MANET 환경 하에서 멤버 노드간의 협력에 의해 분산된 인증서를 이용한 인증 서비스에 관한 연구

이대영* · 송상훈** · 배상현**

MANET Certificate Model Using Distributed Partial-Certificate with Cooperation of Cluster Member Node

Lee Dae-Young* · Song Sang-Hoon** · Bae Sang-Hyun**

이 논문은 2004년 조선대학교 연구지원비에 의해 지원 되었음.

요 약

Ad-Hoc 네트워크 기술이 미래의 이동 인터넷 기술로서 이동통신망(Mobile Network) 뿐만 아니라 공중무선랜망(WPAN) 그리고 유비쿼터스 망 등에 광범위하게 활용되기 위해서는 개선 및 보완되어야 할 기술적인 문제들이 많다. 특히, 최근 네트워크 보안의 허점을 이용한 보안상 공격이 급증하고 있는 상황에 반하여 Ad-Hoc 라우팅 프로토콜 연구 대부분은 보안 위협 요소를 배제하고 안전한 환경을 가정한 채 수행되고 있다. 또한 Ad-Hoc 네트워크가 무선 매체 특성상 더욱많은 보안상 위협에 노출되기 쉽고 유선에서 사용되던 보안 메커니즘을 그대로 적용되는 것이 부적합함을 고려할 때 Ad-Hoc 보안에 관한 연구는 더욱 활발히 이루어져야 할 것이다. 따라서 본 논문에서는 Ad-Hoc 네트워크의 특성을 고려하여 중앙 집중적인 인증기관이나 키 분배센터에 의존하지 않고 클러스터를 구성하고 있는 멤버 노드들 간의 협력적이고, 분산 된 인증서를 이용한 인증 서비스를 제공할 수 있는 모델을 제안한다. 아울러 시뮬레이션을 통해 제안한 모델의 확장성과, 견고성을 평가해 본다.

ABSTRACT

Ad-Hoc network technology is a mobile internet technology of the future that will be used widely not only in Mobile Network but also in Wireless Personal Area Network (WPAN) and Ubiquitous Network. For this to occur, distributed routing protocol design, loop prevention for link information, reduction in overhead for control messages and route restoration algorithm must be improved or complemented. Security techniques that can guarantee safe com-munication between Ad-Hoc nodes must also be provided. This study proposes and evaluates a new authentication mechanism for MANET. The mechanism segregates the roles of certification authority to keep with the dynamic mobility of nodes and handle rapid and random topological changes with minimal over-head. That is, this model is characterized by its high expandability that allows the network to perform authentication service without the influence of joining and leaving nodes. The efficiency and security of this concept was evaluated through simulation.

키워드

MANET, Certificate Management MANET Security

* 조선대학교 자연과학대학 전산통계학과
** 교신저자, 조선대학교 전산통계학과 교수

접수일자 : 2006. 10. 12

Ⅰ. 서 론

ad-hoc 네트워크는 중앙관리의 부재, 무선의 고유특성인 잦은 연결의 단절, 그리고 이동 호스트들의 이동으로 인한 잦은 위상 변경, 자원의 제약성과 같은 특성을 지닌다. 따라서 ad-hoc 네트워크 프로토콜은 유선 기반 네트워크 프로토콜에 비해 확장성(scalability) 및 견고성(robustness), 이동성(mobility) 그리고 경량성(lightweight) 등의 보장을 고려하여 설계하여야 한다. ad-hoc 네트워크 기술은 미래의 이동 인터넷 기술로서 이동 통신망 뿐만 아니라 공중 무선망 그리고 유비쿼터스 네트워크 망 등에 광범위하게 활용된다. 이를 위해서는 분산적인 라우팅 프로토콜의 설계, 링크 정보의 루프 방지, 제어 메시지의 오버헤드 감소, 경로 복구 알고리즘 등을 개선 및 보완 되어야 할 것이다. 그리고 Ad-hoc 노드 간 안전한 통신을 보장할 수 있는 보안 기법이제공되어야 한다. 특히 본 논문에서는 Ad-hoc 네트워크에서 인증서를 이용한 인증 서비스를 제공하기 위한 방안을 연구해 본다. 일반적으로 유선 네트워크 상에서의 키 관리는 인증기관(CA, Certification Authority)이나 신뢰 할 만한 키 분배 서버(Key Distribution Server)를 통하여 이루어 진다. 일반 호스트들은 그들의 서비스를 받아 키를 안전하게 분배받아 사용한다. 따라서 Ad-Hoc 네트워크는 신뢰할 만한 제 삼의 기관이 부재하므로 네트워크에 참여한 노드들끼리 협력적으로 키의 분배 및 관리를 수행하여야한다 [1][2].

