A Review on API Security Risk and Vulnerability Assessment

1st Kiran Dangol
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
S20223340@sistc.nsw.edu.au

2nd Sridharan Kothandapani
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
S20222222@sistc.nsw.edu.au

3rd Rohit Mahat
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
S20230316@sistc.nsw.edu.au

Shweta
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
S20242206@sistc.edu.au

5th Arman Roohi
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
armanr@sistc.edu.au

6th Saman Shojae Chaeikar
Department of Cybersecurity
Sydney International School of Technology and Commerce
Sydney, Australia
0000-0002-2958-6901

2024 IEEE Consumer Life Tech (ICLT) | 979-8-3315-1933-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICLT63507.2024.11038691

***Abstract*— Numerous previous security breaches have highlighted the importance of strong cybersecurity practices, particularly in the areas of vulnerability assessment and API security. Incidents such as the 2017 Equifax data breach, which exposed personal information about over 147 million people, highlight the crucial importance of proactive security measures in preventing significant financial losses and reputational harm for businesses. APIs, which are critical for modern technology, enable effective communication between software programs and account for more than 83% of internet traffic. However, its extensive implementation poses serious security weaknesses. This research investigates the many forms of API vulnerabilities and related risks across protocols such as REST, SOAP, and HTTP. It also examines previous studies on API security and vulnerability assessment, emphasizing the likelihood and severity of certain attacks and providing mitigating measures. The paper recommends ways to improve API security, with a focus on blockchain technology and AI/machine learning, to safeguard systems, data, and users.**

Keywords— API security, vulnerability assessment, application security

I. Introduction

The need for strong cybersecurity protocols has been highlighted by earlier security breaches, especially in the domain of vulnerability assessment and API security. Numerous well-publicized events have shown the devastating effects of poor security procedures, including the 2017 Equifax data breach that exposed the private information of over 147 million people [1]. Proactive security measures are desperately needed since these breaches not only cause large financial losses for companies but also damage reputations and undermine consumer confidence.

An Application Programming Interface (API) is a set of protocols that enables different software applications to communicate and exchange data efficiently. APIs have become fundamental to modern technology, accounting for over 83% of internet traffic, as they facilitate seamless integration, automation of tasks, and enhanced data sharing across various platforms [2]. For instance, financial institutions use APIs to connect internal applications with business partners and customer-facing services, thereby streamlining operations and improving user experiences.

This report provides the detail information about the types of the types of the vulnerabilities and associate threat of APIs either that is Representational State Transfer (REST), Simple Object Access Protocol (SOAP), or Hypertext Transfer Protocol (HTTP). This report also provides the overview of the previous work done by different authors in the field of API security and vulnerability assessment in the literature review section of the report. This paper provides the likeliness of the threat and corresponding consequences. More importantly this report includes various recommendations including blockchain mitigation to the vulnerabilities and also the use of AI and machine learning in enhancing the security of the APIs in case of transferring file securely.

II. Literature Review

This report provides a comprehensive study of some of the recent research paper and articles which is being used as the main source of the reference to write this paper. The main summarized articles are listed below.

API Vulnerabilities in cloud Computing Platforms

Ibrahim et al. [3] discuss the securities and vulnerabilities of APIs in cloud Computing. The authors propose methods to detect the attacks by using the vulnerabilities in the Cloud API. According to the authors, cloud computing offers the benefits like elasticity and scalability to the wide adoptions, however the authors also addresses that it attracts the security via APIs specially REST and SOAP protocol which seems to be most popular protocol used by cloud APIs. The paper highlights that REST and SOAP have their own kind of characteristics and Vulnerabilities. API authentication Service attack which can expose sensitive information and the API Exhaustion attack which is DoS attack are the attacks described in this paper. The authors conducted the experiment to detect the attacks where authors simulate those attacks by Using the AD3 algorithm. The experiment concluded that sensitive information could expose data in plain text and using the tool called Siege could generate many API requests which can exploit the vulnerability of cloud API. The main strength of this paper is that the author has focused the cloud computing and provided the information that how the APIs can be vulnerable in leaking sensitive credentials. And provided comprehensive information regarding the API attacks and detection using AD3 algorithm. The weakness of this paper is that it has not proposed any mitigation strategies, and the dataset used in the experiment is limited dataset which

might not cover a wide range of the attack scenario. Overall, the paper focuses on securing the Cloud APIs demonstrating different attacks and detection methods.

