http://dx.doi.org/10.7840/kics.2014.39B.4.200

시그니처를 이용한 향상된 Accountable 인터넷 프로토콜

박 기 태◆, 이 재 훈°, 정 희 영*

Improved Accountable Internet Protocol Using Signature

Gi-tae Park◆, Jae-hwoon Lee°, Hee-young Jung*

요 약

AIP(Accountable Internet Protocol)는 호스트의 공개키로부터 도출되는 해쉬 값을 호스트 식별자 주소로 이용함으로써 호스트에게 주소에 대한 책임 추구성(Accountability)를 제공하기 위한 미래 인터넷 구조의 하나이다. AIP에서는 하나의 호스트가 패킷을 전송하면, 중간에 있는 라우터는 패킷을 폐기한 후에 소스 주소를 검증함으로써 특히 비대칭 경로가 있는 경우에는 패킷의 폐기로 인한 성능의 저하가 발생하게 된다. 본 논문에서는 소스 호스트로부터 전송되는 첫 번째 패킷에 주소의 검증을 위한 공개키와 이의 위조를 방지하기 위한 시그니처, 그리고 재전송 공격을 방어하기 위한 타임스탬프 정보를 포함하고, 이 정보를 이용하여 패킷을 폐기하지 않고 패킷에 포함되어 있는 소스 주소를 검증할 수 있는 향상된 AIP 메커니즘을 제안한다. 제안 메커니즘의 보안 안정성을 평가하였으며, 제안 메커니즘이 보안에 강하면서도 패킷 폐기로 인한 지연을 줄일 수 있다는 장점을 가진다는 것을 확인하였다.

Key Words : Future Internet, Network security, Accountable Internet Protocol (AIP), Self-certifying address, public key/private key pair

ABSTRACT

Accountable Internet Protocol (AIP) is one of the future Internet architectures to provide accountability concept by using the self-certifying address that is derived by the public key of the host. In AIP, when a host sends a packet, a domain that is located between the source and the destination hosts discards the packet in order to verify the source IP address. Therefore, performance degradation can occur due to packet discard especially when there is asymmetric route. In this paper, we propose the improved AIP mechanism to verify the source IP address without discarding the packet by including the timestamp, public key value and the signature for protecting from forfeiting the source address. Security safety of the proposed mechanism is evaluated and the proposed mechanism can provide the more robust security as well as reducing the latency due to discarding packets.

Ⅰ. 서 론

인터넷은 단순하면서도 정교한 구조를 가지고 있어서 상당히 성공적인 기술이라 할 수 있지만 보안의 관점에서 보면 심각한 단점을 가지고 있다고 할 수 있다. 특히 IP 계층은 보안 취약점에 상당히 노출되어

※ 본 연구는 한국전자통신연구원에서 수행하는 "고품격 미래인터넷을 위한 식별자 기반 네트워킹 기술 연구" 과제의 지원으로 수행되었습니다.
◆ First Author : Dept. of Information and Communications Eng., Dongguk University, miraheel@dongguk.edu, 학생회원
° Corresponding Author : Dept. of Information and Communications Eng., Dongguk University, jaehwoon@dongguk.edu, 종신회원
* 한국전자통신연구원 ID통신연구실, hyjung@etri.re.kr, 정회원
논문번호 : KICS2014-03-090, Received March 17, 2014; Revised April 7, 2014; Accepted April 7, 2014

있다. 호스트는 자신이 전송하는 데이터 트래픽의 소스 IP 주소를 쉽게 위조할 수 있어서 인터넷에 공격이 발생하는 경우에 공격자를 추적하는 것이 어려우며 또한 공격자가 희생자라고 불리우는 제 3의 호스트의 주소로 자신의 주소를 위조하여 설정한 후에 트래픽을 전송하면, 트래픽에 대한 응답 트래픽이 희생자에게로 전송되는 반사 공격도 가능하게 된다.

