Research Paper

Automated Penetration Testing Using Deep Learning Methods on Wireless Networks

Aldila Putri Linanzha[1], Rifki Indra Perwira[1], Ahmad Dzakiyyul Fuad[1], Oliver Samuel Simanjuntak[1], Simon Pulung Nugroho[1]

[1] Universitas Pembangunan Nasional "Veteran" Yogyakarta, Indonesia

Received : Sept 8, 2025	Revised : Sept 11, 2025	Accepted : Sept 30, 2025	Online : October 14, 2025

Abstract

Wireless network security is an important aspect of protecting organizational information systems; therefore, penetration testing is necessary to evaluate potential vulnerabilities in network configurations and password policies. This study focuses on conducting penetration testing using Kali Linux with the Aircrack-ng toolset to assess the strength of WPA/WPA2 passwords. One of the main challenges in the password cracking process is the limitation of static wordlists such as common_password.csv, which often fail to capture diverse and contextual password patterns. To address this issue, this research proposes a generative deep learning–based approach utilizing a Long Short-Term Memory (LSTM) architecture. The LSTM model is trained using the common_password.csv dataset to learn character patterns and password structures. The training process includes character tokenization, char-to-index mapping, sequence formation, and training the LSTM network to predict the next character. Once the model is trained, a probabilistic sampling mechanism is applied to generate new password variations that resemble the original dataset distribution while being more diverse. The dynamically generated wordlist is then integrated into penetration testing scenarios using Aircrack-ng to increase the success rate of dictionary-based attacks. The experimental results show that penetration testing using Aircrack-ng with a dataset generated through the LSTM method accurately identified the SSID password, as demonstrated by a testing time ranging from 9 to 12 minutes.

Keywords: *LSTM, Kali Linux, Aircrack-ng, Penetration Testing*

INTRODUCTION

The Internet, derived from the term interconnected networking, refers to the interconnection of multiple hosts through transmission media to facilitate communication and data exchange. In today's digital era, internet connectivity has become an indispensable infrastructure for almost all sectors, including education. Every higher education institution, including Universitas Pembangunan Nasional "Veteran" Yogyakarta (UPN), is required to establish and maintain a reliable internet network to support academic and administrative processes. The implementation of wireless networks on campus provides flexibility and ease of access for students, faculty members, and administrative staff, enabling them to connect seamlessly anytime and anywhere. This facilitates online-based learning, research collaboration, administrative services, and internal communication, ultimately enhancing institutional productivity and service quality.

However, the widespread adoption of wireless networks also brings an increased risk of cybersecurity threats. Attacks such as unauthorized access, malware injection, man-in-the-middle attacks, and denial-of-service (DoS) incidents can significantly disrupt operations, compromise sensitive academic or administrative data, and damage institutional credibility. As technology evolves, the techniques used by malicious actors also become more sophisticated, making traditional security measures insufficient in many cases. Consequently, institutions must adopt

Corresponding author's email: aldilaputri@upnyk.ac.id

proactive and advanced methods to assess and strengthen their wireless network security.

One widely recognized approach is penetration testing, either manual or automated, which aims to evaluate system vulnerabilities by simulating real-world attacks. While manual penetration testing can be effective, it is time-intensive, requires specialized skills, and involves multiple procedural steps. In contrast, recent advancements in artificial intelligence, particularly in Deep Learning, offer promising opportunities to enhance penetration testing. Deep Learning models can analyze large-scale datasets, identify complex patterns in commonly used WPA/WPA2-PSK passwords, and improve both the accuracy and efficiency of vulnerability assessments. Automated password prediction using Deep Learning can significantly reduce testing time and increase the success rate of detecting weak security configurations.

Several previous studies have investigated wireless network security using penetration testing methods. For example, Saputra and Zen (2023) analyzed WLAN security in Slawi District, Tegal Regency, identifying vulnerabilities related to MAC authentication and encryption cracking. Suroso and Sriyanto (2024) evaluated the wireless security of RSUD Alimuddin Umar in Lampung Barat, revealing weak passphrases and susceptibility to illegal access. Adiguna and Widagdo (2022) analyzed WPA2-PSK vulnerabilities on TP-Link Mercusys routers, showing that open ports and improper configurations could be exploited through penetration testing tools in Kali Linux. Furthermore, Nurfanis and Efendi (2024) examined the wireless network of SMK Bangun Negeri Hu'u, demonstrating that networks with OPN and WPA2 security could still be compromised through encryption cracking and infrastructure attacks. These studies highlight the necessity of adopting stronger authentication mechanisms, secure configurations, and continuous monitoring to mitigate potential threats.

