PCMJ

Progress in Computer and Mathematics Journal (PCMJ)
volume 2 [August, 2025]
e-ISSN: 3030-6728
Website: fskmjebat.uitm.edu.my/pcmj

AI RECOMMENDATION PENETRATION TESTING TOOL FOR SQL INJECTION: LINEAR REGRESSION

Norshahira Elliyana Ahmad Fuad
Universiti Teknologi MARA (UiTM)
shahiraelliyana294@gmai.com

Shahadan Saad*
Universiti Teknologi MARA (UiTM)
shahadan@fskm.uitm.edu.my

Article Info

Abstract

This project addresses to the urgent need for enhanced security assessments by incorporate artificial intelligence (AI) into penetration testing. This approach is essential because of the evolving landscape of cyber threats and the continuous advancement of technology. The issue statement highlight the importance of updating security measures to avoid the latest threats and vulnerabilities. Based on this, the research employ a linear regression technique in the AI penetration testing. This technique involves the development and implementation of an algorithm designed to improve the efficiency of security evaluations by suggest the suitable penetration testing tools. The important of this study lies in its response to the labor and time-consuming nature of current testing methods. The project objective is to apply an AI linear regression algorithm to improve not only the efficiency and flexibility of penetration testing but also to provide strong protection against many cybersecurity threats by suggesting the suitable tools. By suggesting the best tools, this approach reduces the manual effort required, and reduce time-consuming to choose the suitable tools, allowing security professionals to focus on more complex tasks. This project, Extreme programming methodology was used to construct the flow of the project. To make sure this project going smoothly many hardware and software were used, such as Django, Nginx, VMware and many more. In this project, many tests were carried out such as unit testing and integration testing. While doing this project, several limitation were found such as the low accuracy of the tools suggestion, the installation problem and the result that not detail. This limitation can be improve in future works where they will be many advance technology in the future. In conclusion, the objective of this project is success because the linear regression algorithm was able to be insert in the penetration testing framework.

Received: August 2024
Accepted: March 2025
Available Online: August 2025

Keywords: Artificial Intelligence (AI), Penetration Testing, Linear Regression, Tool Suggestion, Extreme Programing Methodology, Django.

PCMJ
Progress in Computer and Mathematics Journal (PCMJ)
volume 2 [August, 2025]
e-ISSN: 3030-6728
Website: fskmjebat.uitm.edu.my/pcmj

INTRODUCTION

Cyber threats such as unauthorized access, data breaches, and various forms of cyberattacks pose serious challenges to organizations. Consequently, robust security measures like penetration testing are crucial for safeguarding these assets. Traditionally, penetration testing has been a key security assessment tool to identify vulnerabilities in systems, networks, and applications by simulating real-world attacks

This project explores the application of artificial intelligence (AI) in penetration testing, specifically utilizing a linear regression algorithm to enhance the accuracy and efficiency of security assessments. The study aims to develop an AI-driven penetration testing framework that can predict future security risks and recommend the most appropriate tools for identifying vulnerabilities within an organization's IT infrastructure. The primary focus is on employing linear regression techniques to analyze data, providing valuable insights that aid in making informed security decisions.

The rapidly evolving nature of cyber threats, coupled with the pace of technological progress, underscores the need for innovative security assessment methods. Current penetration testing techniques are often labor-intensive and time-consuming, which can lead to overlooked vulnerabilities. Integrating AI into these processes promises to streamline testing procedures, enhance accuracy, and provide stronger protection against emerging cybersecurity risks.

This project leverages tools such as Nmap, SQLMap, and Gobuster to collect data on vulnerabilities, which are then analyzed using AI-driven techniques. The research aims to demonstrate that AI, particularly through the use of linear regression, can significantly improve the penetration testing process by making it more efficient, accurate, and capable of real-time threat detection and reporting. The significance of this study lies in its potential to offer a non-bias penetration testing framework and a secure data management system, both of which are critical to advancing cybersecurity practices.

LITERATURE REVIEW

Penetration testing, originating from military network defense technologies, was developed in the early 1970s to assess computer system security (Zheng et al., 2020). Initially used by the Department of Defense to identify security flaws, it spurred the creation of more secure systems (Mamilla, 2021). Penetration testing simulates attacks to evaluate network security, making it

a vital tool for organizations to detect and prevent potential threats (Zheng et al., 2020). However, with the increasing complexity of systems in large enterprises, performing penetration testing without expert assistance has become increasingly challenging.

