifying the process of data analysis (Rabzelj et al., 2020).

The objective of this Campus Ready correlation heatmap report is to present further insight into the patch availability and vulnerability severity relationship in order to better help the company make informed decisions regarding resource deployment and remediation priority. By identifying the gap between critical vulnerabilities and patches, Campus Ready can design a vulnerability management plan that addresses the essential gaps, improves planning, and secures its e-Assessment platform. This assessment ultimately facilitates data-driven decision-making, encouraging an adaptive and proactive approach to cybersecurity (Rabzelj et al., 2020).

3.7. Risk Evaluation Criteria

The effectiveness of the studys detection and evaluation of exposures to unpatched software in the Campus Ready e-Assessment system and the feasibility of the suggested solutions and they are intended to be verified (Mell et al., 2005).

Identification and Categorization of Vulnerabilities

First, the researchs "identification and categorization of vulnerabilities" will be assessed. This will center on how well the study can identify and categorize vulnerabilities especially serious ones resulting from unpatched software. Examining whether all notable vulnerabilities were found and properly categorized in accordance with their severity levels is part of this process (Shahid et al., 2022, p. 12).

Table 3: Categorization of Vulnerabilities

Severity	Description
None*	Represents the absence of impact; no exploitable risk is associated with the vulnerability.
Low	Vulnerabilities with minimal impact, often requiring limited attention unless in large volumes.
Medium	Moderate impact vulnerabilities that may be exploited in certain conditions but are not critical.
High	Vulnerabilities that pose a significant risk and require prompt remediation to prevent impact.
Critical	Severe vulnerabilities that pose an immediate and high-impact threat, demanding urgent action.

Source (s): (NVD – NIST Home, n.d.)

Risk Scoring

A risk scoring system will be implemented to quantify the risk associated with each vulnerability. Risk scoring will help in the dataset to compute a composite score based on factors like severity level. This scoring system will allow vulnerabilities to be ranked and prioritized effectively to enable targeted mitigation strategies (NVD – NIST Home, n.d.).

To calculate the Risk Score for vulnerabilities in the table, the study will typically need to use a formula that combines the Severity Score, the Impact Scope. Therefore, one simple approach is to multiply the Severity Score with the Impact Score (or a scaled value based on the impact) (NVD – NIST Home, n.d.).

Where:

Severity Score is a value between 0 and 10 (already provided) (NVD – NIST Home, n.d.)
Impact Scope can be quantified as follows:

- None (0.0) – No threat (NVD – NIST Home, n.d.)
- Low (0.1-3.9) – Minor threat or vulnerability (NVD – NIST Home, n.d.)
- Medium (4.0-6.9) – Moderate threat (NVD – NIST Home, n.d.)
- High (7.0-8.9) – Significant threat that requires attention (NVD – NIST Home, n.d.)
- Critical (9.0-10.0) – Immediate threat requiring urgent action. (NVD – NIST Home, n.d.)

Severity and Risk Assessment

Second, the researchs ability to evaluate the impact and associated risks of each vulnerability on the platforms overall security will be assessed based on two key criteria: the second is the "severity and risk assessment" which will be closely scrutinized to ascertain the accuracy of the assessments made during the data analysis phase (Mell et al., 2005 pp.2-9).

After the preliminary evaluation, the vulnerabilities found will be examined and categorized according to their possible influence on the Campus Ready e-Assessment platform and their respective severity levels (critical, high, medium and low v4.0 Ratings). Industry-standard vulnerability scoring systems, including the Common Vulnerability Scoring System (CVSS) v4.0 Ratings, will serve as the classifications compass (NVD – NIST Home, n.d.) & (Mell et al., 2005, p. 3-2). Since vulnerabilities related to unpatched software are the main focus of the investigation and special attention will be paid to them (Mell et al., 2005, p. 3-2):

Table 4: CVSS v4.0 (Version 4.0) Ratings for Risk Evaluation

Severity	Severity Score Range
None*	**0.0**
Low	**0.1-3.9**
Medium	**4.0-6.9**
High	**7.0-8.9**
Critical	**9.0-10.0**

Source (s): (NVD – NIST Home, n.d.)

Feasibility of Recommendations

Third, an evaluation of the "feasibility of recommendations" will be conducted to ascertain whether the suggested solutions are not only workable but also economical and in line with industry best practices, all of which increase the probability that they will be implemented successfully (Mell et al., 2005, pp. 5-2 & 6-1).

Scalability and Adaptability of Recommendations

For making the proposal relevant and applicable for wider scope, the "scalability and adaptability" of proposed patches would be verified to determine if they would be viable for similar systems with similar weaknesses. These will collectively serve as the integrated framework to determine to what extent research has augmented the mechanisms of cybersecurity in Campus Ready e-Assessment environments (Mell et al., 2005, pp. 4-2).

Metrics for Evaluating Recommendations (Implementation)

Lastly, the study will employ specific measures and metrics in determining the effectiveness of the proposed measures to mitigate vulnerabilities on the Campus Ready e-Assessment platform. One of the metrics employed in computing the average time taken to apply patches for identified vulnerabilities is the "Time to Patch." The lower the patching time, the more effective the vulnerability management process. In addition, the "Reduction in Vulnerability Count" would be monitored to calculate the percentage reduction in identified vulnerabilities

when the proposed controls are implemented. This metric reflects the general impact of the recommended measures on the security stance of the platform (Roumani, 2021).

In order to maintain harmony with documented standards "Compliance Scores" will be ascertained by measurement of compliance with ISO/IEC 27001:2022 framework. This will maintain confirmation of the dependability and industry suitability of the recommendations. Lastly, "Scalability Ratings" will assess the scalability of the proposed solutions to fit other systems or environments in a bid to determine their broader utilization. Since these benchmarks will provide an overall framework for ascertaining effectiveness, sustainability and compliance with best practices like ISO/IEC 27001:2022.

3.8. Reliability and Validity

The actions listed below will be implemented to guarantee the correctness and dependability of the results:
The penetration testing and vulnerability scanning tools chosen (Nmap, OpenVAS, OWASP ZAP) are industry-standard and well-respected tools that are used universally for their precision in vulnerability detection. Their performance, which will be repeated over several scans, will be consistent, thereby providing assurance in the validity of the data gathered. For result consistency, the vulnerability scans and penetration testing will be repeated several times before and after installation. Cross-referencing the results from various sources and utilization of various technologies to scan the identical systems will assist in validating the findings (Shahid et al., 2022).

3.9. Strategy Development

In the Stage of Development of Strategy, wherein strategies will be formulated to eliminate the vulnerabilities identified and test their feasibility for suggested implementations based on PDCA approach, patch and vulnerability and configuration management will predominantly be highlighted and they are consisting short as well as long term approaches. Short-term suggested strategies might involve emergency patches or quick fixes to mitigate immediate risks while long term suggested solutions will entail redesigning platform components to enhance patch management processes and implementing more rigorous testing protocols to prevent patches from disrupting functionality (Ganji et al., 2020; Laurence, 2023).

Additionally, automated patch tools will be introduced to streamline the patching. In terms of change management and risk reduction that is a written plan will be created to minimize disruptions to users of the platform while prioritizing remediation of high-priority vulnerabilities.

This could include the design of a system of tiered prioritization of patch release as well as enabling features such as more advanced user access controls, data encryption, and backup redundancy to safeguard the platform from attack by unpatched software. The feasibility and viability of these proposals will be established based on consideration of the cost of deployment, available staff expertise, and deployment time. The plans will be validated to guarantee that they can be deployed without disrupting ongoing processes or platform responsiveness.

The study will focus on the implementation of ISO/IEC 27001:2022 for Information Security Management Systems. The standards will be utilized to address vulnerabilities, patch management and configuration policies. Since, through the implementation of this standard, Campus Ready will institutionalize best practices in cybersecurity and risk management to make the platform secure, reliable and compliant while promoting its growth and credibility in the education sector (Proença & Borbinha, 2018).

3.10. Ethical Considerations

Since this research involves sensitive material, following stringent ethical norms is essential to maintaining the studys integrity and safeguarding the interests of all those involved. Prior to starting any vulnerability checks "authorization and permissions" will be carefully sought from the e-learning platform administrators. To achieve mutual understanding and agreement, it will be necessary to explicitly define the scope and bounds of the evaluations. This will help to prevent any ethical breaches that may result from illegal access or activities.

In addition, all study findings will be anonymized in order to remove any sensitive or personally identifying information (PII) from the reports in the area of "data privacy and confidentiality." This pledge of confidentiality will comply with any data protection laws to ensure that any information gathered throughout the research process is kept and used only for this study (Mccallister et al.

Finally, the methodology of "non-intrusive testing" is crucial: vulnerability assessments will be conducted without affecting the platform's regular operations. As part of this, vulnerabilities will not be actively exploited whereas they protect platform security and data integrity all the way through the study process. The study seeks to maintain the highest standards of professionalism and accountability in cybersecurity research within the framework of the Campus Ready platform by abiding by these ethical guidelines (Zhang et al., 2023).

4. FINDINGS

4.1. Introduction

Digital transformation in educational technology has highlighted the importance of addressing unpatched software vulnerability, particularly during the development and deployment of platforms like Campus Ready. Vulnerabilities arising from outdated/unpatched software components or weak configurations can significantly compromise platform security, operational continuity and data privacy.

To evaluate and mitigate these risks the used that vulnerability scanning and penetration testing techniques were conducted using Nmaps OpenVAS wASP ZAP and O Since these tools provide actionable insights into the nature and severity of security flaws forming the basis for this comprehensive analysis (Shahid et al., 2022). The vulnerability scan can include several important sections to provide the user with an in-depth view of the discovered vulnerabilities. This allows us to understand the vulnerabilities, assess the potential risks and prioritize remediation efforts accordingly.

The primary goal of the vulnerability scan and penetration testing is to identify and assess potential security weaknesses in network infrastructure, servers and web applications of the target systems on Campus Ready e-Assessment platform. Therefore, the findings (some vulnerability scanning and penetration testing techniques) have been conducted in 20.01.2025.

4.2. Overview of Vulnerability Scanning and Penetration testing Results

4.2.1. Total Cyber Security Vulnerabilities Identified in Campus Ready

This section provides a high-level overview of the results from the vulnerability scan, summarizing the overall findings. The table below summarizes the cyber security vulnerabilities identified during the scanning and testing processes, categorized by their severity, affected systems and other critical attributes:

Vulnerability scans and penetration tests were done on certain servers, networks and applications. This section explains the potential vulnerabilities found through these tests. The vulnerabilities are grouped by their severity with higher severity meaning a bigger risk of data breaches, loss of integrity or problems with the availability of the targets.
Campus Ready e-Assessment platform it quietly new platform and it is still-undergoing development process; therefore, the results of Vulnerability scans and penetration testing will be enough to define the cyber security vulnerabilities caused by unpatched software.

Below are the total number of vulnerabilities found by severity. Critical vulnerabilities are the most severe and should be evaluated first.

Table 5: Detected Cyber Security Vulnerabilities on Campus Ready e-Assessment Platform

Vul. ID	Type of Vulnerability	Severity Score	Severity Level	Affected Software/ System	Impact Scope	Patch Possible?	Discovery Date	Detection Source
1	Absence of Anti-CSRF Tokens	5	Medium	Web Application	Potential CSRF attacks	NO	20.01.2025	OWASP ZAP
2	Missing Anti-clickjacking Header	5	Medium	Web Application	UI redirection vulnerability	NO	20.01.2025	OWASP ZAP
3	Content Security Policy (CSP) Header Not Set	7,5	High	Web Application	Cross-site scripting risks	NO	20.01.2025	OWASP ZAP
4	X-Content-Type-Options Header Missing	2	Low	Web Application	Risk of MIME-type sniffing	YES	20.01.2025	Nmap
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	Web Application	Potential data exposure	YES	20.01.2025	OWASP ZAP
6	Server Leaks Version Information via 'Server' Header	3	Low	Web Server	Information disclosure	YES	20.01.2025	OWASP ZAP
7	Strict-Transport-Security Header Not Set	5,5	Medium	Web Application	Risk of downgrade attacks	NO	20.01.2025	Nmap
8	TCP Timestamps Information Disclosure	2,6	Low	Network Infrastructure	Uptime information leakage	YES	20.01.2025	OpenVAS
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	SSH Server	Potential data integrity issue	YES	20.01.2025	OWASP ZAP
10	Open TCP Port: 22	5	Medium	Web Server	Unsecured access point	NO	20.01.2025	Nmap
11	Open TCP Port: 443	3,5	Low	Web Server	Potential unauthorized access	YES	20.01.2025	Nmap
12	Open TCP Port: 80	3	Low	Web Server	Potential unauthorized access	YES	20.01.2025	Nmap
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	Web Application	Risk of session hijacking	YES	20.01.2025	OWASP ZAP
14	Missing security header: Referrer-Policy	5	Medium	Web Application	Information leakage via referrers	NO	20.01.2025	Nmap
15	Server Software and Technology Found	2,5	Low	Web Server	Information disclosure	YES	20.01.2025	OpenVAS
16	Security.txt file is missing	5	Medium	Web Server	Missing security contact details	NO	20.01.2025	OWASP ZAP
17	HTTP OPTIONS Enabled	5	Medium	Web Server	Potential method discovery attack	NO	20.01.2025	Nmap
18	Cookie without SameSite Attribute	2	Low	Web Server	Web Server: A cookie lacks the SameSite attribute, allowing cross-site requests and increasing vulnerability to CSRF, script inclusion, and timing attacks.	YES	20.01.2025	OWASP ZAP

Source: Arthur's own work

4.2.2. Cyber Vulnerabilities Breakdown

Table 6: Cyber Vulnerabilities More in Detail

Vul. ID	Type of Vulnerability	Severity Level	Description
1	Absence of Anti-CSRF Tokens	Medium	"No Anti-CSRF tokens were found in the HTML submission form. Cross-Site Request Forgery (CSRF) is an attack where a user is tricked into sending an HTTP request to a target site without their knowledge, making actions happen on the attacker's behalf. This happens because the application uses predictable URLs or form actions. CSRF takes advantage of the trust a website has in its user, while Cross-Site Scripting (XSS) takes advantage of the user's trust in the website. CSRF attacks are not always cross-site, but they can be. Other names for CSRF include XSRF, one-click attack, session riding, confused deputy, and sea surf. CSRF is most dangerous when the victim is logged in to the target site, authenticated with HTTP, or on the same local network as the target. CSRF is often used to perform actions with the victim's privileges, but it can also be used to steal sensitive information. The risk of data theft is higher when the target site is also vulnerable to XSS, as XSS can allow CSRF to bypass security restrictions (OWASP, 2012).
2	Missing Anti-clickjacking Header	Medium	The response does not protect against clickjacking attacks. It is advised to add either a Content-Security-Policy header with the 'frame-ancestors' directive or an X-Frame-Options header to prevent such attacks (Mozilla, n.d.).
3	Content Security Policy (CSP) Header Not Set	High	Content Security Policy (CSP) is an extra layer of security that helps detect and prevent threats like Cross-Site Scripting (XSS) and data injection. These attacks can lead to problems such as data theft, website changes, or malware spread. CSP uses a set of standard HTTP headers that let website admins define trusted sources of content that browsers can load on a page. This includes content like JavaScript, CSS, HTML frames, fonts, images, and embedded objects such as Java applets, ActiveX controls, and audio and video files (World Wide Web Consortium (W3C), n.d.).
4	X-Content-Type-Options Header Missing	Low	The X-Content-Type-Options header, which is used to prevent MIME sniffing, was not set with the 'nosniff' value. As a result, older versions of Internet Explorer and Chrome can analyze the response body, potentially interpreting it as a different content type than the one declared. On the other hand, both current (early 2014) and older versions of Firefox will use the specified content type, if provided, instead of performing MIME sniffing (OWASP, n.d.).
5	Cross-Domain JavaScript Source File Inclusion	Low	The page includes one or more script files from an external domain (Shaikh, 2020).
6	Server Leaks Version Information via 'Server' Header	Low	The web/application server is revealing version information through the 'Server' HTTP response header. This can help attackers identify potential vulnerabilities in the server (Microsoft Archiveddocs, 2010).
7	Strict-Transport-Security Header Not Set	Medium	HTTP Strict Transport Security (HSTS) is a web security policy that tells supported user agents, like web browsers, to only use secure HTTPS connections (i.e., HTTP over TLS/SSL). HSTS is an IETF standard, defined in RFC 6797 (OWASP, n.d.).
8	TCP Timestamps Information Disclosure	Low	The remote host utilizes TCP timestamps in accordance with RFC1323/RFC7323, which may inadvertently allow an observer to estimate its uptime (Borman, Braden and Jacobson, 1992; Borman et al., 2014).
9	Weak MAC Algorithm(s) Supported (SSH)	Low	The remote SSH server is configured to permit insecure MAC algorithms (Baushke & Bider, 2012).
10	Open TCP Port: 22	Medium	TCP Port 22 is commonly used for Secure Shell (SSH), which provides secure access for remote system administration and file transfers. However, if left open without proper configuration or monitoring, this port can become a significant security vulnerability. For instance, attackers can exploit outdated SSH implementations to gain unauthorized access. Brute force or dictionary attacks on weak SSH credentials can also compromise the system. Additionally, enabling SSH with insecure MAC algorithms or protocols increases the risk of man-in-the-middle attacks (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020).
11	Open TCP Port: 443	Low	TCP Port 443 is typically used for secure HTTP (HTTPS) traffic. While it is a standard configuration for secure web communication, an open Port 443 without adequate monitoring or patching can expose services to vulnerabilities. For example, outdated SSL/TLS protocols or weak ciphers could be exploited, allowing attackers to decrypt sensitive data during transmission. Misconfigured HTTPS services might also leak server information, enabling attackers to identify and exploit further vulnerabilities (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020).
12	Open TCP Port: 80	Low	TCP Port 80 is commonly used for HTTP traffic. Unlike Port 443, it does not encrypt data, making communications susceptible to interception and manipulation (e.g., man-in-the-middle attacks). An open Port 80 is particularly risky if the associated web server software is outdated or misconfigured. Attackers may exploit known vulnerabilities in web server versions to launch attacks such as SQL injection or remote code execution. Additionally, leaving HTTP open can bypass secure HTTPS implementations,

