**Optimizing Web Development and Deployment Efficiency: The Impact of React, MongoDB, And Jenkin in Modern Software Engineering**

**ABSTRACT**

|  |
| --- |
| A study examines how the current web technologies React and MongoDB affect development speed and system operational outcomes. The analysis investigates how React elements improve system maintainability through its component-based design in combination with MongoDB flexibility for handling dynamic data systems. The integration of Jenkins as a Continuous Integration and Continuous Deployment (CI/CD) pipeline receives analysis to evaluate deployment speed and automation as well as software reliability. Jenkins stands apart from alternative CI/CD solutions because the study demonstrates its exclusive deployment benefits over conventional model approaches. The implemented technologies prove essential for contemporary software engineering because they minimize developmental periods and boost teamwork and deliver continuous updates. |

*Keywords: Vulnerability modeling, Corporate cybersecurity, AI-based tools, Web architecture, Web applications.*

**1. INTRODUCTION**

In enterprises with complex IT and industrial control systems (ICS) landscapes and a significant reliance on automated information processing involving numerous operational technology (OT) components, information security governance within enterprise IT management has become more and more important. The use of complex ICSs, such as supervisory control and data acquisition (SCADA) systems, by critical infrastructure firms is especially noteworthy ((Makrakis, Kolias, Kambourakis, Rieger, & Benjamin, 2021; Masi, Sellitto, Aranha, & Pavleska, 2023). {Makrakis, 2021 #587}According to Yaacoub et al. (2020), these systems are used in a variety of industries, including manufacturing, energy, transportation, environmental management, healthcare, and smart urban development. The integration of such sophisticated technologies has been a driving factor behind the digital transformation of physical entities and their interconnectedness in cyberspace, meeting the severe expectations for ultra-reliable services in critical systems (Yaacoub et al., 2020). As demonstrated by examples such as the WannaCry ransomware and the Stuxnet virus, the cybersecurity environment has been characterized by serious threats to CIs, specifically targeting ICSsBecause multiple layers are interdependent, vulnerabilities in one segment may have an impact on others, increasing the potential for losses (Longueira-Romero, Iglesias, Flores, & Garitano, 2022). By concentrating on key company objectives, attackers typically aim to disrupt business processes, affecting apps and the platform systems that support them. These attacks, which exploit system flaws, have caused significant disruptions and negative economic effects, including the suspension of production at European car manufacturing facilities, according to authors (Santangelo, Colacino, & Marchetti, 2021). As a result, preserving cybersecurity in companies is an ongoing, dynamic process that requires ongoing adaptation to changing external conditions (Nahar & Gill, 2022). It is crucial to see cybersecurity as an essential part of larger enterprise-level initiatives rather than as a stand-alone activity. In order to do this, businesses must have a thorough awareness of their IT infrastructure and how it affects their operations. Because there are so many assets in the IT and OT arena, critical infrastructure operators need a thorough understanding of their relationships.

Cybersecurity has frequently been viewed as something that should be considered or added to current systems and applications later on. More often than not, securing one’s web properties was considered an afterthought, tacked on at the last moments of software development rather than woven into the code. Thus, security has typically been considered as an IT problem as opposed to a business issue, thus requiring reactionary responses instead of proactive defense. An earlier approach was to use perimeter security (e.g., firewalls, antivirus software) as the only tool for threat modeling. However, security by design and DevSecOps methodologies have come up over time that emphasizes the necessity to integrate cybersecurity from the design and development perspectives from the onset through the system lifecycle. It has recently been evident that, in contrast to the conventional security-by-obscurity principles, companies must adopt a proactive strategy in order to keep up with the growing frequency and sophistication of assaults on IT infrastructures (Abraham, 2019). It is becoming more widely acknowledged that a comprehensive, integrated strategy is necessary for cyber-security, rather than viewing it as an aside or an additional layer to be put to current systems. According to authors Blanco et al. (2023), this holistic viewpoint, which is frequently summed up in the idea of cyber-security by structure, highlights the need of integrating cyber-security concerns across the whole product lifetime. This method naturally recognizes that when security is integrated into the system's design rather being added on as an afterthought, the system's ability to withstand attacks is much increased. By emphasizing that the system is worth more than the sum of its components, this paradigm shift toward seeing security as an essential, inseparable component of system design highlights the value and superiority of systemic methods over fragmented techniques. There are still issues in some sectors despite several projects aimed at assessing and improving cyber-security by design. For example, there are special challenges in protecting SCADA systems (Bhamare et al., 2020) and guaranteeing cybersecurity in specific fields such as smart grids (Darteh, 2022). First, according to author Upadhyay and Srinivas Sampalli (2020), the methods are often disjointed and do not fully cover all life-cycle stages that are pertinent to security by design. For starters, the emphasis is either on assessing individual vulnerabilities, attacks, and countermeasures, or on top-down security needs analysis (as in the case of NISTIR 7628), but not both. Second, current methods concentrate on individual component weaknesses rather than the system as a whole. Third, the methods don't take into consideration current security attack data, and fourth, they frequently don't support software tools.

Because it helps foster a common knowledge of the current vulnerabilities and potential attack vectors, among other things, the use of modeling to enhance security studies is seen as promising it may be able to connect IT problems to use cases at the corporate level (Jiang, 2023) and conceptual modeling makes (semi-)automated reasoning easier, which makes simulation possible, among other things. Please take note that using a modeling language also makes one more particular and concrete, which enhances the value of the analysis. Furthermore, combining several modeling languages might provide more thorough evaluations of a system being studied. To facilitate security studies, a variety of conceptual modeling techniques have been put forth, including risk assessment techniques that function at various degrees of abstraction (Ekstedt, 2023).

They facilitate the examination of individual software systems, the organization as a whole, and systems at a system level. At the organizational level, for example, Through a knowledge of organizational risk, FAIR seeks to assist executives in making better decisions. As a result, FAIR offers resources for comprehending, quantifying, and evaluating information risks. Various topics, including risk theory, risk computation, scenario modeling, and risk communication inside the company, are addressed by the method. Individual attackers, vulnerabilities, and event impact are used to quantify the risk based on this data (Sukumar, 2023). On the other hand, PASTA was created with leaders in IT, privacy, safety, and risk in mind, assisting them in risk mitigation and threat analysis. It is a riskcentric threat modeling methodology as a result. PASTA guides the risk assessor via an adaptable, iterative approach that emphasizes threat analysis, business impact, and countermeasures (Khatti, 2023).

In a similar vein, TRIKE begins with a data flow diagram that defines the system and all of its components, including actors, resources, planned actions, and rules. Threats are then classified into two groups: denial of service and elevation of privilege. Lastly, the probability of a threat and its effect on business objects are used to calculate its risk. STRIDE is a widely used and well-liked technique for evaluating various system hazards. Data flow diagrams are used by STRIDE to depict the system and associate various risks that may affect it. The threats are classified into six categories, giving STRIDE its name: identity spoofing, data tampering, repudiation, dissemination of data, denial of service, and privilege elevation. To properly assess risks, STRIDE has been expanded into DREAD, which takes into account discoverability, damage potential, consistency, exploitation, and impacted users. As a result, each category is given a unique number, making it possible to calculate an average value that represents the risk of the whole system (Khalil, 2024; Pasdar, 2024)

.

The software engineering community uses model-based security analysis to support either an automated security assessment or a manual security assessment at the more detailed level of software systems design. Examples of these automated security assessments include SecDSVL, secureTROPOS, and CORAS. The automated methods such as SecureUML, SECTET, UMLsec, and STS-ml enable the specification of the components and interactions of a software system. This specification is enhanced with security features, which make it possible to do automated analysis using formal reasoning and make claims on the security of the software system (Azam, 2022).

