C# 기반의 패킷 분석기 설계

조호 · 이민경 · 배지훈 · 정인묵 · 황소영 · 유동희

부산가톨릭대학교

Design of Packet analyzer Using C#

Ho Cho · Minkyoung Lee · Inmuk Jung · Jihoon Bae · Soyoung Hwang · Dong-Hui Yu

Catholic University of Pusan

E-mail : dhyu@cup.ac.kr

요 약

패킷은 데이터 전송에서 송신측과 수신측에 의하여 하나의 단위로 전송되는 집합체를 의미한다. 본 논문에서는 패킷 스니핑을 통해 네트워크 환경에서 송/수신되는 패킷들을 모니터링 함으로써 사용자 보안에 대한 취약점을 최소화하고, 패킷 전송 과정에서 사용자가 정의한 규칙에 의해 필터링하거나 네트워크 트래픽에 대한 통계 수집 등의 기능을 제공하는 C# 기반 패킷 분석기의 설계를 제안한다.

키워드

Packet, Analyzer, C#, Sniffing

Ⅰ. 서 론

패킷 스니핑은 네트워크 통신을 도청하는 행위이다. 이때 사용 되는 도구를 패킷 분석기(packet analyzer, network analyzer) 또는 패킷 스니퍼(packet sniffer, network sniffer) 라고 한다. 현재 유명한 패킷 분석기로 Wireshark, tcpdump, snoop 등이 있다. 그러나 사용법이 어려운 경우도 있으며 네트워크 초급자들이 알아보기에 힘든 경우가 있다.

본 논문에서는 C# 언어를 통해 윈도우 폼 기반으로 네트워크 초급자들과 정보보안이나 네트워크 통신을 공부하는 학생들이 사용하기 쉽고 알아보기 쉬운 패킷 분석기의 설계를 제안한다.

논문의 구성은 다음과 같다. 2장에서는 TCP/IP 데이터 구조와 패킷의 송수신 과정을 제시하고 3장에서는 제안한 분석기의 설계 및 구현에 대해 다룬다. 마지막 3장에서 논문의 결론을 맺는다.

Ⅱ. TCP/IP 데이터 구조와 패킷의 송수신 과정

(1) TCP/IP 데이터 구조

TCP/IP 프로토콜은 네 계층의 개념적인 모델에 맵핑 되어 있다. TCP/IP 프로토콜의 네 계층은 어플리케이션(Application), 프랜스포트(Transport), 인터넷(Internet), 네트워크 인터페이스(Network Interface) 계층이다.

그림 1은 TCP/IP 프로토콜 계층 구조를 보여준다.

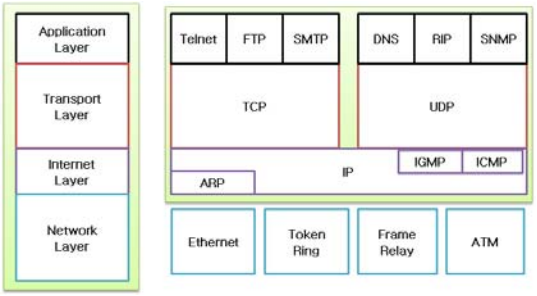

그림 1. TCP/IP 계층 구조

어플리케이션 계층은 다른 계층의 서비스에 접근할 수 있게 하는 어플리케이션을 제공하고 어플리케이션들이 데이터를 교환하기 위해 사용하는 프로토콜을 정의한다. 가장 많이 알려진 어플리케이션 계층 프로토콜에는 HTTP(HyperText transfer Protocol), FTP(File transfer Protocol), SMTP(Simple Mail transfer Protocol), Telnet (Terminal emulation Protocol)이 있다. 또한 TCP/IP 네트워크를 사용하거나 관리하는 것을 도와주는 DNS(Domain Name System), RIP(Routing Information Protocol), SNMP(Simple Network

Management Protocol)가 있다.

트랜스포트 계층은 어플리케이션 계층에 세션과 데이터그램 통신 서비스를 제공한다. 핵심 프로토콜은 TCP(Transmission Control Protocol), UDP(User Datagram Protocol)가 있다.

인터넷 계층은 어드레싱(addressing), 패키징(packaging), 라우팅(routing) 기능을 제공한다. 핵심 프로토콜로는 IP(Internet Protocol), ARP(Address Resolution Protocol), ICMP(Internet Control Message Protocol), IGMP(Internet Group Management Protocol)가 있다.

