Towards Developing a Scalable Cyber Risk Assessment and Mitigation Framework

1st Adeel A. Malik
Department of Computer Science
Mercer University
Macon, Georgia, USA
malik_aa@mercer.edu

2nd Deepak K. Tosh
Department of Computer Science
University of Texas at El Paso
El Paso, Texas, USA
dktosh@utep.edu

***Abstract*—The increasing dependence on digital technology and the internet has made cybersecurity a critical issue for organizations, with cyber-attacks becoming more frequent and sophisticated. In this context, cyber risk evaluation and mitigation have become essential components of modern cyber infrastructures to ensure the security and resilience of digital assets and services in the face of ever-evolving cyber threats. This paper aims to emphasize the significance of the Cyber-threats and Vulnerability Information Analysis to proactively understand the cyber risks and abnormalities in real-time and provide appropriate mitigation strategies. Our work incorporates an inferencing layer to our AI-engine focusing on cyber risk assessment and mitigation. This inferencing layer prioritizes significant risks and presents a mitigation plan to address them. We discuss the key steps and processes implemented as part of the cyber risk and mitigation (CRAM) framework including use of machine learning algorithms for risk assessment and mitigation. Furthermore, we evaluate and compare the effectiveness of the mitigation plan using strategies provided by the MITRE Corporation, a trusted source in cybersecurity. Overall, this paper highlights the importance of incorporating a real-time risk assessment and mitigation system in organizations' cybersecurity infrastructure. Our proposed framework provides a practical and efficient solution to identify and address potential cyber threats, minimizing the risk of data breaches and financial loss.**

***Index Terms*—Cyber-threats and Vulnerability Information Analyzer (CyVIA), Cyber Risk Assessment and Mitigation (CRAM), Infrastructural Security Evaluation, Vulnerability Classification, NVD, MITRE.**

I. Introduction

Cyber risk assessment and mitigation (CRAM) is the process of reducing the potential for adverse outcomes, where the risk is analyzed and strategies to reduce the risk are determined. This process helps organizations to reduce financial losses, improve operational efficiency, and promote a safe and secure environment. However, the degree of uncertainty, changing conditions, human bias, limited resources, and complexity in our cyber infrastructures make the risk evaluation process more challenging [1]–[3]. According to PWC, many companies left security behind while rushing towards pandemic-inspired changes without evaluating or mitigating risks associated with remote work, digitization, or cloud adoption [4], providing exploitation opportunities to adversaries. On the other hand, employees behind the keyboard are also unwitting participants responsible for cyber attacks; 82% of breaches in 2022 involved a human element [5]. The CrowdStrike Falcon OverWatch team estimates that an attacker needs just 1 hour and 38 minutes on average to move from a compromised host to the next host within the target infrastructure [6], eventually compromising the entire network. This exposure led to a rapid increase in cybercrimes, predicted to cost the world USD 7 trillion in the year 2022 [7].

Cyber attacks are not limited to external adversaries, people working within the organizations are also a considerable threat [8], [9]. 73% of the attacks are carried out by external actors, 18% by internal, and 39% from partner facilities [5]. Understanding how to perform security risk assessment is the first step in securing cyber infrastructures [10], [11]. Many risk assessment frameworks [12]–[16] are proposed over time. Additionally, several risk management standards such as ISO 31000, NIST framework, IEC/ISO 31010, etc., also provide the basic principles, guidelines, and techniques to manage risk [17], [18]. However, the intricate nature of contemporary cyber infrastructures, which results from the diversity of devices employed within the network, presents a significant challenge for cybersecurity procedures [19]. Moreover, the degree of complexity varies among organizations, depending on how the network is segmented. As such, understanding the interplay between network segmentation and cybersecurity complexity is crucial for developing effective security measures in modern cyber infrastructures [20].

Numerous vulnerability scanning tools have been developed and evaluated in recent years [21]–[24] to evaluate risk. However, despite their usefulness in detecting and addressing specific vulnerabilities, these tools typically do not provide a comprehensive assessment of the overall risk profile of the computing infrastructure. Furthermore, many of these tools are proprietary and primarily designed for commercial use, which limits the customization flexibility for specific security needs. As a result, there is a need for more sophisticated and customizable tools that can provide a comprehensive and tailored approach to cybersecurity risk assessment in modern computing infrastructures. and demands a more generic and adaptive framework [20]. Furthermore, traditional risk assessment approaches such as qualitative methods, quantitative

‡ This work is supported by National Science Foundation (NSF) Award #2239609.

methods, hybrid models, and cyber-risk insurance models have been proposed and evaluated in the literature, but they often lack concrete implementation examples to guide their practical application [25]. While these theoretical models provide a framework for understanding and managing cybersecurity risks, their effectiveness in real-world scenarios depends heavily on their customization and adaptation to the specific characteristics of the target computing infrastructure [26].

Given the unique challenges and opportunities posed by modern cyber infrastructures, there is an urgent need for practical and context-aware risk assessment approaches that can effectively identify and manage cybersecurity risks. To this end, it is essential to establish a comprehensive risk assessment and mitigation system that can detect and respond to cyber attacks in a timely and proactive manner, thereby reducing their impact and preventing their recurrence. Our proposed CRAM framework is based on a quantitative risk assessment model [11], that encompasses not only cyber risk analytics [20], but also related mitigation strategies, allowing the organizations to achieve their security goals and objectives. This works is focused on providing more meaningful insights to cyber defenders about the applicable risks. We evaluate the proposed framework on a diverse network segment composed of popular operating systems and commonly used applications, and compare the results with information obtained from MITRE.

