프로젝트 목표 및 내용

2.1 프로젝트 목표

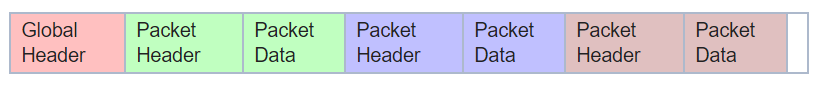
국내외에서 기업 내부의 기밀정보가 이메일·메신저 등 네트워크를 통해 유출되는 경우가 빈번하게 발생하고 있지만 네트워크를 통한 기술유출 대응방안 연구는 미흡한 실정이다. 따라서 본 프로젝트의 결과물인 파일 복원기능을 통해 기업의 보안 담당자 혹은 감사원이 유출된 파일에 대한 리스크 분석을 용이하게 하고, 국가별 IP traffic 통계 자료를 통해 기술유출 경로 파악을 용이하게 한다. 또한 I/O Traffic 통계 및 최다 사용 프로토콜 정보를 제공함으로써 기업 내 네트워크 품질 개선을 용이하게 하고자 한다.

2.2 프로젝트 내용

2.2.1 서비스 방식

본 프로그램은 Cross Platform을 지원하고, Web 어플리케이션의 형태로 제공된다. 이 때, 분석대상이 되는 덤프파일 사이즈가 클 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 사용자가 패키지를 자신의 개인PC나 서버에 선택적으로 설치할 수 있게 한다. 이는 사용자가 프로그램을 자신의 스타일에 따라 설치하여 사용할 수 있다는 것을 의미한다.  
최근 사용자들은 대용량의 덤프파일을 분석할 때 Packet Spliter를 이용하여 파일을 분할한 후 분할 단위별로 분석을 수행한다. 왜냐하면 대용량의 덤프파일에 많은 시간이 소요되는데 이에 따라 본 제품도 한번에 업로드할 수 있는 덤프파일의 용량을 100MB로 제한하도록 한다.

2.2.2 전체 프로세스

본 프로그램의 전체적인 프로세스를 설명하기 전에 덤프파일의 구조를 설명한다. 사용자로부터 업로드 될 덤프파일의 형식은 PCAP과 PCAP-NG이며 이 덤프파일은 Windows 환경의 Wireshark 프로그램이나 Linux의 TCP dump를 통하여 쉽게 얻을 수 있다. 다음은 PCAP 파일의 구조를 그림으로 나타낸 것이다.   


**[그림 2-1] 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.  
다음으로 Packet Header와 Packet Data를 확인 할 수 있다. 이 두 가지는 하나의 쌍으로 이루어져 있으며, 이 한 쌍을 Packet으로 부르기로 한다.

PCAP File의 Structure를 기반으로 전체프로세스를 살펴보면 다음과 같다.