네트워크 인터페이스 계층은 TCP/IP 패킷을 네트워크 매체로 전달하는 것과 네트워크 매체에서 TCP/IP 패킷을 받아들이는 과정을 담당한다. OSI모델과 비교했을 때 네트워크 인터페이스 계층은 OSI 모델에서 데이터 링크 계층과 물리적 계층에 해당한다. TCP/IP는 서로 다른 네트워크 형태를 연결하는데 사용되어 질 수 있다. 여러 네트워크 형태로는 이더넷(Ethernet), 토큰링(Token Ring)과 같은 LAN기술과 X.25, 프레임 릴레이(Frame Relay)와 같은 WAN기술을 포함한다.

(2) 패킷 송수신 과정

그림 2와 그림 3을 보면 패킷 전송측과 수신측의 패킷 전송 형태를 볼 수 있다. 네트워크로 데이터가 보내질 때는 데이터가 스택에서 아래로 전해지고, 네트워크에서 받을 때는 스택의 위로 전해진다. TCP/IP의 4계층 구조는 데이터가 프로토콜 스택의 어플리케이션 계층에서부터 하위의 물리적 네트워크까지 전달되는 방식으로 처리되는 것을 볼 수 있다. 스택의 각 계층은 전달이 제대로 되었는지 확인하기 위해, 제어 정보를 추가한다. 이 제어 정보는 전송되는 데이터의 앞에 위치하기 때문에 헤더라고 부른다. 모든 계층에서의 배달 정보 추가를 가리켜 캡슐화라고 한다.

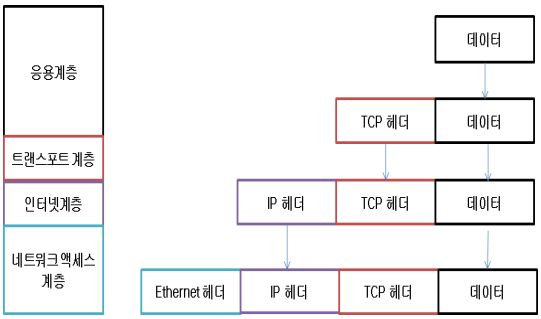

그림 2. TCP/IP구조 전송형태 - 송신측

데이터를 받았을 때 반대 작업이 이루어진다. 각 계층은 상위 계층으로 데이터를 전송하기 전에 그 헤더를 벗겨낸다. 정보가 스택의 위로 올라갈 때는 하위 계층에서 받은 정보를 헤더와 데이터로 해석한다.

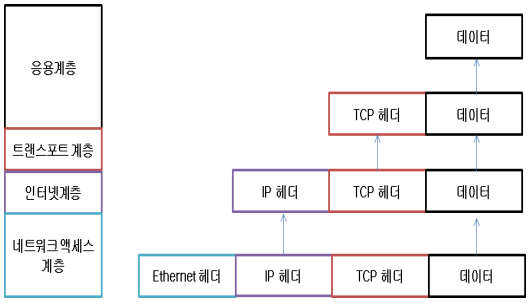

그림 3. TCP/IP구조 전송형태 - 수신측

III. 패킷 분석기 설계

제안하는 패킷 분석기의 주요 기능은 다음과 같다. 우선, 물리계층을 통해 사용자 시스템과 통신하는 여러 시스템들의 정보를 알 수 있다. 링크 계층에 해당하는 발신지 MAC주소, 수신지 MAC 등의 정보와 네트워크계층에서 담당하는 발신지와 수신지의 IP주소를 포함한 IP헤더 정보를 담고 있다. 또한 전송계층에서는 송/수신지 포트번호를 포함한 TCP 또는 UDP헤더 정보를 다루며, 마지막으로 응용계층에서 규정된 프로토콜에 따라 통신을 수행한다. 제안한 패킷 분석기는 C# 기반 sharpcap 라이브러리, winpcap 라이브러리를 참조하여 구현을 진행하였다.