The following sections discuss: Section II highlights the related works, Section III discusses the proposed architecture of our framework, Section IV evaluates the proposed architecture and discusses the findings, and Section V concludes the paper and provide future directions.

II. Related Works

Risk assessment and mitigation are crucial processes for identifying potential risks and vulnerabilities in an organization's systems and processes. As noted by Malik et al. [20], this can be a time-consuming process that involves careful analysis to ensure that potential threats are identified and addressed appropriately. In a vulnerability management survey, Kritikos et al. [21] emphasize the importance of addressing vulnerabilities during the software development lifecycle. They provide an analysis of various vulnerability assessment tools, which can help organizations to better support the process. Similarly, Chalvatzis and Katos [22] focus on evaluating three commonly used vulnerability scanners, Nessus, OpenVAS, and NMAP, to determine which tool performs better in terms of risk evaluation. In another study, Mburano et al. [23] evaluate vulnerability scanners against benchmarks provided by the Open Web Application Security Project (OWASP) and Web Application Vulnerability Security Evaluation Project (WAVSEP). Furthermore, El-Alfy et al. [24] evaluate the performance of two popular vulnerability assessment tools, Nessus and Burp Suite, against SCADA devices. Their evaluation considers factors such as accuracy, scalability, and the results produced by each tool. Overall, these studies demonstrate the importance of vulnerability scanners in the process of evaluating risk and the value of using effective tools and methods to ensure the security and resilience of an organization's digital assets.

Many qualitative, quantitative, and hybrid models are proposed in addition to vulnerability scanning tools [27]–[32]. Zambon et al. [28] utilize a Qualitative Time Dependency (QualTD) model combined with standard risk assessment methods to assess risks to the authentication and authorization system of a multinational company. Ayyub et al. [33] propose a quantitative risk assessment model that relies heavily on subject matter experts and historical data to determine risk. Aksu et al. [34] introduce a CVSS v3-based risk assessment methodology that is asset and vulnerability centric, but only limited to traditional computer networks. Jajodia et al. [35] present a modeling and visualization tool that maps the entire network and potential cyber threats to improve overall security posture. Malik et al. [36] propose a quantitative risk assessment framework that evaluates infrastructural risk based on data obtained from online vulnerability sources and the computing environment. Allouch et al. [37] use both qualitative and quantitative risk analysis for the safety of unmanned aerial vehicles based on international safety standards ISO 12100/13849 and a probabilistic model-based risk analysis method, i.e., Bayesian Networks (BN), the Bayesian Networks (BN) takes uncertainties into account and dynamically adjusts recommendations and constraints [38]. The authors find that the information fed from the qualitative method to the quantitative model produced better results and visualizations.

More recently, many studies propose cyber risk insurance models and CRAM frameworks [25]. Shackelford et al. [39] suggest improving cybersecurity by implementing appropriate security controls and evaluating insurance coverage by analyzing the cost-benefit of their cyber risk exposure. Marotta et al. [40] provide an overview of cyber insurance from both industry and academic viewpoints, and offer a potential course of action. The authors conclude that primary challenges in the cyber insurance process include the dynamic nature of systems, difficulty in identifying existing countermeasures, uncertain impact, extensive interdependence, and added liability. Eling et al. [41] examine the obstacles associated with cyber risk insurance, such as insufficient data, inadequate modeling approaches, the potential for changes in risk, and the unpredictable accumulation of risks. The authors propose various strategies to overcome these challenges. In [25], the authors propose a four-stage CRAM framework that employs quantitative methods to estimate the probability of a cyber attack, predict the expected loss using generalised linear models, and develop mitigation strategies using insurance. They also emphasize the importance of implementing business continuity and disaster recovery processes. Meanwhile, in [42], the authors present a risk assessment and mitigation framework that offers an estimation and prediction of future vulnerability growth, thereby aiding IT managers in planning for software procurement.

The identification of potential risks and vulnerabilities in modern cyber infrastructures through risk assessment and mitigation processes is critical. Various studies have highlighted the significance of utilizing vulnerability scanners to evaluate risk and ensure the security and resilience of an organization's digital assets. Qualitative, quantitative, and hybrid models have been proposed to assess cyber risks, while cyber risk insurance models and CRAM frameworks have been suggested as potential strategies for addressing such risks. However, due to the complicated nature of modern cyber infrastructures, these approaches are failing to provide continuous risk assessment. Moreover, they are either not publicly available, for proprietary use, or have not been directly implemented on an industrial case study. Therefore, there is a pressing need for a publicly available framework that can adapt and provide insights into new or unseen risks to keep cyber defenders informed of recent cybersecurity trends. We aim to provide such a framework that can be adapted in any industrial setting, offering not only continuous risk assessment but also guidance on how to mitigate identified risks.

