Journal of KIIT. Vol. 21, No. 11, pp. 171-181, Nov. 30, 2023. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2023.21.11.171

악의적인 네트워크 터널링 탐지를 위한 특징 비교 방법

김 민 수*

Feature Comparison Method for Detecting Malicious Network Tunneling

Minsoo Kim*

본 논문은 2022학년도 목포대학교 연구년교수 지원에 의하여 연구되었습니다.

요 약

네트워크 터널링은 중간에 도청이나 방화벽의 차단으로부터 안전하게 두 단말 네트워크 사이에서 안전한 통신을 하는 기술이다. 터널링 서버와 클라이언트는 터널링 방법에 따라 지정된 프로토콜을 사용하여 통신한다. 터널링 기법은 오픈소스로 개발되어 VPN처럼 안전하게 외부에서 내부로 접속하는 도구로 이용되지만, 악성코드와 같이 악의적인 공격 목적으로 이용되기도 한다. 이러한 악의적인 터널링에 대응하는 방법으로 주로 통계적 기법이나 기계학습이 연구되었으며, 사용되는 정보는 패킷의 내용이나 크기가 중심이었다. 그러나 암호화된 채널이거나 헤더의 크기를 일정하게 유지하면, 탐지 효과가 떨어질 것이다. 본 논문에서는 악의적인 목적의 DNS, HTTP, ICMP, 그리고 VoIP 터널링 기법의 특징을 분석한다. 그리고 기존의 탐지 방법을 보완할 수 있는 탐지 요소와 방법을 제안한다. 특히 터널링에서 이용가능한 네트워크 서비스의 헤더 특징과 악용 가능성을 살펴본다. 이러한 연구 결과를 통해서 악의적인 터널링을 탐지하거나 차단하는 모듈을 개발하는데 기여할 수 있을 것이다.

Abstract

Network tunneling is a technology for secure communication between two terminal networks safely from eavesdropping or firewall blocking in the middle. Tunneling server and client communicates using a designated protocol depending on the tunneling method. Tunneling technique is developed as an open source and is used as a tool to safely connect from the outside to the inside like a VPN. But it is also used for malicious attacks by malware. Statistical techniques or machine learning were studied as methods to counter such malicious tunneling, mainly using packet content or size information. However, if the channel is encrypted or the header size is kept constant, the detection effect will be reduced. In this paper, we analyze the characteristics of DNS, HTTP, ICMP, and VoIP tunneling techniques for malicious purposes. We propose detection elements and methods that can complement existing detection methods. In particular, we examine the header characteristics and possible abuse of the network services available in tunneling. These research results will be able to contribute to the development of modules that detect or block malicious tunneling.

Keywords

network tunneling, covert channels, malicious tunneling, network monitoring, tunneling detection

* 목포대학교 정보보호학과 교수
- ORCID: https://orcid.org/0000-0001-7056-1952

· Received: Aug. 01, 2023, Revised: Sep. 19, 2023, Accepted: Sep. 22, 2023
· Corresponding Author: Minsoo Kim
Dept. of Information Security, Mokpo National University
Tel.: +82-61-450-2716, Email: phoenix@mokpo.ac.kr

Ⅰ. 서 론

오늘날 사이버 공격은 매우 다양한 방향으로 이루어지고 있으며, 그 중에서 터널링(Tunneling) 기법은 공격자를 숨기기 위한 방법으로 활용되고 있다. 공격자가 목표 시스템에서 정보를 탈취할 때, 공격 프로그램에서는 보안 프로그램과 정책을 우회하고 공격 행위를 숨기기 위해 터널링 기법을 사용하는 경우가 많다.

네트워크 터널링으로 자주 사용되는 방법은 VPN, DNS, HTTP, ICMP, UDP 등이 있다. 특히 DNS와 HTTP는 방화벽에서 차단하지 않는 경우가 많기 때문에, 이 서비스를 이용한 터널링이 악성코드에서 발견되고 있다. 터널링 기법은 오픈소스나 상용도구로도 많이 알려져 있을 만큼, 터널링의 악용 사례가 발견되고 있다. 국내외 연구에서는 DNS, HTTP, ICMP 등의 터널링 행위를 탐지하기 위한 연구가 진행되었다. 통계적 방법과 기계학습 방법이 주로 적용되었다[1]. 그러나 암호화된 채널을 사용하거나 헤더의 크기를 일정하게 한다면, 그 효과가 떨어질 것이다.

본 논문에서는 DNS, HTTP, ICMP, 그리고 VoIP 터널링의 동작 과정을 분석하였다. 이러한 분석 과정을 통해서 기존 연구에서 다루지 않았던 패킷의 필드나 동작 행위에 관심을 가졌다. 그래서 기존의 탐지 방법으로 해결하지 못하는 부분을 보완할 수 있는 방법을 제시하였다. 이러한 결과가 터널링을 통하여 데이터 유출, 공격자 명령 송출, 데이터 삽입 등의 행위를 줄일 수 있는데 기여할 수 있을 것이다. 그리고 이 결과를 활용하여 터널링을 이용한 악성코드나 잠재적인 위협에 대응하고자 한다.

Ⅱ. 터널링 기법

2.1 네트워크 터널링(Network tunneling)

네트워크 터널링은 하나의 네트워크에서 다른 네트워크로 가상의 파이프를 통해서 안전하게 데이터를 보내는 기술이다. 이 기술은 프라이빗 네트워크이거나 통신 내용을 숨기기 위해서 사용된다. 터널링에는 암호화 기법이 주로 적용되어, 보안 채널의 역할을 수행한다. 일반적으로 패킷을 특정 프로토콜로 캡슐화하여 전송한다. 네트워크 터널링 기법으로 가장 대표적인 것이 VPN이다[2].

터널링 서버와 클라이언트 통신은 터널링 방법에 따라 지정된 프로토콜을 사용하여 통신한다. VPN의 경우 IPsec이나 SSL/TLS의 암호화 통신을 통하여 터널링을 수행한다. 사용자가 터널링 클라이언트 프로그램으로 접속하면, 터널링 클라이언트는 터널링 서버로 터널링 정보를 캡슐화하여 전송한다. 터널링 서버는 응답 정보를 캡슐화하여 클라이언트에게 보낸다. 이러한 과정을 통해서 사용자는 원격지 서버와 안전하게 통신할 수 있다.

