Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.32, NO.2, Apr. 2022 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2022.32.2.193

머신러닝을 활용한 알려지지 않은 암호통신 프로토콜 식별 및 패킷 분류*

구 동 영†‡
한성대학교 (교수)

Identification of Unknown Cryptographic Communication Protocol and Packet Analysis Using Machine Learning*

Dongyoung Koo†‡
Hansung University (Professor)

요 약

알려지지 않은 암호통신 프로토콜은 개인 및 데이터 프라이버시를 보장한다는 장점이 있을 수 있으나, 악의적 목적에 사용될 경우 기존의 네트워크 보안 장비를 이용하여 이를 식별하고 대응하는 것이 불가능에 가깝다. 특히, 실시간으로 오가는 방대한 양의 트래픽을 수작업으로 분석하는 데에는 한계가 존재한다. 따라서, 본 연구에서는 머신러닝 기법을 활용하여 알려지지 않은 암호통신 프로토콜의 패킷 식별과 패킷의 필드 구분을 시도한다. 순차 패턴과 계층적 군집화, 그리고 피어슨 상관계수를 활용하여 알려지지 않은 암호통신 프로토콜이라 하더라도 그 구조를 자동화하여 분석할 가능성을 확인한다.

ABSTRACT

Unknown cryptographic communication protocols may have advantage of guaranteeing personal and data privacy, but when used for malicious purposes, it is almost impossible to identify and respond to using existing network security equipment. In particular, there is a limit to manually analyzing a huge amount of traffic in real time. Therefore, in this paper, we attempt to identify packets of unknown cryptographic communication protocols and separate fields comprising a packet by using machine learning techniques. Using sequential patterns analysis, hierarchical clustering, and Pearson's correlation coefficient, we found that the structure of packets can be automatically analyzed even for an unknown cryptographic communication protocol.

Keywords: Cryptographic protocol, Packet analysis, Sequential Pattern, Hierarchical Clustering, Pearsons's Correlation Coefficient

I. 서 론

알려지지 않은 네트워크 트래픽은 악의적 목적 활용 등 보안상 문제를 일으킬 수 있어 네트워크 건전성 유지를 위해서는, 네트워크상에서 오가는 패킷에 대한 보안성 검토가 필수적이다. 알려지지 않은 네트워크 프로토콜을 따르는 패킷을 분석하는 기법을 프로토콜 역공학(Protocol Reverse Engineering, PRE)이라 하는데, 전송되는 데이터가 사람이 읽을 수 없는 바이너리 프로토콜과 같은 다수 요인으로 인

Received(01. 12. 2022), Modified(02. 08. 2022), Accepted(02. 23. 2022)
* 본 연구는 한성대학교 교내학술연구비 지원과제임

† 주저자, dykoo@hansung.ac.kr
‡ 교신저자, dykoo@hansung.ac.kr(Corresponding author)

하여 전문가의 수작업에 의한 분석은 효율성 저하의 제약이 있다. 따라서 대용량 트래픽에 대한 수동 분석의 효율성 저하 문제를 개선하고 정확성과 효율성을 높이기 위한 머신러닝 기법을 이용한 트래픽 분석 기술의 자동화 연구가 활발히 이루어지고 있다 [1,2,3,4]. 특히 암호통신 프로토콜의 사용 비중이 지속적으로 증가하는 상황에서 송수신되는 데이터의 내용을 알 수 없는 암호통신의 특성으로 인하여 불법 및 탈법 행위에 악용되는 위험을 방지하기 위한 분석 및 대응 기술의 필요성이 보다 강조되고 있다.

본 연구에서는 구조가 공개되지 않은 암호통신 프로토콜의 패킷 구조 분석을 목적으로 하며, 패킷 내 바이트 단위 상관관계 분석을 통하여 필드 경계를 파악하고자 한다.

