(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2010년07월27일
(11) 등록번호 10-0972743
(24) 등록일자 2010년07월21일

(51) Int. Cl.
H04L 9/32 (2006.01) *H04L 12/24* (2006.01)
(21) 출원번호 10-2007-0102363
(22) 출원일자 2007년10월11일
심사청구일자 2007년10월11일
(65) 공개번호 10-2009-0037009
(43) 공개일자 2009년04월15일
(56) 선행기술조사문헌
Authentication Scheme of Mobile Router Using Temporary Certification of Access Router in MANEMO(2007.08.24)
KR1020060091935 A
KR1020050053857 A
전체 청구항 수 : 총 3 항

(73) 특허권자
숭실대학교산학협력단
서울 동작구 상도동 511
(72) 발명자
정수환
서울 강남구 대치동 316 은마 아파트 8동 305호
노효선
전남 순천시 남정동 223-3
(74) 대리인
이풍우

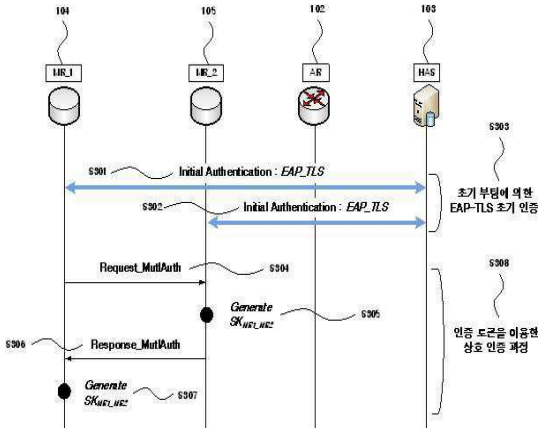
심사관 : 성인구

(54) 마니모의 이동 애드 혹 네트워크에 속한 이동 라우터 간에인증 토큰을 이용한 상호 인증 방법

(57) 요 약

본 발명은 NEMO(Network Mobility)의 중첩된 이동 네트워크 환경에서 경로 최적화를 제공하기 위해 개발된 MANEMO(MANET & NEMO)에서 중첩된 이동 네트워크에 존재하는 이동 라우터들 간에 액세스 라우터에서 발급해주는 인증 토큰을 사용하여 상호 인증을 제공하는 방법에 관한 것이다.

대 표 도 - 도3



특허청구의 범위

청구항 1

삭제

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

마니모(MANEMO)의 이동 애드 혹 네트워크에 속한 이동 라우터 간에 인증 토큰을 이용한 상호 인증 방법에 있어서,

최초 이동 라우터가 부팅 되는 과정에서 무선인증서방식(EAP-TLS)을 통한 전체 인증과정을 통해 홈 인증 서버와 이동 라우터 간의 세션키 및 이동 라우터와 엑세스 라우터 간의 세션키를 분배하되, 이동 라우터(101)가 인증 서버(103)에게 초기 인증을 수행하고, 이동 라우터(101)과 인증 서버(103)간에 마스터키 및 세션키를 상호 공유하는 제1 단계;

이동 애드 혹 네트워크 도메인에 속한 상기 이동 라우터(101)가 상기 엑세스 라우터(102)로부터 인증 토큰을 발행받되, 이동 라우터(101)는 직접적인 통신을 위한 인증 토큰을 액세스 라우터(102)에 요청하기 위해 다음 식

$$Request\text{-}AuthT:[ID_{MR},nonce,Y_{MR}]E_{SK_{MR\text{-}AR}},HMAC$$

과 같은 메시지를 생성하여 전송하는 제205 단계(S205);

상기 액세스 라우터(102)는 상기 메시지를 수신하고 다음 식과 같은 메시지

$$AuthenticationToken:[ID_{MR},nonce,tp,Y_{MR}]Sign_{KR_{AR}}$$

로 인증 토큰을 생성하는 제206 단계(S206);

상기 액세스 라우터(102)는 생성된 인증 토큰이 상기 제1 단계에서 생성된 세션키로 암호화된 다음 식과 같은

$$Response\text{-}AuthT:[AuthToken_{MR},KU_{AR}]E_{SK_{MR\text{-}AR}}$$

메시지를 이동 라우터(101)에 전송하는 제207 단계(S207)를 포함하는 제2 단계;

상기 발행받은 인증 토큰을 이용하여 이동 애드 혹 네트워크 도메인에 속한 이동 라우터(104,105) 간의 상호 인증을 수행하는 제3 단계를 포함하고, 여기서, Request-AuthT는 EAP-TLS 초기 인증 과정이 끝난 이후에 수행되며 이동 라우터(101)가 직접적인 통신을 위한 인증 토큰을 액세스 라우터(102)에 요청하기 위한 메시지로써, Request-AuthT 메시지의 ID_{MR}은 이동 라우터(101) MR의 IP 주소이고, nonce는 이동 라우터(101) MR에서 수도 랜덤 함수를 이용하여 생성하는 임의의 수이고, Y_{MR}은 이동 라우터(101) MR의 디피 헬만