			exposing sensitive traffic to eavesdropping (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020).
13	Insecure Cookie Setting: Missing Secure Flag	Low	A cookie was set without the Secure flag, allowing it to be transmitted over unencrypted connections (OWASP, n.d.).
14	Missing security header: Referrer-Policy	Medium	The response headers are missing the Referrer-Policy HTTP security header and the 'referrer' meta tag is not present. This poses a risk because if a user visits a page (e.g., 'http://example.com/pricing/') and clicks a link to another site (e.g., 'https://www.google.com'), the browser will send the full originating URL in the Referer header if the Referrer-Policy is not set. This URL may contain sensitive information and could be used for user tracking (developer.mozilla.org, n.d.)
15	Server Software and Technology Found	Low	The risk is that an attacker could leverage this information to conduct targeted attacks based on the identified software version and type (OWASP, 2019).
16	Security.txt file is missing	Medium	While the absence of a security.txt file does not pose an immediate risk, it is important as it provides an official channel for reporting vulnerabilities and security issues (CISA, 2023).
17	HTTP OPTIONS Enabled	Medium	The primary risk is the potential exposure of debug HTTP methods on the server. If these methods allow access to sensitive information, such as authentication credentials or secret keys, they could pose a security threat (Nedim, 2020)
18	Cookie without SameSite Attribute	Low	A cookie has been set without the SameSite attribute, making it vulnerable to transmission in cross-site requests. This attribute is critical for protecting against threats like cross-site request forgery (CSRF), cross-site script inclusion and timing attacks (West & Goodwin, 2016; Nginx, 2024).

Source: Arthur's own work

The dataset presented above was used to derive quantitative findings primarily focused on the Severity Score which serves as the key numerical variable. This quantitative data enabled an analysis of the distribution and relationships between vulnerabilities based on their severity. The analysis also evaluated potential correlations with categorical variables such as Severity Level and Detection Source to gain deeper insights into the patterns and trends within the dataset.

Visualized Data Using Python: "Pandas and Matplotlib" for Analyzing Findings

The findings of the Campus Ready e-Assessment platform's vulnerabilities were analyzed and visualized using Python libraries like Pandas and Matplotlib (see Appendix A). It was achieved through raw data presentation to visualize meaningful insights to make it easier to identify patterns and relationships within the data (Khadka, 2019).

Pandas has been used for efficient data manipulation and organization (Khadka, 2019). The dataset was loaded into a Pandas DataFrame and contained data on vulnerabilities, severity levels, patch availability, and affected systems. This allowed the application of filters, grouping, and sorting to separate the data into critical vulnerabilities or specific types of software flaws (Khadka, 2019).

Among these visualization options were bar charts, pie charts, and heat maps for distribution and correlation analysis, all generated by Matplotlib (Khadka, 2019).

Bar Plot of Vulnerabilities by Severity Level:

The bar plot shows the total number of vulnerabilities categorized according to Severity Level(Low, Medium, High) after findings. It emphasizes the fact that most vulnerabilities in the dataset fall into the Low Severity category(9 instances)after that, Medium Severity(6 instances) and High Severity accounted for the least number ofVulnerabilities(3 instances).This suggests that the vulnerabilities identified don't fall into any most-critical but present a mixed bag of security issues or vulnerabilities that need attention. Distribution gives a feel of the risk landscape, where organisations can show and point out that most vulnerabilities are not exigencies or emergencies but still deserve some attention.

Bar Plot of Vulnerabilities by Detection Source:

This finding describes the frequency of vulnerabilities detected by security tools such as OWASP ZAP, Nmap and OpenVAS. The plot shows that OWASP ZAP is the most prolific tool to use, identifying 10 vulnerabilities of the 18 analyzed, while Nmap and OpenVAS were able to detect 6 and 2, respectively. This distribution shows the extent to which specific security tools are applied in vulnerability analysis, thus indicating OWASP ZAP as more significant towards web application spaces, followed by Nmap and OpenVAS, whose value lies in detecting network- and server-based vulnerabilities.

Distribution of Severity Scores:

The histogram illustrates the distribution of Severity Scores, revealing how the vulnerabilities are spread across different severity levels based on their scores (ranging from 0 to 10). The majority of vulnerabilities are clustered in the lower to medium range of scores (around 2 to 5), indicating that most vulnerabilities identified in the dataset are relatively minor. However, a few vulnerabilities reach higher severity scores, with one reaching the highest score of 7.5, suggesting that while most vulnerabilities pose lesser risks, there are still some high-impact issues that could potentially be more harmful if not addressed.

Box Plot of Severity Scores by Severity Level:

This box plot compares the Severity Scores across the three severity levels (Low, Medium, High), showing that how the scores are distributed within each level. It indicates that Low Severity vulnerabilities have the lowest scores, with a range between 2 and 3.5, signifying that they are less critical for Campus Ready. Medium Severity vulnerabilities have scores ranging from 3 to 6, indicating a moderate level of risk, while High Severity vulnerabilities have the highest scores, with one reaching 7.5. The box plot emphasizes that as the

severity level increases, so does the severity score, highlighting the direct relationship between the two.

This heatmap illustrates the correlation between the Severity Score of vulnerabilities and the binary variable "Patch Possible?" This is again perfect correlation, 1.00, being reflected along the diagonal, representing each variable being entirely correlated with itself. The values off the diagonal, however, depict a significant negative Spearman's correlation of -0.77 between Severity Score and Patch Possible.

That is, higher severity vulnerabilities are not likely to have patches readily available and pose challenges for prioritizing remediation work. Colors indicate this relationship, with dark red strong correlations (1.00) on one end and shades of blue reflecting negative correlations.

It is an analysis of the correlation heatmap for Campus Ready, against which relationship between severity and patch availability is understood fully to aid decision-making in the allocation of remediation within the organization. This analysis reveals a gap between the very severity of vulnerabilities when compared alongside patch availability, relevant and objects further other than just beside sharing informatives. This perhaps led Campus Ready to a customized design and reorganization of its vulnerabilities management so that key shortcomings in planning are addressed effectively for the e-assessment platform to heighten its resilience. All in all, this analysis promotes data-driven decision-making processes, leading to a stronger, proactive, and adaptive stance toward organizational cybersecurity.

4.2.3. Key Trends Observed After Findings

Based on the identified vulnerabilities in the table "Cyber Vulnerabilities More in Details" in the Campus Ready e-Assessment platform, several key trends have emerged that highlight recurring issues and areas of concern.

Lack of Essential Security Headers

The majority of the security problems are a result of the lack or improper setup of security headers, including Content-Security-Policy (CSP), X-Content-Type-Options and Strict-Transport-Security. These are essential in safeguarding against attacks like Cross-Site Scripting (XSS) and data injection. Their absence indicates shortcomings in the application of HTTP security best practices to make web applications susceptible to malicious attacks (OWASP, n.d.).

Open Ports and Exposed Services

Multiple open TCP ports, including 22, 443 and 80, were detected which expands the attack surface of the platform. While these ports may be part of expected configurations, their presence without proper monitoring or regular updates can lead to risks, especially when associated services are outdated or misconfigured. Such exposure can potentially allow unauthorized access to applications, networks or sensitive data (Borman, Braden, & Jacobson, 1992).

Insufficient Cookie Security Configurations

Cookies on the platform were found to be missing critical security attributes, such as the Secure and SameSite flags. Without these protections, cookies may be exposed to risks such as Cross-Site Request Forgery (CSRF) and session hijacking, compromising user data and session integrity. This oversight indicates weaknesses in cookie management practices (West & Goodwin, 2016).

Vulnerabilities from Missing Files and Protocols

The platform lacks critical files and protocols, such as a security.txt file and Referrer-Policy headers. The absence of a security.txt file hinders vulnerability reporting, while missing Referrer-Policy headers could lead to sensitive information leaks during user navigation. Such gaps reduce the platform’s ability to maintain transparency and protect user privacy (CISA, 2023; Mozilla, n.d.).

Cross-Domain Script Inclusions

The inclusion of cross-domain JavaScript files from third-party domains introduces additional vulnerabilities. Without rigorous validation of these resources, attackers could exploit these inclusions to inject malicious code, undermining platform security. This issue emphasizes the risks of dependency on external scripts without adequate scrutiny (Shaikh, 2020).

Exposure of Server Information

The server was found to leak version details via HTTP response headers. This information could assist attackers in identifying specific vulnerabilities associated with the server's software versions, enabling more targeted exploits. Such exposure highlights a failure to minimize information disclosure, a critical aspect of secure server configurations (Microsoft Archived Docs, 2010).

Weak Cryptographic Protocols

The SSH server on the platform supports outdated/weak MAC algorithms which weakens the encryption standards in place. This vulnerability compromises secure communication and increases susceptibility to attacks like cryptographic downgrade or brute force (Baushke & Bider, 2012).

Lack of Anti-CSRF Tokens

Forms on the platform lack anti-CSRF tokens, leaving them vulnerable to Cross-Site Request Forgery (CSRF) attacks. Such attacks could force users to execute unwanted actions on the platform without their consent, exploiting their authenticated sessions. This gap reflects inadequate session protection mechanisms (OWASP, 2012).

Improper Handling of Transport Layer Security (TLS)

The lack of HTTP Strict Transport Security (HSTS) headers shows out insufficient enforcement of secure HTTPS connections. This oversight increases the risk of man-in-the-middle attacks and data interception during communications, undermining the confidentiality of sensitive data (OWASP, n.d.).

Recurring Low-Severity Vulnerabilities

Several low-severity vulnerabilities, such as missing SameSite attributes and open ports, were identified. Although individually these may not pose significant risks, their cumulative effect can escalate exposure, particularly when combined with other vulnerabilities. Addressing these issues proactively is essential to minimizing the overall risk (Nginx, 2024).

4.3. Relation Between Observed Cyber Vulnerabilities and Unpatched Campus Ready e-Assessment Platform

The vulnerabilities identified in the Campus Ready e-Assessment platform are closely connected to the risks posed by unpatched software. These unpatched systems fail to address known weaknesses, leaving the platform exposed to various cyber threats. Below is an in-depth explanation of the relationship between these vulnerabilities and the platform's unpatched nature (Temizkan et al., 2012):

Missing Security Headers

Several vulnerabilities, such as missing Content-Security-Policy (CSP) headers and Strict-Transport-Security (HSTS) headers, were identified as consequences of unpatched software. These missing headers expose the platform to risks like Cross-Site Scripting (XSS) and man-in-the-middle (MITM) attacks. The absence of regular updates leaves these protections unimplemented, significantly increasing the platform's attack surface and vulnerability to exploitation (Calzavara et al., 2018; Rashidi et al., 2020).

Open Ports and Misconfigured Services

Unpatched software contributes to misconfigured services, as evidenced by open TCP ports (22, 80, 443). While open ports are expected for operational purposes, unpatched systems fail to implement secure configurations, making them susceptible to unauthorized access and attacks on outdated protocols. These vulnerabilities can be exploited for privilege escalation or service disruption, particularly when associated services lack the latest security updates (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020)

Insecure Cookie Attributes

Vulnerabilities like missing Secure and SameSite cookie attributes indicate the use of outdated software that does not adhere to modern web security standards. These insecure cookie configurations allow attackers to exploit session management flaws, potentially leading to session hijacking or unauthorized access. Unpatched systems exacerbate these risks by failing to enforce best practices for secure cookie handling (Priyawati et al., 2022; Rashidi et al., 2020).

Legacy Systems and Outdated Protocols

Unpatched software often supports legacy systems that rely on outdated protocols, such as weak SSH Message Authentication Code (MAC) algorithms. These weaknesses compromise encrypted communication channels, allowing attackers to intercept or manipulate data. Without timely updates, these vulnerabilities remain active, jeopardizing the security of sensitive communications (Bäumer et al., 2024; Rashidi et al., 2020).

Delays in Applying Patches During Digital Transformation

Digital transformation (Software Installation & Upgrades) efforts often prioritize scalability and functionality, resulting in delayed patching for interconnected components. Vulnerabilities such as the absence of anti-CSRF tokens and exposed Referrer-Policy headers are indicative of this challenge. These delays create a window of opportunity for attackers to exploit known

issues before updates can be applied (Priyawati et al., 2022; Florian Plainer et al., 2020; Rashidi et al., 2020).

Absence of Proactive Security Measures

Findings such as the missing security.txt file highlight a lack of proactive security planning. Unpatched systems are less likely to include mechanisms for reporting vulnerabilities, which delays the resolution of security issues and increases exposure to potential attacks. Addressing this oversight is essential for ensuring timely identification and mitigation of risks (Findlay & Abdou, 2022; Rashidi et al., 2020).

Cumulative Impact of Low-Severity Vulnerabilities

Recurring low-severity vulnerabilities, including open ports and minor header misconfigurations, collectively contribute to a higher risk profile. While individually these issues may seem negligible, their cumulative effect can be significant when combined with other unpatched vulnerabilities, creating pathways for more severe exploits (Rashidi et al., 2020).

4.4. Impact of Unpatched Software on Cybersecurity Vulnerabilities in Campus Ready e-Assessment Platform During Digital Transformation Processes (Software Upgrades)

Increased Attack Surface During Upgrades

Unpatched software creates significant entry points for cyberattacks during Campus Ready's digital transformation (Software Installation & Upgrades). As the platform integrates new technologies and upgrades its systems, vulnerabilities in legacy software or outdated components are often overlooked. These gaps can be exploited by attackers, particularly during periods of transition when systems may be more vulnerable (Divya et al., 2024; Osborne & Boone, 2023).

Data Breach Risks

Unpatched vulnerabilities, such as outdated libraries and insecure configurations, expose Campus Ready's e-Assessment platform to unauthorized access risks. Sensitive applicant and institutional data become more vulnerable during upgrades, as interactions between new modules and older systems create exploitable security gaps for attackers to target personal and academic records (Divya et al., 2024; Osborne & Boone, 2023).

Operational Downtime

Software upgrades for Campus Ready are prone to temporary disruptions. Unpatched vulnerabilities can extend these downtimes due to security incidents or system crashes, reducing the platform's availability. This is especially critical during high-demand periods like admissions cycles, where uninterrupted service is essential (Divya et al., 2024; Osborne & Boone, 2023).

Compromised System Integrity

Unpatched software during Campus Ready's digital upgrades undermines the platform's system integrity. Vulnerabilities such as insecure configurations or missing security headers provide pathways for attackers to inject malicious code, alter system functionality, or gain unauthorized privileges, eroding trust in the platform (Divya et al., 2024; Osborne & Boone, 2023).

Financial and Reputational Costs

Campus Ready faces financial risks from cyberattacks exploiting unpatched software, including costs for incident response, regulatory fines and emergency patch implementations. Additionally, data breaches or prolonged outages damage the institution's reputation, reducing stakeholder trust and affecting future partnerships or user adoption (Divya et al., 2024; Osborne & Boone, 2023).