2.2 악의적인 터널링

터널링 기술은 방화벽을 우회, 개인 네트워크 주소 숨기기, 특별 주소로 데이터 보내기, 안전한 암호 채널 제공 등의 기능을 수행할 수 있다[1]. 따라서 이러한 터널링 기법을 악성코드에서 사용하면, 방화벽을 우회하여 안전하게 정보를 유출하고 프로그램을 설치할 수 있게 된다. 일반적으로 이러한 악의적인 터널링을 은닉채널(Covert channel)이라 부르며, 정상적인 통신 프로토콜에서 숨겨진 방법을 통하여 다른 의도의 정보를 주고받기 위한 것이다[3]. 은닉채널은 크게 저장 채널(Storage channels)과 타이밍 채널(Timing channels)로 구분할 수 있다. 저장 채널은 통신 프로토콜의 어느 공간에 정보를 실어 보내는 것을 말한다. 타이밍 채널은 송수신 간격, 특정 플래그의 설정 등을 사용하여 정보를 은닉하는 방법을 말한다.

해커들은 은닉채널을 생성하여 C&C 혹은 백도어(Backdoor)로 사용하고 있으며, 은닉채널을 악용한 악성코드 또한 증가하고 있는 추세이다. 예를 들어, Pisloader는 DNS 터널링을 통해서 C&C 서버와 통신을 수행한다[4]. 이러한 터널링 기법을 응용한 악성코드가 점점 증가하고 있다. [5]에서는 ICMP 트래픽을 사용하여 윈도우 시스템에 악성 프로그램을 다운로드 하는 pingback 악성코드를 설명하였다.

III. 터널링 위협과 대응 연구

여기에서는 DNS, HTTP, ICMP, 그리고 VoIP 터널링이 어떻게 악의적인 목적으로 이용될 수 있는지를 설명한다. 그리고 이러한 행위를 방지하기 위하여 어떠한 연구가 있었는지 설명한다. [1]에서는 기존의 여러 터널링 방지 연구 기법을 비교하여 설명하고 있어서, 이 부분을 참고하였다.

3.1 DNS 터널링

DNS 터널링은 DNS 프로토콜을 이용하여 데이터를 터널링하는 기술이다. DNS 패킷은 대부분의 방화벽에서 차단하지 않기 때문에 터널링으로 이용될 수 있다. DNS 터널링을 이용한 악성코드는 Pisloader[4], FrameworkPOS, C3Pro-Raccoon 등이 있다.

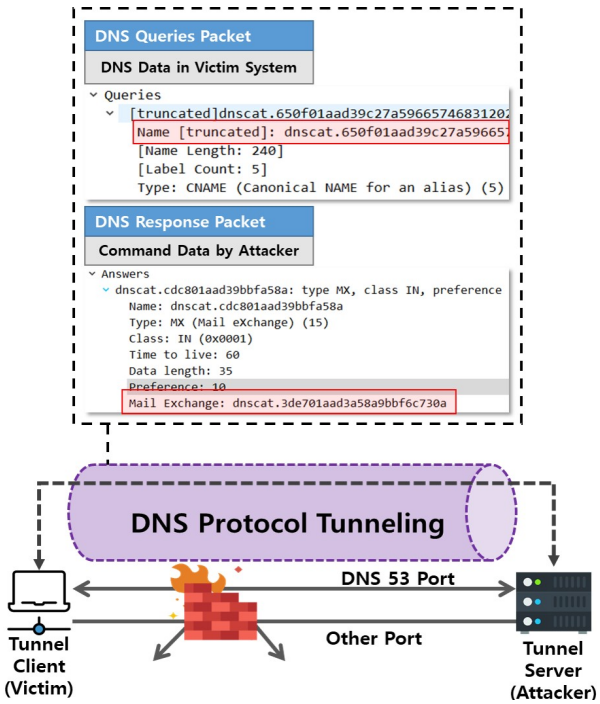

그림 1. DNS 터널링 예
Fig. 1. DNS tunneling example

DNS 패킷은 요청(Questies)과 응답(Answer) 레코드가 포함되어 있다. 일반적인 상황에서 IP 주소를 요청할 때는 유형(Type)이 A로 되어 있다. 이는 IP 주소를 반환받기 위함이다. 그림 1에서는 DNS 유형 중 MX(Mail Exchange) 레코드를 이용하여 터널링을 수행하는 과정을 보여준다. MX 레코드는 이 메일 메시지를 수신할 수 있도록 메일 서버의 주소를 알려줄 때 사용된다. DNS 응답에 명령을 실어서 보내면, 피해 시스템에서는 DNS 요청으로 명령 실행 결과를 보낸다. DNS 유형은 모두 DNS 터널링에 이용될 수 있다. 다만 유형 중에서 A와 MX 외에는 잘 사용되지 않는다.

[6]에서는 의심스런 도메인 정보를 블랙리스트에 추가하여 터널링을 막는 방법을 연구하였다. 이 연구에서는 Elasticsearch를 사용하여 DNS 호스트 이름을 계산하여 블랙리스트에 추가하였다. [7]에서는 DNS 터널링을 탐지하기 위해 CNN 알고리즘을 적용하여 DNS 패킷의 길이, 특정 문자 빈도, 엔트로피(Entropy) 등으로 악성코드를 구분하는 방법을 연구하였다. [8]에서는 DNS 캐시 서버에 기존 요청 정보가 저장되므로 그 캐시 적중률을 계산하였다. 캐시가 적중하지 못한 경우, AMS(Access Miss Count)로 기록하고, DNS 쿼리에 대한 AMS 횟수를 통계적 처리하고 임계값과 비교하여 탐지하였다. 이 연구에서는 DNS 터널링 탐지를 위한 모니터링과 필터링 시스템을 제안하였다.

[9]에서는 DNS 비밀 터널 동작을 식별하기 위해 2단계 방법을 사용하였다. 첫 번째는 대량의 트래픽 분석을 통해 일반 트래픽을 구분하였고, 두 번째는 5가지 DNS 터널링을 기계학습(Machine learning) 방법으로 분류하였다. 주로 DNS 요청 도메인 정보, 패킷 크기, 요청 횟수를 비교하였다.