Ad-Hoc 네트워크 환경에서 분산적인 인증기관에 의한 인증서 서비스 메커니즘은 다음과 같은 몇 가지 사항을 고려(consideration)해야 한다.

첫째, Ad hoc 네트워크의 특성상 중앙 관리 노드의 부재로 인해, Ad hoc 네트워크가 구성 될 때 마다 사전에 키 정보를 분배받는 형태를 취하게 된다. 이는 네트워크 구성 및 서비스 수행의 지연을 초래 할 수 있다.

둘째, 네트워크의 마스터 비밀키나 각 CA 노드들에게 분배된 부분 비밀키가 손상이 되었을 경우, 인증서 서명에 사용되는 기존의 비밀키를 해제하고 새로운 비밀키를 구축하는 메커니즘이 반드시 필요하다. 특히, CA의 수가 임계값 밑으로 떨어졌을 경우 원활한 CA에 의한 인증서 서비스를 위해 마스터 키의 재생성이 요구된다. 이는 온라인상으로 동적인 마스터 비밀키를 재구축하는 것은 불가능하며, 이는 견고성마저 매우 취약하게 된다. 따라서 본 논문에서는 새로운 보안 메커니즘으로 높은 확장성과 견고성을 지원 할 수 있는 인증키 관리 방안을 제안한다. 즉, Ad hoc 네트워크의 확장성을 반영하기 위해 네트워크 구성 초기에 CA 노드들은 키 관리자로부터 부분 비밀키를 사전에 분배 받을 필요 없이 온라인상에서 서로 인증서 서명에 사용될 마스터 비밀키를 동적으로 생성하고 재구축 할 수 있는 메커니즘을 제안한다.

Ⅱ. 관련연구

2.1. 비밀키 공유기법(secret shring)[3][4]

secret sharing은 임의의 비밀키를 여러 사용자들이 나누어 갖도록 함으로써 단일 사용자가 자신의 키만으로는 원래의 비밀키를 복원 할 수 없도록 하는 오래된 방법이다.

비밀키 공유 기법 중 대표적인 방법은 전체 n개의 부분키들로 분할되었을 때 이중에서 k개만을 얻으면 원래의 비밀키를 복원할 수 있는 (k,n)임계치 암호화 기법이다.

2.1.1 능동적인 secret sharing[1]

secret sharing은 원래의 키가 여러 호스트에 분산되므로, 임의의 시간 동안 k개의 부분키들을 얻기 어렵다는 가정 하에서 안전하다. 하지만 충분한 시간이 주어진다면 이러한 가정은 틀리게 될 수도 있다. 따라서 규칙적으로 분산된 부분키들은 update시킬 필요가 없다. 부분키 update는 다음과 같이 랜덤의 update 다항식 $f_{update}(x)$을 생성하여 원래의 다항식 f(x)에 더함으로써 이루어진다.

$$f(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \cdots + a_1x + a_0 \pmod{p}$$

$$f_{update}(x) = b_{k-1}x^{k-1} + b_{k-2}x^{k-2} + \cdots + b_1x \pmod{p}$$

$$f_{new}(x) = f(x) + f_{update}(x) = (a_{k-1} + b_{k-1})x^{k-1} + \cdots + (a_1 + b_1)x + a_0$$

따라서 임의의 수정된 부분키 $S_{i,update}$는 $f_{new}(id_i)$를 계산하면 얻어진다. 그러나 이러한 방법은 f(x) 자체를 복원해야 하므로 추가적인 통신 오버헤드를 유발한다. 따라서 실제로는 $f_{update}(x)$에 각각의 부분키 소유자 id를 대입하여 새로운 부분키 $\hat{S}_i, I=1,2...k$를 생성하고, 각각의 부분키 소유자에게 전송해서 update 된 부분키를 만들도록 한다.