Amplification of Vulnerabilities Through Interdependencies

Campus Ready's digital transformation creates interdependencies between various systems and applications. Vulnerabilities in one module can cascade through interconnected systems, amplifying security risks across the platform. This highlights the need for robust patch management to mitigate cascading threats and maintain a secure operational environment (Divya et al., 2024; Osborne & Boone, 2023).

4.5. Root Couse Analysis

Root Cause Analysis (RCA) is a systematic approach that helps to define the underlying causes of cybersecurity vulnerabilities in the Campus Ready to ensure that issues are addressed at their source rather than their symptoms. For Campus Ready's e-Assessment platform, RCA is essential in uncovering real weaknesses behind the issues has been found in the "Findings" section, such as delays in patch management or misconfigurations during digital transformation (Software Installation & Upgrades) (Zhou et al., 2007).

By analyzing why certain vulnerabilities arise, whether due to resource constraints, poor communication or compatibility issues between legacy and new systems, RCA enables the implementation of long-term solutions. This proactive approach enhances the platform's resilience, reduces repeated security incidents and ensures compliance with cybersecurity standards like ISO/IEC 27001. Ultimately, RCA helps Campus Ready build a secure, efficient, and trusted platform, supporting its mission in the educational technology space (Zhou et al., 2007; ISO 27001:2022, 2023).

4.5.1. Observed Challenges of Implementing Patch Management in Campus Ready

Right now, as realized that, implementing patch management for the Campus Ready e-Assessment platform comes with significant challenges, particularly as the platform evolves during digitalization/software upgrades. Ensuring that all vulnerabilities are addressed promptly is critical, yet several obstacles hinder effective patch management.

Firstly, one major challenge lies in the team, team members cannot fully understand the importance of patch management. Non-technical staff can see patches as unnecessary or burdensome, overlooking their critical role in maintaining security, please check the point "3.2.2 Campus Ready GmbH Overview". This lack of awareness is leading to delayed updates, leaving the platform exposed to potential vulnerabilities (Dissanayake et al., 2022).

Secondly, resource and budget constraints, both in terms of IT security personnel and budget pose another challenge because they are quite new startup and looking for investments or fundings. The Campus Ready team must balance patch management with other operational priorities. Limited staff availability and lack of awareness and the absence of automation tools mean that patches may be applied inconsistently or overlooked entirely (Dissanayake et al., 2022).

Thirdly, there is a pervasive concern that applying patches could lead to system downtime or disrupt the platform's performance during crucial periods, such as university admissions. Past negative experiences with failed updates or compatibility issues can reinforce this fear, causing hesitation in implementing patches promptly (Dissanayake et al., 2022).

Fourthly, the sheer number of patches released by software vendors can be overwhelming. Prioritizing which patches to apply first especially when balancing critical security updates with less urgent fixes is a challenge for the Campus Ready team. This is further complicated by unexpected patch releases which may require immediate attention and testing (Dissanayake et al., 2022).

Fifth, End-user resistance is another barrier to patch management. Campus Ready Team may view updates as unnecessary complications, especially when the platform is functioning well. Without a clear understanding of the risks posed by unpatched vulnerabilities, users may resist changes that they perceive as disruptive (Dissanayake et al., 2022).

Finally, Campus Ready currently is not working for applying patches with a proof, please check the findings which leads to delays in addressing critical vulnerabilities. Without automated patch systems to prioritize and deploy patches, the platform is prone to unpatched software remaining active for extended periods (Dissanayake et al., 2022).

4.6. Summary of Findings

The security assessment of the Campus Ready e-Assessment platform identified multiple cybersecurity vulnerabilities, primarily due to unpatched software, weak configurations and missing security measures. Using Nmap, OpenVAS, and OWASP ZAP, the study revealed 18 vulnerabilities, categorized into high (3), medium (6) and low (9) severity levels. Common issues included missing essential security headers (CSP, Strict-Transport-Security), open TCP ports, weak cookie security, outdated cryptographic protocols and exposed server information. The findings highlighted a strong correlation between these vulnerabilities and poor patch management that are leading to risks such as unauthorized access, data breaches and operational disruptions. Facing issues in implementing patch management included lack of awareness, resource constraints, fear of system downtime and end-user resistance. During digital transformation (software upgrades and installations), these weaknesses amplify security risks, increase attack surfaces and compromise system integrity.

5. Implementation (Suggested Actionable Solutions)

In this section, this study focuses on providing comprehensive solutions to address the vulnerabilities identified within the Campus Ready e-Assessment platform. By leveraging ISO/IEC 27001:2022 ISMS guidelines, the study outlines practical strategies for implementing robust patch management, vulnerability management and configuration management frameworks. It emphasizes the importance of identifying existing gaps, deploying automated tools and adhering to industry best practices to mitigate risks effectively. Additionally, the study integrates a metrics-driven approach to track performance and ensure continuous improvement through the Plan-Do-Check-Act (PDCA) cycle (Ganji et al., 2020). This study's solutions aim to enhance Campus Ready's overall security posture, reduce vulnerabilities and establish a resilient cybersecurity framework for the platform.

5.1. Introduction of ISO/IEC 27001:2022

ISO/IEC 27001:2022 is a globally known standard to manage information security, designed to protect sensitive data through a structured framework of best practices. Built on the Plan-Do-Check-Act (PDCA) cycle, it emphasizes continuous improvement and addresses critical areas like access control, cryptography and vulnerability management. For Campus Ready, an AI-driven e-Assessment platform, adopting ISO/IEC 27001:2022 is essential for securing sensitive applicant and institutional data to ensure compliance with regulatory requirements, and managing cybersecurity risks (Ganji et al., 2020; Laurence, 2023).

By implementing this standard, Campus Ready can mitigate vulnerabilities such as unpatched software or misconfigurations, ensuring data integrity and privacy. Certification also builds trust among stakeholders, enhances the platform's credibility and demonstrates a commitment to global security standards. Additionally, the standard supports operational resilience through robust incident response mechanisms and prepares the platform for scalability, aligning with its growth objectives. In adopting ISO/IEC 27001:2022, Campus Ready can ensure a secure, compliant and reliable foundation for innovation and expansion in the educational technology market (Ganji et al., 2020; Laurence, 2023).

The ISO/IEC 27001:2022 standard outlines 93 security controls listed in Annex A which are divided into four categories. Small and medium enterprises (SMEs) may need to prioritize implementing these controls based on the results of their risk assessments. According to ISO (2022), the categories and controls in Annex A include (Ho et al., 2024; Atqan et al., 2024):

- **A.5 Organizational Controls** (Atqan et al., 2024).
- **A.6 People Controls** (Atqan et al., 2024).
- **A.7 Physical Controls** (Atqan et al., 2024).
- **A.8 Technological Controls** (Atqan et al., 2024).

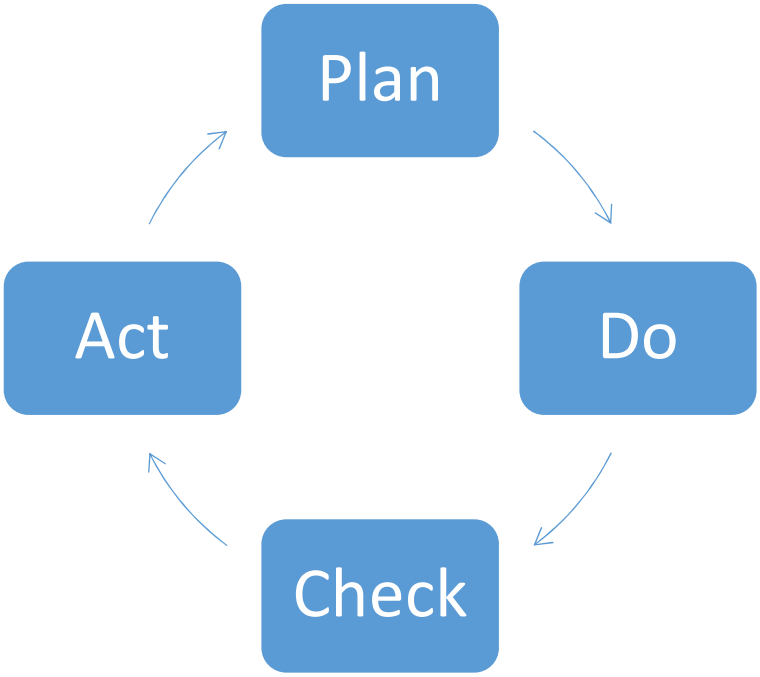


Figure 8: Continuous Improvement Cycle (PDCA)

5.1.1. Plan: Adapting to Evolving Threats

The first step in the PDCA cycle involves regularly updating security policies and procedures. As cybersecurity threats evolve, policies should reflect the latest risks and incorporate updated standards, such as those introduced in ISO/IEC 27001:2022. This stage includes reviewing threat intelligence, assessing new vulnerabilities and identifying changes in regulatory or operational requirements. For Campus Ready, adapting policies ensures the organization remains proactive in addressing risks associated with its digital transformation/software development journey (Ganji et al., 2020; Laurence, 2023).

Table 7: Plan for Mitigating Risks According ISO/IEC 27001:2022

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	ISO/IEC 27001:2022 Control Nr.
1	Absence of Anti-CSRF Tokens	5	Medium	Configuration Management	A.8.9, A.8.18
2	Missing Anti-clickjacking Header	5	Medium	Configuration Management	A.8.9, A.8.18
3	Content Security Policy (CSP) Header Not Set	7,5	High	Configuration Management	A.8.9, A.8.24
4	X-Content-Type-Options Header Missing	2	Low	Configuration Management	A.8.9, A.8.18
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	Configuration Management	A.8.9, A.8.18
6	Server Leaks Version Information via 'Server' Header	3	Low	Vulnerability Management	A.8.8, A.5.7
7	Strict-Transport-Security Header Not Set	5,5	Medium	Configuration Management	A.8.9, A.8.24
8	TCP Timestamps Information Disclosure	2,6	Low	Patch Management	A.8.8, A.8.19
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	Patch Management	A.8.4,A.8.8, A.8.19
10	Open TCP Port: 22	5	Medium	Patch Management	A.8.4,A.8.8, A.8.19
11	Open TCP Port: 443	3,5	Low	Patch Management	A.8.8, A.8.19
12	Open TCP Port: 80	3	Low	Patch Management	A.8.8, A.8.19
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	Configuration Management	A.8.9, A.8.18
14	Missing security header: Referrer-Policy	5	Medium	Configuration Management	A.8.9, A.8.18
15	Server Software and Technology Found	2,5	Low	Patch Management	A.8.8, A.8.19
16	Security.txt file is missing	5	Medium	Vulnerability Management	A.8.8, A.5.37
17	HTTP OPTIONS Enabled	5	Medium	Vulnerability Management	A.8.8, A.5.7
18	Cookie without SameSite Attribute	2	Low	Configuration Management	A.8.9, A.8.18

Source (s): Arthur`s own work**;** Jhala, 2014; ISO, 2022).

5.2. First Step: In Order to Identify the Current State

Understanding the existing vulnerabilities and gaps in the Campus Ready e-Assessment platform is the first step. This involves a comprehensive assessment of the organization's security posture, including identifying unpatched systems, configuration errors and vulnerabilities were found in the findings section along with ISO/IEC 27001:2022 controls. An accurate inventory of IT assets, software and systems is essential for this phase. Therefore, ISO/IEC 27001:2022 can emphasize asset management under **A.8.1 (Operational planning and control)** which requires maintaining an updated record of all organizational assets (Atqan et al., 2024).

An effective approach to strengthening the Campus Ready e-Assessment platform should begin with Asset Discovery where tools like SolarWinds or ManageEngine should be utilized to create a comprehensive inventory of all servers, endpoints and software within the system. This step will ensure a clear understanding of the digital landscape, identifying assets that may be vulnerable or require updates (Sjögård, 2022).

Following this, a Baseline Assessment is conducted, performing a detailed GAP Analysis to check where the gaps (were already found in the findings section) is against the ISO/IEC 27001:2022 controls. Since this process will identify weaknesses such as unpatched software, outdated configurations or misalignments with regulatory standards. Once gaps are identified, the next step is “Prioritization” where vulnerabilities and weaknesses are classified based on their criticality and potential impact on business operations. This prioritization will help to allow for focused remediation efforts to ensure that the most urgent and high-risk issues are addressed first, thus optimizing resource allocation and reducing overall cybersecurity risks (Atqan et al., 2024; Chiara, 2023).

Table 8: GAP Analysis to check where the gap is against the ISO/IEC 27001:2022 controls

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	ISO/IEC 27001:2022 Control Nr.
1	Absence of Anti-CSRF Tokens	5	Medium	Configuration Management	A.8.9, A.8.18
2	Missing Anti-clickjacking Header	5	Medium	Configuration Management	A.8.9, A.8.18
3	Content Security Policy (CSP) Header Not Set	7,5	High	Configuration Management	A.8.9, A.8.24
4	X-Content-Type-Options Header Missing	2	Low	Configuration Management	A.8.9, A.8.18
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	Configuration Management	A.8.9, A.8.18
6	Server Leaks Version Information via 'Server' Header	3	Low	Vulnerability Management	A.8.8, A.5.7
7	Strict-Transport-Security Header Not Set	5,5	Medium	Configuration Management	A.8.9, A.8.24
8	TCP Timestamps Information Disclosure	2,6	Low	Patch Management	A.8.8, A.8.19
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	Patch Management	A.8.4,A.8.8, A.8.19
10	Open TCP Port: 22	5	Medium	Patch Management	A.8.4,A.8.8, A.8.19
11	Open TCP Port: 443	3,5	Low	Patch Management	A.8.8, A.8.19
12	Open TCP Port: 80	3	Low	Patch Management	A.8.8, A.8.19
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	Configuration Management	A.8.9, A.8.18
14	Missing security header: Referrer-Policy	5	Medium	Configuration Management	A.8.9, A.8.18
15	Server Software and Technology Found	2,5	Low	Patch Management	A.8.8, A.8.19
16	Security.txt file is missing	5	Medium	Vulnerability Management	A.8.8, A.5.37
17	HTTP OPTIONS Enabled	5	Medium	Vulnerability Management	A.8.8, A.5.7
18	Cookie without SameSite Attribute	2	Low	Configuration Management	A.8.9, A.8.18

Source (s): (Atqan et al., 2024; Ho et al., 2024; ISO, 2022)

This table 8 is a comprehensive representation of potential vulnerabilities in Campus Ready system and their alignment with the ISO/IEC 27001:2022 standards for information security management in order to determining how these vulnerabilities can be managed and mitigated according to the ISO/IEC 27001:2022 standards. It will help with risk assessment, prioritization and compliance in order to ensure that a systematic and structured approach to information security. This ultimately strengthens the security framework of the system and mitigate the potential for security incidents.

Configuration Management

- **8.4 Access to the source code**
  To ensure controlled access to the platform's source code, preventing unauthorized modifications and securing the integrity of configurations (Laine, 2024; ISO,2022).
- **5.37 Documented operating procedures**
  To establishe clear guidelines and operational standards for managing configurations across systems (Ho et al., 2024; ISO, 2022).
- **8.9 Configuration management**
  To directly addresses maintaining secure and consistent configurations across all components of the system (Laine, 2024; ISO, 2022).
- **8.18 Use of utilities with privileged rights**
  To monitor and restrict the use of utilities with elevated privileges to prevent unauthorized or accidental configuration changes (Ho et al., 2024; ISO, 2022).
- **8.19 Installation of software on systems in operation**
  To regulate how new software is installed or modified in operational environments to avoid misconfigurations (Ho et al., 2024; ISO, 2022).
- **8.24 Use of cryptography**
  Ensures encryption methods align with best practices for securing configuration files and data exchanges (Ho et al., 2024; ISO, 2022).
- **8.8 Handling technical vulnerabilities**
  To incorporate addressing configuration-related vulnerabilities as part of overall technical vulnerability management (Laine, 2024; ISO, 2022)

Patch Management

- **5.37 Documented operating procedures**
  To provide step-by-step guidelines for applying patches consistently and securely (Ho et al., 2024; ISO, 2022).
- **8.8 Handling technical vulnerabilities**
  To include managing vulnerabilities addressed by patches, ensuring they are tested and applied promptly (Laine, 2024; ISO,2022).
- **8.19 Installation of software on systems in operation**
  To focuse on the secure deployment of patches and updates to avoid disruptions in operational systems (Ho et al., 2024; ISO, 2022).

Vulnerability Management

- **5.37 Documented operating procedures**
  To establishe a standard process for identifying, assessing, and addressing vulnerabilities (Laine, 2024; ISO, 2022).
- **8.8 Handling technical vulnerabilities**
  To focuse on identifying, prioritizing, and mitigating vulnerabilities through proactive measures (Laine, 2024; ISO, 2022).
- **5.7 Information on the threat situation**
  To ensure that vulnerability management strategies are informed by up-to-date threat intelligence (Laine, 2024; ISO, 2022).