III. System Architecture

In this section, we present a formal overview of the core components and objectives of the Cyber-threats and Vulnerability Information Analyzer's (CyVIA) inferencing engine, i.e., a core component of the AI-based prediction engine. We delve into the integrated components and their interactions within the overarching framework. Comprehensive information about the CyVIA, its functionality, and other components as illustrated in Figure 2, can be found in [20]. The AI-based prediction engine is a crucial component within the CyVIA framework, serving two primary objectives. Firstly, it expedites the vulnerability analysis process for cyber defenders by furnishing relevant attack types corresponding to the assessed infrastructure. Secondly, it prioritizes risks based on their significance and furnishes a comprehensive mitigation plan for addressing them. In this work, our primary focus is the inferencing engine, as depicted in Figure 1, we discuss each component of the inferencing engine in the subsequent subsections.

A. CyVIA Vulnerability Classifier (VC)

In order to facilitate human judgment of risk, CyVIA employs a vulnerability classifier that has been trained on vulnerability data spanning two decades. Given the size and dimensionality of the vulnerability data, a Linear Support Vector Classifier (Linear SVC) model has been trained and tuned for classification purposes. To further enhance computational efficiency, Principal Component Analysis (PCA) has been applied to the Linear SVC model to reduce noise and features in the dataset. This approach focuses the classifier on the most important features, leading to better accuracy and efficiency. The trained classifier is capable of predicting attack types associated with text-based vulnerability descriptions, thereby streamlining the analysis process. The CyVIA API utilizes the vulnerability classifier to generate a list of applicable attack types for a given computing infrastructure, providing a comprehensive assessment of associated risks.

B. CyVIA Context-Aware Summary Generator

A key component of CyVIA that significantly contributes to its capacity to offer precise and practical guidance to cyber defenders concerning identified risks is the ability to extract specific feedback and actions from a diverse range of mitigation strategies present in its knowledgebase. This component plays a pivotal role in enabling efficient and timely risk mitigation and management. Through the extraction of targeted feedback and actions, CyVIA can provide cyber defenders with guidance that is aligned with the specific context of the identified risk. This level of specificity enhances the effectiveness of the guidance provided, as it enables cyber defenders to address the risk in a manner that is tailored to the organization's unique security needs and priorities. Moreover, the ability to extract targeted feedback and actions allows for streamlined decision-making and more efficient allocation of resources, as cyber defenders can focus their efforts on the most critical risks. The summary generator's methodology involves the utilization of an abstractive text summarization approach employing a pre-trained natural language processing (NLP) pipeline. The approach involves the extraction of the most salient sentences from the given text based on their respective rankings.

C. CyVIA Knowledgebase

The CyVIA knowledgebase is a NoSQL database that is organized in a document-oriented manner. Its primary purpose is to serve as a repository for security controls, policies, procedures, reported vulnerabilities, network nodes, and other relevant data. Information in the knowledgebase is stored based on identified relationships between vulnerabilities, weakness types, network nodes, operating systems, applications, and other relevant factors. The goal of the knowledgebase is to make information available to cyber defenders in the most useful form possible, allowing them to quickly identify threats and understand how to mitigate them effectively.

D. CyVIA API

The CyVIA API is a fundamental component of the CyVIA AI-based prediction engine, serving as the core of the system's risk analytics and mitigation capabilities. The API is developed using the Flask REST API framework and is primarily responsible for managing communication and interaction between all internal and external components in a timely and sequential manner. When a request for analysis is made for a specific network or node, the CyVIA API initiates the necessary functions to collect and process the relevant information from all other components. During this process, the API interacts with external sources, including the MITRE repository and API, to gather the pertinent data [43]. The API provides a JSON-formatted response that contains details about the requested host/node or the entire infrastructure, such as the severity of discovered vulnerabilities, top 10 vulnerabilities and weakness types, identified attack types, and affected products. In addition, the mitigation plan includes information related to targets, potential impacts, and preventive measures.

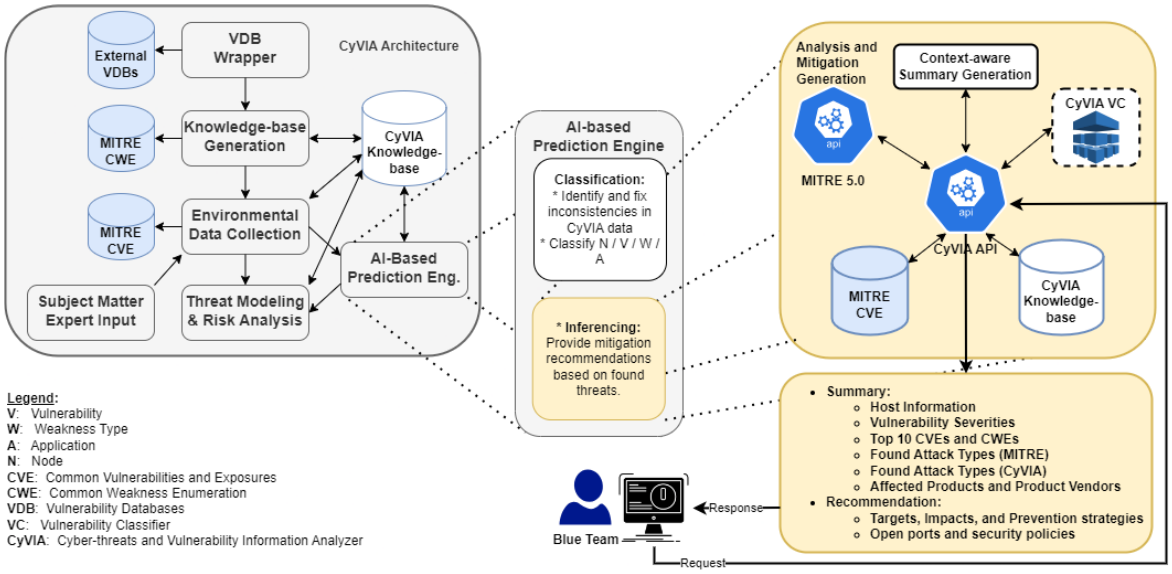
CyVIA Architecture
External VDBs
VDB Wrapper
MITRE CWE
Knowledge-base Generation
CyVIA Knowledge-base
MITRE CVE
Environmental Data Collection
AI-Based Prediction Eng.
Subject Matter Expert Input
Threat Modeling & Risk Analysis
AI-based Prediction Engine
Classification:
* Identify and fix inconsistencies in CyVIA data
* Classify N / V / W / A
* Inferencing:
Provide mitigation recommendations based on found threats.
Analysis and Mitigation Generation
Context-aware Summary Generation
CyVIA VC
api
MITRE 5.0
api
CyVIA API
MITRE CVE
CyVIA Knowledge-base
Summary:
Host Information
Vulnerability Severities
Top 10 CVEs and CWEs
Found Attack Types (MITRE)
Found Attack Types (CyVIA)
Affected Products and Product Vendors
Recommendation:
Targets, Impacts, and Prevention strategies
Open ports and security policies
Response
Request
Blue Team
Legend:
V: Vulnerability
W: Weakness Type
A: Application
N: Node
CVE: Common Vulnerabilities and Exposures
CWE: Common Weakness Enumeration
VDB: Vulnerability Databases
VC: Vulnerability Classifier
CyVIA: Cyber-threats and Vulnerability Information Analyzer