2.1.2. 검증 가능한 secret sharing[1]

만일 임의의 부분키 소유자가 악의적인 생각으로 다른 부분키 소유자들이 원래의 비밀키를 복원하지 못하도록 부분키가 아닌 엉뚱한 랜덤 값을 전송한다고 가정했을 경우, 원래의 비밀키를 복원하리라 기대했던 사용자는 Lagrange 보간법에 의해 전혀 엉뚱한 값을 추출하게 된다. 이것을 방지하기 위해서는 임의의 부분키 소유자가 전송한 부분키가 유효한 값인지 검증할 방법이 필요하다.

비밀키 공유기법은 몇몇 사용자들에게 부분 비밀키를 분배하여 단일 부분 비밀키로는 원래의 비밀키를 복원 할 수 없도록 하는 방법이다. (k,n) 비밀키 공유기법[3][5]은 전체 n개의 부분 비밀키중에서 최소 k개를 병합하여 원래의 비밀키를 복원할 수 있는 기법이다. 비밀키 S는 n개의 부분 비밀키 소유자 $id_1, id_2, ..., id_n$들에게 분할된다. 이 비밀키의 생성과 분할은 신뢰된 임의의 키 관리자의 역할이다.

2.2. 분산 인증기관기법[6][7]

분산인증 기법은 (k,n) 임계치 암호화 기법을 처음으로 Ad hoc 네트워크에 적용한 방법이다. 이 방법에서는 클라이언트, 서버, combiner와 같은 세 종류 노드의 구성을 가정하며 서버와 combiner는 인증기관의 일부 기능으로 포함된다. 또한 이 방법은 공개키 암호화 방식을 사용하는 것을 가정하며, 인증기관의 공개키는 모든 네트워크에 공개되어 있고 비밀키는 각각의 서버에 부분키로 분산되어 있다. 이들 서버 중 임의의 하나가 dealer의 역할을 수행하며, dealer는 각 서버의 공개키를 분산시키는 역할은 물론각 사용자의 공개키에 대한 초기 인증서를 발행하는 역할을 한다. 이 초기 인증서는 임의의 호스트가 네트워크에 가입할 때 이들을 인증하기 위한 방법으로 사용된다. 이 방법에서 부분키가 분산된 각 서버는 자신이 소유한 부분키로 서명한 부분인증서를 저장하고 있고, 클라이언트의 요청에 따라 임의의 서버(combiner)가 이러한 부분 인정서를 k개 이상 수집하면 원래의 인증서를 복원하여 클라이언트에게 전달한다.

분산 인증기관 기법은 인증서의 갱신이나 update후에 서버들간의 동기화 기법을 요구하므로 많은 통신 오버헤드가 유발된다. 더구나 새로운 부분 인증서는 기존의 부분 인증서를 더하여 생성하므로 잦은 이동을 수반하는 네트워크에서는 원래의 인증서로 복원이 가능하지 않게 되는 경우도 있다.

2.3. 신뢰기관을 사용하지 않는 임계치 암호화 기법[8]

2.3.1 기본 프로시저

pedesen은 Desmedt와 Frankel이 제안한 임계치 암호화 기법을 다음 두가지측면에서 개선하였다. 첫째, 비밀키를 선택하고 분배하는 신뢰기관을 사용하지 않도록 하는 것이다. 둘째로 멤버들이 검증 가능한정확한 비밀키들을 공유할 수 있도록 하였다. 각 노드는 안전하게 자신의 부분키를 생성한다. 각 노드 P_i는 소수인 p, q 그리고 발생자(generator) g를 공유하고 있다.