하나의 호스트로부터 전송되는 패킷의 IP 주소가 다른 호스트에게 설정된 IP 주소를 위조하였는지 아니면 진짜인지를 검증할 수 있는 한 가지 방법은 자기-인증 주소를 이용하는 것이다. 자기-인증 주소는 PKI(public key infrastructure)와 같은 전역 키 관리 메커니즘 없이 주소 자체적으로 주소의 소유 여부를 검증할 수 있는 방법이다. 자기-인증 주소에서는 호스트에게 공개키와 개인키 쌍이 부여되며, 호스트의 공개키 또는 공개키에 대한 해쉬 값이 호스트의 주소로 설정된다. CGA(Cryptographically Generated Address)와 HIP(Host Identity Protocol)는 자기-인증 주소를 이용하는 프로토콜이다[1,2]. CGA는 호스트의 공개키를 기반으로 IP 주소를 생성하는 것을 목적으로 하고 있어서 소스 호스트가 접속되어 있는 도메인 외부에서는 소스 주소의 진위 여부를 판별하는 방법은 정의되어 있지 않다. 또한 HIP는 두 개의 호스트 간의 암호학적 결합을 정의하는 것으로써 멀티캐스트와 같은 점-대-다중점 방식의 통신에서는 사용할 수 없다.

AIP(Accountable Internet Protocol)은 “책임 추구성(Accountability)”이라는 개념을 이용하여 인터넷 보안을 향상시킬 수 있는 네트워크 구조이다[3,4]. AIP에서는 호스트와 도메인에 각각 별도의 공개키/개인키 쌍이 부여되며, 공개키에 대한 해쉬 값이 호스트 또는 도메인을 위한 “자기-인증” 주소로 사용된다. 공개키 기반의 자기-인증 주소를 사용하면 소스 주소가 가짜이거나 위조되었는지 아니면 원본인지를 검증할 수 있고, 만일 주소가 위조되었다고 판단되면 해당 주소를 소스 주소로 사용하는 패킷을 전송하지 않고 폐기함으로써 IP의 상위 계층에게 보안, 신뢰, 그리고 강인성 등의 장점을 제공할 수 있다. XIA나 SCION 등은 AIP에서 정의된 자기-인증 주소 체계를 이용하여 인터넷 보안을 향상시킬 수 있도록 정의된 미래 인터넷 구조이다[5-7]. 그렇지만 AIP에서는 특히 비대칭 경로가 존재하는 곳에 위치한 도메인에서 소스 주소의 위조 여부를 검증하기 위하여 호스트로부터 전송되는 첫 번째 패킷을 폐기하게 되어, 만일 소스 호스트로부터 목적지 호스트까지의 경로에 여러 개의 비대칭 경로가 존재하게 되면 그 수만큼 동일한 패킷이 폐기되는 단점이 발생하게 된다. 또한 인터넷 토폴로지의 변화로 인하여 소스 호스트와 목적지 호스트 간에 설정되는 경로가 변경되는 경우에도 패킷 폐기로 인한 단점이 발생하게 된다.

이 논문에서는 비대칭 경로가 존재하는 경우에 소스 호스트로부터 전송되는 패킷을 폐기하기 않고 소스 주소의 진위를 검증할 수 있는 향상된 AIP 구조를 제안한다. 이 논문의 구성은 다음과 같다. 제 2 장에서는 AIP의 동작과 단점에 대해서 설명하고 제 3 장에서 이 논문에서 제안하는 시그니처를 이용하여 향상된 AIP 기법에 대해서 설명한다. 제 4 장에서는 제안된 기법의 보안 관련 안정성 평가에 대해서 설명하고 제 5 장에서 결론을 맺는다.