공개값이고, $E_{SK_{MR\text{-}AR}}$은 ID_{MR}, nonce, Y_{MR}를 세션키 $SK_{MR\text{-}AR}$로 암호화한다는 것이고, HMAC는 암호화 해쉬 기능을 이용한 메시지 인증 코드이고, AuthenticationToken은 액세스 라우터(102)가 이동 라우터(101)의 요청 메시지 Request-AuthT를 수신한 후 인증 토큰을 생성하는 메시지로써, AuthenticationToken 메시지의 ID_{MR}, nonce, Y_{MR}은 Request-AuthT 메시지의 값과 같고, tp는 시간 정보로써 액세스 라우터(102)가 인증 토큰을 생성한 시간이고, $Sign_{KR_{AR}}$은 ID_{MR}, nonce, Y_{MR}, tp를 액세스 라우터(102)의 개인키 KR_{AR}로 서명한다는 것이고, Response-AuthT는 상기 206 단계에서 생성된 인증 토큰을 상기 1단계에서 생성된 세션키 $SK_{MR\text{-}AR}$로 암호화한 메시지로써, $AuthToken_{MR}$은 이동 라우터(101)가 초기 인증 이후 액세스 라우터(102)의 개인키로 서명하여 발급한 이동 라우터(101)를 위한 인증 토큰이고, KU_{AR}은 액세스 라우터(102)의 공개키이고, $E_{SK_{MR\text{-}AR}}$은 $AuthToken_{MR}$, KU_{AR}를 세션키 $SK_{MR\text{-}AR}$로 암호화한다는 것을 특징으로 하는 마니모의 이동 애드 혹 네트워크에 속한 이동 라우터 간에 인증 토큰을 이용한 상호 인증 방법.

청구항 6

제 5항에 있어서,

상기 제3 단계는 이동 라우터 MR1(104)이 이동 라우터 MR2(105)와 안전한 통신을 하기위해 다음 식과 같은 메시지 $Request\text{-}MutlAuth{:}AuthToken_{MR1}$를 생성하여 이동 라우터 MR2(105)에게 전송하는 제304 단계(S304);

상기 이동 라우터 MR2(105)는 액세스 라우터(102)의 공개키 KU_{AR} 을 이용하여 수신된 인증 토큰을 검증하고, 디피헬만(Diffie-Hellman)값을 이용하여 세션키 Generate SK_{MR1_MR2}를 생성하는 제305 단계(S305);

상기 이동 라우터 MR2(105)는 상기 생성된 세션키 Generate SK_{MR1_MR2}를 이용하여 자신의 인증 토큰을 다음 식과 같은 메시지 $Response\text{-}MutlAuth{:}AuthToken_{MR2},HMAC'$를 생성하고 이동 라우터 MR1(104)에게 전송하는 제306 단계(S306);

상기 제306 단계(S306)의 메시지를 수신한 이동 라우터 MR1(104)는 상기 인증 토큰을 액세스 라우터(102)의 공개 키 KU_{AR} 을 이용하여 검증하고, 이동 라우터 MR2(105)와 같은 세션키 Generate SK_{MR1_MR2}를 생성하고, 전달받은 메시지의 HMAC'값과 동일한 값을 생성하여 무결성을 검증하는 제307 단계(S307)를 포함하되, 여기서, Request-MutlAuth는 하나의 액세스 라우터(102)가 관할하는 애드 혹 도메인에 속한 이동 라우터(104,105) 간에 상호 비밀 통신을 원할 경우 통신을 원하는 이동 라우터(104,105)들 간에 상호 인증을 위해 주고 받는 메시지로써, $AuthToken_{MR1}$은 MR1(104)이 초기 인증 이후 액세스 라우터(102)의 개인키로 서명하여 발급한 MR1(104)을 위한 인증 토큰이고, Response-MutlAuth는 MR1이 상호 인증을 요청한 MR2(105)가 상기 제306 단계의 메시지를 수신한 다음 인증 토큰을 발행한 액세스 라우터(102)의 공개키를 이용하여 MR1(104)의 인증 토큰을 검증한 다음에 전송하는 메시지로써, $AuthToken_{MR2}$는 MR2(105)가 MR1(104)의 인증 토큰에 대한 검

증이 성공할 경우 이동 라우터(102)에게 발급받은 자신의 인증 토큰이고, HMAC'는 MR1(104)이 전송한 인증 토큰에 포함되어 있는 MR1(104)의 디피헬만 공개값 Y_{MR1} 을 디피헬만 키 생성방법을 통해 생성한 공유 세션키 $SK_{MR1-MR2}$ 를 해쉬하여 생성한 값인 것을 특징으로 하는 마니모의 이동 애드 혹 네트워크에 속한 이동 라우터 간에 인증 토큰을 이용한 상호 인증 방법.

청구항 7

제 6항에 있어서,

상기 제306 단계(S306)에서 생성된 메시지의 HMAC'는 다음의 식 $HMAC'=h[AuthToken_{MR1}, AuthToken_{MR2}, Generate\,SK_{MR1-MR2}]$ 과 같이 생성되되, 여기서, h는 $AuthToken_{MR1}$, $AuthToken_{MR2}$, $GenerateSK_{MR1-MR2}$ 를 해쉬한다는 것이고, $AuthToken_{MR1}$ 은 MR1(104)의 인증 토큰이고, $AuthToken_{MR2}$ 는 MR2(105)의 인증 토큰이고, $GenerateSK_{MR1-MR2}$ 는 MR2(105)가 MR1(104)에 대한 인증을 성공한 다음 생성한 공유 비밀키인 것을 특징으로 하는 마니모의 이동 애드 혹 네트워크에 속한 이동 라우터 간에 인증 토큰을 이용한 상호 인증 방법.

명 세 서

발명의 상세한 설명

기 술 분 야

[0001] 본 발명은 MANEMO(MANET & NEMO, 이하 “MANEMO”라고 함.) 환경의 이동 애드 혹 네트워크 도메인에서 인증 토큰을 이용한 이동 라우터 간의 상호 인증 기법에 대한 발명으로 이동 네트워크 환경에서의 이동 라우터 인증과 이동 애드 혹 네트워크 환경에서 이동 단말에 대한 인증 기술에 속한다.

배 경 기 술

[0002] 본 발명은 액세스 라우터가 발행하는 인증 토큰을 이용하여 MANEMO의 이동 애드 혹 네트워크에 속해 있는 이동 라우터 간에 직접적인 통신 요청 시 상호 인증을 신속하게 제공하기 위한 방법에 관한 것이다.