Step 1 : 공개키 생성

각 노드 Pi는 난수인 Xi를 생성하고 hi=gXi 를 계산하여 다른 노드들에게 전송한다. 각 노드가 모든 hi를 획득하게 되면 공개키 h를 생성할 수 있다.

$$h = \prod_{i-1}^{n} hi$$

step 2 : 검증 가능한 비밀키 분배

노드 Pi는 각자가 선택한 난수 X_i를 y절편으로 하는 임의의 다항식 $f_i(z)$를 생성한다. 이때 $f_i(z)$는 임계값 k라 할 때, k-1차 다항식이며 $f_i(0)$은 X_i이다.

$$f_i = f_{i0} + f_{i1}X + f_{i2}X^2 + \cdots + f_{iK-1} \cdot X^{K-1}$$

$$(f_{i0} = X_i)$$

분배되는 부분키의 검증을 위해 노드 Pi는 자신이 생성한 다항식에 대해서 검증 계수 Fij를 계산하여 브로드캐스트 한다.

$$F_{ij} = g^{fij} (j = 0,..., k-1)$$

모든 노드가 K-1개의 검증 계수 정보를 전송하였을 때, P_i는 자신이 생성한 난수 값에 대한 부분 키 S_{ij}를

생성하여 다른 노드들에게 전송한다.

$$S_{ij} = g^{fi(j)}(j = 0,...,n)$$

step 3: 부분키 검증

각 노드 P_i는 전송 받은 부분키들을 조합하기 전에 다음과같은 연산을 통해 부분키 정당성을 검증한다.

$$g^{Sji} = \prod_{l=0}^{K-1} F_{jl}{}^{il}$$

만일 검증시 실패가 발생하는 경우 Pi는 에러 메시지를 브로드캐스팅하고 해당 노드에게 부분 키의 재전송을 요청한다.

step 4: 부분 비밀키 생성

각 노드 Pi는 전송 받은 부분 키인 Si에 대한 연산을 통해 공개키에 대응하는 비밀키를 연산한다.

$$S_i = \prod_{j=1}^{n} S_{ji}$$

step 5: 비밀키 획득

각 노드 Pi는 임계치 이상의 부분 비밀키인 S_i를 얻으면 다음과 같은 임계치 암호화 기법에 의해 완전한 비밀키를 획득 할 수 있다.

$$S = \prod_{j=1}^{k} S_j$$

Ⅲ. 인증키 관리 모델

본 논문에서는 초기 신뢰 기관으로부터 오프라인 상으로 인증서 서명을 분배 받지 않고 통신에 참여하는 노드들이 온라인 상에서 협력적으로 비밀키를 공유하고 인증서를 발행 할 수 있는 모델을 제안한다. 이 기법은 유선 환경을 기반으로 제안되었던 신뢰기관을 이용하지 않는 임계치 암호화 기법을 기반으로 Ad-Hoc 환경에 적합하게 적용한 것이다. 본 논문에서 제안하는 인증키 관리 모델을 기술하기 전에, 일반적인 평면 구조의 Ad-Hoc 네트워크에 신뢰기관을 이용하지 않은 임계치 암호화 기법을 적용 할 경우 발생하는 문제점을 분석하고, 이를 해결하기 위한 가정 및 접근 방법을 제시한다.

3.1. 문제점 분석

평면적 구조의 일반적인 Ad-Hoc네트워크를 유선 환경을 기반으로 제안된 신뢰기관을 이용하지 않는 임계치 암호화 기법을 적용하고자 할 때 다음과 같은 중요한 문제들이 발생한다.

첫째, 시그널 오버헤드가 매우 크다. CA노드 i는 자신을 제외한 나머지 CA 노드들과 협의하여 공개키 및 부분 서명키를 생성하기 위해 키 값들을 교환해야 한다. 즉, n-1개의 h_i 값과 임의의 난수 X_i의 부분 값 n-1개를 각각 n-1개의 노드들에게 분배해야 한다. 따라서 n개의 CA노드들이 전송해야 할 부분 키 메시지의 총 개수는 n(n-1)개이다. 그런데 일반적으로 Ad-Hoc 네트워크의 경우 플루딩(flooding)방식으로 메시지가 전송되므로 실제 전달되는 메시지는 n(n-1)보다 더욱 증대될 것이다. 이와 같은 문제점은 Ad-Hoc 네트워크의 확장성을 저해하는 요인이 된다.

둘째, CA 노드간 키 일치율(key agreement)가 낮다. CA노드는 네트워크상에 임의로 분산되어 있다. Ad-Hoc 네트워크는 무선 매체의 특성상 전달되는 메시지가 도청 및 변조되기 쉽다. 또한 네트워크의 토폴로지가 동적으로 변화하며 네트워크 경로가 불안정하여 메시지가 도중에 유실될 가능성도 매우 높다. 특히 CA노드의 수가 많을수록 키 일치를 이루어야 할 노드수가 많아지므로 키 일치가 성공적으로 이루어질 확률이 감소된다.

