Journal of KIIT. Vol. 21, No. 5, pp. 111-120, May 31, 2023. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2023.21.5.111

MQTT 기반 IoT 환경에서 LSTM 및 슬라이딩 윈도우를 이용한 악성 트래픽 탐지 방법

이든*[1], 임승순*[2], 최선오**

Malicious Traffic Detection Method using LSTM and Sliding Window in MQTT based IoT Environment

Deun Lee*[1], SeungSoon Im*[2], and SunOh Choi**

본 연구는 정부(과학기술정보통신부)의 5G기반 스마트 센서 검증 플랫폼 구축 과제의 지원을 받아 수행된 연구임 (No. 2019-0-00135)

요 약

최근 사물인터넷(IoT) 분야의 급격한 성장과 시장의 수요가 증가하면서 일상생활과 전반적인 산업영역에서 IoT 기기의 사용이 늘어나고 있다. 그러나 많은 IoT 기기들이 보안 대책을 충분히 고려하지 않은 상태로 시장에 출시되었다. 이를 대상으로 하는 공격 기법과 다양한 악성코드들이 등장함에 따라 보안적인 측면이 중요해지고 있다. IoT 환경에서는 적은 리소스로 효율적인 통신을 지원하는 MQTT 프로토콜이 광범위하게 사용되고 있으며, MQTT 기반 IoT 환경에 대한 보안 위협이 증가할 것으로 예상된다. 본 연구에서는 IoT 환경에서 MQTT 프로토콜에 대한 악성 트래픽을 탐지하고 분류하기 위한 슬라이딩 윈도우 기반 LSTM 딥러닝 모델을 제안하였다. 기존에 공개된 MQTT 트래픽 데이터셋과 본 연구에서 IoT 환경을 구축하고 수집한 데이터셋을 바탕으로 제안한 모델을 이용하여 기존 연구보다 정확도가 개선됨을 보였다. 또한 제안된 모델을 이용하여 실시간으로 악성 MQTT 트래픽을 탐지할 수 있음을 실험을 통하여 확인하였다.

Abstract

Recently, with the rapid growth of the Internet of Things(IoT) field and increasing market demand, the use of IoT devices is increasing in daily life and overall industrial areas. However, many IoT devices have been introduced to the market without sufficient consideration of security. With the emergence of attack techniques and various malicious codes, the security aspect is becoming increasingly important. In the IoT environment, the MQTT protocol, which supports efficient communication with small resources, is widely used, and security threats to the MQTT-based IoT environment are expected to increase. In this study, we proposed a sliding window based LSTM deep learning model to detect and classify malicious traffic for the MQTT protocol in the IoT environment. Using the proposed model based on the previously published MQTT traffic dataset and the dataset collected in this study, it was shown that the accuracy was improved compared to previous studies. In addition, we showed that the malicious MQTT traffic is detected in the real time environment.

Keywords

internet of things, IoT, IoT attack, MQTT, deep learning, LSTM, sliding window

* 전북대학교 소프트웨어공학과
- ORCID[1]: https://orcid.org/0009-0004-7808-6948
- ORCID[2]: https://orcid.org/0009-0006-5442-7132
** 전북대학교 소프트웨어공학과 교수(교신저자)
- ORCID: https://orcid.org/0000-0002-0654-7109

· Received: Mar. 16, 2023, Revised: May 02, 2023, Accepted: May 05, 2023
· Corresponding Author: Sunoh Choi
Dept. of Software Engineering, Jeonju, South Korea
Tel.: +82-63-270-4784, Email: suno7@jbnu.ac.kr

Ⅰ. 서 론

사물인터넷(IoT, Internet of Things)은 통신 기술의 발전과 시장의 수요가 맞물리면서 급속도로 발전하고 있다. IoT 기기를 이용하면 편의성을 증대시킬 수 있고, 신속하고 즉각적으로 제어가 가능하며, 효율적으로 대량의 데이터를 수집할 수 있다는 장점 덕분에 다양한 산업영역과 일상생활에 IoT 장치가 널리 보급되고 있다. Transforma Insights가 2022년 발표한 글로벌 IoT 예측 보고서[1]에 의하면 2021년 말에는 약 113억 개의 활성 IoT 장치가 있었으며, 2030년에는 약 294억 개로 증가하여 연평균 성장률 12%를 기록할 것으로 예측된다. 또한 Marketsand Markets이 2022년에 발표한 보고서[2]에 따르면 IoT 솔루션 및 서비스 시장 규모가 연평균 18.8% 성장하여 2022년 약 2,431억 달러에서 2027년 약 5,750억 달러에 이를 것으로 전망된다.

일반적으로 IoT 기기는 컴퓨터에 비해서 성능과 전력이 낮고, IoT 제품을 제조하는 기업 수가 증가하면서, IoT 기기에 대한 표준화되고 통합된 경량 프로토콜의 필요성이 대두되었다. IoT 기기에는 다양한 종류의 프로토콜이 사용되지만, 세션 계층에는 MQTT, AMQP, CoAP, XMPP, DDS 등의 프로토콜이 사용된다[3]. 이 중 MQTT는 대표적인 IoT 프로토콜로 적은 리소스와 낮은 네트워크 대역폭에서도 효율적으로 동작하며, 발행/구독(publish/subscribe) 구조를 채택하여 안정적으로 대량의 기기가 통신할 수 있다는 장점이 있다. HiveMQ와 IIoT World가 진행한 산업용 IoT(IIoT, Industrial IoT)에 대한 설문조사[4]에서 응답자의 50.3%가 현재 장비를 연결하기 위해 MQTT 프로토콜을 사용한다고 답하였다. 또한 응답자의 1/3이 MQTT 프로토콜을 현재 배포된 시스템에서 사용하고 있다고 답하였으며, 다른 30%는 MQTT를 사용할 계획이 있다고 답하였다.