Mobile Application Web API Reconnaissance

Mendoza and G. Guofei [4] explain the security inconsistencies that appear because of the input validation that occurs between mobile applications and the corresponding web API services. The authors introduce API hijacking, which is attacks that occur in the client and server side because of the validation. According to the author of this paper, attackers can bypass these inconsistencies and security checks that leads to unauthorized access and data breaches. The author also explains about the WAR Droid system which employs Blackbox testing system to uncover inconsistencies in the variation logic, which can normally lead to security vulnerabilities. The system was tested on 10,000 popular free applications from google detecting logic problematic APIs in over 4000 apps. The paper contributed to static analysis and dynamic testing of the web APIs in identifying the validation inconsistencies. The paper's core strength is providing insight into the security flaws and the mitigation technique of them. However, the paper only focuses on the android systems and does not cover the area of IOS and other operating system applications available in the market. In conclusion, the papers describe the vulnerabilities that arise from the inconsistencies of input validation and the WAR Droid system to uncover the inconsistencies and mitigation technique.

RESI API security: Testing and Analysis

Kajavalta's work [5] addresses security vulnerabilities of REST API implementations that is specifically analyzing M-files Cloud Management API. The author listed all API common vulnerabilities and security testing of those vulnerabilities to analyze and improve the security structure of the M-Files API. The author has covered the fundamental knowledge of HTTP, RESTful API and JSON Web Token protocols. Paper also includes the security testing including White and Black Box Testing, Manual and Automated testing and the list of tools used in the testing which are Postman, Burp suite, OWASP ZAP etc. According to this thesis paper Broken Authentication, Broken Authorization, Excessive Data Exposure, Injection Flaws, Mass Assignment, Security Misconfiguration, Insufficient Logging and Monitoring, Improper Asset Management, Lack of Resources and Rate Limiting are the most common API threats and vulnerabilities. Taking about the strength of thesis paper is show the theoretical and practical implication of API security using effective testing and automation but it limits the scope to other kind of the APIs and depended on the limited numbers of tools. In summary, the author provided important contribution in knowing and improving the API security specially for M-files Cloud Management.

Challenges of Security testing for RESTful APIs

Alhrabi and Moulahi in [6] discussed the complexity in performing the security testing for RESTful APIs. According to the paper, the author faces difficulties ensuring safe data transmission, limiting rate to prevent DoS attack and to protect the abuse of API. The types of security testing discussed in the paper include authorization and authentication testing, input validation testing, vulnerability scanning and fuzzing.

The authors contributed to recommending integrated security testing into API development life cycle, conducting regular vulnerability assessments, using proper encryption and implementation of rate limiting. Overall, the paper provides the comprehensive coverage of security challenges of RESTful API and different security methods and recommendations to improve the security practices. However, the author has missed some information regarding the current evolving technologies that can be implemented in API security.

An API for securing Sharing of Electronic Health Records is a Public Blockchain

Javier et al. [7] integrate API of Electronic Health Record (EHR) systems with a blockchain technology to increase the security and accessibility of EHR. EHR and the isckchain Integration are the main key information listed by the authors. According to the authors, EHR stores sensitive medical information including diagnosis, medication and blockchain integration refers to get the already provided security features to mitigate the limitation of EHR. The strength of their work is the integration of blockchain technology with EHR systems to improve security and accessibility. But the implementation of blockchain in EHR API can be complex and paper does not reflect the ethical issues that may appear in the integration of the blockchain in EHR APIs.

III. Discussion

APIs are essential for current cloud computing and mobile apps, but their broad use creates major security concerns and vulnerabilities. Broken authentication and authorization procedures are among the most common vulnerabilities discovered, allowing attackers to obtain unauthorized access to sensitive information. For example, API authentication service assaults might disclose sensitive data if authentication processes are not strong [3]. Furthermore, input validation problems can result in injection attacks and discrepancies between client and server-side validations, allowing attackers to exploit these vulnerabilities [4]. Another major issue is excessive data exposure, in which APIs mistakenly release more information than necessary, making it simpler for attackers to collect sensitive data [5]. Denial of Service (DoS) attacks, such as API exhaustion attacks, increase API security by flooding the API with requests and disrupting service [3].

