CAN 프레임 간격 기반 침입 탐지 방법론

고성지*, 유수경*, 서영재*, 장재현*, 곽병일**

*, ** 한림대학교 소프트웨어학부 (학부생, 교수)

CAN Frame Interval-Based Intrusion Detection Methods for In-Vehicle Network

Seong-ji Ko*, Soo-Kyung Yoo*, Yeong-Jae Seo*, Jae-Hyun Jang*, and Byung Il Kwak

*, ** Hallym University Software (Undergraduate student, Professor)

요 약

차량의 Electronic Control Unit (ECU)는 효율적인 통신을 위해 차량 내부 네트워크인 Controller Area Network (CAN)을 이용한다. 이러한 CAN bus는 보안 기능 없이 설계되어 있어서 접근 제어나 인증 없이 공격자가 CAN 네트워크에 쉽게 접근할 수 있다. 본 논문은 CAN bus의 경량화된 CAN ID Sequence와 Time interval Sequence 피처 기반의 침입 탐지 방법론을 제안한다. 해당 방법론은 높은 탐지 성능을 나타내며, 4ms의 실시간 탐지 시간을 확보하였다. 또한, 최적의 Sequence 길이를 확보하기 위해 Sequence 길이 변화에 따른 탐지 성능을 비교하였다.

I. 서론

CAN (Controller Area Network) bus는 보안 기능 없이 설계되어 기밀성, 무결성, 가용성 공격에 취약하다. 차량에 탑재된 ECU (Electronic Control Unit)들은 CAN bus로부터 수신하는 모든 메시지를 인증하지 않기 때문에, 공격자 관점에서 악의적으로 생성한 CAN 메시지를 전송하더라도 추가적인 보안 기능을 적용하지 않고 수신한다. 이로 인해, 차량의 구동에 관련된 주요 기능들을 제어할 수 있는 취약점이 발생할 수 있다. 또한, CAN bus 자체의 암호화 기능이 없으며, 멀티 마스터 특성으로 인해 CAN bus에 접속한 모든 ECU 및 노드들은 CAN 메시지를 브로드캐스팅 및 수신할 수 있다. 이로 인해, CAN bus에 접속한 공격자는 추가적인 장비 또는 모듈을 설치하지 않고도 CAN bus에 전송되는 모든 CAN 메시지들을 스니핑 할 수 있는 취약점이 있다.

최신 차량은 CAN bus 탑재를 필수로 하고 있으며, 새로운 기술 적용을 위해 미래 차량에는 더욱 더 많은 ECU 및 센서들의 탑재를 필요로 하고 있다. 이러한 부분은, 미래 차량의 성능이 향상되더라도 CAN bus 상에서 생성 및 전송되는 네트워크 트래픽의 크기가 점차적으로 커질 수 있음을 의미한다. 그러한 이유로, In-Vehicle Network (IVN)을 위한 차량 침입 탐지 모듈은 많은 양의 네트워크 트래픽을 실시간으로 처리할 수 있도록 경량화가 요구된다. 딥러닝을 위한 고사양 장비는 전력 소모, 시간 및 비용의 증가, 모델의 크기에 따른 연산량 증가 문제를 야기하므로 경량화가 필수적이다.

IVN 침입 탐지 연구로 Desta et. al. 은 Time interval Sequence 기반의 딥러닝 침입 탐지 방법론을 제안하였으며, Song et. al.은 CAN ID Sequence 기반의 경량화된 딥러닝 침입 탐지 방법론을 제안하였다 [1],[2]. 하지만, 이러한 연구들에서 사용한 딥러닝 알고리즘은 실시간성이 요구되어 IVN 침입 탐지에서 단점으로 작용하며 그 사용이 제한될 수 있다. 결과적으로 침입 탐지를 위한 경량화된 피처 설계 및 알고리즘의 구성이 필요하다. 따라서 우리는 IVN에서의 적용 가능한 경량화된 Sequence 피처 및 머신러닝 알고리즘을 사용한 침입 탐지 방법론을 제안한다.