그러나 IoT의 급격한 성장은 보안이 결여된 많은 장치의 생산으로 이어졌다. B. Zhao, et al[5]은 136만 개의 IoT 장치와 1.4만 개의 MQTT 서버에 대한 실증적 연구를 진행하였고, 28.25%의 IoT 장치가 N-days 공격에 취약하며, 88%의 MQTT 서버에 암호 보호가 없다는 것을 확인하였다. 또한 IoT 헬스케어 기기에 관한 연구[6]에서는 보안이 적용되지 않은 기기의 OTA 업데이트시 이진 코드와 인증 암호를 탈취할 수 있음을 보였다. IoT 장치를 해킹하여 악용한 대표적인 사례로는 Mirai Botnet에 의한 DDoS 공격이 있다. IoT 장치에 대한 공격은 사이버 세계를 넘어 현실 세계까지 영향을 미쳐 막대한 피해를 입힐 수 있으며, IoT 장치가 수집하고 전송하는 정보 중에는 민감한 정보가 포함된 경우도 있다. 실제로 2021년 국내 아파트의 월패드가 해킹되어 세대간 통화를 위한 카메라로 촬영된 영상을 판매한다는 글이 다크웹에 게시되었고[7], 독일의 보안업체 리큐리티랩스(Recurity labs)는 독일 남부 지역의 전력 공급을 외부에서 무단으로 차단할 수 있음을 입증하였다[8]. 따라서 IoT에 대한 보안 솔루션 도입과 다양한 환경에서 발생할 수 있는 보안 문제의 해결이 시급하다.

본 연구에서는 현재 IoT 환경에서 널리 사용되는 MQTT 프로토콜을 타겟으로 하는 여러 공격 기법을 탐지하기 위해 슬라이딩 윈도우 기반 LSTM[9] 딥러닝 모델 기반 시스템을 제안하였다. 또한 제안된 모델이 실제 환경에서 동적으로 실시간 탐지가 가능함을 보이기 위해 침입 탐지 시스템(IDS, Intrusion Detection System)을 구현하여 실험을 진행하였다. 딥러닝 모델을 학습시키기 위한 데이터셋으로는 MQTT 프로토콜 데이터셋인 MQTTset[10]과 본 연구에서 라즈베리파이 기반 IoT 환경을 구축하여 수집한 데이터를 사용하였다.

본 논문의 기여도를 나열하면 다음과 같다. 첫째, 급격하게 발전하는 IoT 분야의 고도화되고 증가하는 위협을 탐지하기 위해 슬라이딩 윈도우 기법을 사용하여 LSTM(Long Short-Term Memory) 기반 악성 트래픽 탐지 모델을 제안하였고, 기존 연구에 비해 성능이 향상되었음을 보였다. 둘째, 제안된 모델을 실제 환경에서 사용할 수 있음을 확인하기 위해 딥러닝 기반 침입 탐지 시스템을 구축하고 실시간 탐지를 진행하였다.

본 논문의 구성은 다음과 같다. 2장은 네트워크 환경에서 머신러닝 및 딥러닝 기반의 공격 탐지에 대한 기존 연구를 살펴본다. 3장에서는 본 연구에서 제안하는 LSTM 및 슬라이딩 윈도우를 이용한 악성 트래픽 탐지 모델과 데이터 전처리 방식에 대해 자세히 기술한다.

또한 데이터 수집을 위해 구축한 IoT 환경과 제안된 모델을 이용해 구현한 IDS에 대해 설명한다. 4장에서는 제안된 모델의 성능을 MQTTset과 본 연구에서 수집한 데이터셋을 이용하여 평가하고 기존 연구와 비교한다. 마지막으로 5장에서는 결론에 대해 기술하고 본 연구의 한계점과 향후 연구 방향에 대하여 기술한다.

II. 관련 연구

네트워크 상에서 발생하는 다양한 공격을 탐지하고 분류하기 위해 머신러닝 및 딥러닝 기반 NIDS (Network-based IDS)를 사용한 많은 연구들이 진행되어왔다.

I. Vaccari, et al[10]은 MQTT 환경에서 머신러닝 기반 악성 트래픽 탐지 연구의 필요성을 제시하고, 10가지 IoT 센서에서 수집된 정상 데이터 및 5가지 공격 트래픽 데이터가 포함된 MQTTset을 공개하였다. 또한 수집된 데이터셋에 Random Forest, Naïve Bayes, Decision Tree, DNN(Deep Neural Network), Gradient Boost 등 머신러닝 및 딥러닝 모델을 적용하여 분류 정확도를 측정하였다. 정확도는 Random Forest가 91%로 가장 높았고, Decision Tree, Neural Network가 그 뒤를 이었다. 그러나 기본적인 DNN만 사용한 점과 단일 패킷만 사용한 점 등을 고려하였을 때 추가적인 실험이 필요하다.

Yang, et al[11]은 MQTTset에 앙상블(Ensemble) 모델을 적용하였다. 앙상블 모델은 다수의 약한 알고리즘을 결합하여 강한 알고리즘을 생성하는 모델이다. 해당 연구에서는 KNN, Logistic Regression, Decision Tree, AdaBoost, XGBoost 등 7개의 머신러닝 알고리즘으로부터 출력된 값에 Random Forest를 적용시켜 정확도를 개선하고자 하였다. 그러나 91.07%의 정확도와 0.9097의 F1-score로 판단하였을 때, 기존 연구에 비해 크게 개선되지는 않았다.

