TimeShield Algorithm: A Delay-Based Defense Against Evil Twin Attacks in Wi-Fi Networks

Huu Ton LE[1], Nhat Quang DOAN[2], Anh Tuan GIANG[2*], Hoang Ha NGUYEN[2], Anthony BUSSON[3]
[1]*CMC University, Hanoi, Vietnam*
[2]*University of Science and Technology of Hanoi,*
Vietnam Academy of Science and Technology,
18 Hoang Quoc Viet, Hanoi, Vietnam
[3]*Laboratoire de l'informatique du Parallelisme, UMR 5668*
University Lyon 1- ENS de Lyon- UCBL- CNRS- Inria, Lyon, France
*Email: giang-anh.tuan@usth.edu.vn (Corresponding Author)

Abstract—**This paper proposes TimeShield, a novel algorithm developed to counteract "evil twin attacks" in Wi-Fi networks by introducing a controlled time delay (Δ_t) to prevent unauthorized disassociation attempts. The TimeShield algorithm aims to enhance network resilience by strategically managing delay intervals to preserve legitimate connections while thwarting malicious disconnections. To evaluate the algorithm's performance and adaptability, we conducted extensive simulations using the NS-3 network simulator with a variety of traffic rates from 1 Mbps up to 300 Mbps. Our NS-3 simulation results demonstrate that TimeShield achieves a consistent success prevention rate of up to 99% with the appropriate time delay value, showcasing its effectiveness in maintaining network stability and security. With increasing traffic rates, the algorithm requires a more computational resource to handle the shorter delay time. This study highlights TimeShield's scalability and potential as a flexible solution for wireless network security.**

Index Terms—**Evil Twin Rogue Access Point Attack, WiFi networks, Delay-based, Intrusion Detection System, NS-3 simulation**

I. Introduction

Wireless (Wi-Fi) networks are ubiquitous in modern communication, providing convenient access to the internet for devices ranging from laptops to smartphones. However, the growing dependence on these networks has raised significant security concerns, particularly regarding the protection of sensitive information transmitted over the air. Among the various threats that compromise the integrity and confidentiality of Wi-Fi communications, the evil twin attack stands out as one of the most dangerous. An evil twin attack occurs when an attacker sets up a rogue access point (AP) that mimics the legitimate Wi-Fi network, tricking users into connecting to the malicious AP. Once connected, the attacker can intercept sensitive information, including login credentials, personal data, and even manipulate communications. This type of attack is particularly concerning in public Wi-Fi environments, where users may be less cautious about verifying network authenticity.

The scale of this issue is significant. Public Wi-Fi networks, especially in cafes, airports, and hotels, are prime targets, with millions of unsuspecting users at risk annually. A large part of public Wi-Fi users do not check the legitimacy of the network before connecting, leaving them vulnerable to attacks. The economic impact of recovering from such attacks is also substantial. According to Kaspersky 2018 reports [1], the average cost for a business to recover from a cyberattack, including evil twin attacks, is around $300,000 to $500,000, accounting for factors like data breaches, operational disruptions, and legal consequences. For small businesses, these costs could be devastating. Globally, the FBI's Internet Crime Complaint Center [2] reported that cybercrime resulted in 10.3 billion in losses in 2022, a figure that includes Wi-Fi attacks like the evil twin. This highlights the financial burden and recovery efforts required to mitigate these types of threats.

Despite the presence of robust encryption protocols such as WPA2 and WPA3, these attacks remain effective because they exploit user trust rather than protocol weaknesses. Attackers can easily create convincing fake networks that fool users into connecting, bypassing security mechanisms. As attackers become more sophisticated, it is imperative to develop mechanisms that can not only detect but also prevent the evil twin attack in real time.