3.2 HTTP 터널링

HTTP 터널링은 HTTP 프로토콜을 통해 데이터를 터널링하는 기술이다. 데이터를 HTTP 요청과 응답으로 캡슐화 하여 전송한다. 웹 서비스를 하는 곳에서는 방화벽에서 HTTP 패킷을 차단하지 않는 경우가 많아서 터널링으로 많이 이용된다. HTTP 터널링 도구인 Httptunnel이나 ProxyTunnel에서는 공격자는 프락시 또는 피해 시스템에 터널링 서버 프로그램을 구동시킨다. 이 프로그램은 그림 2처럼 웹서버로 동작되어 HTTP 터널링을 수행한다. 공격자는 TCP 통신 데이터를 HTTP 패킷으로 캡슐화하여 전송한다. 전송 데이터가 클 경우에는 여러 패킷으로 나누어 전송한다.

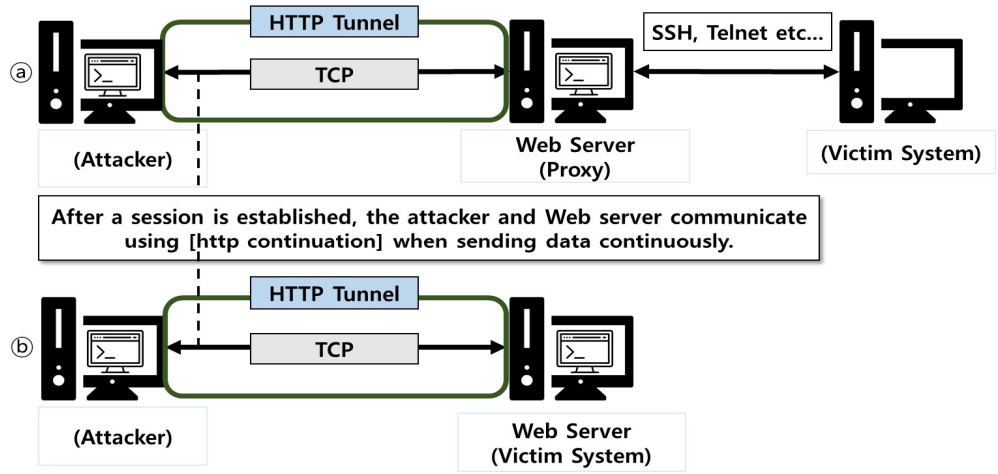

그림 2. HTTP 터널링 연결 예
Fig. 2. HTTP tunneling connection example

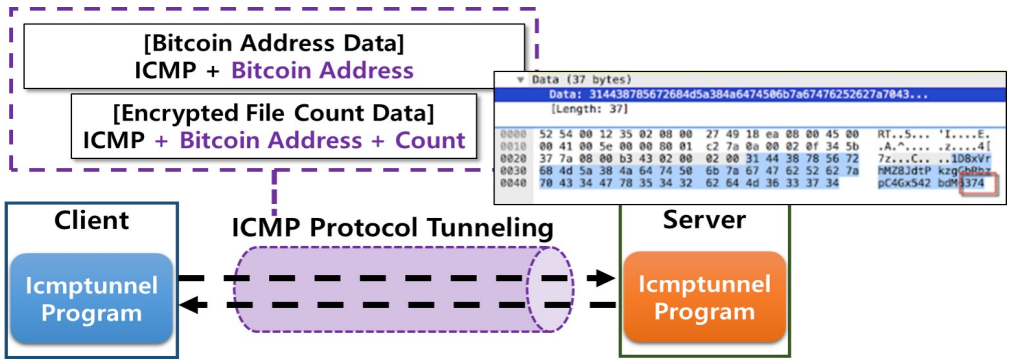

그림 3. NanoLocker 랜섬웨어가 사용한 ICMP 터널링
Fig. 3. ICMP tunneling by NanoLocker ransomware

HTTP 터널링을 이용한 악성코드는 ProxyBack, HTTP-RAT 등이 있다. ProxyBack 악성코드는 HTTP 터널링을 통해서 GET 방식으로 명령어를 전달한다[10]. 이 경우 명령어 패킷은 크기가 작지만, 응답 패킷은 일정 크기 이상의 패킷이 발생할 수 있다. 또한, 웹서버를 운용하지 않아도 HTTP 터널링 서버 프로그램이 동작 중일 수 있다.

[11]에서는 HTTP 터널링을 이용하는 악성 트로이목마를 탐지하기 위해 트래픽 시퀀스 분석을 이용하였다. 특정 트래픽의 반복 횟수를 Bi-LSTM 기계학습을 사용하여 비교하였다. [12]에서는 HTTPS 터널링을 탐지하기 위해 페이로드(Payload)를 비교하였으며, 페이로드 크기와 패킷의 반복 횟수에 대하여 SVM과 기계학습 방법을 사용하였다. [13]에서도 HTTPS 페이로드, 크기, 반복횟수를 대상으로 MLP와 LSTM 알고리즘을 사용하였다.

3.3 ICMP 터널링

ICMP 터널링은 ICMP 패킷을 이용하여 데이터를 터널링하는 기술이다. ICMP는 네트워크 연결 상태를 파악하기 위해서 사용되는 프로토콜이기 때문에, 방화벽에서 차단하지 않는다. 따라서 ICMP 패킷을 이용한다면, 방화벽을 쉽게 우회할 수 있다.

ICMP 터널링을 사용하는 악성코드로는 NanoLocker, Regin, Tspy_Small.CBE 등이 있다. 그림 3은 NanoLocker 랜섬웨어가 ICMP 터널링을 사용하여 정보를 전달하는 것을 보여준다[14]. 이 랜섬웨어는 ICMP Echo Request 패킷을 생성하여 패킷의 데이터 영역에 비트코인 주소 등의 정보를 삽입한다. ICMP 터널링 도구 중에서 NativePayload 처럼 TTL 값을 변경하여, TTL 값에 데이터를 보내는 것도 있다.