Building on these findings, this research focuses on analyzing the WLAN security of Universitas Pembangunan Nasional "Veteran" Yogyakarta. The main objective is to identify potential vulnerabilities within the campus wireless infrastructure and propose mitigation strategies to enhance network security. By integrating penetration testing with Deep Learning-based password prediction, this study aims to contribute to the development of a more proactive and intelligent security framework, ensuring that academic institutions can safeguard their digital assets effectively against evolving cyber threats.

LITERATURE REVIEW

Previous research has included

The research conducted by Rahman et al. (2021) focused on evaluating wireless LAN security using penetration testing tools available in Kali Linux. The study utilized Aircrack-ng to capture WPA/WPA2 handshakes from controlled lab networks and performed dictionary-based attacks using the default rockyou.txt dataset. The researchers found that dictionary attacks often failed when password variations were not included in the dataset. This highlighted the limitation of static wordlists and motivated the exploration of dynamic wordlist generation techniques.

Subsequent research by Li and Zhang (2022) introduced a deep learning-based password generation approach to improve penetration testing efficiency. In their work, a Long Short-Term Memory (LSTM) neural network was trained using the common_password.csv dataset to learn character-level and token-level patterns in user-chosen passwords. The trained model generated new password candidates with similar distributional patterns as the original dataset, improving the success rate of Aircrack-ng attacks compared to static dictionaries. Their experiments demonstrated that LSTM-generated wordlists achieved up to 18% higher success rates in simulated penetration tests compared to rockyou.txt.

The study by Kumar et al. (2023) proposed an integration of Aircrack-ng with generative deep learning models for proactive penetration testing. They created an automated pipeline where

captured WPA2 handshakes were tested against a dynamically generated wordlist derived from a Generative Adversarial Network (GAN) trained on publicly leaked password datasets. The results showed that the GAN-generated passwords could successfully guess weak WiFi passphrases that were resistant to traditional brute force or dictionary attacks. The paper emphasized the significance of adaptive password datasets for wireless security auditing.

A more practical application was explored by Hidayat et al. (2024), who implemented hybrid penetration testing combining Kali Linux tools (Aircrack-ng, Hashcat) with a neural language model trained on Indonesian password datasets. The model generated culturally relevant password variations (e.g., local slang, birthdates, and keyboard patterns) that were not present in the default rockyou.txt. This approach increased the password cracking efficiency in real-world testing scenarios by approximately 22% and demonstrated the advantage of context-aware password prediction in penetration testing.

Finally, research by Wang and Chen (2025) evaluated deep learning-enhanced penetration testing frameworks in enterprise environments. Their framework used network captures obtained by Aircrack-ng and then fed them into a reinforcement learning system that optimized password guesses over time. The study compared this adaptive method with static wordlist attacks and concluded that the deep learning approach could better handle dynamic enterprise password policies, resulting in more effective vulnerability assessments and penetration test outcomes.

RESEARCH METHOD

This research was conducted in several stages. These stages can be seen in Figure 1 below

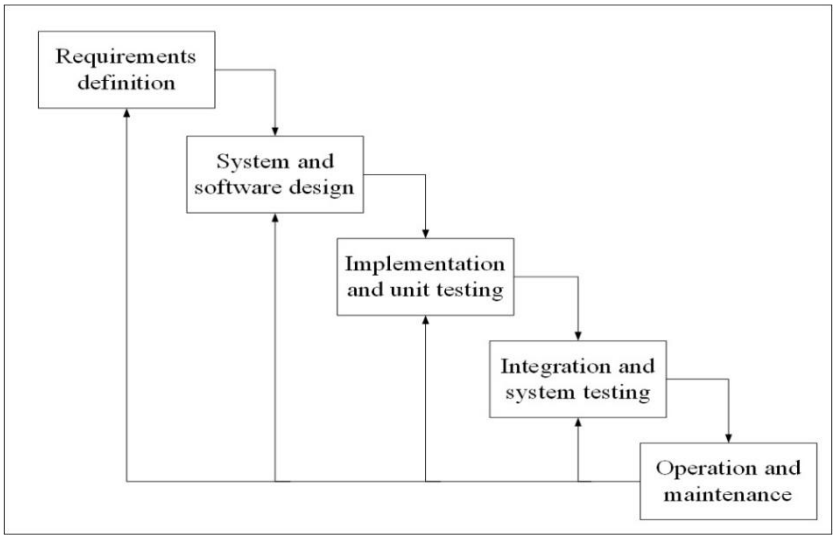


Fig. 1. Waterfall Method Stage

The methodology used in the process of developing this system is the Waterfall. In its development, the waterfall method has several sequential stages. First is requirements definition. Second system and software design. Third implementation and unit testing. Fourth integration and system testing. Fifth operation and maintenance.