[0003] NEMO(Network Mobility, 이하 “NEMO”라고 함.)는 네트워크의 이동성을 지원하며 이동 라우터를 통해 이동 라우터의 서브넷에 존재하는 다양한 이동 단말과 고정된 단말의 지속적인 인터넷 연결을 제공해 준다. 또한, NEMO는 이동 라우터의 서브넷에 PAN(Personal Area Network) 또는 다른 이동 네트워크가 중첩되어 존재할 수 있으며 이러한 중첩된 이동 네트워크의 지속적인 인터넷 접속 또한 중첩된 이동 네트워크의 상위 이동 라우터를 통해 지원한다. 이러한 NEMO의 중첩된 네트워크 환경에서보다 효과적으로 경로 설정 지원을 하기 위해 IETF NEMO 작업그룹에서 NEMO Basic Support 프로토콜을 표준화하였다. 그러나 NEMO Basic Support 프로토콜의 경우 중첩된 이동 네트워크의 레벨이 깊어질수록 데이터를 전달하기 위한 이동 라우터 간 경로 설정 과정에서 핀볼 라우팅 문제와 같은 경로 최적화 문제가 발생한다. NEMO의 중첩된 이동 네트워크에서의 핀볼 라우팅 문제와 같은 경로 최적화를 해결하기 위해 MANEMO가 제안되었으며 MANEMO는 중첩된 이동 네트워크에 존재하는 이동 라우터를 이동 애드 혹 네트워크 환경으로 구성하여 중첩된 이동 네트워크 내에 존재하는 이동 라우터 간에 이동 애드 혹 네트워크 라우팅 프로토콜을 이용하여 이동 라우터 간에 직접적인 경로를 설정할 수 있도록 하였다.

[0004] NEMO의 중첩된 이동 네트워크에서의 경로 최적화를 위해 개발된 MANEMO의 경우 NEMO에서처럼 안전한 통신을 위해 이동 라우터를 인증하는 것은 매우 중요하다. 또한 MANEMO의 경우 이동 라우터들이 자유롭게 이동 애드 혹

네트워크 도메인으로 이동하여 포함될 수도 있으며 이동 애드 혹 네트워크 도메인을 벗어나 다른 곳으로 이동할 수 있다. 때문에 악의적인 목적을 가진 이동 라우터가 존재할 수 있으며 정상적인 이동 라우터들 간의 정상적인 경로 설정을 방해하거나 정상적인 이동 라우터로 위장하여 인터넷 연결을 시도할 수 있는 위협이 언제나 존재한다. 이러한 위협을 예방하기 위해서 기존 NEMO환경에서 이동 라우터와 이동 노드들을 인증하기 위해 제안된 AAA(Authentication, Authorization and Accounting, 이하 "AAA"라고 함.) 서버를 이용한 인증 기술 또는 PKI(Public Key Infrastructure, 이하 "PKI"라고 함.)의 인증서 기반의 인증 기술 등 몇 가지 인증 기술들을 적용해 볼 수 있다. 그러나 이러한 인증 기술들의 경우 다음과 같은 몇 가지 문제점을 가지고 있다.

[0005] AAA 서버 모델의 인증 기술의 경우 이동 라우터 또는 이동 라우터의 서브넷에 존재하는 이동 노드 그리고 고정된 노드를 인증할 때 홈 도메인에 존재하는 홈 인증 서버를 통해 인증을 받는다. 때문에 중첩된 이동 네트워크의 레벨이 증가할 경우 이동 라우터의 하위 서브넷에 존재하는 이동 라우터가 인증을 받기 위해서는 중첩된 이동 네트워크의 상위 이동 라우터들의 홈 도메인에 있는 홈 인증 서버들에게 인증 메시지가 전달되어야 한다. 이럴 경우 이동 라우터를 인증하기 위한 전체 인증 시간이 증가하게 되고, 인증 패킷이 전달되는 과정에서 발생하는 터널들로 인한 패킷 오버헤드가 증가하는 문제가 생기며 이것은 중첩된 이동 네트워크 레벨이 증가할 경우 더 크게 증가한다.

[0006] 또한, PKI의 인증서 기반의 인증 모델의 경우에서 위에서 언급한 문제가 존재한다. MANEMO의 이동 애드 혹 네트워크 도메인에 속한 이동 라우터들이 각기 다른 인증기관으로부터 발급받은 인증서를 사용할 경우 서로를 인증하기 위해 인증서를 확인할 수 있는 상위 인증기관의 공개키 정보를 알아야 한다. 이러한 문제를 해결하기 위해 PKI에서는 디렉토리 서버를 사용하며, 사전에 디렉토리 서버에 인증서와 공개키 정보를 보관한다. 보관된 정보들은 상호 인증을 위해 인증서의 공개키 정보가 필요한 이동 라우터가 디렉토리 서버로 요청한 후 필요한 공개키 정보 또는 인증서를 전달받은 다음 다른 인증기관으로부터 발급된 상대 이동 라우터의 인증서를 검증하게 된다. 이런 과정을 매번 다른 인증기관으로부터 인증서를 발급받은 이동 라우터를 만나게 될 때마다 수행해야함으로 전체적인 인증 시간과 인증을 위한 시그널링 메시지의 수가 증가하게 된다.

발명의 내용

해결 하고자하는 과제

[0007] 본 발명은 상기와 같은 문제점을 해결하기 위해 안출된 것으로서, 본 발명의 목적은 MANEMO 환경의 이동 애드혹 네트워크 도메인에 속한 이동 라우터들 간에 직접적인 통신을 원할 경우, 직접적인 통신을 원하는 이동 라우터들 간의 상호 인증을 신속하게 할 수 있도록 액세스 라우터가 발행하는 인증 토큰을 사용하는 방법을 제공함에 있다.