II.배경지식

2.1. CAN

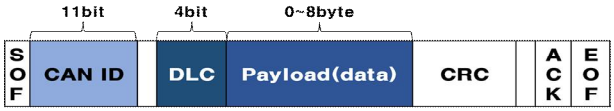

Fig 1. CAN Data Frame

CAN은 ECU 또는 차량 내 탑재된 다양한 센서 간의 통신을 제공하는 직렬 통신 프로토콜이다. 차량 내부 네트워크로써 센서와 ECU 사이의 엔진/차체/센서데이터와 같은 중요 정보들을 실시간으로 통신하기 위해 CAN 프로토콜은 최대 1Mbit/s의 비트 전송률을 가진다[3]. 이러한 CAN 프로토콜은 Data frame, Remote frame, Error frame, Overload frame 총 4개 frame으로 구성되는데, 본 논문에서는 Data frame만을 침입 탐지에 적용하였다. 해당 Data frame의 형태는 Fig 1과 같으며, 각 필드에 대한 설명은 Table 1에 나타내었다.

Table 1. CAN Data Frame Field

Field	설명
Start of Frame (SOF)	CAN 메시지 전송 시작과 관련하여 모든 노드를 동기화하고 알리는 데 사용된다.
CAN ID	메시지를 수신해야 하는 ECU 식별 번호에 사용되고, 크기는 11bit이다. 메시지의 우선순위는 이 Filed에 의해 설정되며, 일반적으로 값이 작을수록 우선순위가 높아진다.
DLC	Data Field의 길이를 나타내는 Control Field의 일부이며 범위는 0-8 byte이다.
Data Field (Payload)	수신된 ECU에 의해 해석되는 애플리케이션 Payload Data를 포함한다.

2.2. 공격 시나리오

본 연구에서의 적용한 공격 시나리오는 Denial of Service (DoS), Fuzzy, Malfunction이며 Table 2에 나타내었다.

Table 2. Attack Method

공격기법	설명
DoS Attack	공격자는 CAN Bus 장악을 위해 충돌 시 높은 우선순위를 가지도록 CAN ID를 '000'으로 설정하여 CAN Bus에 주입한다.
Fuzzy Attack	공격자는 랜덤하게 CAN ID를 선택하고 CAN 메시지를 반복적으로 주입하여 무차별 공격을 수행한다. 공격자는 해당 공격을 통해 차량의 이상 반응을 일으킬 수 있다.
Malfunction Attack	차량의 특정 기능 제어를 위해 공격자는 미리 파악한 CAN ID 및 Data Field를 설정하여 CAN Bus에 공격을 수행한다.

III. 방법론

3.1. 데이터 전처리

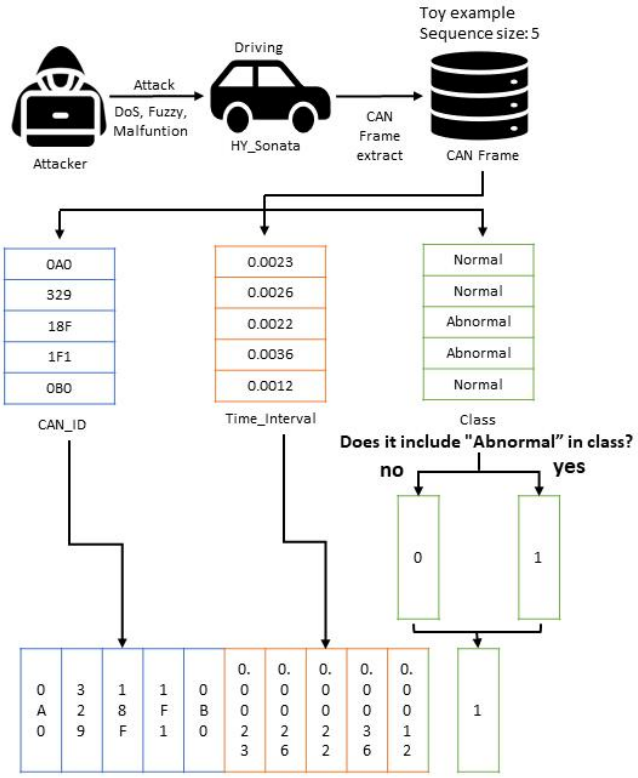

Fig 2. Data extraction and preprocessing process

본 논문은 경량화된 피처 구성을 위해 CAN ID Sequence와 CAN message 간의 Time interval을 추출하였으며, 해당 데이터를 Random Forest 알고리즘에 적용을 통해 IVN에서의 침입을 탐지한다. 또한, 정상 및 공격 데이터 분류를 위해 라벨링 과정을 수행하였다. Fig. 2는 본 연구에서 진행한 전체 데이터 전처리과정을 나타낸다.