Ⅱ. AIP(Accountable Internet Protocol) 동작과 단점

AIP에서는 AD:EID 형태의 자기-인증 주소가 사용된다[3,4]. 여기에서 AD는 호스트가 속해 있는 자치 도메인(AD: Autonomous Domain)을 나타내는 식별자이며 AD는 해당 도메인의 공개키를 입력으로 하는 해쉬 결과 값으로 도출된다. EID(End-point Identifier)는 전역적으로 유일한 호스트 식별자이며 호스트의 공개키에 대한 해쉬 값으로 결정된다. 이러한 자기-인증 주소를 이용하면 호스트와 도메인이 자신의 해당 주소를 가지고 있다는 것을 외부의 인증 기관에 의존하지 않고 증명할 수 있다.

AIP에서는 호스트가 자신이 전송하는 패킷의 소스 주소를 변조할 수 없도록 하기 위하여 uRPF(Unicast Reverse Path Forwarding) 기법을 확장한 메커니즘을 제공한다. uRPF는 소스 호스트로부터 전송된 패킷을 수신한 도메인은 패킷을 수신한 인터페이스와 도메인에서 소스 호스트로 향하는 역 경로를 위한 인터페이스가 동일한 대칭 경로의 경우에만 패킷을 다음 홉으로 전달하는 자동 필터링 메커니즘이다[8]. uRPF는 단일 인터페이스를 가진 클라이언트가 자신이 전송하는 패킷의 소스 주소 (특히 네트워크 주소 부분)를 변조하는 경우, 이를 자동으로 방지할 수 있도록 하기 위하여 클라이언트가 접속되어 있는 네트워크에 접속되어 있는 에지 라우터에서 동작하는 것이 효율적이다. 그렇지만 이 방식은 호스트가 여러 개의 네트워크에 접속되어 있는 멀티-호밍의 경우 또는 경로의 비대칭성이 존재하는 코어 네트워크에서는 동작하지 않는다. AIP에서는 uRPF 메커니즘에 경로의 비대칭성이 존

재하는 라우터에서 수신한 패킷이 유효한지를 자동으로 검증할 수 있도록 하기 위하여 uRPF에 다음의 두 번째 기법을 결합한다. AIP에서는 다음과 같이 두 부분에서 패킷의 소스 주소를 검증한다. 첫 번째로, 호스트가 접속되어 있는 네트워크에 접속되어 인터넷 서비스를 제공하는 첫 번째 홉 라우터는 호스트가 자신의 주소를 변조하지 않았다는 것을 검증한다. 두 번째로, 소스 호스트로부터 전송되는 패킷을 수신한 AD는 해당 패킷을 전송한 "이전" AD가 AD로부터 소스 호스트로 향하는 것과 동일한 (즉, 대칭) 경로인지를 검증한다. 그림 1은 AIP에서 패킷의 소스 주소를 검증하는 절차를 보여준다. 패킷을 전송하는 소스 호스트와 동일한 네트워크에 접속되어 있는 라우터(즉, 첫 번째 홉 라우터) R이 검증되지 않은 소스 호스트로부터 전송되는 패킷을 수신하면, R은 수신한 패킷을 폐기하고 검증 패킷 V를 소스 호스트에게 전송한다. 검증 패킷 V에는 R이 소스 호스트로부터 수신하여 폐기한 패킷의 소스와 목적지 AIP 주소, 패킷의 해쉬 값, 그리고 패킷을 수신한 R의 인터페이스 정보가 포함된다. R은 V를 R만 알고 있는 비밀 키인 rs(router secret)와 메시지 인증 코드(MAC: Message Authentication Code)를 이용하여 V를 검증한다. 소스 호스트는 수락, 소스 호스트의 공개키와 V, 그리고 위의 정보를 자신의 개인키를 이용하여 생성되는 시그니처 정보를 라우터로 전송함으로써 자신이 EID의 소유자임을 증명한다. 만일 호스트로부터 전송된 서명이 유효하다고 판단되면, R은 소스 호스트에 대한 정보를 등록하고 이 이후부터 소스 호스트로부터 전송되는 패킷을 다음 홉으로 전송한다. R은 소스 호스트로부터 전송된 첫 번째 패킷을 폐기하고 검증 패킷을 전송하기 때문에, 소스 호스트는 초기에 전송한 패킷을 재전송해야 한다.