II. 관련 연구

2.1 암호통신 프로토콜 패킷

패킷은 네트워크에서 데이터 전송을 위해 일정 크기로 자른 전송 단위로, 일반적으로 헤더(header), 페이로드(payload), 트레일러(trailer)로 구성된다. 패킷의 헤더에는 주요 제어 정보들이 포함되고, 페이로드에는 데이터가 담겨 전송되며, 트레일러는 오류 검출 등에 사용된다. 각 패킷은 필드로 구분되어 있는데, 필드 구분을 위하여 동일 필드 내에 있는 바이트들은 유사한 패턴을 지니고 있다고 가정한다.

본 연구에서는 다수의 암호통신 프로토콜 중에서 표준화가 진행되고 있는 QUIC(Quick UDP Internet Connections)의 특정 버전인 gQUIC Q046 [5]을 알려지지 않은 프로토콜로 가정하여 해당 패킷을 학습시키지 않고 분류에 이용함으로써, 암호프로토콜 패킷의 분류 및 분석을 시도한다.

QUIC은 TCP 기반 암호통신의 지연한계를 극복하기 위하여 UDP 기반으로 개발된 HTTP(Hypertext Transfer Protocol)의 세 번째 메이저 버전으로, [그림 1]과 같이 Google에 의하여 주도적으로 개발되며 HTTP-over-QUIC으로 시작하였으나 2018년 11월 HTTP/3으로 명칭이 변경되며 IETF에 의하여 표준화가 진행되고 있다.

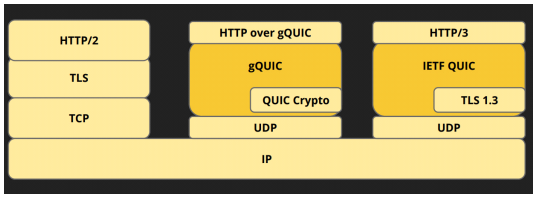


Fig. 1. Comparison of HTTP/2 and QUIC

2.2 암호통신 프로토콜 패킷 분석의 자동화

Chen et al. [6]은 수작업에 의한 특징 추출 및 오프라인 분석의 한계점을 극복하기 위하여 서로 다른 애플리케이션의 정적/동적 행동 특성(시계열 정보)을 RKHS(Reproducing Kernel Hilbert Space)를 이용하여 다중 채널 이미지로 변환한 후 CNN을 적용하여 트래픽 분석이 가능한 프레임워크를 개발하였다. 이와는 대조적으로 시계열 정보에 해당하는 네트워크 트래픽을 이미지 학습 등에 효과적으로 알려진 CNN에 적용함에 있어, 1차원 이미지 형태로 활용하는 연구 또한 Wang et al. [7]에 의하여 진행되었다. [7]에서는 암호통신 분류 과정에서 분할 정보 기반의 머신러닝을 활용할 때 국부 최적해를 구하는 제약을 해결하기 위하여 1차원 CNN을 활용하여 암호통신에서의 비선형 관계로 표현되는 사용자 행동을 특정하는 기법을 제시하였다. Lotfollahi et al. [8]은 머신러닝을 통한 학습에서 전문가의 수작업에 의존하는 매개변수 설정으로 인한 전반적인 특징 추출의 한계점을 지적하며, CNN과 SAE(Stacked Auto-Encoder)의 두 가지 딥러닝 기법을 활용하여 자동으로 가상사설망(Virtual Private Network, VPN)을 이용한 통신과 그렇지 않은 통신을 구별하는 Deep Packet 메커니즘을 제시하였다.

본 연구에서는 선행연구에서와 같은 사용자 행동 패턴 등에 의존하지 않고 프로토콜의 특징을 파악하기 위하여 머신러닝을 활용한 암호통신 패킷 구조 분석을 시도한다.

III. 제안 분석 기법

본 절에서는 알려지지 않은 암호통신 패킷 분석 기법을 단계별로 설명한다. 패킷 특성 탐색을 통한 프로토콜 분류를 위하여 계층적 군집화(Hierarchical Clustering) 기법을 사용하고 순차 패턴 분석(Sequential Pattern Analysis)과 패킷의 바이트 간 연관성을 피어슨 상관계수(Pearson's Correlation Coefficient)를 사용하

여 분석하고 시각화함으로써 패킷 내 필드를 구분하고 그 의미를 파악한다.