5.3. Second Step: An Effective Patch Management Strategy

Patching is essential for addressing vulnerabilities before attackers have the chance to exploit them. Campus Ready is struggling with timely patching, especially for high-severity vulnerabilities, and delays or incomplete patching significantly increase exposure to risks. Therefore, ISO/IEC 27001:2022 can highlight key controls to address this challenge (ISO, 2022; Dissanayake et al., 2022).

ISO/IEC 27001:2022 control A.8.8 will handle technical vulnerabilities emphasizes systematic identification, prioritization and remediation of vulnerabilities, including timely patch management. Also, A.8.19 which means Installation of software on systems in operation will ensure that patching processes are carefully controlled to avoid introducing new risks during deployment. Furthermore, another control A.5.37 which means Documented operating procedures will support the establishment of clear, consistent processes for managing patches effectively (ISO, 2022; Dissanayake et al., 2022; Ho et al., 2024).

Policy

To boost the security posture of the Campus Ready e-Assessment platform, implementing a detailed patch management strategy is crucial. Therefore, the first step involves creating a comprehensive patch management policy. This policy should outline clear timelines for addressing vulnerabilities based on their criticality. For example, critical vulnerabilities should be patched within 24 - 48 hours to minimize exposure to severe risks. High-risk vulnerabilities should be addressed within 7 days, balancing security needs with operational demands. Medium and low-risk vulnerabilities can be patched within 30 days, allowing time for thorough testing and resource management (Kemendi & Pal Michelberger, 2024; Jhala, 2014; Australian Cyber Security Centre, 2010).

This structured approach has been established at below **Table 9 Patch Management Areas** to ensure timely remediation while minimizing disruptions to operations.

Table 9: Patch Management Areas

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	KPI
8	TCP Timestamps Information Disclosure	2,6	Low	Patch Management	In 30 days
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	Patch Management	In 30 days
10	Open TCP Port: 22	5	Medium	Patch Management	In 30 days
11	Open TCP Port: 443	3,5	Low	Patch Management	In 30 days
12	Open TCP Port: 80	3	Low	Patch Management	In 30 days
15	Server Software and Technology Found	2,5	Low	Patch Management	In 30 days

Source (s): Arthur`s own work

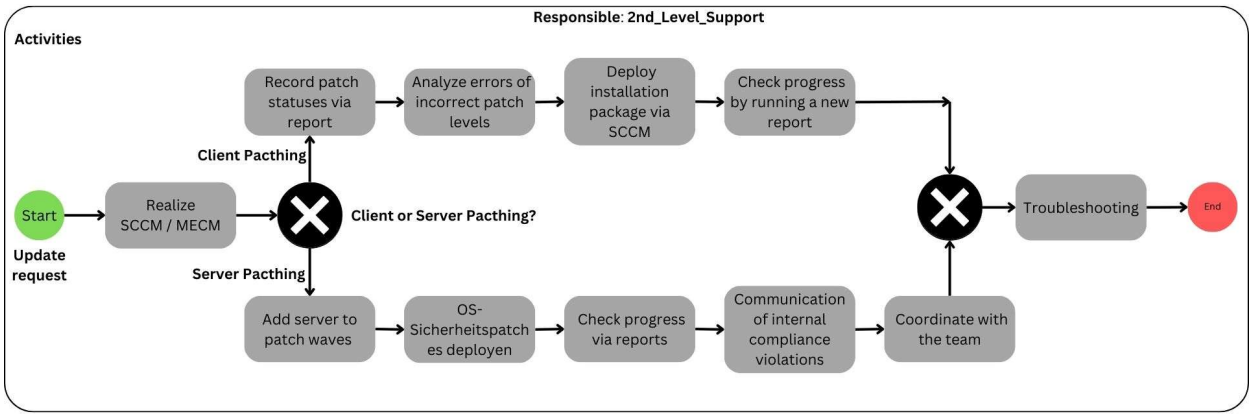


Figure 9: Patch Management Process (Procedure)

Source (s): Arthur`s own work

The patch management procedure outlined in Figure 9, utilizing SCCM (System Center Configuration Manager) and MECM (Microsoft Endpoint Configuration Manager), should be implemented to ensure that Campus Ready systems remain updated and compliant with internal policies, such as ISO/IEC 27001:2022. The procedure begins with recording current patch statuses via reports and categorizing devices into patch waves based on criticality and downtime availability. Depending on whether it's client or server patching, appropriate activities are planned, such as deploying installation packages through SCCM and MECM and monitoring progress via reports (Cooper, 2023).

Errors or failures should be analyzed and troubleshooting is coordinated with the 2nd-Level Support team to resolve issues. Compliance violations are communicated to stakeholders and progress is verified through updated reports to ensure successful deployment. The process concludes with finalizing compliance documentation and closing the patch wave, ensuring all systems meet security requirements (ISO, 2022).

Automation Tools

Automation can play a vital role in streamlining the patch management process. Tools such as MECM and SCCM can automate the deployment of patches, reducing reliance on manual processes that are prone to errors. By leveraging automation, Campus Ready can ensure consistent and timely application of patches across all systems, regardless of their size or complexity, because this is not only saving time and resources but also enhances the reliability and scalability of the patch management process (Cooper, 2023; Roumeliotis, 2022).

Validation

Validation is another critical component of the solution. After deploying patches, conducting vulnerability scans using tools like OpenVAS or OWASP ZAP or Nmap will help Campus Ready to ensure that the patches have been successfully applied and that no residual vulnerabilities remain (Shahid et al., 2022; Jhala, 2014). Since, this step is essential for maintaining confidence in the platform's security. Additionally, tracking patching progress against predefined Service Level Agreements (SLAs) will help to measure the effectiveness of the patch management strategy and identifies areas for continuous improvement (Yang et al., 2011).

5.4. Third: Vulnerability Management Framework

Routine identification and remediation of vulnerabilities are critical to maintaining a secure the platform. A lack of structured vulnerability management processes often leads to recurring risks. By leveraging these risks, ISO/IEC 27001:2022 can outline several controls to address these challenges, such as A.8.8 Handling technical vulnerabilities which emphasizes systematic identification, assessment and remediation of vulnerabilities to reduce exposure (ISO, 2022; Laine, 2024).

To support this, A.5.37 Documented operating procedures will help to ensure that vulnerability management practices are standardized and consistently applied, while A.5.7 Information on the threat situation highlights the importance of staying informed about emerging threats to proactively address risks. By implementing these controls, organizations can effectively mitigate vulnerabilities and maintain a secure environment (ISO, 2022; Laine, 2024).

Table 10: Vulnerability Management Areas

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	KPI
6	Server Leaks Version Information via 'Server' Header	3	Low	Vulnerability Management	In 30 days
16	Security.txt file is missing	5	Medium	Vulnerability Management	In 30 days
17	HTTP OPTIONS Enabled	5	Medium	Vulnerability Management	In 30 days

Source (s): Arthur`s own work

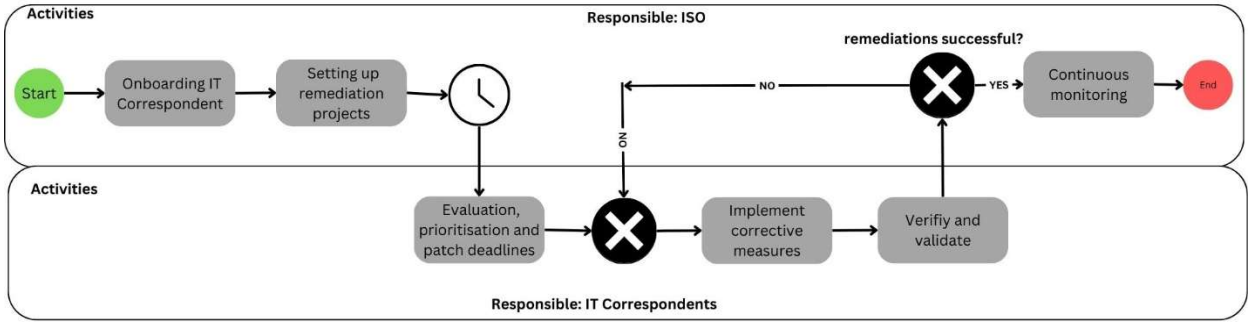


Figure 10: Vulnerability Management Process (Procedure)

Source (s): Arthur`s own work

Vulnerability Management Process (Procedure) has been created in the **Figure 10** as explained more in detail at below, since this procedure is important for ensuring the security, reliability and compliance of the Campus Ready platform. It provides a proactive, structured approach to managing vulnerabilities in order to decrease risks and fostering trust among users and stakeholders of Campus Ready.

Onboarding IT Correspondent

Campus Ready should onboard IT Correspondents by setting up access to tools like Nmap, OpenVAS or OWASP ZAP, depending on the platform's requirements (Shahid et al., 2022). The IT Correspondent should receive training on handling these tools to effectively scan for and manage vulnerabilities. During onboarding, the Correspondent's access should be activated for their assigned location and they should be guided through logging in securely. This process ensures the Correspondent is equipped to start monitoring and addressing vulnerabilities across the Campus Ready platform (Shahid et al., 2022).

Setting Up Remediation Projects

Campus Ready should establish remediation projects to address vulnerabilities identified during scans. These projects should include a detailed plan for fixing vulnerabilities, specifying actions such as applying patches, reconfiguring settings or removing insecure software. The vulnerabilities should be prioritized based on their severity and the criticality of the systems they affect. Progress on these projects should be tracked using reports generated by tools like Nmap, OpenVAS, or OWASP ZAP to ensure timely resolution (Shahid et al., 2022; NVD – NIST Home, n.d.).

Evaluation, Prioritization and Patch Deadlines

The IT Correspondent should analyze the scan results from the selected tool to assess the vulnerabilities identified in the Campus Ready platform. Vulnerabilities should be prioritized using a risk-based scoring system, such as CVSS, to focus on the most critical problems first. A structured approach should be used to categorize vulnerabilities into critical, high, medium, or minor, and patch deadlines should be set accordingly. For instance, critical vulnerabilities should have short remediation windows to minimize exposure (Shahid et al., 2022; NVD – NIST Home, n.d.).

Implementing Corrective Measures

Corrective measures/actions to address vulnerabilities should be implemented based on the remediation plan. These measures might include deploying patches, reconfiguring security settings or removing software flagged as vulnerable. Campus Ready should ensure these actions are performed promptly and efficiently to decrease the risk of exploitation and maintain the platform's integrity (Shahid et al., 2022, Ho et al., 2024).

Verify and Validate

Once corrective actions are implemented, Campus Ready should conduct follow-up scans using the same tool to verify that the vulnerabilities have been successfully addressed. These scans will confirm reduced risk scores or the absence of the previously identified issues. The organization should validate that the implemented measures/actions are effective and ensure continuous improvement in the platform's security (Shahid et al., 2022, Ho et al., 2024).

Continuous Monitoring

Campus Ready should adopt continuous monitoring practices by leveraging the remediation dashboards and reports provided by tools like Nmap, OpenVAS, or OWASP ZAP. Regular scans should be scheduled weekly, monthly or after significant system changes to detect new vulnerabilities and ensure resolved issues do not recur. This proactive approach will

strengthen the platform's resilience against emerging threats (Shahid et al., 2022; Ho et al., 2024; Jhala, 2014).

5.5. Fourth: Strengthen Configuration Management

The "Misconfigurations" are a common root cause of security breaches in the Campus Ready Platform. Therefore, Configuration management strategy should be implemented to ensure that all systems are set up securely and consistently, reducing the risk of exposure. Therefore, ISO/IEC 27001:2022 can provide several controls to mitigate these risks, such as A.8.9 Configuration management which emphasizes secure and consistent system setup and A.8.4 Access to the source code**,** ensuring that access is restricted and monitored to prevent unauthorized changes (ISO, 2022; Laine, 2024; Ho et al., 2024).

Additionally, A.5.37 Documented operating procedures will support maintaining standardized practices, while A.8.18 Use of utilities with privileged rights restricts tools that could be exploited to escalate privileges. To safeguard operational systems, A.8.19 Installation of software on systems in operation can enforce controls on software deployment. The standard also addresses secure data handling through A.8.24 Use of cryptography**,** ensuring data protection and A.8.8 Handling technical vulnerabilities**,** which calls for proactive identification and mitigation of vulnerabilities (ISO, 2022; Laine, 2024; Ho et al., 2024).

Table 11: Configuration Management Areas

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	KPI
1	Absence of Anti-CSRF Tokens	5	Medium	Configuration Management	In 30 days
2	Missing Anti-clickjacking Header	5	Medium	Configuration Management	In 30 days
3	Content Security Policy (CSP) Header Not Set	7,5	High	Configuration Management	In 7 days
4	X-Content-Type-Options Header Missing	2	Low	Configuration Management	In 30 days
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	Configuration Management	In 30 days
7	Strict-Transport-Security Header Not Set	5,5	Medium	Configuration Management	In 30 days
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	Configuration Management	In 30 days
14	Missing security header: Referrer-Policy	5	Medium	Configuration Management	In 30 days
18	Cookie without SameSite Attribute	2	Low	Configuration Management	In 30 days

Source (s): (ISO, 2022; Laine, 2024; Ho et al., 2024).

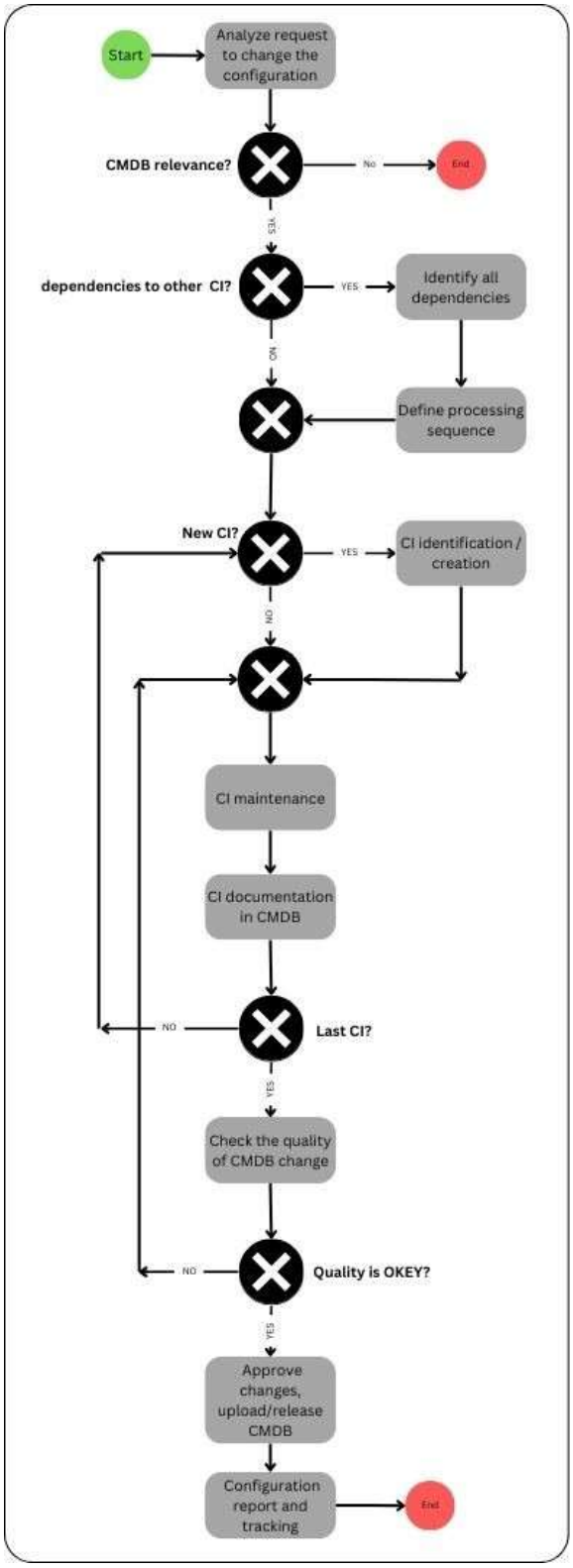


Figure 11: Configuration Management Process (Procedure)

Source (s): Arthur`s own work

Campus Ready requires a robust and streamlined configuration management procedure to address evolving cybersecurity challenges and maintain a secure and compliant infrastructure. The process integrates detailed procedural steps with standardized practices, automation and monitoring to create a comprehensive framework.