J. G. Lee, et al[12]은 저사양 MQTT IoT 환경에 적합하도록 차원감소가 적용된 딥러닝 모델을 제안하였다. MQTTset을 대상으로 PCA(Principal Component Analysis)를 적용하고, LightGBM(Light Gradient Boosting Model) 분류 모델을 이용하여 성능 확인 결과, 원본 데이터셋을 15%로 축소하여도 전체 데이터셋을 사용한 모델과 유사한 성능을 보였다. 정확도는 93.9%이며, F1-score는 0.938을 달성하였다. 그러나 클래스간 데이터 불균형을 고려하지 않았다는 한계점이 있다.

J. Kim, et al[13]은 LSTM-RNN 기반의 침입 탐지 기법을 제안하고 KDD Cup 1999 데이터셋을 이용하여 성능을 평가하였다. 해당 데이터셋은 정상 트래픽, DoS, U2R(User to Root Attack), R2L(Remote to Local Attack), Probing attack으로 이루어져있다. 성능 평가 척도로 DR(Detection Rate)과 FAR(False Alarm Rate)를 사용하였으며, DR은 98.88%, FAR는 10.04%, 정확도는 96.93%를 달성하였다. 그러나 전체 데이터셋의 10%만 사용한 점과 너무 오래된 데이터셋을 사용하였다는 한계점이 있다.

Z. Li, et al[14]은 IDS를 위한 합성곱 신경망 (CNN, Convolutional Neural Networks) 기반의 트래픽 분류 모델을 제안하고, KDD'99 데이터셋을 보완한 NSL-KDD 데이터셋을 이용하여 성능을 평가하였다. CNN 모델은 ResNet50과 GoogLeNet을 사용하였고, NSL-KDD Test+에서 ResNet50이 79.14%의 정확도와, 0.7912의 F1-score를 보였고, GoogLeNet은 77.04%의 정확도와 0.7650의 F1-score를 보였다.

III. 딥러닝 기반 악성 MQTT 트래픽 탐지

기존 연구[10]-[12]에서는 단일 패킷 기반 딥러닝 모델을 사용하였다. 즉, 하나의 패킷에 대하여 해당 패킷이 어떠한 공격인지 판단하고자 하였다. 그러나 일반적으로 프로토콜은 연결, 인증, 데이터 전송, 종료 등의 순서가 있으며, 이러한 순서에 따라 패킷을 여러 번 전송한다. 따라서 하나의 패킷 데이터만으로는 해당 패킷이 악성인지 정상인지 탐지하기가 어렵다.

예를 들어 비밀번호 인증을 사용하는 MQTT 브로커를 대상으로 bruteforce 공격을 수행할 때, 공격자가 보내는 인증 패킷과 정상 디바이스가 초기에 보내는 인증 패킷을 구별하기 어렵다. 또한 정상 디바이스가 보내는 MQTT publish 데이터와 유사한 데이터로 공격자가 DoS 공격을 수행하는 경우 탐지가 까다롭다는 문제점이 있다.

따라서 본 연구에서는 순환 신경망(RNN, Recurrent Neural Network) 바탕인 LSTM 기반 모델을 사용하였으며, 데이터 전처리 과정에서 슬라이딩 윈도우를 적용하였다. RNN은 은닉층 노드에서 나온 결과값을 출력층 방향으로 보내는 동시에 다시 은닉층의 다음 입력으로 보내는 특징을 가지고 있다. LSTM은 RNN의 장기 의존성 문제를 보완한 신경망으로 시계열 데이터와 자연어 처리 등 순차 데이터(Sequential data)에서 높은 정확도를 보인다. 일반적인 프로토콜에는 패킷의 순서가 있으므로 본 연구에서는 LSTM을 채택하였고, 단일 패킷으로 정확한 탐지가 어렵다는 점을 해결하기 위해 데이터 전처리 과정에 슬라이딩 윈도우를 적용하여 다중 패킷을 입력으로 사용하였다.

3.1 데이터 전처리

본 연구에서는 MQTT 트래픽 데이터를 전처리하여 딥러닝 알고리즘의 입력으로 사용하였다. 전처리 과정은 다음과 같다. 원시 패킷 데이터에서 TCP 및 MQTT 레이어의 byte sequence를 추출하고, 포트 번호를 제거하기 위해 TCP 헤더의 앞 4바이트를 삭제하였다. MAC, IP 주소, 포트 번호 등의 정보는 IoT 기기와 네트워크 환경에 따라 달라지므로 이러한 정보는 배제하였다. 기존 연구[10]에서도 네트워크 구성에 독립적인 탐지를 위해서 이러한 데이터는 제거하여 전처리하였다. 기존 연구에서는 전처리 과정에서 특징 추출(Feature extraction)을 수행하였지만, 본 연구에서는 패킷의 특성을 고려하여 특징 추출하지 않은 연속된 바이트 데이터를 사용하였다.

