리소스 레코드 부분암호화를 이용한 DNS 변조공격 탐지 프로토콜 연구

심재화*, 민재원*, 최영현*, 정태명**
*성균관대학교 전자전기컴퓨터공학과
**성균관대학교 정보통신공학부
{jhsim, jwmin, yhchoi}@imtl.skku.ac.kr*, tmchung@ece.skku.ac.kr**

A Study on DNS Poisoning Attack Detection Protocol Based on Partial Encryption of Resource Record

Jae-Hwa Sim*, Jae-Won Min*, Young-Hyun Choi*, Tai-Myoung Chung**
*Dept of Electrical and Computer Engineering, Sungkyunkwan Univ.
**School of Information Communication Engineering, Sungkyunkwan Univ.

요 약

최근 인터넷을 이용한 금융거래가 활발해지면서 피싱이나 파밍과 같은 공격을 통한 개인정보 유출 사고가 빈번히 발생하고 있다. 특히 파밍의 경우, 공격자가 DNS 정보를 변조하여 사용자가 올바른 URL을 입력하더라도 악의적 사이트로 컴퓨터가 접속을 하기 때문에 위험성이 매우 높다. 이러한 공격들을 방지하기위하여 여러 연구가 진행되었지만, DNS 정보의 검증을 위한 추가적인 절차를 필요로 하거나 과도한 네트워크 트래픽을 유발할 수 있는 문제점을 가지고 있다. 따라서 본 논문에서는 이러한 문제점을 극복하고자 DNS 리소스 레코드(Resource Record)의 부분 암호화를 이용하여 DNS 변조 공격을 탐지 하는 프로토콜을 제안한다.

1. 서론

최근 인터넷을 통한 금융 거래가 활발해지면서, 개인 금융정보 유출과 관련된 사고들이 빈번히 발생하고 있다. 이 중에서 사용자의 개인정보를 가로채는 공격들에는 대표적으로 피싱(Phishing)과 파밍(Pharming)이 있다[4].

피싱은 개인정보(Private)와 낚시(Fishing)의 합성어로 불특정 다수에게 메시지나 메일을 보내서 특정 웹사이트로 유도해 이용자들의 개인정보를 탈취하는 수법이다. 피싱 공격은 사용자가 위조된 웹사이트로 접속을 하더라도 주의를 기울인다면 정상 URL과 유사한 URL을 확인하여 피해를 막을 수 있다. 파밍 공격은 피싱보다 한 단계 더 발전된 공격 방법으로써 DNS 캐시 포이즈닝(DNS Cache Poisoning)과 같은 DNS(Domain Name Service) 공격을 통하여 사용자가 URL주소를 이용하여 접속하고자 하는 사이트의 IP주소를 요청하더라도 DNS는 위조된 웹사이트의 IP주소를 반환한다. 이용자는 올바른 URL 주소를 가지고 사이트에 접속을 시도하더라도 실제로는 악성사이트로 접속이 되기 때문에 사용자는 스스로 위조된 사이트에 접속한지 여부를 판단할 수 없다. 이러한 이유로 파밍 공격을 통해 공격자는 사용자의 개인정보를 피싱 공격보다 쉽게 유출 할 수 있다.

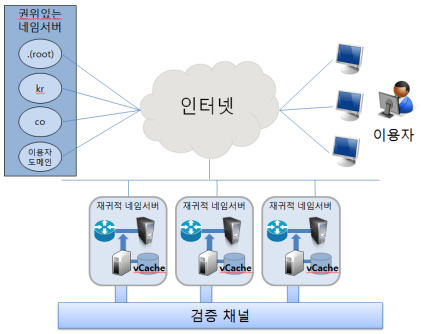
이러한 파밍 공격을 극복하고자 기존의 DNS 정보 전달 체계에 신뢰성을 부여할 수 있는 방법들이 제안되었다[3][5]. 하지만 제안된 방식들은 과도한 네트워크 트래픽 증가를 야기하거나 DNS 캐시의 변조 여부를 탐지하기 위해 추가적인 통신채널을 필요로 하여 실제 DNS 체계에 적용하기에는 많은 어려움을 가지고 있다[3]. 또한 IETF (Internet Engineering Task Force)에서 제안한 DNSSEC은 DNS가 가지는 근본적인 취약점을 보완하기 위하여 RRSIG(Resource Record Signature)등의 항목을 추가하여 신뢰성을 제공한다. 하지만 이로 인해 전체 패킷 사이즈가 증가하여 네트워크 트래픽이 증가하는 문제점이 존재한다.