In this paper, we propose a novel delay-based algorithm designed to thwart evil twin attacks by introduc-

979-8-3315-7896-1/26/$31.00 ©2026 IEEE

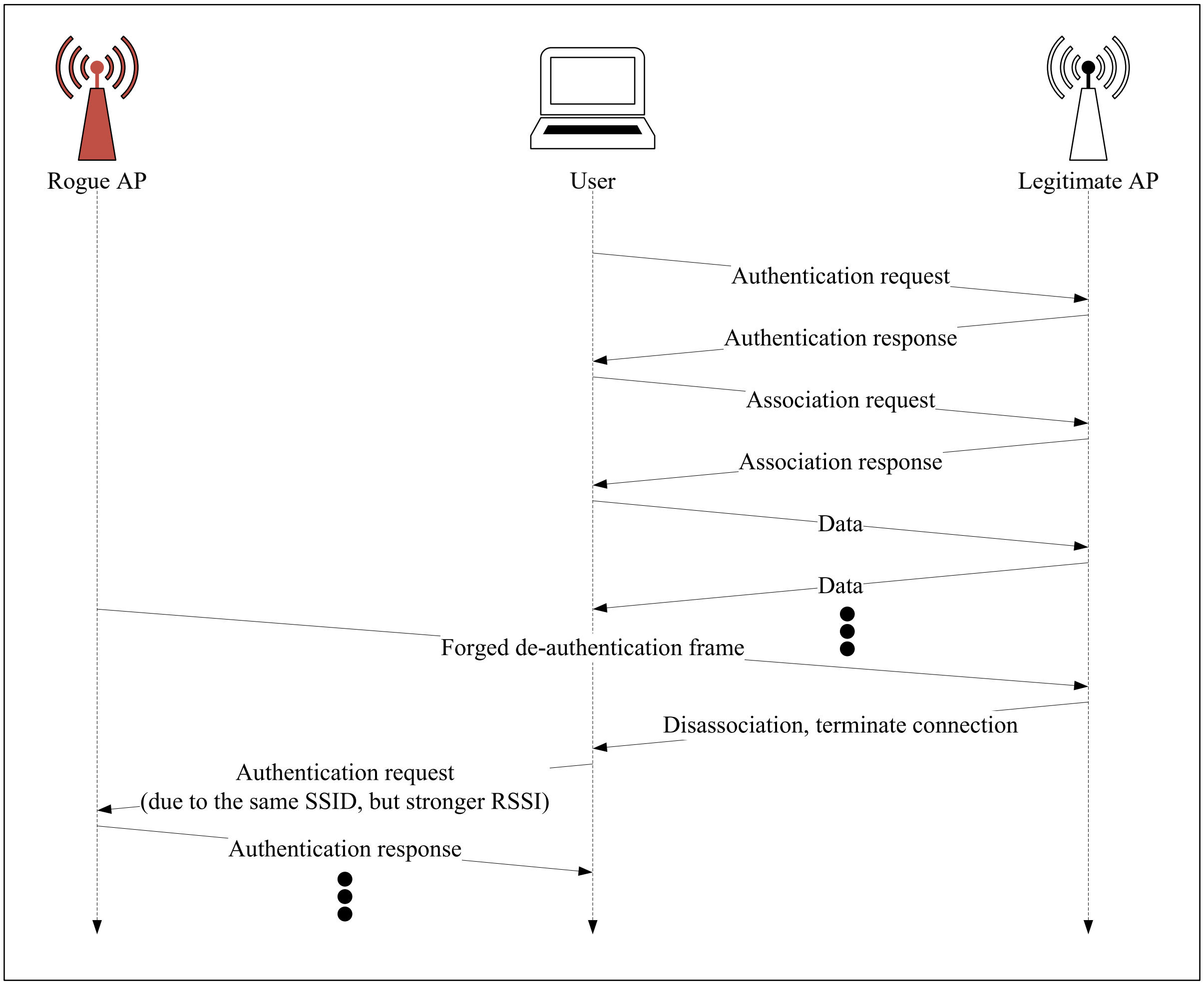


Fig. 1: A rogue AP plays man-in-the-middle requesting legitimate AP to tear down connection to an active user and redirecting that user to it by broadcasting stronger RSSI (Received Signal Strength Indicator).

ing a controlled delay time. Our approach introduces extra time adapted to the legitimate AP's communication flow rate. Through a combination of protocol analysis and network simulation, this mechanism offers a comprehensive solution to a critical vulnerability in Wi-Fi security. The rest of the paper is organized as follows: in Section II, the technical details of the Evil Twin attack and some related works are explained and summarised. The TimeShield algorithm is proposed in Section III. Then, we validate our algorithm by NS-3 simulation in Section IV. Finally, we conclude our paper in Section V.