MQTTset의 패킷 길이 분포는 그림 1과 같다. IP 헤더를 제외한 평균 패킷 길이는 44바이트이며, 100바이트 이하의 패킷은 전체의 99.5%를 차지한다. 따라서 본 연구에서는 각 패킷의 길이가 100바이트가 되도록 패딩하였다. MQTTset 데이터셋의 클래스에 따른 평균 세션 길이는 표 1과 같다. MQTT 프로토콜은 TCP 기반이이므로, 평균 세션 길이는 전체 패킷 개수를 SYN 패킷 개수로 나누어 계산할 수 있다. 정상 클래스(IoT 센서)는 부팅 후 커넥션을 수립한 이후, 해당 커넥션을 재부팅 이전까지 사용하므로 세션이 매우 긴 반면, Bruteforce 등 공격은 연결을 여러 번 시도하므로 평균 세션이 짧은 것을 볼 수 있다. 또한 슬라이딩 윈도우 길이에 따른 딥러닝 탐지 정확도와 전처리 시간은 표 2와 같다. 본 연구에서는 MQTT 평균 세션 길이와 적당한 전처리 시간을 고려하여 슬라이딩 윈도우의 크기를 30으로 설정하였다.

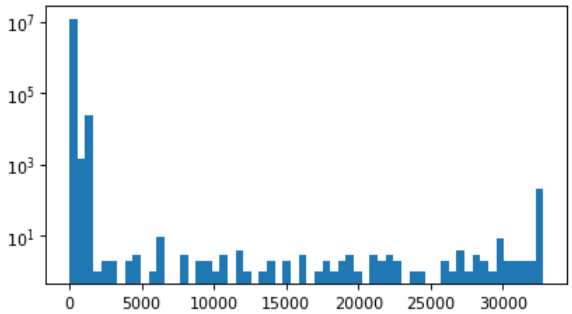

그림 1. MQTTset 패킷 길이 분포
Fig. 1. Packet length histogram of MQTTset

표 1. 클래스별 평균 MQTT 세션 길이
Table 1. Average MQTT session length by class

Class	Average session length
Bruteforce	9.14
Flood	612
DoS	8680
Malformed	11.11
SlowITe	7.98
Legitimate	1191570

표 2. 슬라이딩 윈도우 길이에 따른 MQTTset의 탐지 정확도와 전처리 소요 시간
Table 2. Detection accuracy and preprocessing time of MQTTset according to sliding window size

Window size	Accuracy	F1-score	Preprocessing Time
1	91.64%	0.9132	-
2	97.73%	0.9773	16s
5	99.66%	0.9965	40s
10	99.87%	0.9987	1m 18s
20	99.83%	0.9983	2m 39s
30	99.97%	0.9997	3m 55s
50	99.99%	0.9999	6m 30s
100	99.99%	0.9999	13m 5s

MQTTset의 클래스별 패킷 개수와 pcap 파일 크기는 표 3과 같다. MQTTset은 정상 클래스가 전체의 98.6%를 차지하는 비율 불균형 문제가 있으며, 이를 해결하기 위해 패킷 개수가 적은 클래스는 패킷을 복사하여 증강(augmentation)하였다. 트래픽 클래스는 총 6가지 종류이고 Legitimate, Bruteforce, DoS, Flood, Malformed, SlowITe 트래픽이 있다. 전처리 이후 각 클래스당 패킷 개수는 50,000개, 총 패킷 개수는 300,000개이다.

MQTTset은 10가지 정상 디바이스(IoT 센서)에 대한 패킷을 하나의 클래스로 묶어 제공하고 있다. 그러나 각 센서는 특정한 MQTT 토픽에 대해서만 publish 하므로, 이를 고려하여 정상 클래스에 슬라이딩 윈도우를 적용할 때는 먼저 IP별로 패킷을 분리한 뒤, 각 센서별로 슬라이딩 윈도우를 적용하고 마지막에 정상 클래스로 합쳐 사용하였다.

표 3. 클래스별 패킷 개수와 파일 크기
Table 3. Number of packets and file size per class

Class	# of Packet	Size	Preprocessed
Legitimate	11,915,716	1.01 GB	50,000
Bruteforce	14,501	1.33 MB	50,000
Flood	613	7.83 MB	50,000
DoS	130,223	47.5 MB	50,000
Malformed	10,924	0.99 MB	50,000
SlowITe	9,202	788 KB	50,000

3.2 딥러닝 기반 악성 트래픽 탐지 모델

딥러닝 모델은 LSTM을 사용하였고, 학습과 검증을 위해 데이터셋을 7:3 비율로 나누었다. 모델에는 4개의 레이어를 사용하였으며, 첫 번째 레이어는 LSTM을 사용하였고, 나머지는 레이어는 Dense를 사용하였다. 활성화 함수는 ReLU를 사용하였고, 마지막 레이어는 다중 분류를 위해 Softmax를 적용하였다. 옵티마이저는 Adam을 사용하였고, 과적합을 예방하기 위해 early stopping을 적용하였다.

모델 평가 지표로는 정확도(Accuracy)와 F1-score를 사용하였다. 전통적으로 기계학습과 딥러닝 분야에서는 분류 모델의 성능을 혼동 행렬(Confusion matrix) 상의 4개의 값을 이용하여 계산한다. 정확도는 전체 데이터에 대해서 참을 참으로, 거짓을 거짓으로 예측한 것의 비율이다. 그러나 입력 데이터가 편중될 경우 정확도만으로 평가하는 것은 적절하지 않다. 정밀도(Precision)는 모델이 참이라고 예측한 것 중에서 실제 참인 것의 비율이다. 재현율(Recall)은 실제 참인 것 중에서 모델이 참이라고 예측한 것의 비율이다. F1-score는 정밀도와 재현율의 조화평균으로 trade-off 관계에 있는 두 값을 반영할 수 있어 널리 사용된다.