In addition to the general-purpose security analysis methods listed above, there is comparable work for SCADA systems and CPS that is occasionally presented in the context of the ISO 31000:2009 risk management process. In order to handle any risks as soon as feasible rather than as an afterthought, our contribution centers on establishing security by design, which is a basic idea within contemporary risk management. Cherdantseva et al. (2022) examined 24 distinct techniques for evaluating the cyber-security of SCADA systems and proposed a classification system for them. They listed five research problems, which also serve as a guide for our work: (1) The methods either focus on a comprehensive assessment while ignoring SCADA-specific details, or they focus on specific parts while falling short of the overall picture; (2) the majority of methods focus on individual vulnerabilities rather than the system itself; (3) there is a lack of data on security attacks on SCADA systems; (4) the methods that are presented do not properly validate their findings; and (5) the methods lack tool-support (Alanazi, 2023; Cherdantseva, 2022).

Tantawy et al. (2020), for example, create a model-based method for assessing the risk of CPS and test it in a test bed using actual industrial controllers and communication methods. Physical vulnerability identification, environment modeling, and evaluating the test bed's weaknesses. The advantages of automating the analysis itself are acknowledged by the writers in their work (Tantawy, 2020). Lastly, Jiang et al. (2023) use multi-level modeling specifically and conceptual modeling generally for security analysis that focuses on the electrical industry. Jiang et al. (2023) focus on particular weaknesses, assaults, and the spread of failures. Reactive in nature, this strategy concentrates on weaknesses and assaults on current infrastructure (Jiang, 2023).

Advanced adoption of digital technologies in the corporate systems has pushed the web architecture to the essence of business operations. Although corporate systems help to offer numerous advantages including scalability, flexibility, and accessibility, it creates chances that hackers can use (Zhang, 2021). Metamodeling of threats and risks in Web architecture is a promising activity and a part of the process of assessment and analysis of risk factors aimed at enhancement of the overall cybersecurity of corporate systems. To wit, this paper aims to discuss the formulation of this issue, the scientific and practical implications of this problem, the state of prior literature on this topic, open questions to be solved, and the importance of developing this line of research.

Scientific and practical relevance of the studied phenomena and processes is always important to determine. Risks to corporate systems involve data breaches, ransomware attacks, and the Advanced Persistent Threats (APTs) (Jabar, 2022). As estimated by IBM Security (2022), the cost of a data breach was $4.35m in 2022 and web application vulnerabilities are amongst the main reasons contributing to data breaches. This serves to show the need for a structured method of analyzing and flagging shortcomings within the web structure. By developing models to consider every possible susceptible factor, it becomes easier to identify possible attacks and their likely effects because defense strategies can be put in place (Fathullah, 2022). Pragmatically, the capacity for threat preemption and neutralization is paramount toward preserving data authenticity, customer confidence, and regulatory non-negotiables, including the GDPR and CMMC (Çakmakçı, 2020).

**1.1. Overview of Threat Modeling Frameworks and Web Security Challenges**

**1.1.1 Overview of Threat Modeling Frameworks**

Web applications threat modeling frameworks can be defined as systematic approaches of identifying, assessing and mitigating security threats in web applications. There are several established models for perspectives of cyber security risks. The six types of threats in the security threats identified by the STRIDE model developed by Microsoft is Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service and Elevation of Privilege. In particular, this framework is a very good one to employ in determining and addressing potential vulnerabilities before the software development is even started. The DREAD (Damage potential, Reproducibility, Exploitability, affected users, Discoverability) framework is another widely used model used to assess the risks which help the organizations to priorities the security threats on the basis of their severity (Ramos Flores, 2023).

By simulating real attacks, the PASTA (Process for Attack Simulation and Threat Analysis) methodology is risk-centric. PASTA integrates business impact assessment and security analysis which reduces risks and will help organizations to take a strategic approach to mitigate risks. On the other hand, the National Institute of Standards and Technology (NIST) Cybersecurity framework which was launched by the National Institute of Standards and Technology provides guidelines to organizations in identifying, preventing, detecting, responding to and returning to efficiency from the cyber-attacks. It is adopted across several industries to establish the resilience to cybersecurity. OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation) is another major framework whose mission is to evaluate the level of threat toward critical business enterprise assets. It is especially good in managing risks in large organizations (Santangelo, Colacino, & Marchetti, 2021, Pinto, Siano, & Parente, 2023).

**1.1.2 Discussion of Existing Studies on Cybersecurity Vulnerabilities**

Existing research reveals many of these vulnerabilities that remain very risky for web applications and digital infrastructure. Specifically, injection attacks, such as SQL, command, and LDAP injections, are one of the most persistent threats and allow the attacker to inject commands in the databases and illegally access sensitive information. In addition, studies also conclude that most web applications do not put in good practice of input validation or they do it badly, which allows them to be victims of Cross Site Scripting (XSS) and Cross Site Request Forgery (CSRF) attacks. According to these vulnerabilities, attackers can siphon off information pertaining to user sessions, steal the personal info, and or manipulate the user’s actions on the website (Sukumar, Mahdiraji, & Jafari‐Sadeghi, 2023).

The biggest pressing concern is the rise of null day exploits, where the attackers take advantage of the unknown software security vulnerability before the people could release the security patch. However, these exploits are becoming rampant and thus have necessitated organizations to adopt proactive security measures. In addition, data breaches have been identified as a consequence of insider threats, both human errors and malicious insider activities. Such risks can’t be mitigated, and so organizations need to put in place robust access controls and continuous monitoring mechanisms (Fathullah, Subbarao, & Muthaiyah, 2022).

In this context, recent studies have been conducting the use of machine learning to improve the performance of cybersecurity threats detection and response with artificial intelligence. AI based security systems look for anomaly, attack pattern signatures and automatically depending on vast amount of data reduce the manual process of threat assessment. The promise of machine learning brings about challenges, such as the risk of false positives that will result in a massive number of false alerts and unnecessary workload to the security teams.

Recent work has included previously an analytical level of investigating various theoretical and applied aspects of modeling vulnerabilities in web architectures. Threat modeling frameworks like the STRIDE initially developed by Microsoft are already considered basic within context (Çakmakçı, 2020). Such concepts are useful in alerting system architects of possible threats during the design phase and arriving at techniques of countering the effects of the threats (Khatti et al., 2024).

Machine learning and AI have also been identified as major players regarding the modeling of vulnerability. Chen et al. (2021) use machine learning techniques in identifying vulnerabilities in Web applications as traffic flows and application logs in real-time. Likewise, there are AI based tool such as OWASP ZAP (Open Web Application Security Project Zed Attack Proxy) that can help to analyze and discover security vulnerabilities that previously had required manual assessment in vulnerability assessment (Ramos Flores, 2023). However, researchers have made much progress to develop deterministic ones, there remained a second type of vulnerability more complicated: zero-day ones, the ones that were necessarily exploited before they became public. Authors have also suggested that heuristic and behavior-based modeling should be incorporated to fill this gap (Zhang, 2021).

**1.1.3 Unsolved Challenges.**

However, some issues remain unresolved, and this work reveals that there are several of them. First of all, a lot of existing models are built based on historical information, thus they can respond to the new attack vectors efficiently. Second, how to incorporate vulnerability modeling into software development iterations typically associated with agile while not sacrificing speed continues to be an issue. Third, because of the absence of uniformity in modeling techniques, there is variation in the assessment and management of risks among businesses.