셋째, CA노드들은 네트워크상에 임의로 분산되어 있고, CA 역할을 수행하는 노드들이 동적으로 변경될 수 있으므로, 키 일치에 중요한 파라미터인 키 일치 참여 노드 수(n)와 임계값(k)를 정확히 파악하는 것이 어렵다. 따라서 키 일치에 모든 노드가 참여할 CA 노드 수와 노드 수에 종속적인 임계값을 올바르게 파악하는 것은 매우 어려운 일이다. 또한 참여 노드 수와 임계값의 동적인 반영이 이루어 질 수 있어야 한다.

3.2. 가정 및 접근 방법

전술한 문제점들을 해결하기 위해 다음과 같은 방안이 간구 될 수 있다.

첫째, 시그널 오버헤드를 줄이기 위해 CA노드들간에 플루딩 되는 메시지의 수를 감소시켜야 한다. 만일 유니캐스트 전송 방식이 사용된다면 전송 메시지 수가 크게 감소 될 수 있다. 둘째, CA노드간 키 일치 확률을 높이기 위해 키 협의에 참여하는 노드들의 수(n)를 가능한 제한

해야 한다. 셋째, 키 일치 파라미터인 참여 노드 수(n)와 임계값(t)을 모든 CA 노드들이 동일하게 정확히 파악할 수 있도록 협의에 참여하는 노드를 동적으로 파악 할 수 있는 Ad-Hoc 네트워크 구조를 기반 구조로 삼아야 한다.

따라서 본 논문에서는 다음과 같은 기법으로 전술한 요구사항을 만족 시킴으로 문제점을 해결하였다. 첫째, 클러스터 기반 Ad-Hoc 네트워크에서 head 노드만이 CA로서 공개키 및 부분 서명키 분배에 참여하도록 한다. 둘째, 클러스터 헤드간 교환되는 부분 키 정보는 클러스트 백본망을 이용하여 유니캐스트된다. 셋째, 궁극적으로 완전 분산 CA구조를 갖는다. 초기에 클러스터 헤드들에 의해 마스터 공개키와 비밀키 구축이 완료된 후, 클러스터 멤버 노드들에게도 부분 서명키를 분배하여 모든 노드가 CA로서 역할 수행을 가능하게 한다. 넷째, 인접 클러스터 테이블을 통해 인접 클러스터 헤드에 관한 정보가 동적으로 관리되는 특성을 이용하여 CA노드들이 CA 노드 수와 임계값에 관한 정보를 획득하도록 한다.

또한 본 논문에서 제안된 모델은 다음과 같은사항을 가정한다.

첫째, Ad-Hoc 네트워크는 단일 도메인을 구성한다. 둘째, 라우팅 보안(Secure Routing)[9]이나 사용자 인증의 부분은 본 논문에서 다루지 않는다. 기본적으로 Ad-Hoc 노드간에 전송되는 메시지는 데이터의 무결성과 기밀성이 보장된다. 셋째, 각 Ad-Hoc 노드들은 임의의 생성자(generator)를 알고 있다. 넷째, 악의적인 내부 공격의 가능성은 배제한다.

3.3. 인증키 관리 모델

3.3.1 공개키와 검증 가능한 부분 비밀키의 생성

Ad-Hoc 네트워크로 구성된 단일 도메인 내에서 사용되는 도메인 공개키와 인증성 서명 비밀키 쌍을 생성 구축하는 과정이다. 이 단계에서는 대표적으로 클러스터 헤드들이 CA 역할을 수행하며, 도메인에서 사용될 공개키와 마스터 비밀키를 생성하는데 참여한다. 헤드들은 각자 생성한 임의의 난수를 이용하여 도메인의 공개키를 협력적으로 생성하고 검증 가능한 부분 비밀키들을 분배한다.

1) 도메인의 CA 공개키 생성

① 부분 공개키 분배

다음 그림과 같이 각 CA는 임의의 난수 X_i를 발생시키고 부분 공개키 값이 h_i를 생성하여 클러스터 헤드 백본망을 통해 다른 CA들에게 유니캐스트 한다.

