Journal of Autonomous Intelligence (2024) Volume 7 Issue 5
doi: 10.32629/jai.v7i5.1318

ORIGINAL RESEARCH ARTICLE

E-learning systems application programming interfaces security management

Ahmed Yusuf Mai-inji[1], Longe Olumide Babatope[2], Kingsley Eghonghon Ukhurebor[3,*], Adewale O. Adesina[1], Vivian Nwaocha[1], Idris Ismaila Sinan[1], Udochukwu Chidiebere Nwankwo[1], Emmanuel Lyada[4], Moses Ashawa[5]

[1] *Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), National Open University of Nigeria, Abuja 900001, Nigeria*

[2] *Faculty of Computational Sciences & Informatics, Academic City University, P.O. Box AD 421, Accra, Ghana*

[3] *Department of Physics, Edo State University, Uzairue, P.M.B. 04 Auchi 312001, Edo State, Nigeria*

[4] *Education Department, African Union, P.O. Box 3243, Roosvelt Street W21K19 Addis Ababa, Ethiopia*

[5] *Department of Computer Science, Glasgow Caledonian University, Cowcaddens Road Glasgow G4 0BA Scotland, UK*

*** Corresponding author:** Kingsley Eghonghon Ukhurebor, ukeghonghon@gmail.com

ABSTRACT

The platforms for end users to explore and use were designed with proper effort by the program developers. Programmers take every precaution to create a system with a robust user interface and minimal maintenance requirements. Numerous lines of code are used by software programs to carry out processes and complete tasks that end users specify. Learning institutions use online education to better meet the needs of each student at a lower cost while looking for ways to supply high-quality educational content. This has a significant impact on the development of open and distance learning as well as online education. Additionally, it is expected that the use of the digital education system will keep expanding due to changes in student needs and technological developments in online education. To identify and communicate with application software, electronic communication requires a protocol. Application programming interfaces (APIs), which allow applications to speak with each other and share information, are one of these methods of communication. However, this research is designed to improve the privacy trust model in relation to eLearning platforms for the prevention of personal digital data and to curb the present cyber threat in the distance and open online learning environment. The research will conduct a review of past literature on users' security models and trust privacy in an e-learning environment. It will be established in this research that digital data breaches are imminent and require a proper security solution. The study will provide an overview of the techniques and indicators of privacy breaches and develop a model that integrates trust and privacy in e-learning environments by contextualizing the peculiarities of open and distance studies (for online learners). The research is aimed at improving the existing eLearning security model. Also, a case study or practical example illustrating the application of the proposed API security solutions in real-world e-learning scenarios was conducted based on a survey of institutions as well as a survey of e-learning end users in order to gain a more tangible grasp of the concepts presented in this study.

Keywords: programming interfaces; e-learning environment; security; policies

1. Introduction

Today's online learners expect to have better accessibility to e-learning platform data and services via a wide range of digital tools and platforms[1–3]. Institutions must now provide their assets in a way that is nimble, flexible, secure, and scalable in order to satisfy the expectations of students[4,5]. To support device communications, APIs provide an institution with the appropriate data and services. They make it simple for programs to connect with one another using a simple protocol like HTTP. Applications that communicate with the back-end system are created by developers using APIs. Using an API administration platform, an API must be managed and secured after it has been created. According to AltexSoft, a Data Science and Software

ARTICLE INFO

Received: 8 October 2023
Accepted: 8 December 2023
Available online: 18 June 2024

COPYRIGHT

Copyright © 2024 by author(s).
Journal of Autonomous Intelligence is published by Frontier Scientific Publishing. This work is licensed under the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).
https://creativecommons.org/licenses/by-nc/4.0/

Engineering Consulting Company founded in 2007 in the US, an API is a set of programming codes that enables data transmission between one software product and another. Data exchange can also be part of this product code[6]. The mechanisms that enable two or more application software components to exchange information with one another via predefined protocols are called APIs. For example, the weather bureau's software system contains daily weather data. Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud system[7].