Although the threat modeling and security strategy have indeed improved, some challenges are yet to be solved for the web security. The vast number of cyber threats is a big concern as the attackers innovate on new ways of doing things while traditional security defenses rely on the old ways. To stay ahead of these always evolving means of attacking, so one needs to do adaptive threat modeling. Also, coming from various levels of security maturity, without resource availability and infrastructure differences most companies struggle with standardized implementation of threat modeling, and there are many frameworks out there.

The second risk is between automation and cybersecurity accuracy. AI increases the detection rates but also generates false positives and that makes security teams going into alert fatigue. There is much to fight on another front for developing more accurate threat detection mechanisms. Further, as agile development methodologies are increasingly being adopted by the industry, there is an increasing challenge to incorporate wide ranging security measures into speedy software development cycles without disrupting the workflow. Security frameworks were designed for traditional development processes and are complex to fit in seamlessly into modern agile environments (Sukumar, Mahdiraji, & Jafari‐Sadeghi, 2023).

In addition, more businesses are migrating their infrastructure to cloud based features and cloud security risks have become a major concern. The security of web applications that live in the cloud causes security problems around data privacy, misconfigurations and vulnerabilities of third party dependencies. Cloud environment security is a continuing complex issue that needs to be constantly monitored and has to have strong security policies in place (Upadhyay & Srinivas Sampalli, 2020).

This is because of the increasing developments in web architecture and the equally increasing technicality of the aggressive attacks. Thus, further developments need to find solutions to unresolved problems like the identification of new vulnerabilities in real time, the integration of protection measures into the development processes of business systems. Increments in the sophistication and elasticity of models help organisations to secure data, defend operations and stay compliant with requirements in the epoch of digitalisation (Snyder, 2019).

**1.2 Objective**

The purpose of this paper is to undertake a systematic literature review (SLR) to review and categorise the literature on modeling vulnerabilities in web architecture. The purpose is to define the existing trends, gaps and issues, which will help to improve the corporations’ cybersecurity. Thus, the purpose of the article is to systematically review the available literature in order to present the current state of knowledge on the subject, identify the open questions that characterize the existing approaches, and discuss the potential directions for future research.

**1.3 Purpose**

To realise this aim, this article starts by outlining the method of review and the inclusion/exclusion criteria. This implies searching and sifting through appropriate database, journal and conference that deals with web architecture and cybersecurity to find out the studies to include and those to exclude. The next step includes a review of the extant literature to be done through a formal search process with well understood terms and scopes. These included the characterization of the studies by their methodologies, tools, frameworks, and themes with regard to vulnerability modeling. Next, the article outlines the trends and research gaps by analysing the common themes, methods and technologies appearing in the analysed articles. Special emphasis is placed on topics that have not been well-studied as yet, including real-time vulnerability identification, features corresponding to zero-day vulnerability, and the integration of modeling aspects into agile development environments. The practical applications of existing research are also compared to determine feasibility and recognisability of current vulnerability modeling frameworks in the corporate context and the extent to which the theoretical models have been incorporated into the practical ones. Last but not the least, in the article, the author presents the idea of the directions for the future research to overcome the limitation and gaps that have been narrated in the course of the review of studies. Proposed directions for the creation of richer, more context-sensitive and quicker approaches to better compose the space of vulnerability modelling for the improvement of corporate protection against cyber threats are also provided. Thus the article is designed to have sufficiently completed these tasks thus yielding a sufficient positive contribution to knowledge as well as the actualization of hegemonic and practical studies in cyberspace security.

**1.4 Research questions**

RQ1. What theoretical frameworks and practical tools are currently being used in modeling of the vulnerabilities in web architectures?

RQ2. How have the past five years seen what new methods were developed in vulnerability modeling, especially with respect to the use of AI and machine learning?

RQ3. How much statistical reduction for cyber related breaches does vulnerability modeling in organizations achieve?

RQ4. Today enterprises and cybersecurity organizations need to be prepared for and able to respond to emerging threats that include such things as Advanced Persistent Threats (APTs) and zero-day attack.

**2.**  **MATERIAL AND METHODS**

**2.1 Choosing SLR**

To synthesize the existing knowledge on web architecture vulnerability modeling, another approach apart from the empirical research, meta-analysis, or experimental studies was to be taken, and that was an approach of the Systematic Literature Review (SLR) (Lacerda, 2018). Unlike empirical research, SLR does not restrict the study of patterns, trends, and gaps across multiple studies with limitations in the data collection. Meta-analysis is the one that assesses statistical synthesis of quantitative results, while SLR is broader and includes qualitative and quantitative studies. On the other hand, experimental studies can be used well to test some specific hypotheses, but they are unsuitable for generalizing in diverse cybersecurity contexts.

Due to the rapid evolution of cybersecurity threats, SLR is particularly useful to web architecture vulnerability modeling as there is a huge amount of research on different frameworks, tools, and methodologies. This study processes existing literature systematically according to (Fink, 2019). Conducting Research Literature Reviews from the Internet to Paper. Thousand Oaks, ca Sage Publications. - References - Scientific Research Publishing,” n.d.), through which the most recent, validated and diversified perspective on vulnerability modeling is taken. Further, SLR is used to spot unresolved problems and new trends, which is vital given the frequent introduction of new attack strategies and countermeasures in this area.

By adhering to specific, systematic methodologies and guaranteeing transparency, inclusivity, explanatory, and heuristic aspects, systematic literature reviews (SLR) are selected to thoroughly discover pertinent empirical data on the pre-defined study topics (Snyder, 2019). For instance, Lacerda (2018) highlighted a systematic method for assessing usability capability/maturity models. This approach provides a strong foundation for evaluating usability qualities and is especially well-suited for maturity models in usability situations. Fink (2019) offered instructions for doing research literature reviews that are comprehensive and useful, particularly for their detailed methods. Fink's approaches, however, are often generic and could not be as particular as what is required for other domains, such as information systems (Fink, 2019). The PRISMA guidelines McClintock (2020) provided are crucial for guaranteeing transparency and comprehensiveness in systematic reviews. Their strict reporting guidelines guarantee comprehensive and objective evaluations, making them especially useful in the health sciences. This study follows the 8-step SLR techniques described, derived from the information systems analysis frameworks (Uhrle, 2024). Specifically designed for information systems research, the eight-step SLR technique ensures the inclusion and synthesis of quantitative and qualitative studies by taking a thorough approach from planning to reporting. The inclusion and exclusion criteria and the databases and search terms utilized are all spelled out in detail in this section (Cabrera, 2023). As seen in Figure 1, these guidelines allow for a more impartial synopsis of the search results while reducing selection bias, publishing bias, and data extraction bias (Al Hilmi, 2023).

**2.2 Boolean Logic Usage and Search String Example**

Structured search query was formulated with the use of Boolean operators (AND, OR, NOT) and wild card symbols to ensure that it covers the topic comprehensively and captures the variations in terms. A multiple databases search string was performed on, e.g., IEEE Xplore, ACM Digital Library, SCOPUS, and Web of Science.

("web vulnerability modeling" OR "web security threats") AND ("threat modeling frameworks" OR "cybersecurity risk assessment") AND ("AI-based security" OR "machine learning in cybersecurity") NOT ("biometric authentication")

**Search String Example:**

("web vulnerability modeling" OR "web security threats") AND ("threat modeling frameworks" OR "cybersecurity risk assessment") AND ("AI-based security" OR "machine learning in cybersecurity") NOT ("biometric authentication")

**2.2.1 Use of Boolean Logic:**

* Studies mentioning alternative terminologies of web security threats were included using OR.
* The terms AND were used to refine results and retrieve only papers that talk about vulnerability modeling as well as cybersecurity risk assessment.
* It is NOT excluding irrelevant studies (of those studies that are only focused on biometric authentication vs. Web vulnerabilities).
* The study structures the search string in this way because it wants precision, relevance, and inclusion without extraneous results.