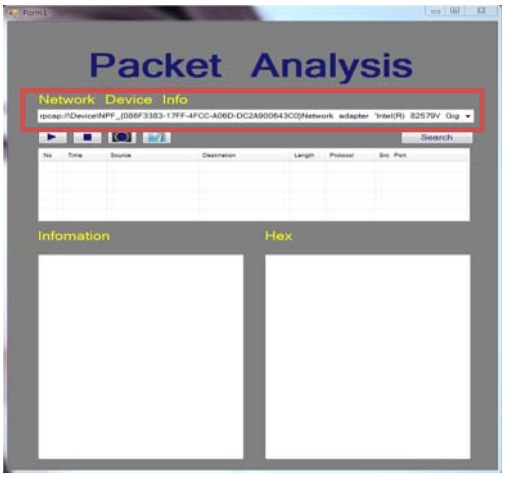

그림 4. Network Device 검색

그림 4에서는 검색 버튼을 누르면 나의 PC에 있는 디바이스 장치를 검색하는 과정이다.

그림 5는 네트워크 장치로부터 실시간으로 데이터를 캡쳐하는 과정을 나타낸 것으로 재생 버튼을 누르면 캡쳐가 시작된 시점부터 주어진 패킷의 번호와 시간, 송신지·수신지의 주소, 프로토콜 종류, 패킷 크기 등의 정보를 보여주며 중단 버튼을 누르면 윈도우는 네트워크 계층에 따라 분류된 패킷의 헤더 정보를 한 눈에 보여준다.

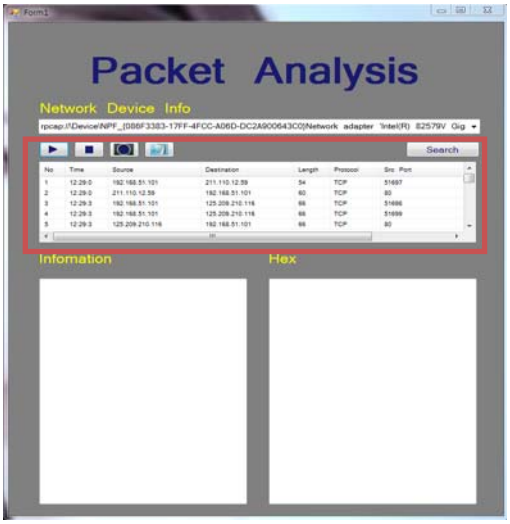

그림 5. 실시간 네트워크 송/수신 정보 출력

그림 6은 캡쳐된 패킷 정보를 분석하여 보여주는 것이다.

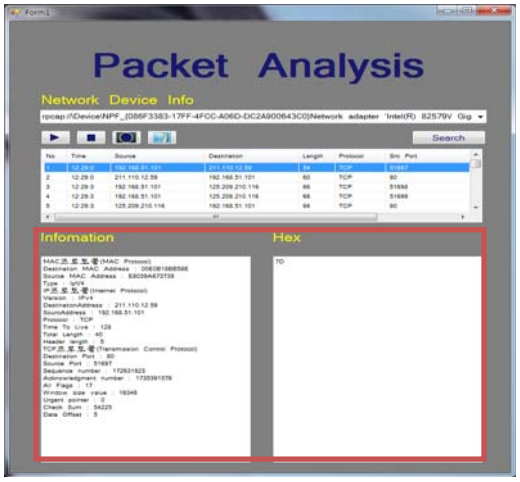

그림 6. 송/수신 정보 및 Hex 값

일반적으로 대중화된 패킷 캡쳐 프로그램들을 살펴보면 캡쳐한 패킷 정보들을 이미지 또는 텍스트 파일로 저장한다. 제안하는 패킷 분석기에서도 이러한 기본적인 기능을 포함하면서 처음 사용하는 학생, 초보자들을 위해 통신 흐름을 이해할 수 있는 이미지 기반 설명 기능을 구현 중에 있다.

Ⅳ. 결 론

본 논문에서는 TCP/IP 환경에서 송수신되는 데이터를 캡처하고 분석하는 패킷 분석기의 설계를 제안하였다. 이를 통해 네트워크 초보자들이 쉽게 활용하고 TCP/IP 각 계층의 기능과 전송 형태를 눈으로 직접 확인하며 이해할 수 있는 툴을 제시하였다.

향후에는 현재 개발한 프로그램을 보완하여 스마트폰에서 활용할 수 있는 안드로이드기반 패킷 캡쳐 앱도 개발하고자 한다.

참고문헌

[1] Laura Chappell, “(개정판) 와이어샤크 네트워크 완전 분석”, 2014.
[2] Richard Blum, "C# 네트워크 프로그래밍", 2004.
[3] http://sourceforge.net/projects/sharppcap/
[4] http://luyin.tistory.com/250