Detecting End-Point (EP) Man-In-The-Middle (MITM) Attack based on ARP Analysis: A Machine Learning Approach

Jerry John Kponyo[1], Justice Owusu Agyemang[2] and Griffith Selorm Klogo[3]

[1,2,3]Faculty of Electrical/Computer Engineering, Kwame Nkrumah University of Science and Technology, Ghana

***Abstract*:** End-Point (EP) Man-In-The-Middle (MITM) attack is a well-known threat in computer security. This attack targets the flow of information between endpoints. An attacker is able to eavesdrop on the communication between two targets and can either perform active or passive monitoring; this affects the confidentiality and integrity of the data flow. Several techniques have been developed by researchers to address this kind of attack. With the current emergence of machine learning (ML) models, we explore the possibility of applying ML in EP MITM detection. Our detection technique is based on Address Resolution Protocol (ARP) analysis. The technique combines signal processing and machine learning in detecting EP MITM attack. We evaluated the accuracy of the proposed technique using linear-based ML classification models. The technique proved itself to be efficient by achieving a detection accuracy of 99.72%.

***Keywords*:** ARP, Internet Protocol, MITM, Machine Learning.

1. Introduction

End-Point (EP) Man-In-The-Middle (MITM) is an eavesdropping kind of attack, where in a communication session between two client devices **A** and **B**, the attacker deceives **A** by pretending to be **B**. This enables the attacker to read or modify messages (passive/active monitoring) sent from **A** to **B** (shown in Figure 1). The current implementation of the Address Resolution Protocol (ARP) is 'stateless', hence making it possible for this kind of attack to occur. ARP is a protocol used by the data link layer (layer 2) to map Internet Protocol (IP) Addresses to Media Access Control (MAC) addresses [1]. Before data encapsulation in a data link layer frame, the host sending the packet needs to know the recipient's MAC address. Given the IP address of a host, to find its MAC address, the source node broadcasts an ARP request packet which asks about the MAC address of the owner of the IP address. This request is received by all nodes inside the Local Area Network (LAN). The node that owns this IP address replies with its MAC address (unicast) [2].

ARP is a stateless protocol hence it accepts ARP replies without considering if an ARP request was sent [2]. This weakness can be exploited by an attacker to initiate MITM attack. In reference to Figure 1, the attacker after initiating MITM attack becomes the next hop for the exchange of information between client **A** and **B**. Data flow between the two endpoints can be intercepted and read or modified. This affects the integrity and confidentiality of the transit data. A denial of service (DoS) attack can occur if the attack drops the received packet without forwarding it to the appropriate destination.

Figure 1. MITM Attack

This research work combines ML and signal processing by analysing ARP packets to create a detection engine (classifier) for MITM attack detection. The rest of the paper is organized as follows: Section 2 reviews related works; Section 3 describes our proposed solution. The approach used in the MITM detection is described in Section 4. Section 5 discusses the results and Section 6 is the conclusion.

2. Review of Related Works

Although MITM attack has been known for some time, it is still considered a significant threat [3, 4], and have gained much attention over the past years. This can be attributed to the fact that the attack is easy to achieve and very difficult to detect.

A number of techniques have been proposed by researchers in detecting and defending against this security threat. Intrusion Detection Systems (IDs) have been used to detect and prevent MITM in Wired Local Area Networks (LANs) [5]. A unicast ARP request has been proposed as a replacement for broadcast ARP request [6]. Encryption-based ARP that utilizes public key cryptography has also been proposed [7, 8]. An approach to prevent ARP cache poisoning by monitoring Domain Name Host Configuration (DHCP) acknowledgment messages has also been proposed [9]. Other researchers have proposed a voting-based ARP spoofing resistant protocol to address EP MITM attack [10, 11, 12].

