Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.25, NO.2, Apr. 2015 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2015.25.2.303

DNS 증폭 공격 탐지를 위한 근실시간 DNS 질의 응답 분석 시스템에 관한 연구*

이 기 택,† 백 승 수, 김 승 주‡
고려대학교 정보보호대학원

Study on the near-real time DNS query analyzing system for DNS amplification attacks*

Ki-Taek Lee,† Seung-soo Baek, Seung-joo Kim‡
Center for Information Security Technologies(CIST), Korea University

요 약

DNS 증폭 공격은 새로운 타입의 DDoS 공격의 일종이며 이러한 DNS 공격이 요즘 빈번히 발생하고 있는 실정이다. 기존 연구에서는 DNS 캐쉬의 트래픽의 증폭량을 탐지하여 다량의 패킷을 보내는 IP를 탐지하기도 하며, 질의 요청과 응답 패킷 사이즈의 크기 비율을 탐지하기도 하였다. 하지만, 이와 같은 DNS 패킷 트래픽 기반의 탐지 방법은 주소체계 변화에 따른 패킷 사이즈를 감당할 수 없으며, 분산된 공격에도 취약하다. 또한, 실시간 환경에서의 분석이 불가능하였다. 하지만, 본 논문에서는 실제 ISP의 DNS 데이터를 중심으로 근실시간 통합 분석이 가능한 DNS 질의 응답 분석 시스템을 구축하여 효율적인 DNS 증폭공격 탐지 방법을 제시한다.

ABSTRACT

DNS amplification is a new type of DDoS Attack and nowadays the attack occurs frequently. The previous studies showed the several detection ways such as the traffic analysis based on DNS queries and packet size. However, those methods have some limitations such as the uncertainty of packet size which depends on IP address type and vulnerabilities against distributed amplification attack. Therefore, we proposed a novel traffic analyzing algorithm using Success Rate and implemented the query analyzing system.

Keywords: DNS, amplification attack, Network, UDP, DNS query

I. 서 론

수 많은 네트워크 안의 컴퓨터들은 IP주소를 이용하여 서로를 구별하고 통신을 한다. 사람들이 네트워크를 통해 원격의 컴퓨터에 접속하기 위해서는 IP주소를 이용하여야 한다. 하지만, 연속적인 숫자로 되어 있는 IP주소를 일일이 외우는 일은 쉽지 않은 일이다. 그래서 사람이 인식할 수 있는 주소체계인 도메인 주소 체계(Domain Name System)가 만들어졌다. 우리는 이러한 주소 체계를 효율적으로 이용하기 위해 DNS를 이용한다. DNS(Domain Name Server)는 네트워크의 호스트를 식별하기 위해 계층적으로 이름을 부여하는 시스템이다. 이 시스템을 통해서 우리는 외우기 어려운 IP 주소 대신에 인터넷의 URL(Uniform Resource Locator)

접수일(2014년 12월 18일), 수정일(2014년 2월 10일), 게재확정일(2014년 2월 11일)
* 본 연구는 방위사업청과 국방과학연구소의 지원(계약번호 UD100002KD) 및 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구결과입니다.
† 주저자, zizihacker@korea.ac.kr
‡ 교신저자, skim71@korea.ac.kr(Corresponding author)

을 주소처럼 사용하여 편리한 생활을 하게 되었으며, 인터넷에서 하루 평균 수십조 이상의 DNS 질의가 이용되고 있는 현실이다[1].

DNS는 인터넷 서비스를 위하여 누구나 접속하도록 설계 되어 있으며, 해당 프로토콜은 요청 및 응답이라는 단순한 절차로 이루어져 있다. 또한 분산 환경에서 여러 DNS가 운영되므로 자유롭게 관리되기도 한다. 하지만, 이러한 공개적 접근과 분산 환경을 이용하여 최근 많은 해커들이 사이버 공격을 위해 DNS를 이용하고 있다. 특히, DNS 증폭공격(DNS amplification attack)을 통해서 특정 목표에 대한 공격이 두들어 지고 있다. DNS 증폭 공격은 작은 DNS 질의를 이용하여 대용량의 응답을 만들어 내는 일종의 서비스 거부(Denial of Service) 공격중 하나이다[2]. 2013년 3월 Spamhau.org 에 대규모 DNS 증폭 공격이 있었으며, 6월에는 대한민국 통합 전산 센터에 대한 공격도 있었다[3]. 최근 보고서에 의하면 이러한 공격 유형은 2014년 1월부터 8월까지에는 183%나 증가하였다[4].