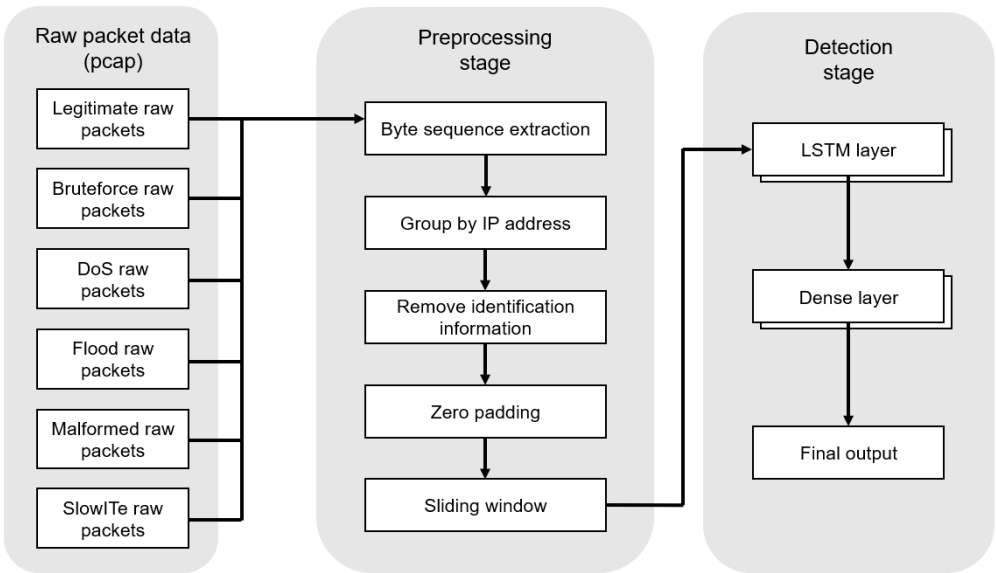

그림 2. 데이터 전처리 및 학습 과정
Fig. 2. Data preprocessing and model training

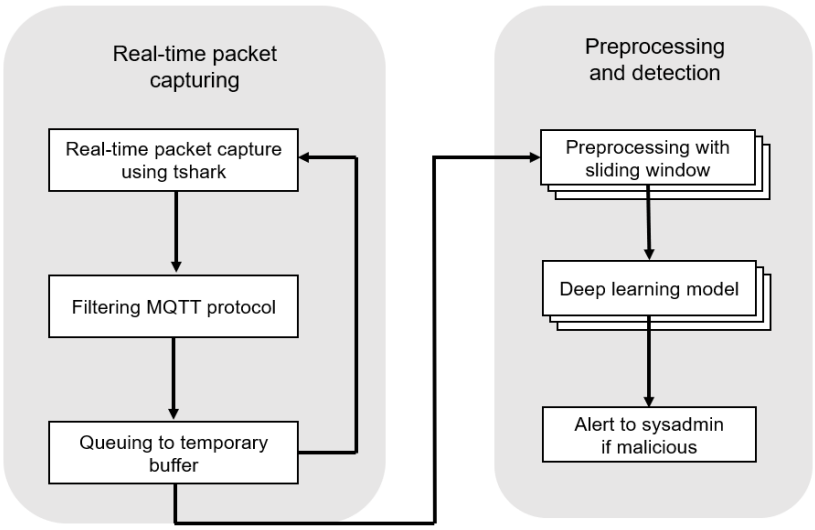

그림 3. 딥러닝 기반 실시간 침입 탐지 시스템
Fig. 3. Deep learning based real-time IDS

$$Accuracy = \frac{TP+TN}{N} \quad (1)$$

$$\mathrm{Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = 2\frac{\mathrm{Precision} \times Recall}{\mathrm{Precision} + Recall} \quad (4)$$

3.3 딥러닝 기반 실시간 침입 탐지 시스템

본 연구에서 제안한 딥러닝 모델을 이용하여 실시간 탐지가 가능함을 보이기 위해 침입 탐지 시스템(IDS)를 구축하였다. IDS에서 트래픽 탐지는 다음과 같은 순서로 진행된다. 먼저 특정 인터페이스로부터 패킷을 tshark를 이용하여 실시간으로 캡처한다. 이때, TCP 포트 번호를 기준으로 MQTT 패킷을 필터링한다. 캡쳐된 패킷은 큐 형태의 버퍼에 저장된다. 버퍼에 패킷이 채워지면 슬라이딩 윈도우를 적용하여 딥러닝 모델에 입력한다. 탐지 결과 악성 트래픽으로 분류될 경우 알림을 발생시킨다.

Ⅳ. 실험 및 분석

4.1 실험 환경

데이터 전처리 및 딥러닝 모델 학습은 AMD Ryzen 7 3700x, 32GB RAM, NVIDIA GeForce RTX 3080, Windows 10 64-bit OS에서 Python 3.11, Tensor Flow 2.8 및 scikit-learn, Numpy 라이브러리를 이용하였다. 패킷 캡처에는 tshark 및 pyshark 라이브러리를 사용하였다.

본 연구에서 추가적인 데이터 수집을 위해 구축한 MQTT 기반 IoT 환경은 다음과 같다. Raspberry Pi 3B+/4B SBC(Single board computer)에 온도, 습도, 조도, 도어(Hall effect) 센서를 연결하고, Python에서 paho-mqtt 라이브러리를 이용하여 IoT 센서 퍼블리셔로 사용하였다. 브로커는 Mosquitto 2.0.15를 이용하여 구축하였다. 악성 패킷은 MQTTSA[15], MQTT SlowITe[16] 및 MQTT-malaria툴을 이용하여 생성하였다. MQTT 기반 IoT 환경의 개략도는 그림 4와 같다. 중앙 브로커에 무선으로 연결된 IoT 센서들과 이를 공격하는 악성 노드로 구성되어 있다. 또한 실제 실험 환경은 그림 5에서 볼 수 있다. IoT 센서 노드 4개와 공격자 노트북, Wi-Fi 공유기 등으로 구성되어 있다.