Instructional institutions have been seeking ways to address the needs of their students by delivering high-quality educational materials in the most efficient way possible. This led to practically all tertiary institutions adopting online education as a result. Additionally, it is predicted that the online learning sector will grow dramatically over time due to technological improvements and changing student demands. This extension would not have been possible without APIs. Application communication and resource sharing are made possible by these software design interfaces. They provide capabilities that enable information interchange between two different software applications in online learning systems. APIs are used by programmers to create apps that communicate with the back-end infrastructure. An API administration platform must be used to manage an API once it has been created. In contrast, it should be highlighted that not all of the industry's use of APIs has been beneficial. This is due to the fact that putting APIs into use raises a number of issues, with cyber security taking the lead.

This is made more difficult by the security application programming interfaces' (APIs) poor usability. Many suggestions that are based on more general best practice guidelines for software engineering and API design have been made in an effort to help developers by making cryptography libraries easier to use. They organize knowledge about these suggestions in their article. They found 65 publications, examined them, and provided 883 suggestions. Find seven key techniques to make APIs more usable using theme analysis. The majority of the suggestions focus on assisting API developers with the construction and organization of their code to make it more accessible to and understandable by programmers. They discover that relatively few suggestions for API usability have been experimentally verified and that recommendations pertaining to useable security APIs are even less common[8].

APIs provide features on online platforms that enable communication and information sharing between two or more software programs. This research focuses on managing the API system, which is essentially the sole way for application programs to communicate with one another. Implementing this study will likely

result in fewer data breaches because it is getting harder to secure the whole stack of digital data. The use of APIs as an attack vector by unauthorized users to take over and manipulate institutional application software has emerged in recent years. The implementation of APIs must include a variety of security safeguards, with cybersecurity being the main concern. APIs play an important role in the development of e-learning platforms. Users are given a safe environment for enhanced knowledge acquisition through the adoption of effective API management policies and the monitoring of digital learning systems. The importance of APIs in developing security policies, on the other hand, is frequently overlooked by online institutions. The study will also inform programmers about the value of implementing sound security measures at the very beginning of the development process. This study will inform educators on the value of implementing API security for the defence of e-learning platforms. The management of digital information requires APIs. Adhering to standard usability heuristics is one way to create security APIs that do not have glaring usability issues. All security APIs should go through extensive research and development to improve their usability for online educational platforms. Tutors and participants will benefit the most from the research by having a safe environment for their facilitation and study, respectively. The administration of online studies may also find this information useful when developing policies on data safeguards. The importance of and best practices for API security must be understood by all parties, including users of the eLearning environment, application developers, and decision-makers for online platforms. This would increase the security of the online materials' legitimacy. Hence, the research focuses on managing the API system, which is essentially the sole way for application programs to communicate with one another. The usability of the API in the protection of user personal information is brought to the forefront by the fact that the online educational system is built on software applications of various types. Implementing this study will likely result in fewer data breaches because it is getting harder to secure the whole stack of digital data. The study will also inform programmers about the value of implementing sound security measures at the very beginning of the development process. Tutors and participants will benefit the most from the research by having a safe environment for their facilitation and study, respectively. The administration of online studies may also find this information useful when developing policies on data safeguards.

2. Literature survey

The study discusses similar work in security development, policy, and API deployment. APIs are a type of intermediary application that facilitates efficient and seamless resource sharing between different programs. Web applications are typically assembled as a configuration of heterogeneous web services that interrelate with each other via web APIs[9]. According to Hussain et al.[10], an API is a technique that enables the distribution of services across several domains in a simple, cost-effective, and scalable way. A set of protocols, operations, processes, tools, descriptions, and features known as APIs are used to exchange, create, and build upon current services as well as create novel ones across many domains. Applications in the context of APIs involve programs designed with a particular drive. Interfaces can be thought of as a contract of service between two applications. This agreement describes communication requests and responses between applications. Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud system[7]. Scholars have made a lot of findings in the area of API security[11]. The new technical wave of web applications, also known as APIs, is transforming how businesses operate as well as the trends and plans for collaborative enterprises. A graphic user interface is not required for communication between components of software. Machine-readable interfaces, or APIs, allow software products to communicate data and features. APIs are the interfaces that software developers use the most frequently. Developers have access to particular features that are protected by APIs, giving them a powerful tool with which to create their own software by constructing it from API calls. For developers, security APIs have been and continue to be a huge burden[12]. Learning and comprehending security APIs through practice is a far cry from reality. To be able to create accessible