그 동안 이러한 DNS 증폭 공격을 탐지하기 위한 많은 노력이 있어왔다. Cache DNS의 트래픽의 증폭량을 탐지하여 다량의 패킷을 보내는 IP를 탐지하기도 하며[5], 질의 요청과 응답 패킷 사이즈의 크기 비율을 탐지하기도 한다[6]. 이러한 방법 모두 네트워크 트래픽에 기반한 탐지 방법으로 다음의 제한 사항을 갖는다. 첫째, DNS 요청에 대한 패킷 사이즈는 IPv4, IPv6, DNSDEC 등 요청방식에 따라 다르므로 동일한 DNS 질의라 하더라도 패킷 사이즈가 달라지게 되어 오탐 확률이 높다. 둘째, 분산되어 있는 DNS 캐쉬의 증폭량을 각각 탐지한다면 DNS 트래픽 정책의 한계 이하 공격 시 탐지가 어렵다. 이러한 분산된 패킷이 결합해 하나의 네임서버(Name Server)나 별도의 공격목표를 집중 공격할 수도 있기 때문이다.

본 논문에서는 실제 ISP의 DNS 데이터를 중심으로 근실시간 통합 분석이 가능한 DNS 캐쉬 환경을 구축함으로써 효율적인 DNS 증폭공격 탐지 방법을 제시한다. 본 논문은 다음과 같이 구성된다. 2장에서는 DNS 공격을 탐지하는 관련된 연구들과 제약사항을, 3장에서는 효과적인 DNS 증폭공격 탐지를 위한 알고리즘을 제안하며, 4장에서는 제안된 알고리즘을 이용한 DNS 분석기 구축 및 모니터링 방법 구체화 방안을 제시한다. 5장에서는 실험 결과 및 평가를 제시하며, 끝으로 6장에서는 결론과 향후 연구방향을 제시한다.

II. 관련 연구 및 제약사항

2.1 DNS 서버 운영 원리

DNS를 사용하기 위해서는 Fig.1.과 같이 사용자는 특정 서비스를 위한 URL을 웹 브라우저에 입력한다. 입력된 URL은 사용자의 PC 또는 장치에 있는 hosts 파일 내의 정보를 먼저 확인하며, 없을 경

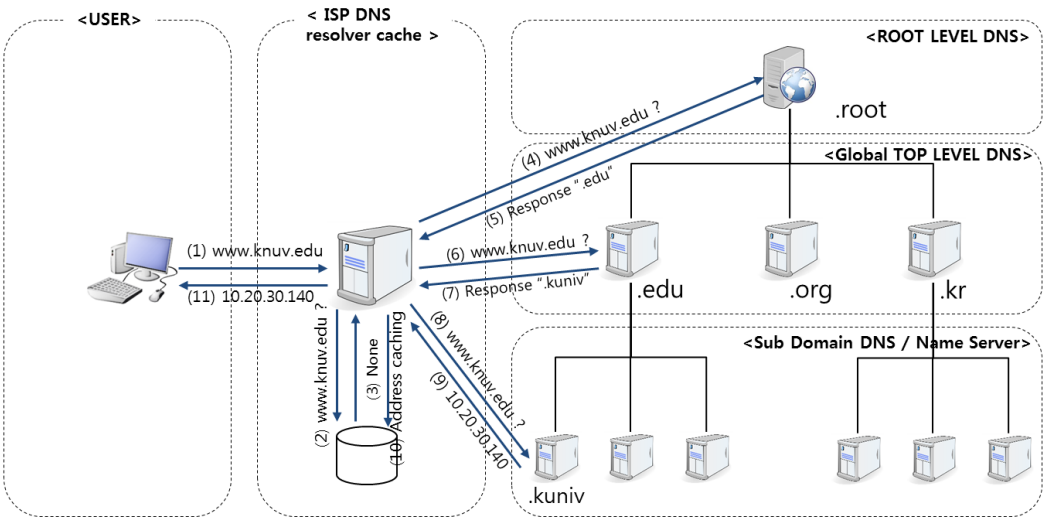


Fig. 1. Example of DNS query requests and responses

우 캐쉬 DNS로 질의를 전송한다. Cache DNS로 질의를 전송하면, Cache DNS는 자신의 캐쉬 내에서 해당 정보를 확인하고 응답하게 된다. 정보가 없는 경우 타 DNS로 순차적으로 질의하여 정보를 가져오게 된다. 일반적으로 순차적으로 질의 하는 순서는 ROOT DNS → Global Top-Level Domain → Name Server 의 순서로 질의하며, Name Server에서 응답받은 결과를 Cache DNS는 자신의 캐쉬 내에 정보를 저장하고, 사용자에게 전달하고 한다.