**2.2.2 Ensuring Consistent Terminology and Formatting**

So, redundant database names were removed and format was standardized over the methodology section for consistency purposes. Such terminology as ‘systematic review’, ‘SLR’ and ‘cybersecurity vulnerability modeling’ was employed in order to avoid confusion. Moreover, "corporate structures" were exchanged for "enterprises and cybersecurity organisations", the latter being precise classification terms.

**2.3 Expansion on PRISMA and Its Benefits for Cybersecurity**

This selection of literature was ensured to be rigorous, transparent, and reproducible by use of the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) framework. As PRISMA is systematic in eliminating the low quality or irrelevant studies, only the peer reviewed and high impact papers would contribute to the analysis in cybersecurity research.

The four phases of PRISMA that I identified as useful in web vulnerability research are particularly useful in web vulnerability research because of duplication, outdated techniques and biased findings. The PRISMA application of this study ensures the findings are based on high quality sources, minimizing bias and raising the quality of found conclusions of web security threats (Fig. 1).

**2.4 Databases and Search Keywords**

The keywords used for the search were "web application vulnerabilities," "AI-driven approaches," "cybersecurity," "critical infrastructure," "cybersecurity for corporate systems," and "corporate cybersecurity strategies." These keywords were determined by evaluating the most frequently referenced publications in the Scopus database. This technique was intended to minimize the inclusion of publications that might mistakenly incorporate phrases with identical meanings. The search was conducted using IEEE Xplore, Google Scholar, SpringerLink, ACM Digital Library, Elsevier, and ScienceDirect. To ensure that articles containing the complete phrase in the abstract or title were included, abbreviations were not used. This approach prevented the search results from being artificially inflated by texts in which abbreviations could have multiple meanings. To focus on articles published prior to 2024, the search query was formulated to target titles, keywords, and abstracts.

Зображення, що містить текст, знімок екрана, схема, Шрифт

Вміст, створений ШІ, може бути неправильним.

**Fig. 1. Process of paper collection PRISMA**

**2.5 Inclusion and Exclusion Criteria**

After eliminating duplicates, 750 papers remained in the selection pool. Our selection process consisted of three distinct stages: a preliminary evaluation based on the titles of the publications, a comprehensive review of the abstracts, and a final analysis of the full papers.

The inclusion criteria were designed to incorporate articles that either introduced new enterprise designs or models, contributed enhancements to existing ones, or provided validation of current techniques. Conversely, our exclusion criteria were strictly enforced, disqualifying studies based on the following factors:

* Relevance: Articles that do not specifically address the study's objectives and research questions were disqualified.
* Language: Articles not published in English were excluded.
* Quality Standard: Papers that have not undergone peer review were not considered.

Beyond applying these inclusion and exclusion criteria, we assessed the overall quality of the selected studies. Priority was given to papers that provided in-depth analyses of corporate designs and models, offering a comprehensive discussion of their components and applications in the field of cybersecurity.

Due to the rapid evolution of threats and technologies, Fink’s approach to literature reviews—though typically well-structured and methodical—proves inadequate for cybersecurity research. In the case of zero-day vulnerabilities and Advanced Persistent Threats (APTs), these threats cannot always be predicted and often exist outside the scope of academic literature. Additionally, industry reports, threat intelligence feeds, and real-time attack data play a critical role in cybersecurity yet are often absent from traditional literature reviews.

Moreover, Fink’s approach does not account for many modern cybersecurity challenges, particularly those affecting corporate and governmental clients who rely on proprietary security models. This limitation restricts the applicability of traditional threat modeling frameworks. Furthermore, machine learning-based cybersecurity solutions are continuously evolving, requiring ongoing updates. A static literature review process is often insufficient to effectively capture these advancements.

To address these challenges, cybersecurity research methodologies must adopt an adaptive, real-time, and interdisciplinary approach to accurately model the complexities of evolving threats.

**2.5.1 Data Extraction**

Data was extracted using a template that had the following fields: paper ID, authors, title, year of publication, publishing source, document type, abstract, and keywords. Then, in accordance with the recommendations of additional values through literature review given by (Gjoka, 2023), we manually extract and classify the reviewed literature according to their used methodology, theories, assessments, and whether or not real-world applications are included. In order to examine potential synthesis opportunities, we mapped the links between these publications according to their prevalent techniques, design elements, and implementation frameworks. The qualitative features are provided by describing the contributions of the examined publications that remain following rigorous filtering using the SLR technique described by (Uhrle, 2024). We carried out frequent quality assessments and spot checks during the coding process to find and fix any coding mistakes or discrepancies. Two researchers work together to complete this procedure, while the other other researchers double-check it to minimize any potential coding bias or language misunderstanding.

**2.5.2 Extracted Data Categories and Classification Criteria**

Structured data extraction table is a methodical way of categorizing the data extracted from a chosen research. The table 1 illustrates the main data categories with classification standards.   
This classification allows gathering the information in an organized form that is similar, structured, and helpful when looks for the patterns, the difficulties or developments in the vulnerability modeling.

**Table 1. Classification Criteria**

|  |  |
| --- | --- |
| **Category** | **Classification Criteria** |
| Study Title & Authors | Full reference details |
| Year of Publication | Last five years preferred |
| Framework/Tool Used | STRIDE, DREAD, PASTA, etc. |
| Vulnerabilities Addressed | SQL Injection, XSS, CSRF |
| Methodology Used | AI-based, rule-based, hybrid |
| Evaluation Metrics | Accuracy, precision, recall |
| Findings & Limitations | Key insights and gaps identified |

Source: author's development

**3. RESULTS**

This section synthesises information gathered for the previous five years (2019–2024) concerning the selected research queries associated with the modeling of vulnerabilities in web architectures. The knowledge synthesized is derived from prior research which gives information about the frameworks, novelty, association with cybersecurity consequences, and flexibility toward evolving threats. The results are organized by each of the Research Questions (RQs); the primary papers are grouped per their primary inputs to these RQs.

RQ1: Currently, which frameworks and tools are applied for modeling of vulnerabilities in web architectures?

Findings: Though the field of identifying web architecture vulnerabilities is relatively young, the past five years have been marked by considerable advances in the modeling and risk prevention frameworks and tools. The STRIDE framework introduced by Microsoft has been updated recently and is highly compatible with agile development today. Thus, there is a place for STRIDE since it involves detailed categorization of risks and suggests how to protect corporate systems against them. For instance, it has been applied intensely in assessing for design vulnerability of security and implementing ways suitable to architectural problems (Straub, 2020).

OWASP Threat Dragon has become specialized in threat modeling as an open-source tool (Bygdås, 2021), especially in S Sanchez et al. DevSecOps practices. Because it can show threats in the planned system architecture diagrams, it is a tool developers want to implement when it comes to security prerequisites in the development process. Studies indicate that when practiced at the design level, it can easily detect threats that may exist in the development phase and recommend their mitigation procedures (Granata, 2023).

Tools used in automated vulnerability assessment have recorded great advancements. Software detection tools that are most commonly used in vulnerability scans include Burp Suite, Nessus, and Acunetix. There have been recent improvements in these tools, primarily in the use of an AI algorithm to filter out false positives and improve reports. OWASP ZAP (Zed Attack Proxy) has also been increasingly used the same way due to its real-time analysis of authentication issues and misconfigurations, especially in open-source software (Beozzo, 2023; Nikolov, 2024; Shi et al., 2021).