II. Background and related works

Evil Twin Attack Definition and Mechanism

An evil twin attack is a type of cyberattack in which an attacker sets up a rogue Wi-Fi access point (AP) that impersonates a legitimate AP. This rogue AP, often appearing indistinguishable from the authentic network in name (SSID) and configuration, is strategically set up to deceive users into connecting, thereby gaining unauthorized access to their data or potentially compromising their devices. Such attacks primarily target unsecured and public Wi-Fi networks, where user devices are more likely to auto-connect to familiar network names.

Attack Execution

The process of executing an evil twin attack typically involves a series of processes. First, attackers identify and clone a target network's SSID and security settings to create a duplicate. This step is referred to as the AP Cloning. After that, attackers may actively de-authenticate users connected to the legitimate network, forcing them to reconnect to the attacker's rogue AP. Users are commonly unaware of the switch since the network name appears identical. Once connected, any unencrypted data transmitted by the user (such as credentials, emails, and private messages) is intercepted. Attackers may also inject malicious content, redirect traffic to phishing websites, or install malware on the victim's device.

Figure 1 describes in detail the technique to de-authenticate an active user and redirect the user to the attacker's rogue AP. In the beginning, the user follows the standard procedure to perform the connection to the legitimate AP (including a request for authentication and a request for association). After being connected, the data will be exchanged between the two parties. The attacker, after monitoring for a while, now has sufficient information to forge a fake de-authentication frame. He sends this fake frame to the legitimate AP. The legitimate AP thinks this frame has been sent by its active user and, thus, simply performs the disassociation process. Once the user is disconnected, it will be redirected to the attacker's rogue AP since the attacker maximizes the transmission power of the rogue AP so that the user, by default, will reconnect to the rogue AP (in case of the same SSID, the one who has stronger SNR will be chosen).

The consequences of evil twin attacks are severe, affecting both individual users and enterprises, thus leading to some notable impacts:

- Data Theft: Sensitive user information, including login credentials, financial data, and personal files, can be intercepted and stolen.
- Device Compromise: Attackers can exploit vulnerabilities in the device's software, leading to malware installation and persistent access.
- Financial Loss: On a broader scale, enterprises impacted by this attack type may incur significant recovery costs, including data restoration, incident response, and potential legal liabilities.

Related works

Over the past decade, researchers and practitioners have proposed numerous strategies to counteract evil twin attacks due to their highly deceptive and damaging nature. These countermeasures can generally be grouped into four main categories: authentication-based solutions, rogue access point (RAP) detection, machine learning and artificial intelligence (AI)-driven techniques, and enhancements to wireless encryption protocols.

a) Authentication-Based Solutions:: One of the earliest lines of defense involves strengthening the authentication mechanisms between users and access points. For instance, protocols like WPA3 and IEEE 802.1X offer mutual authentication to ensure both the client and the AP can verify each other's identities. Despite these advances, evil twin attacks continue to thrive as attackers often exploit the user side of the connection, where trust in network names (SSIDs) overrides strict authentication checks. Huang et al. [3] introduced Phyfinatt, a framework that undermines physical layer fingerprinting by demonstrating that even sophisticated physical-layer-based authentication systems can be bypassed.

b) Rogue Access Point Detection Techniques:: Another widely studied approach is detecting rogue APs using network-side features. Lin et al. [4], Pu et al. [5] and [11] explored client-agnostic and spatial techniques to identify unauthorized APs, including fingerprinting based on MAC addresses, signal strength patterns, and transmission behavior. These solutions often involve monitoring from a central controller or distributed sensor nodes to spot anomalous behavior indicative of rogue APs. However, they may require additional infrastructure or cannot function effectively in environments with high device mobility or signal interference.

c) Machine Learning and AI-Based Methods:: With the advancement of AI, recent studies have applied machine learning models to identify and classify malicious behavior in wireless environments. Shakya et al. [6] implemented a reinforcement learning model to adaptively identify threats in Wi-Fi environments, while da Silva et al. [8] used the AWID3 dataset to train classifiers that can distinguish between normal and attack traffic. Similarly, Pang et al. [7] investigated adversarial attacks and poisoning in the model training process, exposing vulnerabilities in AI-based detection schemes. Although promising, these models require large datasets and careful tuning to avoid false positives and maintain real-time performance.

d) Encryption and Protocol Enhancements:: Enhancing existing wireless communication protocols to add more robust cryptographic protection has also been a focus. For example, Nguyen et al. [9] proposed a lightweight envelope-based encryption method tailored for wireless LANs, while Shrivastava et al. [10] introduced EvilScout, a detection and mitigation framework built for SDN-enabled Wi-Fi networks. These enhancements aim to make the underlying protocol stack more resistant to spoofing and injection attacks, but often require firmware or hardware updates not feasible in legacy systems.