이때 리소스 레코드(Resource Record)를 이용하게 되는데 보통 해당 도메인 네임(domain name)과 IP주소를 매핑하여 놓은 zone 파일을 사용하여 해당 도메인 네임이 가지는 속성 정보를 지칭하여 사용하고 있다. 예를 들면 www.knuv.edu의 IP주소가 10.20.30.140인 경우 "www.knuv.edu 1800 IN A 10.20.30.140" 이라는 형태를 사용한다. 여기서 A 타입 또는 AAAA 타입의 경우에는 각각 IPv4, IPv6 의 IP 주소가 존재함을 알 수 있으며, 해당하는 주소가 없을 경우에는 응답 실패에 해당하는 NX 타입의 응답을 하게 된다.

2.2 DNS 증폭공격(DNS amplification attack)

DNS 증폭 공격(Amplification Attack)은 DDoS 공격의 한 종류이다[2]. 이러한 공격 형태는 DNS resolver나 NTP 서버 같은 Open Internet Service를 이용해서 공격 목표에 능력 이상의 많은 양의 패킷을 보낸다. 이렇게 되면, 공격 목표는 진짜 사용자에게 필요한 결과를 보낼 수 없게 된다. DNS 증폭 공격은 다음의 세 가지 특징을 가진다. 첫째, DNS 증폭공격은 UDP 프로토콜을 사용하고, 53번 포트를 이용한다. 둘째, 약 1000개 이상의 UDP패킷을 초당 전송하여 커다란 용량의 패킷 덩어리를 만들어 낸다. 이러한 패킷 덩어리는 Open Recursion DNS 서버를 이용하여 수많은 UDP 패킷을 만들어 낸다. 셋째, 공격자는 이러한 UDP 패킷이 나올 수 있도록 유도하기 위해서 실제 서버가 있는 DNS 질의를 이용하기 보다는 허수의 다량의 질의를 발생시키게 된다. 결국 이러한 질의를 통해서 UDP가 생성되고, DNS의 응답 메시지는 확장되어 공격 목표 시스템으로 돌아가게 되는 것이다. 다시 말하면, 공격자가 DNS 증폭 공격을 쉽게 할 수 있는 이유는 DNS 질의는 대부분 UDP로 구성되어 있으며, 전 세계적으로 많은 Open DNS Cache 서버가 존재하며, DNS 질의에 대한 응답 트래픽은 질의 트래픽보다 클 수밖에 없는 프로토콜이기 때문이다(요청 시에는 100kb 정도이나 응답은 400kb이상으로 돌아옴). 이로 인해 공격자는 DNS를 통한 증폭 공격으로 DDoS 공격을 꾸준하고, 지속적으로 수행할 수 있다.

2.3 기존 연구 및 제약사항

그 동안 DNS 증폭 공격을 탐지하고 보호하는 많은 연구가 진행되어져 왔다. 먼저 Ye 등[5]은 DNS 증폭 공격에 대한 아키텍처를 제안하였다. 이는 DNS에서 증폭 공격을 직접 탐지하는 것이 아니라 각각 캐쉬 DNS와 연결된 미러링 로그에 의한 트래픽 분석을 하며, 공격으로 탐지 시에는 Blacklist를 두어 해당 IP를 제한하는 방식이다. 이러한 분산된 분석 및 통제 방식은 일정 한계 이하의 분산된 트래픽 공격을 하였을 경우 탐지하기 힘들다. 두 번째는 Yu 등[6] 이 언급한 DNS 트래픽 시각화 분석 방법이 있다. IDS 룰을 이용하여 탐지 하는 방법을 이용한 것이다. 이 논문에서는 중앙 집중형으로 트래픽량을 분석하고 있다. 하지만, 모든 트래픽 분석을 위해 분산된 캐쉬 DNS 패킷의 정보를 통째로 수집하고 있어 실시간 분석이 힘들다. 셋째로, Wei min 등[7] 이 언급한 DNS DDoS 공격 감소 방안이 있다. 이 논문에서는 TTL-expired 된 도메인에 대해서 데이터를 삭제하지 않고 유지하고 있으며, 또한, cached domain, NX domain을 남겨서 지속 유지하고 있다. 하지만, Cache를 오래 유지하고 있는 것은 DNS 저장에 대한 부하 상승을 초래하고, 실제 DNS 증폭 공격에 대한 공격은 서브도메인(sub domain)을 생성하여 공격하기 때문에, 실제 서비스에 많은 제한이 따른다. 마지막으로 Rozekrans 등[8]이 작성한 논문에서는 증폭 요소(amplification factor)를 정의 하여 측정을 하였다. 해당 요소는 (질의 패킷 사이즈 / 응답 패킷사이즈)로 정의하여 측정하였다. 하지만, 특정 트래픽 사이즈로 비율을 정하게 되면, 동일한 질의라 하더라도 DNS 속성이 IPv4일 때와 IPv6 일 때 모두 다르게 된다. 그러므로 동일한 행위에 대한 판단결과가 매번 다를 수 있다. 또한, 위 논문들은 실험을 위한 제한된 수량의 데이터 환경에서 구현을 한 결과로 실제 적용하기에