[3]에서는 ICMP 정상 트래픽의 패턴을 분석하고 HTTP 터널링이나 SSH 터널링에서의 ICMP 패킷 비중을 비교하는 방법을 제시하였다. [15]에서는 정상적인 ICMP 요청과 응답에는 페이로드의 크기가 일정하다는 특징으로 터널링을 감지할 수 있다고 하였다. [16]에서는 ICMP 체크섬, ID(Identifier), SEQN(Sequence Number)와 페이로드 부분을 비교하였다. Random Forest 알고리즘을 사용하여 위 데이터에 대한 학습을 수행하였다.

3.4 VoIP 터널링

VoIP(Voice over Internet Protocol) 터널링은 VoIP 데이터를 터널을 통해 안전하게 전송하는 기술이다. VoIP 통신은 UDP 위에서 SIP(Session Initiation Protocol)를 통해 통화를 연결하고 RTP(Real-time Transport Protocol)로 음성통화를 수행한다. SIP는 멀티미디어 통신에 대한 방법을 결정한다. RTP는 멀티디미어 데이터를 주고받는 것이다.

VoIP 패킷은 일반적으로 방화벽에서 차단되지만, VoIP 터널링을 사용하는 기업망에서는 방화벽을 우회하여 VoIP 통신이 가능하다. VoIP 통신에서 암호화 기법을 사용하여 비밀통신을 사용할 수도 있다. 일반적인 VoIP(RTP) 패킷은 SIP 통신(연결) 이후 RTP 데이터를 주고받는 형태를 가진다. 그래서 사용자 사이에는 SIP를 통한 세션 연결이 있어야 한다. 현재 VoIP의 SIP나 RTP 터널링을 이용한 악성코드는 발견되지 않고 있는데, VoIP는 일반적으로 방화벽에서 차단하기 때문이다.

[17]에서는 VoIP가 해커의 MITM 공격에 이용될 수 있음을 보여주며, VoIP가 보안에 취약할 수 있음을 경고하고 있다. 그러나 VoIP의 RTP나 SIP에 대한 터널링 탐지 방법은 연구되고 있지 않다. VoIP가 방화벽을 우회하기 어렵기 때문에 악성코드에서 이용되지 않고 있는 것으로 판단된다.

IV. 악의적인 터널링 대응 방법 제안

터널링을 통하여 불법적인 정보 전송에는 충분한 데이터 공간이 필요하다. 악성코드는 데이터 채널을 통하여 정보를 유출하거나 특정 코드를 다운로드 한다. 여기에서는 DNS 터널링, HTTP 터널링, UDP 터널링, ICMP 터널링, VoIP 터널링 등에서 데이터를 주고 받는 과정을 분석한다. 그리고 기존의 연구를 보완할 수 있는 서비스 영역과 방법을 제시하고자 한다.

4.1 DNS 터널링 특징 및 대응 방법

그림 4와 같이 DNS 패킷 구조를 보면, 가변적인 필드가 있다. 요청과 응답 레코드에 Name 필드와 응답 레코드의 RData(Resource Data) 부분이다. Name 필드는 널 문자로 끝나는 데이터가 들어간다. 이러한 공간에는 상당히 많은 데이터를 실어 보낼 수 있게 된다. DNS 요청 패킷은 헤더와 Query 부분만 포함되고, DNS 응답 패킷은 나머지 부분도 포함된다.

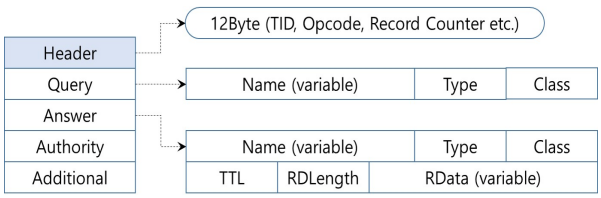

그림 4. DNS 패킷 구조
Fig. 4. DNS packet structure

그런데, 터널링이 악용되는 경우에는 다음과 같은 특징이 나타난다. 터널링은 공격자와 피해자 시스템 사이에 직접 터널링이 되거나, 공격자가 도메인에 붙어서 연결하는 형태로 이루어질 수 있다. 터널링에서 DNS 요청의 Name 필드와 DNS 응답 패킷의 RData 필드는 암호화된 16진수 값이 들어가게 된다. 이러한 경우 데이터 길이가 일반적인 패킷보다 크게 나타난다. 만약 Base128이나 Base64로 암호화되어 있다면, 읽을 수 있는 ASCII 문자도 나타날 수 있다. 그림 5는 DNS 터널링 도구를 통해서 위와 같은 동작을 확인한 것이다. 이 그림에서는 DNS 패킷의 유형을 TXT나 Null로 바꾸어 터널링을 시도한 것도 확인할 수 있다.