만일 소스 호스트로부터 전송된 패킷이 소스 호스트가 접속되어 있는 AD의 경계를 지나서 전송되는 경우에는, 패킷을 수신한 AD는 패킷의 소스 주소가 유효한지를 판단해야 한다. 만일 패킷이 AD B로부터 AD A로 전송되면, AD A는 다음과 같은 검사를 수행한다. 만일 A가 B를 신뢰할 수 있는 도메인이라고 판단하면, A는 수신한 패킷을 다음 AD로 전송한다. 그렇지 않으면, A는 패킷이 도착한 인터페이스와 A로부터 소스 호스트로의 "역 경로" 인터페이스가 동일한지를 판단하여 만일 동일한 인터페이스를 거치는 대칭 경로라고 판단되면 A는 수신한 패킷을 다음 도메인으로 전송한다. 그렇지 않고 만일 A가 비대칭 경로라고 판단하면, A는 A는 EID 검사에서 사용한 것과 동일한 방법을 이용하여 수신한 패킷을 폐기하고 AD:EID에 대한 검증 패킷 V를 소스 호스트로 전송한다. 만일 A가 소스 호스트로부터 유효한 응답을 수신하면, A는 소스 호스트의 AD:EID 주소와 패킷을 수신한 인터페이스 번호를 포함하는 정보를 자신의 캐쉬 테이블에 등록하고, 이 이후부터 AD:EID를 소스 주소로 하는 패킷이 테이블에 등록되어 있는 인터페이스를 통하여 들어오면 A는 수신한 패킷을 다음 AD로 전송한다.

AIP에서는 소스 호스트로부터 목적지 호스트로의 경로 상에 위치한 도메인으로의 경로와 해당 도메인에서 소스 호스트로의 역 경로가 다른 "비대칭" 경로인 경우에는 소스 주소의 진위를 검증하기 위하여 해당 도메인에서는 소스 호스트로부터 전송된 패킷을 폐기하고 소스 호스트에게 검증 패킷을 전송한다. 만일 소스 호스트로부터 목적지 호스트까지의 경로에 여러 개의 비대칭 경로가 존재하는 경우에는 비대칭 경로가 위치한 도메인들은 소스 호스트로부터 전송되는 패킷을 폐기하고 소스 주소를 검증하기 위한 검증 패킷을 전송하게 되며, 비대칭 경로의 개수만큼 소스 호스트는 동일한 패킷을 재전송함으로써, 특히 원격 검침과 같이 한 번씩의 패킷 교환으로 이루어진 트래픽의 경우에는 동일한 패킷의 반복적인 폐기와 재전송으로 인한 비효율성이 발생하게 된다. 또한 인터넷 토폴로지가 변화해서 소스 호스트와 목적지 호스트 간에 새로운 경로가 설정되는 경우에도 역시 패킷 폐기와 검증 패킷의 전송이 반복되게 되어 종단간 성능이 저하되게 된다.

이 논문에서는 비대칭 경로가 존재하는 경우에도 경유 도메인에서 패킷을 폐기하지 않고 소스 호스트에 설정된 주소를 검증할 수 있는 시그니처 기반의 향상된 AIP 기법을 제안한다.

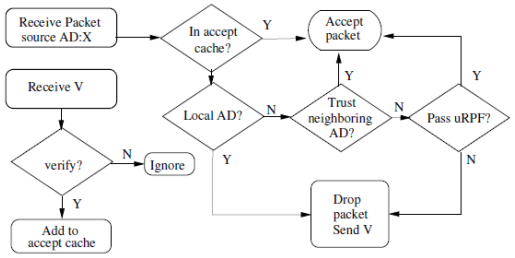

그림 1. AIP 프로토콜에서 소스 호스트 검증 과정
Fig. 1. Source host verification procedure in AIP protocol