Fig. 1. CyVIA AI-based Inferencing Engine

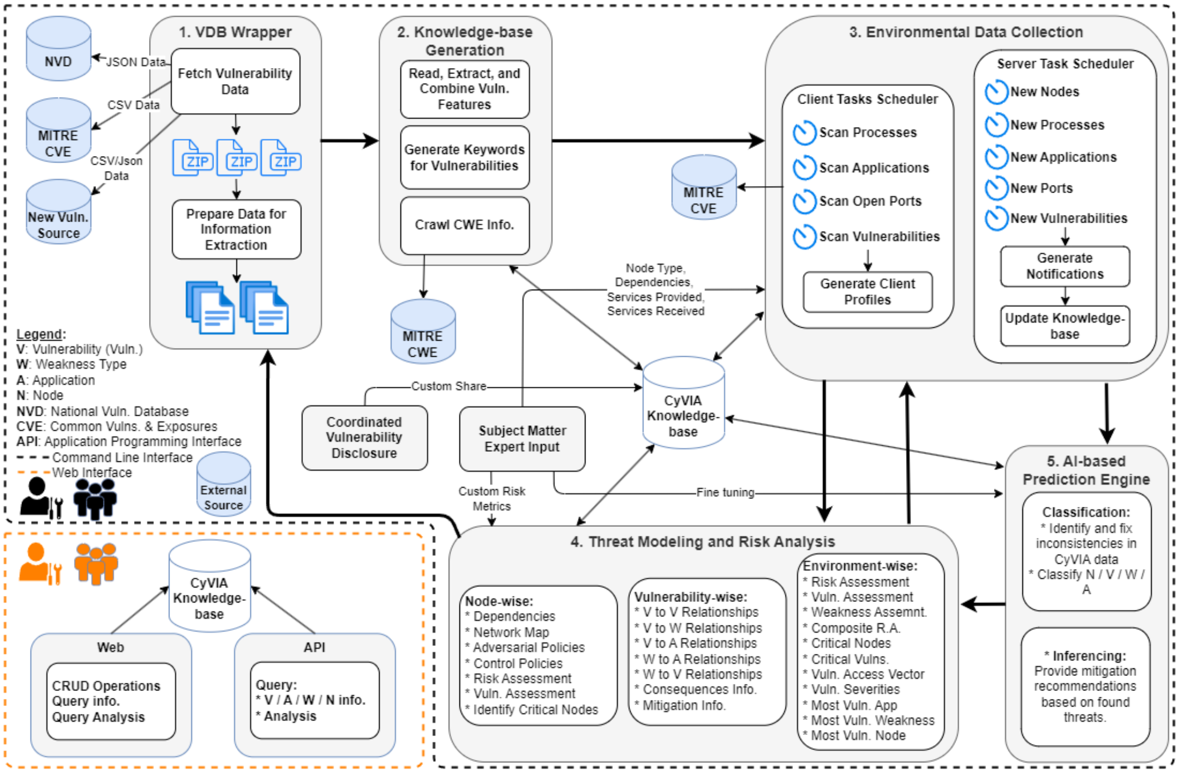

Fig. 2. CyVIA Architecture

Moreover, the API has additional embedded functionality, including the ability to check the status of the knowledge base, find vulnerabilities for specific OS or products, etc.

E. MITRE API and CVE Repository

The external components referenced, including MITRE's Common Vulnerabilities and Exposures (CVE) system, MITRE's API for the CVE system, and MITRE's Common Attack Pattern Enumeration and Classification (CAPEC) system, are all maintained by MITRE. The CyVIA API interacts with these external components as needed to obtain relevant information [43]–[45].

