CAN Bus 보안 강화를 위한 Sequence와 Time interval 기반 침입 탐지 및 공격 분류

김형선*, 강도희*, 곽병일**

*,** 한림대학교 소프트웨어학부 (학부생, 교수)

Sequence and Time Interval-based intrusion detection and attack classification for CAN Bus security

Hyeong-seon Kim*, Do-hee Kang*, Byung-il Kwak**

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

요 약

차량에 탑재된 전자 제어 장치 (Electronic Control Unit, ECU)는 Controller Area Network (CAN) 통신 프로토콜을 이용하여 차량의 상태를 공유한다. 하지만, CAN Bus는 메시지 암호화 및 인증과 같은 보안 기능을 가지고 있지 않아 사이버보안 위협을 받을 수 있다. 본 논문에서는 차량의 CAN Bus에 대한 침입 탐지뿐만 아니라, 공격을 효과적으로 분류할 수 있는 Sequence 및 Time Interval 기반 피처와 침입 탐지 방법을 제안한다. 제안한 방법론을 통해 정확도 99.88%, F1-Score 99.27%의 성능을 확보하였다.

I. 서론

현대 차량은 점점 더 많은 전자 제어 장치 (Electronic Control Unit, ECU)를 탑재하여 차량 내부 시스템 관리 및 제어한다. 이러한 ECU들은 주행 안전성, 연비 최적화, 편의 기능 제공 등 다양한 역할을 수행하며, 이들 간의 원활한 상호 작용이 중요하다. 자동차 내부의 각 ECU는 Controller Area Network (CAN) 통신 프로토콜을 사용하여 정보를 교환한다. 하지만 CAN Bus는 정보를 주고받는 과정에서 메시지의 암호화나 인증과 같은 보안 기능을 제공하지 않아, 사이버보안 위협을 받을 수 있다.

이러한 보안 기능 부재로 인해 CAN Bus를 통한 시스템 침해나 악의적으로 차량이 제어될 수 있다. 따라서, 자동차 내부의 CAN Bus에 대한 침입 탐지 및 공격 분류 기술 개발이 중요하다. 본 논문에서는 CAN Bus에서의 발생 가능한 공격 탐지 및 식별 방법을 제안한다. 이를 위해 연속된 CAN Dataframe Sequence 및 Time Interval을 기반으로 피처를 추출하였다. 해당 피처를 XGBoost 알고리즘에 적용한 결과, 높은 정확도와 F1-Score를 달성하여 제안한 피처 기반 방법이 효과적임을 확인할 수 있었다.

II. 배경지식

2.1 CAN

차량의 고성능화로 차량 내 ECU 수가 증가하였으며, CAN은 차량용 네트워크에 적합한 표준 통신 프로토콜로 사용된다.[1] CAN은 High-Speed CAN과 Low-Speed CAN으로 나뉘며, 엔진 및 브레이크와 같은 제어에 CAN High를, 전조등 및 멀티미디어 기기와 같은 편의 기능에 CAN Low를 사용한다. 메시지 전송은 Broadcast 방식으로 이루어지며 Data, Remote, Error, Overload Frame이 사용된다. Fig.1은 CAN Data Frame을 나타낸다.

Data Frame은 ECU 간 실질적인 통신을 위해 사용되며, Data Field에 0-8 bytes의 메시지를 담고 있다. 이러한 Data의 길이를 나타내는

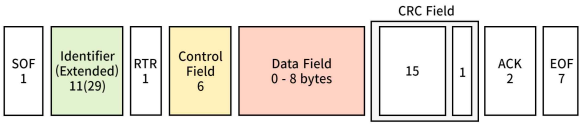

Fig.1 CAN Data Frame

DLC (Data Length Code)는 Control Field에 포함되어 있다. Identifier Field의 값을 통해 수신 여부를 결정하고, CRC Field를 통해 Data Frame의 오류 여부를 판별한다. 메시지 전송은 상태 확인 후 Idle일 때 메시지를 전송하거나 수신 모드로 들어간다. 2개 이상의 Dataframe 충돌 발생 시 Identifier 필드 값을 통해 우선순위를 결정하고, 수신 시 Identifier 필드 값이 작을수록 높은 우선순위를 가지게 된다.[2]