과제 해결수단

[0008] MANEMO 환경의 이동 애드 혹 네트워크 환경에서 이동 라우터 간 상호 인증시 발생하는 패킷 오버헤드와 전체 인증 메시지 수를 줄이기 위해 액세스 라우터에서 발행하는 인증 토큰을 사용하여 상호 인증을 신속하게 할 수 있게 한다.

효 과

[0009] 이상 설명한 바와 같이, 본 발명에 따르면 MANEMO 환경에 존재하는 이동 애드 혹 네트워크 도메인에 속한 이동 라우터 간 상호 인증 시 기존 AAA 서버 모델이나 PKI 인증서 기반의 인증 모델에 비해서 전체 인증 메시지 수를 줄일 수 있다. 이는 초기 인증과정이 끝난 이후 액세스 라우터로부터 발급받은 인증 토큰을 통해 상호 인증을 수행하기 때문에 AAA 서버 모델처럼 이동 라우터를 인증하기 위해 홈 인증 서버까지 인증 시그널링 메시지를 보내지 않아도 되며, PKI 인증서 기반 인증 기법에서처럼 다른 인증기관의 공개키 또는 인증서를 획득하기 위해 디렉토리 서버와의 시그널링 메시지를 주고받지 않아도 된다. 이렇게 인증 메시지가 줄어들게 되면 이동 라우터 간에 상호 인증하는 과정에서 전체 상호 인증 수행 시간이 감소하게 되며, 또한 인증 메시지에 대한 패킷 오버헤드가 AAA 서버 모델에서처럼 발생하지 않는다. 그리고 본 발명의 경우 PKI 인증서 기반의 인증 기법에서처럼 공인 인증서를 사용하지 않기 때문에 인증서에 포함된 인증서 체인에 의한 인증서 검증 시 발생하는 이동 라우터의 계산 부담의 감소를 가능하게 한다.

발명의 실시를 위한 구체적인 내용

[0010] 상기 목적을 달성하기 위한 본 발명에 따른 MANEMO 환경에서 액세스 라우터가 발행하는 인증 토큰을 이용한 이동 라우터 간 상호 인증 방법은, 최초 이동 라우터가 부팅 되는 과정에서 EAP-TLS(Extensible Authentication Protocol Transport Layer Security)를 통한 전체 인증 과정을 통해 홈 인증 서버와 이동 라우터 간의 세션키 및 이동 라우터와 액세스 라우터 간의 세션키를 분배하는 제1 단계, 이동 애드 혹 네트워크 도메인에 속한 이동 라우터가 액세스 라우터로부터 인증 토큰을 발행받는 제2 단계, 발행받은 인증 토큰을 이용하여 이동 애드 혹 네트워크 도메인에 속한 이동 라우터 간의 상호 인증 수행 과정과 비밀 통신을 위한 비밀키 공유 하는 제3 단계를 포함한다.

[0011] 이하에서는 도면을 참조하여 본 발명을 보다 상세하게 설명한다.

[0012] 도 1은 본 발명에서 이동 라우터(101)가 초기 부팅 되는 과정에서 EAP-TLS 전체 인증 과정(S101 ~ S111)을 통해 홈 인증 서버(103)로부터 인증을 받고 이후 액세스 라우터(102)로부터 발행되는 인증 토큰을 암호화하기 위해 사용될 세션키 생성에 사용된 EAP-TLS 전체 인증 과정을 설명하기 위한 도면이다. 도면 1을 참조하면 무선 네트워크 환경에서 이동 라우터(101)와 홈 인증 서버(103)인 EAP-TLS 서버 간의 TLS 핸드쉐이크를 EAP 프로토콜을 통해 확장한 방법으로서 상호 인증과 키 분배에 대한 메커니즘을 포함하고 있다. 이동 라우터(101)가 최초 이동 애드혹 네트워크 도메인에서 초기 부팅 되면 액세스 라우터(102)가 이를 인식하고 EAP-TLS 메시지를 통해 핸드쉐이크를 시작한다. EAP-TLS 핸드쉐이크 과정을 통해 이동 라우터(101)와 홈 인증 서버(103)는 마스터키를 생성(S107, S108)한 후 보관하게 되고 이후 이동 라우터(101)와 액세스 라우터(102) 간의 안전한 통신을 위한 세션키를 홈 인증 서버(103)가 생성하여 액세스 라우터(102)에게 전달해 준다. 이때 액세스 라우터(102)에게 전달된 키는 이동 라우터(101)에서도 마스터키를 이용하여 동일하게 생성할 수 있다.

[0013] 도 2는 본 발명에 따른 이동 라우터(101)가 액세스 라우터(102)로부터 인증 토큰을 발행받는 과정을 설명하기 위한 도면이다. 도 2를 참조하면 도 1에서 설명한 EAP-TLS 전체 인증 과정(S102)을 통해 이동 라우터(101)와 액세스 라우터(102) 간에 안전한 인증 토큰 전달을 위한 세션키 SK_{MR-AR}(S203)를 공유하게 된다. 이동 라우터(101)가 초기 인증(S201)을 끝낸 후 이동 애드 혹 네트워크 도메인에서 상호 인증 시 사용하게 되는 인증 토큰을 액세스 라우터(102)에게 발급받기 위해 인증서 발급 요청 메시지(S205)를 다음의 식과 같은 형식으로 메시지를 생성하여 액세스 라우터(102)로 전달한다.