Linear Regression Algorithm

Linear regression is a key machine learning approach for predicting a continuous outcome based on one or more predictor variables, commonly used in estimation problems (Tang, Lu, Pang, Li, & Su, 2019). The method assumes that samples in the same class correspond to the same linear subspace, represented by a linear equation. Linear regression has been evaluated on datasets like the Fisher iris, Forensic Glass, Japanese credit, and Pima Indian Diabetes datasets (Şahın, Akleylek, & Kiliç, 2022), and is widely used in pattern recognition and face categorization scenarios (Tang et al., 2019).

There are several types of linear regression models, including simple linear regression, multiple linear regression (MLR), polynomial regression, and the least square method (LSM). Simple linear regression models a linear relationship between a single independent variable xxx and a dependent variable yyy, where β0\beta_0β0 is the intercept and β1\beta_1β1 is the slope (Maulud & Abdulazeez, 2020). Multiple linear regression extends this by examining the relationship between several explanatory variables and one response variable. The least square method (LSM) is commonly used to find the best-fitting line or curve by minimizing the squared differences between observed and predicted values (Maulud & Abdulazeez, 2020).

Machine Learning In Penetration Testing

In the rapidly evolving cybersecurity field, traditional penetration testing methods often struggle to keep pace with emerging threats. Machine learning (ML) enhances penetration testing by automating vulnerability detection, making it faster and more accurate. ML models can simulate attacks on security systems, uncovering vulnerabilities more efficiently than manual methods. For example, reinforcement learning has been shown to find the same number of vulnerabilities in network intrusion detection systems (NIDS) in half the time of manual inspections (Apruzzese et al., 2022).

In cybersecurity, ML is increasingly applied to tasks like SQL-injection attack assessments. By using ML algorithms, security professionals can simulate realistic attacks on databases to uncover flaws that might be missed by traditional methods (Apruzzese et al., 2022). These

advancements demonstrate ML's potential to transform penetration testing by enhancing both its effectiveness and scope, helping security professionals stay ahead of potential attackers.

METHODOLOGY

Extreme Programming (XP) was a software development process that was part of the agile methodology. It is a flexible and iterative method to software development that focuses on client happiness, continual feedback, and the flexibility to adapt to changing requirements. Kent Beck launched XP in the late 1990s with the goal of improving software quality and responsiveness to changing client needs.

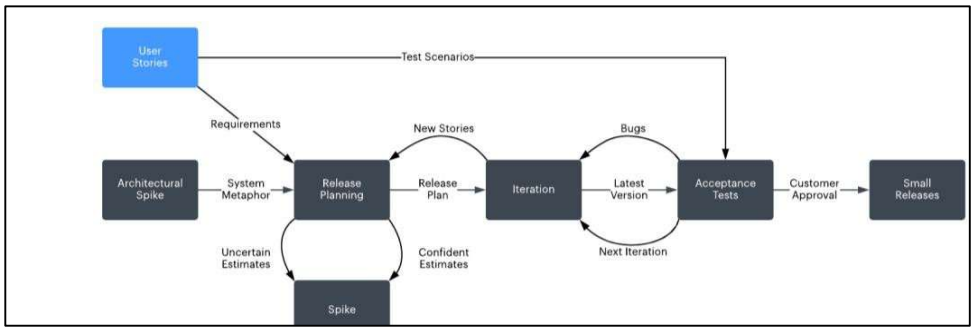

Figure 1: Extreme Programming Methodology

Extreme Programming was ideal for projects where requirements were bound to change often. It include a collection of methods and ideas that contributed to the development process's overall agility and flexibility. Extreme Programming (XP) was a collection of criteria, ideas, and practices that contributed to its agile and customer-focused approach to software development.

System Workflow

This phase involved creating a flowchart that visually represented the project, helping developers and users understand how data flowed through the penetration testing process and how the system was designed. The flowchart served as a communication tool, ensuring everyone had a shared understanding of the system's features, including user input, AI analysis, authentication, reporting, and continuous improvement processes.

The user flowchart detailed the process for interacting with the AI Recommendation Penetration Testing Tool for SQL Injection using linear regression. Starting at the homepage,

users could log in or register, access the dashboard, input target criteria, and receive tool suggestions. They could also run specific penetration testing tools like Nmap, SQLMap, or Gobuster, view their history, and log out to end the session.

The AI flowchart illustrated how the system processed user input, analyzed data using the linear regression method, and generated responses. If users disagreed with the AI's suggestions, they could provide feedback, which was used to revise and align the context, and the updated information was saved to the database.