Some proposed techniques are complex to implement on Low-Embedded devices and also others involve the change in the entire protocol. Recently, Internet Control Message Protocol (ICMP) analysis has been proposed as a means to detect MITM attacks in LANs [13]. This is a very good technique that applies signal processing in detecting MITM attacks. A burst of ICMP request packets is sent to an endpoint. The payload sizes of the ICMP request packets are

modulated according to an excitation signal. As a result, the impulse response extracted from the Round-Trip Time (RTT) is used to model the network environment in the perspective of two communicating hosts. When a third-party intercepts traffic, the harmonic composition of the impulse response between the host changes significantly; which formed the basis of their MITM detection.

Even though the ICMP echo-analysis is a good MITM detection mechanism, it has a few flaws. Since the detection is based on flooding an endpoint with ICMP request packets, it can easily be picked up as a Denial of Service (DoS) attack. Besides ICMP packets can be blocked by the attacker which defeats the detection mechanism. Furthermore, an ICMP request/reply packet is 98 bytes each [14]. Even though the size of the packet seems insignificant, a burst of such requests/replies can have a significant effect on the systems performance.

3. Proposed Solution

Our proposed solution is based on the Address Resolution Protocol (ARP) analysis. Our technique addresses the limitations of using ICMP echo-analysis in EP MITM detection. We chose ARP because it has a minimum packet size of 42 bytes (without padding; shown in Figure 2) and a maximum packet size of 60 bytes (with padding; shown in Figure 3) for each ARP request and response. Besides, ARP cannot be blocked by an attacker since it is a default protocol used in mapping IP addresses to MAC addresses. We modulate a burst of ARP requests using a maximum length sequence (MLS) [15] which is a pseudorandom binary sequence. We determine the RTTs based on the ARP requests and responses. To address the stateless nature of ARP, we modified the address resolution protocol to include a binary sequence and also a sequence number. The binary sequence and the sequence number were embedded in the ARP request and response and enabled us to match a corresponding ARP request to its response. Based on the RTTs we compute the system's response and the energy of the response. Using the system's response and energy response as features we evaluated the detection mechanism based on a number of linear machine learning models.

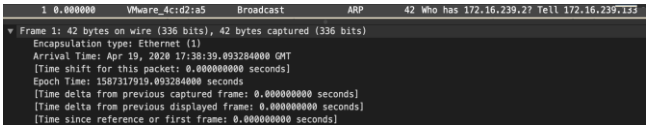

Figure 2. ARP Packet – 42 bytes (Without Padding)

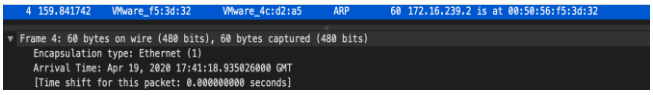

Figure 3. ARP Packet – 60 bytes (With Padding)

3.1 Research Contributions

The contribution of this study is as follows:

1. A novel method for detecting MITM attack based on ARP analysis. This method is applicable for both Wired and Wireless LANs.
2. A 'stateful' address resolution protocol.

4. Methodology

In this section we describe the MITM attack model used throughout the study. We also enumerate the attacker's requirements, attack vectors and capabilities.

4.1 Attack Scenario

Figure 4 shows the attack scenario used. Client **A** communicates wireless with Client **B**. Client **C** however intercepts the traffic between client **A** and **B** through ARP poisoning; hence traffic sent from **A** to **B** is routed through **C**. **C** can either perform active or passive monitoring.

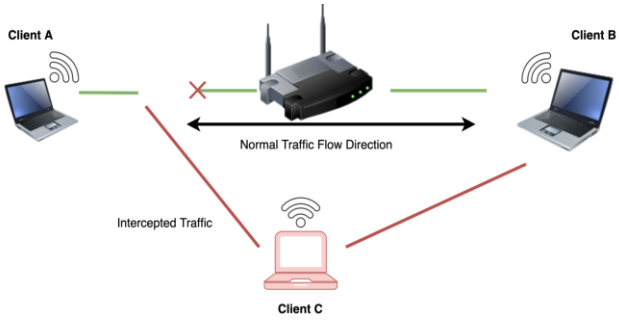

Figure 4. Attack Scenario

The detection mechanism is based on the fact the time taken for an ARP request to be sent directly from **A** to **B** will vary from the instance where an attacker intercepts the traffic.