[0014] $$Request\text{-}AuthT:[ID_{MR}, nonce, Y_{MR}]E_{SK_{MR\text{-}AR}}, HMAC$$

여기서, Request-AuthT는 이동 라우터(101)가 직접적인 통신을 위한 인증 토큰을 액세스 라우터(102)에 요청하기 위한 메시지로써, Request-AuthT 메시지의 ID_{MR}은 이동 라우터 MR의 IP 주소이고, nonce는 이동 라우터 MR에서 수도(pseudo) 랜덤 함수를 이용하여 생성하는 임의의 수이고, Y_{MR}은 이동 라우터 MR의 디피헬만(Diffie-Hellman) 공개 값이고, $E_{SK_{MR\text{-}AR}}$은 ID_{MR}, nonce, Y_{MR}를 세션키 $SK_{MR\text{-}AR}$로 암호화한다는 것이고, HMAC는 암호화 해쉬 기능을 이용한 메시지 인증 코드이다.

[0015] 위의 식에서처럼 이동 라우터(101)는 자신의 ID 정보와 임의의 수 그리고 자신의 디피헬만(Diffie-Hellman) 공개 값 Y_{MR}을 이동 라우터(101)와 액세스 라우터(102) 간에 공유하고 있는 세션키 SK_{MR-AR}(S203)로 암호화한 메시지와 ID_{MR} 과 nonce 그리고 Y_{MR}을 세션키 SK_{MR-AR}(S203)로 해쉬한 HMAC 값을 함께 액세스 라우터(102)에게 보낸다. 이때 전달되는 메시지에 포함된 디피헬만(Diffie-Hellman) 공개 값은 인증 토큰을 이용하여 상호 인증을 수행한 후 인증된 두 이동 라우터 간의 비밀 통신을 위해 사용될 비밀키를 생성하기 위해 사용된다. 이러한 인증 토큰

요청 메시지(S205)를 액세스 라우터(102)가 수신하면 메시지를 복호화하고 HMAC을 통해 메시지를 검증한 후 이동 라우터에게 전달할 인증 토큰을 생성(S206)한다. 인증 토큰은 액세스 라우터(102)가 RSA 알고리즘을 이용하여 생성한 개인키로 이동 라우터가 보내준 ID_{MR}, nonce, Y_{MR} 그리고 타임 스템프 값을 서명하여 다음의 식과 같이 생성한다.

$$AuthenticationToken:[ID_{MR},nonce,tp,Y_{MR}]Sign_{KR_{AR}}$$

여기서, AuthenticationToken은 액세스 라우터(102)가 이동 라우터(101)의 요청 메시지 Request-AuthT를 수신한 후 인증 토큰을 생성하는 메시지로써, AuthenticationToken 메시지의 ID_{MR}, nonce, Y_{MR}은 Request-AuthT 메시지의 값과 같고, tp는 시간 정보로써 액세스 라우터(102)가 인증 토큰을 생성한 시간이고, $Sign_{KR_{AR}}$은 ID_{MR}, nonce, Y_{MR}, tp를 액세스 라우터(102)의 개인키 KR_{AR}로 서명한다는 것이다.

[0016] 삭제

[0017] 위의 식과 같이 생성된 인증 토큰(S206)은 Responce_AuthT 메시지(S207)를 통해 액세스 라우터의 공개키 KU_{AR}과 함께 이동 라우터에게 다음의 식과 같이 전달된다.

[0018]
$$Response\text{-}AuthT:[AuthToken_{MR},KU_{AR}]E_{SK_{MR\text{-}AR}}$$

여기서, Response-AuthT는 제206 단계에서 생성된 인증 토큰을 제1 단계에서 생성된 세션키 $SK_{MR\text{-}AR}$로 암호화한 메시지로써, $AuthToken_{MR}$은 이동 라우터(101)가 초기 인증 이후 액세스 라우터(102)의 개인키로 서명하여 발급한 이동 라우터(101)를 위한 인증 토큰이고, KU_{AR}은 액세스 라우터(102)의 공개키이고, $E_{SK_{MR\text{-}AR}}$은 $AuthToken_{MR}$, KU_{AR}를 세션키 $SK_{MR\text{-}AR}$로 암호화한다는 것이다.

[0019] 위와 같은 과정을 통해 이동 애드 혹 네트워크 도메인에 속한 이동 라우터들은 액세스 라우터(102)로부터 인증 토큰을 발행받아 보관하고 있게 된다. 이렇게 함으로서 PKI의 인증서 기반 인증 방법에서 사용하는 인증서를 사용하지 않기 때문에 디렉토리 서버로부터 인증서의 공개키나 인증서를 요청하게 될 때 발생하는 전체 인증 메시지의 증가 또는 인증서에 포함되어 있는 인증서 체인에 따른 인증서 검증 시 발생하게 되는 이동 라우터의 인증서 계산 부담 등을 줄이게 된다.

[0020] 도 3은 액세스 라우터(102)로부터 발급받은 인증 토큰을 이용하여 통신을 원하는 이동 애드 혹 네트워크 도메인

에 포함된 이동 라우터 간에 상호 인증을 수행하는 과정과 이를 통해 안전한 데이터 전송을 위한 비밀키를 공유하는 과정을 설명하기 위한 도면이다. 초기 부팅 과정을 통해 EAP-TLS 전체 인증과정(S301, S302)을 끝내고 액세스 라우터(102)로부터 인증 토큰을 발급받은 이동 애드 혹 네트워크 도메인에 속한 이동 라우터 MR_1(104) 과 MR_2(105)가 직접적인 통신을 원할 경우 우선 상호 인증하는 과정(S304~S307)을 수행한다. 상호 인증 수행을 위해 MR_1(104)은 자신의 인증 토큰을 Request_MutlAuth 메시지 (S304)에 포함하여 다음과 같이 MR_2(105)로 보낸다.