Random Forest 알고리즘에 적용하기 위해 라벨 인코딩 기법을 이용하여 CAN ID Sequence를 숫자로 변환하였다. 정해진 길이만큼의 CAN ID Sequence와 Time interval Sequence를 하나의 행으로 벡터화하였다.

3.2. 데이터 라벨링

정상 및 공격 데이터 라벨링을 위해 3.1 데이터 전처리과정에서 추출한 고정된 길이의 Sequence Data에 1개라도 공격 CAN 메시지가 포함되어있을 경우 해당 데이터의 class를 1, 그렇지 않으면 0으로 라벨링을 진행한다.

3.3. 모델 학습

본 논문에서는 여러 결정 트리를 사용하여 앙상블 한 Random Forest 모델을 사용했다. Random Forest의 하이퍼 파라미터는 모두 디폴트값으로 설정하였다. 또한, 학습 및 평가를 위해 본 연구에 사용된 데이터셋을 8:2의 비율로 랜덤하게 분할 하였다.

IV. 실험 및 평가

본 장에서는 IVN에서의 공격 및 정상 CAN 트래픽이 포함된 데이터셋을 사용하여 실험을 진행하였다.

4.1. 데이터셋

본 논문에서는 Hacking and Countermeasure Research Lab (HCRL)에서 제공하는 “Survival Analysis Dataset for automobile IDS” 데이터셋을 이용하였다[4]. 해당 데이터셋은 주행 중인 Hyundai Sonata 차량에서 추출된 데이터셋으로 DoS, Fuzzy, Malfunction, Replay 공격이 적용되었다. 본 연구에서는 공격의 특징이 명확하게 드러나는 DoS, Fuzzy, Malfunction 공격에 대해서만 초점을 두어 실험을 진행하였다. 또한, Sequence 길이의 최적화를 위해 탐색하고자 하는 Sequence 길이를 2부터 50까지 설정하였으며, CAN ID Sequence 및 Time interval Sequence 피처에 적합한 성능 비교를 통해 최적의 Sequence 길이를 도출하였다.

4.2. 성능지표

앞서 설명한 실험에서 학습된 모델의 성능을 평가하기 위해 Area Under Curve (AUC) metrics를 이용하였다. threshold에 따른 False Positive Rate (FPR), True Positive Rate (TPR)을 통한 ROC-curve의 아래 면적인 AUC metric을 사용하여 성능을 확인하였다.

4.3. 실험 환경

본 논문에서 Sequence 길이에 따른 침입 탐지 방법론 실험은 Intel (R) Core (TM) i5-12400F CPU, NVIDIA GeForce RTX 3060, RAM 24.0GB, 그리고 Windows 10 Pro 64-bit 운영체제 환경에서 수행되었다. 머신러닝 알고리즘 적용을 위해 Python 3.8 IDE를 사용하였고, sklearn, numpy, pandas, matplotlib 등의 라이브러리를 사용하였다.