programming interfaces for security mechanisms with fundamental usability features like learnability and fault tolerance, it is imperative to have a deeper grasp of developers' requirements for security APIs. Without the active participation of the developer community, an API initiative cannot be successful. To create mobile applications or a custom interface between several applications, application developers employ APIs. Consequently, it is essential for developers to be aware of the APIs that are available, as well as their features and usage. For APIs to be used effectively in Apps, developers should have access to a testing environment.

Users' personally identifiable information (PII) must be delivered securely from the entry device to the online server system in the e-learning system network for verification. The PII is encrypted using a variety of different keys as it travels through the network because the online platform cannot realistically expect to securely exchange secret keys with every device. Utilizing a digital learning system application that are highly reliant on APIs, such as the Internet, can connect with one another via network communication protocols.

Tasnim et al.[13] proposed an API security model, which they named "API Recommendation for Security Orchestration, Automation, and Response (APIRO)". They designed a model to alleviate data breaches and deny information availability by utilizing a wide diversity of data augmentation procedures. The model applied the "convolutional neural network (CNN)" for prediction while a specific word was embedded. Through experimentation, they demonstrated the efficiency of APIRO in endorsing APIs for several responsibilities, utilizing three security devices as well as 36 augmentation procedures. Our test results show that APIRO is capable of reaching 91.9% Top-1 accuracy. APIRO performs better in terms of Top-1, Top-2, and Top-3 Accuracy and mean reciprocal rank (MRR) as compared to the contemporary baseline by 26.93%, 23.03%, and 20.87%, respectively.

By easing connections between microservices and clients and acting as a single interface request and permission filtering mechanism, the API gateway ensures safe platform-wide connectivity. By using asymmetric encryption, communication fields are effectively encrypted, lowering the possibility of data leakage during transmission. The suggested solution in their article efficiently addresses the platform's security and reliability issues while reducing the risk of attacks on exposed microservice interfaces. In order to increase the security of the platform, this article describes the unique architecture of an API gateway and an asymmetric encryption technique for a cross-border logistics compliance platform[14].

The service API is used by management to realize flexible configuration of business services and separation of front and back ends. The API gateway authenticates users, forwards requests from users by means of the API gateway between the front-end user and the back-end API service set, and responds to the appropriate API service. As can be observed, the API gateway plays a critical role in the business network and is inherently vulnerable to security risks in traditional designs[15].

3. APIs administration

Today's online users demand to be able to access company data and services via a number of digital tools and channels. Enterprises must open their assets in a secure, scalable, agile, and adaptable way to satisfy customer expectations. APIs are a company's window into its data and services. They make it possible for programs to quickly exchange messages using a simple protocol like HTTP. APIs are used by developers to create apps that communicate with the back-end infrastructure. An API administration platform must be used to administer an API after it has been developed.

Online educational platforms may unleash the unique potential of their assets by publishing APIs to internal, partner, and external developers with the support of an API management platform (MP). Through developer interaction, business insights, analytics, security, and protection, it provides the fundamental features necessary to guarantee a successful API operation. In order to maximize investments in digital

transformation, e-Learning providers can use insights provided by an API MP to speed up outreach across digital channels, encourage more online education adoption, and monetize digital assets.