Analyze Change Requests

All configuration change requests should be analyzed to ensure alignment with Campus Ready's security objectives and operational needs. This step evaluates potential impacts on existing systems to ensure minimal disruption and adherence to security protocols (ISO, 2022; Ho et al., 2024).

Dependency Identification and Impact Analysis

Dependencies between Configuration Items (CIs) are identified to assess how changes might affect other components. If new CIs are required, they should be created and documented. This analysis will ensure seamless integration of changes and reduces the risk of unintended consequences (Tawo & Ajayi, 2025).

Define and Standardize Processing Sequence

A clear sequence for processing configuration changes is established. Standardization is achieved through the adoption of CIS Benchmarks, which provide best practices for configuring servers, databases, and applications. For example, mandatory HTTPS usage and disabling unnecessary services minimize vulnerabilities and create a consistent security posture across the platform (Kepuska & Tomasevic, 2024; Laine, 2024).

CI Creation, Maintenance and Documentation

Each CI (Configuration Item) is created or updated as needed, ensuring relevance to the Configuration Management Database (CMDB). Proper documentation of configurations and their dependencies ensures that all changes are transparent and traceable (MEDUPE, 2009).

Quality Assurance and Approval

Before releasing updates, a quality check of CMDB entries ensures adherence to security baselines and operational requirements. Once verified, changes are approved and released, updating the CMDB to reflect the new configurations (MEDUPE, 2009).

Automation for Consistency and Efficiency

Automation through Infrastructure as Code (IaC) tools such as Ansible, Puppet or Terraform ensures secure configurations are deployed consistently across all environments. For instance, Terraform can enforce critical SSH settings like disabling password-based login, reducing manual errors and accelerating implementation (Baushke & Bider, 2012; Hasan et al., n.d.).

Continuous Monitoring and Drift Detection (Incident Management)

CMDBs and drift detection tools continuously monitor configuration states, identifying unauthorized or non-compliant changes. For example, if an unapproved service is enabled or a port is opened, the system can alert administrators or automatically revert to secure settings. This ensures that configurations remain aligned with Campus Ready’s security baselines (MEDUPE, 2009).

Reporting and Tracking

A configuration report should be generated to document changes, enabling ongoing tracking and accountability. Since, this supports compliance efforts and provides a clear record for audits and reviews (MEDUPE, 2009).

5.6. Fifth: Establishing a Metrics-Driven Approach (KPIs)

To measure the success of implemented strategies, Campus Ready needs to track key performance indicators (KPIs) that align with ISO/IEC 27001:2022 objectives. Since this will help Campus Ready to ensure that the strategies are effective and facilitates continuous improvement (Kemendi & Pal Michelberger, 2024).

Defining Metrics

Firstly, one critical KPI is the Percentage of Systems Patched Within SLA (Service Level Agreement). Since this metric evaluates the timeliness of addressing vulnerabilities by measuring the proportion of systems updated within agreed timelines, such as patching critical vulnerabilities within 48 hours or high-risk ones within 7 days or Medium and Low risks within 30 days. Meeting these timelines is crucial for minimizing exposure to potential exploits in order to ensure that Campus Ready's e-Assessment platform remains secure during digitalization/software upgrades (Kemendi & Pal Michelberger, 2024; Jhala, 2014; Australian Cyber Security Centre, 2010).

Another vital metric is the number of critical vulnerabilities found and remediated within timelines. Since this KPI reflects the organization's efficiency in detecting and resolving the most pressing security risks. Tracking this number will ensure that critical vulnerabilities are not only identified but also effectively addressed, reducing the likelihood of data breaches or service disruptions (Kemendi & Pal Michelberger, 2024; Jhala, 2014).

A third metric, frequency of configuration drift and time taken to rectify, assesses how often systems deviate from their secure baseline configurations and how quickly these deviations are corrected. Frequent or unresolved configuration drift can introduce vulnerabilities, making it imperative to monitor and mitigate such instances promptly.

Table 12: KPIs For Patching Durations

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	ISO/IEC 27001:2022 Control Nr.	KPI
1	Absence of Anti-CSRF Tokens	5	Medium	A.8.9, A.8.18	In 30 days
2	Missing Anti-clickjacking Header	5	Medium	A.8.9, A.8.18	In 30 days
3	Content Security Policy (CSP) Header Not Set	7,5	High	A.8.9, A.8.24	In 7 days
4	X-Content-Type-Options Header Missing	2	Low	A.8.9, A.8.18	In 30 days
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	A.8.9, A.8.18	In 30 days
6	Server Leaks Version Information via 'Server' Header	3	Low	A.8.8, A.5.7	In 30 days
7	Strict-Transport-Security Header Not Set	5,5	Medium	A.8.9, A.8.24	In 30 days
8	TCP Timestamps Information Disclosure	2,6	Low	A.8.8, A.8.19	In 30 days
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	A.8.4,A.8.8, A.8.19	In 30 days
10	Open TCP Port: 22	5	Medium	A.8.4,A.8.8, A.8.19	In 30 days
11	Open TCP Port: 443	3,5	Low	A.8.8, A.8.19	In 30 days
12	Open TCP Port: 80	3	Low	A.8.8, A.8.19	In 30 days
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	A.8.9, A.8.18	In 30 days
14	Missing security header: Referrer-Policy	5	Medium	A.8.9, A.8.18	In 30 days
15	Server Software and Technology Found	2,5	Low	A.8.8, A.8.19	In 30 days
16	Security.txt file is missing	5	Medium	A.8.8, A.5.37	In 30 days
17	HTTP OPTIONS Enabled	5	Medium	A.8.8, A.5.7	In 30 days
18	Cookie without SameSite Attribute	2	Low	A.8.9, A.8.18	In 30 days

Source (s): Arthur`s own work**;** Jhala, 2014; ISO, 2022).

Reporting

Regularly publishing security performance reports for internal and external stakeholders can reinforce transparency and accountability. These reports should include trends in KPI performance, for example, the percentage of timely patches applied or the frequency of configuration drifts rectified. They also provide an opportunity to showcase progress, highlight lessons learned from past vulnerabilities and demonstrate a commitment to continuous improvement. For Campus Ready, these reports can foster trust among users and partners, showing that the platform prioritizes security and resilience (ISO, 2022).

Benchmarking

Benchmarking is another important solution for Campus Ready's performance against industry standards and peer organizations allows for meaningful comparisons. For instance, industry benchmarks may indicate a 95% patch success rate within SLAs, serving as a target for Campus Ready to achieve or surpass. In addition, benchmarking will help to highlight areas where Campus Ready excels and identifies opportunities for adopting advanced tools or refining processes. It will ensure that the organization's cybersecurity efforts align with best practices and remain competitive in the educational technology sector (Kemendi & Pal Michelberger, 2024; Jhala, 2014; Australian Cyber Security Centre, 2010).

5.7. Sixth: Continuous Improvement Cycle

Maintaining a robust security posture requires a commitment to ongoing refinement and adaptation. ISO/IEC 27001:2022 advocates the **Plan-Do-Check-Act (PDCA) cycle** which is a structured framework for continuous improvement. This cycle can help Campus Ready to ensure that security strategies evolve in response to emerging threats and changing business requirements, reinforcing Campus Ready's ability to safeguard its e-Assessment platform (Ganji et al., 2020; Laurence, 2023).

5.7.1. Do: Implementation and Tracking

The "Do" phase has been created in the Table 13 at shown below which focuses on executing remediation measures and closely monitoring their effectiveness. This involves deploying updates, addressing vulnerabilities identified during audits and training staff on updated protocols. Tools such as automated patching systems or configuration management solutions can enhance implementation efficiency. Tracking the outcomes of these measures ensures that solutions are delivering the intended security improvements, enabling Campus Ready to protect sensitive data and platform functionality (Ganji et al., 2020; Krisztián & Cism, 2013; Cohen,2012).

Table 13: Categorization of Vulnerabilities by Management Area and Recommended Actions (Measures)

Vul.ID	Type of Vulnerability	Severity Score	Severity Level	Management Area	ISO/IEC 27001:2022 Control Nr.	Recommended Actions
1	Absence of Anti-CSRF Tokens	5	Medium	Configuration Management	A.8.9, A.8.18	Implement Anti-CSRF tokens to secure HTML submission forms.
2	Missing Anti-clickjacking Header	5	Medium	Configuration Management	A.8.9, A.8.18	Implement the Content-Security-Policy or X-Frame-Options header to safeguard against Clickjacking attacks.
3	Content Security Policy (CSP) Header Not Set	7,5	High	Configuration Management	A.8.9, A.8.24	Set up CSP headers to protect against XSS and data injection attacks.
4	X-Content-Type-Options Header Missing	2	Low	Configuration Management	A.8.9, A.8.18	Configure the X-Content-Type-Options header to 'nosniff' to avoid MIME-sniffing vulnerabilities.
5	Cross-Domain JavaScript Source File Inclusion	2,5	Low	Configuration Management	A.8.9, A.8.18	Restrict external JavaScript sources and validate their inclusion.
6	Server Leaks Version Information via 'Server' Header	3	Low	Vulnerability Management	A.8.8, A.5.7	Eliminate or obscure version information in server response headers to reduce exposure to targeted attacks.
7	Strict-Transport-Security Header Not Set	5,5	Medium	Configuration Management	A.8.9, A.8.24	Activate HTTP Strict Transport Security (HSTS) to ensure the use of secure HTTPS connections.
8	TCP Timestamps Information Disclosure	2,6	Low	Patch Management	A.8.8, A.8.19	Disable TCP timestamps to prevent uptime information disclosure.
9	Weak MAC Algorithm(s) Supported (SSH)	2,6	Low	Patch Management	A.8.4,A.8.8, A.8.19	Reconfigure SSH settings to disable weak MAC algorithms.
10	Open TCP Port: 22	5	Medium	Patch Management	A.8.4,A.8.8, A.8.19	Restrict access or apply updates to secure services running on port 22.
11	Open TCP Port: 443	3,5	Low	Patch Management	A.8.8, A.8.19	Monitor and secure services running on port 443.
12	Open TCP Port: 80	3	Low	Patch Management	A.8.8, A.8.19	Monitor and update services running on port 80.
13	Insecure Cookie Setting: Missing Secure Flag	2,5	Low	Configuration Management	A.8.9, A.8.18	Set the Secure attribute for cookies to ensure encrypted transmission.
14	Missing security header: Referrer-Policy	5	Medium	Configuration Management	A.8.9, A.8.18	Add Referrer-Policy header to control sensitive URL information sharing.
15	Server Software and Technology Found	2,5	Low	Patch Management	A.8.8, A.8.19	Regularly update server software to prevent exploitation of known vulnerabilities.
16	Security.txt file is missing	5	Medium	Vulnerability Management	A.8.8, A.5.37	Include a security.txt file to establish a dedicated channel for reporting security issues.
17	HTTP OPTIONS Enabled	5	Medium	Vulnerability Management	A.8.8, A.5.7	Disable unnecessary HTTP methods to reduce attack surface.
18	Cookie without SameSite Attribute	2	Low	Configuration Management	A.8.9, A.8.18	Include the SameSite attribute in cookies to protect against cross-site request forgery (CSRF).

Source (s): Arthur`s own work**;** Jhala, 2014; ISO, 2022).

Lack of Anti-CSRF Tokens

The lack of Anti-CSRF tokens, which carries a medium severity score of 5, exposes the Campus Ready e-Assessment platform to cross-site request forgery (CSRF) attacks. As part of Configuration Management and in line with ISO/IEC 27001:2022 controls A.8.9 and A.8.18, Campus Ready should integrate Anti-CSRF tokens into HTML submission forms to validate and authorize requests, thereby reducing this risk (OWASP, 2012; Laine, 2024; ISO, 2022; Ho et al., 2024).

Missing Anti-clickjacking Header

This vulnerability, with a medium severity score of 5, indicates insufficient protection against clickjacking attacks. Aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.18 under Configuration Management, Campus Ready should add Content-Security-Policy or X-Frame-Options headers to prevent malicious framing of the platform's web applications (Mozilla, n.d.; Laine, 2024; ISO, 2022; Ho et al., 2024)

Content Security Policy (CSP) Header Not Set

With a high severity score of 7.5, this critical vulnerability increases the risk of XSS and data injection attacks. As a Configuration Management issue aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.24, Campus Ready should configure CSP headers to restrict executable content sources, significantly reducing exposure to these attacks (World Wide Web Consortium (W3C), n.d.; Laine, 2024; ISO, 2022; Ho et al., 2024)

X-Content-Type-Options Header Missing

This low-severity vulnerability, scored at 2, exposes the platform to MIME-sniffing risks. Falling under Configuration Management and aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.18, Campus Ready should set the X-Content-Type-Options header to "nosniff," ensuring browsers respect declared content types (OWASP, n.d.; Laine, 2024; ISO, 2022; Ho et al., 2024)

Cross-Domain JavaScript Source File Inclusion

With a low severity score of 2.5, this vulnerability exposes the platform to risks from unvalidated external JavaScript sources. Mapped to ISO/IEC 27001:2022 controls A.8.9 and A.8.18 under Configuration Management, Campus Ready should restrict and validate external JavaScript sources to minimize exposure to malicious scripts (Shaikh, 2020; Laine, 2024; ISO, 2022; Ho et al., 2024).

Server Leaks Version Information via 'Server' Header

This low-severity vulnerability, scored at 3, provides attackers with version information that can be exploited. As a Vulnerability Management concern aligned with ISO/IEC 27001:2022 controls A.8.8 and A.5.7, Campus Ready should remove or obfuscate server header version information to reduce this risk (Microsoft "Archiveddocs", 2010; Laine, 2024; ISO, 2022)

Strict-Transport-Security Header Not Set

With a medium severity score of 5.5, this vulnerability leaves the platform without enforced HTTPS connections. Falling under Configuration Management and aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.24, Campus Ready should enable HTTP Strict Transport Security (HSTS) to ensure secure browser communications (OWASP, n.d.; Laine, 2024; ISO, 2022; Ho et al., 2024).

TCP Timestamps Information Disclosure

This low-severity vulnerability, scored at 2.6, exposes server uptime information. As a Patch Management issue mapped to ISO/IEC 27001:2022 controls A.8.8 and A.8.19, Campus Ready should disable TCP timestamps to limit reconnaissance opportunities for attackers (Borman, Braden and Jacobson, 1992; Borman et al., 2014; Laine, 2024; ISO, 2022; Ho et al., 2024).

Weak MAC Algorithm(s) Supported (SSH)

With a low severity score of 2.6, this vulnerability compromises encryption strength. Falling under Patch Management and aligned with ISO/IEC 27001:2022 controls A.8.8 and A.8.19, Campus Ready should reconfigure SSH settings to disable weak MAC algorithms, strengthening the platform’s security (Baushke & Bider, 2012; Laine, 2024; ISO, 2022; Ho et al., 2024).

Open TCP Port: 22

This medium-severity vulnerability, scored at 5, could allow unauthorized access. Aligned with ISO/IEC 27001:2022 controls A.8.8 and A.8.19 under Patch Management, Campus Ready should restrict access and ensure updates are applied to secure services running on this port (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020; Laine, 2024; ISO, 2022; Ho et al., 2024).

Open TCP Port: 443

With a low severity score of 3.5, this vulnerability poses potential risks if services are outdated. Falling under Patch Management and aligned with ISO/IEC 27001:2022 controls A.8.8 and A.8.19, Campus Ready should monitor and secure services running on port 443 to prevent exploitation (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020; Laine, 2024; ISO, 2022; Ho et al., 2024).

Open TCP Port: 80

A low-severity vulnerability, scored at 3, may expose services to risks if improperly managed. Under Patch Management and aligned with ISO/IEC 27001:2022 controls A.8.8 and A.8.19, Campus Ready should regularly monitor and update services running on this port (Joost Oortwijn & Gañán, 2024; Rashidi et al., 2020; Laine, 2024; ISO, 2022; Ho et al., 2024)

Insecure Cookie Setting: Missing Secure Flag

This low-severity vulnerability, scored at 2.5, exposes cookies to risks during transmission. Falling under Configuration Management and aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.18, Campus Ready should set the Secure attribute for cookies to ensure encrypted transmissions (OWASP, n.d.; Laine, 2024; ISO, 2022; Ho et al., 2024)

Missing Security Header: Referrer-Policy

With a medium severity score of 5, this issue risks sensitive URL information exposure. Under Configuration Management and mapped to ISO/IEC 27001:2022 controls A.8.9 and A.8.18, Campus Ready should add the Referrer-Policy header to control information sharing and enhance user privacy (developer.mozilla.org, n.d. ; Laine, 2024; ISO, 2022; Ho et al., 2024).