In the next section, we provide an evaluation of how risk analytics generated by CyVIA can facilitate the comprehension and expedient mitigation of risk for any cyber infrastructure by cyber defenders.

IV. Evaluation

This section presents the analytical summary generated by CyVIA's inferencing engine and its potential to provide valuable insights to cyber defenders. We provide examples of how this summary can enhance a cyber defender's understanding of the evaluated cyber infrastructure. We first compare the text descriptions, attack types, and mitigation techniques provided by MITRE and CyVIA in the following subsection. Later, we provide additional insights that CyVIA provides to increase cyber situational awareness for the defenders. By examining these aspects, we can highlight the strengths and limitations of each source of information and provide insights into the effectiveness of their respective approaches.

A. Comparing MITRE Data with CyVIA's Summarized Information

We present Table I as a means to facilitate the comparison between two sources of information related to various CVEs found within the evaluated cyber infrastructure. Table I displays a subset of information, including vulnerability descriptions that have been shortened by approximately 20-30%, while keeping the most useful features of the text in place. This has been achieved by reducing the length of the texts, as is evident from the lengths of the vulnerability descriptions. In addition, CyVIA attack types, which are derived from a collection of 36 most commonly used types of cyber attacks gathered from MITRE, NVD, and other sources, are also compared to MITRE attack types in Table I. We have observed that CyVIA attack types use more commonly used terminologies by a zero to intermediate level of cyber defenders. Moreover, we have utilized a context-aware summary generator to extract the most relevant actions from the available prevention techniques using CyVIA's inferencing engine. In this regard, we have been able to reduce the length of the text by approximately 55-80%.

B. CyVIA Vulnerability Classifier (VC)

To provide a classification example, let us consider the following vulnerability description of CVE-2022-31177:

Description: Flask-AppBuilder is an application development framework built on top of Flask python framework. In versions prior to 4.1.3 an authenticated Admin user could query other users by their salted and hashed passwords strings. These filters could be made by using partial hashed password strings. The response would not include the hashed passwords, but an attacker could infer partial password hashes and their respective users. This issue has been fixed in version 4.1.3. Users are advised to upgrade. There are no known workarounds for this issue.
MITRE Attack Type: CWE-916 - Use of Password Hash With Insufficient Computational Effort.
CyVIA Attack Type: Sensitive Data Exposure.

Based on our analysis, it can be inferred that MITRE and CyVIA utilize distinct terminologies for identifying attack types. Specifically, our analysis indicates that MITRE relies on more technical terminologies compared to CyVIA, which tends to use more commonly understood terms. The findings are presented in Table II [46]. The results underscore the significance of employing appropriate vocabularies that are more effective in aiding an average cyber defender's case.

C. CyVIA Context-Aware Summary Generator

To illustrate, let us consider the following example of a mitigation strategy:

Full Text: Assume all input is malicious. Use an accept known good input validation strategy, i.e., use a list of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, boat may be syntactically valid because it only contains alphanumeric characters, but it is not valid if the input is only expected to contain colors such as red or blue. Do not rely exclusively on looking for malicious or malformed inputs. This is likely to miss at least one undesirable input, especially if the code's environment changes. This can give attackers enough room to bypass the intended validation. However, denylists can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright. (Length: 1124)
Summary: When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. Use an accept known good input validation strategy, i.e., use a list of acceptable inputs that strictly conform to specifications. (Length: 384)

Upon investigating, we see the effectiveness of the CyVIA summary generator in extracting key actions that can be performed to mitigate risk. Our analysis shows that the CyVIA summary generator significantly reduces the amount of text and accurately identifies and can extract the key actions to mitigate risk.

TABLE I
COMPARISONS

CVE	MITRE Description	CyVIA Description	MITRE AttackType	CyVIA Attack Type	MITRE Prevention	CyVIA Prevention
CVE-2022-29216	TensorFlow is an open source platform for machine... (length: 638)	TensorFlow open source platform machine learning... (Length: 450)	CWE-94 Improper Control of Generation of Code ('Code Injection')	Code Injection	Run your code in a jail or similar... (Length: 538)	Run your code... (Length: 133)
CVE-2016-7914	The assoc array insert into terminal node function... (Length: 412)	The assoc array insert into terminal node... (Length: 300)	CWE-125 Out-of-bounds Read	Sensitive Data Exposure	Assume all input is malicious... (Length: 1409)	When performing input validation... (Length: 380)
CVE-2013-1229	TMSSNMP Service in TelePresence... (Length: 216)	TMSSNMP Service TelePresence Manager... (Length: 180)	CWE-20 Improper Input Validation	Denial of Service	For any security checks... (Length: 914)	Understand all the potential ... (Length: 412)
CVE-2022-21668	pipenv is a Python development... (Length: 1143)	pipenv Python development workflow tool... (Length: 892)	CWE-20 Improper Input Validation	Code Injection	Inputs should be decoded and... (Length: 661)	Avoid double-decoding and... (Length: 159)
CVE-2019-9854	LibreOffice has a feature where... (Length: 902)	LibreOffice feature documents... (Length: 706)	CWE-22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	Directory Traversal	Ensure that error messages only... (Length: 1031)	In the context of path traversal... (Length: 328)

