2. 목표 및 내용

2-2. 내용

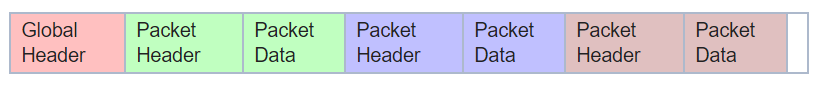
2-2-1. 사용자 환경

본 프로그램은 Cross Platform을 지원하고, 설치에 관한 번거로움을 줄이기 위해 Web 어플리케이션의 형태로 제공된다. 이 때, 분석대상이 되는 덤프파일 사이즈가 클 경우, 실제 서버로 업로드 되는 시간이 길어지기 때문에 불편함을 초래할 수 있다. 하지만 웹 환경에서 작동하지 않을 경우, 사용자의 프로그램 접근성이 떨어지게 된다. 이에 따라 제품을 패키지화하여 사용자가 자신의 PC에 패키지를 설치하고 localhost의 지정된 포트를 통해 설치된 패키지를 이용할 수 있게 함으로써 이러한 문제를 해결할 것이며, 이러한 방식은 사용자가 패키지를 자신의 개인PC나 서버에 선택적으로 설치할 수 있게 한다. 이는 사용자가 프로그램을 자신의 스타일에 따라 설치하여 사용할 수 있다는 것을 의미한다.

2-2-2. 전체 프로세스

본 프로그램의 전체적인 프로세스는 다음과 같다.

  
[그림 2-1] 프로그램의 전체 흐름도

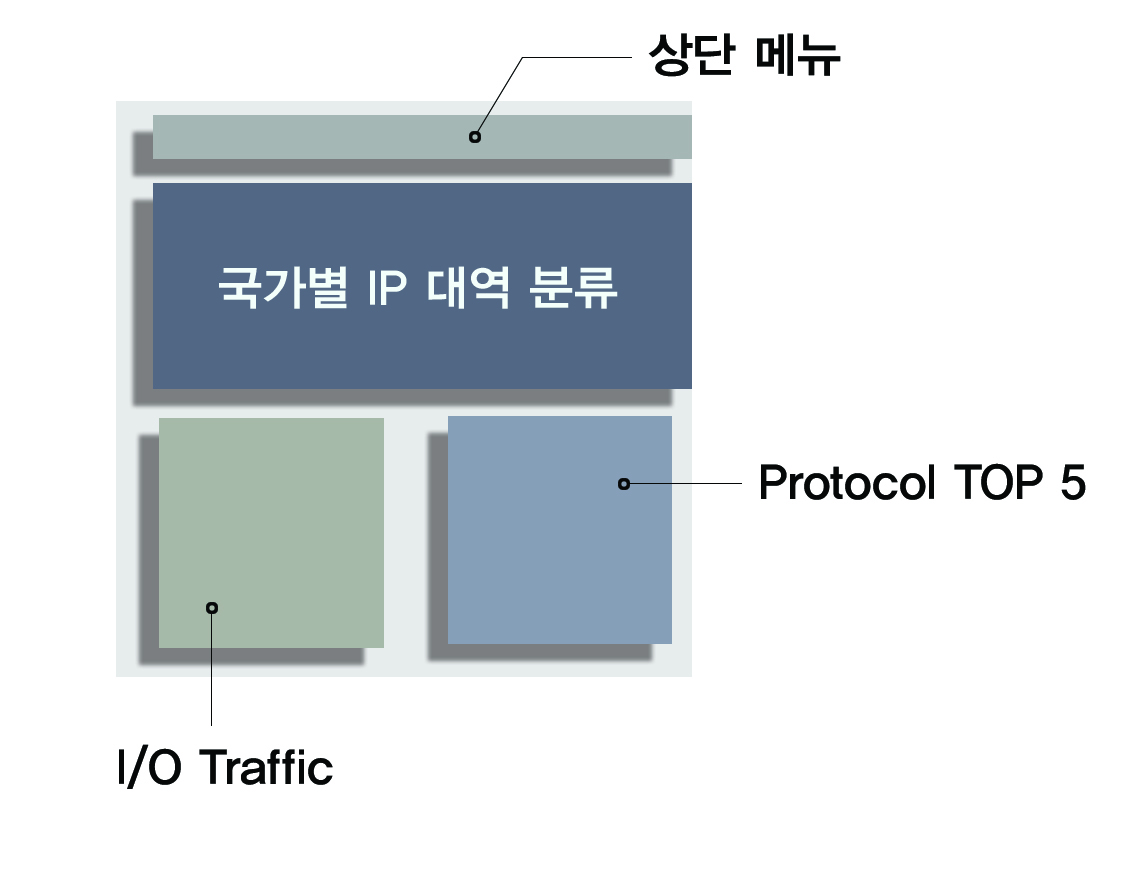
우선 사용자가 패킷 덤프파일을 업로드 하게 되면 대시보드의 그래프를 출력하기 위해 업로드 된 덤프파일에서 관련 정보를 추출해야 한다. 이를 위해 우선 덤프파일의 파싱 작업이 수행되는데 이는 업로드 된 파일이 분석 가능한 덤프파일인지 판별하고, 대표적인 덤프파일의 포맷인 pcap과 pcap-ng포맷을 구분한 후에 각 패킷을 분할하여 파싱 작업을 수행한다. 다음은 PCAP 파일의 구조를 그림으로 나타낸 것이다.  