Server Software and Technology Found

This low-severity vulnerability, scored at 2.5, is related to outdated server software. Falling under Patch Management and aligned with ISO/IEC 27001:2022 controls A.8.8 and A.8.19, Campus Ready should regularly update server software to prevent exploitation of known vulnerabilities (OWASP, 2019; Laine, 2024; ISO, 2022; Ho et al., 2024).

Security.txt File is Missing

This medium-severity vulnerability, scored at 5, hinders the reporting of vulnerabilities. Aligned with ISO/IEC 27001:2022 controls A.8.8 and A.5.37 under Vulnerability

Management, Campus Ready should add a security.txt file to facilitate responsible vulnerability reporting (CISA, 2023)

HTTP OPTIONS Enabled

With a medium severity score of 5, this vulnerability increases the attack surface. Falling under Vulnerability Management and mapped to ISO/IEC 27001:2022 controls A.8.8 and A.5.7, Campus Ready should disable unnecessary HTTP methods to reduce exposure (Nedim, 2020; Laine, 2024; ISO, 2022).

Cookie Without SameSite Attribute

This low-severity vulnerability, scored at 2, risks CSRF attacks. Falling under Configuration Management and aligned with ISO/IEC 27001:2022 controls A.8.9 and A.8.18, Campus Ready should set the SameSite attribute for cookies to safeguard against cross-site request forgery (CSRF) attacks (West & Goodwin, 2016; Nginx, 2024; Laine, 2024; ISO, 2022; Ho et al., 2024).

5.7.2. Check: Evaluation through Audits

Regular internal audits and penetration tests should be conducted by Campus Ready (IT security specialist) during the "Check" phase to evaluate the security posture and uncover new risks. Since, these audits can provide a comprehensive review of implemented strategies, verifying compliance with ISO/IEC 27001:2022 and ensuring that identified vulnerabilities have been mitigated or not. Also, Penetration test should be used to replicate real-world attack scenarios and uncover potential vulnerabilities which can offer Campus Ready actionable insights into areas needing further attention (Ganji et al., 2020; Krisztián & Cism, 2013; Cohen,2012).

In addition to internal measures, external audits and penetration testing should form an integral part of Campus Ready’s security strategy to ensure comprehensive risk evaluation and compliance. However, external audits can be conducted by independent third-party professionals to provide an objective assessment of the platform’s adherence to ISO/IEC 27001:2022 standards. Therefore, these audits validate the effectiveness of implemented strategies, identify overlooked vulnerabilities and ensure that all security measures align with industry best practices (Ganji et al., 2020; Krisztián & Cism, 2013; Cohen,2012).

Penetration tests can be performed by external security experts, simulate real-world attack scenarios that go beyond traditional assessments. These tests can reveal potential weaknesses in the system that internal teams may have missed, offering actionable

insights for improving the platform's resilience. For Campus Ready, penetration tests will be particularly valuable in identifying exploitable vulnerabilities in unpatched software, misconfigurations and integration points during software upgrades processes (Ganji et al., 2020; Krisztián & Cism, 2013; Cohen,2012).

Since the combination of external audits and penetration tests can help Campus Ready maintain a robust and proactive security posture. These measures complement internal reviews, fostering continuous improvement, stakeholder trust and alignment with global security standards (Knowles et al., 2016).

5.7.3. Act: Refining Strategies

The final stage, "Act," involves updating and refining security strategies based on findings from audits and lessons learned. This iterative process will help Campus Ready to ensure that any gaps or inefficiencies identified are addressed promptly. For Campus Ready, this stage should involve revising patch management timelines, enhancing staff training or adopting new tools to streamline vulnerability management. By embedding this adaptability, the organization will build resilience against both current and future threats (Ganji et al., 2020; Krisztián & Cism, 2013; Cohen,2012).

Table 14: Refining Security Strategies in the 'Act' Phase

Stage	Description	Key Actions	Outcome
Act	Updating and refining security strategies based on findings from audits and lessons learned.	Analyze audit findings and lessons learned from incidents.	Improved and more effective security strategies.
		Identify gaps and areas needing improvement in existing security controls and procedures.	Enhanced alignment with security frameworks and standards.
		Update security policies, procedures, and configurations accordingly.	Strengthened organizational resilience against evolving threats.
		Communicate changes and provide training to relevant stakeholders.	Increased awareness and adherence to updated security measures.
		Monitor the effectiveness of implemented updates through ongoing testing and audits.	Continuous improvement in security posture and compliance with standards like ISO/IEC 27001.

Source (s): Arthur`s own work

6. Validation of Research Questions & Hypothesis

6.1. Research Questions

The thesis will help to answer and prove the following Research Questions and Hypohesis:

RQ 1: How does unpatched software during software upgrades lead to specific cyber vulnerabilities in the Campus Ready e-Assessment platform (Software), such as data breaches and malware attacks?

According to founded results (check the section Findings section), it is clear to see unpatched software significantly contributes to vulnerabilities across multiple dimensions. The Campus Ready e-Assessment platform's software upgrades processes exacerbate this issue as older software components interact with newer systems and introducing exploitable gaps. Vulnerabilities such as the absence of security headers (e.g., Content-Security-Policy, Strict-Transport-Security) and open TCP ports (e.g., 22, 443, 80) provide entry points for unauthorized access and data interception. These vulnerabilities expose sensitive data to threats like session hijacking and cross-site scripting (XSS) and directly correlating to risks like data breaches and malware propagation. Moreover, legacy systems supporting weak cryptographic protocols (e.g., outdated MAC algorithms) further compromise secure communication channels and creating additional risks for sensitive data and system integrity.

RQ 2: How can an effective patch, vulnerability and configuration management strategy be recommended to address and mitigate the risks associated with unpatched software in the Campus Ready e-Assessment platform, in accordance with ISO/IEC 27001:2022 guidelines?

The implementation of a structured strategy (Please check the implementation section) leveraging ISO/IEC 27001:2022 controls is vital for mitigating these risks. The study suggests prioritizing patch management through tools like MECM and SCCM to automate and optimize the patching process, ensuring prompt resolution of high-risk vulnerabilities. Configuration management, guided by standards such as CIS Benchmarks to ensure consistent security across servers, applications and databases. Vulnerability management tools like OpenVAS, combined with regular scans and audits to ensure proactive detection and mitigation of risks. Utilizing metrics such as patch SLA adherence and tracking configuration drift rates provides actionable insights for continuous improvement. These strategies, underpinned by ISO controls like A.8.8 (Handling Technical Vulnerabilities) and A.8.9 (Configuration Management) to enable Campus Ready to address vulnerabilities holistically.

6.2. Hypotheses

H1: Unpatched software during software upgrades significantly increases cybersecurity vulnerabilities in Campus Ready e-Assessment platform (Software) and it is making them more susceptible to attacks such as data breaches and malware infections.

The analysis of this study's results supports this hypothesis approach. For example, unpatched software was identified as a root cause for issues such as absent security headers and open ports. These vulnerabilities amplify the risk of advanced threats like cross-site scripting and man-in-the-middle attacks, making the system more vulnerable to data breaches. The correlation between high-severity vulnerabilities and unpatched systems further emphasizes the critical role of timely updates in mitigating risks.

H2: Implementing effective patch, vulnerability and configuration management strategies, aligned with ISO/IEC 27001:2022 guidelines, is expected to significantly reduce vulnerabilities and enhance the overall security risks posture of the Campus Ready e-Assessment platform.

The findings (Findings Section) validate this hypothesis. The suggested implementation section of ISO-compliant management frameworks focusing on automation, standardization and continuous monitoring addresses the identified vulnerabilities effectively. For example, enabling HTTP Strict Transport Security (HSTS) and regular audits ensure secure data transmission and compliance with best practices. Metrics-driven approaches, such as tracking patching within SLA, demonstrate measurable improvements in security and aligning with ISO/IEC 27001:2022 objectives.

7. LIMITATIONS AND FURTHER RESEARCH

7.1. Limitations

One of the main constraints of this study is the absence of real-time implementation and evaluation of the proposed strategies. While the study provides a detailed roadmap for patch, vulnerability and configuration management aligned with ISO/IEC 27001:2022 guidelines, these strategies were tested in a simulated environment rather than a live operational context. This restricts the ability to observe long-term impacts, operational challenges and unforeseen outcomes that might arise during practical deployment. Additionally, critical factors such as user compliance, system downtime during patching and actual threat mitigation could not be fully assessed and leaving a gap in validating the effectiveness of the recommendations.

On the other hand, the study was conducted within the constraints of limited resources which impacted the comprehensiveness of the analysis. Campus Ready, being a startup, has restricted access to personnel, funding and advanced security infrastructure. Therefore, these constraints influenced the depth of testing, the scope of data collection and the ability to simulate complex attack scenarios. Furthermore, the lack of automation tools for vulnerability management and the absence of a dedicated cybersecurity team resulted in delayed processes and manual interventions, potentially skewing the findings. This limitation suggests that solutions need tobe scalable and capable of being applied in settings with limited resources.

7.2. Further Research

Future studies should concentrate on implementing and assessing the proposed strategies in practical environments specifically within Campus Ready's operational systems to assess their effectiveness, identify operational challenges and enable iterative improvements. Expanding the study to include other threat vectors like phishing, insider threats and third-party integration risks would provide a more comprehensive understanding of security issues. Additionally, exploring the use of automation through AI and machine learning to increase patch management and vulnerability detection could improve response times and reduce human error.

Besides, implementing vulnerability, patch and configuration management strategies to mitigate the risks, also other kind of vulnerability strategies can be also implemented such as Security Incidents Handling, ITIL-Based Change Management and Continuous Vulnerability Scanning (Amalka Peliarachchi & Janaka Wijayanayake, 2023).

Assessing the impact of user training programs on reducing cybersecurity risks would also be valuable in strengthening Campus Ready's security culture. Integrating other frameworks, such as NIST and GDPR, alongside ISO standards could improve compliance and provide more robust protection (Ngalim, 2023). Examining how emerging technologies, like cloud computing and blockchain affect cybersecurity in the context of digital transformation (Software Installation & Upgrades) will help the platform stay adaptable. Lastly, conducting a cost-benefit analysis of the proposed strategies will help prioritize resources effectively to ensure optimal security within financial constraints (Mishra & Saikat Gochhait, 2023).

8. DISCUSSION & CONCLUSION

In this research, it was analyzed the effects of unpatched software vulnerabilities on Campus Ready's e-Assessment platform's cybersecurity in software installation & upgrading in digital transformation. Findings affirmed that unpatched software heavily boosts threats such as including data breaches and functionality disruptions are validating the hypothesis (H1). Further, vulnerability scans identified problems like insecure cookie configuration and absence of anti-CSRF tokens etc., and root cause analysis identified challenges in patch management due to the unavailability of resources and complex IT infrastructure.

To fix the aforesaid issues, the study provided and implemented ISO/IEC 27001:2022-compliant solutions such as Patch Management to automate patches and patching high-impact vulnerabilities and secondly, Vulnerability Management for applying CVSS v4.0-scored risk decisions and active scanning and thirdly, Configuration Management for developing hardened baseline configurations and lastly Continuous Improvement for incorporating a PDCA cycle for continuous improvement of security. These steps adequately stifle risks and advance the cybersecurity position of the platform.

This research recognized the importance of proactive cybersecurity to e-Assessment platforms. Unpatched applications create risks to data security and platform stability. The implementation of ISO/IEC 27001:2022-compliant methods, such as the adoption of automated patch management and proper configuration practices, is a great step in reducing such risks.

The above findings are informative to Campus Ready and other such platforms to prioritize the importance of structured cybersecurity interventions to digital transformation (Software Installation & Upgrades).

REFERENCES

Abdulghaffar, K., Elmrabit, N., & Yousefi, M. (2023). Enhancing Web Application Security through Automated Penetration Testing with Multiple Vulnerability Scanners. *Computers*, *12*(11), 235. https://doi.org/10.3390/computers12110235

Al-Ansi, A. M., Jaboob, M., Garad, A. & Al-Ansi, A. (2023). Analyzing augmented reality (AR) and virtual reality (VR) recent development in assessment. *Social Sciences & Humanities Open*, *8*(1), 100532. https://doi.org/10.1016/j.ssaho.2023.100532

Amalka Peliarachchi, & Janaka Wijayanayake. (2023). A-ITIL, ITIL and Agile Based Advanced Framework for Managing Software and IT Related Bau: A Systematic Literature Review. 1(1), 84–97. https://doi.org/10.4038/jdrra.v1i1.8

Anžela Jurāne-Brēmane. (2021). The Digital Transformation of Assessment: Challenges and Opportunities. https://doi.org/10.22364/htge.2021.25

Antonio, S.A. (2017, 1. April). *Designing assessmental scenarios to teach network security* IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/7943063

Atqan, A., Rahmat Mulyana, & Ryan Adhitya Nugraha. (2024). Utilizing ISO 27001:2022 In Information Security Design For BPRCCo SME Digital Transformation. *Ranah Research Journal of Multidisciplinary Research and Development*, *6*(6), 2544–2553. https://doi.org/10.38035/rrj.v6i6.1121

Australian Cyber Security Centre. (2010, February 18). *Strategies to Mitigate Cyber Security Incidents | Cyber.gov.au*. Cyber.gov.au. https://www.cyber.gov.au/resources-business-and-government/essential-cyber-security/strategies-mitigate-cyber-security-incidents/strategies-mitigate-cyber-security-incidents

Bandara, Indra; Balakrishna, Chitra and Ioras, F. (2021). The Need For Cyber Threat Intelligence For Distance Assessment Providers And Online Assessment Systems. In: INTED2021 Proceedings, INTED2021 Proceedings, IATED, pp. 9312–9321.

Bäumer, F., Brinkmann, M., & Schwenk, J. (2024). *Terrapin Attack: Breaking SSH Channel Integrity By Sequence Number Manipulation Terrapin Attack: Breaking SSH Channel*

Integrity By Sequence Number Manipulation. https://www.usenix.org/system/files/usenixsecurity24-baumer.pdf

Baushke, M.D. and bider, denis (2012). *SHA-2 Data Integrity Verification for the Secure Shell (SSH) Transport Layer Protocol*. [online] IETF. Available at: https://www.rfc-editor.org/rfc/rfc6668.

Borman, D.A., Braden, R.T. and Jacobson, V. (1992). *TCP Extensions for High Performance*. [online] IETF. Available at: https://datatracker.ietf.org/doc/html/rfc1323.

Borman, D., Braden, R.T., Jacobson, V. and Scheffenegger, R. (2014). *TCP Extensions for High Performance*. [online] IETF. Available at: https://datatracker.ietf.org/doc/html/rfc7323.

Brykczynski, B. & Small, R. (2021). Reducing Internet-based intrusions: Effective security patch management. *IEEE Software*, *20*(1), 50–57. https://doi.org/10.1109/ms.2003.1159029

Calzavara, S., Rabitti, A., & Bugliesi, M. (2018). Semantics-Based Analysis of Content Security Policy Deployment. *ACM Transactions on the Web*, *12*(2), 1–36. https://doi.org/10.1145/3149408

Campus-Ready. (2024). Campus-Ready.com- Home. https://campus-ready.com

Ching-Huang, et al. (2008). A Study and Implementation of Vulnerability Assessment and Misconfiguration Detection. https://doi.org/10.1109/APSCC.2008.212

CISA (2023). *security.txt: A Simple File with Big Value | CISA*. [online] Available at: https://www.cisa.gov/news-events/news/securitytxt-simple-file-big-value.

Cohen, Eli B.(2012).*Issues in Informing Science & Information Technology, Volume 9 (2012)*. (2025). Google Books. https://books.google.de/books?hl=en&lr=&id=p5fnRLNu8ncC&oi=fnd&pg=PA331&dq=PDCA+ISO/IEC+27001&ots=zNrpyIlyn0&sig=h8YM2knFEDnJbPVJwwvLF4oOBX4&redir_esc=y#v=onepage&q

Cooper, A. (2023, June 21). *Mind the Security Gap: Evaluating the Effectiveness of the UK Cyber Essentials Scheme and its Suitability for Large Enterprises*. University of Strathclyde.

https://pureportal.strath.ac.uk/en/publications/mind-the-security-gap-evaluating-the-effectiveness-of-the-uk-cybe

Dempsey, K., Johnson, A., Scholl, M., Stine, K., Department of Defense Chief Information Officer, Defense-wide Information Assurance Program, Booz Allen Hamilton & PricewaterhouseCoopers LLP. (2011). Special Publication 800-137 Information Security Continuous Monitoring for Federal Information Systems and Organizations. In Special Publication 800-137 Information Security Continuous Monitoring for Federal Information Systems and Organizations. https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-137.pdf

Dennis, C. (2018). Why is patch management necessary? Network Security, 2018(7), 9–13. https://doi.org/10.1016/s1353-4858(18)30068-0

Developer.mozilla.org. (n.d.). *Referer header: privacy and security concerns - Web security | MDN*. [online] Available at: https://developer.mozilla.org/en-US/docs/Web/Security/Referer_header:_privacy_and_security_concerns.