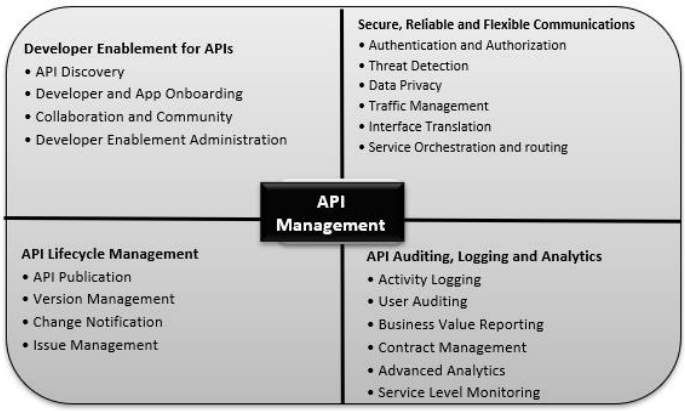

Figure 1. API management capabilities.

You may build, evaluate, and manage APIs using a scalable and secure platform for API administration. The following features should be available from an API MP: "developer enablement for APIs; secure, reliable, and flexible communications; API lifecycle management; API auditing, logging, and analytics." **Figure 1**, as adapted from Brajesh[16], summarizes API management capabilities.

4. API security vulnerabilities

Application vulnerabilities are the easiest route to giving access to unwanted elements. It is important to address these challenges while developing any application. Some API security vulnerabilities include the following:

- Broken object-level authorization: This happens when a user logs in and has the ability to access or change data that the requestor hasn't had access to.
- Broken functional-level authorization occurs when the doctrine of least privilege has not been applied as a result of intricate access regulations.
- Excessive data exposure: It happens when API answers to a request return more data than is required.
- Improper asset management happens when you lack comprehensive documentation and an API repository.
- Lack of resources and rate limiting: When there is no restriction on the quantity and size of requests, it occurs.
- Injection flaws happen when data is incorrectly digested and validated.

These were some of the most prevalent or well-known flaws in API security (**Figure 2**).

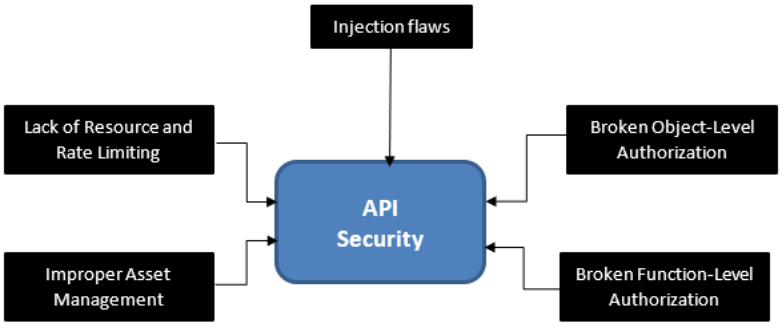

Figure 2. API security vulnerabilities.

5. API security

Access to valuable and protected data and assets is made possible by APIs. In order to safeguard the basic assets from unauthenticated and illegitimate access, API security is of the utmost importance. Since APIs are programmatic in nature and accessible to the general public, they are also vulnerable to a particular form of attack. The technique of protecting APIs from assaults is known as API security. Because they are usually accessible via public networks and can be used by anyone, APIs are frequently well documented or simple to reverse-engineer, according to Advanced API Security: Protecting the digital economy and empowering innovation[17]. There are a variety of devices that can access online educational resources, and they all require communication and data sharing.

Aside from programmers themselves, consumers who employ cryptography tools may wind up using the Security API regularly. For instance, tutors may use the Security API to construct each signature and for identity authentication at a result recording authority that generates student scores. Using API management would stop security breaches in online educational systems in the current environment of digital studies.

6. API security solutions

Users of the eLearning environment, application developers, and policymakers for online platforms all need to be aware of the value of and best practices for API security. This would improve the authenticity of the internet resources' security. The following topics must be thoroughly understood in order to improve e-platforms and are also shown in **Figure 3**.