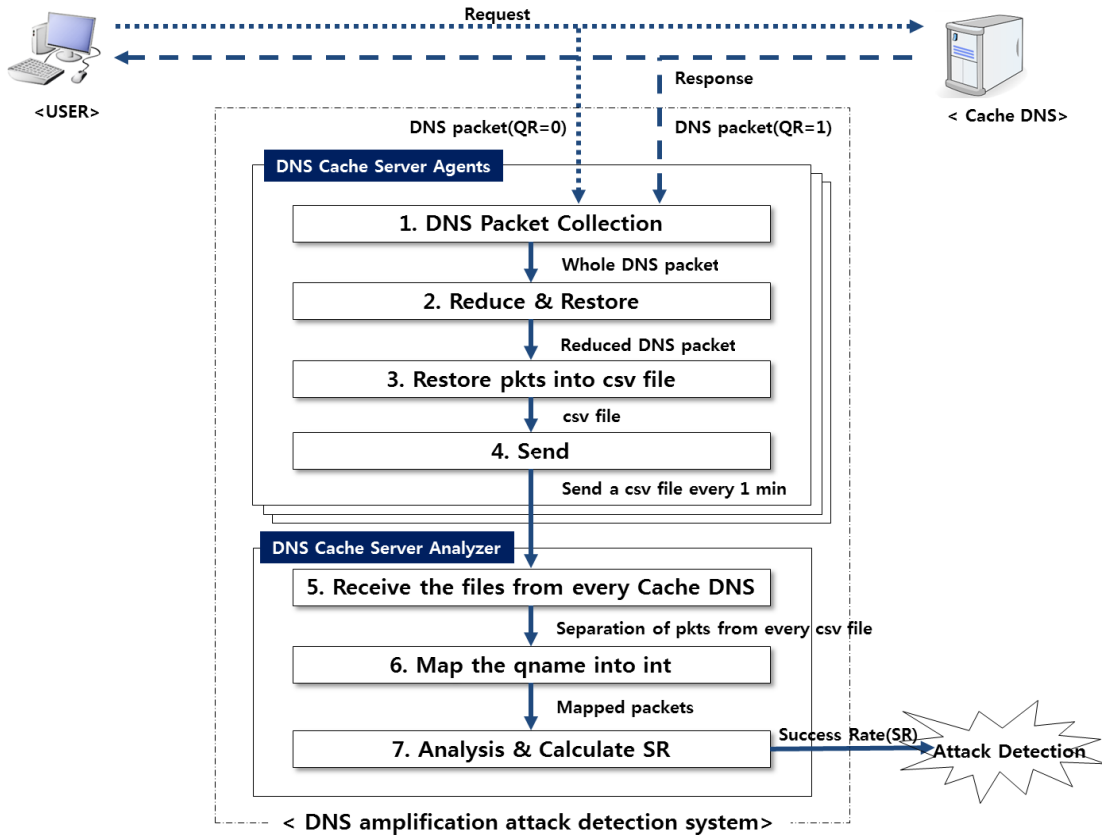


Fig. 2. Overview of proposed DNS amplification attack detection system

는 많은 제약 사항이 있다.

그러므로 위 제약사항을 극복하기 위해서는 다음의 보안 요구사항이 필요하다. 첫째, 중앙 집중형으로 트래픽만 분석하는 것 보다, 실효성 있는 DNS 패킷을 중앙 집중형으로 분석해야 한다. 왜냐하면, 각 Cache DNS의 트래픽 제한 한계를 극복할 수 있는 공격이 존재하기 때문이다. 둘째, 임의의 서브도메인 공격을 탐지하기 위한 실제 성공이 되는 트래픽만 분석을 해야 한다. 왜냐하면, 증폭 공격은 NX 타입의 응답을 유도하는 공격이 주류를 이루고 있고 이를 위해서는 임의의 서브도메인 명을 생성하여 공격하기 때문에 실재하는 도메인 주소와 존재하지 않는 도메인을 분리하여 계산하여야 한다. 셋째, 근실시간 분석을 위한 Cache DNS의 저장 간격을 최소화해야 한다. 그러므로 본 논문에서는 위의 제약사항을 극복하고 DNS 증폭 공격을 효율적으로 탐지할 수 있는 알고리즘과 다량의 DNS 패킷을 근실시간으로 분석할 수 있는 방법을 제안하겠다.