TABLE II
MITRE AND CYVIA VULNERABILITY ATTACK TYPES

CVE	MITRE AT	CyVIA AT
CVE-2022-23594	Out-of-bounds Read	Unauthorized Access
CVE-2022-32151	Improper Certificate Validation	Man-in-the-middle
CVE-2022-27237	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	Cross-site Scripting
CVE-2014-9090	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	Denial of Service
CVE-2003-0819	Improper Restriction of Operations within the Bounds of a Memory Buffer	Buffer Overflow
CVE-2019-20916	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	Directory Traversal
CVE-2019-9850	Improper Input Validation	Code Injection
CVE-2022-24761	Inconsistent Interpretation of HTTP Requests ('HTTP Request /Response Smuggling')	Server-side Request Forgery

D. Mitigation Strategies

Table III displays a list of top 10 anticipated attacks that are expected to affect the entire evaluated infrastructure. The list is prioritized from high to low priority, and it requires immediate attention. Here, we present the mitigation plan for the highest-priority risk, which is Code Injection:

1) Terminate the client session after each request.
2) Use only SSL communication.
3) Turn all pages to non-cacheable.
4) Use a web server that employs a strict HTTP parsing procedure, such as Apache [REF-433].
5) Run your code in a jail or similar sandbox environment that enforces strict boundaries between the process and the operating system.
6) With Struts, write all data from form beans with the bean's filter attribute set to true.
7) Refactor your program so that you do not have to dynamically generate code.
8) Be especially careful to validate all input when invoking code that crosses language boundaries, such as from an interpreted language to native code.
9) For any data that will be output to another web page, especially any data that was received from external inputs, use the appropriate encoding on all non-alphanumeric characters.
10) Use an input validation framework such as Struts or the OWASP ESAPI Validation API. Note that using a framework does not automatically address all input validation problems; be mindful of weaknesses that could arise from misusing the framework itself (CWE-1173).
11) ...more...

Through the implementation of the proposed mitigation plan for code injection, developers can incorporate the recommended techniques to improve the quality of their code and prevent any potential exploitation. Furthermore, tailored mitigation plans are available for each type of attack, providing cyber defenders with targeted strategies to mitigate the risk of security breaches.

E. Overall Risk Analytics

Table IV depicts an evaluation of the cyber infrastructure consisting of 15 nodes equipped with commonly used operating systems and applications, outlining the infrastructure-wide risks. The analysis identifies Raspbian node as the most

TABLE III
PREDICTED ATTACK TYPES

	Attacks	Count
1	Code Injection	34,511
2	Sensitive Data Exposure	15,000
3	Directory Traversal	12,882
4	Cross-site Scripting	12,362
5	Unauthorized Access	10,835
6	Buffer Overflow	8,943
7	Denial of Service	8,103
8	Memory Based Attack	5,567
9	Privilege Escalation	4,129
10	Man-in-the-middle	1,708
11	Unknown Attack	1,145
12	Server-side Request Forgery	418
13	Credentials	300
14	Web Session Cookie	41
15	Command and Control	32
16	Host Redirection	29
17	Disabling Security Tools	22
18	Brute Force Attack	6

vulnerable, with the highest number of detected vulnerabilities. These vulnerabilities are categorized into 215 MITRE-defined attack types and 18 CyVIA-defined attack types. Table V presents a prioritized list of attacks against the entire network, ranked from most to least vulnerable, along with a corresponding mitigation plan for each type of attack.

TABLE IV
NODE-WISE VULNERABILITIES, AFFECTED PRODUCTS, MITRE AND CYVIA ATTACK TYPES

Node	CVEs	APs	MITRE	CyVIA
Raspbian	133,298	7,266	215	18
Debian10	48,037	3,978	211	18
Win2016	12,211	663	113	13
Win2012	10,102	663	115	14
Win8	4,360	306	105	14
CentOS	2,967	239	63	13
Win10	2,609	293	98	13
Ubuntu16	2,244	122	93	12
Win7	1,731	162	90	12
Fedora33	1,482	170	44	12
Win11	1,030	151	89	12
Ubuntu20	490	64	64	12
Ubuntu18	259	41	61	12
openSUSE15	103	7	18	10
Router1	27	6	10	6

TABLE V
CYVIA INFRASTRUCTURE-BASED TOP 10 MOST VULNERABLE PRODUCTS

S#	Product	CVEs	CWEs
1	Microsoft MPI ...	6,377	97
2	jackd 5+nmu1	3,070	87
3	chromium 90.0.4430...	1,468	59
4	Windows 8.1 Enterprise	1,107	62
5	Windows Server 2012 R2	949	42
6	ssh 1:7.9p1-10	748	73
7	SQL Server 2017	640	37
8	zip 3.0-11+b1	584	54
9	SQL Server 2017	516	14
10	SQL Server 2017	516	14

Additionally, these attack types can be analyzed in detail, allowing cyber defenders to focus on individual vulnerabilities. For each vulnerability, relevant information such as affected products, versions, prevention stages, and strategies are available, enabling defenders to take appropriate measures as part of the process.

```
CVE: CVE-2022-29216
MITRE Attack Type: CWE-94 * Improper Control of
Generation of Code ('Code Injection')
CyVIA Attack Type: Code Injection
Affected Product: tensorflow
Affected Product Version(s): <2.6.4, >=2.7.0rc0,
<2.7.2, >=2.8.0rc0, <2.8.1, >=2.9.0rc0, < 2.9.0
Target(T): Access Control, Non-Repudiation,
Integrity.
Prevent(P) at Stage: Architecture and Design,
Implementation, Operation, Testing.
P_Strategy(PS): Environment Hardening, Input
Validation, Compilation or Build Hardening.
```

A typical risk assessment framework usually relies on various tools or frameworks to collect data, followed by cyber defenders assessing the outcomes to gauge risk severity and actionability. Yet, with CyVIA, the entire process, from gathering data to generating analysis, is entirely automated. The inferencing engine detailed in this work notably cuts down CyVIA's workload, allowing cyber defenders to precisely identify risks discovered and ways to address them. Overall, CyVIA offers ongoing risk monitoring and threat-focused analytics that capture evolving network configurations without being limited by time or space constraints.