Finally, the admin flowchart outlined the steps for administrators to manage the system. Admins could log in, view and edit user information, manage user feedback, and log out after completing these tasks.

System Architecture

The system architecture for the penetration testing tools recommendation system was a conceptual model that outlined the system's structure, behavior, and interactions, ensuring smooth operations between hardware and software components. The architecture was divided into two main parts: Front End and Back End.

The Front End consisted of Django, JavaScript, HTML, and CSS, responsible for handling user interfaces and interactions. On the Back End, several layers were involved: the Business Layer, AI Model Layer (implementing the Linear Regression Algorithm), Pentest Tools, and the Data Layer.

In the Data Layer, SQLite was used as the database management system. Its lightweight, server-less, and self-contained features made it ideal for storing user information, tool execution history, suggestions, and feedback. Django's ORM facilitated easy interaction with the SQLite database, allowing for efficient database operations without needing to write raw SQL queries.

By combining Django on the frontend and SQLite in the Data Layer, the system efficiently processed client requests while integrating the AI model for generating relevant pentest tool recommendations.

PCMJ
Progress in Computer and Mathematics Journal (PCMJ)
volume 2 [August, 2025]
e-ISSN: 3030-6728
Website: fskmjebat.uitm.edu.my/pcmj

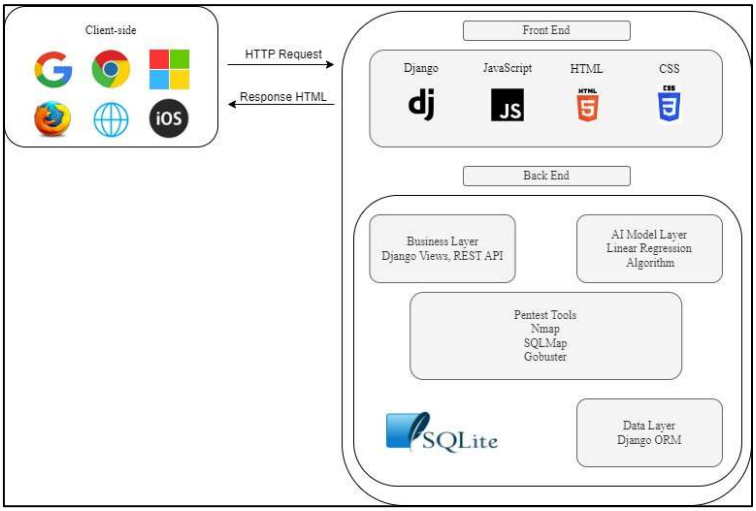

Figure 2: System Architecture

RESULT AND DISCUSSION

The project aimed to assess the feasibility and effectiveness of integrating a linear regression algorithm into the penetration testing process. Although accuracy was not the primary focus, the project successfully demonstrated the integration of linear regression to enhance penetration testing methodologies.

The testing phase involved validating the system's implementation and functionality through integration and unit testing. Integration testing ensured that all modules, including the linear regression algorithm, data collection, and analysis, worked together seamlessly, while unit testing focused on verifying individual components such as model training, pentest record creation, and tool suggestion history.

The project successfully integrated linear regression into the penetration testing framework, with tools like Nmap, SQLMap, and Gobuster used to assess target systems. The system demonstrated its ability to provide relevant tool suggestions, even though accuracy was not emphasized, validating the potential of using AI in real-world penetration testing scenarios. Nmap and SQLMap were recommended based on user input, and the system effectively analyzed features to provide suitable tool suggestions. Similarly, SQLMap and Gobuster were suggested for other scenarios, showcasing the system's ability to extract meaningful information from user input to match penetration testing needs.

PCMJ
Progress in Computer and Mathematics Journal (PCMJ)
volume 2 [August, 2025]
e-ISSN: 3030-6728
Website: fskmjebat.uitm.edu.my/pcmj