III. DNS 증폭공격 탐지를 위한 시스템 구조

시스템은 DNS Cache 서버에 설치되어 DNS Traffic을 수집하고, 전송하는 DNS Cache Server Agent(이하 Agent)와 Agent에서 전송된 데이터를 근 실시간 가공, 분석하여 결과를 보여주는 DNS Query Analyzer(이하 Analyzer)로 구성되어 있다. Analyzer는 각 DNS Cache 서버에서 수집된 결과를 수신(Receive)하고, 이를 분류 및 저장(Mapping & Restore), 분석(Analysis)을 한다. 전체적인 시스템 구조도는 Fig. 2. 와 같다.

3.1 DNS Cache Server Agent

Agent는 Cache 서버 당 설치되는 Traffice 수집 및 가공을 하기 위한 프로그램이다. Agent는 수집(Collect), 가공(Filter and Reduce), 저장(Restore), 그리고 전송(Send) 역할을 한다.

Agent는 트래픽을 수집하기 위해 packet capture(libpcap)를 사용하여 트래픽을 수집한다. DNS Packet을 수집하면, RFC883에 따라 Table.1.과 같은 정보를 획득할 수 있다[9]. Transaction ID란, 임의의 질의를 발생하는 프로그램에 의해 할당되는 16bit 식별자이다.

이러한 식별자는 대응하는 응답에 복사되고 질의자의 질의에 대해 매칭되는 응답인지를 확인하는데 사용될 수 있다.

QR은 메시지가 질의 또는 응답인지를 나타내는 1bit 필드이며 설명은 Table. 2. 와 같다.

RCODE는 Response CODE를 의미한다. 4bit필드로서 해당하는 값에 대한 구체적 내용은 Table. 3. 과 같다.

Agent는 패킷을 다음과 같이 가공하여 다음과 같은 정보만 CVS 데이터로 남긴다.

system time, src ip, dst ip, transaction id(TID), qr, rcode, qname, qtype, qclass

Agent는 micro second 기준으로 60초 동안 저장 후, Analyzer로 데이터를 전송한다. 60초는 100대 이상의 Agent가 Analyzer로 전송할 때, Analyzer에서 지연없이 데이터를 처리할 수 있기 때문이다. 만약, 60초 동안 처리하는 데이터가 많은 경우, 시간을 더 줄일 수 있으나, 이는 DNS Cache 서버에 부하를 야기할 수 있고, DNS 서비스에 영향을 미칠 수 있으므로, 60초를 기준으로 한다. 전송한 데이터는 정해진 기간 동안 DNS 시스템 내에 보관하고, 30일 후에는 삭제한다.

Table 1. The structure of DNS Packet

Source IP Address									
Destination IP Address									
Source Port									
Destination Port (=53)									
Transaction ID (Random No)									
QR	Opcode	AA	TC	RD	RA	Z	AD	CD	RCODE
QDCOUNT									
ANCOUNT									
NSCOUNT									
ARCOUNT									
QNAME				QTYPE			QCLASS		

Table. 2. The description of QR field

QR	Description
0	Request
1	Response

Table. 3. The description of RCODE field

RCODE	Description
0	No error condition
1	Format error - The name server was unable to interpret the query.
2	Server failure - The name server was unable to process this query due to a problem with the name server.
3	Name Error - Meaningful only for responses from an authoritative name server, this code signifies that the domain name referenced in the query does not exist.
4	Not Implemented - The name server does not support the requested kind of query.
5	Refused - The name server refuses to perform the specified operation for policy reasons. For example, a name server may not wish to provide the information to the particular requestor, or a name server may not wish to perform a particular operation (e.g. zone transfer) for particular data.

3.2 DNS Query Analyzer

Analyzer는 각 Agent로부터 CVS 데이터를 수신한다. 수신하는 데이터를 Database에 저장하기 전 qname을 mapping하여 저장한다. qname을 빠르게 처리하기 위해 qname별 int 값으로 생성 후, mapping 한다. src ip와 dst ip는 mariaDB에서 제공하는 INET_ATON 함수로 정수로 변환하여 저장하고, 그 외 나머지 항목은 value를 그대로 Database에 각 항목별로 저장한다. ip에 대해 확인하는 경우는 INET_NTOA함수를 사용한다.