[0021] $$Request\text{-}MutlAuth : AuthToken_{MR1}$$

여기서, Request-MutlAuth는 하나의 액세스 라우터(102)가 관할하는 애드 혹 도메인에 속한 이동 라우터(104,105)들 간에 상호 비밀 통신을 원할 경우 통신을 원하는 이동 라우터(104,105)들 간에 상호 인증을 위해 주고 받는 메시지로써, $AuthToken_{MR1}$ 은 MR1이 초기 인증 이후 액세스 라우터의 개인키로 서명하여 발급한 MR1을 위한 인증 토큰이다.

[0022] 위와 같은 메시지를 전달받은 MR_2(105)는 액세스 라우터(102)로부터 인증 토큰을 발행 받을 때 함께 전달받았던 액세스 라우터의 공개키 KU_{AR}을 이용하여 MR_1(104)의 인증토큰을 검증하게 된다. 이때 인증 토큰에 포함된 MR_1(104)의 디피헬만(Diffie-Hellman) 공개 값인 Y_{MR1}과 자신의 디피헬만(Diffie-Hellman) 비밀 값을 이용하여 비밀키 Generate SK_{MR1_MR2}(S305)를 생성한다. 이렇게 생성된 비밀키(S305)를 이용하여 MR_2(105)는 MR_1(104)에게 자신의 인증 토큰을 다음과 같은 Response_MutlAuth 메시지(S306)를 통해 전송한다.

[0023] $$Response\text{-}MutlAuth : AuthToken_{MR2}, HMAC'$$

여기서, Response-MutlAuth는 MR1이 상호 인증을 요청한 MR2가 제306 단계의 메시지를 수신한 다음 인증 토큰을 발행한 액세스 라우터의 공개키를 이용하여 MR1의 인증 토큰을 검증한 다음에 전송하는 메시지로써, $AuthToken_{MR2}$는 MR2가 MR1의 인증 토큰에 대한 검증이 성공할 경우 이동 라우터에게 발급받은 자신의 인증 토큰이고, HMAC'는 MR1이 전송한 인증 토큰에 포함되어 있는 MR1의 디피헬만 공개값 Y_{MR1}을 디피헬만 키 생성방법을 통해 생성한 공유 세션키 $SK_{MR1\text{-}MR2}$를 해쉬하여 생성한 값이다.

[0024] 상기 메시지를 수신한 이동 라우터 MR_1(104)는 인증 토큰을 액세스 라우터(102)의 공개 키 KU_{AR}을 이용하여 검증하고, 서명에 포함된 디피헬만(Diffie-Hellman) 키 분배 장식에서 정의하고 있는 방법으로 이동 라우터 MR_2(105)가 생성한 것과 동일한 비밀키 Generate SK_{MR1_MR2}를 생성하고, 상기 전달 받은 메시지의 HMAC'값과 동일한 값을 생성함으로써 메시지의 무결성을 검증하여, 이동 라우터들 간에 인증 토큰을 사용하여 상호 인증 및 비밀키를 공유(S307)할 수 있다. 이때 Response_MutlAuth 메시지(S306)에 포함되어 전달되는 HMAC'는 다음과 같이 생성된다.

[0025] $$HMAC' = h[AuthToken_{MR1}, AuthToken_{MR2}, Generate\ SK_{MR1\text{-}MR2}]$$

여기서, h는 $AuthToken_{MR1}$, $AuthToken_{MR2}$, $GenerateSK_{MR1\text{-}MR2}$를 해쉬 한다는 것이고, $AuthToken_{MR1}$은 MR1(104)의 인증 토큰이고, $AuthToken_{MR2}$는 MR2(105)의 인증 토큰이고, $GenerateSK_{MR1\text{-}MR2}$는 MR2(105)가 MR1(104)에 대한 인증을 성공한 다음 생성한 공유 비밀키이다.

[0026] 위에서처럼 액세스 라우터(102)에서 발행한 인증 토큰을 통해 이동 애드 혹 네트워크 도메인에 속한 이동 라우터 간에 상호 인증을 수행하게 되면 기존 AAA 서버 모델에서처럼 이동 라우터를 인증하기 위해 홈 인증 서버에게 인증을 받는 전체 인증 과정을 수행하지 않게 되며 이로 인해 발생했던 전체 인증 시간의 지연을 줄일 수 있다. 또한 AAA 서버 모델에서 홈 인증 서버와의 터널로 인해 발생 했던 패킷 오버헤드 문제가 발생하지 않는다.