Dimitrios Sargiotis. (2024). *Data Security and Privacy: Protecting Sensitive Information*. 217–245. https://doi.org/10.1007/978-3-031-67268-2_6

Dissanayake, N., Jayatilaka, A., Zahedi, M. & Babar, M. A. (2022). Software security patch management - A systematic literature review of challenges, approaches, tools and practices. *Information And Software Technology*, *144*, 106771. https://doi.org/10.1016/j.infsof.2021.106771

Dissanayake, N., Zahedi, M., Jayatilaka, A., & Babar, M. A. (2021). A grounded theory of the role of coordination in software security patch management. *A Grounded Theory of the Role of Coordination in Software Security Patch Management*. https://doi.org/10.1145/3468264.3468595

Divya, Anand, A., Bhatt, N., & Johri, P. (2024). Assessing the Impact of Software Patching on Vulnerabilities: A Comprehensive Framework for Faulty and Safe Patches. *International Journal of Reliability, Quality and Safety Engineering*. https://doi.org/10.1142/s0218539324500311

Greenbone OpenVAS (n.d.). Federal Office for Information Security. https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Freie-Software/Tools/OpenVAS/OpenVAS_node.html

Fertig, T., Schütz, A. E., Weber, K. & Müller, N. H. (2019b). Measuring the Impact of E-Assessment platforms on Information Security Awareness. In *Lecture notes in computer science* (S. 26–37). https://doi.org/10.1007/978-3-030-21814-0_3

Findlay, W., & Abdou, A. (2022). *Characterizing the Adoption of Security.txt Files and their Applications to Vulnerability Notification*. https://doi.org/10.14722/madweb.2022.23014

Flavio Rodrigues Pereira, Fonseca, C., & Maria Inês Correia. (2023). *The application of Business Intelligence methodologies and tools: their role in cybersecurity*. https://doi.org/10.23919/cisti58278.2023.10211279

Florian Plainer, Klaus Kieseberg, & Kieseberg, P. (2020). *Assessing the sovereignty and security of the Austrian internet*. https://doi.org/10.1109/icssa51305.2020.00011

Ganji, D., Christos Kalloniatis, H. Mouratidis, & Saeed Malekshahi Gheytassi. (2020). *Approaches to Develop and Implement ISO/IEC 27001 Standard - Information Security Management Systems: A Systematic Literature Review*. The International Journal on Advances in Software is published by IARIA. https://www.researchgate.net/profile/Lea-Daling/publication/340886820_Media_Comparison_for_Instruction-based_AR_Usage_in_Collaborative_Assembly/links/5ea2b5d092851c87d1b105aa/Media-Comparison-for-Instruction-based-AR-Usage-in-Collaborative-Assembly.pdf#page=49

Gebremeskel, B. K., Jonathan, G. M., & Yalew, S. D. (2023). Information security challenges during digital transformation. *Procedia Computer Science*, *219*, 44–51. https://doi.org/10.1016/j.procs.2023.01.262

Gowher Hassan, M. & Hassan, G. (2023). *TECHNOLOGY AND THE TRANSFORMATION OF ASSESSMENTAL PRACTICES: a FUTURE PERSPECTIVE*. https://www.semanticscholar.org/paper/TECHNOLOGY-AND-THE-TRANSFORMATION-OF-ASSESSMENTAL-A-GowherHassan-Hassan/833587f018ccaf7ccab61c5c9788c52c204a1c5e

Hasan, M., Sifuddin Ansary, Head, B., & Biz, N. (n.d.). Cloud Infrastructure Automation Through IaC (Infrastructure as Code). In *International Journal of Computer*. https://core.ac.uk/download/pdf/555335051.pdf

Ho, H., Ko, R., Mazerolle, L., Gilmour, J., & Miao, C. (2024). Using Situational Crime Prevention (SCP)-C3 cycle and common inventory of cybersecurity controls from ISO/IEC 27002:2022 to prevent cybercrimes. *Journal of Cybersecurity*, *10*(1). https://doi.org/10.1093/cybsec/tyae020

Jhala. Y., N. (2014). Network Scanning & Vulnerability Assessment with Report Generation. ResearchGate. https://www.researchgate.net/publication/263779662_Network_Scanning_Vulnerability_Assessment_with_Report_Generation

Joost Oortwijn, & Gañán, C. (2024). *Patch Pilgrimage: Exploring the Landscape of TCP Reflective Attacks and User Patching Expedition*. https://doi.org/10.1145/3605098.3635982

Jung, B., Li, Y. & Bechor, T. (2022). CAVP: A context-aware vulnerability prioritization model. Computers & Security, 116, 102639. https://doi.org/10.1016/j.cose.2022.102639

ISACA. (n.d.). *Practical patch management and mitigation*. https://www.isaca.org/resources/isaca-journal/issues/2019/volume-3/practical-patch-management-and-mitigation

ISO. (2022, October). *ISO/IEC 27001 standard – information security management systems*. ISO. https://www.iso.org/standard/27001

ISO 27001:2022 Patch Management and System Updates Policy Template. (2023, December 27). ISO Templates and Documents Download. https://iso-docs.com/blogs/iso-27001-isms/iso-27001-2022-patch-management-and-system-updates-policy-template

Kang, K.-D., Park, G., Kim, H., Alian, M., Kim, N. S., & Kim, D. (2021). NMAP: Power Management Based on Network Packet Processing Mode Transition for Latency-Critical Workloads. *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*. https://doi.org/10.1145/3466752.3480098

Kemendi, A., & Pal Michelberger. (2024). Process security methods and measurement in the context of standard management systems. *Engineering Management in Production and Services*, *16*(2), 148–165. https://doi.org/10.2478/emj-2024-0019

Kepuska, K. & Tomasevic, M. (2024). A lightweight framework for cyber risk management in Western Balkan higher assessment institutions. *PeerJ Computer Science*, *10*, e1958. https://doi.org/10.7717/peerj-cs.1958

Kiennert, C., Kaaniche, N., Laurent, M., Rocher, P.-O., & Garcia-Alfaro, J. (2017). Anonymous Certification for an e-Assessment Framework. Lecture Notes in Computer Science, 70–85. https://doi.org/10.1007/978-3-319-70290-2_5

Kotronoulas, G., Miguel, S., Dowling, M., Fernández-Ortega, P., Colomer-Lahiguera, S., Bağçivan, G., Pape, E., Drury, A., Semple, C., Dieperink, K. B., & Papadopoulou, C. (2023). An overview of the fundamentals of data management, analysis, and interpretation in quantitative research. Seminars in Oncology Nursing, 39(2), 151398. Sciencedirect. https://doi.org/10.1016/j.soncn.2023.151398

Khadka, B. (2019). *Data analysis theory and practice : Case: Python and Excel Tools*. Www.theseus.fi. https://www.theseus.fi/handle/10024/335764

Khlifi, Y., & El-Sabagh, H. A. (2017). A Novel Authentication Scheme for E-Assessment s Based on Student Behavior over E-learning Platform. *International Journal of Emerging Technologies in Learning (IJET)*, *12*(04), 62. https://doi.org/10.3991/ijet.v12i04.6478

Krisztián, G., & Cism, C. (2013). pp © The Author(s). In *Conference Proceedings compilation* (pp. 133–141). https://old2.kgk.uniobuda.hu/sites/default/files/11_Horvath_Gergely_Krisztian.pdf

Knowles, W., Baron, A., & McGarr, T. (2016). The simulated security assessment ecosystem: Does penetration testing need standardisation? *Computers & Security*, *62*, 296–316. https://doi.org/10.1016/j.cose.2016.08.002

Laine, M. (2024). Preparing DevOps for ISO 27001 certification. *Theseus.fi*. http://www.theseus.fi/handle/10024/874323

Laurence, P. (2023). *Establishment of a Centralized Policy Framework for IT/IS Governance based on a PDCA method*. https://doi.org/10.1109/hnicem60674.2023.10589150

Lim, W. M. (2024). What Is Quantitative Research? An Overview and Guidelines. *Australasian Marketing Journal (AMJ)*, *0*(0). https://doi.org/10.1177/14413582241264622

MEDUPE. (2009).Configuration management database in an information and communication technology environment.Mini-dissertation submitted in partial fulfilment of the requirements for the degree Masters of Business Administration at the North West University. Medupe_tj.pdf

Mell, P., Scarfone, K., & Romanosky, S. (2005). Common Vulnerability Scoring System. *IEEE Security and Privacy Magazine*, *4*(6), 85–89. https://doi.org/10.1109/msp.2006.145

Mell, P., Scarfone, K. & Romanosky, S. (2007). *The Common Vulnerability Scoring System (CVSS) and its Applicability to Federal Agency Systems*. https://www.semanticscholar.org/paper/The-Common-Vulnerability-Scoring-System-%28CVSS%29-and-Mell-Scarfone/948283e36c643b746e464a812815c0342f37625e

Mccallister, E., Grance, T., & Scarfone, K. (n.d.). *Guide to Protecting the Confidentiality of Personally Identifiable Information (PII) Recommendations of the National Institute of Standards and Technology Special Publication 800-122*. https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-122.pdf?__hstc=40977085.07430159d50a3c91e72c280a7921bf0d.1530662400146.1530662400147.1530662400148.1&__hssc=40977085.1.1530662400149&__hsfp=1773666937

MicrosoftArchiveddocs. (2010, July 14). How To: Use URLScan. Learn.microsoft.com. https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648552(v=pandp.10)

Mishra, S., & Saikat Gochhait. (2023). Emerging Cybersecurity Attacks in the Era of Digital Transformation. https://doi.org/10.1109/iciccs56967.2023.10142357

Mohajan, H. K. (2021). Quantitative research: A successful investigation in natural and social sciences. ResearchGate; www.researchgate.net. https://www.researchgate.net/publication/348237026_Quantitative_Research_A_Successful_Investigation_in_Natural_and_Social_Sciences

Mozilla. (n.d.). *X-Frame-Options*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options

Mukul, E., & Büyüközkan, G. (2023). Digital transformation in assessment: A systematic review of assessment 4.0. https://www.semanticscholar.org/paper/Digital-transformation-in-assessment%3A-A-systematic-Mukul-B%C3%BCy%C3%BCk%C3%B6zkan/76ae7503db2b7c9aee2a23c5479f9de905272fa7

M. Rabzelj, C. Bohak, L. Š. Južnič, A. Kos and U. Sedlar, "Cyberattack Graph Modeling for Visual Analytics," in *IEEE Access*, vol. 11, pp. 86910-86944, 2023, doi: 10.1109/ACCESS.2023.3304640

Nedim (2020). HTTP OPTIONS and Default page vulnerabilities. [online] TECHCOMMUNITY.MICROSOFT.COM. Available at: https://techcommunity.microsoft.com/blog/iis-support-blog/http-options-and-default-page-vulnerabilities/1504845

Nesara Dissanayake, Mansooreh Zahedi, Asangi Jayatilaka, and Muhammad Ali Babar. 2022. Why, How and Where of Delays in Software Security Patch Management: An Empirical Investigation in the Healthcare Sector. Proc. ACM Hum.-Comput. Interact. 6, CSCW2, Article 362 (November 2022), 29 pages. https://doi.org/10.1145/ 3555087

Dissanayake, N., Jayatilaka, A., Zahedi, M., & Babar, M. A. (2021, August 19). Software Security Patch Management -- A Systematic Literature Review of Challenges, Approaches, Tools and Practices. ArXiv.org. https://doi.org/10.48550/arXiv.2012.00544

N., & Roumeliotis, N. (2022). AppLocker bypass toolkit. Unipi.gr. https://dione.lib.unipi.gr/xmlui/handle/unipi/14487

Nginx (2024). Allowed HTTP Methods | NGINX Documentation. [online] Available at: https://docs.nginx.com/nginx-management-suite/acm/how-to/policies/allowed-http-methods/

Ngalim, B. (2023). Integrating NIST and ISO Cybersecurity Audit and Risk Assessment Frameworks into Cameroonian Law. Journal of Cybersecurity Education, Research and Practice, 2024(1). https://doi.org/10.32727/8.2023.29

Nmap. (2024). *Nmap*. Nmap.org. https://nmap.org/

NVD-NIST - Home. (n.d.). NVD - Vulnerability Metrics

Osborne, C., & Boone, D. (2023, November 7). *Effective Patch Management and Government Systems*. Www.dair.nps.edu. https://www.dair.nps.edu/handle/123456789/5017

O., Mariya, & O., Mariya. (2025). *Offensive and Defensive Cyber Security Strategies*. CRC Press. Offensive and Defensive Cyber Security Strategies: Fundamentals, Theory and ... - Mariya Ouaissa, Mariyam Ouaissa - Google Books

OWASP (n.d.). *HTTP Strict Transport Security · OWASP Cheat Sheet Series*. [online] cheatsheetseries.owasp.org. Available at: https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html.

OWASP. (n.d.). Owasp.org. Security Headers. https://owasp.org/www-community/Security_Headers

OWASP. (2012). *Cross-Site Request Forgery Prevention · OWASP Cheat Sheet Series*. Owasp.org. https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html

OWASP. (2021). *A05 Security Misconfiguration - OWASP Top 10:2021*. Owasp.org. https://owasp.org/Top10/A05_2021-Security_Misconfiguration/

Priyawati, D., Rokhmah, S., & Utomo, I. C. (2022). Website Vulnerability Testing and Analysis of Website Application Using OWASP. *International Journal of Computer and Information System (IJCIS)*, *3*(3), 142–147. https://doi.org/10.29040/ijcis.v3i3.90

Proença, D., & Borbinha, J. (2018). Information Security Management Systems - a maturity model based on ISO/IEC 27001. In Lecture notes in business information processing (pp. 102–114). https://doi.org/10.1007/978-3-319-93931-5_8

Pulliainen, T. (2016). Linux Patch Management: Comparison of Practical Implementations [Master's thesis]. In *Master's Degree Programme in Information Technology*. https://www.theseus.fi/bitstream/handle/10024/115562/Pulliainen_Teemu.pdf?sequence=1

Rashidi, S., Shurpali, P., Sridharan, S., Rashidi et al., N., Mudigere, D., Nair, K., Smelyanski, M., & Krishna, T. (2020, August 1). Scalable Distributed Training of Recommendation Models: An ASTRA-SIM + NS3 case-study with TCP/IP transport. IEEE Xplore. https://doi.org/10.1109/HOTI51249.2020.00020

Rathore, S. (2023, June 30). *The Impact of AI on Recruitment and Selection Processes: Analysing the role of AI in automating and...* ResearchGate; International Consortium of Academic Professionals for Scientific Research. https://www.researchgate.net/publication/372011217_The_Impact_of_AI_on_Recruitment_and_Selection_Processes_Analysing_the_role_of_AI_in_automatin

Ridzuan, F., & Zainon, W. M. N. W. (2019). A Review on Data Cleansing Methods for Big Data. *Procedia Computer Science*, *161*(1), 731–738. https://doi.org/10.1016/j.procs.2019.11.177

Roumani, Y. (2021). Patching zero-day vulnerabilities: an empirical analysis. *Journal of Cybersecurity*, *7*(1). https://doi.org/10.1093/cybsec/tyab023

Rott, K. J. & Schmidt-Hertha, B. (2024). Transforming adult assessment in the digital age: exploring environmental, contentand technological changes. *International Journal Of Lifelong Assessment*, *43*(4), 319–323. https://doi.org/10.1080/02601370.2024.2367395

Salmons, J. (2023). *Quantitative Research with Non-experimental Designs*. Sage Research Methods Community. https://researchmethodscommunity.sagepub.com/blog/quantitative-research-with-non-experimental-designs

Scarfone, K. A., & Mell, P. M. (2010). Intrusion Detection and Prevention Systems. In *Springer eBooks* (pp. 177–192). https://doi.org/10.1007/978-3-642-04117-4_9

Schmidt, J. T. & Tang, M. (2020). Digitalization in Assessment: Challenges, Trends and Transformative Potential. In *Springer eBooks* (S. 287–312). https://doi.org/10.1007/978-3-658-28670-5_16

Security, C., Manubolu, G., Mohammad, T., & Sainio, P. (2024). *A COMPREHENSIVE SECURITY TESTING FRAMEWORK FOR PLC-BASED INDUSTRIAL AUTOMATION AND CONTROL SYSTEMS*.