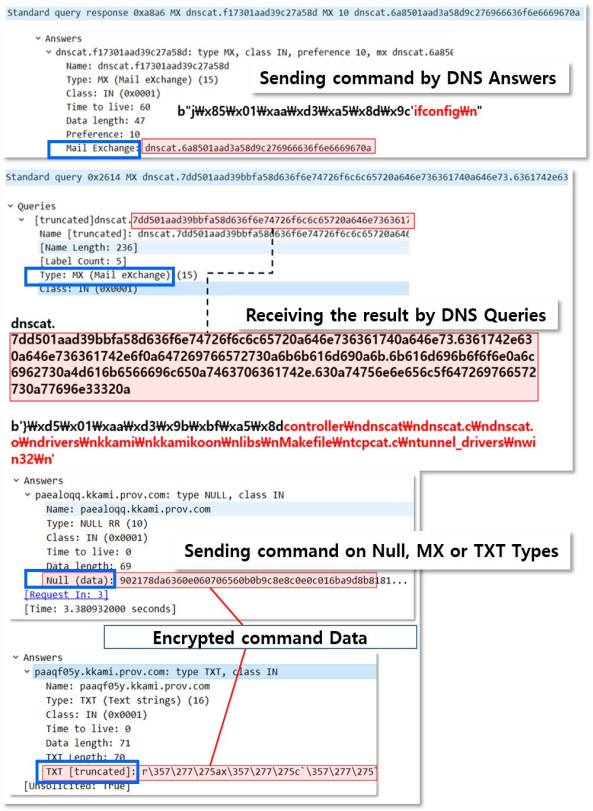

그림 5. DNS 터널링 특징
Fig. 5. DNS tunneling features

현재까지의 연구에서 터널링 탐지에 사용한 DNS 터널링 필드와 방법을 표 1에서 비교하였다. 기존 연구에서는 주로 DNS 패킷의 길이, 데이터(Name과 RData), 도메인의 적절성 등에 대하여 비교하였다. 도메인의 적절성은 화이트리스트나 DNS 캐시를 통해서 구현되었다. 그러나 DNS 특성상 알려지지 않는 도메인으로의 연결이 많아서, 이 방법의 효과는 크지 않을 것이다. DNS 패킷 길이와 데이터는 통계적 방법이나 기계학습 방법이 사용되었다. 그런데 DNS 패킷 길이는 길이를 일정하게 보내는 경우 의미가 없어진다. 터널링에 Name 부분이 이용된다면, 이 부분은 암호화된 형태로 기술되므로, IP 주소나 도메인 이름의 구조와 맞지 않게 된다. 학습 과정이 요구되는 기계학습 알고리즘을 적용하는 것보다, 구조적 특징 비교가 더 적당하다.

본 논문에서 DNS 터널링 탐지에 추가적으로 제시하는 방법은 다음과 같다. 첫째, 도메인 적절성 검사에 화이트리스트나 DNS 캐시는 자주 사용되지 않는 도메인에 대하여 오류가 발생할 수 있다. 따라서, Name 필드가 도메인이나 IP 주소에 대한 정규 표현식으로 표현되지 않음을 탐지할 수 있다. 둘째, Type 필드는 DNS 패킷의 유형을 나타내므로, 해당 도메인의 특성에 따라 A나 NS와 같은 특정 타입만 허용하는 탐지 규칙을 세울 수 있다. 셋째, TTL 필드는 4바이트 정수이며, DNS 캐시에 정보를 유지하는 시간을 나타내는데, 이 필드에 상위 2바이트를 작은 데이터를 보내는 채널로 사용할 수 있다. TTL 필드가 데이터 채널로 사용되지 않는다면, 일반적으로 같은 도메인에서는 TTL 값이 일정하게 유지될 것이다. 따라서 TTL 값의 변화를 탐지할 수 있다.

4.2 HTTP 터널링 특징 및 대응 방법

HTTP 요청과 응답 패킷은 그림 6과 같은 구조로 전송된다. HTTP 패킷을 보면, Entity Body 부분은 가변적으로 많은 양의 데이터가 전송될 수 있다. HTTP 터널링에서는 HTTP 패킷의 구조를 가지고 있으면서, Entity Body 부분을 통해서 다른 통신 채널을 구성하는 방법으로 동작된다.

표 1. DNS 터널링에 대한 탐지 특징과 방법 비교
Table 1. Comparison of detection features and method for DNS tunneling

Reference	Features in use	Comparison method
[6]	query name, answer name	- Compare with whitelist domain using Elasticsearch
[8]	DNS cache	- Threshold for hit rate of DNS cache
[7]	data length, query name, answer name, RData	- Apply CNN algorithm to packet length, entropy, and frequency of specific characters
[9]	header, RData, data length	- Classifying known DNS tunneling with machine learning
Suggested features	query name, answer name query type, answer type answer TTL	- Compare with regular expression for Name fields - Check the commonly used types such as A and NS - Check whether TTL maintains a constant value within the domain

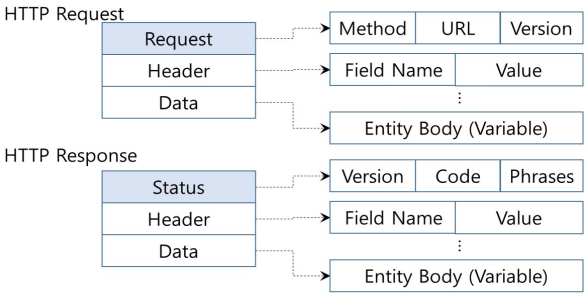

그림 6. HTTP 패킷 구조
Fig. 6. HTTP packet structure

기존의 연구에서는 HTTP 패킷의 반복 횟수를 기반으로 판단하고 있다(표 2). 터널링은 큰 데이터를 주고받는 경우가 많아서, 패킷의 반복 횟수를 기반으로 판단하는 것도 의미가 있다. 그러나 최근의 웹 사이트에서 동영상 등 대량의 데이터를 주고받는 경우가 많다. HTTP 패킷을 통해 한 번에 데이터를 전송할 수 있는 크기는 최대 1,514바이트로, 크기가 제한되어 있다. 데이터가 적으면 패킷 하나로 전달되지만, 데이터가 크다면 여러 패킷으로 전달되고 HTTP Continuation과 세션 ID로 연결을 한다. 그림 7에서는 Httptunnel 도구로 TCP 연결을 수행할 때의 패킷을 살펴볼 수 있다. HTTP 패킷이 1,514바이트를 넘어섰을 때, 패킷에서 "TCP segment of a reassembles PDU" 또는 "HTTP Continuation"를 볼 수 있다.

본 논문에서는 HTTP Continuation 패킷이 많이 발생할 경우, 추가적인 비교 방법을 제시한다. 그림 8의 경우 HTTP 연결 패킷의 특징이 나타난다. 여기에는 응답하는 패킷의 ACK 값이 전달받은 패킷의 SEQ와 LEN 값을 더한 값이 나타나는 특징도 있다. 이렇게 패킷이 크기가 커지는 경우는 이상 증상이라고 판단할 수 있다. 따라서 HTTP Continuation 패킷의 횟수와 SEQ와 ACK 값의 변화를 같이 비교할 수 있다.