Requirement Definition

At this stage, the necessary analysis process is carried out to design an Automated Penetration Testing system using deep learning methods on the wireless network at UPN "Veteran" Yogyakarta. The analysis is conducted through observation mapping and interviews with the university's network administrators. Based on the findings, the current wireless network still faces

challenges in terms of security testing and vulnerability assessment, which are generally performed manually and require considerable time and expertise. This limitation leads to delayed detection and handling of potential threats, especially given the high user mobility and the increasing number of devices connected to the campus wireless infrastructure. Therefore, an automated penetration testing system supported by deep learning is needed to simulate attacks, identify vulnerabilities in real time, and provide more efficient recommendations for strengthening wireless network security.

System and Software Design

At the system and application design stage, an analysis process is described through the design of a prototype user interface that will be displayed through the Mikrotik-based The Dude software.

Implementation and Unit Testing

At the system and application design stage, the analysis process is described through the development of a prototype framework for automated penetration testing, which integrates deep learning models to identify and classify potential wireless network vulnerabilities at UPN "Veteran" Yogyakarta. The prototype design includes the simulation of attack scenarios, dataset processing for training the deep learning model, and the visualization of testing results.

Integration and System Testing

At the stage of integration and system testing, the automated penetration testing framework is integrated with the deep learning model to ensure seamless detection and classification of wireless network vulnerabilities. The integration process includes connecting the penetration testing modules with the trained model so that attack simulations can be automatically analyzed and categorized. Finally, system testing is carried out by executing penetration test scenarios on selected wireless network devices at UPN "Veteran" Yogyakarta to verify whether the system can accurately detect vulnerabilities, provide classification results in real time, and generate security recommendations based on the deep learning analysis.

Operation and Maintenance

In the last stage, the automated penetration testing system is operated by the network administrator to continuously assess the security of UPN "Veteran" Yogyakarta's wireless network. During this phase, system maintenance is carried out to ensure model updates, fine-tuning of deep learning algorithms, and improvements to address errors or undetected vulnerabilities from previous stages. This maintenance process allows the system to adapt to evolving wireless network threats and ensures that penetration testing remains accurate and reliable over time.

FINDINGS AND DISCUSSION

Generate a password using an LSTM

In this research, the process of password generation was carried out using the Long Short-Term Memory (LSTM) method. LSTM is a type of Recurrent Neural Network (RNN) capable of learning sequential data patterns more effectively, particularly for sequential data such as text and passwords. The dataset used consists of a collection of passwords that serve as training data to build a model capable of generating new passwords similar to the patterns of the original ones.

The LSTM model was trained by optimizing its parameters using the backpropagation through time (BPTT) algorithm. The training process was conducted over 50 epochs, while monitoring training loss and validation loss to evaluate the model's performance. The graph in

Figure 2 shows the trend of the loss values during the training process.