To address these vulnerabilities, a variety of detection and mitigation strategies are provided. Static and dynamic testing are critical for identifying security problems in APIs, as demonstrated by the WAR Droid system, which uses black box testing to find anomalies in validation logic [4]. Comprehensive security testing methodologies, including white-box and black-box testing, as well as automated tools like Postman, Burp Suite, and OWASP ZAP, are required for full security assessments [5]. Algorithmic detection approaches, like the AD3 algorithm, have also demonstrated potential for simulating and detecting attacks on cloud APIs [3]. Furthermore, integrating blockchain technology with APIs, especially in sensitive applications such as Electronic Health Records (EHR), improves security and accessibility by utilizing blockchain's inherent security features, though this approach adds complexity and ethical considerations [7].

Despite these advancements, limitations still exist highlighting the need for additional research and practical solutions. Several studies are limited by small datasets and breadth, potentially ignoring larger attack scenarios [3,4]. Furthermore, the findings' generalizability is limited by their focus on specific platforms, such as Android, rather than other

operating systems [4]. Some articles also lack complete mitigation solutions, underlining the need for stronger security measures [3,6]. Continuous improvement in security measures is critical, including frequent vulnerability assessments, suitable encryption techniques, and the deployment of rate limitation to protect against data breaches and DoS attacks [6]. As API security threats grow, it is critical to adapt and improve security measures to protect the safety and integrity of API-based systems.

Table I provides the likeliness of the threats and associate vulnerabilities and consequences of the most occurred threat and vulnerabilities according to OSWAP top 10 [8].

TABLE I. LIKELIHOOD OF THREAT AND CONSEQUENCES

Threat	Likelihood	Impact	Consequences
Injection Attacks [8]	High	High	Unauthorized access, data breach, manipulation, loss of integrity
API Hijacking [8]	High	High	Exploitation of logic inconsistencies and unauthorized access
Man in the Middle [8]	Medium	High	Interception, confidentiality breaches, reputational damage
Cross Site Scripting (XSS) [5,8]	Medium	Medium	Data theft, session hijacking
Authentication Bypass [5]	High	High	Unauthorized access to APIs can be obtained by taking advantage of weak authentication procedures
Broken Access control [5,8]	Medium	High	Unauthorized users may be able to access restricted data or capabilities due to inadequate access restrictions
Sensitive Data Exposure [5,8]	Medium	High	Inappropriately exposing sensitive data using APIs might result in data breaches and noncompliance with regulations
Security Misconfiguration [5]	Medium	Medium	Inadequately designed security settings can be taken advantage of to sidestep security controls
Rate limiting Bypass [5]	Medium	Medium	Bypassing rate restricting can prompt Forswearing of Administration assaults or unreasonable asset utilization
Broken Function-Level Authorization [5,8]	Medium	Medium	Unauthorized access to the information and resources, data breach
Mass Assessment [8]	Medium	Medium	Advised treatment of information can permit assailants to change object properties they shouldn't approach
Denial of service (Dos)	Medium	High	Loss availability, disruption, revenue loss
Third-Party API Vulnerabilities	Low	Medium	Security blemishes in outsider APIs can be taken advantage of to think twice about in general security of your application

After the study of articles and outsource about API, risk in the perspective of API protection is the possible occurrence of a threat or a weakness being exploited. Probability plays an important role in decision making to although it is necessary to allocate security measures to risks Probability assessment is essential in prioritization as it involves identification of risk which are most likely to occur and therefore require attention. The likelihood is typically categorized into three levels: through high, medium and low level of communication.