3.1 알려지지 않은 패킷 구조 분석 절차

다양한 프로토콜이 포함된 네트워크 트래픽에서 알려지지 않은 프로토콜에 속하는 패킷 분류를 수행하고 특정 프로토콜에 대한 패킷 구조 분석을 [그림 2]와 같이 4단계로 구분하여 진행된다.

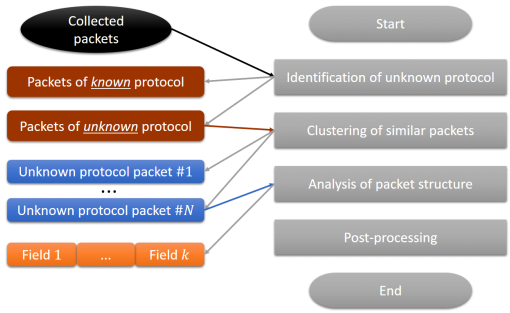


Fig. 2. Packet Analysis Overview of Unknown Protocol

3.1.1 프로토콜 선별

첫 단계에서는, 다수의 프로토콜 중 알려지지 않은 프로토콜만을 분류한다. 알려진 프로토콜을 따르지 않는 패킷의 계층적 특성을 고려하여 하위 계층의 헤더를 제외한 페이로드 부분만을 추출한다.

3.1.2 유사 패킷 분류

알려지지 않은 프로토콜 또한 다수의 유형이 존재하고 동일 프로토콜이라 하더라도 문맥에 따라 여러 형태의 패킷 구조를 가질 수 있다. 따라서 이 단계에서는 선별 과정을 거친 패킷을 계층적 군집화 기법을 적용하여 유사한 특징들을 가진 패킷들을 묶는다.

3.1.3 패킷 구조 분석

알려지지 않은 특정 프로토콜을 따르는 패킷 그룹의 구조 분석 단계로, 각 패킷을 구성하는 필드의 개별적인 특성을 분석하여 필드의 경계와 성질을 유추한다. 동일 구조의 패킷들은 바이트들 사이에 유사한 패턴을 나타낸다는 가정에서 해당 성질을 분석하여 패킷의 필드를 구분한다.

3.1.4 패킷 패턴 분석(후처리)

패킷의 특성별로 구분된 개별 패킷들의 흐름을 종합하여 특정 패킷이 지니는 의미를 파악한다.

본 연구에서는 유사 패킷 분류와 패킷 간 특성 분석을 통한 패킷 구조 분석에 초점을 둔다.

3.2 유사 패킷 분류

알려지지 않은 여러 프로토콜을 따르는 패킷들을 프로토콜별로 분류하기 위하여, 패킷 구조 분석의 전 단계로 다른 형태의 구조를 가지는 여러 패킷 중에서 유사 패턴을 보이는 패킷을 그룹화한다.

패킷에 대한 사전정보가 없는 상태에서 다른 형태의 구조를 가진 패킷들을 분류하기 위하여 계층적 군집화 기법을 적용한다. 계층적 군집화는 상향식 접근법(bottom-up approach)을 취하며 비슷한 유형을 군집으로 묶어 가면서 최종 하나의 군집으로 통합될 때까지 결합하는 과정을 반복하는 알고리즘으로, k-Means 군집화와 달리 사전에 군집의 수를 정할 필요가 없으므로 패킷의 필드 구조를 모르는 상황에서도 적용이 용이한 장점이 있다. 계층적 군집화에서 서로 다른 패킷 간의 계층적 분류는 계통도(Dendrogram)를 통하여 확인할 수 있는데, 데이터 배열의 인덱스 사이 거리는 데이터의 위치 차이를 나타내고 높이는 군집 사이의 거리를 의미한다.

3.3 패킷 구조 분석

프로토콜 규격에 대한 사전정보가 없는 상태에서 패킷 내 필드를 구성하는 바이트들은 공통된 패턴을 보일 것이라는 가정하에, 순차 패턴을 사용한 패킷 내 필드 구분을 시도한다. 또한, 패킷을 구성하는 바이트 사이에서의 유사도에 기반한 계층적 군집화를 적용하여 동일 필드가 동일 군집으로 분류될 확률이 높음을 확인하고 그 결과를 다시 피어슨 상관계수를 이용하여 유사도를 측정하여 검증한다.

