# 프로토콜 역공학 기반 정의되지 않은 프로토콜의 프레임 구조 인식 알고리즘 연구

박지환, 임완수

금오공과대학교 전자공학과(항공기계융합전공)

ehfdkdhswlgh@kumoh.ac.kr, wansu.lim@kumoh.ac.kr

# Research on Frame Structure Recognition Algorithm of Undefined Protocols Based on Protocol Reverse Engineering

Park Jihwan, Lim Wansu

Kumoh National Institute of Technology

## 요 약

본 논문은 프로토콜 역공학 기술을 이용하여 기존에 정의되지 않은 프로토콜의 프레임의 구조를 파악하는 알고리즘을 제안한다. 프레임 구조를 파악하기 위하여 먼저 계층적 군집화 기법을 이용하여 같은 형태의 프레임을 그룹화하고, 프레임을 구성하는 다양한 종류의 비트 시퀀스를 추출하기 위하여 프로토콜 역공학에서 사용하는 빈번한 시퀀스 탐색 알고리즘을 적용한다. 기 제안된 빈번한 시퀀스 탐색 알고리즘은 프레임을 구성하는 비트 시퀀스 길이의 다양성을 인정하지 않고 모든 비트 시퀀스의 길이를 고정한 후 추출하였으나, 본 논문은 다양한 길이의 비트 시퀀스를 추출할 수 있도록 알고리즘을 개선하였다. 마지막으로, 프레임을 구성하는 필드 값과 알고리즘에서 추출한 비트 시퀀스를 비교하여 프레임 구조 파악 정확도를 계산하여 제안한 알고리즘의 우수성을 증명하였다.

## Ⅰ. 서 론

정의되지 않은 프로토콜의 해석은 사전 정보가 없으므로 매우 어려운 과정이다. 프로토콜 역공학은 이러한 정의되지 않은 프로토콜의 구조와 작동 방식을 분석하여 이해하는 과정이다. 하지만 프로토콜 역공학의 복잡성은 상당하다. 예를 들어, SAMBA 프로젝트는 마이크로소프트의 서버 메시지 블록(SMB) 프로토콜에 대한 프로토콜 명세를 생성하는 데 12년이 걸렸다[1].

정의되지 않은 프로토콜의 구조와 작동 방식을 이해하는 데에 프레임에서 빈번하게 등장한 시퀀스를 찾는 것은 프로토콜 역공학에서 매우 중요한 과정이다. 빈번한 시퀀스를 찾는 것은 프레임 내의 반복되는 패턴이나 구조를 파악하는 데 도움이 되며, 이를 통해 프로토콜의 특성을 이해하거나 특정 필드의 의미나 값의 범위를 추론할 수 있다. 이렇게 얻은 정보는 정의되지 않은 프로토콜의 구조를 밝혀내는 데 기여할 수 있다.

프로토콜 역공학은 크게 Network trace와 Execution trace를 입력으로 활용하며, 그 결과는 PFSM (Protocol Finite State Machine)과 PF (Protocol Format)로 분류될 수 있다[1]. Network trace는 Tcpdump, Wireshark, 그리고 Microsoft 네트워크 모니터 등의 도구로 캡처한 네트워크 트래픽의 원시 패킷(raw packets)을 의미한다. 반면에 Execution trace는 응용 프로그램에서 실행된 프로그램 코드를 의미하며, 이는 통신 호스트 간에 실행된 단일 실행 과정을 추적한다. PFSM은 메시지가 두 호스트 간에 어떻게 전환되는지에 대한 상태와 순서를 정의하며, PF는 각 필드가 가진 고유한 의미론적 키워드와 필드 경계가 구조화된 방식을 나타낸다. 프로토콜 역공학에서의 주요 접근법으로는 Discoverer[2]와 Polyglot[3]이 있다. Discoverer는 Network trace를 입력으로 받아 PF를 도출하는 데 중점을 두는 반면, Polyglot는 Execution trace를 입력으로 받아 PF를 도출한다.

본 논문은 Network trace의 일종인 Pcap 데이터를 입력으로 사용하여 정의되지 않은 프로토콜의 프레임에서 PF를 도출하는데 필요한 빈번한 시퀀스를 찾는 새로운 방법론을 제안한다. 기존의 접근법과 비교했을 때, 제안하는 방법은 가변 길이의 시퀀스를 성공적으로 추출할 수 있음을 보여준다. 또한, 빈번한 시퀀스를 탐색하기 전에 계층적 군집화 방법을 적용하여 유사한 형태의 프레임들을 그룹화 함으로써, 다양한 프로토콜의 프레임 데이터를 처리할 수 있게 되었다.