4.4. 실험 결과

본 실험에서 적용한 Time interval Sequence, CAN ID Sequence 피처들에 대한 AUC Score를

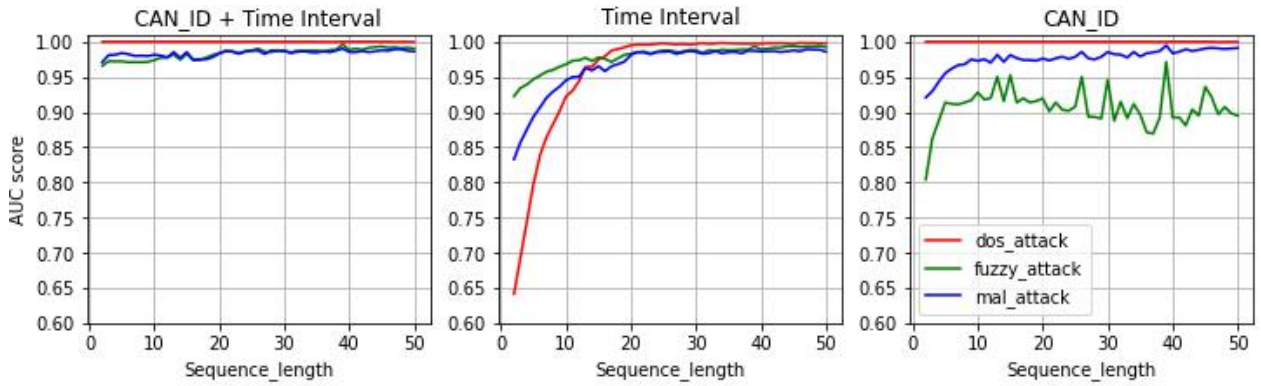

Fig 3. Sequence 데이터 종류 및 길이에 따른 공격 탐지 성능 - AUC score (Sequence 길이: 2-50)

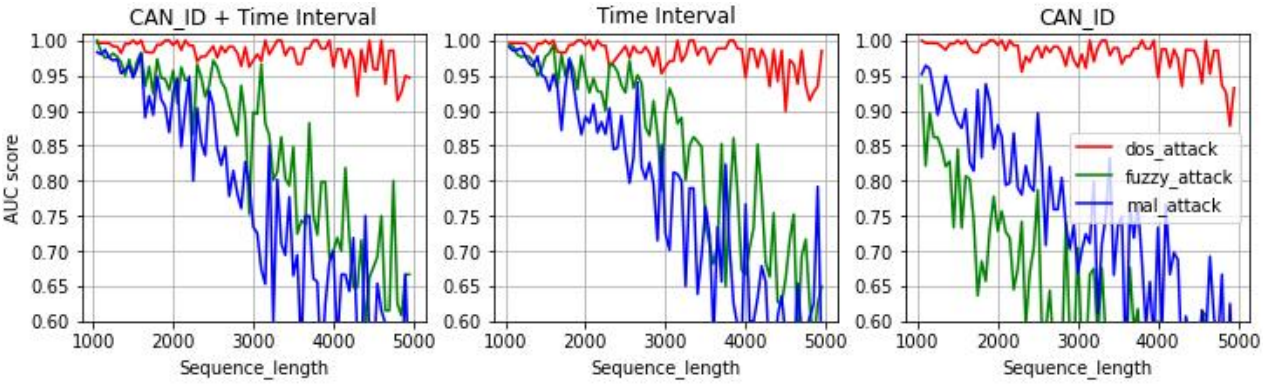

Fig 4. Sequence 데이터 종류 및 길이에 따른 공격 탐지 성능 - AUC score (Sequence 길이: 1000-5000)

Fig 3에 나타내었다. Fig 3에서는 Sequence 길이 2-50까지의 AUC 성능을 나타내었으며, Sequence 길이 1000 이상에서의 AUC 성능을 파악하기 위해 Fig 4와 같이 추가 실험을 진행하였다.

4.4.1. Time interval Sequence + CAN ID Sequence 피처 동시 적용

본 논문에서 사용한 Time interval Sequence 및 CAN ID Sequence 피처를 적용할 경우, DoS 공격에서는 CAN ID '000'의 존재 여부가 곧 클래스를 의미하므로 탐지 모델 역시 해당 CAN ID '000'이 있는지를 학습하여 Sequence 길이와 상관없이 AUC score가 모두 1.0을 기록했다. Malfunction 및 Fuzzy 공격에서는 CAN ID Sequence와 Time interval Sequence를 동시에 사용했을 때, 짧은 Sequence 길이가 3으로 설정되더라도 약 0.98 AUC score에 수렴하는 것을 확인하였다. 따라서, 두 개의 피처를 동시에 사용했을 때, 탐지 성능이 Sequence 길이가 변화하더라도 매우 우수함을 확인하였다. 3장에서 언급한 Random Forest의 하이퍼 파라미터는 모두 디폴트 값을 사용하였고, 학습된 모델을 통과시켰을 때의 탐지 시간을 확인해 본 결과, 4ms의 시간이 소요되었다. 이는 실시간성이 중요한 침입 탐지에 있어서 유의미한 결과로 볼 수 있다.