IV. 실 험

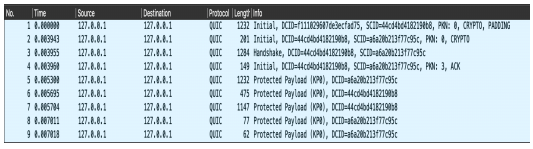
본 제안 기법의 검증을 위한 간이 실험을 위하여 널리 알려진 네트워크 프로토콜 분석기인 Wire

Shark [9]에 의하여 식별되지 않는 암호통신 패킷을 분석대상 데이터셋으로 설정하였다.

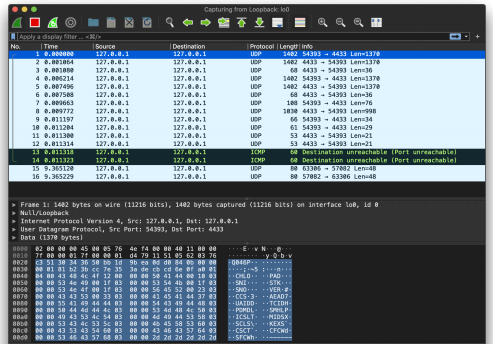
4.1 데이터셋

실험에 사용한 데이터셋은 알려진 표준 암호통신 프로토콜인 TLSv1.2, TLSv1.3과 IETF QUIC defat-27 및 gQUIC Q046 암호통신 프로토콜을 따라 송수신되는 패킷을 수집하여 분석하였다.

[그림 3]과 같이 국제표준화기구에 의하여 개발중인 IETF QUIC draft-27은 WireShark에 의하여 프로토콜 식별이 가능하나, Google에 의해 제안된 QUIC 프로토콜(gQUIC)의 Q046 버전은 실험 당시 WireShark에 의하여 단순 UDP 패킷으로 분류되어 암호통신 프로토콜의 식별이 정확히 이루어지지 않는 것을 알 수 있다. 따라서, 본 실험에서는 gQUIC Q046 패킷을 알려지지 않은 암호통신 패킷으로 가정하여 패킷 분류 및 구조 분석을 수행하였다. [그림 4]와 같이 세션을 생성하는 초기에만 long header를 가진 패킷이 수집되고 이후에는 데이터 송수신을 위한 short header를 가진 패킷만이 수집되었기에, gQUIC Q046 프로토콜을 따르는 패킷이라 하더라도 long header와 short header를 가진 패킷 수집 비율에서는 차이가 발생하였다.

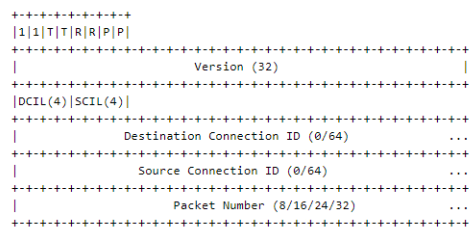


(a) IETF QUIC-draft 27

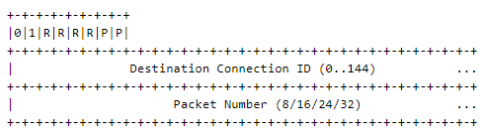


(b) Google QUIC (gQUIC) Q046

Fig. 3. Protocol identification by WireShark IETF QUIC vs. gQUIC



(a) Long header format of gQUIC Q046



(b) Short header format of gQUIC Q046

Fig. 4. gQUIC Q046 header format

4.2 프로토콜 선별

알려지 않은 암호통신 프로토콜은 네트워크 프로토콜 분석기인 WireShark의 분석기능을 활용한 tshark의 파이썬(Python)용 래퍼 라이브러리인 PyShark [9]를 이용하여 식별하였다.