표 2. HTTP 터널링에 대한 탐지 특징과 방법 비교
Table 2. Comparison of detection features and method for HTTP tunneling

Reference	Features in use	Comparison method
[12]	payload, packet length, packet count	- Use machine learning such as SVM, decision tree, and MLP
[18]	payload, packet count, sequence number	- Use MaMPF probability-based algorithm
[11]	packet count	- Use machine learning (Bi-LSTM)
[13]	payload, packet length, packet count	- Use MLP and LSTM algorithms
Suggested features	continuation packet, SEQ/ACK/LEN payload of response packets	- Compare the number of Continuation packets and flag - Determine normal response packets from web services using machine learning

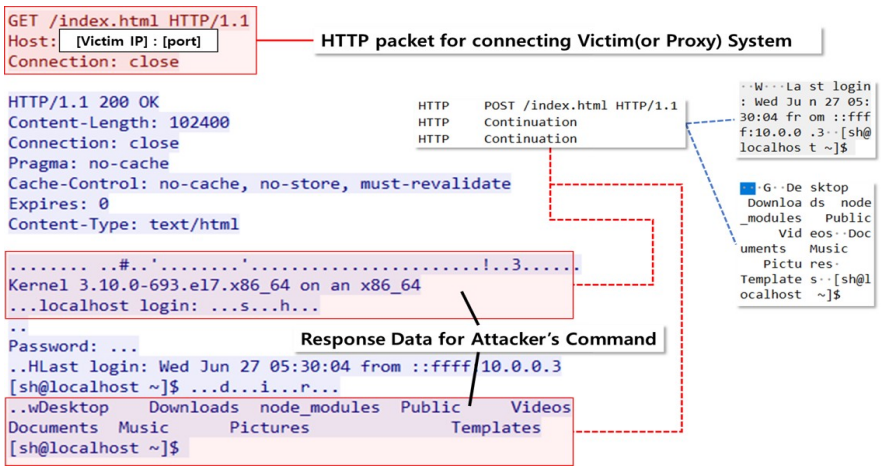

그림 7. HTTP 터널링 패킷
Fig. 7. HTTP tunneling packets

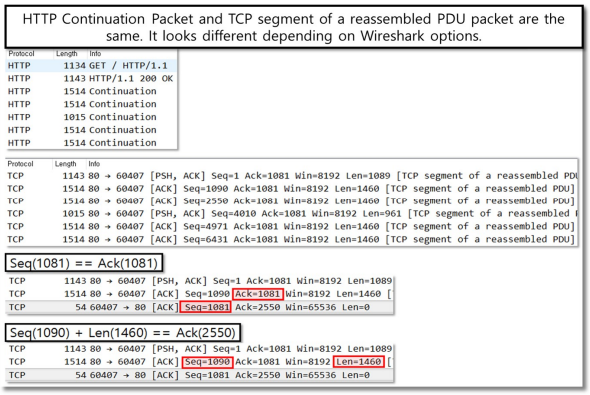

그림 8. HTTP 터널링 - TCP 패킷 특성
Fig. 8. HTTP tunneling - TCP packet fretures

HTTP 페이로드의 경우, 기존의 연구에서는 MLP 등의 기계학습 알고리즘을 적용하고 있다. 이 방법은 의미가 있으나, 암호화된 채널을 사용할 경우 구분하는 것이 쉽지 않다. HTTP 터널링은 HTTP 패킷 위에서 SSH 터널링이나 비밀 채널을 구성하는 형태로 많이 동작된다.

본 논문에서는 현재의 웹 서비스의 HTTP 패킷을 중심으로 학습하는 방법을 제시한다. 만약, 웹 서비스를 하지 않는 시스템이라면, 웹 응답 패킷이 발생할 때 방화벽에서 차단하면 된다. 웹 서비스를 하고 있다면, 응답 패킷이 정상적인 웹 서비스에서 제공되지 않는 유형인지 판단할 수 있다. 내부 사용자 측에서 발생하는 웹 요청 패킷의 경우는 기존의 방법처럼 기계학습으로 정상행위 모델링을 수행하면 된다.

4.3 ICMP 터널링 특징 및 대응 방법

ICMP 터널링은 그림 9의 ICMP 헤더 중에서 주로 페이로드 부분에 데이터를 실어 보내는 방식을 사용한다. ICMP 터널링은 ICMP 요청과 ICMP 응답 패킷의 페이로드를 수정하여 데이터를 보낸다.

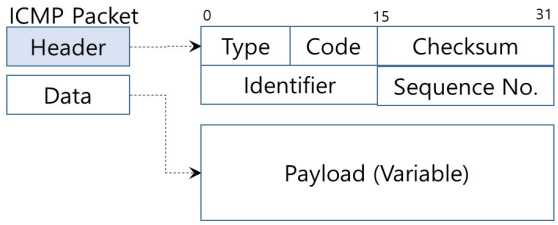

그림 9. ICMP 패킷 구조
Fig. 9. ICMP packet structure

페이로드를 사용하면, 최대 1,472바이트까지 보낼 수 있다. 원격에서 이곳에 명령을 실어 보내고, 그에 대한 응답을 받을 수 있다.

일반적으로 ICMP 데이터 공간은 윈도우즈 32바이트, 리눅스 56바이트로 일정하게 사용된다. 그리고 데이터의 내용도 일정하다. 따라서 ICMP 페이로드 크기가 변한다면, 쉽게 알 수 있다. 기존의 연구도 이러한 특성에 초점을 맞추고 있다(표 3).

그런데, 페이로드 크기를 유지하면서, 페이로드에 데이터를 조금씩 추가하면서 비밀 통신을 수행하는 경우에는 탐지하기 어려울 것이다. 페이로드 전체에 데이터를 실어보내는 것이 아니라 페이로드 뒷부분에 일부 데이터를 보내는 방식으로 탐지를 회피할 수 있다. 이러한 부분은 정상적인 페이로드 패턴을 학습하는 방법으로 해결할 수 있을 것이다.

본 논문에서는 정상적인 ICMP Echo 요청과 응답 패킷의 추가적인 특성을 분석하였다. 요청 패킷의 ID와 SEQN은 응답 패킷에서도 같은 값을 갖게 하여 구분한다.

표 3. ICMP 터널링에 대한 탐지 특징과 방법 비교
Table 3. Comparison of detection features and method for ICMP tunneling

Reference	Features in use	Comparison method
[3]	payload size, payload pattern, ICMP packet rate	- Compare the regularity of ICMP payload size and ICMP packet rate
[15]	payload size	- Detect by regularity of normal ICMP payload size
[16]	checksum, ID, SEQN, payload	- Determine whether packets are abnormal by multiple decision trees and Random Forest algorithm
Suggested features	ID and SEQN payload content	- Track changes in ID and SEQN - Learning changes in payload content in addition to payload size

그림 10에서 리눅스와 윈도우즈 시스템 모두 ICMP 패킷의 요청과 응답 쌍에서 ID와 SEQN이 같음을 알 수 있다. SEQN도 일정하게 증가한다. 그러나 ICMP 터널링 도구나 NanoLocker와 같은 악성코드에서는 ID와 SEQN이 불규칙하게 변하는 경우가 많다. 이러한 불규칙성을 탐지 규칙으로 추가하는 것도 가능하다.