V. CONCLUSION AND FUTURE WORK

Risk mitigation in cyber infrastructures is imperative due to the escalating frequency and complexity of cyberattacks, which can have severe consequences for organizations, including damage to reputation, operations, finances, and even human lives. To assist in this effort, we present an AI-based prediction engine to identify and infer detected risks within a given cyber infrastructure. The engine's primary responsibility is to provide cyber defenders with information on the risks' severity and mitigation strategies. Additionally, with the aid of the CyVIA API, defenders can engage with the engine to obtain additional insights on the risks. Going forward, we aim to make the API publicly accessible to enable individuals to interact with and learn more about the latest trends in cybersecurity. We also plan to utilize this framework as a coordinated vulnerability disclosure process through a website that will allow external cyber defenders to interact with CyVIA.

REFERENCES

[1] Y. Ashibani and Q. H. Mahmoud, "Cyber physical systems security: Analysis, challenges and solutions," *Computers & Security*, vol. 68, pp. 81–97, 2017.

[2] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—a survey," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1802–1831, 2017.

[3] R. Alguliyev, Y. Imamverdiyev, and L. Sukhostat, "Cyber-physical systems and their security issues," *Computers in Industry*, vol. 100, pp. 212–223, 2018.

[4] PWC, *Cyber-ready today and for tomorrow*, 2023 (Online; accessed Mar 1, 2023). https://riskproducts.pwc.com/.

[5] Verizon, *2022 Data Breach Investigations Report*, 2023 (Online; accessed Jan 23, 2023). https://www.verizon.com/business/resources/reports/dbir/.

[6] Crowdstrike, *2022 Global Threat Report*, 2023 (Online; accessed Mar 1, 2023). https://www.crowdstrike.com/.

[7] C. Ventures, *Boardroom Cybersecurity 2022 Report*, 2023 (Online; accessed Jan 23, 2023). https://cybersecurityventures.com/.

[8] B. Bulgurcu, H. Cavusoglu, and I. Benbasat, "Information security policy compliance: an empirical study of rationality-based beliefs and information security awareness," *MIS quarterly*, pp. 523–548, 2010.

[9] R. Moore, *Cybercrime: Investigating high-technology computer crime*. Routledge, 2014.

[10] D. Landoll, *The security risk assessment handbook: A complete guide for performing security risk assessments*. CRC Press, 2021.

[11] A. A. Malik and D. K. Tosh, "Quantitative risk modeling and analysis for large-scale cyber-physical systems," in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–6, IEEE, 2020.

[12] K. Kandasamy, S. Srinivas, K. Achuthan, and V. P. Rangan, "Iot cyber risk: A holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process," *EURASIP Journal on Information Security*, vol. 2020, no. 1, pp. 1–18, 2020.

[13] K. Tam and K. Jones, "Macra: a model-based framework for maritime cyber-risk assessment," *WMU Journal of Maritime Affairs*, vol. 18, pp. 129–163, 2019.

[14] K. Tam and K. Jones, "Cyber-risk assessment for autonomous ships," in *2018 international conference on cyber security and protection of digital services (cyber security)*, pp. 1–8, IEEE, 2018.

[15] P. Radanliev, D. C. De Roure, R. Nicolescu, M. Huth, R. M. Montalvo, S. Cannady, and P. Burnap, "Future developments in cyber risk assessment for the internet of things," *Computers in industry*, vol. 102, pp. 14–22, 2018.

[16] M. Al Fikri, F. A. Putra, Y. Suryanto, and K. Ramli, "Risk assessment using nist sp 800-30 revision 1 and iso 27005 combination technique in profit-based organization: Case study of zzz information system application in abc agency," *Procedia Computer Science*, vol. 161, pp. 1206–1215, 2019.

[17] I. Iso *et al.*, "Risk management–principles and guidelines," *International Organization for Standardization, Geneva, Switzerland*, 2009.

[18] I. I. IEC, "31010: 2009–risk management–risk assessment techniques," *IEC: Geneva, Switzerland*, 2009.

[19] K. Kim, I. Kim, and J. Lim, "National cyber security enhancement scheme for intelligent surveillance capacity with public iot environment," *The Journal of Supercomputing*, vol. 73, pp. 1140–1151, 2017.

[20] A. A. Malik and D. K. Tosh, "Dynamic risk assessment and analysis framework for large-scale cyber-physical systems," *EAI Endorsed Transactions on Security and Safety*, vol. 8, no. 30, 2022.

[21] K. Kritikos, K. Magoutis, M. Papoutsakis, and S. Ioannidis, "A survey on vulnerability assessment tools and databases for cloud-based web applications," *Array*, vol. 3, p. 100011, 2019.