[그림 2-2] PCAP File Structure

우선 Global Header 부분에는 PCAP파일인지 PCAP-NG파일인지를 나타내는 Magic Number가 포함 되어있다. PCAP파일의 Magic Number는 0xA1B2C3D4이고 PCAP-NG파일은 0x1A2B3C4D이다. 따라서 이것을 확인하면 정상적인 덤프파일인지 여부를 확인 할 수 있으며 추가로 PCAP과 PCAP-NG 포맷을 구분할 수 있다.  
바로 뒤부터는 패킷들의 헤더와 해당 패킷의 데이터가 존재한다. 이를 순차적으로 프로그램에서 Loading 하여 패킷들의 정보를 읽어와 분석 루틴을 수행할 수 있는데 이 때 분석루틴은 각 패킷이 모두 동일하지 않으므로 개별적으로 분석하여야 한다. 예를 들어 각 패킷별로 Protocol 정보가 다를 수 있으므로 각 Protocol별로 파싱 모듈을 제작하여 Protocol 구분 후에 분기하여 각 모듈별로 별도 분석을 하여야 한다.  
이러한 루틴으로 분석을 한 이후에는 대시보드 뿐만 아니라 그래프로 나타내야 할 정보들을 추출할 수 있다. 이렇게 추출된 정보를 분석결과라고 한다.  
분석결과를 도출하였다면 화면에 표시할 그래프별로 구조화 작업을 수행한다. 구조화 작업이란 Back-End 단에서 도출된 분석결과를 Front-End 단으로 전송하기 위함과 동시에 해당 분석결과를 사용자가 Save 및 Load 할 수 있도록 기능을 제공하기 위한 작업을 말한다. 분석결과를 구조화 할 때는 XML파일형태로 파일을 생성한다. XML파일로 분석결과를 구조화하면, Front-End와의 연동작업 시 간편하게 많은 정보를 체계적으로 전송할 수 있으며, Front-End 단에서 사용자의 분석파일을 Load하여 보여줄 때에도 통일된 모듈로 개발이 가능하다.

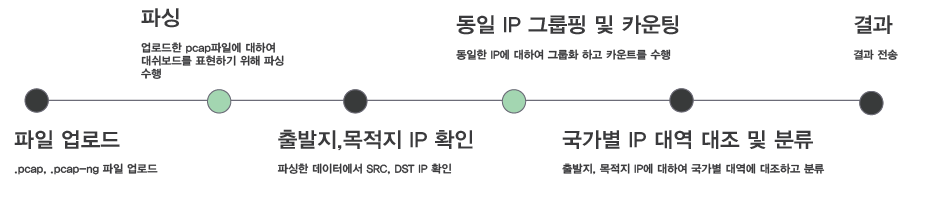
2-2-3. 대시보드 화면

분석 후 사용자에게 제일 처음으로 보여지는 화면이며, 다음 그림과 같이 구성되어 있다.