Ⅲ. 시그니처 기반의 향상된 AIP 프로토콜

이 논문에서 호스트는 공개키/개인키 쌍을 가지고 있다고 가정한다. 소스 호스트가 목적지 호스트와 통신하고자 하는 경우에는, 소스 호스트가 전송하는 첫 번째 패킷 헤더에 그림 2에 나타나 있는 것과 같이 기존의 AIP 프로토콜에서 정의되어 있는 헤더 형식에 타임스탬프(Timestamp), 호스트의 공개키(Public key), 그리고 시그니처(Signature)가 포함된다. 타임스탬프는 소스 호스트가 패킷을 전송하는 시각을 나타내며 패킷의 식별자(identification)과 함께 뒤에서 언급할 재전송 공격을 방지하기 위하여 사용된다. 또한 소스 호스트로부터 전송되는 패킷의 소스 EID 주소 정보는 소스 호스트의 공개키의 해쉬 값으로 정해지므로, 공개키를 이용하면 소스의 EID 주소 정보의 진위를 검증할 수 있다. 만일 악의를 가진 공격자가 소스 호스트의 공개키를 얻게 된다면 소스 호스트의 공개키로부터 소스 호스트의 EID를 도출하는 것이 가능하게 되며, 이런 경우에는 공격자가 자신의 존재를 은닉하고 다른 호스트에게 공격을 가하기 위하여 다른 호스트의 공개키로부터 EID를 도출하여 자신의 주소 대신에 사용할 수도 있을 것이다. 시그니처는 소스 호스트에게 설정되어 있는 소스 EID 정보를 다른 악의에 찬 호스트가 위조하여 사용할 수 없도록 하기 위한 목적으로 사용된다. 소스 호스트가 전송하는 첫 번째 패킷에 포함되는 시그니처는 다음과 같은 정보를 이용하여 생성된다.

시그니처 = {소스 AD:EID 주소 ‖ 목적지 AD:EID 주소 ‖ Identification ‖ Timestamp ‖ 소스 공개키}$_{\text{소스 개인키}}$

<table>
<tr><td>Ver</td><td colspan="4">... standard IP Headers ...</td></tr>
<tr><td>...</td><td>random pkt id (32)</td><td>#dest (4)</td><td>next-dest (4)</td><td>#srcs (4)</td></tr>
<tr><td colspan="5">Source EID (160 bits)</td></tr>
<tr><td colspan="5">Source AD (top-level) (160 bits)</td></tr>
<tr><td colspan="5">Dest EID (160 bits)</td></tr>
<tr><td colspan="5">Dest AD (next hop) (160 bits)</td></tr>
<tr><td colspan="5">Dest AD stack (N*160 bits)</td></tr>
<tr><td colspan="5">Source AD stack (M*160 bits)</td></tr>
<tr><td colspan="2">Timestamp</td><td colspan="3">Public key</td></tr>
<tr><td colspan="5">public key</td></tr>
<tr><td colspan="5">signature</td></tr>
</table>

그림 2. AIP 패킷 헤더 구조.
Fig. 2. AIP packet header format.

그림 3은 도메인에서 패킷을 수신하는 경우에 패킷의 소스 주소를 검증하는 절차를 보여준다. 소스 호스트로부터 목적지 호스트까지의 경로 상에 위치한 도메인이 소스 호스트로부터 전송된 패킷을 수신하면, 도메인은 자신의 캐쉬 테이블에 소스 호스트에 대한 정보가 등록되어 있는 지를 확인한다. 만일 도메인의 캐쉬 테이블에 소스 호스트에 대한 정보가 없으면, 도메인은 패킷에 포함되어 있는 정보를 이용하여 시그니처를 검증한 후에 만일 시그니처가 유효하다고 판단되면 소스 호스트의 AIP 주소와 공개키 등의 정보를 자신의 캐쉬 테이블에 등록한다. 그런 후에 도메인은 수신한 패킷을 다음 도메인으로 전송한다. 이러한 방식으로 소스 호스트로부터 전송된 패킷이 목적지 호스트에 도착하게 되면 소스 호스트로부터 목적지 호스트까지 세션이 설정되게 된다. 패킷을 수신한 목적지 호스트는 소스 호스트에게 패킷을 수신하였다는 확인 패킷을 전송한다. 목적지 호스트로부터 전송된 확인 패킷을 수신한 소스 호스트는 기존의 AIP 패킷 헤더에 타임스탬프와 시그니처가 포함된 패킷을 전송한다. 두 번째 패킷 이후부터의 패킷에 포함되는 시그니처는 다음의 정보를 이용하여 생성된다.