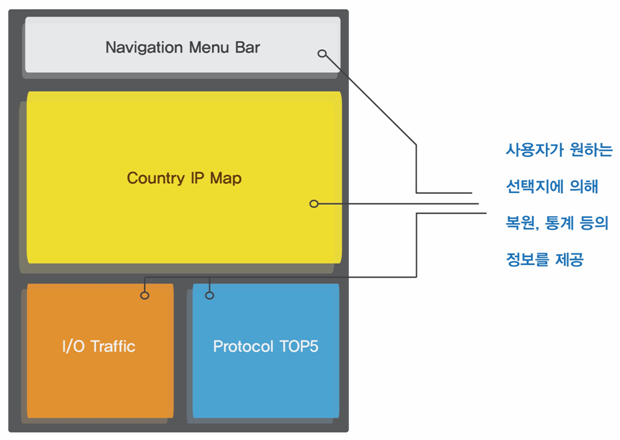


**[그림 2-2] 프로그램의 전체 흐름도**

우선, 사용자가 파일업로드를 수행하면, 파일의 Global Header내의 매직넘버를 확인하여 파일의 포맷을 구분한다. 정상적인 덤프파일로 인식되면, 덤프파일의 전체 Packet을 파싱한다. 전체 Packet을 파싱하는 목적은 개별 Packet을 파싱하여 분석하기 위해 덤프파일 내에 존재하는 Packet의 개수를 확인하기 위함이다.  
다음으로 파싱된 전체 Packet을 Packet별로 분할한다. 앞서 언급했듯이 Packet에는 Header와 Data가 존재하는데 이것을 분석하면 각 Packet별로 필요한 정보를 얻을 수 있으므로 분할 후 순차적으로 각 Packet을 파싱한다.  
Packet을 파싱한 이후에는 사용자 선택지에 따라 필요한 정보를 추출하게 된다. 결론적으로 파일업로드, 전체 Packet 파싱, Packet 분할, 각 Packet별 파싱은 공통된 작업이며, 이 후에 사용자 선택지에 따라 분기점이 나뉘어 분석 및 파일 복원 프로세스가 선택적으로 실행되게 된다.

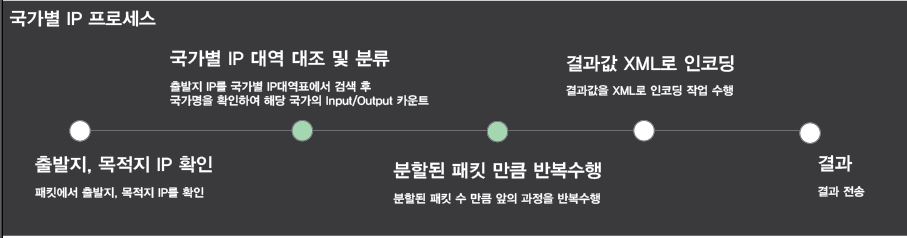
2.2.3 대시보드 화면

대시보드 화면에는 사용자들에게 총 3가지의 정보를 보여준다. 첫 번째는 국가별 트래픽량을 Input Traffic과 Output Traffic별로 나누어서 보여준다. 국가별 트래픽량은 Geo Map 형식으로 표현되며 세계지도 그림 위에 트래픽량이 많을수록 진한 색으로 표기하여 사용자들이 한눈에 파악할 수 있도록 한다. 두 번째로 전체 덤프파일의 Packet 중 Input Traffic과 Output 트래픽의 비율을 퍼센테이지(%)로 환산하여 원 그래프 형식으로 나타내주며, 마지막으로 가장 많이 사용된 5가지의 프로토콜을 추출하여 나타내준다. 다음 그림은 대시보드의 UI를 간략하게 나타낸 것이다.



**[그림 2-3] 대시보드 UI 구성**

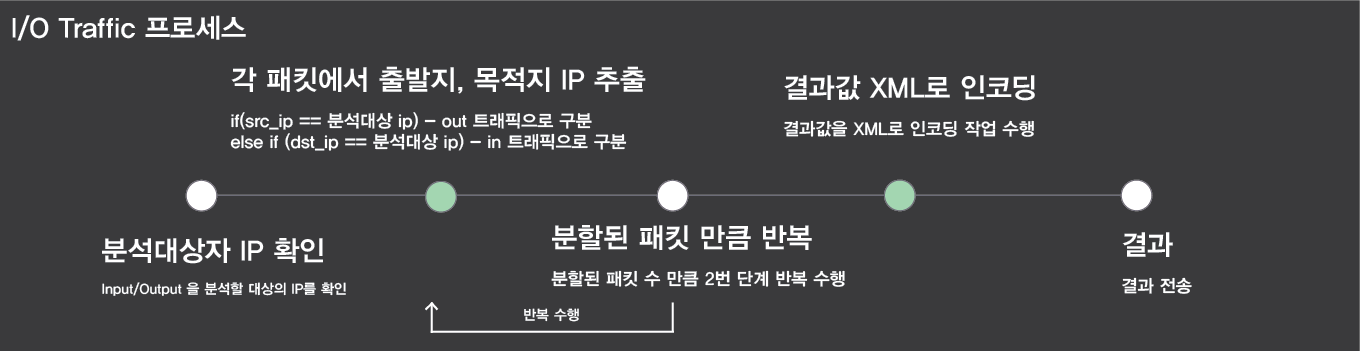
최상단에는 Navigation Menu Bar를 배치하고 메뉴를 탭 형식으로 제공하여 사용자가 기능을 간편하게 찾아서 덤프파일 내의 원하는 정보를 제공받을 수 있다. 다음으로 Country IP Map(국가별 트래픽 통계량), I/O Traffic Graph, Top 5 Protocol 정보를 위 그림과 같이 제공 받을 수 있다.  
다음은 국가별 트래픽 통계를 추출하는 과정을 세분화한 그림이다.