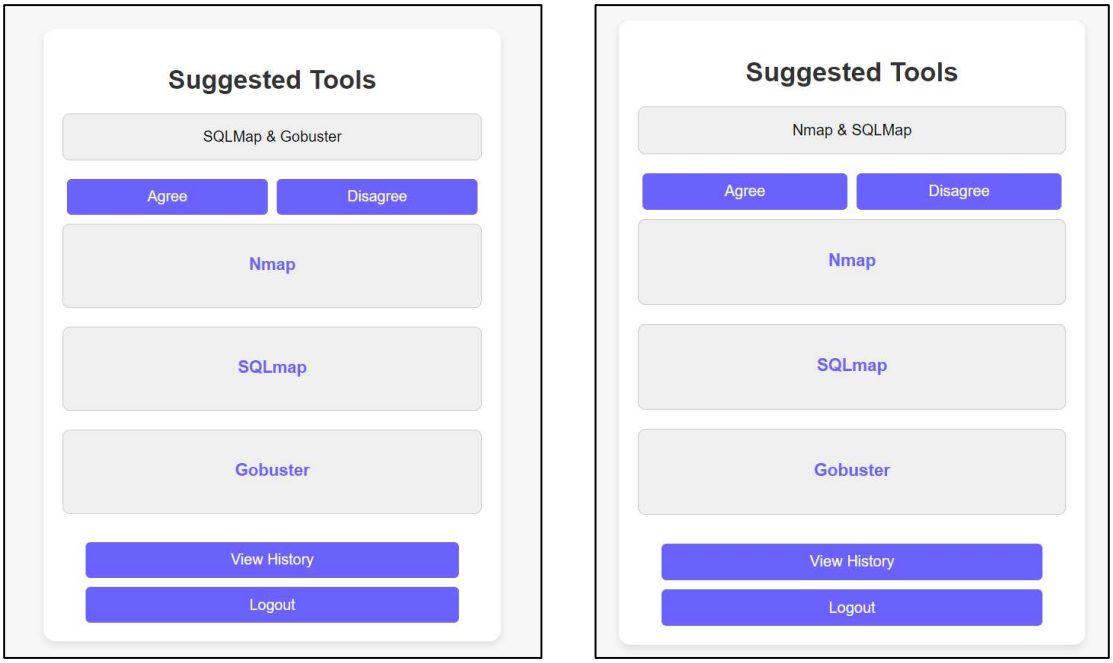

Figure 3: Suggested Tools Page

Nmap identified open ports and mapped the attack surface of the target system, while SQLMap detected SQL injection vulnerabilities and retrieved database information. Gobuster found hidden directories on web servers that could lead to further security investigations. Additionally, admins could upload CSV files to train the model and view accuracy results, and the system allowed the conversion of PKL files to BKA files for download, confirming the successful execution of these functions.

CONCLUSION

In conclusion, incorporating AI into penetration testing enhances productivity and flexibility, addressing the limitations of traditional methods and providing robust defenses against evolving cybersecurity threats. Despite some limitations in tool recommendations and execution details, future improvements can make the AI Recommendation Penetration Testing Tool a more powerful and reliable asset in cybersecurity.

REFERENCES

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

Abu-Dabaseh, F., & Alshammari, E. (2018). Automated Penetration Testing: An Overview. Computer Science & Information Technology. https://doi.org/10.5121/csit.2018.80610

Aldridge, I. (2023). The AI Revolution: From Linear Regression to ChatGPT and beyond and How It All Connects to Finance. The Journal of Portfolio Management, 49(9), 64–77. https://doi.org/10.3905/jpm.2023.1.519

Alessandro Confido, Ntagiou, E. V., & Wallum, M. (2022). Reinforcing Penetration Testing Using AI. https://doi.org/10.1109/aero53065.2022.9843459

Apruzzese, G., Laskov, P., de Oca, E. M., Mallouli, W., Rapa, L. B., Grammatopoulos, A. V., & Franco, F. D. (2022). The Role of Machine Learning in Cybersecurity. Digital Threats: Research and Practice, 4(1). https://doi.org/10.1145/3545574

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., & Akin, E. (2023). A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions. Electronics, 12(6), 1–42. https://doi.org/10.3390/electronics12061333

Biju, J. M., Gopal, N., & Prakash, A. J. (2019). Cyber attacks and its different types. International Research Journal of Engineering and Technology, 6(3), 4849-4852

C. Gnaneswar Raju, V. Amudha, & G Sajiv. (2023). Comparison of Linear Regression and Logistic Regression Algorithms for Ground Water Level Detection with Improved Accuracy. https://doi.org/10.1109/iconstem56934.2023.10142495

Candel, C. J. F., Sevilla Ruiz, D., & García-Molina, J. J. (2022). A unified metamodel for NoSQL and relational databases. Information Systems, 104, 101898. https://doi.org/10.1016/j.is.2021.101898