4.4.2. Time interval Sequence 단일적용

CAN ID Sequence를 보지 않고 CAN 메시지 간의 시간 격차를 나타낸 Time interval Sequnce 만을 통해 공격을 탐지하였을 경우, Sequence 길이 20에서부터 높은 탐지 성능을 달성하였다. Fig 3의 Time interval 그래프와 같이 Sequence 길이가 증가할수록 (최대 50) 성능이 지속적으로 증가함을 확인하였다.

4.4.3. CAN ID Sequence 단일적용

Time interval Sequence 정보를 보지 않고 CAN ID Sequence로만 성능을 확인했다. Fig 3의 가운데 그래프에서 DoS 공격의 경우 AUC score가 약 1.0으로 가장 높은 성능을 달성하였다. 추가로, Malfunction 공격 및 Fuzzy 공격에서는 DoS 공격에서와는 다르게 특정한 CAN ID를 번갈아 설정하여 공격 또는 랜덤하게 CAN ID와 Data Field Value를 설정하기 때문에, AUC 성능이 DoS 공격의 탐지 성능보다 낮음을 확인하였다. 다만, Malfunction 공격의 경우, 공격에서 사용한 CAN ID가 2가지이고, CAN ID Sequence가 Fuzzy 공격에서보다 유사한 경우가 많아서 Fuzzy 공격에서의 침입 탐지 성능보다 높음을 확인하였다.

4.4.4. Sequence 길이 변화에 따른 탐지 성능 비교

앞서 실험에서는 Sequence 길이 2에서 50까지 그 길이가 증가함에 따라 높은 탐지 성능으로 그 값이 수렴하는 것을 확인하였다. 이러한 Sequence 길이 증가에 따른 높은 AUC score의 상관관계 확인을 위해 Sequence 길이를 1000부터 5000까지 변화 간격을 50으로 설정하여 성능을 비교하였다.

실험 결과, Fig 4에서 나타낸 것과 같이 길이가 1000 이상일 경우 AUC score가 1.0에 수렴하지 않고 성능이 기존보다 더욱 낮아짐을 확인하였다. 이러한 결과는 낮은 Sequence 길이가 길어짐에 따라 높은 AUC score를 가지지만, Sequence 길이 1000 이상에서는 피처의 기능을 정상적으로 수행하지 못함을 확인하였다.

V. 결론

본 논문에서는 낮은 보안성을 가지는 CAN Bus에 대한 침입 탐지 방법론을 제안하였으며, 경량화된 피처로서 CAN ID Sequence 및 Time interval Sequence를 구성하고 머신러닝 알고리즘인 Random Forest를 사용하여 높은 탐지 성능을 확인하였다. 또한, Sequence 길이와 탐지 성능의 수렴 여부를 확인하기 위해, Sequence 길이 변화에 따른 침입탐지 성능을 비교하였다.

향후 연구에서는 자율주행 차량에서의 경량화된 침입탐지 연구를 수행할 예정이다.

[참고문헌]

[1] Araya Kiborm Desta, Shuji Ohira, Ismail Arai, Kazutoshi Fujikawa, "ID Sequence Analysis for Intrusion Detection in the CAN bus using Long Short Term Memory Networks", IEEE, 2020.

[2] Hyun Min Song, Ha Rang Kim and Huy Kang Kim, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network", IEEE, 2016.

[3] ROBERT BOSCH GmbH, Postfach 50, D-7000 Stuttgart 1.

[4] Mee Lan Han, Byung Il Kwck, Huy Kang Kim, "Anomaly intrusion detection method for vehicular networks based on survival analysis, ELSEVIER, 2018.