시그니처 = {소스 AD:EID 주소 ‖ 목적지 AD:EID 주소 ‖ Identification ‖ Timestamp}$_{\text{소스 개인키}}$

만일 소스 호스트와 목적지 호스트 사이에 세션이 설정되어 있는 상태에서 인터넷 토폴로지의 변화로 인하여 경로가 변경되면, 소스 호스트와 목적지 호스트 사이에서 새로운 경로에 포함되는 도메인은 소스 호스트에 대한 캐쉬 테이블 엔트리를 가지고 있지 않기 때문에, 소스 호스트로부터 전송된 패킷 헤더에 포함되는 시그니처를 검증할 수 없게 된다. 이를 위하여 새로운 경로에 포함되는 도메인에 캐쉬 테이블에 등록되어 있지 않은 소스 호스트로부터 전송된 패킷을 수신하면, 도메인은 패킷에 포함되어 있는 시그니처와 AIP 주소를 검증하기 위한 검증 패킷을 소스 호스트에게 전송한다. 검증 패킷을 수신한 소스 호스트는 패킷 헤더에 자신의 공개키 정보를 포함한 후에 패킷을 다시 전송한다. 재전송된 패킷을 수신한 도메인은 자신의 캐쉬 테이블에 소스 AIP 주소와 소스 호스트에 대한 공개키 정보를 등록한 후에 주소를 검증한 후 패킷을 다음 도메인으로 전송하고, 새로운 경로에 포함되는 도메인에서는 동일한 방법으로 등록과 검증, 그리고 패킷 전송 과정을 반복한다.

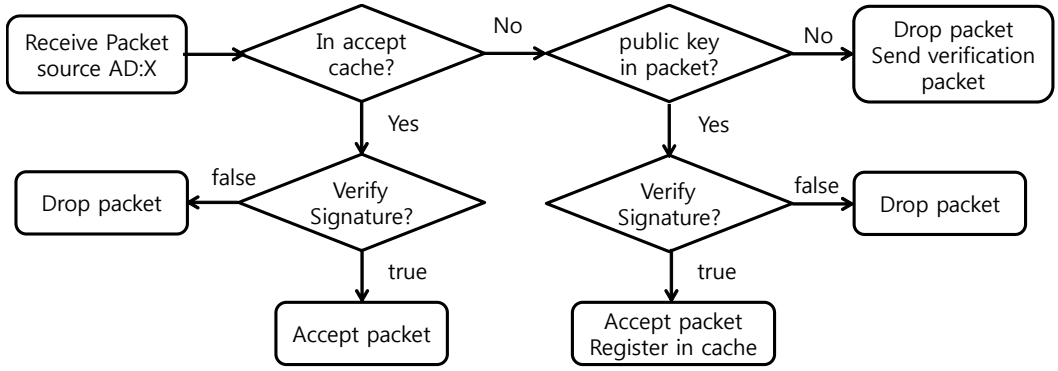

그림 3. 향상된 AIP 프로토콜에서 소스 호스트 검증 과정.
Fig. 3. Source host verification procedure in the improved AIP protocol.

Ⅳ. 제안 프로토콜의 안정성 평가

4.1 패킷 전송의 지연 감소

소스 호스트로부터 목적지 호스트 간에 패킷이 전송되는 경로 상에 위치한 도메인은 패킷 헤더에 있는 정보를 이용하여 신원을 검증할 수 있기 때문에 도메인이 패킷을 폐기하고 검증 패킷을 소스 호스트로 전송한 후에 다시 검증 확인 패킷을 수신하는 등의 과정이 없어서 패킷 전송 시의 지연을 줄일 수 있다. 이는 패킷 헤더에 포함되어 있는 정보를 이용하여 신원을 검증할 수 있기 때문이다.