따라서 본 논문에서는 이러한 단점들을 해결하고자 DNS 정보에 부분 암호화를 통한 DNS 공격탐지 프로토콜을 제안한다. 개인키를 통해 DNS 서버가 DNS 정보를 서명하고 사용자가 이를 공개키로 검증할 수 있게 하여 DNS 정보에 신뢰성을 보장하면서 불필요한 암/복호화 과정을 제거하여 오버헤드 또한 최소화였다.

논문의 구성은 다음과 같다. 2장에서는 기존에 제안된 DNS 공격 차단 방법과 IETF에서 제안한 DNSSEC에 대해서 소개를 하고, 3장에서는 본 논문에서 제안하는 DNS 변조공격 탐지 프로토콜에 대해 설명한다. 4장에서는 DNSSEC과 본 논문에서 제안하는 프로토콜을 비교 분석한다. 마지막으로 5장에서는 향후 수행해야 할 연구 내용에 대해 언급하고 결론을 맺는다.

2. 관련연구

2.1 DoX(Domain Name Cross Referencing)



(그림 1) DoX의 구조[3]

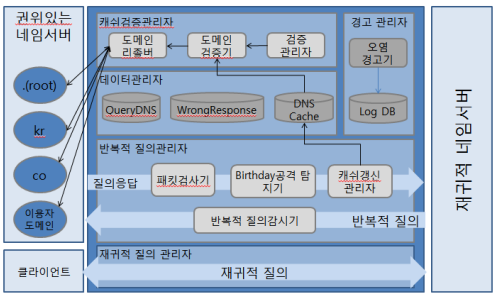
기존의 DNS 서버에서 발생하는 DNS 캐시 포이즈닝을 막기 위해서 Lihua Yuan 등은 여러 개의 RNS(Recursive Name server)들의 정보 공유를 통해서 캐시 오염을 막는 시스템을 제안 하였다[3].

RNS들은 DNS 캐시 포이즈닝을 막기 위하여 Peer-to-Peer(P2P) 방식으로 연결해 저장된 캐시 정보들을 공유한다. 이때 RNS는 대량의 리소스 레코드를 공유하기 위하여 Verification Channel이라는 독립적인을 데이터 전송 통로를 설정한다. 독립적인 경로를 통해 교환된 캐시 정보들은 Verification Cache라고 하는 반영구적인 장치에 데이터가 저장이 되고 이때 공유된 정보가 서로 다를 경우에 해당 캐시 정보가 오염되었다고 판단한다.

하지만 제안된 방식에서는 리소스 레코드를 보관하고 있는 RNS는 캐시 값의 오염 여부를 판별하기 위해 추가적인 Verification Channel을 구성해야 한다. 또한 최신의 데이터를 업데이트 하기위해 많은 양의 DNS 정보를 주고받아야 하는데 이는 해당 네트워크의 트래픽을 과도하게 증가 시켜 네트워크에 부담을 줄 수 있다.

2.2 Recursive DNS의 캐시 정보 신뢰성 향상 기법

해당 연구에서는 DNS 캐시 포이즈닝 발생하기 전에 나타나는 공격 패턴을 기반으로 DNS 캐시의 신뢰성을 향상시키는 방법을 제시 하였다[5].



(그림 2) DNS 캐시 포이즈닝 탐지 시스템 구조[5]

RNS의 관점에서 DNS 캐시 포이즈닝은 발생하기 전에 크게 2가지의 증상을 보인다. 첫 번째는 RNS가 ANS(Authoritative Name Server)에 질의응답을 할 때 TID(Transaction ID)를 이용하여 자신이 전송한 패킷인지 여부를 확인한다. 이 때 공격자는 TID의 값을 임의로 작성하여 ANS 응답확인 패킷인 것처럼 위조한다. 이 과정에서 RNS는 자신이 요구하지 않은 임의의 TID값을 가진 수많은 패킷을 수신하게 된다. 두 번째로 DNS 캐시 포이즈닝 공격이 성공한 경우 RNS는 올바른 응답 메시지를 두 번 전송 받게 된다. 이러한 두가지 사실을 바탕으로 제안된 논문에서는 DNS 서비스 체계상에서 독립적으로 잘못된 캐시 정보를 탐지 할 수 있는 기법을 제시 하였다.

제안된 연구에서는 DNS 체계의 변경 없이 적용이 가능하나 해당 논문에서 제시하는 DNS 공격 검출 시스템을 모든 DNS서버에 적용해야하는 어려움이 있다. 그리고 DNS 응답메시지를 기반으로 공격 여부를 판단하기 때문에 DNS 하이재킹과 같은 다른 공격에 취약할 수 있다.