## Ⅱ. 본론


그림 1. 정의되지 않은 프로토콜의 프레임에서 빈번한 시퀀스 탐색 개요

Fig. 1. Finding Frequent Sequences Overview in Frames of Undefined Protocols

그림1은 다양한 프로토콜이 존재하는 네트워크 트래픽에서 사전에 정의되지 않은 프로토콜을 추출하고, 그 프로토콜에서 사용하는 프레임을 구성하는 다양한 시퀀스를 찾는 과정이다.

본 연구에서 사용된 데이터 셋은 TCP 프레임 194개, ARP 프레임 217개, GQUIC Q043 프레임 41개, 그리고 TLS 프레임 486개를 정의되지 않은 프로토콜의 프레임으로 가정하고 구성하였다. QUIC는 HTTP/3를 위한 전송 프로토콜로, TCP의 지연 문제를 해결하기 위해 UDP를 기반으로 개발되었다[4]. 입력 데이터는 네트워크 프로토콜 분석 도구인 WireShark의 기능을 활용하는 tshark의 Python 래퍼 라이브러리인 PyShark를 이용해 Pcap 데이터를 로드하고, 각 프레임의 16진수 문자열을 추출하여 저장한다.

다양한 프로토콜을 사용하는 프레임들을 프로토콜별로 분류하기 위해, 빈번한 시퀀스 탐색 알고리즘에 앞서, 서로 다른 구조를 지닌 다양한 프레임 중 유사한 패턴을 보이는 프레임들을 그룹화 해야 한다. 사전에 프레임에 대한 정보가 없는 상태에서 이러한 분류를 수행하기 위해 계층적 군집화

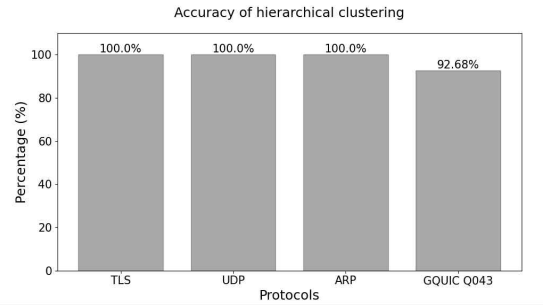


그림 2. 계층적 군집화의 정확도

Fig. 2. Accuracy of Hierarchical Clustering

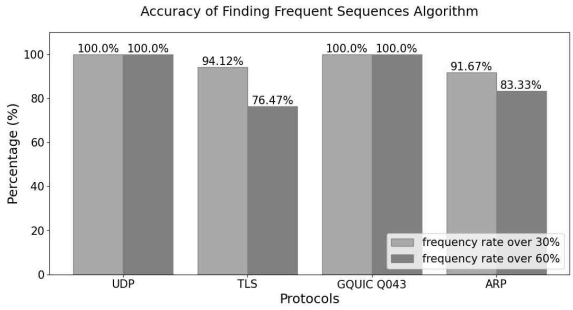


그림 3. 빈번한 시퀀스 탐색 알고리즘의 정확도

Fig. 3. Accuracy of Finding Frequent Sequences Algorithm

(Hierarchical Clustering) 방법을 적용하였다. 계층적 군집화는 상향식 접근법을 사용하여 유사한 유형의 그룹을 군집으로 만들어 나가는 알고리즘이다. 이는 k-Means 군집화와 달리 군집의 수를 사전에 정해 둘 필요가 없는 장점이 있어, 프레임의 구조를 모르는 상황에서도 적용 가능하다. 군집의 수는 직접 설정할 수도 있지만, 본 연구에서는 Silhouette 지표를 활용하여 최적의 군집 수를 자동으로 결정하는 방식을 구현하였다. 유사도 측정은 유클리드 거리를 사용하였다. 계층적 군집화를 수행하기 위해서는 모든 프레임의 크기가 동일해야 하는데, 실제 프레임들은 다양한 길이를 가지고 있다. 이 문제를 해결하기 위해, 프레임을 최소 길이 또는 최대 길이로 맞추거나, 프레임 길이의 중간값이나 평균값에 맞추어 0으로 패딩하거나 자르는 방식으로 크기를 통일하였다. 이렇게 조정된 프레임 크기별로 군집화 성능을 비교한 결과, 프레임 길이를 평균값에 맞추는 방식이 가장 높은 정확도를 보여 이를 사용하였다.

그림 2는 계층적 군집화 기법을 활용하여 데이터 셋에 대한 군집화의정확도를 나타내고 있다. 여기서 TLS, UDP, ARP 프레임들은 100%의 완벽한 정확도를 보여주며, 이는 해당 프레임들이 유사한 패턴을 지니고 있음을 시사한다. 반면, GQUIC Q043 프레임은 41개라는 상대적으로 적은 수의 프레임을 가지고 있어, 다른 프로토콜들에 비해 더 낮은 정확도인 92.68%를 나타내고 있다. 이는 해당 프레임의 수가 상대적으로 적어, 군집화에 있어서 더 큰 분산을 가져서 그럴 수 있다.