**[그림 2-4] 국가별 I/O 트래픽 통계 프로세스**

파싱 된 Packet에서 출발지와 목적지 IP를 확인하여 내부 데이터베이스에 미리 입력된 국가별 IP 대역표에서 검색 후 대조작업을 한 후 해당 IP의 국가명을 확인하여 분류한다. 이 과정은 앞서 분할된 패킷의 수만큼 반복적으로 수행하면서 분석된 정보를 저장한다. 모든 분석과정이 종료되면 결과값을 XML 포맷으로 분석정보를 구조화하여 인코딩 한다. 이렇게 구조화된 XML 데이터는 Front-End로 전송되어 디코딩 된 뒤, 사용자에게 분석결과로 출력되게 된다.

다음으로 전체 Packet의 I/O 트래픽량을 분석하기 위한 프로세스를 그림으로 나타낸 것이다..



**[그림 2-5] 전체 패킷의 I/O 트래픽 통계 프로세스**

우선 분석 대상자의 IP를 확인한다. 이 후 분할된 Packet에서 출발지, 목적지 IP를 추출하여 분석대상자의 IP와 대조한다. 이 작업은 Inbound Traffic과 Outbound Traffic을 구분하기 위함이다. Source IP가 분석대상자의 IP와 같다면 Outbound Traffic으로 구분하고, Destination IP와 분석대상자 IP가 같다면 Inbound Traffic으로 구분한다. 이러한 과정을 분할된 Packet의 수만큼 반복하여 전체 패킷에 대해 Inbound 및 Outbound Packet 수를 카운트하여 XML데이터로 인코딩 한 후 Front-End로 전송하게 된다.

다음은 Top5 프로토콜을 분석하기 위한 프로세스이다.