2.3 DNSSEC

기존의 DNS는 DNS 캐시 포이즈닝 공격과 같은 위협에 취약하다는 문제점을 가지고 있다. 이러한 침해 공격으로부터 DNS 데이터베이스 시스템을 근본적으로 보호해야 할 필요성이 제기 되었다. DNSSEC은 이러한 취약점을 극복하기 위해 IETF에서 개발된 DNS 확장 표준 프로토콜이다[6].

DNSSEC은 리스소 레코드로 구성된 DNS 정보에 공개키 암호화방식의 전자서명 메커니즘을 적용하여 DNS 정보의 무결성과 인증을 제공하는 방법이다. 따라서 DNS 정보를 공격하여 발생할 수 있는 MITM 공격 및 DNS 캐시 포이즈닝 대처 할 수 있도록 설계 되었다.

2.3.1 DNSSEC 작동방식

사용자가 특정 웹사이트에 접속할 때 해당 유저 컴퓨터의 스텁 리졸버(Stub Resolver)는 RNS를 통해 ANS에게 질의한다. 먼저 데이터를 전달받은 서버는 공개키를 통하여 ANS에게 해당 웹사이트의 IP주소를 요청하게 된다. 이때 IP주소만 요청하는 것이 아니라 DNSSEC 키를 함께 요청한다. 해당 ANS는 암호화 된 리소스 레코드 뿐만 아니라 이를 검증하기 위한 추가적인 4개의 보안 리소스 레코드를 함께 전달하여, 전송도중 변조여부를 판별 할 수 있다.

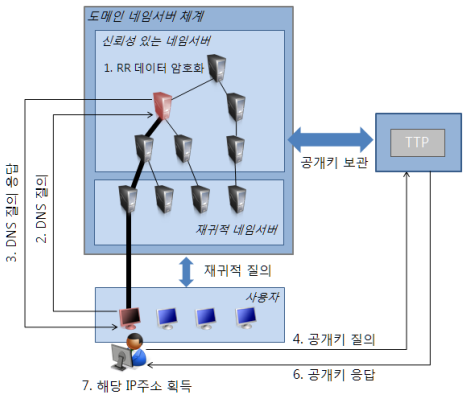
2.3.2 DNSSEC 문제점

DNSSEC의 설정은 루트 도메인에 DNSSEC이 설정되어야 이 루트 도메인으로부터 신뢰할 수 있는 데이터를 제공 받을 수 있다. 안전하게 DNS 리소스 레코드를 전달할 수 있지만 DNSSEC을 적용함에 있어 발생하는 실질적인 오버헤드 때문에 현재 DNSSEC 적용은 한정적으로 이

루어지고 있다[7].

3. 제안 모델

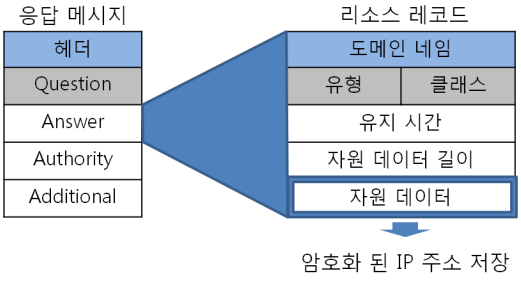
본 절에서는 DNSSEC보다 작동이 가볍고 더불어 DNS 캐시 포이즈닝 같은 DNS 공격을 차단할 수 있는 새로운 프로토콜을 제안한다. (그림 3)는 제안하는 프로토콜의 구조를 나타낸다.



(그림 3) 제안하는 모델

3.1 초기 리소스 레코드의 암호화 데이터 생성

기존의 DNS의 리소스 레코드 위임체계에 비대칭 키를 이용하여 보안을 강화한다. 먼저 ANS들은 비대칭키쌍을 생성한다. 그 중 개인키는 DNS 서버 자체에 안전하게 보관하고 대응하는 공개키는 TTP(Trusted Third Party)에 전달하여 보관을 한다. 본 논문에서는 공개키를 제공하는 TTP는 안전하다고 가정한다. 각각 DNS 서버에 등록된 리소스 레코드의 IP정보는 자기의 개인키로 암호화를 해서 저장하고 있다. ANS들은 루트서버로부터 전달받은 DNS 정보들을 캐싱하여 보관하는데, 이때 리소스 레코드의 IP정보가 암호화된 상태 그대로 캐싱한다. 암호화된 IP 정보를 가지는 리소스 레코드는 DNS 질의가 있을 때 해당 응답메시지에 포함되어 사용자에게 전달된다. (그림 4)는 사용자에게 응답되는 DNS 응답메시지와 리소스 레코드를 나타낸다.