A high likelihood suggests that the threat is highly likely to happen. Reasons for a high likelihood include open space, susceptibility to attacks, and several potential attackers. For instance, injection attacks such as SQL injection is categorized as high likelihood since it has been widely publicized, easily executable with the help of tools, and typically stems from programmer errors. Such threats demand effective and instant measures such as the input validation and use of prepared statements. Moderate impact means that a threat is moderately likely to be leveraged. Low-likelihood threats are less likely to manifest as compared to high-likelihood threats, but they are also risky. For instance, we have rate limiting bypass and broken function level authorization. Although, these might not be high-likelihood threats as much as high-impact threats, these can still be problems that need to be addressed. Measures that can be taken to prevent or control medium-likelihood threats include access control measures, security audits, and operation surveillance. Low likelihood threats are those that have low chances of being attacked. They could have certain conditions attached or are limited in their access. For instance, Cross-Site Request Forgery (CSRF) can be classified as low likelihood if the general use of anti-CSRF tokens is observed. Still, even low probability threats cannot be disregarded because they can cause certain harm. These threats are well managed provided the organization undertakes frequent security audits and complies with the current recommended standards. Likelihood involves the intricacy of the exploit, the expertise needed to accomplish the vulnerability, the tools available for the exploit, vulnerability frequency and its surrounding environment, given the API. This assessment assists in determining the real possible risk and directs effort on preventing hazards that are most likely to occur and their likely impact.

Thus, knowing about the possibility of threats and their classification let organizations prioritize their security processes: the potentially real and dangerous threats are to be countered in the first instance, whereas one should remain wary of the less probable, but still possible, threats.

IV. RECOMMENDATIONS

As the growing trend of the AI and machine learning, AI has the major effect on the emphasizing to manage vulnerability more effectively and enhance the could security more securely [9]. Since AI and the machine learning has been overtaking most of the industry, using in the Cybersecurity in terms of training the vulnerability testing for different kinds of treats can possibly help to mitigate most of the common types of vulnerabilities of APIs that is mentioned in OSWAP Top 10.

To reduce these threats, organizations must have a thorough strategy for API security. It requires the implementation of strong authentication and authorization systems, input validation, rate limitation, and frequent security testing and monitoring. Furthermore, it is essential to use safe coding methods, employ encryption techniques, and consistently monitor API traffic for any abnormalities [10]. Organizations should also contemplate the use of API gateways, which function as a centralized access point for APIs, offering supplementary security measures, such as authentication, rate limitation, and traffic monitoring. API management tools facilitate the administration and protection

of APIs at every stage, including design, implementation, and ongoing maintenance, for organizations.

Organizations should also contemplate the use of API gateways, which function as a centralized access point for APIs, offering supplementary security measures, such as authentication, rate limitation, and traffic monitoring. API management tools facilitate the administration and protection of APIs at every stage, including design, implementation, and ongoing maintenance, for organizations.

The security risks and vulnerabilities linked to APIs are substantial and must not be disregarded. Given the significant role that APIs play in contemporary software systems, it is imperative for organizations to prioritize API security and implement optimal strategies to minimize these risks. Organizations may safeguard their systems, data, and customers from the possible repercussions of API security breaches by deploying strong security measures and being alert to new risks.

The introduction of the blockchain technology along with the AI and Machine learning can also be one of the best solutions to improve the security the of APIs including the cloud as well as mobile and web applications. As mentioned in article [11], introducing the blockchain in the API can mitigate the limitations that current APIs are facing. And in the article [7], blockchain can be the major solution to secure sharing of the EHRs without compromising confidentiality and access control. So blockchain can be used one only in sharing the EHRs but also can be used in different industries where confidentiality and access control are most important factor.

V. Conclusion

APIs are critical to modern cloud computing and mobile apps, but their widespread use poses significant security concerns and vulnerabilities. Common difficulties including faulty authentication and authorization, vulnerabilities in input validation, and excessive data exposure demand strong security measures [5]. Comprehensive static and dynamic testing, algorithmic detection methods, and blockchain technology integration, particularly in sensitive applications like Electronic Health Records (EHR), are all effective solutions [7,12]. These procedures serve to reduce vulnerabilities, secure sensitive data, and assure system integrity.