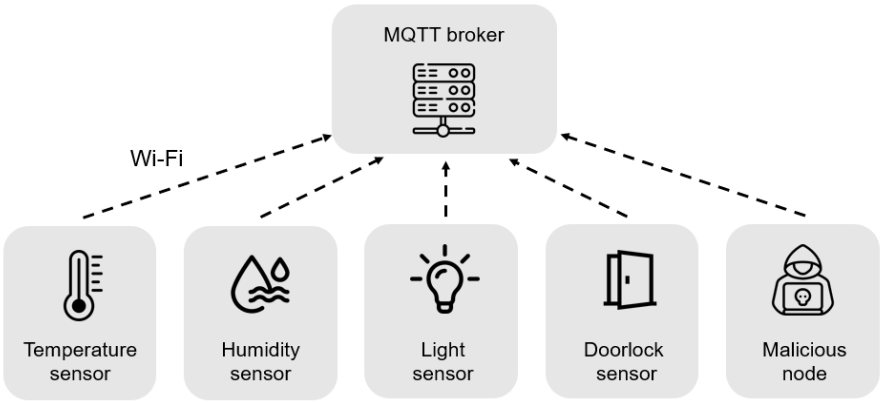

그림 4. MQTT 기반 IoT 환경
Fig. 4. MQTT-based IoT environment

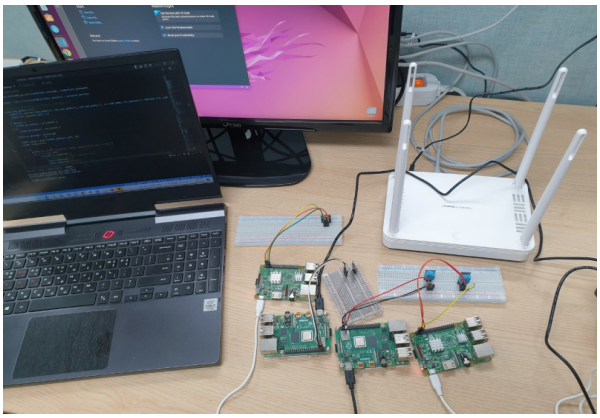

그림 5. IoT 실험 환경
Fig. 5. IoT experimental testbed

4.2 딥러닝 데이터셋 구성

딥러닝 학습 데이터셋은 MQTT 프로토콜 기반 사물인터넷 트래픽을 포함한 MQTTset을 사용하였다. 해당 데이터셋은 온도계, 습도계, 가스 센서, 모션 센서, 도어락 등의 10가지 IoT 디바이스로부터 수집된 정상 트래픽과 MQTT 공격 툴을 사용하여 수집한 악성 트래픽을 원시 데이터와 특징 추출된 CSV 파일로 제공한다. 공격 방식으로는 MQTT 사용자 인증을 우회하기 위한 무차별 대입(Bruteforce) 공격, 다수의 커넥션을 이용해 브로커의 리소스를 고갈시키는 DoS(Denial of Service) 공격, 단일 커넥션에서 리소스를 포화시키기 위해 대량의 데이터를 전송하는 Publish Flood 공격, 브로커 소프트웨어에서 예외를 발생시키기 위해 잘못된 데이터를 전송하는 공격, SlowITe 공격이 사용되었다[10]. SlowITe는 MQTT 환경에서 오래 지속되는 다수의 커넥션을 이용하여 최소의 대역폭과 리소스를 사용하는 DoS 공격이다[16].

MQTTset은 수집된 트래픽을 5개의 악성 클래스와 하나의 정상 클래스로 분류하여 제공한다. 또한 본 연구에서는 라즈베리파이 기반 MQTT IoT 환경을 구축하고 추가로 데이터를 수집하여 실제 환경에서도 활용 가능함을 보였다.

표 4. 전처리 전 데이터셋 크기
Table 4. Dataset size before preprocessing

Class	MQTTset	Ours
Legitimate	11,915,716	119,527
Bruteforce	14,501	10,986
Flood	613	49,971
DoS	130,223	91,606
Malformed	10,924	30,165
SlowITe	9,202	73,272

4.3 실험 결과

MQTTset 데이터셋에 슬라이딩 윈도우를 적용하여 그림 2와 같이 전처리한 뒤, LSTM 기반 모델로 학습하고 평가한 결과는 표 5, 7과 같다. 정확도는 99.97%이고, F1-score는 0.9997이다. 학습 과정에서 정확도와 손실 함수 그래프는 그림 6와 같다. 본 연구에서 MQTT 기반 IoT 환경을 구축하고 수집한 데이터셋을 이용하여 학습/평가한 결과는 표 6과 같다. 정확도는 99.42%이며, F1-score는 0.9942이다.