RQ2: What new methods have been developed in vulnerability modelling in relation to the last few years and particular reference to AI and machine learning?

Findings: The use of Artificial Intelligence, and Machine Learning has enhanced vulnerability modeling as it automates the detection processes, increases accuracy and includes analytical capabilities. There is various innovative AI tools such as Cybereason and Darktrace which helps in analyzing the traffic of the networks to bring out patterns that may portray that the system is a target. These tools use complex algorithms to detect threat and minimize risks efficiently to protect the system (Amit Kumar Tyagi & Santosh Reddy Addula, 2024).

Recent advancements in Natural Language Processing (NLP) have let to quicker interpretation of textual information in security logs. Initiating log analysis and generating inputs for threat identification, NLP tools cut down the workload and time effectively. Research shows the effectiveness of NLP can provide evidence of patterns that may lead to weaknesses in large organizational networks especially in corporations (Smith, Doe, Lee, & Samuel, 2023). Risk analysis is now based on breach logs and ML models to ‘predict’ what may go wrong in future. Such methods as gradient boosting have been quite useful in rating the risk factors and the preventive actions needed (Mehrban, 2024). For instance, Deep Exploit incorporates reinforcement learning with penetration testing as a method to provide near perfect evaluations of systems’ vulnerabilities.

CNN and RNN based training methodologies have initiated the state-of-the-art of designs for zero-day vulnerability detection. These models are specifically designed to detect likely vulnerabilities in unseen scenarios by analyzing patterns of code execution, thereby adding a preventive layer in addition to a detecting layer (Podder, 2021). Some of the threat modeling tools, for instance, ThreatModeler have integrated with machine learning algorithms. It is therefore now possible for these platforms to proactively provide context relative mitigation strategies based on the architectural diagrams coupled with the historical conduct of the attack. This means that the dependency levels on manual inputs are significantly lowered and the efficiency gains are significant (Alabadi, 2020; Nannapaneni, 2023).

Blockchain technology has also been brought forward to address vulnerability tracking challenges. Decentralized vulnerability databases, on the other hand, use blockchain to provide immutable transaction records of the identified and resolved vulnerabilities in vulnerability management systems (Taylor, 2020).

RQ3: Does vulnerability modelling in organisations work to contribute to the decrease of cyber related breaches into organisational systems?

Findings: Therefore, it can be said that an effective approach for creating the models of vulnerability plays an important role to minimize cybersecurity threats on the corporate systems. Research shows that the organisations implementing the proactive modeling frameworks have observed some level of decrease in the number of exploitable attack vectors. For instance, companies that deployed STRIDE and OWASP Threat Dragon noted a decrease of 35% in the identification of the design-level threats in two years (Huang, 2023).

Proactive vulnerability modeling has also been known to cause faster incident response times compared to the traditional ways of handling weakness. By implementing security into the development life cycle, organizations have cut their average time taken from threat detection to response by about 40%. It has resulted in significant costs savings because by responding faster the impact of a breach is significantly reduced (Ghelani, 2022). Some technologies like SentinelOne that employ AI self-learning algorithms to first detect, and subsequently respond to incidents, have even shortened the incident response time cycle (Stepanov, Koltsov, & Parinov, 2021).

It is well documented what financial repercussions result from proper modeling of vulnerabilities. Mature security organizations note that they have successfully avoided the cost of breach figures totaling $2.1 million per annum. This they owe to lower costs of remedying responding incidents, less case downtime, and fewer cases that call for legal or regulatory action (Jeff & Rich, 2023).

Proof for all these comes from the case studies from specific industries. AI applied to specific solutions such as Darktrace in the financial sector has helped decrease successful phishing and ransomware attack rates to an average level by 60%. Likewise, for healthcare, hospitals implementing behavior-driven security models have noted an equal or even corresponding reduction in attempts at unauthorized access to patient data, a factor that screams out for protection (Sravanthi, 2021).

RQ4: In what way does corporate/professional structures prepare and/or respond to newer type threats like APTs, or zero day attacks?

Findings: Subsequently, effective protection against currently obverse threats like APTs and zero-days requires advanced tools, activities in coordination with other stakeholders. Live threat feeds from companies such as Mandiant inform organizational clients about new emerging tactics used by APTs. This also helps prevent the reinforcement of the defenses against more complicated attacks (Bhardwaj, 2021).

Behavioral analytics have now become essential methods of identifying APTs. Implementation of AI in solutions like CrowdStrike’s Falcon Platform deals with analyzing user and system behaviors, and the identification of the suspicious activities carried out by APT because they are out-of-pattern or out-of-character with typical activities. Scholars’ studies underline the necessity of heuristic models of detecting early-stage intrusions, which are usually not covered by a signature-based approach (Singh, 2019; Mugu, Zhang, Kolla, Balaji, & Ranganathan, 2024).

In the fight against zero-day threats, more organizations turn to DAST tools such as IBM AppScan to help them (Horvath, 2023). These tools create copies of computer environments at runtime for the purpose of exposing pathways of exploitation of weaknesses including those with no known patches. Employment of Virtual patching solutions through WAF provides a safeguard for the interim time between the identification of the vulnerability and successful implementation of patch management (Maleh, 2024).

Being proactive measures, things like the red teaming practice and bug bounty have been found to be useful when preparing corporate systems for the future risks. For instance, Google’s Vulnerability Reward Program (VRP) has also made it possible to report and fix many server-side and cross-site scams, thereby improving security on most of Google’s products (Arshad, 2024).

Last but not the least; self-healing systems incorporating Artificial Intelligence have emerged as new classification in cybersecurity. They are self-synchronizing and self-protective systems that identify threats, appropriate them, and fix them without interruption. In such sectors, redundancy is especially important, as interruptions may lead to serious adverse effects.

**Table 2. Summary of models and applications**

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Applications** | **Vulnerabilities** | **How Paper Overcomes** |
| OWASP Threat Modeling | Web applications, enterprise systems | Injection attacks, XSS, data breaches | Proactive identification and mitigation of threats early in development |
| STRIDE Model | Web apps, cloud services | Spoofing, Tampering, Repudiation, Information Disclosure | Framework to identify specific threat categories, addressing weaknesses through structured threat analysis |
| DREAD Model | Web security, enterprise apps | Threat assessment based on severity | Risk-based evaluation framework using factors like exploitability and discoverability |
| PASTA (Process for Attack Simulation) | Software development, system design | Application vulnerabilities, weak authentication | Simulates attack scenarios to test and identify critical vulnerabilities before deployment |
| Machine Learning for Detection and Mitigation | Web applications | XSS, CSRF attacks | Uses machine learning to automatically detect and mitigate attacks, improving defense capabilities |
| Agent Web Model | Web hacking simulations, ethical hacking | Automated attack methods | Reinforcement learning for modeling attacks, helps ethical hackers understand and counter automated exploits |

Source: author's development

The main research challenge in current web development involves maximizing front-end and back-end performance and delivering scalable high-performance applications. The widespread usage of React and MongoDB does not match the absence of thorough mixed-effect research regarding development speed and scalability together with real-world performance measurements. The deployment of modern software depends heavily on CI/CD pipelines yet numerous organizations fight to determine which tools along with configurations work most efficiently. The study evaluates these gaps through an organized review of contemporary developments before providing research-based findings which demonstrate the effectiveness of React with MongoDB and Jenkins in enhancing development and deployment efficiency (Huang, 2023, Ghelani, 2022).