(그림 4) DNS 응답메시지와 리소스 레코드

3.2 IP정보 암호화를 통한 안전한 리소스 레코드 전달

사용자는 먼저 접속하고 싶은 사이트가 있을 경우 자체 PC의 DNS 캐시에 도메인 질의를 하여 해당 도메인의 IP주소가 있는지 여부를 확인한다. 만약에 IP정보가 없다면 사용자는 RNS에 접근하여 해당 IP정보 유무를 확인한다. 만약 해당 도메인의 IP정보가 있을 경우 사용자에게 IP정보를 응답하지만 IP정보가 없을 경우 상위 DNS 서버로 DNS 질의를 진행 한다. 마지막으로 해당 도메인을 가지고 있는 DNS에 도달하면 원하는 도메인에 대한 IP주소를 질의 하고, 그에 대한 암호화 된 IP주소를 획득하게 된다. (그림 4)는 재귀적 질의방식에서 실제 DNS 정보를 가지고 있는 해당 DNS 서버와 사용자간의 DNS 정보 가 전달되는 과정을 나타낸다.

해당 DNS에서 암호화가 된 IP주소를 가진 리소스 레코드를 전달하더라도 경유하는 DNS 서버들은 암호화된 IP주소를 복호화하지 않고 캐시에 DNS 정보를 보관, 전달하는 역할만 한다. 최종적으로 리소스 레코드를 받은 사용자는 처음 도메인 정보를 등록 받은 DNS가 IP주소를 암호화할 때 쓴 개인키에 상응하는 공개키를 TTP로부터 획득한 후 IP영역을 복호화하여 요청하였던 도메인 주소의 IP를 확인할 수 있게 된다.

제안된 방식으로 리소스 레코드의 정보를 보낼 경우 악의적인 공격자가 DNS 정보 위조를 시도하더라도 IP주소를 암호화한 서버의 개인키를 알 수가 없어 다른 IP주소로 변조를 할 수가 없다. 따라서 IP주소를 가지고 있는 DNS만 올바른 개인키로 암호화가 가능하기 때문에 신뢰성 있는 DNS 정보의 전달이 가능하다. 만약 DNS 정보가 공격자에 의해서 변조가 되더라도 리소스 레코드의 IP주소를 해당 DNS의 공개키로 복호화했을 때 IP주소 형태의 데이터가 나올 확률은 굉장히 낮기 때문에 DNS 정보의 변조여부를 판단할 수 있다. 이에 따라 MITM(Man In The Middle attack)공격이나 DNS 캐시 포이즈닝 같은 DNS 변조 공격들을 탐지할 수 있다.

4. 평가

2장에서 제시한 기존의 다른 프로토콜과 본 논문에서 제안하는 프로토콜에 대해 비교 설명 한다.

<표 1> DNS 공격 방지 기법 비교

구분	DoX	DNS캐쉬 신뢰성 향상기법	DNSSEC	제안방법
패킷크기	동일	동일	증가	동일
적용방법	캐시 공유	DNS 변조 징후 탐지	전자서명	전자서명
오버헤드	검증 채널 생성	탐지 시스템 구축	RR 확장, 서명 검증	IP주소 암/복호화

4.1 전달되는 정보의 크기

본 논문에서 전달되는 메시지크기는 기존의 DNS 메시

지와 같다. 단지 리소스 레코드에 저장되어 있는 IP가 암호화 되어 있는 값이라는 점이 다른점이다. 반면에 DNSSEC같은 경우는 사용자에게 안전한 DNS 정보의 전달을 위해서 4가지의 리소스 레코드를 추가해서 사용자에게 전달한다. DNS 정보 전달에 많은 헤더와 데이터의 추가는 네트워크망이나 데이터 처리에 많은 비용을 필요로 하게 된다. 따라서 본 논문에서 제안한 방식은 추가적인 헤더나 데이터의 필요성이 없기 때문에 DNS 전달에 더 효율적이라 할 수 있다.

4.2 해당 범위

DNSSEC 같은 경우는 ANS에서 해당 RNS까지의 구간에 대해 리소스 레코드의 보안을 제공한다. DNS 정보가 RNS 서버까지 안전하게 전달이 된다 하더라도 RNS에서 사용자까지의 연결에서 안전을 보장할 수 없다. 이때 악의적인 사용자가 Birthday attack이나 MITM와 같은 공격을 시행 할 경우 DNS 정보가 탈취 될 수 있다. 본 논문에서 제안하는 방법은 DNS 정보를 가지고 있는 ANS로부터 암호화 되어 직접 질의를 하는 사용자까지 암호화된 IP정보가 전달되기 때문에 MITM이나 DNS 캐시 포이즈닝 공격으로부터 안전하다.