In summary, while various approaches offer partial solutions to mitigate evil twin attacks, each comes with its own trade-offs. Authentication mechanisms are only as strong as the user behavior they depend on. Detection mechanisms can be accurate but may suffer from latency or resource overhead. AI-based systems bring adaptability but introduce new vectors for adversarial manipulation. Finally, protocol-level enhancements improve baseline security but often face deployment challenges. This context motivates the development of lightweight and adaptive strategies such as the proposed TimeShield algorithm, which targets the attack execution phase with minimal reliance on client-side modification or external infrastructure.

III. TimeShield Algorithm

In this section, we propose the TimeShield Algorithm, a simple, adaptive, and thus effective Wifi MAC-de-authentication attack prevention algorithm. The fundamental idea of this algorithm is that instead of disassociation immediately right after having received the MAC de-authentication frame, the AP adds an extra waiting time and only performs the disassociation process once this time has expired. In our algorithm, the amount of extra time adapts to the individual data communication flow rate.

Algorithm 1 The TimeShield Algorithm

```
Listening;
pkt = ReceivingPkt (dest i);
rate = UpdateRate (dest i);
if pkt $\neq$ de_auth then
    goto Listening;
else
    $\Delta_t$ = ComputeDeltaT (rate);
    while $\Delta_t > 0$ do
        if (ReceivingPkt (dest i) $\neq$ de_auth) then
            goto Listening;
        else
            $\Delta_t - -$;
        end if
    end while
    disassociation ();
end if
```

The pseudo-code of the TimeShield Algorithm is given in the Algorithm 1. First, the AP is in the listening state, where it is ready to receive incoming packets. Once it receives a packet from a destination, let's say $dest_i$, the AP updates the individual flow rate of the $dest_i$ in case the received frame was not a MAC de-authentication frame. In case the received frame was a MAC de-authentication frame, the AP, based on the individual flow rate, computes the Δ_t to wait before disassociation with the client. During this Δ_t interval, if the AP receives another frame coming from the same destination as the MAC de-authentication frame's destination that is not the MAC de-authentication frame, then the AP considers the previous MAC de-authentication frame as a faked one. Thus, the disassociation process is canceled, and the AP keeps communication with the client. An illustration of this algorithm is also provided in Figure 2

A. Computing the minimal Δ_t

As the TimeShield algorithm requires to compute an extra delay interval referred to as the Δ_t, a natural question is: what is the good value for this delay? A straightforward answer to this question is that Δ_t should be the minimum time for the algorithm to reach a specific threshold of success rate in the prevention of attacks, for example, 90% of success prevention. In the following Theorem, we assume that the individual data flow follows a Poisson distribution with a rate of λ.