도면의 간단한 설명

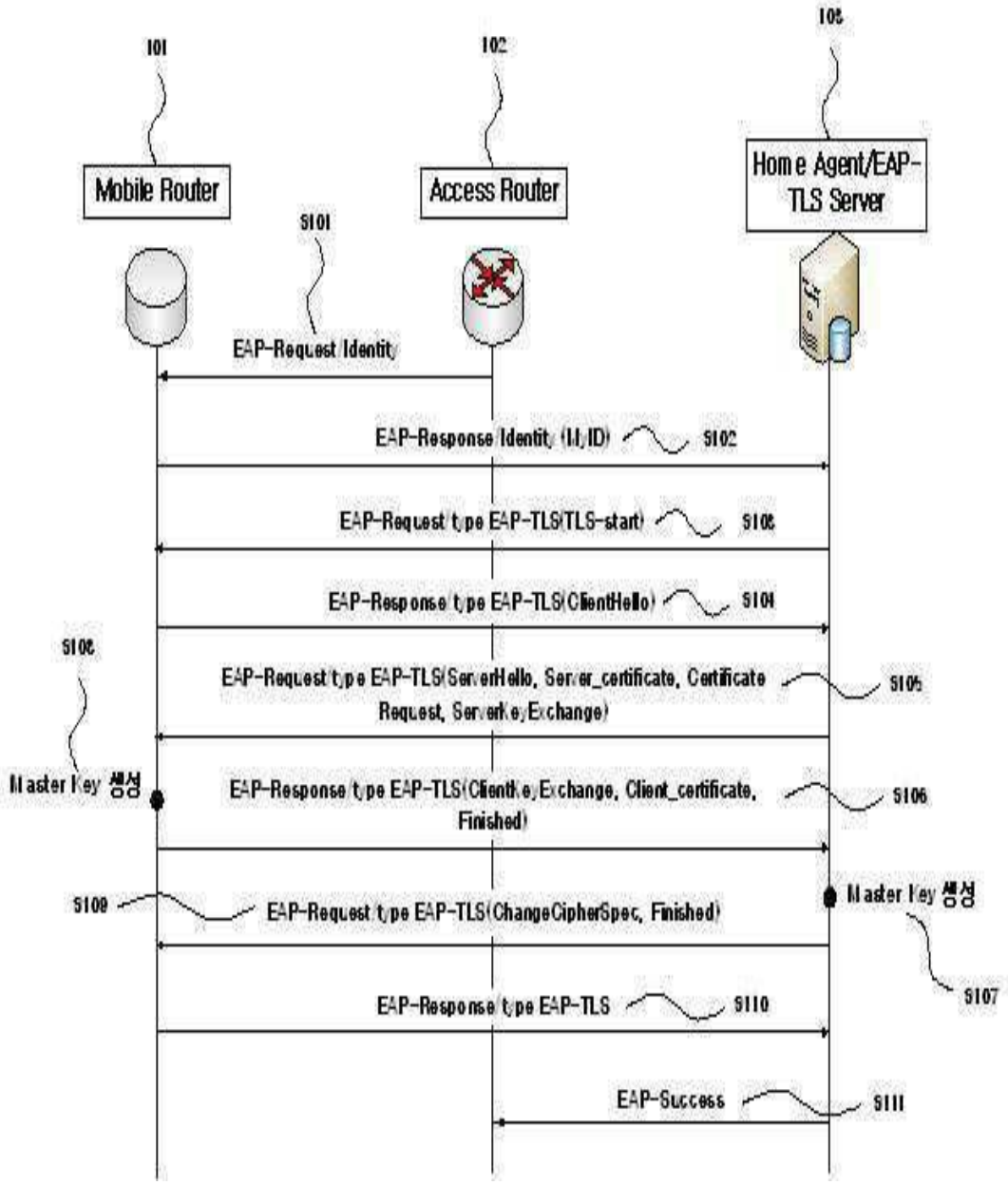
[0027] 도 1은 본 발명에 따른 이동 라우터가 초기 부팅 되었을 때 EAP-TLS를 통해 초기 인증 수행과정을 설명하기 위한 도면,

[0028] 도 2는 본 발명에 따른 이동 애드 혹 네트워크 도메인에 속한 이동 라우터가 액세스 라우터로부터 인증 토큰을 발행받는 과정을 설명하기 위한 도면,

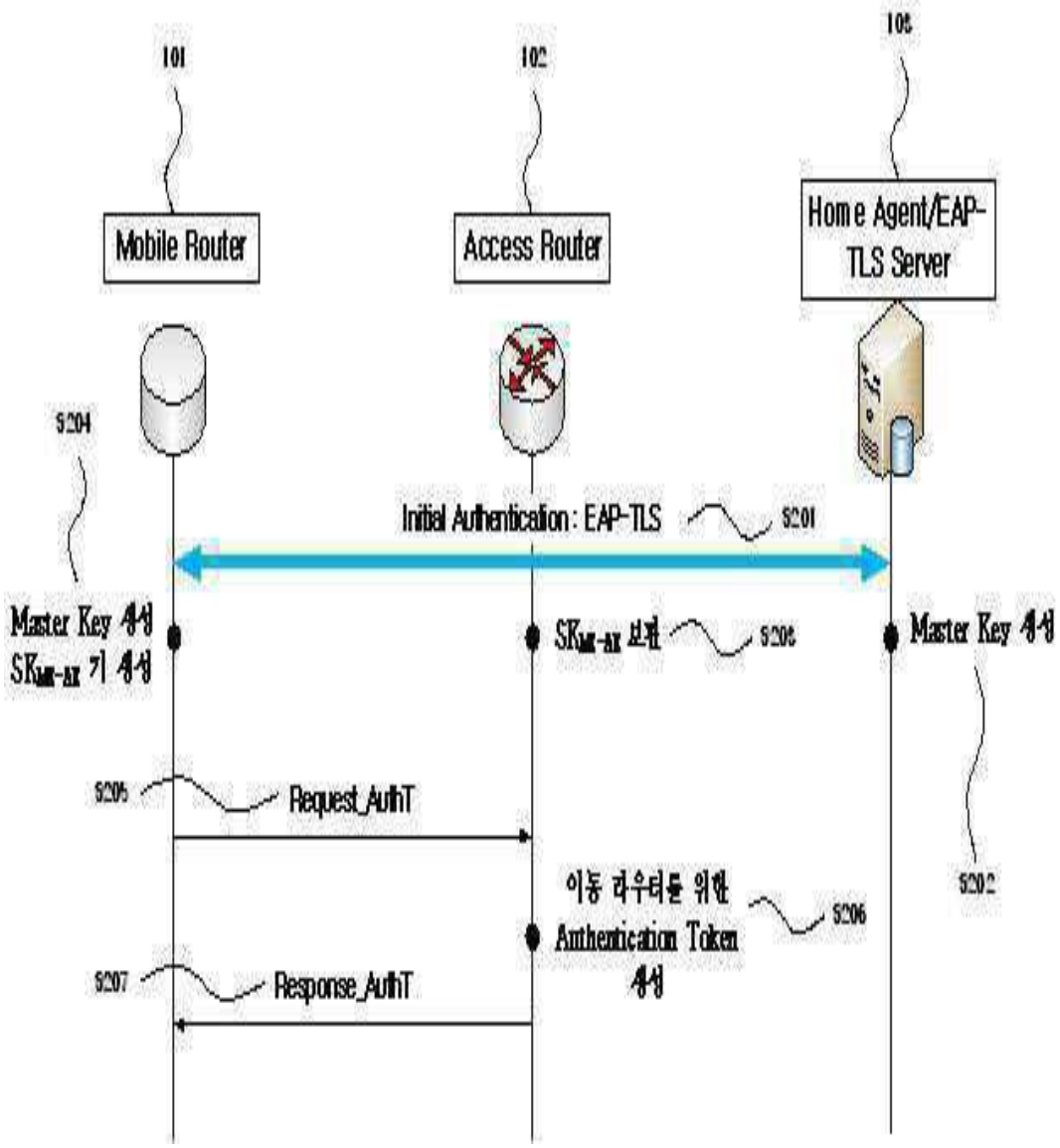
[0029] 도 3은 본 발명에 따른 인증 토큰을 발행받은 이동 라우터 간에 인증 토큰을 이용한 상호 인증 과정을 설명하는 도면이다.