4.2 ARP Analysis

The address resolution protocol has a default size of 42 bytes as shown in Figure 5.

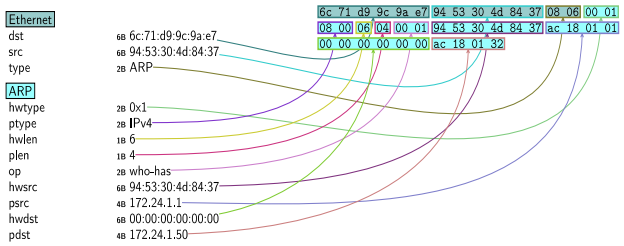

Figure 5. Sample ARP Request

An 18 bytes padding can be added to the default 42 bytes to obtain a 60 bytes frame (as shown in Figure 6).

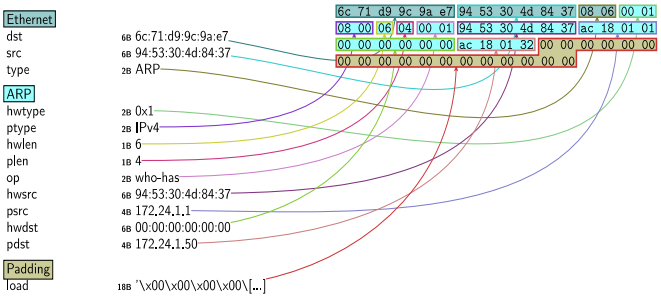

Figure 6. Sample ARP request with padding

The logical connection between Client **A** and **B** is modelled as a Linear Time Invariant (LTI) system as shown in Figure 7.

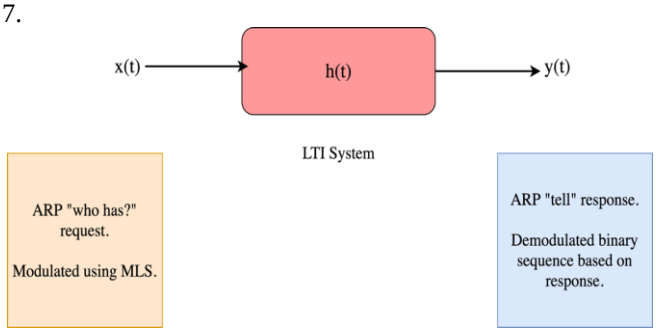

Figure 7. Channel Modelling

We first determined the impulse response of the system by modulating a burst of ARP request packets with a maximum length sequence (MLS). MLS are generated using maximal linear feedback shift registers. The last byte of the padding is encoded with the bit value generated from the MLS. A random sequence number is encoded in the 8 bytes that precedes the last byte. A sample ARP request and reply is shown in Figure 8 and 9 respectively.

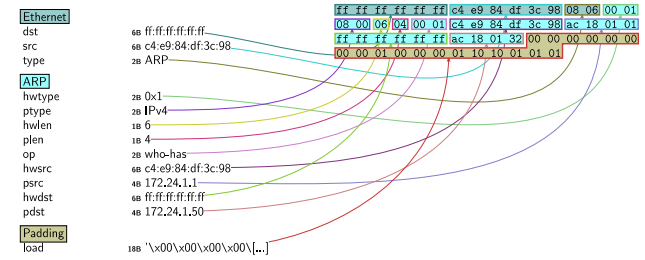

Figure 8. Custom ARP request packet

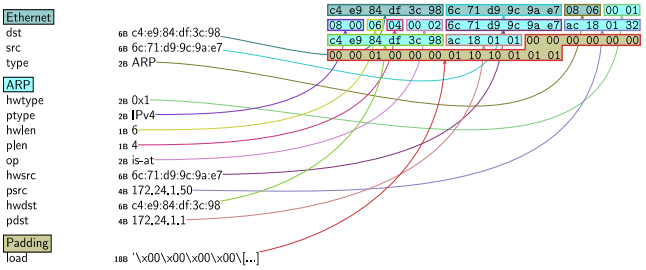

Figure 9. Custom ARP reply packet

At point **A** we were able to determine the RTTs of the burst of ARP requests based on the sequence numbers encoded in the padding payload. Using the RTT for each ARP request/reply packet, we were able to characterize the harmonic response of the channel. Using Parseval's theorem,

$$E_h = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{Y[n]}{X[n]}\right]^2 \quad (1)$$

we computed the energy of the impulse response of the channel; where $\left[\frac{Y[n]}{X[n]}\right]$ is the transfer function of the system's impulse response.