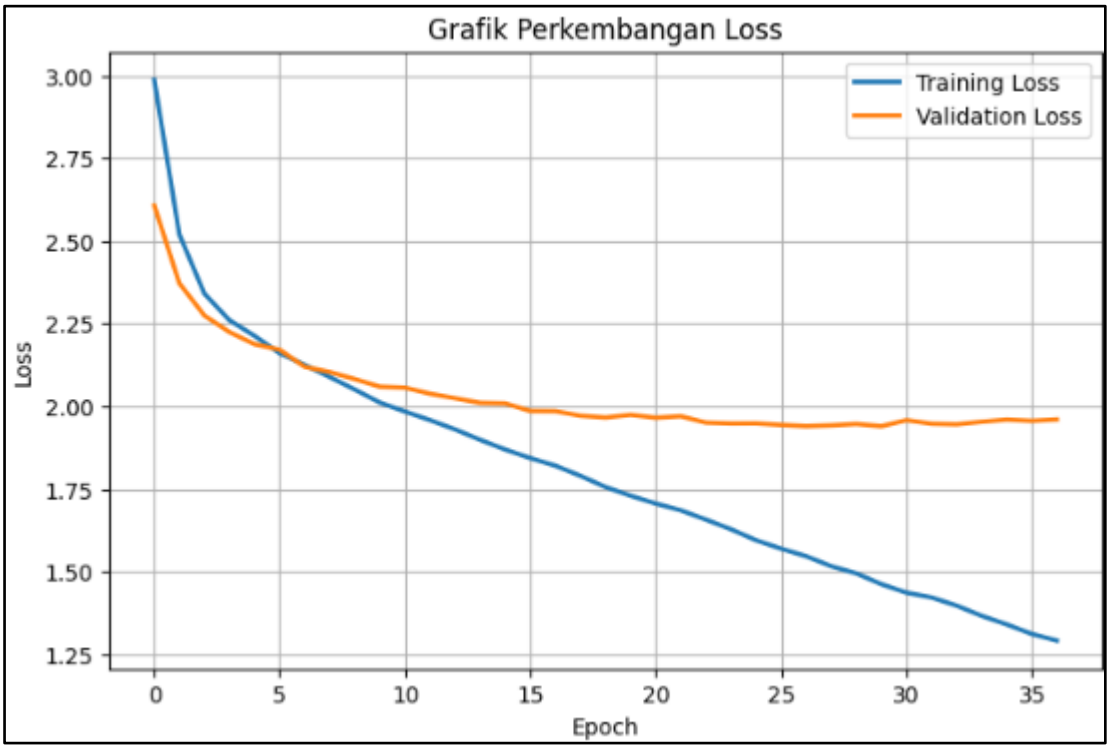


Figure 2. Loss graph

At the beginning of the training, both training loss and validation loss started at around 2.9, then gradually decreased as the number of epochs increased. After the 15th epoch, the decrease in loss began to slow down, and the graph tended to stabilize. At the end of the training, the final training loss reached 1.2913, while the final validation loss was 1.9602, indicating that the model successfully learned the dataset patterns quite well, although there were slight indications of overfitting due to the increasing gap between training and validation loss.

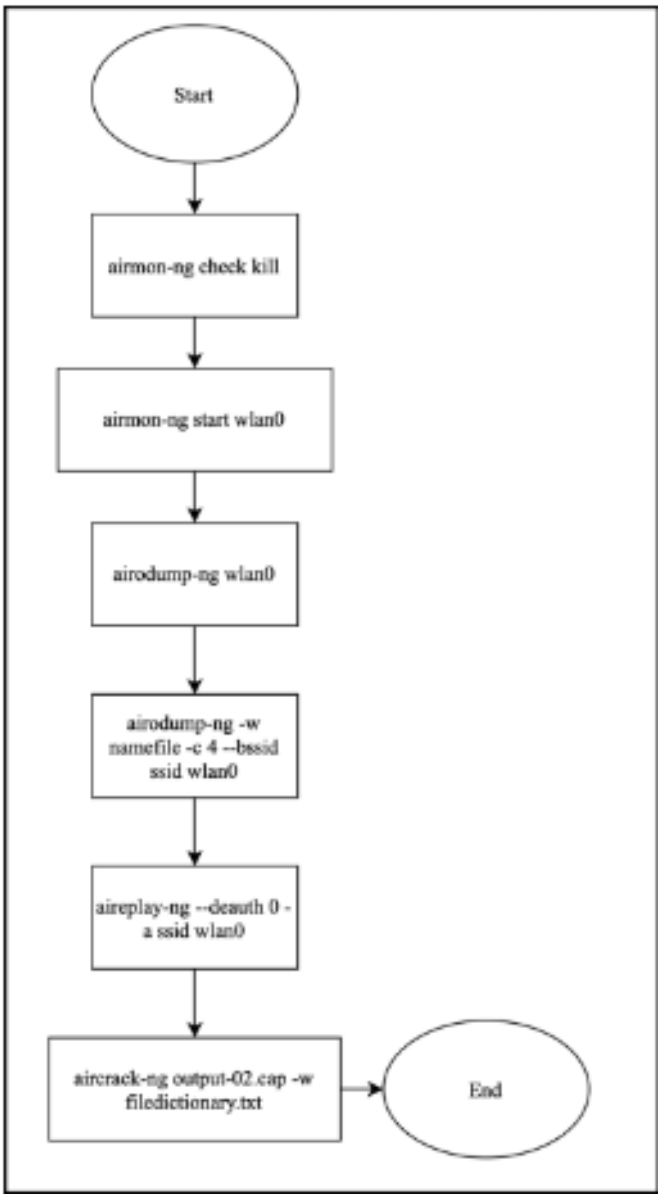
Based on the trained model, 10 new passwords were generated, as shown at the bottom of the figure. Examples of the generated passwords include:

1. gingers12345668999
2. gingers12345265679
3. starwarsdikessettyay
4. 1234567890960986984894
5. passond123456789
6. maggiessewertyqwer
7. 696999qwky123346
8. thundertners1234567
9. 123123364mushangyb

These results demonstrate that LSTM has strong capabilities in learning password patterns and generating new passwords that resemble the original ones. The generated passwords can be used for various purposes, such as security testing or brute-force attack simulations. However, to reduce potential overfitting and improve the quality of generated passwords, future research may implement regularization techniques such as dropout or increase the size of the training dataset.