패킷 헤더의 주요 제어 정보들을 이용하여 전송되는 데이터의 종류, 형태, 패킷의 출발지 및 도착지 주소에 따라 분류하되, 알려진 프로토콜로 구분되지 않은 모든 패킷을 다음 단계의 분석대상으로 선정하였다. 분류된 패킷들은 하위 계층의 헤더를 제거하여 획득한 각 패킷의 페이로드를 바이트 단위의 1차원 배열로 변환하여 이후 학습에 활용하였다.

4.3 유사 패킷 분류

프로토콜 선별 단계에서 WireShark에 의하여 식별되지 않은 프로토콜은 gQUIC Q046 뿐이었으며, 단일 세션에서 획득한 long header 32개와 short header 279개의 총 311개 패킷에 대하여 유사 패킷 분류를 시도하였다.

알려지지 않은 프로토콜을 따르는 패킷들을 전처리하여 1차원 배열 형태로 변환한 후, 단일 연결을 사용하여 계층적 군집화를 적용한 결과는 [그림 5]와 같다. 좌측 노란색으로 표현된 군집과 우측의 초록색으로 표현된 군집으로 구분됨을 확인할 수 있는데, 좌측 노란색 군집은 long header를 가진 패킷 32개와 일치하였고, 우측 초록색 군집은 279개의 short header를 가진 패킷과 일치하였다. 이처럼 서로 다른 패킷 개수에 영향을 받지 않고, 계층적 군

집화를 통하여 상이한 암호통신 규격에 따르는 패킷들의 분류가 가능함을 알 수 있다.

계층적 군집화는 패킷 그룹의 개수를 정함에 있어, k-Means와 달리 군집화가 이루어진 후에 분류된 그룹의 개수를 달리할 수 있는 장점이 있다. 본 실험에서는 최종 결과로부터 2개의 그룹인 gQUIC Q046의 long header 및 short header를 가지는 패킷으로 분류하였으나, 이를 일반화시켜 다수의 알려지지 않은 프로토콜 분류에서는 계층적 군집화의 결과 계통도 등을 활용하여 매개변수를 조절하여 최적의 군집 개수를 추정할 수 있을 것이다.

4.4 패킷 구조 분석

4.4.1 순차 패턴을 이용한 구조 분석

전 단계에서 동일 프로토콜을 따르는 패킷으로 분류된 바이트 단위 1차원 배열로 표현된 패킷의 페이로드를 Pei et al. [10]이 제시한 PrefixSpan을 구현한 순차 패턴에 적용하였으며, 그 결과는 [표 1]과 같다: 하위 계층의 헤더를 제거한 long header 패킷을 입력으로 하였을 때 높은 빈도수의 바이트를 나타내는 배열의 값은 [81, 48, 52, 0, 2, 54, 80, 195]였으며, 해당 바이트가 나타나는 배열의 인덱스는 [0, 1, 2, 3, 4, 5, 14, 17]로 long header의 플래그, 버전, connection ID의 길이, 패킷 번호를 나타내는 배열의 인덱스에 해당한다. 이로부터 long header에서 고정 크기를 가지는 필드는 순차 패턴으로 구분할 수 있었다. 하위 계층의 헤더를 제거한 short header 패킷을 입력으로 한 경우에도, 빈도수 높은 바이트를 나타내는 배열 값이 [64, 8, 80, 97, 86, 125, 126, 128, 144]이며, 해당 바이트가 나타나는 배열의 인덱스는 [0, 1, 2, 3, 4, 5, 6, 7, 8]로 short header의 플래그, destination connection ID, 패킷 번호를 나타내는 인덱스에 해당한다. 이로부터 short header에서도 고정 크기를 가지는 필드는 순차 패턴을 적용하여 필드의 구분이 가능함을 알 수 있다.