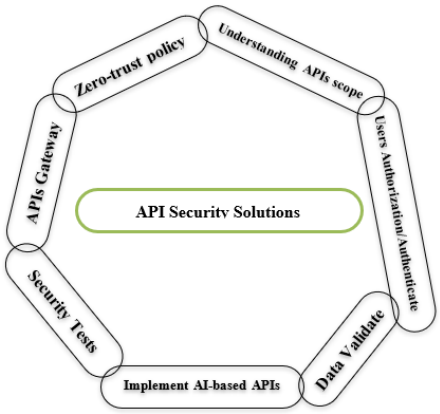

Figure 3. API security solutions.

Users' authorization/authentication: Policies for user authentication become crucial in preventing vulnerabilities connected to device authorization and access. Using a token API in conjunction with user identity authentication can increase the security of standard web application API access and control. Access control can also be utilized depending on the level of access and the user group.

Data validate: Attacks involving code injection or SQL injection are always possible in the absence of a suitable data validation system. A server that has been cleaned and validated will undoubtedly offer good security and reduce the chance of a malicious script being inserted into the source code. Applications like XML or JSON schema validation are capable of thwarting these security risks. The data flow of APIs can be examined to find faults and anomalies using Postman and Chrome Dev Tools.

Implement AI-based API monitoring: Implementations of behavioural AI monitoring analysis can improve API security. This could provide an API regular traffic benchmark, decrease API abuses, and bring consumption patterns under control. e-Learning applications can be fine-tuned to set the threshold to checkmate the traffic flow. Consequently, an AI monitoring-based system can aid in providing a list of gray

zones in applications with potential exploitation possibilities. Applications such as AppDynamics, Prometheus, Sematext Synthetics, and Site24 × 7 Website Monitoring can support API monitoring and risk discovery.

Security tests: Proactive measures are the best way to prevent security breaches. To safeguard APIs, the system must be active. Reactions should be eliminated, and a serious security evaluation or test should be conducted from the start of the application through the end of the process.

APIs gateway: Traffic control is provided by APIs Gateway, which also provides a planned access route to Wright packages. Regulating features allows for controlling things like rate limits, harmful users, and legal logging.

Zero-trust policy: Putting into practice a zero-trust strategy where all users with access, insiders or outsiders, are required to receive authentication, authorization, and ongoing validation for security configuration. By implementing zero-trust policies, organizations can become safe, modern data-driven organizations and find the right answers to problems with ransomware attacks, multi-cloud environments, and securing remote workers. The introduction of a zero-trust policy is supported by many reputable and knowledgeable groups.

Understanding API scope: The breadth of the APIs included both internally developed and commonly used third-party APIs. To comprehend the attack surface or vector, identify potential security risks, and implement the necessary security best practices, a thread model can be constructed. Analyse the API's procedures and security features while paying close attention to the documentation. Threat modelling can be aided by tools like the Microsoft Threat Modelling Tool, VM, Cairis, Threat Dragon, IruisRisk, etc.

7. Institutional survey

IT administrators have the chance to validate the material ahead of the combined release with non-security releases. When there is a special necessity, releases might also be given outside of the monthly schedule. Quality updates prevent operating system (OS) fragmentation by including all previously issued fixes; they are cumulative. When only a portion of the fixes are applied, reliability and vulnerability problems may arise. Two separate kinds of quality upgrades are released on a monthly schedule: "non-security releases; and combined security and non-security releases".

Prior to the combined release with non-security releases, IT administrators can verify the content. Releases may also occur outside of the monthly cycle when there is a unique need for Windows technical documentation for developers and IT pros (learn.microsoft.com). Users are recommended to upgrade their systems once per month to make sure they are using the most recent operating system version and can benefit from recently published updates. The majority of institutions do not properly document their security procedures and policies, which ought to incorporate all pertinent stakeholders, including external partners and software providers.

8. A case study/practical example

The research as stated earlier provided an overview of the techniques and indicators of privacy breaches and develop a model that integrates trust and privacy in e-learning environments by contextualizing the peculiarities of open and distance studies (for online learners), which is aimed at improving the existing e-Learning security model. Also, a case study or practical example illustrating the application of the proposed API security solutions in real-world e-learning scenarios was carried out based on a survey of institutions as well as a survey of e-learning end users in order to gain a more tangible grasp of the concepts presented in this study.