Chowdhary, A., Huang, D., Mahendran, J. S., Romo, D., Deng, Y., & Sabur, A. (2020). Autonomous Security Analysis and Penetration Testing. 2020 16th International Conference on Mobility, Sensing and Networking (MSN). https://doi.org/10.1109/msn50589.2020.00086

Devalla, V., Srinivasa Raghavan, S., Maste, S., Kotian, J. D., & Annapurna, Dr. D. (2022). mURLi: A Tool for Detection of Malicious URLs and Injection Attacks. Procedia Computer Science, 215, 662–676. https://doi.org/10.1016/j.procs.2022.12.068

Garg, D., & Bansal, N. (2021). A Systematic Review on Penetration Testing. 2021 2nd Global Conference for Advancement in Technology (GCAT). https://doi.org/10.1109/gcat52182.2021.9587771

Gaya, M. S., Abba, S. I., Abdu, A. M., Tukur, A. I., Saleh, M. A., Esmaili, P., & Wahab, N. A. (2020). Estimation of water quality index using artificial intelligence approaches and multi-linear regression. IAES International Journal of Artificial Intelligence (IJAI), 9(1), 126. https://doi.org/10.11591/ijai.v9.i1.pp126-134

Gjerding, M., Skovhus, T., Rasmussen, A., Bertoldo, F., Larsen, A. H., Mortensen, J. J., & Thygesen, K. S. (2021). Atomic Simulation Recipes: A Python framework and library for automated workflows. Computational Materials Science, 199, 110731. https://doi.org/10.1016/j.commatsci.2021.110731

Golbaz, S., Nabizadeh, R., & Sajadi, H. S. (2019). Comparative study of predicting hospital solid waste generation using multiple linear regression and artificial intelligence. Journal of Environmental Health Science and Engineering, 17(1), 41–51. https://doi.org/10.1007/s40201-018-00324-z

Hance, J., Milbrath, J., Ross, N., & Straub, J. (2022). Distributed Attack Deployment Capability for Modern Automated Penetration Testing. Computers, 11(3), 33. https://doi.org/10.3390/computers11030033

Heiding, F., Süren, E., Olegård, J., & Lagerström, R. (2023). Penetration testing of connected households. Computers & Security, 126, 103067. https://doi.org/10.1016/j.cose.2022.103067

Hope, T. M. H. (2020). Linear regression. Machine Learning, 67–81. https://doi.org/10.1016/b978-0-12-815739-8.00004-3

Jayasuryapal, G., Pranay, P. M., Kaur, H., & Swati. (2021). A Survey on Network Penetration Testing. 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM). https://doi.org/10.1109/iciem51511.2021.9445321

Jaydev Kumar Mahato, & Sunil Kumar Gupta. (2021). Exploring applicability of artificial intelligence and multivariate linear regression model for prediction of trihalomethanes in drinking water. International Journal of Environmental Science and Technology, 19(6), 5275–5288. https://doi.org/10.1007/s13762-021-03392-1

Jumin, E., Zaini, N., Ahmed, A. N., Abdullah, S., Ismail, M., Sherif, M., Sefelnasr, A., & ElShafie, A. (2020). Machine learning versus linear regression modelling approach for accurate ozone concentrations prediction. Engineering Applications of Computational Fluid Mechanics, 14(1), 713–725. https://doi.org/10.1080/19942060.2020.1758792

Kasim, Ö. (2021). An ensemble classification-based approach to detect attack level of SQL injections. Journal of Information Security and Applications, 59, 102852. https://doi.org/10.1016/j.jisa.2021.102852

Khanzode, K. C. A., & Sarode, R. D. (2020). Advantages and disadvantages of artificial intelligence and machine learning: A literature review. International Journal of Library & Information Science (IJLIS), 9(1), 3.

Kornienko, D. V., Mishina, S. V., Shcherbatykh, S. V., & Melnikov, M. O. (2021). Principles of securing RESTful API web services developed with python frameworks. Journal of Physics: Conference Series, 2094(3), 032016. https://doi.org/10.1088/1742-6596/2094/3/032016

Kumar, M. S., Ben-Othman, J., Srinivasagan, K. G., & Krishnan, G. U. (2019, March 1). Artificial Intelligence Managed Network Defense System against Port Scanning Outbreaks. IEEE Xplore. https://doi.org/10.1109/ViTECoN.2019.8899380

Li, L., Dong, J., Zuo, D., & Wu, J. (2019). SLA-Aware and Energy-Efficient VM Consolidation in Cloud Data Centers Using Robust Linear Regression Prediction Model. IEEE Access, 7, 9490–9500. https://doi.org/10.1109/access.2019.2891567