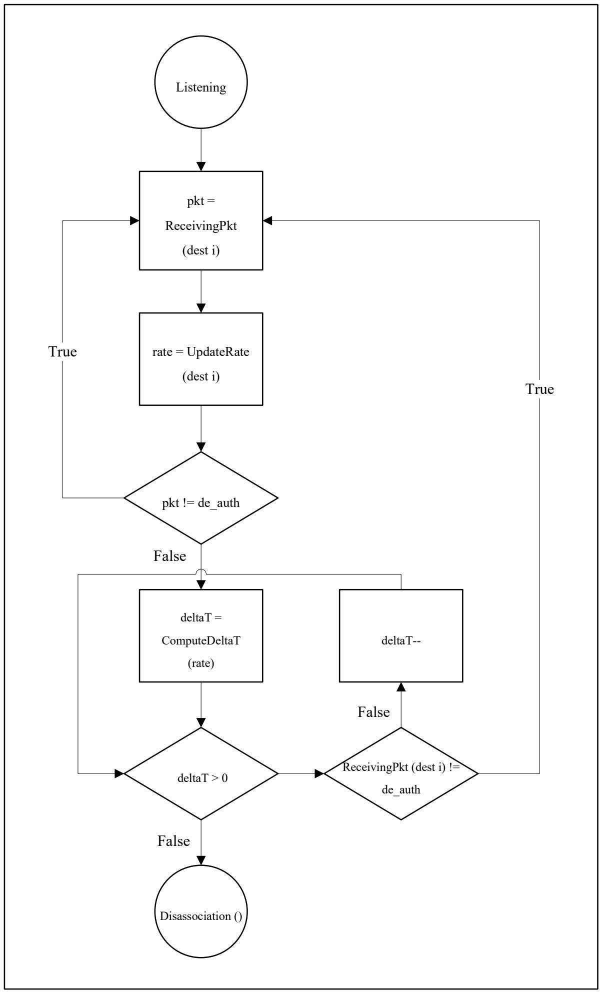


Fig. 2: A flowchart illustration of the TimeShield Algorithm.

Theorem 1. *The minimal or smallest Δ_t that allows the TimeShield algorithm to prevent up to x percentage is as follows:*

$$\Delta_t^* = \frac{-\ln(1-x)}{\lambda} \qquad (1)$$

Proof. The computing of minimal Δ_t is a classic Poisson problem. The main idea of the TimeShield algorithm is to use the real packet against the fake MAC de-authentication packet. Let us say the t_0 is the time that the user receives the fake packet. From that moment, a timer of a length Δ_t is started. Our problem can be modeled as finding the smallest Δ_t, such as

Parameters	**Value**
Legitimate AP tx power	17 dBm
Rogue AP tx power	23 dBm
Client tx power	17 dBm
Transmission rate	1 Mbps, 10 Mbps 100 Mbps, 300 Mbps
Number of samples	100,000
Simulation time	10 seconds
Wi-Fi standard	IEEE 802.11ac IEEE 802.11n

TABLE I: Parameters used in simulations

the probability of at least one more packet arriving during $[0, \Delta_t]$ is greater than x percent. Since we have already assumed that the packet arrival rate (λ) follows a Poisson distribution. Thus, the probability of having no arrival during the interval of Δ_t is:

$$\mathbb{P}(\text{ no arrival in } \Delta_t) = \mathrm{e}^{-\lambda \Delta_t}.$$

Therefore, the probability of having at least one packet during this interval is:

$$\mathbb{P}(\text{ at least one arrival in } \Delta_t) = 1 - \mathrm{e}^{-\lambda \Delta_t}.$$

Due to the fact that we want to achieve up to x percent of attack prevention, then we have:

$$\begin{aligned} 1 - \mathrm{e}^{-\lambda \Delta_t} &\geq x \\ \Rightarrow \mathrm{e}^{-\lambda \Delta_t} &\leq 1 - x \\ \Rightarrow \Delta_t &\geq -\frac{1}{\lambda} \ln(1 - x). \end{aligned}$$

Thus, a minimal Δ_t^* that meets the target success rate is as claimed. □

IV. Simulation Results and Discussions

Simulation scenarios

To validate the usefulness of the TimeShield algorithm, we perform simulations in NS-3 [12] with this algorithm being implemented on the legitimate AP. The validation scenario is simple. We simulate one legitimate AP, one rogue AP, and a client. The transmission power of the rogue AP is set higher compare to the legitimate. The communication rate between the client and the legitimate AP varies from 1 Mbps to 300 Mbps. The rogue AP tries to transmit the forged MAC deauthentication frame at a random time. We also collect $100\,000$ samples per data point and vary the value of Δ_T to see the effect of our proposed algorithm. The details of simulation parameters are given in Table I.

Results and discussions

We plot the simulation results in Figure 3 and Figure 4 in case of high rate transmission and low rate transmission, respectively. In both simulation scenarios, we can observe that the TimeShield algorithm can prevent the rogue AP from interrupting the active user from the legitimate AP up to 99%. For the high-rate transmission case, the extra time required to reach that prevention rate, as they are shown in Figure 3a, and Figure 3b, is 370, and 140 microseconds. In the low-rate transmission case, the amount of required time is much longer, 3500 and 35000 microseconds, as they are shown in Figure 4a, and Figure 4b.