8.1. Survey of institutions

In order to determine the level of security precautions and practices for online platforms while completing studies on various platforms, a Google Form survey questionnaire was developed and distributed to some selected e-Learning administrators for evaluation. Based on the responses received, about 27% of the end users update their system weekly, while 33% update monthly, and only 11% update on a daily basis. According to Microsoft Company, Windows monthly quality updates help you stay productive and protected. They provide your users and IT administrators with the security fixes they need and protect devices so that unpatched vulnerabilities cannot be exploited. Quality updates are cumulative; they include all previously released fixes to guard against fragmentation of the OS. Reliability and vulnerability issues can occur when only a subset of fixes is installed. Quality updates are provided on a monthly schedule as two types of releases:

- Non-security releases.
- Combined security and non-security releases.

Non-security releases provide IT administrators with an opportunity for early validation of that content prior to the combined release. Releases can also be provided outside of the monthly schedule when there is an exceptional need[18]. It is advised that users update their systems once a month to ensure that they are running the most recent version of the operating system and can take advantage of newly released fixes.

Most institutions do not accurately record their security practices and policies, which should involve all relevant parties, including outside partners and software providers. This is a significant component of information security, and it needs to be handled accordingly.

8.2. Survey on e-Learning end users

The end users, also referred to as direct beneficiaries of e-Learning platforms, are online students. A survey was created and given to a small group of chosen e-Learning end users for review in order to gauge the level of caution that online students use when using it. The end-user survey indicated that most online students did not understand the importance of their personally identifiable information, as more than 38.1% were not used to changing their passwords across various online platforms. While 19% change their passwords monthly and quarterly, respectively.

Many users have a good understanding of using a character combination password, which is quite commendable. However, the majority of the online learners that participated in the assessment lack understanding of the importance of frequent device updates, as only 38.1% can update their devices once a month.

9. Conclusion

The management of digital information requires APIs. Adhering to standard usability heuristics is one way to create security APIs that don't have glaring usability issues. All security APIs should go through extensive research and development to improve their usability for online educational platforms. When well-designed APIs are used in online transactions, the end users receive a more secure and dependable system. Online educational platforms may increase the value of each piece of material by publishing APIs to internal, partner, and external developers using an API MP. Through developer engagement, business insights, analytics, security, and protection, it provides the fundamental components necessary to guarantee an effective API operation. Access to important and protected assets and data is made possible through APIs. In order to prevent unauthorized and unlawful access to the underlying assets, API security is essential. APIs are susceptible to a certain type of attack since they are programmatic in nature and widely available. The importance of and best practices for API security must be understood by all parties, including users of the e-Learning environment, application developers, and decision-makers for online platforms. This would increase the security of the

online materials' legitimacy.

Author contributions

Conceptualization, AYM, LOB and KEU; methodology, AYM, LOB and KEU; software, AYM; validation, AYM, LOB and KEU; formal analysis, AYM, LOB and KEU; investigation, AYM; resources, AYM, LOB and KEU; data curation, AYM; writing—original draft preparation, AYM and KEU; writing—review and editing, AYM, LOB, KEU, AOA, VN, IIS, UCN, EL and MA; visualization, AYM, LOB and KEU; supervision, LOB and KEU; project administration, AYM, LOB and KEU; funding acquisition, AYM, LOB, KEU, AOA, VN, IIS, UCN, EL and MA. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

The authors appreciate the authors and publishers, whose articles were used as guides for this study. Also, the authors express gratitude to their respective institutions and the Africa Centre of Excellence on Technology Enhanced Learning (ACETEL), National Open University of Nigeria, Abuja, for supporting this study.

Conflict of interest

The authors declare no conflict of interest.