After determining the harmonic composition of the channel in the normal state, we also determine the system impulse response and energy in the MITM attack state. Figure 10 shows a graph showing the binary sequence and their corresponding RTTs.

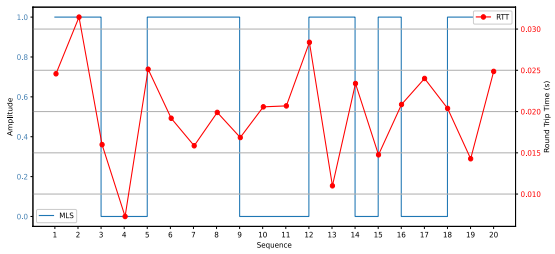

Figure 10. Round-Trip Time

Using the mean of the RTTs together with the energy of the system's impulse response as a feature vectors, we built a detection classifier engine using linear-based machine learning (ML) classification models.

5. Results and Discussion

We evaluated the proposed technique using eight (8) linear-based ML classification models; LinearSVC, SVC, KNN, Decision Tree, Logistic Regression, Random Forest, Gradient Boosting and Gaussian Naive Bayes. These supervised learning models have been applied in other areas of research such as the mitigation of denial of service (DoS) attacks [16] and anomaly-based intrusion detection systems [17]. The dataset contained 5,300 rows of the feature vectors. 80% of the dataset was used in training and the performance of each model was evaluated on the remaining 20%. Figures 10-14 are the confusion matrices of the linear-based models used in this study. The confusion matrix gives a visual representation of the performance of each linear-based ML algorithm used. Each confusion matrix consists of 3 rows and 3 columns. The cells with the green colour represent the number and percentage of the test dataset that were classified correctly. The grey cells also represent the number and percentage of the test data that were misclassified. The deep grey cells give the percentage of correctly classified and misclassified labels. The dark cells represent the overall number of the datasets together with the percentage classification accuracy.

Figure 11a. Linear SVC

From the confusion matrices, Linear SVC (Figure 11a) and Gaussian Naive Bayes (Figure 14b) have a classification accuracy of 99.72% with a misclassification of 0.28%. SVC (Figure 11b) and Logistic Regression (Figure 13a) have an accuracy of 99.62%. The percentage accuracy of Random Forest (Figure 13b) is 99.44%. An accuracy of 99.34% was produced by KNN (Figure 12a), Decision Tree (Figure 12b) and Gradient Boosting classifiers (Figure 14a).