Despite a very high rate of preventing this kind of disassociation attack, the simulation results indicate that in the high-rate transmission cases, the algorithm still effectively prevents disassociation around 350 microseconds. However, the range narrows, showing that higher speeds demand shorter time delays to balance performance and protection. Especially in Figure 3a, the algorithm operates effectively around 140 microseconds, highlighting the need for rapid processing and minimal delays in high-speed networks. This range suggests a trade-off, as achieving prevention requires more precise control over Δ_t to avoid network disruptions.

V. Conclusion

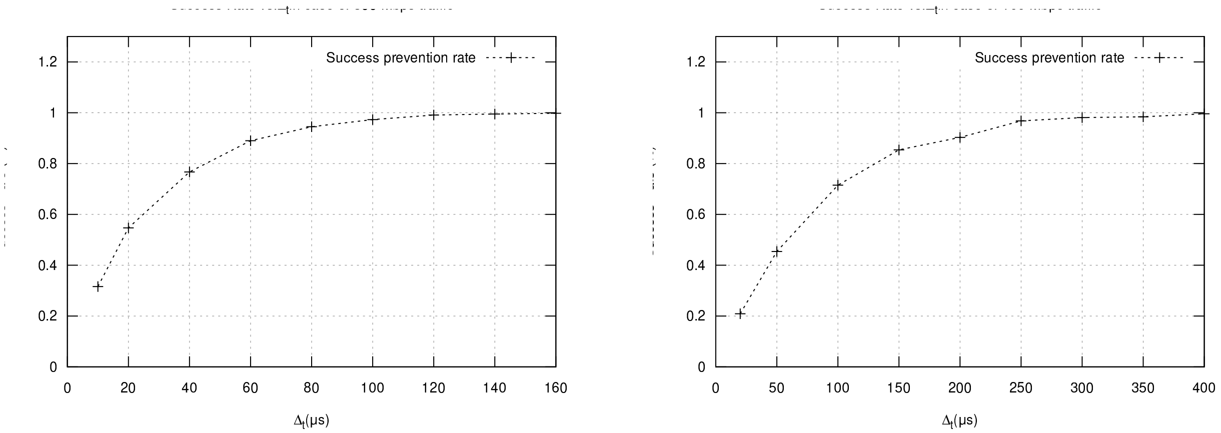
In this study, we presented the TimeShield algorithm, a novel approach designed to counteract the effects of "evil twin attacks" in Wi-Fi networks by introducing a controlled time delay, Δ_t, to prevent malicious disassociation. Through extensive testing across multiple traffic rates, 1 Mbps to 300 Mbps, we analyzed the algorithm's effectiveness in maintaining network integrity against unauthorized interruptions. Our findings demonstrate that the TimeShield algorithm effectively maintains a consistent success prevention rate across varying network speeds with the adaptive delay, indicating strong resilience against disassociation attempts. However, as traffic rates increase, the Δ_t interval becomes much shorter, which requires a heavy computational effort for the AP. This trade-off is worthy to note due to the limitation of computational resources of the AP. A natural way to extend this research is to propose a more general mathematical framework that allows one to evaluate the performance of the TimeShield algorithm with different packet arrival rate distribution and implement this algorithm on a real testbed system.

References

[1] Paraskevas, A. (2022). Cybersecurity in travel and tourism: a risk-based approach. In Handbook of e-Tourism (pp. 1605–1628). Cham: Springer International Publishing.

[2] Kuzior, A., Tiutiunyk, I., Zielińska, A., & Kelemen, R. (2024). Cybersecurity and cybercrime: Current trends and threats. Journal of International Studies (2071-8330), 17(2).