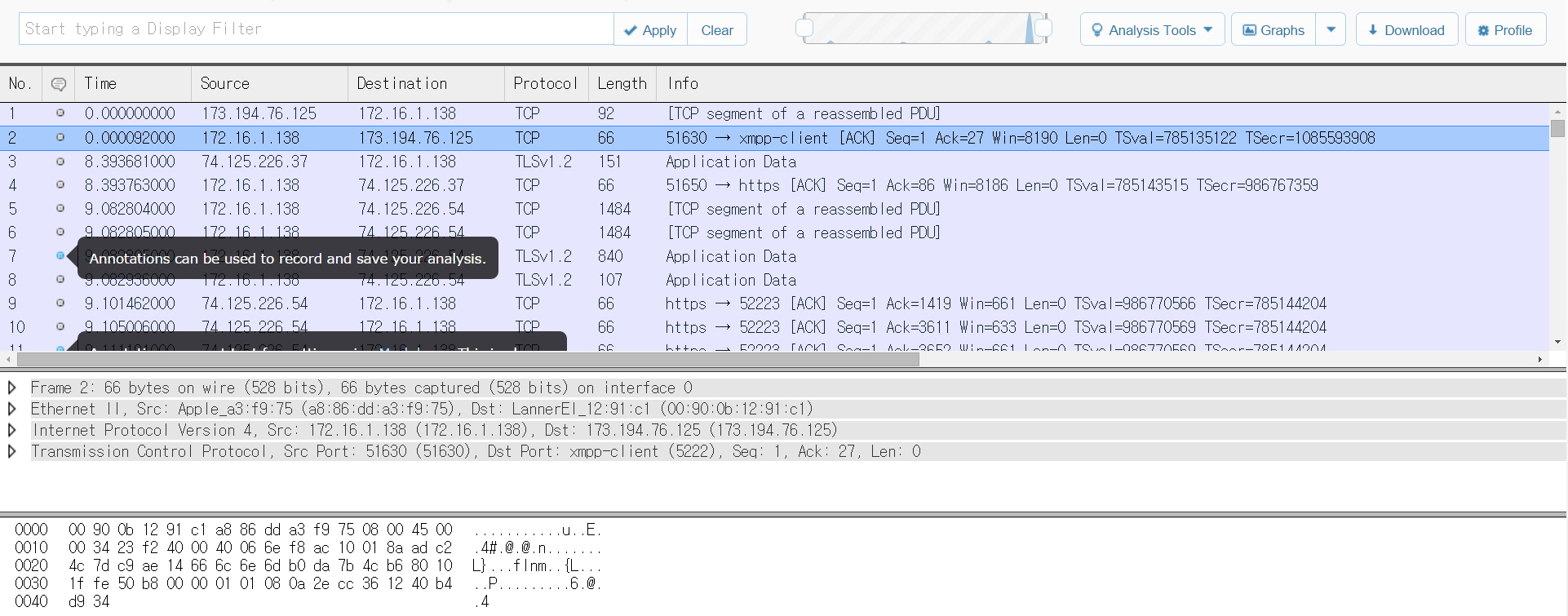


**[그림 2-6] Top5 Traffic 통계 프로세스**

맨 처음으로 분할된 Packet에서 프로토콜 정보를 확인한다. 다음으로 프로토콜 별 카운트 변수에 해당 프로토콜의 개수 정보를 +1 해준다. 이 과정을 분할된 Packet의 수만큼 반복하면 덤프파일 내의 존재하는 Protocol의 종류와 개수를 얻을 수 있는데 이 정보를 내림차순 정렬하여 상위 5개의 프로토콜명과 개수 정보를 XML로 인코딩 한 결과를 Front-End로 전송한다.

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 2-7] CloudShark의 패킷 캡쳐 화면**

Web Application으로 서비스되는 CloudShark의 Packet분석 화면이다. Packet의 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, Packet 사이즈, Packet길이 등이 제공될 예정이다.  
본 Flow화면을 구현할 때에는 분할된 Packet을 순차적으로 분석한 후 분석한 결과를 Front-End에 Ajax를 통해 개별로 전송함으로써 사용자가 분석결과를 기다리는 시간을 줄일 수 있다. 다음은 Flow화면의 구현 프로세스이다.