4.4.2 피어슨 상관계수를 이용한 구조 분석

순차 패턴을 이용한 필드 구분과 독립하여 패킷의 필드를 구분하기 위한 시도로 피어슨 상관계수를 활용하였다. 피어슨 상관계수는 두 변수 X와 Y의 선형 상관관계를 계량화한 수치로, +1과 -1 사이의 값을 가지는데 +1은 완벽한 양의 선형 상관관계를 의미하고, 0은 선형 상관관계가 없음을, -1은 완벽한 음의 선형 상관관계를 나타내는데 두 변수 X, Y의 공분산 $cov(X, Y)$, 변수 X의 표준편차 ρ_X, 변수 Y의 표준편차 ρ_Y에 대하여

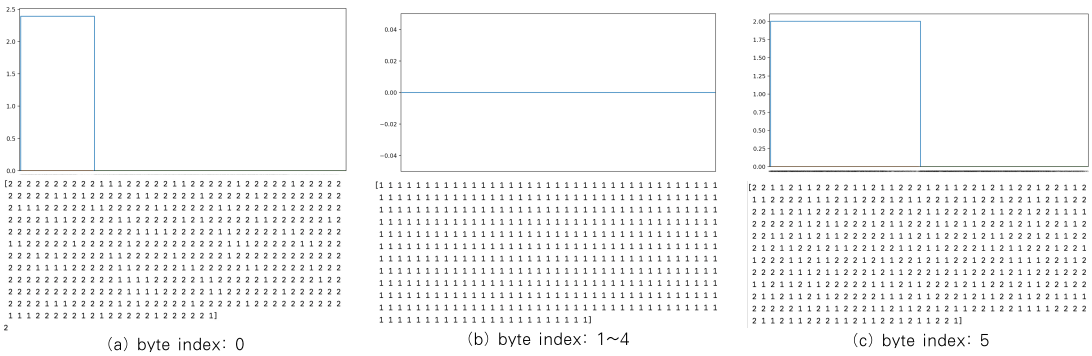
$$\rho_{X,Y} = \frac{cov(X, Y)}{\rho_X \cdot \rho_Y} \quad (1)$$

로 계산되며, 피어슨 상관계수의 입력은 같은 길이의 패킷을 사용한다.

피어슨 상관계수의 계산에 앞서 동일 프로토콜로 분류된 모든 패킷들을 각각의 바이트에 대하여 계층적 군집화를 적용하였다. gQUIC Q046의 long header에 속하는 패킷 430개를 여러 세션으로부터 획득하여 계층적 군집화를 적용한 결과는 [그림 5]와 같다. 상단은 군집화 결과로 생성된 계통도이며,

Table 1. Separation of Fields in Packets with Sequential Pattern Analysis

```
# sequential pattern (SP) analysis
from prefixspan import PrefixSpan
ps = PrefixSpan(data)
ps.minlen = 2 # minimum size of array
b = ps.topk(100) # top 100
# put result of SP into 1-dimmensional array
s=[]
for i in range(len(b)):
for j in range(len(b[i][1])):
s.append(b[i][1][j])
# remove redundant value
list = []
for v in s:
if v not in list:
list.append(v)
print(list)
# save most-frequent index after searching
  the value from previous step
point = []
index = [[] for i in range(len(list))]
for j in range(len(list)):
for i in range(len(data)):
for p in enumerate(data[i]):
if(p[1] == list[j]):
index[j].append(p[0])
point.append(Counter(index[j]).most_common(1)[0][0])
# sort indices according to the size
point = sorted(point)
```



(a) byte index: 0 (b) byte index: 1~4 (c) byte index: 5

Fig. 5. Results of hierarchical clustering on each byte of the packets in the same protocol(Upper: resulting dendrogram, Lower: cluster number of each packet)

하단은 각 패킷의 해당 인덱스 바이트가 속하는 군집 번호이다. 〔그림 5〕의 (a)는 각 패킷의 첫 번째 바이트에 대한 분류로 long header의 플래그에 해당하는 부분이며 송신 및 수신에 따라 값이 달라져 2개의 군집으로 분류되었다. 〔그림 5〕의 (b)는 각 패킷의 두 번째부터 다섯 번째까지 공통된 결과를 보였는데, gQUIC의 버전이 기록된 부분으로 1개의 군집으로 분류되었다. 〔그림 5〕의 (c)는 각 패킷의 여섯 번째 바이트로 송수신 연결 식별번호를 나타내는 부분이기 때문에 송신과 수신에 따라 2개의 군집으로 분류되었다. 군집화 과정에서 임의 부여된 값으로 유사도 분석의 용이성을 위하여 군집 번호가 단조 증가하도록 재정렬한 후, 피어슨 상관계수를 계산하였다.