분석 단계에서는 데이터 저장 시 아래의 표와 같이 저장한다. 저장된 데이터를 이용하여, 아래와 같이 저장하질의 수와 응답성공을 분단위로 계산하여 저장한다.

qname에 대한 QR(0), QR(1), Rcode(0), Rcode(!0)을 1분 단위로 카운트하여 저장한다. 전체 질의를 보기 위해 저장된 모든 QR(0)과

```
Analysis Algorithm(1 min_pkts){
        Request_pkt = 0;   // Initialize the total number of request pkts
        Response_pkt = 0;  // Initialize  the total number of response pkts
        Suc_Res_pkts = 0;  // Initialize  the number of response pkts which domains
exist

        For(every pkt){
        if QR = 0, {Request_pkt ++;  // Check request pkt
        }
        else {Response_pkt ++;  // Check response pkt
                if Rcode = 0, then Suc_Res_pkts ++;   // Count the number of
successful pkt
                }
        }
        SR = Suc_Res_pkts / (Request + Response);  // DNS query Success Rate in 1
min

        return SR;
}
```

Fig. 3. Analysis Algorithm

QR(1)을 분리하여 1분 단위로 카운트 하여 저장한다. 또한, 전체 응답성공률을 보기 위해 저장된 모든 Rcode(0) 카운트를 1분단위로 저장한다. 저장된 QR(0), QR(1), Rcode(0)를 계산하면 전체 응답성공률(Success Rate)을 계산할 수 있다. 위에서 설명한 분석 알고리즘은 Fig. 3. 과 같다.

실시간 출력은 하기 위해 DNS 질의를 나타내는 화면과 응답성공률을 나타내는 화면을 생성한다. DNS 질의를 나타내는 화면은 X축은 system time, Y축은 60초간 QR(0)를 표시한다. 응답성공율을 나타내는 화면은 X축은 system time, Y축은 응답성공율로 표시한다.

IV. 시스템 구현 결과 및 분석

4.1 시스템 구현 환경

제안하는 알고리즘 검증을 위해서 ISP망에 Open DNS로 구성된 CentOS 6.4기반의 리눅스 서버에 BIND 9.9를 설치하고, 98대의 Cache DNS에서 트래픽을 수집하고 분석하였다. 트래픽 수집을 위한 DNS Agent는 C기반에 pcap 라이브러리를 이용하여 작성하였으며, DNS Traffic Analyzer 시스템은 빠른 분석을 위해 메모리 기반의 저장장치를 이용한 분석 영역과 저장 영역을 나누었으며, Apache+PHP+MarisDB를 이용하여 웹에서 모니터링 및 분석을 할 수 있도록 하였다.

각 Cache DNS의 Agent에서 저장, 분류하고 전송되는 데이터는 Analyzer 서버의 DB로 바로 저장될 수 있도록 하였다. Agent에서 전송되는 데이터를 DB로 저장하는 방식은 흔히 사용하는 Web 기반의 전송방식으로 MariaDB에 직접 저장되도록 하였다. 그리고 DB에는 전송된 데이터를 분류하여 저장하도록 하였다.

그리고 우리는 데이터 분석을 위해서 2014년 11월 9일부터 약 2주 동안 2,296,794,604건의 데이터를 수집 / 분석하였으며, 4번의 작은 공격과 2번의 큰 공격을 확인하였다.

4.2 일반적인 DNS 질의 요청 패턴과 비정상적인 DNS 질의 패턴

일반적인 DNS 질의 트래픽은 시간대 별 활동하는 사용자의 수에 따른 전형적인 패턴을 보인다. 사용자가 늘어나는 시간에는 질의 트래픽이 늘어나며, 사용자가 줄어드는 시간에는 질의 트래픽이 줄어드는

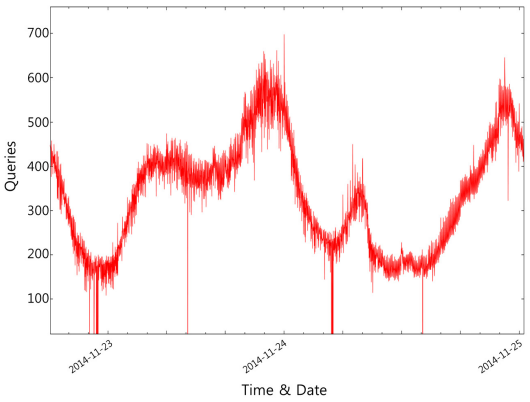


Fig. 4. Normal DNS Request Queries Example

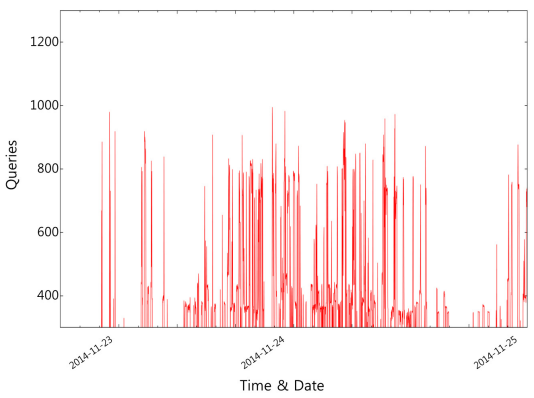


Fig. 5. Abnormal DNS Request Queries Example

형태의 패턴을 보이며 시간대별 사용자 요청 질의 수는 Fig. 4 와 같다.