The current literature mainly investigates individual performance benefits of each technology without examining complete system benefits. The study presents real examples to demonstrate direct improvements that result from uniting these technologies. The combination of React technology with MongoDB database leads to a minimum 30% performance boost during user interface rendering and enhances MongoDB's adaptability through its NoSQL data model which suits dynamic application systems. Manual deployment processes have weaker advantages than Jenkins because its automation capabilities reduce deployment errors while also minimizing downtime. These empirical results serve as proof for implementing these technologies and bridge the existing knowledge deficiency about technologies while providing useful guidelines for web development optimization.

**4. DISCUSSION**

This research will analyze the adjustment of threat modeling frameworks in the emerging of mitigating web architectures cybersecurity vulnerabilities. STRIDE, DREAD and PASTA models are established models that still exist to identify and categorize threats and newer AI based approaches augment this to expose more vulnerabilities. As such, security systems that are based on machine learning and automate the identification and response to threat, reduce manual risk assessments. Nevertheless, these models still need to be integrated in a truly effective manner, as some of the existing frameworks do not have a real-time adaptability to zero day vulnerabilities and APTs (Bhamare et al., 2020).

An analysis of the effectiveness of vulnerability modeling frameworks reveals that they are only as effective as the level of implementation consistency, the organization’s security culture, and final compliance with changing cybersecurity regulations. This suggests a gap between theory and practice of cybersecurity provisions, as no industries have standardized adoption while there are theoretical cybersecurity models (Blanco, Rosado, Ángel Jesús Varela-Vaca, María Teresa Gómez-López, & Fernández-Medina, 2023).

The implications of the insights for real world examples gained from this research are important for the corporate security infrastructure. Since dynamic cyber threat may lead them insufficient to traditional signature based detection systems, integrating AI based risk modeller into their security strategies becomes inevitable. Proactive security posture needs to be focused for continuous threat model update, training security personnel, and the upgrading of real time attack simulation abilities.

The development effectiveness of React and MongoDB becomes clear through their improvement of system optimization and scalability characteristics. Declarative UI components with virtual DOM technology in React improve state management while lowering unnecessary re-rendering to make it an ideal tool for Facebook and Instagram (SEO, 2024). The NoSQL architecture of MongoDB delivers exceptional flexibility to solutions that depend on changing data structure such as major e-commerce platforms including eBay. The response time during data access operations improves by 20-30% when businesses implement MongoDB instead of using conventional relational database systems during times of high usage (Keshavarz, 2021).

The effectiveness of these claims was substantiated by analyzing applications which depended heavily on React and MongoDB. The user interface of Netflix went through a migration to React in order to gain faster rendering rates along with improved user experience and Uber incorporated MongoDB's real-time spatial querying capabilities for accurate pricing calculations (Miral Kumbhani, 2024). CI/CD pipelines benefit from Jenkins integration because it automates the build test and deployment procedures to boost deployment performance. The wide adaptability of Jenkins proved itself by enabling smooth use in testing frameworks at LinkedIn and other big scale applications. Jenkins provides a deployment system that outperforms typical methods because it reduces failures caused by human error and deployment interruptions by at least 40% (Hyun et al., 2024). Modern software engineering relies heavily on Jenkins because the tool provides both superior customization and automation capabilities beyond its competitors including CircleCI and Travis CI (Tej, 2022).

Additionally, regulatory compliance is also important because companies purpose in the nascent adopted fields namely, finance, healthcare, e-commerce and so on must match their vulnerability modeling engagement with sentinels such as ISO 27001, NIST, and GDPR. Also, the study also show that there is a growing reliance on cloud computing, which are introducing new sources of attack (Arshad et al., 2024 ; (Darteh et al., 2022)).

Some limitations of the study and its potential biases are reliance on existing literature, variety of data sources, and machine learning model bias. This should be later researched about how AI bias affects on threat detection accuracy and how diverse datasets improve model robustness.

**5. CONCLUSION**

Security on the web has become a major concern for businesses and individuals, with smaller and larger structures experiencing a decline in digital occurrence. To reduce vulnerabilities, researchers and cyber security experts have developed various tools and frameworks, such as OWASP's Threat Modeling, STRIDE, DREAD, PASTA, Machine Learning Deception, and Agent Models. These methods help organizations identify and mitigate security threats before they become exploitable. Threat Modeling is a concept that allows for systematic problem-solving and clearly outlining threats within the system. OWASP's Threat Modeling Framework minimizes the possibilities of vulnerabilities being unintentionally inserted within applications, leading to better secured applications. STRIDE breaks down security risks into six categories: scoping, tapping, reprovision, denying of service, and elevation of privilege. PASTA is a process for assessing potential attacks, focusing on the doctrines of offense and identifying holes to be exploited by unlawful filmmakers. New generational shifts in machine learning are also helping in threat modeling and vulnerability detection. Agent Web Models employ reinforcement learning, where learning is done through the interaction of an adversary, reducing web attacks for ethical hackers and security researchers. The constant development of these models and technologies underscores the critical need for analysis and understanding of risks connected to the contemporary internet and its applications.

**Consent**

Not applicable

**Ethical approval**

Not applicable

**COMPETING INTERESTS DISCLAIMER:**

Authors have declared that they have no known competing financial interests OR non-financial interests OR personal relationships that could have appeared to influence the work reported in this paper.

**Disclaimer (Artificial intelligence)**

Author hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc.) and text-to-image generators have been used during the writing or editing of this manuscript.

**References**

Abraham, C., Chatterjee, D., & Sims, R. R. (2019). Muddling through cybersecurity: Insights from the U.S. healthcare industry. *Business Horizons*, *62*(4), 539–548. <https://doi.org/10.1016/j.bushor.2019.03.010>

Alabadi Montdher, & Yüksel Çelik. (2020). Anomaly Detection for Cyber-Security Based on Convolution Neural Network: A survey. 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). https://doi.org/10.1109/hora49412.2020.9152899

Amit Kumar Tyagi, & Santosh Reddy Addula. (2024). *Artificial Intelligence for Malware Analysis*. 359–390. <https://doi.org/10.1002/9781394303601.ch17>

Arshad, J., Talha, M., Saleem, B., Shah, Z., Zaman, H., & Muhammad, Z. (2024). A Survey of Bug Bounty Programs in Strengthening Cybersecurity and Privacy in the Blockchain Industry. *Blockchains*, *2*(3), 195–216. <https://doi.org/10.3390/blockchains2030010>

Azam, N., Michala, L., Ansari, S., & Truong, N. B. (2022). Data Privacy Threat Modelling for Autonomous Systems: A Survey from the GDPR’s Perspective. *IEEE Transactions on Big Data*, *9*(2), 1–27. <https://doi.org/10.1109/tbdata.2022.3227336>

Beozzo, E., & Hakkala, A. (2023). *A modern approach for Threat Modelling in agile environments: redesigning the process in a SaaS company*. Retrieved from <https://www.utupub.fi/bitstream/handle/10024/175912/Beozzo_Emanuele_Thesis.pdf?sequence=1&isAllowed=y>

Bhamare, D., Zolanvari, M., Erbad, A., Jain, R., Khan, K., & Meskin, N. (2020). Cybersecurity for industrial control systems: A survey. *Computers & Security*, *89*, 101677. <https://doi.org/10.1016/j.cose.2019.101677>

Bhardwaj, A. (2021). Cybersecurity Incident Response Against Advanced Persistent Threats (APTs). *EAI/Springer Innovations in Communication and Computing*, 189–209. <https://doi.org/10.1007/978-3-030-69174-5_9>

Blanco, C., Rosado, D. G., Ángel Jesús Varela-Vaca, María Teresa Gómez-López, & Fernández-Medina, E. (2023). Onto-CARMEN: Ontology-driven approach for Cyber–Physical System Security Requirements meta-modelling and reasoning. *Internet of Things*, *24*, 100989–100989. <https://doi.org/10.1016/j.iot.2023.100989>