피어슨 상관계수를 gQUIC Q046 패킷에 적용한 결과는 〔그림 6〕과 같다. 동일 길이로 가공된 패킷을 계층적 군집화를 거친 후 유사도가 높은 인접 바이트를 결합하여 필드 경계를 구분할 수 있는데, 바이트 인덱스를 이용하여 (0, 1~4, 5, 6~13, 14~16, 17, 18~)의 필드 경계를 가지도록 분류하였다. 〔그림 6〕의 오른쪽은 상관계수를 시각적으로 표현한 것으로 각 패킷의 인덱스를 기준으로 인접한 바이트의 유사도가 높을수록 노란색에 가까우며 유사도가 낮을수록 보라색을 띄며 유사성이 없는 경우에는 흰색으로 표현된다. 첫 번째 바이트는 유사도가 1로 모두 공통된 특징을 띄는 반면 두 번째에서 네 번째 바이트는 흰색으로 유사도인 것으로부터 첫 번째 바이트와 두 번째 바이트, 그리고 다섯 번째 바이트와 여섯 번째 바이트가 유사도가 낮아 필드의 경계로 여겨질 수 있다.

실제 프로토콜 규격과 비교하면, 인덱스 14~17의 바이트는 패킷 순서를 나타내는 동일 필드로 수집된 패킷에서의 통신 횟수가 적어 값이 모두 0으로

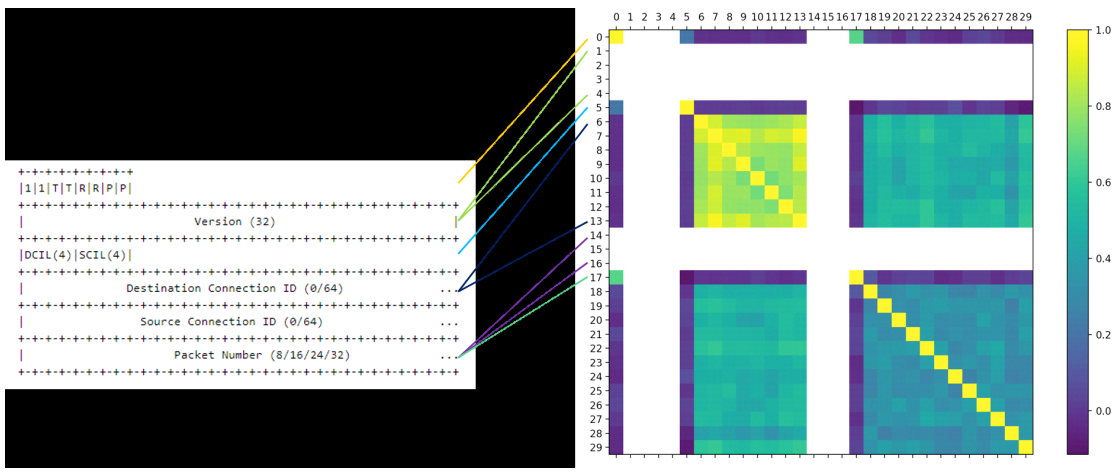


Fig. 6. Separation of Fields in Packets with Pearson's Correlation Coefficient

설정된 반면에 마지막 1바이트인 인덱스 17에 해당하는 바이트만 변경되어 다른 필드로 구분된 것을 알 수 있다. 알려지지 않은 프로토콜의 유형에 따라 상관계수의 편차는 있겠지만, 수치화된 상관계수의 임계(threahold) 값 조절을 통하여 필드 구분의 기준을 세울 수 있을 것이다.