4.2 주소에 대한 검증과 Spoofing 공격에 대한 안정성

본 방식은 패킷 내에 공개키를 포함 시켜 전송하며 AIP 주소는 해당 공개키의 해쉬 값을 이용하여 생성하고 있다. 기존의 AIP에서는 공개키가 포함되지 않았기 때문에 도메인에서 AIP 주소를 검증하기 위해서 검증 패킷을 전송하고, 소스 호스트는 자신의 공개키 정보가 포함되어 있는 검증 확인 패킷과 이 패킷의 위조를 방지하기 위한 시그니처를 도메인에게 전송하는 등의 공개키 획득 과정이 필요하다. 그렇지만 본 논문에서는 소스 호스트가 전송하는 첫 번째 패킷의 헤더에 소스의 공개키를 포함함으로써 도메인에서는 수신한 패킷에 포함되어 있는 공개키를 이용하여 소스 호스트의 EID를 검증할 수 있고 또한 시그니처를 이용하여 이 패킷이 유효한 소스 호스트로부터 전송되었다는 것을 검증할 수 있다. 공개키에 대응하는 개인키는 소스 호스트만이 알고 있는 정보이며, 개인키를 모르면 시그니처를 생성할 수 없다. 따라서 소스 호스트가 통신하는 도중 제 3자는 패킷 헤더에 포함되는 시그니처를 생성할 수 없기 때문에 제 3자가 소스 호스트의 IP주소를 이용하여 패킷을 전송한다는 것은 불가능 하다.

4.3 전송 공격에 대한 안정성

공격자가 비록 시그니처를 생성하지는 못하지만 소스 호스트로부터 전송된 패킷을 수신한 후에 동일한 패킷을 재전송함으로써 목적지 호스트 또는 도메인을 교란시킬 수 있다. 제안된 기법에서는 패킷의 식별자와 타임스탬프가 시그니처를 생성하는데 사용되며, 도메인에서는 현재 시간과 패킷에 포함되어 있는 타임스탬프 값을 비교하여 타임스탬프 값이 현재 시간의 허용 범위를 벗어나는 경우에는 패킷을 폐기한다. 이러한 방법은 도메인이 소스 호스트로부터 전송되는 첫 번째 패킷에 포함되어 있는 타임스탬프 값을 이용하여 현재 시간과 패킷이 생성된 시간의 차를 알 수 있기 때문에 가능하다. 따라서 도메인이 수신한 패킷의 타임스탬프 값을 검사하여 현재 시간의 허용 범위를 벗어나는 경우에는 패킷이 재전송되었다고 간주하기 때문에 공격자의 재전송 공격에 안전하다.

4.4 Session Hijacking 공격에 대한 안정성

공격자는 네트워크를 모니터링 후, 소스 호스트로부터 전송된 패킷의 헤더에 포함되어 있는 순서 번호를 관찰 한 후, 세션을 탈취하기 위해 RST(reset) 패킷을 전송할 수 있다. 하지만 이 경우에도 소스 호스트의 개인키를 모르기 때문에, 공격자는 시그니처를 생성 할 수 없어서 이러한 세션을 가로채기 위한 RST 패킷을 전송할 수 없다.