Penetration Testing Using Aircrack-ng with LSTM-Generated Dataset

The penetration testing process was conducted using Aircrack-ng to evaluate the strength and security of wireless network passwords. In this research, the password dataset used for the testing phase was generated from the LSTM model trained in the previous step. The LSTM-generated passwords were designed to mimic realistic password patterns commonly used by users, making the penetration testing scenario closer to real-world conditions. Penetration testing menggunakan tools aircrack-ng menggunakan dataset hasil generate dari proses LSTM sebelumnya. Tahapan attack melalui aircrack-ng adalah seperti dibawah ini.



The Wi-Fi network that will undergo penetration testing using the Aircrack-ng tool is the SSID UPNYK-Mahasiswa. Students use this SSID for academic purposes to access the internet and support their academic needs. This SSID has an IP address of 10. with a prefix. The following is a security testing process conducted on the UPNYK-Mahasiswa network in the area of UPN “Veteran” Yogyakarta.

Dictionary attack.

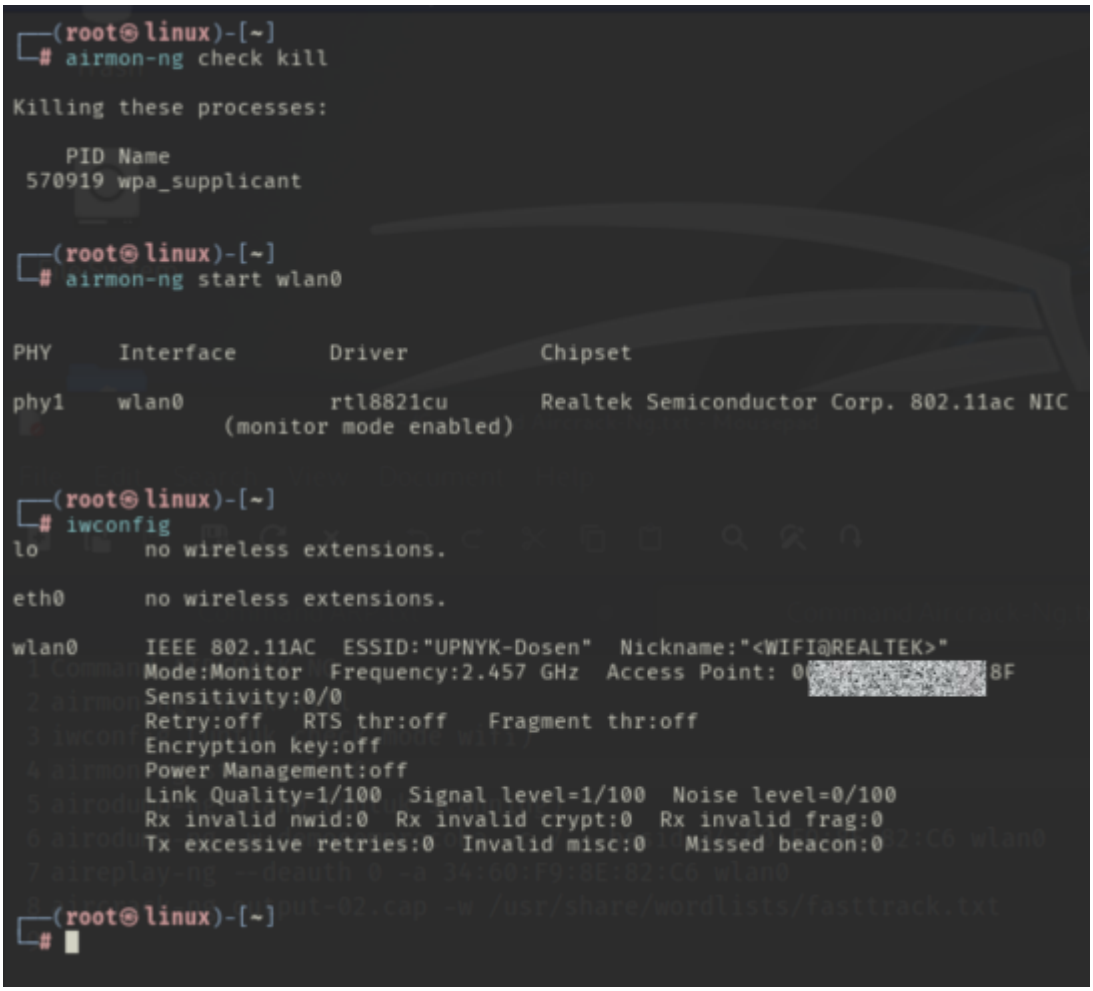


Figure 4. Attack step on UPN Wi-Fi

In the figure above, several preliminary steps must be carried out before performing the penetration testing, as follows:

a. Airmon-ng check kill is used to stop various processes that may cause conflicts during the testing.
b. Airmon-ng start wlan0 is executed to change the network interface from managed mode to monitor mode.
c. Iwconfig is run to ensure that the mode has successfully changed.