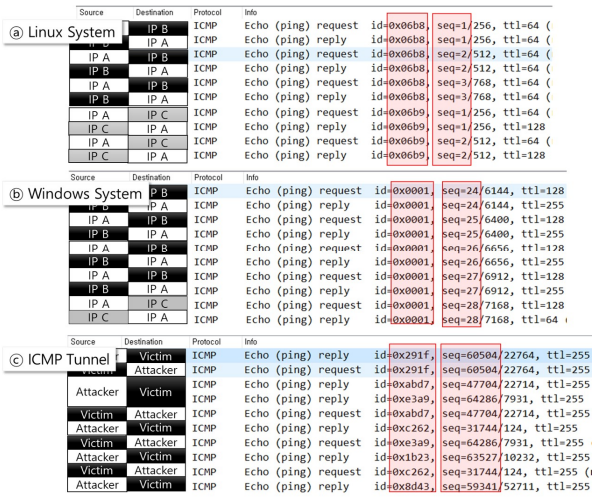

그림 10. ICMP 요청/응답 패킷의 특징
Fig. 10. Characteristics ICMP request/reply packets

4.4 VoIP 터널링 특징 및 대응 방법

VoIP에서 사용되는 RTP와 SIP가 터널링에 사용될 수 있다. RTP에 대한 터널링 도구는 알려지지 않았지만, 테스트해 본 결과 터널링이 가능함을 알 수 있었다. RTP와 SIP 통신 패킷의 구조는 그림 11과 같다. VoIP에서 터널링이 가능한 부분은 SIP Header, SIP Message의 Body, RTP Data의 Payload이다.

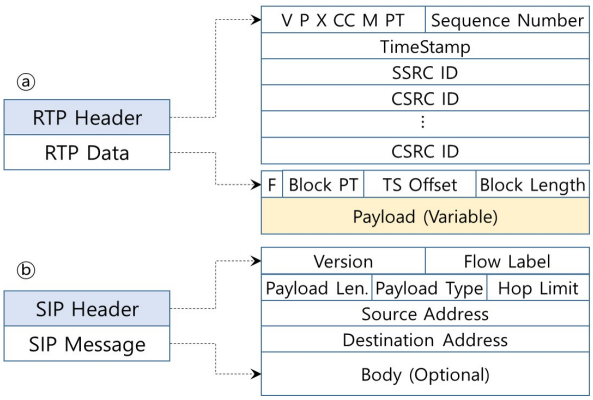

그림 11. VoIP 패킷 구조
Fig. 11. VoIP packet structure

VoIP의 SIP 터널링의 경우, 두 사용자는 서로의 정보를 교환하고 연결정보를 주고받게 된다. SIP Header에는 사용자의 정보와 세션 정보가 들어있게 된다. SIP Body(User part)는 사용자 이름이 들어가는 부분으로 최대 1,024바이트까지 사용할 수 있다. 여기에서 User Part 정보에 터널링 데이터를 넣어 보낼 수 있다. 이러한 경우 사용자 정보가 내부의 타당한 정보인지 검사하는 것으로 확인이 가능하다.

VoIP의 RTP의 Payload 영역에 터널링 데이터를 넣어 보낼 수 있다. 공격자는 이 부분에 암호화 채널을 사용하여 음성 데이터로 위장할 수 있다. 더구나 사용자가 데이터를 암호화한다면, 멀티미디어 데이터와 구분하기 어려울 수 있다. 그림 12는 음성 데이터 영역에 사용자 메시지를 삽입하여 암호화 전송을 시도한 결과이다. VoIP의 RTP Payload에는 음성 데이터가 들어가므로, 음성 데이터의 특징을 모델링하거나 학습한다면, 암호화된 채널과의 차이를 구할 수 있다.

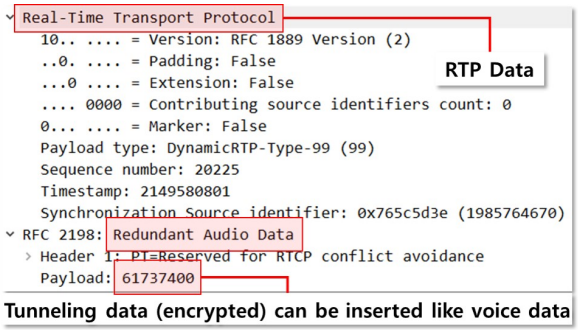

그림 12. RTP 터널을 통한 메시지 전송
Fig. 12. Sending messages throuth RTP tunnels

V. 결론 및 향후 과제

터널링 기법은 오픈소스 도구로 개발될 정도로 많이 알려져 있다. 터널링은 네트워크 단에서 통신 내용을 숨기거나 방화벽을 회피하는 목적으로 이용된다. 이러한 이유로 악성코드에서 공격자와의 통신을 숨기기 위해 자주 이용된다. 터널링은 네트워크에서 주로 사용되는 DNS, HTTP, ICMP 등에서 이루어지고 있다. 이러한 터널링에 대응하기 위한 연구도 진행되었으며, 비정상적인 패킷에 대하여 통계적 방법이나 기계학습 방법을 이용한 탐지가 주를 이루었다.

본 논문에서는 DNS, HTTP, ICMP, 그리고 VoIP 서비스의 터널링에 이용되는 필드와 이용 방법을 연구하였다. 이 과정을 통해서 터널링을 감지할 수 있는 패킷의 필드나 동작 행위를 분석하였다. 그리고 기존의 탐지 방법을 보완 할 수 있는 방안으로 DNS 패킷의 5가지 특징을 이용한 3가지 방법, HTTP 연속 패킷의 2가지 방법, ICMP 패킷 변화를 인식하는 2가지 방법을 제시하였고, VoIP의 터널링 데이터의 특징을 설명하였다. 이러한 연구 결과가 탐지 모듈로 개발하여 보호 네트워크 진입점에서 정보 유출을 방지하고 터널링을 차단할 수 있는 도구로 활용되기를 기대한다.