Ⅴ. 결 론

본 논문에서는 소스 호스트로부터 전송되는 첫 번

째 패킷에 타임스탬프, 소스 호스트의 공개키, 그리고 시그니처 정보를 포함하도록 해서, 소스 호스트와 목적지 호스트 사이의 경로에 위치한 도메인에서 패킷을 폐기하지 않고 소스 주소의 진위를 검증하면서도 재전송 공격에 대비할 수 있는 향상된 AIP 프로토콜을 제안하였다. 이러한 방법을 이용하면 도메인에서 패킷을 폐기하고 검증 패킷을 전송한 후에 소스 호스트로부터 전송되는 공개키 값을 이용하여 소스 주소를 검증하는 절차를 줄일 수 있어서 패킷 폐기로 인한 지연을 줄일 수 있다. 또한 인터넷 토폴로지의 변화로 인하여 종단간 경로가 변경되는 경우에도 소스 호스트로부터 전송되는 패킷을 처음 수신한 도메인만 주소 검증 절차를 구동하고 기존에 설정된 경로가 겹치는 부분에 위치한 도메인에서는 이미 자신의 캐쉬에 저장되어 있는 소스 호스트의 공개키 정보를 이용할 수 있기 때문에 주소 검증으로 인한 지연이 줄어들게 된다.

이 논문에서 제안된 기법에 대한 보안 안전성을 검증하였으며, 향후 연구에서는 이 논문에서 제안된 메커니즘을 이용하여 신뢰성 있는 미래 인터넷 구조를 연구하고자 한다.

References

[1] T. Aura, *Cryptographically generated addresses (CGA)*, RFC 3972, Mar. 2005.

[2] R. Moskowitz and P. Nikander, *Host identity protocol (HIP) architecture*, RFC 4432, May 2006.

[3] D. G. Andersen, H. Balakrishnam, N. Feamster, T. Koponen, D. Moon, and S. Shenker, “Holding the internet accountable,” in *Proc. Hotnets-VI*, Nov. 2007.

[4] D. G. Andersen, H. Balakrishnan, N. Feamster, T. Koponen, D. Moon, and S. Shenker, “Accountable internet protocol (AIP),” in *Proc. ACM SIGCOMM*, Aug. 2008.

[5] A. Anand, F. Dogar, D. Han, B. Li, H. Lim, M. Machado, W. Wu, A. Akella, D. Andersen, J. Byers, S. Seshan, and P. Steenkiste, “XIA: An architecture for an evolvable and trustworthy Internet,” Technical Report CMU-CS-11-100, Carnegie Mellon Univ., Jan. 2011.

[6] X. Zhang, H. C. Hsiao, G. Haskter, H. Chan, A. Perrig, and D. G. Andersen, “SCION: Scalability, control, and isolation on next-generation networks”, in *Proc. IEEE Sym. Security and Privacy,* Oakland, May 2011.

[7] H. C. Hsiao, T. H. J. Kim, S. Yoo, X. Zhang, S. B. Lee, V. Gligor, and A. Perrig, “STRIDE: sanctuary trail – refuge from internet DDoS entrapment,” *ASIACC’13*, May 2013.

[8] P. Ferguson and D. Senie, *Network ingress filtering: Defeating denial of service attacks with emply IP Source Address Spoofing*, RFC 4140, Jan. 1998.

박 기 태 (Gi-tae Park)

2013년 8월 : 동국대학교 정보통신공학과 졸업

2013년 9월~현재 : 동국대학교 정보통신공학과 석사과정

<관심분야> 정보보호, 네트워크보안, 사물인터넷

이 재 훈 (Jae-hwoon Lee)

1985년 2월 : 한양대학교 전자공학과 학사

1987년 2월 : 한국과학기술원 전기및전자공학과 석사

1995년 8월 : 한국과학기술원 전기및전자공학과 박사

1987년 3월~1990년 4월 : 데이콤 연구원

1990년 9월~1999년 2월 : 삼성전자 정보통신부문 선임연구원

1999년 3월~현재 : 동국대학교 정보통신공학과교수

<관심분야> IP 이동성, 다중 액세스 프로토콜, 인터넷 프로토콜, 초고속 통신, 네트워크 보안

정 희 영 (Hee-young Jung)

1991년~현재 : 한국전자통신연구원 근무, 현재 ID통신연구실장

2004년 2월 : 충남대학교 공학박사

2011년~2013년 : 미래인터넷포럼 Architecture WG 의장

<관심분야> 미래인터넷, 5G이동통신, 이동성 관리