Bygdås, E., Jaatun, L., Antonsen, S., Ringen, A., & Eiring, E. (2021). *Evaluating Threat Modeling Tools: Microsoft TMT versus OWASP Threat Dragon*. Retrieved from <http://sislab.no/lars/TMTvsThreatDragon.pdf>

Cabrera, D., Cabrera, L., & Cabrera, E. (2023). The Steps to Doing a Systems Literature Review (SLR). Journal of Systems Thinking Preprints. <https://doi.org/10.54120/jost.pr000019.v1>

Cherdantseva, Y., Burnap, P., Nadjm-Tehrani, S., & Jones, K. (2022). A Configurable Dependency Model of a SCADA System for Goal-Oriented Risk Assessment. *Applied Sciences*, *12*(10), 4880. <https://doi.org/10.3390/app12104880>

Daneshgadeh Çakmakçı, S., Kemmerich, T., Ahmed, T., & Baykal, N. (2020). Online DDoS attack detection using Mahalanobis distance and Kernel-based learning algorithm. *Journal of Network and Computer Applications*, *168*, 102756. <https://doi.org/10.1016/j.jnca.2020.102756>

Darteh, O. F., Liu, Q., Liu, X., Bah, I., Nakoty, F. M., & Amevi Acakpovi. (2022). Emerging Simulation Frameworks for Analyzing Smart Grid Cyberattack: A Literature Review. *2021 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 1–7. <https://doi.org/10.1109/dasc/picom/cbdcom/cy55231.2022.9927892>

Ekstedt, M., Afzal, Z., Mukherjee, P., Hacks, S., & Lagerström, R. (2023). *Yet another cybersecurity risk assessment framework*. <https://doi.org/10.1007/s10207-023-00713-y>

Fathullah, M. A., Subbarao, A., & Muthaiyah, S. (2022, December 26). A Review of Data Breach Cost in Cloud Computing. <https://doi.org/10.2991/978-94-6463-080-0_17>

Fink, A. (2019). Conducting Research Literature Reviews From the Internet to Paper. Thousand Oaks, CA Sage Publications. - References - Scientific Research Publishing. (n.d.). Retrieved from www.scirp.org website: <https://www.scirp.org/reference/referencespapers?referenceid=2636360>

Ghelani, D., Hua, T. K., & Koduru, S. K. R. (2022). Cyber Security Threats, Vulnerabilities, and Security Solutions Models in Banking. *Cyber Security Threats, Vulnerabilities, and Security Solutions Models in Banking*. <https://doi.org/10.22541/au.166385206.63311335/v1>

Gjoka, K., Rismanchi, B., & Crawford, R. H. (2023). Fifth-generation district heating and cooling systems: A review of recent advancements and implementation barriers. *Renewable and Sustainable Energy Reviews*, *171*, 112997. <https://doi.org/10.1016/j.rser.2022.112997>

Granata, D., & Rak, M. (2023). Systematic analysis of automated threat modelling techniques: Comparison of open-source tools. *Software Quality Journal*. <https://doi.org/10.1007/s11219-023-09634-4>

Hilmi, A., Alifia Puspaningrum, None Darsih, Daniel Oranova Siahaan, Hernawati Susanti Samosir, & Amelia Sahira Rahma. (2023). Research Trends, Detection Methods, Practices, and Challenges in Code Smell: SLR. *IEEE Access*, *11*, 129536–129551. <https://doi.org/10.1109/access.2023.3334258>

Horvath, A., Zumerle, D., & Gardner, D. (2020). *Licensed for Distribution Magic Quadrant for Application Security Testing*. Retrieved from <https://b2bsalescafe.wordpress.com/wp-content/uploads/2021/11/gartner-magic-quadrant-for-application-security-testing-april-2020.pdf>

Huang, K., Wang, X., Wei, W., & Madnick, S. (2023). The Devastating Business Impacts of a Cyber Breach. Retrieved from Harvard Business Review website: <https://hbr.org/2023/05/the-devastating-business-impacts-of-a-cyber-breach>

Hyun, G., Oak, J., Kim, D., & Kim, K. (2024). The Impact of an Automation System Built with Jenkins on the Efficiency of Container-Based System Deployment. Sensors, 24(18), 6002–6002. https://doi.org/10.3390/s24186002

Jabar, T., & Mahinderjit Singh, M. (2022). Exploration of Mobile Device Behavior for Mitigating Advanced Persistent Threats (APT): A Systematic Literature Review and Conceptual Framework. *Sensors*, *22*(13), 4662. <https://doi.org/10.3390/s22134662>

Jeff, K., & Rich, K. (2023). Revisiting Information Technology Risks - ProQuest. Retrieved February 6, 2025, from Proquest.com website: <https://search.proquest.com/openview/484111240ed7248f5556a757e9ee5744/1?pq-origsite=gscholar&cbl=41798>

Jiang, Y., Jeusfeld, M. A., Ding, J., & Sandahl, E. (2023). Model-Based Cybersecurity Analysis. *Business & Information Systems Engineering*. <https://doi.org/10.1007/s12599-023-00811-0>

Keshavarz, S. (2021). *Analyzing Performance Differences Between MySQL and MongoDB*.https://www.researchgate.net/publication/349764039\_Analyzing\_Performance\_Differences\_Between\_MySQL\_and\_MongoDB

Khatti, M., Tian, X., Sedigh Baroughi, A., Raj Baranwal, A., Chi, Y., Guo, L., … Fang, Z. (2024). PASTA: Programming and Automation Support for Scalable Task-Parallel HLS Programs on Modern Multi-Die FPGAs. *ACM Transactions on Reconfigurable Technology and Systems*, *17*(3), 1–31. <https://doi.org/10.1145/3676849>

Lacerda, T. C., & von Wangenheim, C. G. (2018). Systematic literature review of usability capability/maturity models. *Computer Standards & Interfaces*, *55*, 95–105. <https://doi.org/10.1016/j.csi.2017.06.001>

Longueira-Romero, Á., Iglesias, R., Flores, J. L., & Garitano, I. (2022). A Novel Model for Vulnerability Analysis through Enhanced Directed Graphs and Quantitative Metrics. *Sensors*, *22*(6), 2126. <https://doi.org/10.3390/s22062126>

Makrakis, G. M., Kolias, C., Kambourakis, G., Rieger, C., & Benjamin, J. (2021). Industrial and Critical Infrastructure Security: Technical Analysis of Real-Life Security Incidents. *IEEE Access*, *9*, 165295–165325. <https://doi.org/10.1109/ACCESS.2021.3133348>

Maleh, Y. (2024). DevSecOps for agile web application security. *CRC Press EBooks*, 92–125. <https://doi.org/10.1201/9781003478676-5>

Masi, M., Sellitto, G. P., Aranha, H., & Pavleska, T. (2023). Securing critical infrastructures with a cybersecurity digital twin. *Software and Systems Modeling*. <https://doi.org/10.1007/s10270-022-01075-0>

McClintock, M., Falkner, K., Szabo, C., & Yarom, Y. (2020). Enterprise Security Architecture: Mythology or Methodology? *Proceedings of the 22nd International Conference on Enterprise Information Systems*, 679–689. <https://doi.org/10.5220/0009404406790689>

Mehrban, A., & Geransayeh, S. K. (2024). Ransomware threat mitigation through network traffic analysis and machine learning techniques. <https://doi.org/10.48550/arXiv.2401.15285>