[그림 2-3] 대시보드 UI 구성

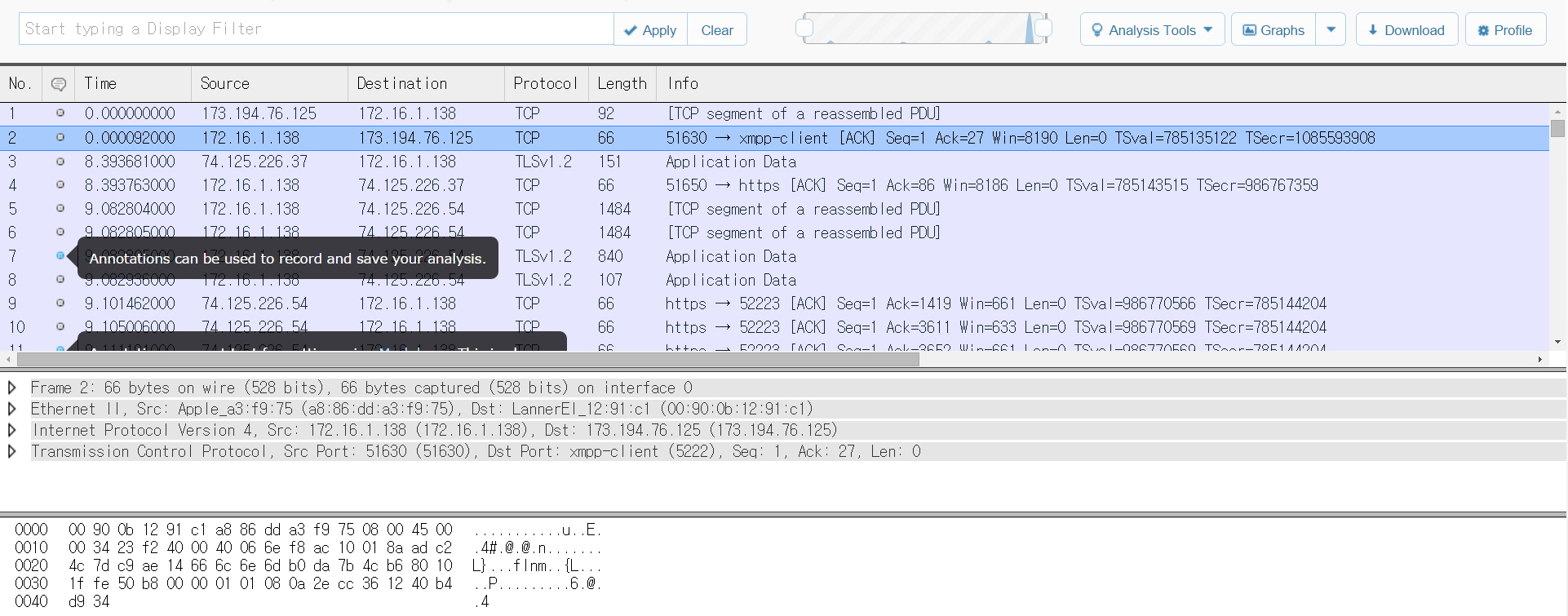
맨 상단에는 네비게이션 바가 위치하고 있다. 이곳에서 사용자가 원하는 형태의 통계정보를 선택적으로 제공받을 수 있다.  
다음으로 국가별 IP 대역 분류 정보가 제공된다. 이곳에서는 세계지도가 렌더링되며 트래픽이 집중된 국가일수록 진한 색으로 표기가 된다. 추가적으로 Inbound Traffic, Outbound Traffic을 구분하여 보여주어 In/Out Traffic의 흐름을 파악 할 수 있다. 국가별 IP 대역 분류는 다음과 같은 프로세스로 진행된다.

  
[그림 2-4] 국가별 IP 대역 분류 프로세스

국가별 IP 대역 분류의 하단에는 Input 및 Output 트래픽의 비율을 원형그래프의 형태로 보여준다. 사용자는 이를 이용해 덤프파일의 패킷흐름을 대략적으로 예측할 수 있다.  
다음으로 우측에는 덤프파일 내 패킷들의 프로토콜 형태를 분류하여 가장 많이 사용된 5가지의 프로토콜을 보여준다. 사용자는 최종적으로 대시보드 화면을 통해 In/Out Traffic과 사용 Protocol에 대한 흐름 파악이 가능하다.

2-2-4. Flow 화면

추가적으로 사용자에게 제공되는 화면은 Flow화면이다. Flow화면은 분석을 위해 패킷별로 보다 자세한 정보를 제공하기 위한 화면이다. 이는 기존의 제품들에서 찾아볼 수 있는 화면과 동일하며 예시화면은 다음과 같다.