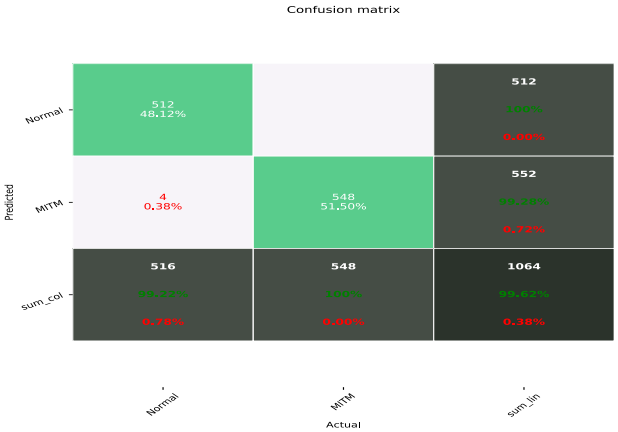

Figure 11b. SVC

Figure 12a. KNN

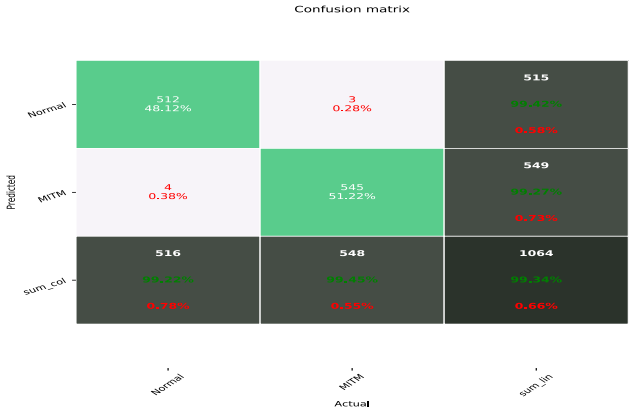

Figure 12b. Decision Tree

Figure 13a. Logistic Regression

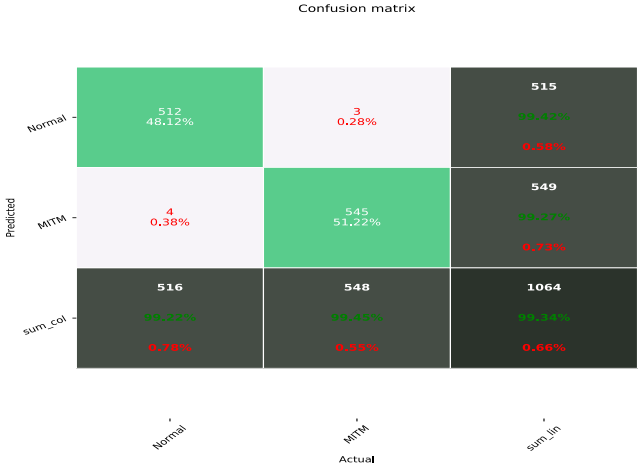

Figure 13b. Random Forest

Figure 14a. Gradient Boosting

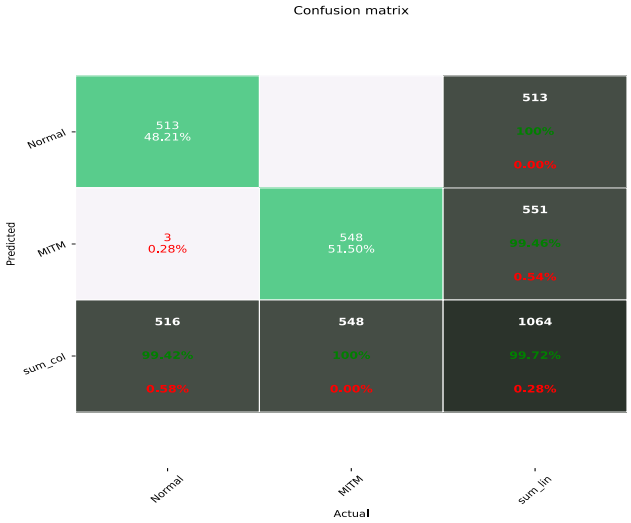

Figure 14b. Gaussian Naïve Bayes

A summary of the percentage accuracy of each model is shown in Table I.

Table I. Percentage accuracy of each model

Model	**Accuracy of Model (%)**
Linear SVC	99.72
Gaussian Naïve Bayes	99.72
SVC	99.62
Logistic Regression	99.62
Random Forest	99.44
KNN	99.34
Decision Tree	99.34
Gradient Boosting	99.34

Linear SVC and Gaussian Naive Bayes produced the highest accuracy among all the other models. All the above models had a higher accuracy as compared to [13] whose model produced an average accuracy of 93.27%.

6. Conclusion

In this study, we have proposed a detection of MITM attack based on ARP analysis. We introduced ‘statefulness’ into the address resolution protocol by adding a padding layer to the frame and encoding a bit value and a sequence number. The proposed technique achieves an accuracy of 99.72% when modeled using linear-based ML classification algorithms.