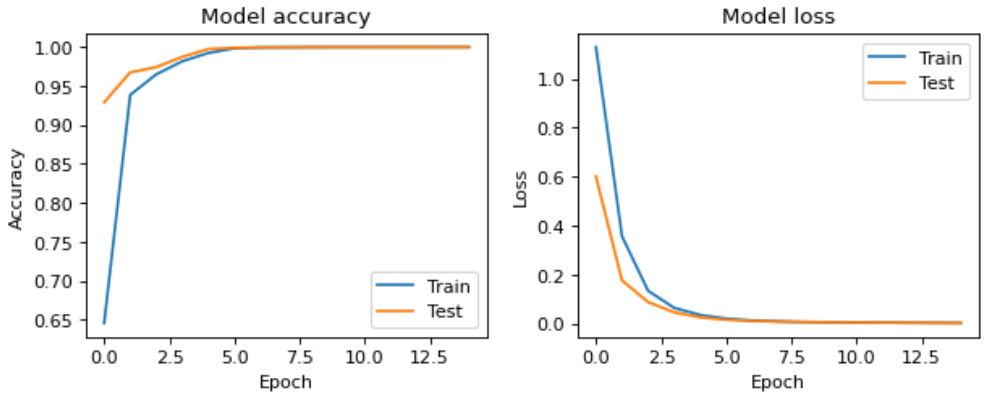

그림 6. Epoch에 따른 정확도 및 손실 그래프
Fig. 6. Accuracy and loss graphs over epochs

표 5. MQTTset에 대한 혼동 행렬
Table 5. Confusion matrix for MQTTset

Act. \ Pre.	Legitimate	Bruteforce	Flood	DoS	Malformed	SlowlTe
Legitimate	15000	-	-	-	-	-
Bruteforce	-	15000	-	-	-	-
Flood	-	-	15000	-	-	-
DoS	-	19	-	14975	6	-
Malformed	-	-	-	-	15000	-
SlowlTe	-	-	-	-	-	15000

표 6. 본 연구에서 수집한 데이터에 대한 혼동 행렬
Table 6. Confusion matrix for our dataset

Act. \ Pre.	Legitimate	Bruteforce	Flood	DoS	Malformed	SlowlTe
Legitimate	15000	-	-	-	-	-
Bruteforce	-	14985	-	-	15	-
Flood	1	-	14963	33	2	1
DoS	15	14	65	14866	40	-
Malformed	2	117	-	25	14854	2
SlowlTe	-	-	-	26	162	14812

MQTTset 데이터셋을 이용한 기존 연구와 정확도 및 F1-score를 비교한 내용은 표 7과 같다. LSTM을 사용할 경우, 기존 신경망에 비해 정확도가 1.2%p 향상되었으며, Naïve bayes 대비 27.3%p가 향상되었다. RNN과의 비교에서는 LSTM의 정확도가 0.9%p 높은 것을 볼 수 있다. 또한 본 연구에서 제안하는 LSTM과 슬라이딩 윈도우를 적용한 방법은 기존 신경망 대비 9.5%p가 향상된 것을 확인할 수 있다.

표 7. MQTTset에 대한 정확도 및 F1 점수 비교
Table 7. Accuracy and F1-score comparison for MQTTset

Classifier	Accuracy	F1-score
Neural network	90.4%	0.902
Random forest	91.5%	0.914
Naïve bayes	64.3%	0.687
Decision tree	91.5%	0.914
Gradient boost	87.9%	0.872
Multilayer perceptron	90.3%	0.901
Stacking ensemble	91.0%	0.909
RNN	90.7%	0.905
LSTM	91.6%	0.913
LSTM with sliding window (proposed method)	99.9%	0.999

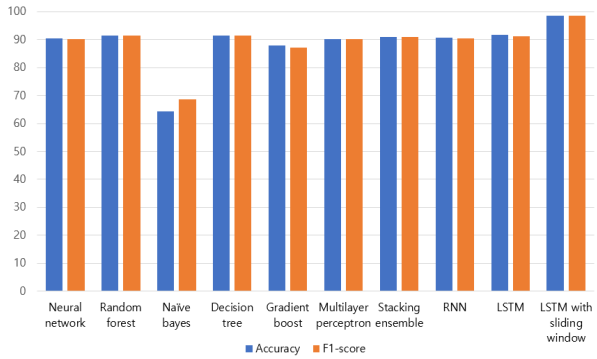

그림 7. MQTTset에 대한 정확도 및 F1 점수 그래프
Fig. 7. Accuracy and F1-score graph for MQTTset

또한 본 연구에서 제안한 악성 패킷 탐지 방법을 적용한 IDS의 성능 평가 결과는 표 8과 같다. 측정 단위는 pps(packet per second)이다. 전처리 과정은 CPU에서 싱글 코어로 진행하였으며, 딥러닝 학습과 평가는 GPU에서 진행하였다.

표 8. 본 연구에서 제안한 방법의 성능 평가
Table 8. Performance evaluation of the proposed method

Type	Packets	Time	Throughput
Preprocessing	12,081,179	235s	51,409 pps
Training	210,000	14.85s	14,141 pps
Testing	90,000	3.832s	23,486 pps
Preprocessing + Testing	90,000	5.583s	16,120 pps

Ⅴ. 결론 및 향후 과제

본 연구에서는 IoT 분야에서 사용되는 대표적인 프로토콜인 MQTT에 대한 악성 트래픽을 탐지할 수 있는 딥러닝 기반 모델을 제안하였다. 프로토콜 데이터의 특성을 고려하여 전처리 과정에서 슬라이딩 윈도우 기법을 적용하였으며, 순환 신경망 구조의 LSTM을 사용하여 기존 신경망 대비 9.5%p 향상된 탐지 정확도를 달성하였다. 또한 이를 기반으로 IDS를 구현하여 실시간 환경에서 사용 가능성을 보였다. IoT 분야가 발전하면서 MQTT 이외의 다른 프로토콜이 사용되거나, 새로운 공격 방식이 등장할 수 있다. 딥러닝 모델을 이용하여 악성 공격을 탐지하면 이러한 경우에도 유연하게 대응할 수 있을 것으로 기대된다.