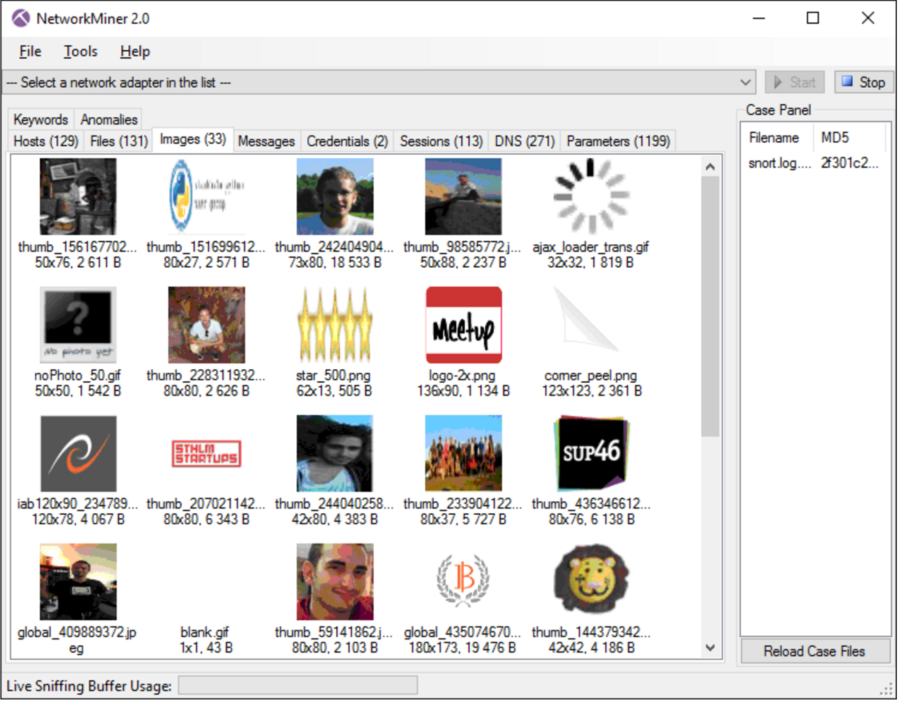


.**[그림 2-8] Flow화면 구현 프로세스**

본 Flow화면에서 중요한 점은 분석 -> 분석정보 구조화 -> 전송이 반복수행 된다는 점이다. 기존 통계정보와 다르게 유저에게 제공되는 정보가 자세하기 때문에 추출해야 할 정보가 많고, 그로 인한 시간지연이 발생할 수 있으므로 분석결과를 Front-End로 비동기전송 방식 중 하나인 Ajax 기술을 사용하여 전송하게 된다. Ajax를 사용하면 Front-End의 화면 전환이 발생하지 않고 지속적으로 업데이트된 정보를 추가할 수 있기 때문이다.  
분할된 Packet의 개수만큼 프로세스가 반복수행 되면서 분석을 완료하면 프로세스를 종료한다.

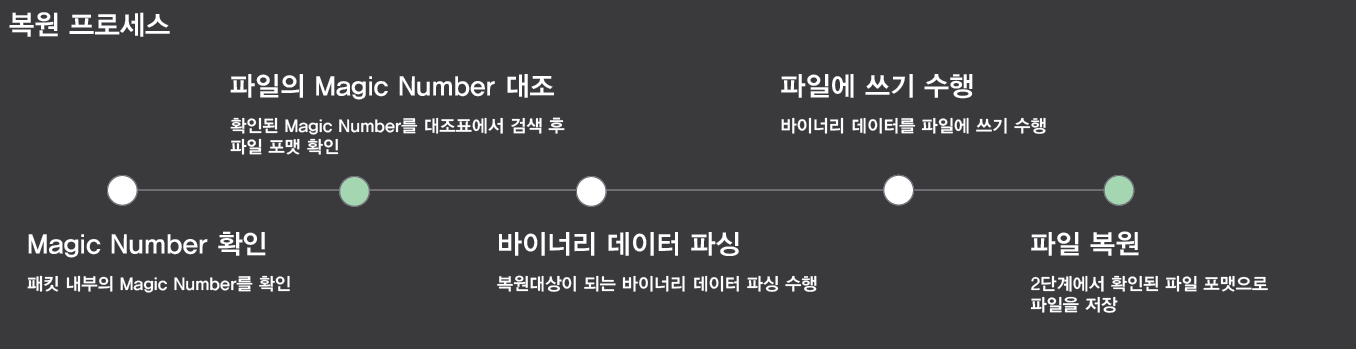
2.2.5 파일복원

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



**[그림 2-9] NetworkMiner의 이미지 복원화면**

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 이미지 썸네일 기능을 제공한다..



**[그림 2-10] 파일 복원 프로세스**

파일 복원을 하기 위해서는 우선 분할된 Packet에서 Magic Number를 확인하여 확인된 Magic Number를 내부 데이터베이스에 존재하는 파일의 Magic Number 대조표에서 대조하여 파일 포맷정보를 확인한다. 파일 포맷의 정보가 확인되었다면 파일의 바이너리 데이터를 읽어오게 된다. 이 후 파일입출력 함수를 이용해 파일에 해당 바이너리 데이터를 쓰고, Magic Number를 통해 확인된 파일 포맷으로 저장하면 파일 복원이 완료된다.