The study has shown that ARP analysis is a good technique for detecting EP MITM attack.

Future works will explore how this technique can be implemented in enterprise wired and wireless LANs since the attack scenario used was only based on a single point network.

Acknowledgement

Authors would like to acknowledge the support of MTN Ghana in providing funding for this research.

References

[1] D. C. Plummer, "An Ethernet Address Resolution Protocol", RFC 826, 1982 [Online]. Available https://tools.ietf.org/html/rfc826.

[2] G. A. Sukkar, R. Saifan, S. Khwaldeh, M. Maqableh, I. Jafar, "Address Resolution Protocol (ARP); Spoofing Attack and Proposed Defense", Communications and Network, Vol. 8, pp. 118-120, 2016.

[3] M. Conti, N. Dragoni, V. Lesyk, "A Survery of Man in the middle attacks", IEEE Communications Surveys & Tutorials", Vol. 18, No. 3, 2016.

[4] CAPEC, "Capec-94: Man-in-the-middle attack", 2019 [Online]. Available http://capec.mitre.org/data/definitions/94.html.

[5] J. Belenguer, C. T. Calafate, "A low-cost embedded IDS to monitor and prevent man-in-the-middle attacks on wired LAN environments", Proc. Int. Conf. SecureWave Emerging Secur. Inf. Sys. Technol., pp. 122-127, 2007.

[6] B. Isaac, "Secure ARP and Secure DHCP Protocols to Mitigate Security Attacks", International Journal of Network Security, vol. 8, pp. 107-118, 2009.

[7] D. Bruschi, A. Ornaghi, E. Rosti, "S-ARP: A Secure Address Resolution Protocol", Proc. 19th Annu. Comput. Secur. Appl. Conf., pp. 66-74, 2003.

[8] W. Lootah, W. Enck, P. McDaniel, "TARP: Ticket-Based Address Resolution Protocol", Computer Networks, vol. 51, pp. 4322 – 4337, 2007.

[9] R. Philip, "Secure Wireless Networks from ARP Cache Poisoning", Mater's Thesis, San Jose State University, 2007. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.646.8014&rep=rep1&type=pdf

[10] S. Y. Nam, D. Kim, J. Kim, "Enhanced ARP: Preventing ARP poisoning-based man-in-the-middle attacks", IEEE Commun. Lett., vol. 14, No. 2, pp. 187-189, 2010.

[11] S. Y. Nam, S. Juravey, S.-S. Kim, K. Choi, G. S. Choi, "Mitigating ARP Poisoning-Based Man-In-The-Middle Attack in Wired or Wireless LAN", Journal of Wireless Communications and Networks, 2012.

[12] S. Y. Nam, S. Djuraev M. Park, "Collaborative Approach to Mitigate ARP Poisoning-Based Man-In-The-Middle Attack", Comput. Netw. Vol 57, No. 18, pp 3866-38884, 2013.

[13] Y. Mirsky, N. Kalbo, Y. Elovici, A. Shabtai, "Vesper: Using Echno Analysis to Detect Man-in-the-Middle Attacks in LANs", IEEE Transactions on Information Forensics and Security, Vol. 14, No. 6, pp. 1638 – 1653, June 2019.

[14] Internet Control Message Protocol (ICMP), 2019 [Online]. Available https://tools.ietf.org/html/rfc777

[15] Maximum Length Sequence, 2019 [Online]. Available https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.max_len_seq.html.

[16] K. Ganesh Reddy, P. Santhi Thilagam, "Naïve Bayes Classifier to Mitigate DoS Attacks Severity in Ad-Hoc Networks", International Journal of Communication Networks and Information Security (IJCNIS), Vol. 12, No. 2, 2020.

[17] Z. Chiba, N. Abghour, K. Moussaid, A. El omri, M. Rida, "Intelligent and Improved Self-Adaptive Anomaly based Intrusion Detection System for Networks", International Journal of Communication Networks and Information Security (IJCNIS), Vol. 11, No. 12, 2019.