도면

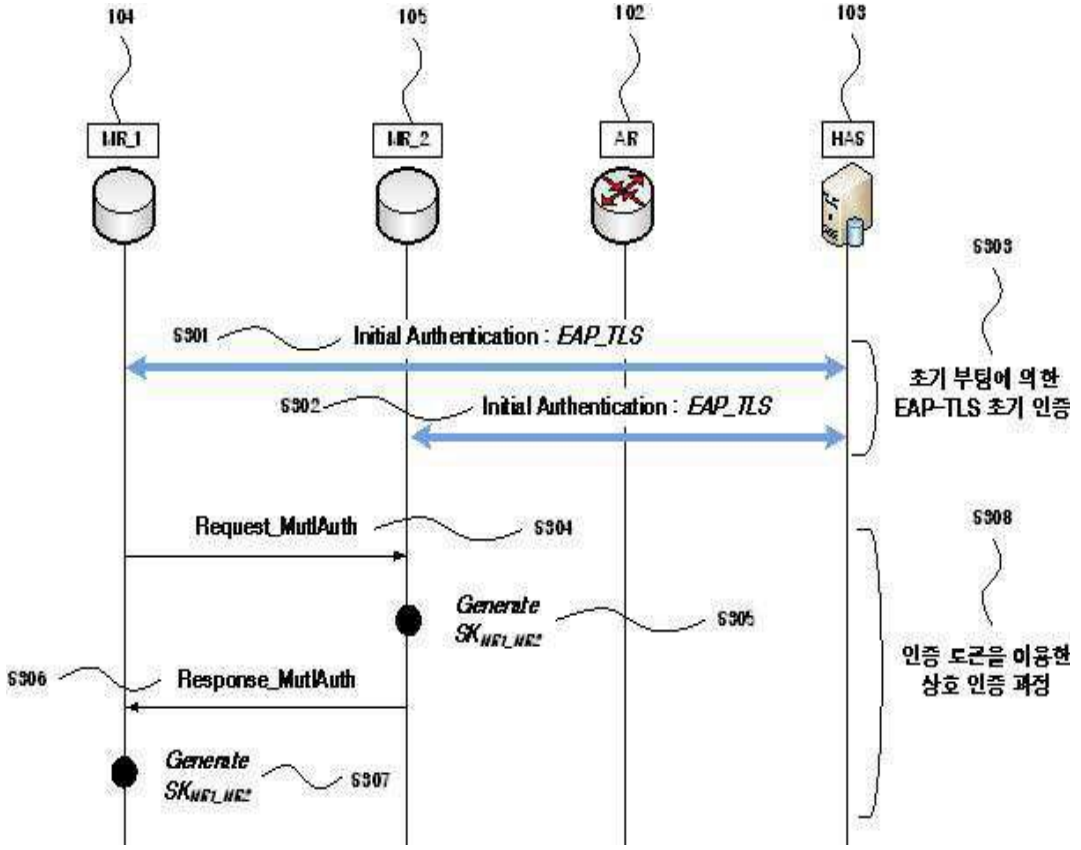
도면1


101
102
103
Mobile Router
Access Router
Home Agent/EAP-TLS Server
S101
EAP-Request Identity
EAP-Response Identity (NAI/ID)
S102
EAP-Request type EAP-TLS(TLS-start)
S103
EAP-Response type EAP-TLS(ClientHello)
S104
S108
EAP-Request type EAP-TLS(ServerHello, Server_certificate, Certificate Request, ServerKeyExchange)
S105
Master Key 생성
EAP-Response type EAP-TLS(ClientKeyExchange, Client_certificate, Finished)
S106
S109
EAP-Request type EAP-TLS(ChangeCipherSpec, Finished)
Master Key 생성
S107
EAP-Response type EAP-TLS
S110
EAP-Success
S111

도면2


101
102
108
Mobile Router
Access Router
Home Agent/EAP-TLS Server
S204
Initial Authentication : EAP-TLS
S201
Master Key 생성
SKam-ar 키 생성
SKam-ar 전달
S208
Master Key 생성
S205
Request_AuthT
S202
이동 라우터를 위한
Authentication Token
생성
S206
S207
Response_AuthT

도면3


104
105
102
103
MR_1
MR_2
AR
HAS
S303
S301
Initial Authentication : EAP_TLS
S302
Initial Authentication : EAP_TLS
초기 부팅에 의한
EAP-TLS 초기 인증
Request_MutlAuth
S304
S308
Generate
S305
인증 토큰을 이용한
상호 인증 과정
S306
Response_MutlAuth
Generate
S307