② 공개키 획득

부분 공개키 분배가 완료되면 CA들은 각가 6개의 부분 공개키 $h_i(1 \le i \le 6)$를 획득하게 된다. 각 CA 노드는 $h = \prod_{i=1}^{n} h_i$ 연산에 의해 도메인 공개키를 생성 보유하게 된다.

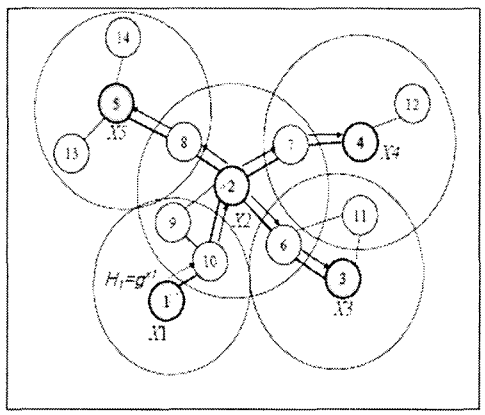

그림 1. 도메인 공개키 생성 단계
Fig. 1. Domain public key generation step

2) 검증 가능한 부분 비밀키 생성

① 검증 계수 교환

CA 노드 p_i는 각각 생성한 난수 x_i를 y절편으로 하는 k-1차(k:임계값)의 임의의 다항식 f_i를 생성한다. 교환하는 부분 비밀키의 타당성 검증을 위해 다항식 f_i에 대한 정보인 $F_{ij} = g^{f_{ji}}$를 클러스터 백본망을 통해 다른 CA 노드들에게 유니캐스트 한다.

② 난수에 대한 부분키 전송

CA 노드 p_i는 자신이 생성한 임의의 수에 대한 부분키 값이 $S_{ij} = g^{f_{ij}}$를 클러스터 백본망을 통해 다른 CA 노드인 p_j들에게 유니캐스트 방식으로 전송한다.

③ 부분키 검증

검증 계수와 난수에 대한 부분키 전송이 모두 완료되면 CA노드 p_j는 전송 받은 부분키 S_{ij}가 정당한 값인지

검증하기 위하여 다음과 같은 계산을 수행한다.

$$g^{Sji} = \prod_{l=0}^{K-1} F_{jl}{}^{il}$$

만일 검증에 실패할 경우에러 메시지를 전송하고 해당 노드에게 실패한 부분키 S_{ij}에 대한 재전송을 요청한다.

④ 부분 서명 비밀키 생성

검증 계수를 사용하여 성공적으로 전송 받은 난수에 대한 부분키의 검증이 모두 완료되면 CA 노드 P_i는 다음 연산을 통해 서명 비밀키에 대한 부분키를 생성한다.

$$S_i = \prod_{j=1}^{n} S_{ji}$$

다음 그림은 각 CA 노드가 자신의 부분 서명 비밀키를 생성하는 단계를 보여주고 있다.

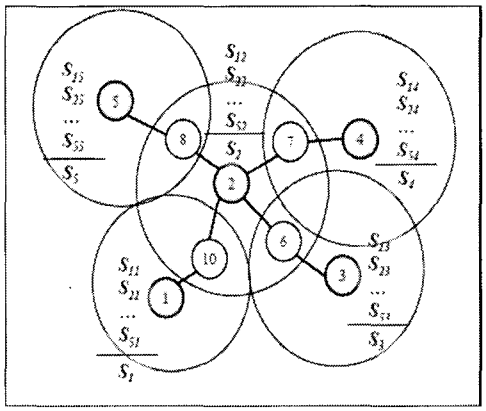

그림 2. 부분 비밀 서명키 생성
Fig. 2. Partial Secret Authorization Key Generation

3.3.2 부분 비밀키의 완전 분산 인증 메커니즘

클러스터의 멤버들에게 부분 비밀키를 분배함으로써 도메인 내에 존재하는 Ad-Hoc 노드가 중앙 집중적인 인증기관을 의존하지 않고 임계치 이상의 이웃 노드와 협력을 통하여 인증서 서비스를 제공 받을 수 있는 완전 분산 인증 구조이다.