하지만, Fig. 5. 와 같은 특정 도메인에 대한 임의의 질의 패턴을 보이기도 한다. 이러한 패턴은 실제 공격 발생 시 나타나는 패턴이며, 다수의 이용자가 비정상적인 질의를 요청 시 나타난다.

4.3 DNS Query Analyzer의 SR 분석을 통한 DNS 증폭공격 탐지

Fig. 5.의 질의 패턴은 실제 공격이 일어나는 것인지 일시적인 현상인지 구분하기 힘들다. 왜냐하면, 정상적인 서비스이지만, 사용자가 특정 시간에 접속이 급증하는 서비스의 경우에도 비슷한 패턴이 나타나기 때문이다. [5, 6, 7, 8]의 경우에도 질의 중심의 트래픽 분석으로 오탐의 가능성이 높다. 우리는 지난 2주간의 DNS 캐쉬 서버 100대를 분석하였고, 응답성공률을 이용한 공격 탐지 기준을 다름과 같이 지정하였다. 따라서, 우리는 다음과 같은 기준을 정한 후 SR 분석을 실시하였다.

1) 응답성공률 99% 이상은 정상 서비스
2) 응답성공률 99% 미만은 공격 발생

상기의 기준은 '13년 12월부터 실 DNS 데이터를 수집하고, 쿼리 및 트래픽 분석을 통한 공격 징후 탐지를 수행하였다. 실제 인터넷 이용환경에서 잘못된 서비스 설정, 프로그래밍 오류, 개발자의 실수 등으로 인해 1% 이내에는 공격은 아니지만, 정상적인 응답을 할수 없는 쿼리가 존재함을 확인하였다. 이러한 테스트 및 시행착오를 통해 응답성공률이 99%미만으로 떨어졌을 경우 공격징후가 일어난다고 판단할 수 있다. 응답성공률 99% 기준은 '14년 6월 23일부터 설정하였으며, 상기 기준으로 '14년 12월 15일(현재)까지 약 6개월 동안 총 455건의 공격 시도 및 공격을 탐지하였다.

본 논문에서 제안하는 시스템을 이용한 결과 Fig. 6.과 같이 빠른 탐지를 위한 그래프를 이용한 시각화가 가능하였으며, 1분당 약 17,000,000개의 DNS 질의 응답 패킷 분석이 가능하였다.

또한, [7, 8]에서 제안했던 실험실 환경의 분석과는 다른 대용량, 근실시간으로 공격을 인지할 수 있었다. 그리고, 단순 쿼리 수 분석이 아닌, SR 분석

Table 4. Comparison our proposed DNS amplification detection system with other detection system

	Proposed Detection System	[5]	[6]	[7]	[8]
Real-Time Detect	O	O	O	X	X
Threshold	O	X	X	O	O
Experimental Data	O	O	O	O	O
Real-Life Data	O	X	X	X	X
Real-Time Visualization	O	X	O	X	X
Enterprise Monitoring	O	X	X	X	X

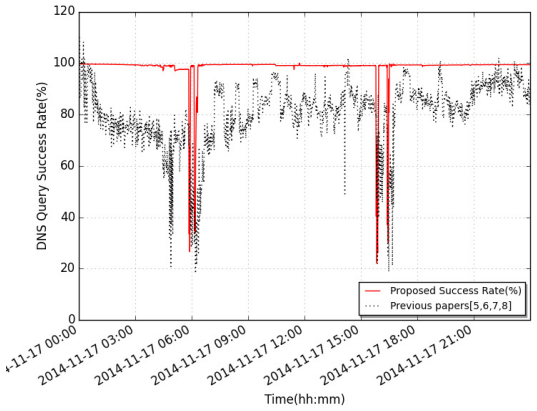


Fig. 6. DNS query analysis with Success Rate

을 통해서 실제 비정상적인 쿼리의 분포를 쉽게 확인할 수 있었다.