V. 결 론

본 연구를 통하여 규격이 공개되지 않은 프로토콜에 대한 패킷 분석을 수행하였다. 실험에서는 gQUIC 046버전의 패킷을 사전 지식 없이 분석하여, 필드 경계를 찾아내고 구조를 분석하였으며, 수집된 패킷의 식별, 필드 구분을 자동으로 수행하는 프로토콜 역공학 자동화의 일환으로 머신러닝 기법을 활용하였다. 바이트 단위에서의 패킷 간 유사도를 계산하고 피어슨 상관계수를 활용함으로써 필드의 경계를 높은 확률로 구분할 수 있는 가능성을 확인하였다. 또한, 이를 확장한 비트 단위에서의 분류를 통한 분류 정확도를 향상시킬 수 있을 것이다. 알고리즘의 매개변수 최적화 등을 통하여 경계 구분의 오류를 줄이고 해당 필드의 의미까지 추론함에 있어서도 인공지능 기법의 활용이 가능할 것이며, 이를 통하여 프로토콜 역공학 기술 및 알려지지 않은 프로토콜을 대상으로 하는 공격 탐지 기술의 발전에 기여할 수 있을 것이다.

References

[1] J. Sherry, C. Lan, R.A. Popa and S. Ratnasamy, "BlindBox: Deep Packet Inspection over Encrypted Traffic," ACM Conference on Special Interest Group on Data Communication (SIGCOMM), pp. 213-226, Aug. 2015.

[2] C. Lan, J. Sherry, R.A. Popa, S. Ratnasamy and Z. Liu, "Embark: Securely Outsourcing Middleboxes to the Cloud," USENIX Symposium on Networked Systems Design and Implementation (NSDI), pp. 255-273, Mar. 2016.

[3] J. Fan, C. Guan, K. Ren, Y. Cui and C. Qiao, "SPABox: Safeguarding Privacy During Deep Packet Inspection at a MiddleBox," IEEE/ACM Transactions on Networking, vol. 25, no. 6, pp. 3753-3766, Oct. 2017.

[4] J. Ning, G.S. Poh, J. Loh, J. Chia and E. Chang, "PrivDPI: Privacy-Preserving Encrypted Traffic Inspection with Reusable Obfuscated Rules," ACM SIGSAC Conference on Computer and Communications Security (CCS), pp. 1657-1670, Nov. 2019.

[5] QUIC Versions, "QUIC Versions," *Internet Assigned Numbers Authority (IANA)*, https://www.iana.org/assignments/quic/quic.xhtml#quic-versions, [Referenced on] 03. 25. 2022.

[6] Z. Chen, K. He, J. Li and Y. Geng, "Seq2Img: A sequence-to-image based approach towards IP traffic classification using convolutional neural networks," International Conference on Big Data (BigData), pp. 1657-1670, Dec. 2017.

[7] W. Wang, M. Zhu, J. Wang, X. Zeng and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," International Conference on Intelligence and Security Informatics (ISI), pp. 43-48, Jul. 2017.

[8] M. Lotfollahi, M.J. Siavoshani, R.S. Zade and M. Saberian, "Deep packet: a novel approach for encrypted traffic classification using deep learning," Methodologies and Application, vol. 24, no. 3, pp. 1999-2012, May 2019.

[9] KimiNewt/pyshark, "PyShark", https://github.com/KimiNewt/pyshark, [Referenced on] 03. 25. 2022.

[10] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu, "PrefixSpan: Mining Sequential Patterns Efficiently by

Prefix-Projected Pattern Growth," International Conference on Data Engineering (ICDE), pp. 215-224, Apr. 2001.

〈저자소개〉

구 동 영 (Dongyoung Koo) 종신회원
2009년 2월: 연세대학교 컴퓨터.산업공학 졸업
2012년 2월: 한국과학기술원 전산학과 졸업 (공학석사)
2016년 2월: 한국과학기술원 전산학부 졸업 (공학박사)
2016년 3월~2017년 3월: 고려대학교 정보대학 컴퓨터학과 연구교수
2017년 4월~현재: 한성대학교 전자정보공학과 조교수
〈관심분야〉 정보보호, 암호 응용, 네트워크 보안, 클라우드/포크/엣지 컴퓨팅 보안