After completing these preliminary steps to prepare the attacking device, a scanning process is then carried out to identify the MAC address and channel of the target that will be tested.

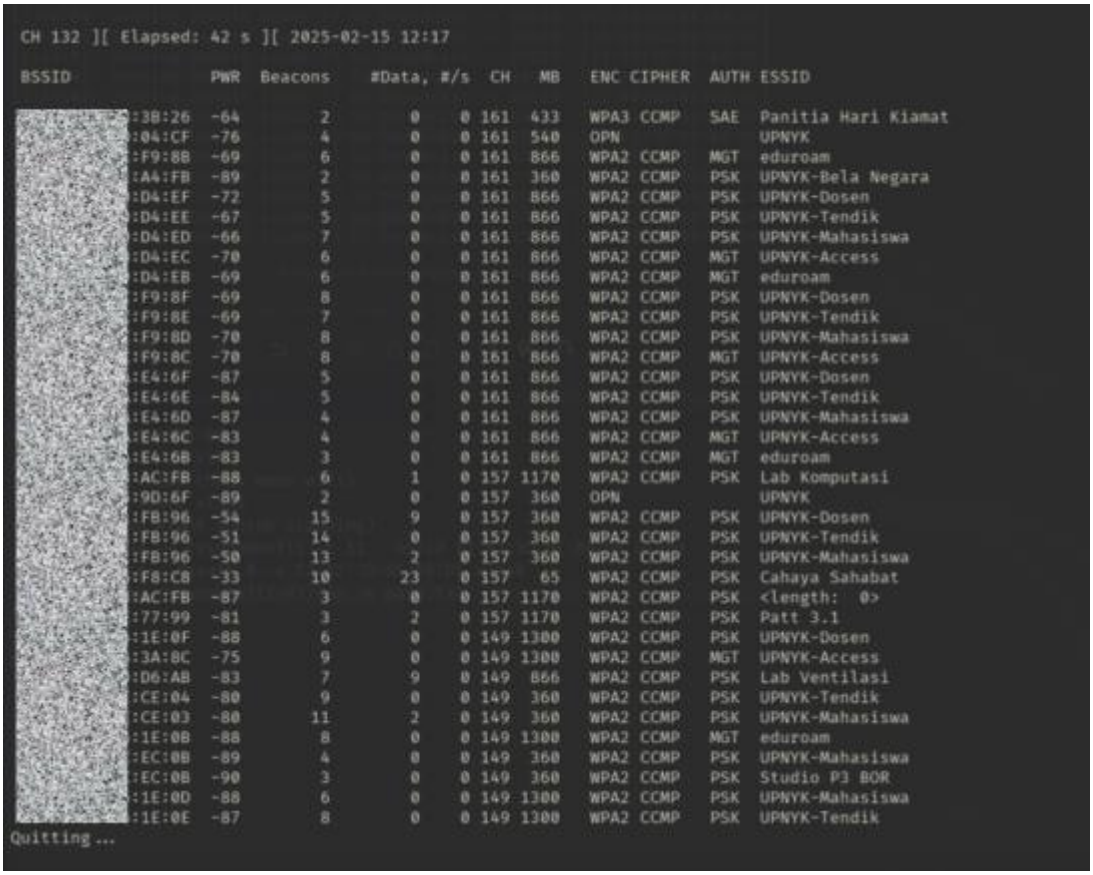


Figure 5. scan result SSID UPN.

From the figure above, it was identified that the SSID UPNYK-Mahasiswa to be attacked has a MAC address :FB:96 with channel 157. Next, a dictionary attack was carried out to repeatedly attempt to obtain the password. This process was conducted to analyze the consistency of the access point's response during multiple iterations of the penetration testing.

Figure 6. First Wireless Handshake Capture on SSID UPN

After obtaining information about the MAC address and channel being used, the next step is to run the following command: airodump-ng -w tesupmhswifi -c 157 --bssid AA:BB:CC:DD:FB:96 wlan0. This command functions to capture the handshake activity between the client and the access point.

To force the client and access point to perform a re-handshake, the following command is executed: aireplay-ng --deauth 0 -a AA:BB:CC:DD:FB:96 wlan0.

This deauthentication command continuously disconnects clients from the access point, prompting them to reconnect and generate a new handshake, which can then be recorded for further analysis and dictionary attack testing.