[22] I. Chalvatzis, D. A. Karras, and R. C. Papademetriou, "Evaluation of security vulnerability scanners for small and medium enterprises business networks resilience towards risk assessment," in *2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 52–58, IEEE, 2019.

[23] B. Mburano and W. Si, "Evaluation of web vulnerability scanners based on owasp benchmark," in *2018 26th International Conference on Systems Engineering (ICSEng)*, pp. 1–6, IEEE, 2018.

[24] M. El, E. McMahon, S. Samtani, M. Patton, and H. Chen, "Benchmarking vulnerability scanners: An experiment on scada devices and scientific instruments," in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 83–88, IEEE, 2017.

[25] A. Mukhopadhyay, S. Chatterjee, K. K. Bagchi, P. J. Kirs, and G. K. Shukla, "Cyber risk assessment and mitigation (cram) framework using logit and probit models for cyber insurance," *Information Systems Frontiers*, vol. 21, pp. 997–1018, 2019.

[26] X. Lyu, Y. Ding, and S.-H. Yang, "Safety and security risk assessment in cyber-physical systems," *IET Cyber-Physical Systems: Theory & Applications*, vol. 4, no. 3, pp. 221–232, 2019.

[27] M. Wooldridge, "Qualitative risk assessment," *Microbial risk analysis of foods*, pp. 1–28, 2007.

[28] E. Zambon, S. Etalle, R. J. Wieringa, and P. Hartel, "Model-based qualitative risk assessment for availability of it infrastructures," *Software & Systems Modeling*, vol. 10, pp. 553–580, 2011.

[29] N. Sharmin and C. Kiekintveld, "Enhancing iot device security: Predicting and analyzing reconnaissance attacks using flags and time-based attributes," in *2023 10th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pp. 23–30, IEEE, 2023.

[30] N. Sharmin, "Bayesian models for targeted cyber deception strategies (student abstract)," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, pp. 16324–16325, 2023.

[31] N. Sharmin, S. Roy, A. Laszka, J. Acosta, and C. Kiekintveld, "Bayesian models for node-based inference techniques," in *2023 IEEE International Systems Conference (SysCon)*, pp. 1–8, 2023.

[32] N. Sharmin, J. Acosta, and C. Kiekintveld, "A systematic approach for temporal traffic selection across various applications," in *2023 32nd International Conference on Computer Communications and Networks (ICCCN)*, pp. 1–7, IEEE, 2023.

[33] B. M. Ayyub, W. L. McGill, and M. Kaminskiy, "Critical asset and portfolio risk analysis: An all-hazards framework," *Risk Analysis: An International Journal*, vol. 27, no. 4, pp. 789–801, 2007.

[34] M. U. Aksu, M. H. Dilek, E. İ. Tatlı, K. Bicakci, H. I. Dirik, M. U. Demirezen, and T. Aykır, "A quantitative cvss-based cyber security risk assessment methodology for it systems," in *2017 International Carnahan Conference on Security Technology (ICCST)*, pp. 1–8, IEEE, 2017.

[35] S. Jajodia, S. Noel, P. Kalapa, M. Albanese, and J. Williams, "Cauldron mission-centric cyber situational awareness with defense in depth," in *2011-MILCOM 2011 Military Communications Conference*, pp. 1339–1344, IEEE, 2011.

[36] A. A. Malik and D. K. Tosh, "Robust cyber-threat and vulnerability information analyzer for dynamic risk assessment," in *2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, pp. 168–173, IEEE, 2021.

[37] A. Allouch, A. Koubaa, M. Khalgui, and T. Abbes, "Qualitative and quantitative risk analysis and safety assessment of unmanned aerial vehicles missions over the internet," *IEEE Access*, vol. 7, pp. 53392–53410, 2019.

[38] N. Sharmin and C. Kiekintveld, "Optimizing crop recommendations for sustainable agriculture: Leveraging bayesian networks in a smart crop recommendation system," in *2023 20th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 20–26, IEEE, 2023.

[39] S. J. Shackelford, "Should your firm invest in cyber risk insurance?," *Business Horizons*, vol. 55, no. 4, pp. 349–356, 2012.

[40] A. Marotta, F. Martinelli, S. Nanni, A. Orlando, and A. Yautsiukhin, "Cyber-insurance survey," *Computer Science Review*, vol. 24, pp. 35–61, 2017.

[41] M. Eling and W. Schnell, "What do we know about cyber risk and cyber risk insurance?," *The Journal of Risk Finance*, vol. 17, no. 5, pp. 474–491, 2016.

[42] B. Biswas and A. Mukhopadhyay, "G-ram framework for software risk assessment and mitigation strategies in organisations," *Journal of Enterprise Information Management*, vol. 31, no. 2, pp. 276–299, 2018.

[43] MITRE, *MITRE API*, 2023 (Online; accessed Mar 1, 2023). https://cveawg.mitre.org/api/cve/.

[44] MITRE, *MITRE CVEs*, 2023 (Online; accessed Mar 1, 2023). https://cve.mitre.org/.

[45] MITRE, *Common Attack Pattern Enumeration and Classification*, 2023 (Online; accessed Mar 1, 2023). https://capec.mitre.org/.

[46] A. A. Malik, *Continuous Risk Assessment for Large-Scale Cyber Systems*. PhD thesis, The University of Texas at El Paso, 2023.