계층적 군집화를 통해 동일한 구조를 가진 프레임을 그룹화 한 후에, 각 그룹에 대해 빈번한 시퀀스 탐색 알고리즘이 실행된다. 이 알고리즘은 탐색할 빈번한 시퀀스의 최소 빈도율과 최소 길이를 설정하는 옵션을 제공한다. 이 알고리즘은 먼저 Sliding Window 알고리즘을 이용하여 설정한 최소 길이에 해당하는 빈번한 시퀀스들을 도출한다. 이후 각 시퀀스가 전체 프레임 중에서 어느 정도 비율로 출현하는지 계산하여 그 빈도를 파악한다. 이렇게 도출한 시퀀스가 설정한 최소 빈도율 이상이라면, 해당 시퀀스는 저장되게 된다. 시퀀스의 도출 과정에서, 동일한 빈도를 가지는 시퀀스가 연속적으로 출현하는 경우에는, 이들 시퀀스들을 병합하는 방식을 채택하였다. 이를 통해 기존에 제안된 시퀀스 추출 알고리즘[5]에서는 고정 길이의 시퀀스만을 도출하는 반면, 본 연구에서는 다양한 길이의 빈번한 시퀀스들을 추출할 수 있음을 보여준다.

추출된 빈번한 시퀀스의 성능 평가는 해당 시퀀스가 실제 프레임 필드영역에 속하는지를 기준으로 수행되었다. 그러나 Sequence Number, ACK Number, Data Offset, Checksum, Identification, Time to Live, Window Size, 그리고 Payload와 같은 필드들은 고유한 값을 가지는 특성 때문에 빈번한 시퀀스 생성이 불가능하다는 점을 고려하였다. 이러한 이유로, 이 필드들은 성능 평가 대상에서 제외하였다. 빈번한 시퀀스 탐색 알고리즘에서 최소 빈도율을 30% 이상, 그리고 60% 이상으로 설정한 경우의 결과를 구하여 성능을 평가하였다. 그림 3은 빈번한 시퀀스 탐색 알고리즘의 성능 평가 결과를 보여주고 있다. UDP와 GQUIC Q043 프레임은 모두 100%의 정확도를 보였으나, 이 결과는 일부 예외적인 상황을 반영한 것이다. 이 프레임들은 동일한 송수신자 간에 비슷한 유형의 프레임을 반복적으로 전송하는 특성을 가진 데이터 셋으로, 이러한 특성이 성능 평가에 유리한 결과를 나타내게 하였다. 그러나, 이러한 특정 상황은 일반적인 경우를 완전히 반영하지 못하므로, 100%의 정확도는 실제 알고리즘의 성능을 정확히 대변하지 않을 수 있다.

## Ⅲ. 결론

본 연구를 통해 정의되지 않은 프로토콜의 프레임에 대한 빈번한 시퀀스를 성공적으로 탐색하였다. 이러한 시퀀스들은 실제 프로토콜의 필드 영역에 속할 가능성이 높음을 확인하였다. 찾은 빈번한 시퀀스는 정의되지 않은 프로토콜의 구조를 이해하고 분석하는데 중요한 역할을 할 것이다.

## ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (2021R1I1A3056900).

## 참 고 문 헌

[1] Baraka, D. Sija, Goo, Young-Hoon, Shim, Kyu-Seok, Hasanova, Huru, Kim, Myung-Sup. "A Survey of Automatic Protocol Reverse Engineering Approaches, Methods, and Tools on the Inputs and Outputs View." Security and Communication Networks, vol. 2018, pp. 1-17, Feb. 2018.

[2] Cui, Weidong, Kannan, Jayanthkumar, Wang, Helen J. "Discoverer: Automatic Protocol Description Generation from Network Traces." USENIX Security Symposium, vol. 14, pp. 1-14, Aug. 2007.

[3] Caballero, J., Yin, H., Liang, Z., Song, D. "Polyglot: Automatic Extraction of Protocol Message Format Using Dynamic Binary Analysis." Proceedings of the 14th ACM Conference on Computer and Communications Security, pp. 317-329, Oct. 2007.

[4] Koo, D. "Identification of Unknown Cryptographic Communication Protocol and Packet Analysis Using Machine Learning." Journal of the Korea Institute of Information Security & Cryptology, vol. 32, no. 2, pp. 193-200, Apr. 2022.

[5] Wang, Y., Zhang, N., Wu, Y. -m., Su, B. -b., Liao, Y. -j. "Protocol Formats Reverse Engineering Based on Association Rules in Wireless Environment." 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, pp. 134-141, Jul. 2013.