V. 결론 및 향후 연구 과제

정상적인 DNS 질의 트래픽은 일반 사용자들이 정상적으로 인터넷 서비스를 이용함에 대한 패턴이다. 비정상적인 패턴은 바이러스, 봇넷 프로그램과 같은 악성 프로그램이 발생시킨다. 사람이 아니라 프로그램이 발생시키는 인터넷 서비스 사용 패턴은 사람이 발생시키는 패턴과 확연히 다르다.

그래서 우리는 새롭게 질의 응답 성공률(SR)을 이용한 DNS 공격 탐지 방법을 제안하였으며, 기존의 논문[5, 6, 7, 8]에서 언급한 공격에 대한 판단 모호성과는 구분되는 실제 공격을 탐지 가능한 수준으로 끌어올렸다.

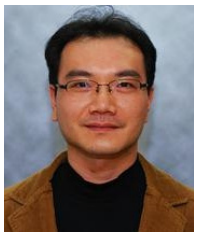
또한, 우리 시스템을 적용하면 근실시간 분석을 통해서 실험실 수준이 아닌 현장에서 직접 사용할 수 있는 DNS 트래픽 분석 시스템 구조를 구축하였다.

하지만, 본 연구의 한계는 어디까지나 DNS 증폭 공격을 효율적으로 탐지하는 것에 있다. 다시 말하면, 해당 공격에 대한 대응은 제한적이다. 그러므로 실시간 IP 주소와 DNS 요청질의의 연관성 분석하고 자동으로 연관 DNS와 IP를 차단할 수 있는 알고리즘에 관한 연구가 필요하다.

References

[1] Korea Network Information Center, http://krnic.or.kr/jsp/dns/dnsInfo/dnsInfo.jsp
[2] http://technet.microsoft.com/en-us/security/hh972393.aspx
[3] www.ahnlab.com
[4] www.symantec.co.kr
[5] YE, Xi, and Yiru YE. "A Practical Mechanism to Counteract DNS Amplification DDoS Attacks." Journal of Computational Information Systems 9:1, pp. 256-272, 2013.
[6] Yu, Huiming, et al. "A visualization analysis tool for DNS amplification attack." Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on. Vol. 7. IEEE, 2010.
[7] Wei-min, Li, Chen Lu-ying, and Lei Zhen-ming. "Alleviating the impact of DNS DDoS attacks." Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on. Vol. 1. IEEE, 2010.
[8] Rozekrans, Thijs, Matthijs Mekking, and Javy de Koning. "Defending against DNS reflection amplification attacks." University of Amsterdam, Tech. Rep, Feb. 2013.
[9] RFC 883 "DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION"

〈저자소개〉



이 기 택 (Ki-taek Lee) 종신회원
2000년 2월 : 동아대학교 기계공학과 졸업
2001년~2003년: (주)해커스랩
2005년~현재: SK브로드밴드 정보기술원 매니저
2011년~현재: 고려대학교 정보보호대학원 정보보호학과 석박사통합과정
〈관심분야〉 정보보호, 시스템 보안, 네트워크 보안, IT서비스 보안, IoT 보안

백 승 수 (Seung-soo Baek) 정회원
2002년 3월: 육군사관학교 전산학과 학사
2007년 9월: 미. 해군대학원(Naval Postgraduate School) 전산학과 석사
2009년~2012년: 제 3 야전군 사령부 참모
2012년~현재: 고려대학교 정보보호대학원 박사과정
2012년~현재: 육군사관학교 전자정보학과 컴퓨터과학 조교수
〈관심분야〉 정보보증, 접근제어, 사이버전

김 승 주 (Seung-joo Kim) 종신회원
1994년~1999년: 성균관대학교 정보공학과 (학사, 석사, 박사)
1998년~2004년: KISA(舊한국정보보호진흥원) 팀장
2004년~2011년: 성균관대학교 정보통신공학부 부교수
2011년~현재: 고려대학교 정보보호대학원 정교수
2002년~현재: 한국정보통신기술협회(TTA) IT 국제표준화 전문가
2004년~현재: 한국정보보호학회 이사
2005년~2006년: 교육인적자원부 유해정보 차단 자문위원
2007년: 국가정보원장 국가사이버안전업무 유공자 표창
2007년~2009년: 전자 정부 서비스 보안 위원회 사이버 침해사고대응 실무위원회 위원
2010년: 방송통신위원회 정보통신망 침해사고 민관합동조사단 위원
2012년 3월~2012년 6월: 선관위 디도스 특별검사팀 자문위원
2013년 4월~2013년 12월: IT보안인증사무국 자문위원
2013년 9월~현재: 중앙선거관리위원회 자문위원
2014년 3월~현재: 헌법재판소 자문위원
〈관심분야〉 보안공학, 암호이론, 정보보증, 정보보호제품 보안성 평가, IoT보안