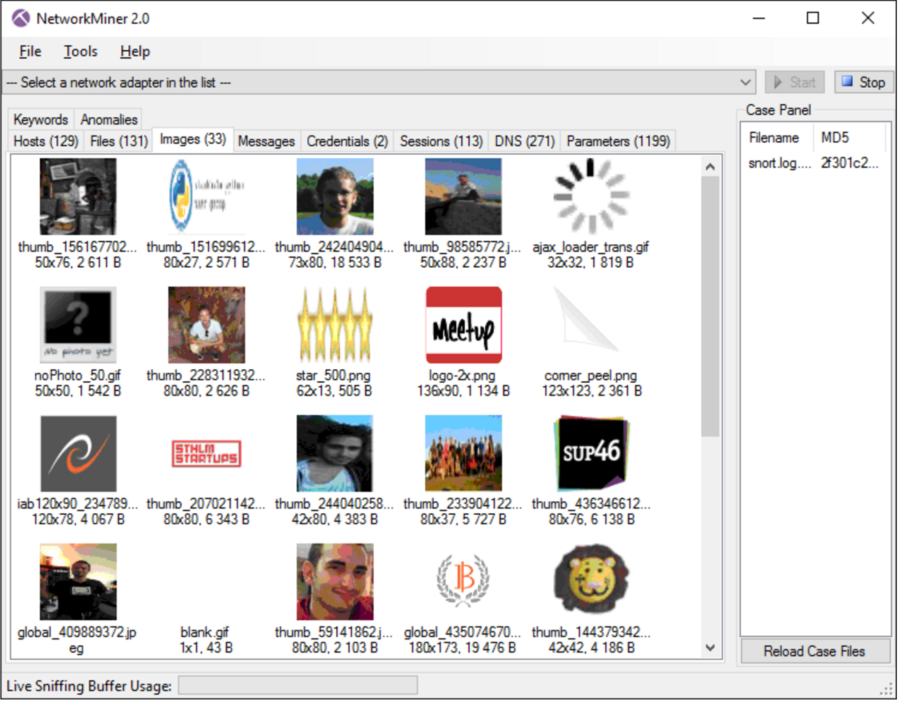


[그림 2-5] CloudShark의 패킷 캡쳐 화면

Web Application으로 서비스되는 CloudShark의 패킷 분석 화면이다. 패킷의 정보를 테이블형식으로 Time, Source IP, Destination IP, Protocol, Length, Info의 필드로 나누어 분류하여 제공한다. 본 제품도 마찬가지로 테이블형식으로 제공하며 제공되지만 기존 제품보다 더욱 자세한 정보가 제공된다. 예를 들어, Source IP, Destination IP, Domain Name, Source Port, Destination Port, TCP/UDP구분, Protocol 종류, Traffic 발생 국가, 패킷의 사이즈, 패킷의 길이 등이 제공될 예정이다.  
본 Flow화면을 구현할 때는 순차적으로 패킷을 분할하여 분석하여 화면에 나타낸다. 왜냐하면 덤프파일의 사이즈가 클 경우 분석시간이 오래 걸릴 수 있으므로 사용자 입장에서 기다리는 시간을 줄일 수 있게 된다.

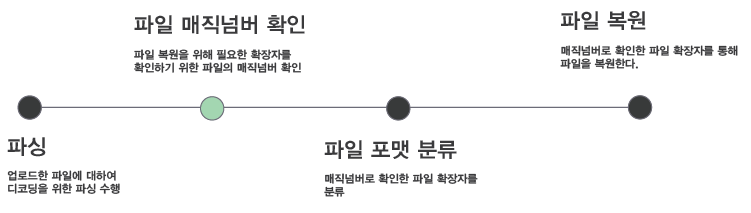
2-2-5. 파일복원

본 프로그램에서는 분석기능에 추가적으로 파일 복원기능을 제공한다. 다음은 파일 복원이 지원되는 패킷 분석 도구인 NetworkMiner로 이미지 복원을 수행한 화면이다.



[그림 2-6] NetworkMiner의 이미지 복원

NetworkMiner는 보이는 것처럼 이미지와 파일을 따로 탭으로 구분하여 준다. 복원된 파일이 이미지일 경우 썸네일 형식으로 사용자가 다운로드 해볼 필요 없이 미리 확인 해 볼 수 있다. 따라서 본 제품도 파일을 분류하여 사용자가 확인하기 편리하도록 하며, 추가적으로 이미지 썸네일 기능을 제공한다.  
분석기능에서의 프로세스는 다음과 같다.



[그림 2-7] 파일 복원 프로세스

우선 덤프파일을 2-2-2 절에서 설명한 것과 같이 파싱 작업을 수행한다. 패킷별로 파싱 작업을 수행한 이후에는 파일의 매직넘버를 확인한다. 파일은 확장자 별로 고유의 매직넘버를 가진다. 따라서 매직넘버를 확인하여 대조하면 파일의 확장자를 알 수 있으므로 분류가 가능하다. 이 때 분류된 파일 중 이미지파일 확장자만 따로 분류하여 썸네일 형식으로 사용자들에게 보여지게 된다.  
파일을 복원 할 때는 패킷에서 파일의 바이너리코드(Binary Code)를 Loading 하여 별도의 파일에 해당 바이너리 코드를 쓴다. 그 이후 매직넘버에 의해 판별된 확장자를 추가하여 저장 해주면 파일 복원이 완료된다.  
파일 복원을 수행할 때는 Flow화면을 구현할 때와 마찬가지로 Global Header 다음의 패킷부터 순차적으로 분석 루틴을 수행하고 파일을 복원하는 즉시 화면에 보여준다. 왜냐하면 Flow화면을 구현할 때와 동일한 이유로 덤프파일의 사이즈가 큰 경우에 분석 속도가 오래 걸리는 문제를 해결하기 위한 방법이다.