1) 클러스터 멤버에게 부분 비밀키 분배

각 클러스터 헤드는 도메인 CA의 공개키와 비밀키 쌍을 생성하는데 대표로 참여한 협력적인 분산 CA 들이 다. 각 클러스터 헤드가 도메인 CA 공개키와 부분서명 비밀키 구축을 완료하면 클러스터 멤버들에게도 부분 서명 비밀키를 분배한다. 이것은 동일 도메인내에 존재하는 모든 Ad-Hoc 노드들이 부분 CA 로서의 기능을 하는 완전 분산 CA 구조를 띄는 것으로 이동성과 가용성이 좋아 키 일치 확률을 높일수 있다.

다음 그림은 각 클러스터에서 모든 멤버 노드들에게 부분 서명키 분배가 완료된 후, 완전 분산 CA구조를 나타내고 있는 것이다. 이는 도메인 내의 모든 Ad-Hoc 노드가 도메인 단일 CA에 대한 부분 CA로서의 기능을 수행 할 수 있게 한다.

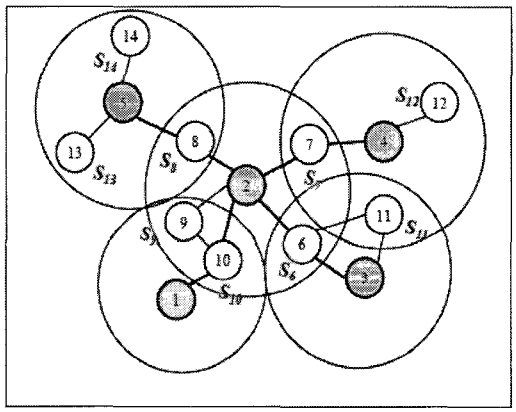

그림 3. 부분 서명 비밀키의 분산
Fig. 3. Complete Distribution of Partial Authorization Secret Key

2) 서명키 생성 및 인증서 발행

위와 같이 생성된 부분 비밀 서명키는 다음 그림과 같은 방식으로 사용된다. 가령, 노드 p2가 인증서 발행을 위해 공개키 h에 대한 서명을수행하고자 할 때, p2는 인접하는 노드들에게 부분 서명키를 요청한다. 이때, p2는 임계치 이상의 부분 서명키를 획득하면 서명키 S를 생성하여 인증서 발행을 수행 할 수 있다.

3.3.3 클러스터내의 새로운 노드의 진입

다음 그림과 같이 새로운 노드가 도메인 내의 한 클러스터 영역에 진입할 경우, 분산 CA로서 원활한 인증서 발급 서비스를 수행 할 수 있도록 부분서명키 S_{id}를 분배 해 줘야 한다.

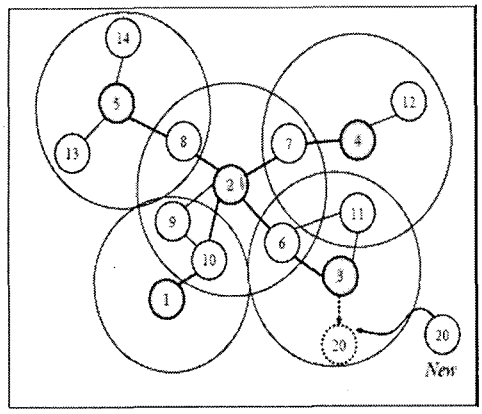

그림 4. 클러스터 내의 새로운 노드의 진입
Fig. 4. Entry of a New Node within a Cluste

새로운 노드의 클러스터 진입 시, 헤드 노드에게 멤버 노드 등록 요청 메시지를 보낸다. 이어 헤드 노드는 새로운 노드 id를 멤버 노드로서 등록하고 id에 대해 부분 비밀키인 Sid를 생성하여 새로운 노드에게 분배한다. 이를 위해 self-initialization메커니즘을 적용하고, 이는 기존에 통신하고 있는 노드들이 새로운 노드 유입에 영향 받지 않고 동일한 서명키로 인증 서비스를 수행할 수 있다. 또한 새로운 노드에게 분산CA 기능을 분배함으로써 높은 확장성을 지원한다.