Secondly, despite developments in API security, limitations remain, highlighting the importance of continued research and practical solutions. Many studies are limited by limited datasets and a specific emphasis, perhaps ignoring larger attacking possibilities [13,14]. The findings' generalizability is further restricted because they focus on systems such as Android, without considering other operating systems. Furthermore, several studies lack thorough mitigation strategies, underlining the importance of constant advancement in security measures such as constant vulnerability assessments, adequate encryption techniques, and rate limitation to prevent data breaches and denial of service attacks.

Lastly, prioritizing security processes depending on the likelihood and severity of attacks is critical. Organizations must develop strong authentication and authorization systems, as well as input validation, rate limitation, and continuous monitoring. The use of AI and machine learning can improve vulnerability management and API security. By implementing these measures, organizations can secure their systems, data, and users against possible security threats. Adopting modern technologies like blockchain and AI, as well as demanding security measures, can assist to preserve user trust and guard against changing API risks.

References

[1] Taherdoost, Hamed, Shamsul Sahibuddin, Meysam Namayandeh, Neda Jalaliyoon, Alaeddin Kalantari, and Saman Shojae Chaeikar. "Smart card adoption model: Social and ethical perspectives." *Science* 3, no. 4 (2012): 1792-1796.

[2] Taherdoost, Hamed, Saman Chaeikar, Mohammadreza Jafari, and Nakisa Shojae Chaei Kar. "Definitions and criteria of CIA security triangle in electronic voting system." *International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol* 1 (2013): 14-24.

[3] F. M. Ibrahim, Z. Kasiran and M. A. M. Arrffin, "API Vulnerabilities In Cloud Computing Platform: Attack And Detection," International Journal of Engineering Trends and Technology, 2020.

[4] A. Mendoza and G. Guofei, "Mobile Application Web API Reconnaissance: Web-to-Mobile Inconsistencies & Vulnerabilities," in IEEE, 2018.

[5] l. Kajavalta, "REST API SECIRITY: TESTING AND ANALYSIS," Faculty of Information Technology and Communication Sciences, 2022.

[6] S. J. Alhrabi and T. Moulahi, "API Security Testing: The Challenges of Security Testing for Restful APIs," International Journal of Innovative Science and Research Technol, vol. Volume 8, no. May 2023, 2023.

[7] M. P. Javier, E. W. Lopez, G. L. Marcelo and K. Y. Solomon, "An API for Secure Sharing of Electronic Health," 2024.

[8] O. A. S. Project., "OWASP," n.d. [Online]. Available: https://owasp.org/www-project-api-security/.

[9] Chaeikar, Saman Shojae, Alireza Jolfaei, and Nazeeruddin Mohammad. "AI-enabled cryptographic key management model for secure communications in the internet of vehicles." *IEEE Transactions on Intelligent Transportation Systems* 24, no. 4 (2022): 4589-4598.

[10] Chaeikar, Shojae. "A prospective study of mobile cloud computing." *International Journal of Advancements in Computing Technology (IJACT)* 5, no. 11 (2013): 198.

[11] Chaeikar, Saman Shojae, Mojtaba Alizadeh, Mohammad Hesam Tadayon, and Alireza Jolfaei. "An intelligent cryptographic key management model for secure communications in distributed industrial intelligent systems." *International Journal of Intelligent Systems* 37, no. 12 (2022): 10158-10171.

[12] Khodadadi, Touraj, Mazdak Zamani, Saman Shojae Chaeikar, Yashar Javadianasl, Marieh Talebkhah, and Mojtaba Alizadeh. "Exploring the Benefits and Drawbacks of Machine Learning in Cybersecurity to Strengthen Cybersecurity Defences." In *2023 IEEE 30th Annual Software Technology Conference (STC)*, pp. 1-1. IEEE, 2023.

[13] Chaeikar, Saman Shojae, Ali Ahmadi, Sasan Karamizadeh, and Nakisa Shoja Chaeikar. "SIKM–a smart cryptographic key management framework." *Open Computer Science* 12, no. 1 (2022): 17-26.

[14] Zeidanloo, Hossein Rouhani, Azizah Bt Abdul Manaf, Rabiah Bt Ahmad, Mazdak Zamani, and Saman Shojae Chaeikar. "A proposed framework for p2p botnet detection." *International Journal of Engineering and Technology* 2, no. 2 (2010): 161.