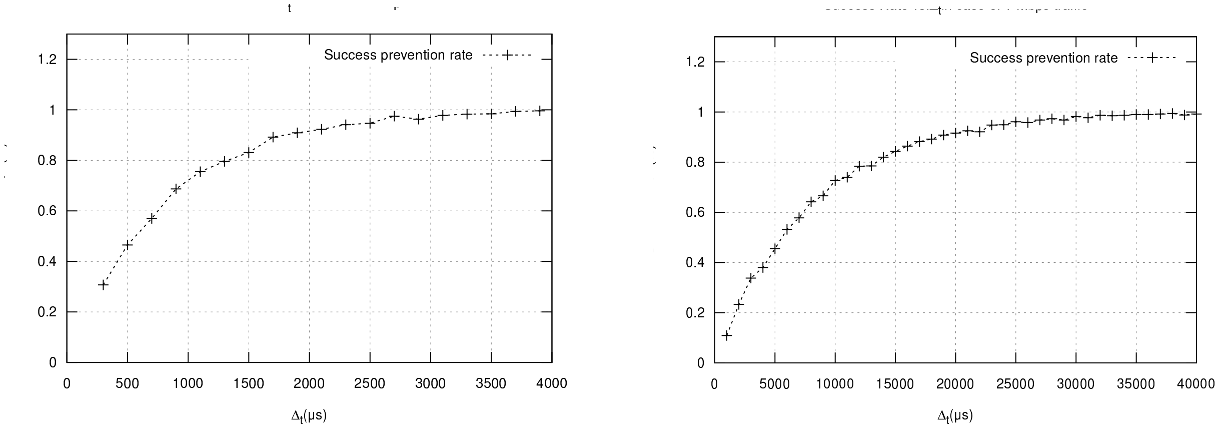
[3] Huang, J., Liu, B., Miao, C., Zhang, X., Liu, J., Su, L., ... & Gu, Y. (2023). Phyfinatt: An undetectable attack framework against phy layer fingerprint-based wifi authentication. IEEE Transactions on Mobile Computing.



(a) 300 Mbps transmission rate between the client and the AP.

(b) 100 Mbps transmission rate between the client and the AP.

Fig. 3: Success preventing probability in case of high rate communication.



(a) 10 Mbps transmission rate between the client and the AP.

(b) 1 Mbps transmission rate between the client and the AP.

Fig. 4: Success preventing probability in case of low rate communication.

[4] Lin, Y., Gao, Y., Li, B., & Dong, W. (2022). Detecting rogue access points using client-agnostic wireless fingerprints. ACM Transactions on Sensor Networks, 19(1), 1–25.

[5] Pu, Q., Ng, J. K. Y., Zhou, M., & Wang, J. (2021). A joint rogue access point localization and outlier detection scheme leveraging sparse recovery technique. IEEE Transactions on Vehicular Technology, 70(2), 1866–1877.

[6] Shakya, V., Choudhary, J., & Singh, D. P. (2023). An advanced actor critic deep reinforcement learning technique for gamification of WiFi environment. Wireless Networks, 1–18.

[7] da Silva, L. M., Andreghetti, V. M., Romero, R. A. F., & Branco, K. R. L. J. C. (2023, October). Analysis and Identification of Evil Twin Attack through Data Science Techniques Using AWID3 Dataset. In Proceedings of the 6th International Conference on Machine Learning and Machine Intelligence (pp. 128–135).

[8] Pang, R., Shen, H., Zhang, X., Ji, S., Vorobeychik, Y., Luo, X., ... & Wang, T. (2020, October). A tale of evil twins: Adversarial inputs versus poisoned models. In Proceedings of the 2020 ACM SIGSAC conference on computer and communications security (pp. 85–99).

[9] Nguyen, T. N., Tran, B. N., & Nguyen, D. H. (2008, August). A lightweight solution for Wireless LAN: Letter-envelop protocol. In 2008 Third International Conference on Communications and Networking in China (pp. 17–21). IEEE.

[10] Shrivastava, P., Jamal, M. S., & Kataoka, K. (2020). EvilScout: Detection and mitigation of evil twin attack in SDN enabled WiFi. IEEE Transactions on Network and Service Management, 17(1), 89–102.

[11] Giang, A. T., Tran, H. T., Le, H. T., Doan, N. Q., & Nguyen, M. H. (2022). Jamming attack in vehicular networks: adaptively probabilistic channel surfing scheme. Wireless Communications and Mobile Computing, 2022(1), 3884761.

[12] NS3 Development Team. (2024). ns-3 [Computer software]. Available from https://www.nsnam.org