2.2 공격 유형 분류

본 논문에서 분류한 공격 유형은 Spoofing, Fuzzing, Flooding, Replay 이며, 아래 Table.1로 나타내었다.[3]

Attack	설명
Spoofing	- 차량의 특정 기능 제어 목적 - 비정상적인 CAN 메시지 주입
Fuzzing	- 무작위 메시지 주입 - 차량의 예기치 않은 동작 유발
Flooding	- 대량의 메시지 전송 - CAN Bus 대역폭 소모 목적
Replay	- 정상적인 트래픽 추출 - 추출한 트래픽 그대로 차량에 주입 - 정당한 사용자로 가장

Table.1 Attack Type

III. 피처 추출

본 논문은 CAN 데이터의 순서와 주기성을 고려하여 총 9개의 피처를 추출했다. 기존 데이터의 Arbitration ID (AID)와 Byte Data는 10진수로 변환하여 사용하였다.

기존의 CAN 데이터 분석은 주로 AID와 Data를 기반으로 이루어졌다[4]. 우리는 추가적으로 CAN 데이터의 DLC도 활용하여 각각의 Time Interval을 추출한 뒤 성능을 비교하였다. 실험 결과, AID와 DLC를 기준으로 추출한 피처에서 높은 탐지 및 분류 성능을 보여 이를 포함한 총 9개의 피처를 추출하였다.

본 논문에서는 해당 피처들을 적용하여 정상 및 4가지 공격 데이터에 대한 탐지 및 분류 성능을 향상시켰다. 추출한 피처는 Sequence와 Time Interval 계열로 구분되며 추출 방법은 아래와 같다.

3.1 AID Sequence

현재 CAN 데이터를 기준으로 하여 이전에 나온 5개의 AID를 Sequence로 추가하였다. 해당 Sequence 피처를 통해 CAN 데이터의 순서에 대한 정보를 제공하게 된다.

3.2 Time Interval Based Feature

AID와 DLC를 고려하여 총 4개의 Time Interval 피처를 추출했다. Fig1.은 Time Interval 계열 피처 추출 방식을 나타낸다. 먼저, Time Interval은 현재 데이터와 직전 데이터의 Timestamp 차를 계산했다. 나머지 3개의 피처는 아래와 같이 추출하였다.

3.2.1 AID Gap Time Interval

동일한 AID를 가지는 데이터끼리의 Time Interval을 구하였다. Fig.2-(a)와 같이, 현재 AID와 동일한 값을 가지는 이전 데이터가 있다면, 현재 데이터와 동일한 AID를 가지는 이전 데이터의 Timestamp 차를 계산하였다.

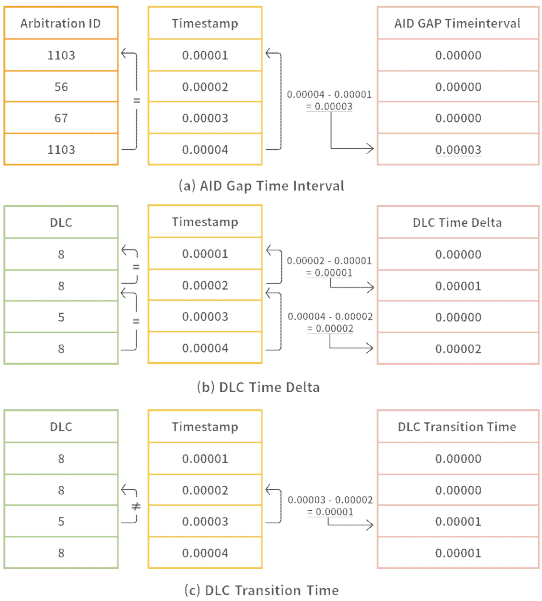

Fig.2 Time Interval Based Feature Extraction

3.2.2 DLC Time Delta

동일한 DLC를 가지는 데이터 간의 Time Interval을 계산하였다. Fig.2-(b)와 같이, 현재 DLC가 8일 때, 이전 데이터 중 DLC가 8인 직전 데이터와의 Timestamp 차를 구하였다.