4.3 적용성

다른 여러 논문에서 제안된 방법의 경우 추가적인 시스템 구축이나 모듈들이 필요 하게 된다. DoX 같은 경우 추가 Verification channel과 Verification cache가 필요하게 된다[3]. DNS 캐시 신뢰성 향상 기법에서는 모든 각각의 DNS 서버마다 해당 논문에서 제안된 기법들이 적용 되어야한다[5]. 이는 개별 시스템 대한 현실적인 적용의 어려움이 있다. 하지만 본 논문에서는 신뢰 할 수 있는 TTP와 해당 ANS의 암호키쌍을 생성 할 수 있는 제너레이터(Generator)만 구현가능 하다면 쉽게 DNS 정보의 IP를 암호화하여 전달할 수 있다.

4.4 연산의 오버헤드

기존의 연구 방식에서는 DNS 캐시 오염 여부를 판별하기 위해 추가적인 Verification Channel이나 검증 시스템을 구현하였다[3][5]. 이는 추가적인 검증 연산이나 각각의 DNS에 캐시 오염 탐지 시스템을 추가함으로써 실제 DNS 전달 과정에서의 연산의 오버헤드가 발생한다. 본 논문에서 제안한 방식은 DNS 리소스 레코드를 암호화하는 DNS 서버와 이를 수신하는 사용자만 검증과정이 필요하다. DNS 정보를 등록 받은 ANS는 가지고 있는 개인키로 1회만 IP영역을 암호화하여 저장한다. 그리고 DNS 정보 요청한 뒤에 해당하는 공개키로 호스트에서 복화화를 진행한다. 후에 충분한 리소스 레코드가 ANS에 등록 되고 나서는 실질적으로 해당 DNS 정보를 요청하는 호스트에서 복호화 1회만 수행하면 된다. 경유하는 DNS 서버의 경우 해당 도메인을 확인하고 전달하거나 자체 DNS 캐시에 저장하면 되므로 추가적인 연산과정이 필요 없어 효율적으로 DNS 전달이 가능하다.

5. 결론

전 세계적으로 증가하는 인터넷 이용자 수와 기술 발전에 따른 보안 취약점의 다양화는 정보 유출과 같은 많은 문제점을 야기하고 있다. 사용자를 위조된 웹사이트로 유도한 뒤 이용자의 개인정보를 빼내는 수법인 파밍을 막기 위하여 여러 가지 기법이나 시스템이 제안되었다. 하지만 이러한 방법들은 기존의 DNS 서버에 추가적인 시스템이나 장치를 요구하여 네트워크상의 추가적인 비용을 초래하거나 네트워크상의 각각의 모든 DNS 서버에 탐지 시스템을 구현을 요구함으로서 실제 적용에 어려움을 가지고 있다.

본 논문에서는 DNS 리소스 레코드 상에서 필요한 부분만을 개인키로 암호화하여 사용자에게 전달하고 TTP를 통해 신뢰 할 수 있는 공개키를 획득한 뒤 복호화하는 방법으로 효율적으로 신뢰할 수 있는 DNS 정보를 전하는 방법을 제시 하였다. 이를 통하여 DNS 캐시 포이즈닝 공격을 효과적으로 탐지할 수 있을 것이라 예상한다.

ACKNOWLEDGEMENT

본 연구는 지식경제부 및 한국산업기술평가관리원의 산업융합원천기술개발사업(정보통신)의 일환으로 수행하였음. [10041244, 스마트TV 2.0 소프트웨어 플랫폼]

참고문헌

[1] 한국인터넷진흥원, “2012년 인터넷이용실태조사 최종보고서”, 2012년 12월

[2] CVE, "http://www.cvedetails.com/", March 2013

[3] Yuan, “DoX: A peer-to-peer antidote for DNS cache poisoning attacks”, IEEE International Conference on Communication Vol5, June 2006

[4] 고웅, “사전 검출을 통함 피싱 및 파밍 예방 시스템”, 한국인터넷정보학회지 제9권 제2호 2008년 11월

[5] 주용완, “Recursive DNS의 캐쉬 정보 신뢰성 향상 기법”, 정보처리학회지 제15권 제4호 제121호, 2008년 8월

[6] IETF, "http://tools.ietf.org/html/rfc5155", March 2013

[7] Ager, "Predicting the DNSSEC overhead using DNS traces", Information Sciences and Systems, March 2006

[8] RFC 4033: DNS Security Introduction and Requirements, March 2005.