Liu, H., Yang, L., & Wu, H. (2022). Design of Embedded Data Acquisition and Management System Based on SQLite Database. 2022 11th International Conference of Information and Communication Technology (ICTech)). https://doi.org/10.1109/ictech55460.2022.00074

Mamilla, S. R. (2021). A Study of Penetration Testing Processes and Tools. Electronic Theses, Projects, and Dissertations. https://scholarworks.lib.csusb.edu/etd/1220/

Maulud, D., & Abdulazeez, A. M. (2020). A Review on Linear Regression Comprehensive in Machine Learning. Journal of Applied Science and Technology Trends, 1(4), 140– 147. https://doi.org/10.38094/jastt1457

Pina, E., Sá, F., & Bernardino, J. (2022). NewSQL Databases Assessment: CockroachDB, MariaDB Xpand, and VoltDB. Future Internet, 15(1), 10. https://doi.org/10.3390/fi15010010

Radmanovic, M. M. (2022). A Comparison of Computing Spectral Transforms of Logic Functions using Python Frameworks on GPU. 2022 57th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST). https://doi.org/10.1109/icest55168.2022.9828786

Saber, V., ElSayad, D., Bahaa-Eldin, A. M., & Fayed, Z. (2023). Automated Penetration Testing, A Systematic Review. 2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), 2023 International, 373–380. https://doi.org/10.1109/MIUCC58832.2023.10278377

Sahin, D. O., Akleylek, S., & Kilic, E. (2022). LinRegDroid: Detection of Android Malware Using Multiple Linear Regression Models-Based Classifiers. IEEE Access, 10, 14246–14259. https://doi.org/10.1109/access.2022.3146363

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. SN Computer Science, 2(3), 1–21. Springer. https://doi.org/10.1007/s42979-021-00592-x

Scotland, A., G. Cosne, A. Juraver, A. Karatsidis, J. Penalver-Andres, E. Bartholomé, Kanzler, C. M., C. Mazzà, D. Roggen, Hinchliffe, C., S. Del Din, & Belachew, S. (2024).

DISPEL: a Python Framework for Developing Measures from Digital Health Technologies. IEEE Open Journal of Engineering in Medicine and Biology, 1–5. https://doi.org/10.1109/ojemb.2024.3402531

Sharma, D. K., Mishra, J., Singh, A., Govil, R., Srivastava, G., & Lin, J. C.-W. (2022). Explainable Artificial Intelligence for Cybersecurity. Computers and Electrical Engineering, 103, 108356. https://doi.org/10.1016/j.compeleceng.2022.108356

Stiawan, D., Idris, Mohd. Y., Abdullah, A. H., Aljaber, F., & Budiarto, R. (2017). CyberAttack Penetration Test and Vulnerability Analysis. International Journal of Online Engineering (IJOE), 13(01), 125. https://doi.org/10.3991/ijoe.v13i01.6407

Sun, X. D., Ren, Z., Yang, P. W., Li, J., Chen, H. Y., & Liu, T. Q. (2019). Artificial intelligence design research on the cyber security penetration testing of power grid enterprises. IOP Conference Series: Earth and Environmental Science, 354, 012104. https://doi.org/10.1088/1755-1315/354/1/012104

Tjoa, S., Buttinger, C., Holzinger, K., & Kieseberg, P. (2021). Penetration Testing Artificial Intelligence. https://www.researchgate.net/publication/349172682_Penetration_Testing_Artificial _Intelligence Votipka, D., Stevens, R., Redmiles, E., Hu, J., & Mazurek, M. (2018, May 1). Hackers vs. Testers: A Comparison of Software Vulnerability Discovery Processes. IEEE Xplore. https://doi.org/10.1109/SP.2018.00003

Vrbančič, G., Brezočnik, L., Mlakar, U., Fister, D., & Fister Jr., I. (2018). NiaPy: Python microframework for building nature-inspired algorithms. Journal of Open Source Software, 3(23), 613. https://doi.org/10.21105/joss.00613

Zheng, S., Wu, Y., Wang, S., Wei, Y., Mu, D., He, H., Han, D., Liao, J., & Chen, H. (2020). PTVis: Visual Narrative and Auxiliary Decision to Assist in Comprehending the Penetration Testing Process. IEEE Access, 8, 194523–194540. https://doi.org/10.1109/access.2020.3033391