그러나 딥러닝 기반 IDS의 탐지 성능이 기존의 규칙 기반(Rule-based) IDS에 비하여 크게 낮아 최적화 연구가 필요한 점, LSTM보다 우수하다고 평가되는 트랜스포머(Transformer) 기반 실험 등이 진행되지 않았다는 점 등에서 본 연구의 한계를 찾을 수 있다. 따라서 이러한 한계점을 극복하기 위한 연구를 본 논문의 향후 과제로 수행할 수 있을 것이다.

References

[1] Transforma Insights, "Global IoT connections to hit 29.4 billion in 2030", https://transformainsights.com/news/global-iot-connections-294 [Accessed: Feb. 20, 2023]

[2] MarketsandMarkets, "IoT Solutions and Services Market - Global Forecast to 2027", https://www.marketsandmarkets.com/Market-Reports/iot-solutions-and-services-market-120466720.html [Accessed: Feb. 20, 2023]

[3] Tara Salman, "Internet of Things Protocols and Standards", https://www.cse.wustl.edu/~jain/cse570-15/ftp/iot_prot [Accessed: Feb. 20, 2023]

[4] Ian Skerrett, "Strong Adoption of MQTT in IIoT", https://www.hivemq.com/blog/strong-adoption-of-mqtt-in-iiot [Accessed: Feb. 20, 2023]

[5] B. Zhao, et al., "A Large-Scale Empirical Study on the Vulnerability of Deployed IoT Devices", in IEEE Transactions on Dependable and Secure Computing, Vol. 19, No. 3, pp. 1826-1840, May 2022. https://doi.org/10.1109/TDSC.2020.3037908.

[6] H. Jeon and S. Lee, "Analysis of Remote Update Vulnerabilities of IoT Healthcare Devices", The Journal of Korean Institute of Information Technology, No. 19, No. 1, pp. 87-97, Jan. 2021. http://dx.doi.org/10.14801/jkiit.2021.19.1.87.

[7] BoanNews, "Apartment wall pad hacking issue, hacked apartment list is being distributed", https://www.boannews.com/media/view.asp?idx=102768 [Accessed: Feb. 20, 2023]

[8] S. J. Kim and D. E. Cho, "Technology Trends for IOT Security", The Korea Contents Association Review, Vol. 13, No. 1, pp. 31-35, Mar. 2015. https://doi.org/10.20924/CCTHBL.2015.13.1.031.

[9] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", Neural Computation, Vol. 9, No. 8, pp. 1735-1780, Nov. 1997. https://doi.org/10.1162/neco.1997.9.8.1735.

[10] I. Vaccari, G. Chiola, M. Aiello, M. Mongelli, and E. Cambiaso, "MQTTset, a New Dataset for Machine Learning Techniques on MQTT", Sensors, Vol. 20, No. 22, Nov. 2020. https://doi.org/10.3390/s20226578.

[11] Y. Yang and Y. Kim, "A Threat detection using Machine Learning in MQTT IoT Environments", Proceedings of the Korea Contents Association Conference, pp. 445-446, Aug. 2021.

[12] J. G. Lee, S. J. Lee, and Y. W. Kim, "Attack

Detection and Classification Method Using PCA and LightGBM in MQTT-based IoT Environment", Jouranl of Information and Security, Vol. 22, No. 4, pp. 17-24, Oct. 2022.

[13] J. Kim, J. Kim, H. L. T. Thu, and H. Kim, "Long Short Term Memory Recurrent Neural Network Classifier for Intrusion Detection", 2016 International Conference on Platform Technology and Service(PlatCon), Jeju, Korea, pp. 1-5, Feb. 2016. https://doi.org/10.1109/PlatCon.2016.7456805.

[14] Z. Li, Z. Qin, K. Huang, X. Yang, and S. Ye, "Intrusion Detection Using Convolutional Neural Networks for Representation Learning", ICONIP 2017: Neural Information Processing, Lecture Notes in Computer Science, Vol. 10638, pp. 858-866, Oct. 2017. https://doi.org/10.1007/978-3-319-70139-4_87.

[15] A. Palmieri, P. Prem, S. Ranise, U. Morelli, and T. Ahmad, "MQTTSA: A Tool for Automatically Assisting the Secure Deployments of MQTT Brokers", 2019 IEEE World Congress on Services (SERVICES), Milan, Italy, pp. 47-53, Jul. 2019. https://doi.org/10.1109/SERVICES.2019.00023.

[16] I. Vaccari, M. Aiello, and E. Cambiaso, "SlowITe, a Novel Denial of Service Attack Affecting MQTT", Sensors 2020, Vol. 20, No. 10, May 2020. https://doi.org/10.3390/s20102932.

저자소개

이 든 (Deun Lee)

2019년 3월 ~ 현재 : 전북대학교 소프트웨어공학과 학사과정
관심분야 : 소프트웨어공학, 인공지능, 네트워크, 보안

임 승 순 (SeungSoon Im)

2018년 3월 ~ 현재 : 전북대학교 소프트웨어공학과 학사과정
관심분야 : 소프트웨어공학, 소프트웨어보안

최 선 오 (Sunoh Choi)

2008년 2월 : 고려대학교 컴퓨터학과(이학사 및 이학석사)
2014년 5월 : Purdue대학교 전자및컴퓨터공학부(공학박사)
2014년 ~ 2019년 : 한국전자통신연구원 정보보호연구본부 선임연구원
2019년 ~ 2021년 : 호남대학교 컴퓨터공학과 조교수
2021년 3월 ~ 현재 : 전북대학교 소프트웨어공학과 부교수
관심분야 : 인공지능보안, 네트워크보안, 데이터보안