https://www.utupub.fi/bitstream/handle/10024/178734/greeshma_manubolu_masters_thesis.pdf?sequence=1

Setiyanto, S., & Setiawan, I. (2022). Data Science With Excel. *International Journal of Computer and Information System (IJCIS)*, *3*(3), 104–110. https://doi.org/10.29040/ijcis.v3i3.79

Shahid, J., Hameed, M. K., Javed, I. T., Qureshi, K. N., Ali, M., & Crespi, N. (2022). A Comparative study of Web Application Security Parameters: Current trends and future directions. *Applied Sciences*, *12*(8), 4077. https://doi.org/10.3390/app12084077

Shaikh, K. (2020, December 2). *"cross domain javascript source file inclusion" reported by OWASP ZAP for trusted resource*. Stack Overflow. https://stackoverflow.com/questions/65109625/cross-domain-javascript-source-file-inclusion-reported-by-owasp-zap-for-truste

Sharma, M., Desai, D., Aditya Ranganathan Arun, Priya. L, & Rajagopalan, N. (2024). *OpenVAS vs the Rest: Unveiling the Competitive Edge in Vulnerability Scanners*. https://doi.org/10.1109/inocon60754.2024.10511864

Sjögård, J. (2022). Designing an automated ICT monitoring system. *Theseus.fi*. http://www.theseus.fi/handle/10024/743675

Souppaya, M., Scarfone, K., Computer Security Division, Scarfone Cybersecurity, U.S. Department of Commerce, National Institute of Standards and Technology, Gina M. Raimondo, James K. Olthoff, Mark Simos, Jamie Brown, Vincent Gilcreest, Glen Pendley, BlackBerry, Cybersecurity & Infrastructure Security Agency (CISA), National Security Agency (NSA), The SECOND Advisories, Carla Brinker, & Ron Colvin. (2022). Guide to Enterprise Patch Management Planning: Preventive Maintenance for Technology. In *NIST Special Publication* [Report]. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r4.pdf

Tawo, O. E., & Ajayi, R. (2025). IT Service Management and Configuration Management Database for Enhancing Efficiency and Compliance. *International Journal of Research Publication and Reviews*, *6*(6), 4365–4381. https://doi.org/10.55248/gengpi.6.0125.0620

Temizkan, O., Kumar, R. L., Park, S., & Subramaniam, C. (2012). Patch Release Behaviors of Software Vendors in Response to Vulnerabilities: An Empirical Analysis. Journal of

Management Information Systems, 28(4), 305–338. https://doi.org/10.2753/mis0742-122228041

Tom, S., Christiansen, D., Berrett, D., Battelle Energy Alliance & DHS National Cyber Security Division Control Systems Security Program. (2008). *Recommended Practice for Patch Management of Control Systems* [Report]. https://inldigitallibrary.inl.gov/sites/sti/sti/4121152.pdf

Villamin, P., Lopez, V., Thapa, D., & Cleary, M. (2024). A Worked Example of Qualitative Descriptive Design: A Step-by-Step Guide for Novice and Early Career Researchers. *Journal of Advanced Nursing*. https://doi.org/10.1111/jan.16481

Wei, Z. (2023). Navigating digital assessment landscapes: unveiling the interplay between assessment behaviors, digital literacyand assessmental outcomes. Journal of the Knowledge Economy. https://doi.org/10.1007/s13132-023-01522-3

West,M. & Goodwin, M. (2016). *draft-ietf-httpbis-cookie-same-site-00*. [online] Available at: https://datatracker.ietf.org/doc/html/draft-ietf-httpbis-cookie-same-site

World Wide Web Consortium (W3C), . (n.d.). W3c.github.io. https://w3c.github.io/webappsec-csp/

WSTG - v4.1 | OWASP Foundation. (n.d.). Owasp.org. https://owasp.org/www-project-web-security-testing-guide/v41/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes.html

Yang, B., Naga Ayachitula, Zeng, S., & Puri, R. (2011). *SLA-driven applicability analysis for patch management*. *3*, 438–445. https://doi.org/10.1109/inm.2011.5990544

Zhang, Y., Guo, Z., & Sun, T. (2023). A Non-Intrusive Automated Testing System for Internet of Vehicles App Based on Deep Learning. *Electronics*, *12*(13), 2873. https://doi.org/10.3390/electronics12132873

Hochschule Neu-Ulm
University of Applied Sciences

APPENDIX A

Appendix A: Result of Python Analysis

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

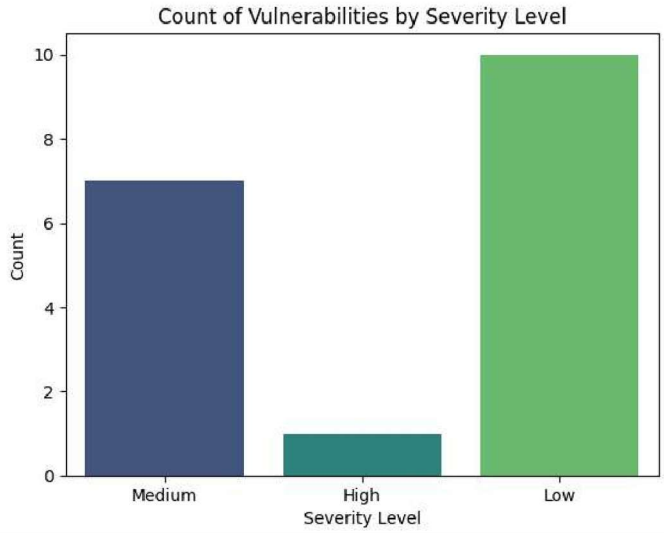
```
# Create a DataFrame for the given data with both "YES" and "NO" for "Patch Possible?"
data = {
    "Vul. ID": [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18],
    "Type of Vulnerability": [
        "Absence of Anti-CSRF Tokens", "Missing Anti-clickjacking Header", "Content Security Policy (CSP) Header Not Set",
        "X-Content-Type-Options Header Missing", "Cross-Domain JavaScript Source File Inclusion",
        "Server Leaks Version Information via 'Server' Header", "Strict-Transport-Security Header Not Set",
        "TCP Timestamps Information Disclosure", "Weak MAC Algorithm(s) Supported (SSH)",
        "Open TCP Port: 22", "Open TCP Port: 443", "Open TCP Port: 80",
        "Insecure Cookie Setting: Missing Secure Flag", "Missing security header: Referrer-Policy",
        "Server Software and Technology Found", "Security.txt file is missing",
        "HTTP OPTIONS Enabled", "Cookie without SameSite Attribute"
    ],
    "Severity Score": [5, 5, 7.5, 2, 2.5, 3, 5.5, 2.6, 2.6, 5, 3.5, 3, 2.5, 5, 2.5, 5, 5, 2],
    "Severity Level": ["Medium", "Medium", "High", "Low", "Low", "Low", "Medium", "Low", "Low", "Medium", "Low", "Low", "Low", "Medium
    "Affected Software/ System": [
        "Web Application", "Web Application", "Web Application", "Web Application", "Web Application",
        "Web Server", "Web Application", "Network Infrastructure", "SSH Server",
        "Web Server", "Web Server", "Web Server", "Web Application", "Web Application",
        "Web Server", "Web Server", "Web Server", "Web Server"
    ],
    "Impact Scope": [
        "Potential CSRF attacks", "UI redirection vulnerability", "Cross-site scripting risks",
        "Risk of MIME-type sniffing", "Potential data exposure", "Information disclosure",
        "Risk of downgrade attacks", "Uptime information leakage", "Potential data integrity issue",
        "Unsecured access point", "Potential unauthorized access", "Potential unauthorized access",
        "Risk of session hijacking", "Information leakage via referrers", "Information disclosure",
        "Missing security contact details", "Potential method discovery attack",
        "A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' r
    ],
    "Patch Possible?": ["NO", "NO", "NO", "YES", "YES", "YES", "NO", "YES", "YES", "NO", "YES", "YES", "YES", "NO", "YES", "NO", "NO",
    "Discovery Date": ["20.01.2025"] * 18,
    "Detection Source": [
        "OWASP ZAP", "OWASP ZAP", "OWASP ZAP", "Nmap", "OWASP ZAP", "OWASP ZAP", "Nmap",
        "OpenVAS", "OWASP ZAP", "Nmap", "Nmap", "Nmap", "OWASP ZAP", "Nmap", "OpenVAS",
        "OWASP ZAP", "Nmap", "OWASP ZAP"
    ]
}
```

```
df = pd.DataFrame(data)

# Display the first few rows of the dataset
print("Preview of the data:")
print(df.head())
```

```
Preview of the data:
   Vul. ID                        Type of Vulnerability  Severity Score  \
0        1                  Absence of Anti-CSRF Tokens             5.0
1        2              Missing Anti-clickjacking Header            5.0
2        3   Content Security Policy (CSP) Header Not Set            7.5
3        4          X-Content-Type-Options Header Missing            2.0
4        5  Cross-Domain JavaScript Source File Inclusion            2.5

  Severity Level Affected Software/ System                 Impact Scope  \
0         Medium           Web Application        Potential CSRF attacks
1         Medium           Web Application  UI redirection vulnerability
2           High           Web Application    Cross-site scripting risks
3            Low           Web Application    Risk of MIME-type sniffing
4            Low           Web Application       Potential data exposure

  Patch Possible? Discovery Date Detection Source
0              NO     20.01.2025        OWASP ZAP
1              NO     20.01.2025        OWASP ZAP
2              NO     20.01.2025        OWASP ZAP
3             YES     20.01.2025             Nmap
4             YES     20.01.2025        OWASP ZAP
```

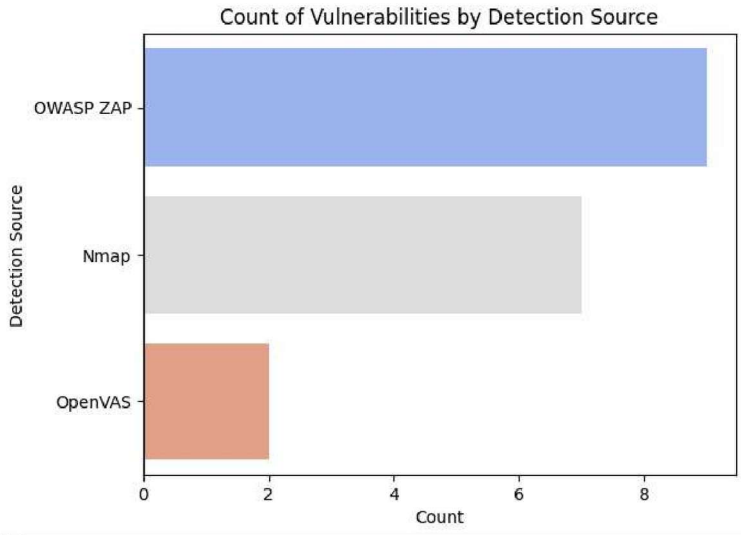
```
# Example visualization 1: Bar plot of vulnerabilities by Severity Level
sns.countplot(data=df, x='Severity Level', palette='viridis')
plt.title("Count of Vulnerabilities by Severity Level")
plt.xlabel("Severity Level")
plt.ylabel("Count")
plt.show()
```

```
<ipython-input-4-f74c0597315b>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `l

  sns.countplot(data=df, x='Severity Level', palette='viridis')
```

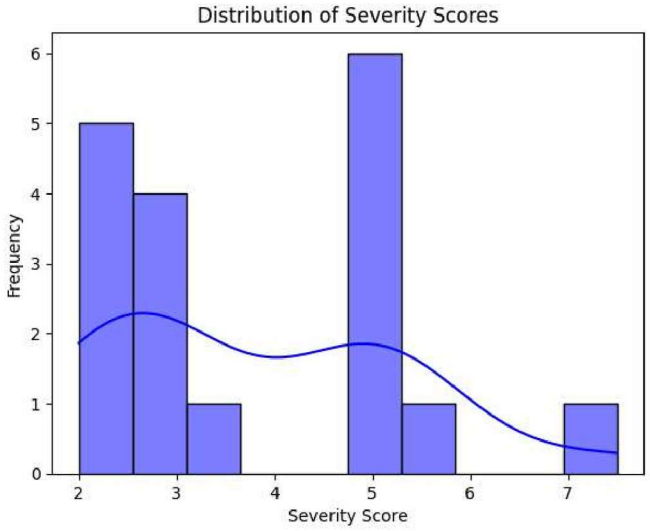



```
# Example visualization 2: Bar plot of vulnerabilities by Detection Source
sns.countplot(data=df, y='Detection Source', palette='coolwarm')
plt.title("Count of Vulnerabilities by Detection Source")
plt.xlabel("Count")
plt.ylabel("Detection Source")
plt.show()
```

```
<ipython-input-5-d3457c090d6e>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `y` variable to `hue` and set `l

  sns.countplot(data=df, y='Detection Source', palette='coolwarm')
```

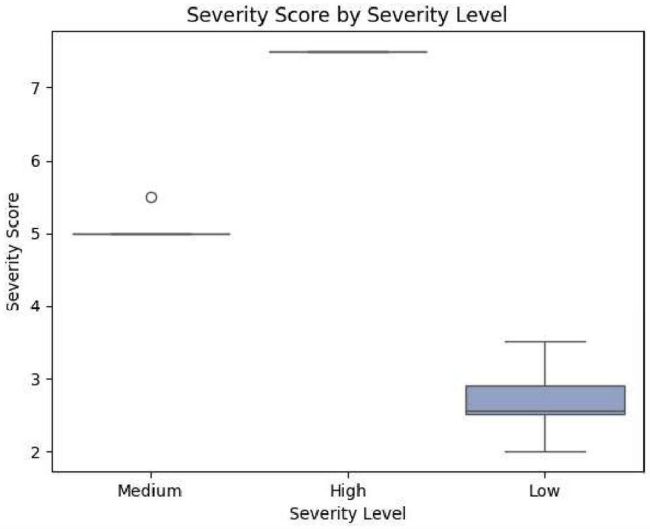



```
# Example visualization 3: Severity Score distribution
sns.histplot(df['Severity Score'], kde=True, bins=10, color='blue')
plt.title("Distribution of Severity Scores")
plt.xlabel("Severity Score")
plt.ylabel("Frequency")
plt.show()
```

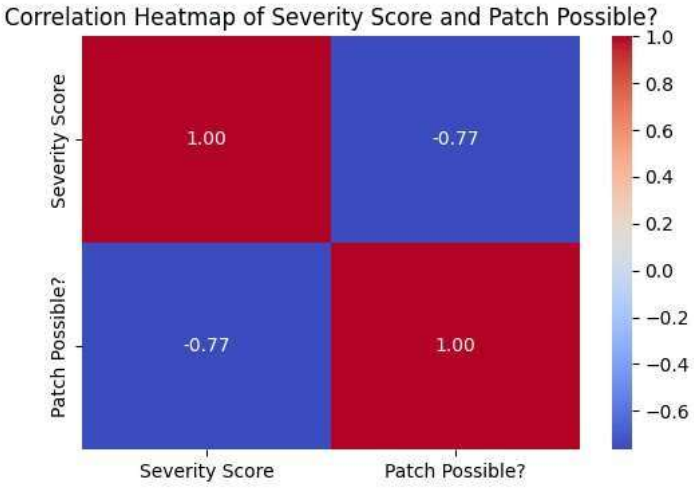



```
# Example visualization 4: Box plot of Severity Score by Severity Level
sns.boxplot(data=df, x='Severity Level', y='Severity Score', palette='Set2')
plt.title("Severity Score by Severity Level")
plt.xlabel("Severity Level")
plt.ylabel("Severity Score")
plt.show()
```

```
<ipython-input-7-b47a50af3399>:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `l

  sns.boxplot(data=df, x='Severity Level', y='Severity Score', palette='Set2')
```




```
# Create DataFrame
df = pd.DataFrame(data)

# Encode 'Patch Possible?' as a binary variable (YES = 1, NO = 0)
df['Patch Possible?'] = df['Patch Possible?'].map({'YES': 1, 'NO': 0})

# Create a correlation matrix for 'Severity Score' and 'Patch Possible Numeric'
correlation_matrix = df[['Severity Score', 'Patch Possible?']].corr()

# Plot the heatmap
plt.figure(figsize=(6, 4))
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm", fmt='.2f')
plt.title("Correlation Heatmap of Severity Score and Patch Possible?")
plt.show()
```


Correlation Heatmap of Severity Score and Patch Possible?
Severity Score
Patch Possible?
1.00
-0.77
-0.77
1.00
Severity Score
Patch Possible?
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6