향후 연구에서는 제안한 방법에 대한 성능 평가를 수행할 계획이다. 현재까지의 연구는 그 가능성에 중심을 두었다. 공개된 도구를 이용한 정보유출 시나리오나 터널링을 사용하는 악성코드 대상으로 기존에 방식을 어느 정도 보완할 수 있는 지를 실험할 계획이다.

References

[1] Z. Sui, H. Shu, F. Kang, Y. Huang, and G. Huo, "A Comprehensive Review of Tunnel Detection on Multilayer Protocols: From Traditional to Machine Learning Approaches", Applied Sciences: Computing and Artificial Intelligence, Vol. 13, No. 3, pp. 1-30, Jan. 2023. https://doi.org/10.3390/app13031974.

[2] Amankatiyar, A. Vishwakarma, A. soni, Hemantjain, and J. Surana, "Research on Tunneling Techniques in Virtual Private Networks", International Journal of Engineering Development and Research, Vol. 5, No. 2, 2017.

[3] S. Sayadi, T. Abbes, and A. Bouhoula, "Detection of Covert Channels over ICMP Protocol", IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, Oct. 2017. https://doi.org/10.1109/AICCSA.2017.60.

[4] J. Grunzweig, M. Scott, and B. Lee, "New Wekby Attacks Use DNS Requests As Command and Control Mechanism", UNIT42 Malware, May 2016. https://unit42.paloaltonetworks.com/unit42-new-wekby-attacks-use-dns-requests-as-command-and-control-mechanism/ [accessed: Aug. 01, 2023]

[5] A. Sharma, "New Windows 'Pingback' malware uses ICMP for covert communication", Bleeping Computer, May 2021. https://www.bleepingcomputer.com/news/security/new-windows-pingback-malware-uses-icmp-for-covert-communication/ [accessed: Aug. 01, 2023]

[6] A. F. Sani and M. A. Setiawan, "DNS tunneling Detection Using Elasticsearch", IOP Conference Series: Materials Science and Engineering, Yogyakarta, Indonesia, Vol. 722, Oct. 2019. https://doi.org/10.1088/1757-899X/722/1/012064.

[7] D. Lambion, M. Josten, F. Olumofin, and M. D. Cock, "Malicious DNS Tunneling Detection in Real-Traffic DNS Data", IEEE International Conference on Big Data, Atlanta, GA, USA, Dec. 2020. https://doi.org/10.1109/BigData50022.2020.9378418.

[8] N. Ishikura, D. Kondo, V. Vassiliades, I. Iordanov, and H. Tode, "DNS Tunneling Detection by Cache-Property-Aware Features", IEEE Transactions on Network and Service Management, Vol. 18, No. 2, Jun. 2021. https://doi.org/10.1109/TNSM.2021.3078428.

[9] B. Wang, G. Xiong, P. Fu, G. Gou, Y. Qin, and Z. Li, "A Two-Stage Method for Fine-Grained DNS Covert Tunnel Behavior Detection", Science of Cyber Security, Vol. 13580, pp. 201-216, Sep. 2022. https://doi.org/10.1007/978-3-031-17551-0_13.

[10] J. White, "ProxyBack Malware Turns User Systems Into Proxies Without Consent", Unit42, Dec. 2015. https://unit42.paloaltonetworks.com/proxyback-malware-turns-user-systems-into-proxies-without-consent/ [accessed: Aug. 01, 2023]

[11] Y. He, Y. Zhu, and W. Lin, "HTTP Tunnel Trojan Detection Model Based on Deep Learning", Journal of Physics: Conference Series, Vol. 1187,

No. 4, 2019. https://doi.org/10.1088/1742-6596/1187/4/042055.

[12] B. Anderson and D. McGrew, "Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and Non-Stationarity", In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, United States, pp. 1723-1732, Aug. 2017. https://doi.org/10.1145/3097983.3098163.

[13] A. Parchekani, S. Nouri, V. Shah-Mansouri, and S. P. Shariatpanahi, "Classification of Traffic Using Neural Networks by Rejecting: A Novel Approach in Classifying VPN Traffic", arXiv Computer Science, Jan. 2020. https://doi.org/10.48550/arXiv.2001.03665.

[14] Adam (@cyberclues), "NanoLocker - Ransomware analysis", malware clipboard, Jan. 2016. http://blog.malwareclipboard.com/2016/01/nanolocker-ransomware-analysis.html [accessed: Aug. 01, 2023]

[15] A. Jeyashankar, "ICMP Attacks - Types & Codes For Log Analysis, Detection & Defense", Security Investigation, Aug. 2021. https://www.socinvestigation.com/icmp-attacks-types-codes-for-log-analysis-detection-defense/ [accessed: Aug. 01, 2023]

[16] D. X. Cho, D. T. H. Thuong, and N. K. Dung, "A Method of Detecting Storage Based Network Steganography Using Machine Learning", Procedia Computer Science, Vol. 154, pp. 543-548, Jan. 2019. https://doi.org/10.1016/j.procs.2019.06.086.

[17] J. N. Obidinnu and A. E. Ibor, "A survey of Attacks on VoIP networks and Countermeasures", West African Journal of Industrial & Academic Research, Vol. 14 No. 1, Jun. 2016.

[18] C. Liu, Z. Cao, G. Xiong, G. Gou, S.-M. Yiu, and L. He, "MaMPF: Encrypted Traffic Classification Based on Multi-Attribute Markov Probability Fingerprints", IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), Banff, AB, Canada, Jun. 2018. https://doi.org/10.1109/IWQoS.2018.8624124.

저자소개

김 민 수 (Minsoo Kim)

2000년 2월 : 전남대학교 전산통계학과(이학박사)
2000년 3월 ~ 2001년 2월 : 한국정보보호진흥원(KISA) 선임연구원
2001년 3월 ~ 2005년 2월 : 전남대학교 시스템보안연구센터 연구교수
2005년 3월 ~ 현재 : 목포대학교 정보보호학과 교수
관심분야 : 침입탐지와차단, 악성코드 분석, 리버싱, 공학설계교육