3.2.3 DLC Transition Time

DLC 값이 변할 때, 이전 데이터와 현재 데이터 간 Time Interval을 계산하였다. Fig.2-(c)처럼 DLC가 변하면 DLC가 변경된 데이터와 직전 데이터 간의 Timestamp 차를 구하였다.

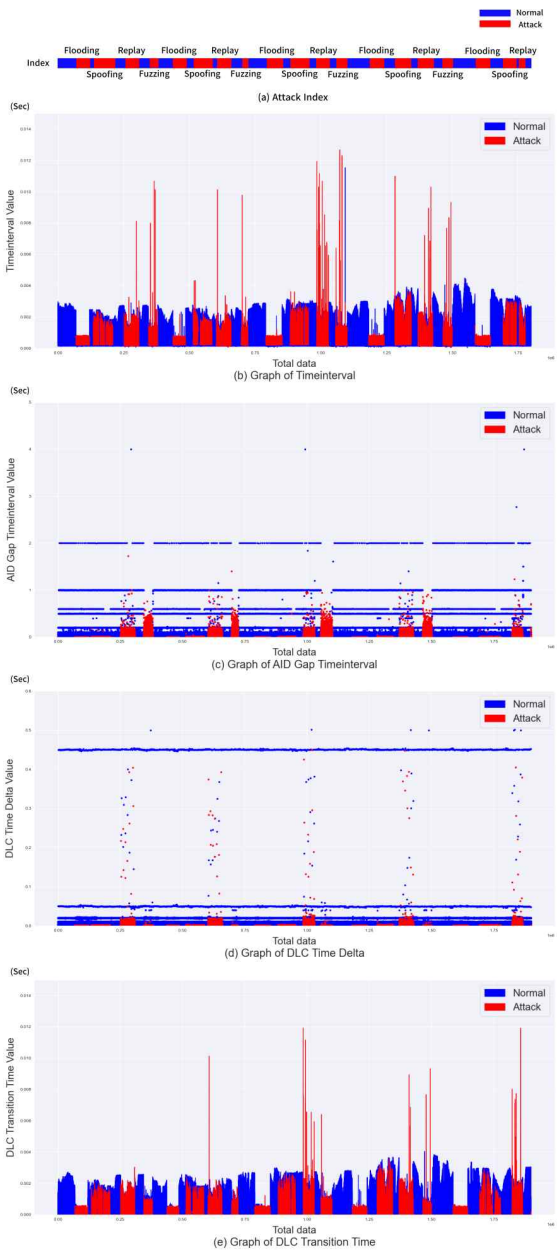

Fig.3 Graph of Time Interval Based Feature

Fig.3은 각 Time Interval 계열 피처 값들을 정상과 공격이 섞여있는 전체 데이터에 대해 나타낸 그래프이다. Fig.3-(a)은 각 공격들에 대한 위치를 나타낸다. 해당 그래프를 살펴보면, 정상과 4가지 공격 사이의 피처 값 변화를 명확하게 확인할 수 있다. 특히, Fig.3-(b)와 Fig.3-(e)에서 해당 피처 값들이 Replay 공격 패턴을 두드러지게 나타남을 확인하였다.

IV. 실험 및 평가

4.1 HCRL 데이터셋

본 논문에서는 침입 탐지 실험을 위해 여러 공격 데이터가 포함되어 있는 HCRL (Hacking and Countermeasure Research Lab) CAR HACKING : ATTACK & DEFENSE CHALLENGE 2020 DATASET 을 사용하였다[3][5]. 해당 데이터셋은 주행 중인 Hyundai Avante CN7 차량에서의 정상, Spoofing, Fuzzing, Flooding, Replay 공격이 적용되었다. 본 논문에서의 실험을 위해 해당 데이터셋을 8:2 비율로 랜덤하게 분할하여 모델의 학습 및 테스트에 적용하였다.

4.2 데이터 전처리

AID와 Byte data를 10진수로 변환한 기존 데이터에 각 피처를 추가하여 실험을 진행하였다. 입력 데이터에 MinMax 정규화를 적용한 뒤, XGBoost 모델을 사용하여 데이터를 분류하였다. 이때, n_estimators 3000, 0.6의 학습률, max_depth 9, colsample_bytree 0.78로 모델 파라미터를 설정하였다.