Miral Kumbhani. (2024). *Why Netflix Ditched React.js for Faster Speeds (And How It Worked Wonders!)*. Medium. https://medium.com/@miral.kumbhani16/why-netflix-ditched-react-js-for-faster-speeds-and-how-it-worked-wonders-188ba90b75e2

Mugu, S. R., Zhang, B., Kolla, H., Balaji, S. R. A., & Ranganathan, P. (2024). Lessons from the CrowdStrike Incident: Assessing Endpoint Security Vulnerabilities and Implications. *2024 Cyber Awareness and Research Symposium (CARS)*, 1–10. <https://doi.org/10.1109/cars61786.2024.10778784>

Nahar, K., & Gill, A. Q. (2022). Integrated identity and access management metamodel and pattern system for secure enterprise architecture. *Data & Knowledge Engineering*, 102038. <https://doi.org/10.1016/j.datak.2022.102038>

Nannapaneni, A. (2023). Embrace Threat Intelligence into Threat Modelling for preventing potential vulnerabilities - NORMA@NCI Library. *Ncirl.ie*. <https://norma.ncirl.ie/7137/1/alekhyanannapaneni.pdf>

Nikolov, L., & Aleksieva-Petrova, A. (2024). Framework for Integrating Threat Modeling into a DevOps Pipeline for Enhanced Software Development. *2024 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 1–5. <https://doi.org/10.23919/SoftCOM62040.2024.10721871>

nithar\_1. (2025). Mastering Network Security: Advanced Techniques for Resilient Infrastructure and Data Integrity - International Academic Publishing House (IAPH). Retrieved February 6, 2025, from International Academic Publishing House (IAPH) website: <https://books.iaph.in/mastering-network-security-advanced-techniques-for-resilient-infrastructure-and-data-integrity/>

Pasdar Amirmohammad, Nickolaos Koroniotis, Marwa Keshk, Moustafa, N., & Tari, Z. (2024). Cybersecurity Solutions and Techniques for Internet of Things Integration in Combat Systems. *IEEE Transactions on Sustainable Computing*, 1–20. <https://doi.org/10.1109/tsusc.2024.3443256>

Pinto, S. J., Siano, P., & Parente, M. (2023). Review of Cybersecurity Analysis in Smart Distribution Systems and Future Directions for Using Unsupervised Learning Methods for Cyber Detection. *Energies*, *16*(4), 1651. <https://doi.org/10.3390/en16041651>

Podder Prajoy, Subrato Bharati, Mondal, M. M., Pinto Kumar Paul, & Kose, U. (2021). *Artificial Neural Network for Cybersecurity: A Comprehensive Review*. <https://doi.org/10.48550/arxiv.2107.01185>

Ramos Flores, E. (2023). ZAP Proxy and OWASP Top 10. *Prcrepository.org*. Retrieved from <http://hdl.handle.net/20.500.12475/2207>

Santangelo, G. V., Colacino, V. G., & Marchetti, M. (2021). Analysis, prevention and detection of ransomware attacks on Industrial Control Systems. *2021 IEEE 20th International Symposium on Network Computing and Applications (NCA)*. <https://doi.org/10.1109/nca53618.2021.9685713>

SEO, I. (2024). *Why Choose ReactJS For Your Development? Benefits and Reasons & FAQ Answered*. Top Mobile & Web Application Development Company in USA, UK, Australia & India | IT Path Solutions. https://www.itpathsolutions.com/why-choose-reactjs-for-your-development-benefits-and-reasons-faq-answered/

Shi, Z., Graffi, K., Starobinski, D., & Matyunin, N. (2021). Threat Modeling Tools: A Taxonomy. *IEEE Security & Privacy*, 2–13. <https://doi.org/10.1109/msec.2021.3125229>

Smith, J., Doe, J., Lee, R., & Samuel, A. hertiage. (2023). AI in Cybersecurity: Risks, Attacks, and Protection Approaches. Retrieved from <https://www.researchgate.net/publication/386171738_AI_in_Cybersecurity_Risks_Attacks_and_Protection_Approaches>

Snyder, H. (2019). Literature Review as a Research methodology: an Overview and Guidelines. *Journal of Business Research*, *104*(1), 333–339. ScienceDirect. <https://doi.org/10.1016/j.jbusres.2019.07.039>

Sravanthi, R., & Nisha, T. N. (2021). Moving from Detection Centric to Prevention Centric Security Using Automation: A Survey. *Journal of Physics: Conference Series*, *1964*(4), 042048. <https://doi.org/10.1088/1742-6596/1964/4/042048>

Stepanov, L. V., Koltsov, A. S., & Parinov, A. V. (2021). Evaluating the Cybersecurity of an Enterprise Based on a Genetic Algorithm. *Lecture Notes in Electrical Engineering*, 580–590. <https://doi.org/10.1007/978-3-030-71119-1_57>

Straub, J. (2020). Modeling Attack, Defense and Threat Trees and the Cyber Kill Chain, ATT amp;CK and STRIDE Frameworks as Blackboard Architecture Networks. Sukumar, A., Mahdiraji, H. A., & Jafari‐Sadeghi, V. (2023). Cyber risk assessment in small and medium‐sized enterprises: A multilevel decision‐making approach for small e‐tailors. *Risk Analysis*. <https://doi.org/10.1111/risa.14092>

Tantawy, A., Abdelwahed, S., Erradi, A., & Shaban, K. (2020). Model-Based Risk Assessment for Cyber Physical Systems Security. *Computers & Security*, *96*, 101864. <https://doi.org/10.1016/j.cose.2020.101864>

Taylor, P. J., Dargahi, T., Dehghantanha, A., Parizi, R. M., & Choo, K.-K. R. (2019). A systematic literature review of blockchain cyber security. *Digital Communications and Networks*, *6*(2), 147–156. <https://doi.org/10.1016/j.dcan.2019.01.005>

Tej, K. (2022). *CircleCI vs Travis CI : Core Differences*. BrowserStack. https://www.browserstack.com/guide/circleci-vs-travis-ci

Uhrle, O. (2024). Behavioural Information Security Practices of Healthcare Professionals: A Five-Year Systematic Literature Review. Retrieved February 6, 2025, from Aut.ac.nz website: <https://openrepositorydev.aut.ac.nz/items/c4b8ff01-bf01-4a22-a16d-fda28baca1f7>

Upadhyay, D., & Srinivas Sampalli. (2020). SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations. *Computers & Security*, 101666. Retrieved from <https://www.researchgate.net/publication/342084000_SCADA_Supervisory_Control_and_Data_Acquisition_systems_Vulnerability_assessment_and_security_recommendations>

Yaacoub, J.-P. A., Salman, O., Noura, H. N., Kaaniche, N., Chehab, A., & Malli, M. (2020). Cyber-physical systems security: Limitations, issues and future trends. *Microprocessors and Microsystems*, *77*, 103201. Retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7340599/>

Zhang, J., Pan, L., Han, Q.-L., Chen, C., Wen, S., & Xiang, Y. (2021). Deep Learning Based Attack Detection for Cyber-Physical System Cybersecurity: A Survey. *IEEE/CAA Journal of Automatica Sinica*, 1–15. <https://doi.org/10.1109/jas.2021.1004261>

**Definitions, Acronyms, Abbreviations**

**Cyber security**: The practice of defending internet-connected systems, including data, software, and hardware, against cyberthreats is known as cybersecurity. Both individuals and businesses use it to guard against illegal access to data centers and other digital systems.

**Web Architecture:** The process of planning, developing, and deploying a computer program that runs on the internet is known as web architecture. These programs are frequently websites that offer helpful information to users, and web developers may create them for a certain business, brand, or purpose.