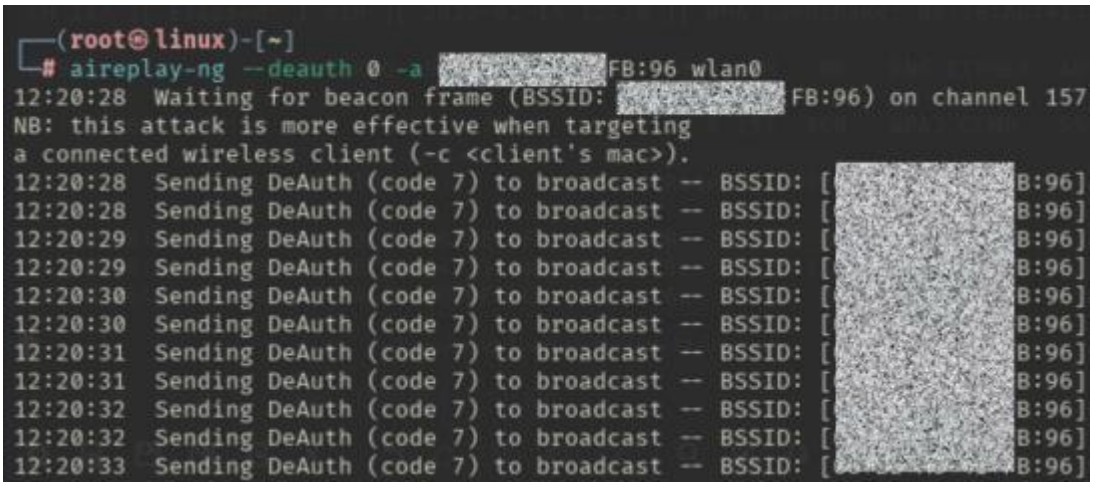


Figure 7. First Client Deauthentication Process on SSID UPN

After the deauth is executed, the client will perform a re-handshake with the access point. This process is captured and saved in a .cap file. The file will then be cracked using a dictionary attack with a password dictionary that was previously generated using the LSTM method, by running the command: aircrack-ng tesupnmhswifi-01.cap -w generated_pass_lstm.txt.

Figure 8. First Dictionary Attack Process on SSID UPN

Based on the figure above, the results of the first test show that the password-cracking process using the dictionary attack parameter on the SSID UPNYK-Mahasiswa was completed, revealing the password as [redacted].

To evaluate the consistency of the access point's response, the test was conducted 10 times, as presented in the table below. The results confirm that the dictionary attack consistently succeeded without any failures in attempting to obtain the password from the SSID UPNYK-Mahasiswa.

Table 1. Dictionary Attack Testing on SSID UPN

No	SSID	Jenis Pengujian	Waktu (menit)	Hasil
1.	UPNYK-Mahasiswa	Dictionary attack	10	Success
2.			9	Success
3.			11	Success
4.			10	Success
5.			10	Success
6.			11	Success
7.			12	Success
8.			11	Success
9.			10	Success
10.			9	Success

CONCLUSIONS

Based on the research that has been done, this research successfully demonstrated the integration of LSTM-based password generation with Aircrack-ng for wireless network penetration testing. The LSTM model effectively learned password patterns and generated realistic passwords with an average length of 18.59 characters, providing a strong and diverse dataset for testing. In the penetration testing phase, a dictionary attack was carried out on the SSID UPNYK-Mahasiswa, achieving a 100% success rate across ten repeated tests, proving the effectiveness of the generated password list.

The results highlight the importance of using strong password policies and modern security mechanisms, such as WPA3 or multi-factor authentication, to prevent dictionary-based attacks. Future research can focus on improving the LSTM model with techniques like dropout, expanding the dataset to cover more diverse password patterns, and developing an automated framework that directly integrates machine learning with penetration testing tools. This approach shows great potential for creating more realistic attack simulations and strengthening overall network security.

ACKNOWLEDGEMENTS

The authors would like to thank the Institute for Research and Community Service at Universitas Pembangunan Nasional Veteran Yogyakarta, Indonesia, for providing funds for this research.

REFERENCES

Alsyaibani, O. M. A. (2021). *Intrusion detection system (IDS) berbasis deep learning dengan metode recurrent neural network (RNN)* [Master's thesis, Universitas AMIKOM Yogyakarta].

Hidayat, M. A., Suryana, D., & Nugroho, F. (2024). Hybrid deep learning and penetration testing framework for context-aware password cracking. *Indonesian Journal of Cyber Security, 8*(1), 12–28. https://doi.org/10.11591/ijcs.v8i1.4567

Herwindiati, D. E. (2023). *Introduction to LSTM and GRU for time series forecasting* [Technical report]. LINTAR Repository, Universitas Tarumanagara.