4.3 성능지표

정상 데이터와 4가지의 공격 데이터에 대한 분류 성능을 확인하기 위해 혼동행렬을 사용하였다. 혼동행렬에서 공격 데이터를 공격으로 탐지했을 때를 True Positive (TP)로 설정하였으며, 정상 데이터를 정상으로 탐지했을 때를 True Negative (TN)로 설정하였다. 혼동행렬을 기반으로 정확도(Accuracy), 정밀도 (Precision), 재현율 (Recall), F1-score를 계산하였다.

4.4 실험 환경

본 실험은 Window 10, Intel(R) i5-12600KF CPU, RAM 32GB, NVIDIA GeForce RTX 3070Ti 에서 수행하였다.

4.5 실험 결과

본 논문에서 적용한 Sequence 및 Time Interval 계열 피처들의 정확도와 F1-Score를 Table.2 에 나타내었다.

4.5.1 단일 피처 적용

Table.2를 살펴보면, 기존 데이터(AID, Byte Data)만 모델의 입력 데이터로 설정했을 때보다 Sequence나 Time Interval 계열의 피처를 단일로 추가했을 때 높은 성능을 보였다. 특히, AID Sequence 혹은 Time Interval 피처를 추가했을 때, F1-Score가 97.25%, 96.95%로 가장 높은 성능 향상을 보였다.

Feature	Accuracy (%)	F1-Score (%)
AID, Byte Data	99.04	92.66
AID Sequence	99.58	97.25
Time Interval	99.56	96.95
AID Gap Timeinterval	99.39	95.84
DLC Time Delta	99.46	96.04
DLC Transition Time	99.25	94.49
Total Feature	99.88	99.27

Table.2 Accuracy and F1-Score by Feature

4.5.2 종합 피처 적용

본 논문에서 추출한 총 9개의 피처를 모두 적용한 실험 결과, 기존 데이터를 사용했을 때에 비해 향상된 99.88%의 정확도를 얻었다. 특히, Replay 공격에 대한 F1-score가 기존 70.54%에서 97.19%로 크게 향상되었다.

V. 결론

본 논문에서는 차량의 CAN bus에서 발생할 수 있는 4가지 공격 유형에 대한 침입 탐지 및 분류 방법을 제안하였다. CAN 데이터의 순서와 주기성을 고려하여 9개의 피처로 분류 정확도를 향상시켰으며, 해당 피처들은 Sequence 및 Time Interval 기반으로 추출하였다. 또한, 기존 데이터에 개별적으로 적용하여 각 피처의 성능을 평가하였는데, 그 결과 AID Sequence와 Time Interval 피처가 가장 우수한 성능을 나타냈다. 이렇게 추출된 9개의 피처를 종합적으로 추가하여 실험한 결과, 분류 정확도 및 Replay 공격에 대한 F1-score가 크게 향상되었음을 확인했다. 이를 통해 제안된 피처 기반 방법이 유용하게 작용함을 입증하였다.

[참고문헌]

[1] ISO 11898-1:2015 Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling (최종 열람일: 2023년 8월 16일) https://www.iso.org/standard/63648.html

[2] 조아람, 조효진, 우사무엘, 손영동, 이동훈.(2012).CAN 버스 공격에 안전한 메시지 인증 및 키 분배 메커니즘.정보보호학회논문지,22(5),1057-1068.

[3] HCRL, “CAR HACKING: ATTACK & DEFFENSE CHALLENGE 2020”, HCRL, 2020 (최종 열람일: 2023년 8월 16일) https://ocslab.hksecurity.net/Datasets/carchallenge2020

[4] H. M. Song, H. R. Kim and H. K. Kim, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network," 2016 International Conference on Information Networking (ICOIN), Kota Kinabalu, Malaysia, 2016, pp. 63-68, doi: 10.1109/ICOIN.2016.7427089.

[5] Hyunjae Kang, Byung Il Kwak, Young Hun Lee, Haneol Lee, Hwejae Lee and Huy Kang Kim. "Car Hacking and Defense Competition on In-Vehicle Network." Third International Workshop on Automotive and Autonomous Vehicle Security, 2021.