References

1. Nneji CC, Urenyere R, Ukhurebor KE, et al. The impacts of COVID-19-induced online lectures on the teaching and learning process: An inquiring study of junior secondary schools in Orlu, Nigeria. Frontiers in Public Health. 2022, 10. doi: 10.3389/fpubh.2022.1054536.
2. Asanga MP, Essiet UU, Ukhurebor KE, et al. Social Media and Academic Performance: A Survey Research of Senior Secondary School Students in Uyo, Nigeria. International Journal of Learning, Teaching and Educational Research. 2023, 22(2): 323-337. doi: 10.26803/ijlter.22.2.18.
3. Ndunagu JN, Ukhurebor KE, Adesina A. Virtual laboratories for STEM in Nigerian higher education: The National Open University of Nigeria learners' perspective. In: Elmoazen, R., López-Pernas, S., Misiejuk, K., Khalil, M., Wasson, B., Saqr, M (Eds.), Proceedings of the Technology-Enhanced Learning in Laboratories Workshop (TELL 2023), 3393, 38-48.
4. Nwankwo W, Ukhurebor KE. Web forum and social media: A model for automatic removal of fake media using multilayered neural networks. International Journal of Scientific & Technology Research. 2020, 9(1), 4371-4377.
5. Hussaini AR, Ibrahim S, Ukhurebor KE, et al. The Influence of Information and Communication Technology in the Teaching and Learning of Physics. International Journal of Learning, Teaching and Educational Research. 2023, 22(6): 98-120. doi: 10.26803/ijlter.22.6.6.
6. AltexSoft a Data Science and Software Engineering consulting company founded in 2007 at US Altexsoft.com (accessed on 26 May 2023).
7. Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud system. Available online: https://aws.amazon.com (accessed on 26 May 2023).
8. Patnaik N, Dwyer A, Hallett J, et al. SLR: From Saltzer and Schroeder to 2021…47 Years of Research on the Development and Validation of Security API Recommendations. ACM Transactions on Software Engineering and Methodology. 2023, 32(3): 1-31. doi: 10.1145/3561383.
9. Wu H, Xu L, Niu X, et al. Combinatorial testing of RESTful APIs. Proceedings of the 44th International Conference on Software Engineering. Published online May 21, 2022. doi: 10.1145/3510003.3510151.
10. Hussain F, Hussain R, Noye B, et al. Enterprise API Security and GDPR Compliance: Design and Implementation Perspective. IT Professional. 2020, 22(5): 81-89. doi: 10.1109/mitp.2020.2973852.
11. Hussain F, Li W, Noye B, et al. Intelligent Service Mesh Framework for API Security and Management. 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). Published online October 2019. doi: 10.1109/iemcon.2019.8936216.
12. Iacono LL, Gorski PL. I Do and I Understand. Not Yet True for Security APIs. So Sad. Proceedings 2nd European Workshop on Usable Security. Published online 2017. doi: 10.14722/eurousec.2017.23015

13. Tasnim Z, Chadni S, Muhammad I, Babar A. APIRO: A Framework for Automated Security Tools API Recommendation. Kennesaw State University Digital Commons@Kennesaw State University. 2023.
14. Ouyang R, Wang J, Xu H, et al. A Microservice and Serverless Architecture for Secure IoT System. Sensors. 2023, 23(10): 4868. doi: 10.3390/s23104868.
15. Pan Y, Jia H, Liu W, et al. Mimicry API Gateway Decision Algorithm Based on Trust Distribution. Journal of Physics: Conference Series. 2023, 2424(1): 012004. doi: 10.1088/1742-6596/2424/1/012004
16. Brajesh D. API Management Chapter 2. 2017. p.16. doi: 10.1007/978-1-4842-1305-6_2.
17. Advanced API security: Protecting the digital economy and Empowering innovation Company. Available online: wib.com (accessed on 26 May 2023).
18. Windows technical documentation for developers and IT pros. Available online: https://learn.microsoft.com/en-us/windows/deployment/update/quality-updates (accessed on 26 May 2023).