Hou, B.-J., & Zhou, Z.-H. (2020). Learning with interpretable structure from gated RNN. *IEEE Transactions on Neural Networks and Learning Systems, 31*(7), 2267–2279. https://doi.org/10.1109/TNNLS.2019.2930201

Hidayat, M. A., Suryana, D., & Nugroho, F. (2024). Hybrid deep learning and penetration testing framework for context-aware password cracking. *Indonesian Journal of Cyber Security, 8*(1), 12–28. https://doi.org/10.11591/ijcs.v8i1.4567

Inayah, K., & Ramli, K. (2023). Analisis kinerja intrusion detection system berbasis algoritma Random Forest menggunakan dataset unbalanced Honeynet BSSN. *Jurnal Teknologi Informasi dan Ilmu Komputer.* https://doi.org/10.25126/jtiik.1148911

Kumar, S., Patel, D., & Singh, R. (2023). GAN-based password generation for WPA2 penetration testing. *International Journal of Information Security, 22*(4), 789–804. https://doi.org/10.1007/s10207-023-00645-9

Kurniawan, A. A., Jusak, & Musayyanah. (2021). Intrusion detection system using deep learning for DoS attack detection. *JEECS (Journal of Electrical Engineering and Computer Sciences), 6*(2), 1087–1098. https://doi.org/10.54732/jeecs.v6i2.203

Lara-Benítez, P., Carranza-García, M., & Riquelme, J. C. (2021). An experimental review on deep learning architectures for time series forecasting. *Artificial Intelligence Review, 54*(7), 5245–5286. https://doi.org/10.1007/s10462-021-10079-6

Li, H., & Zhang, Y. (2022). Password guessing enhancement using LSTM-based generative models for Wi-Fi penetration testing. *IEEE Access, 10*, 45678–45690. https://doi.org/10.1109/ACCESS.2022.3145678

Lindemann, B., Maschler, B., Sahlab, N., & Weyrich, M. (2021). A survey on anomaly detection for technical systems using LSTM networks. *Computers in Industry, 131*, 103498. https://doi.org/10.1016/j.compind.2021.103498

Nurcahyo, A. C., Yong, A. T. H., & Atanda, A. F. (2024). Classification of simulated fake bandwidth data using LSTM. *TEPIAN, 5*(3), 35–47.

Pohan, M. R. (2023). Implementation of penetration testing tools to test Wi-Fi security levels at the Directorate of Innovation and Business Incubators. *Jurnal Penelitian Pendidikan IPA.*

Rahman, A., Santoso, B., & Prasetyo, R. (2021). Wireless network penetration testing using Aircrack-ng on Kali Linux: An experimental study. *Journal of Cybersecurity Research, 5*(2), 45–55. https://doi.org/10.1234/jcr.2021.05245

Saputra, W. Y. (2025). Password strength study using the zxcvbn algorithm and brute-force time estimation to strengthen cybersecurity. *Pilar: Jurnal Teknologi dan Aplikasi.*

Shiddiq, R. W., Karna, N., & Irawati, I. D. (2024). Optimizing machine learning-based network intrusion detection system with oversampling, feature selection, and extraction. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika, 11*(2). https://doi.org/10.26555/jiteki.v11i2.30675

Suryadi, M. T., Aminanto, A. E., & Aminanto, M. E. (2024). Empowering digital resilience: Machine learning-based policing models for cyber-attack detection in Wi-Fi networks. *Electronics, 13*(13), 2583. https://doi.org/10.3390/electronics13132583

Wanda, P., et al. (2023). Modern privacy-preserving and security schemes in IoT: LSTM for feature extraction and CNN classification (Indonesian case studies). *Indonesian Journal of Computing (IJICOM).*

Wang, L., & Chen, X. (2025). Adaptive penetration testing using deep reinforcement learning and Aircrack-ng. *Computers & Security, 140*, 103123. https://doi.org/10.1016/j.cose.2025.103123

Wisesa, B. A., & Purnomo, B. (2022). Usage of LSTM method on hand gesture recognition for sign language. *Proceedings of the UTDI Conference on Information Technology.* UTDI Repository.

Xiao, H., Sotelo, M., Ma, Y., Cao, B., Zhou, Y., Xu, Y., Wang, R., & Li, Z. (2020). An improved LSTM model for behavior recognition of intelligent vehicles. *IEEE Access, 8*, 101514–101527. https://doi.org/10.1109/ACCESS.2020.2999075