3.3.4 마스터 CA 키 재 구축 메커니즘

유동적인 Ad-Hoc 노드들의 움직임으로 분산 CA의 수가 임계치 밑으로 감소되었을 경우, CA 공개키와 비밀 서명키 쌍을 재 구축할 수 있는 메커니즘이 필요하다. 임계치 이하의 부분키로는 서명키를 획득 할 수 없기 때문이다. 또한 마스터 서명키 노출에 대한 위협에 대해 안정성을 보장 하기 위해서도 역시 CA 공캐기와 비밀키 쌍을재 구축 해야 한다.

새로 선정된 헤드 노드는 새로운 임의의 난수를 생성하여 다른 헤드 노드들에게 전송한다. 헤드 노드들은 전송 받은 검증 계수와 부분 키 정보를 통해 새로운 CA 공개키와 부분서명키를 구착한다. 이어 각 클러스터 헤드 노드는 멤버 모드들에게 노드의 id를 기반으로 새로 구축된 서명키에 대한 부분키를 생성하여 분배한다.

Ⅳ. 실험 및 평가

본 논문에서는 유선 환경을 기반으로 제안되었던 신뢰기관을 이용하지 않는 임계치 암호화 기법을 기반으로 Ad-Hoc 환경에 적합한 구조로 적용할 수 있는 모델을 제안하였다. 이 장에서는 제안한 모델과 유선 환경을 기반으로 제안되었던 신뢰기관을 이용하지 않는 임계치 암호화 기법을 상호 비교해봄으로써 효율성과 견고성 검증해 보기로 한다.

CA 비밀키를 구축하기 위해 전송되는 메시지 수를 비교하여 시그널 오버헤드에 대한 효율성을 비교해 본다. 또한 일정한 네트워크 전송 에러율이 존재하는 상황에서 전송되는 부분 키들이 안전하게 전송될 확률, 즉 키 일치 확률을 비교해 본다. 시그럴 오버헤드가 적을수록 키 일치 확률이 낮을수록 각각 효율성과 견고성이 높다고 판단 할 수 있다.

4.1. 시뮬레이션 환경

본 논문에서 제안한 시스템을 시뮬레이션 하기 위하여 NS2(wireless extention)를 이용하며, IEEE 802.11 link layer와TDMA(Time Division Multiple Access)를 따른다. 실험은 500mX500m 영역에서 45개의 호스트를 가지고 600초 동안 수행되었다. 모든 호스트들은 초기에 랜덤하게 배치되고 매초마다 임의의 방향으로 이동한다고 가정한다. 키 일치는 "hello"메시지 주기 사이에 한번씩 발생하도록 하였고, 키 일치를 시도하는 호스트들은 무작위로 선정된다.

먼저 확장성 측면서에 중요 요소인 시그널 오버헤드를 측정하기 위하여, CA 비밀키를 구축하기 위해 각 노드가 전송하는 부분 키 메시지 수를 비교한다. 또한 견고성 측면에서 10%의 네트워크 전송 에러율이 존재하는 상황에서 부분 키들이 각 노드들에게 손실되지 않고 전송될 확률 즉, 키 일치율을 비교한다. 이는 부분 키의 전송이 완료된 후 각 노드가 정상적으로 전송 받은 메시지 수를 측정함으로써 키 일치율을 구할 수 있다.

경로 설정 및 유지를 위한 제어 메시지는 포함되지 않으며 평면적 구조의 Ad-Hoc 네트워크에서는 기본적인 플루딩 방식으로, 클러스터 기반 Ad-Hoc 네트워크에서는 클러스터 백본망을 통한 유티캐스트 방식으로 메시지를 전송한다.

4.2. 성능 분석 및 평가

4.2.1 부분 키 메시지 오버헤드 비교

1) 호스트 수에 따른 메시지 오버헤드 비교

Ad-Hoc 통신에 참여하는 호스트의 수를 증가에 따라 CA 비밀 키 구축을위한 부분키를 전송하는 메시지 수를 비교해 보았다. 다음 그림은 호스트 수의 증가에 따른 부분키전송 메시지 수를 비교한 것이다. 호스트의 수가 증가됨에 따라 DCAS(Distributed Certificate Authentication Servic, 제안모델)과 CCAS(Cryptosystem certificate Authentication Servic) 모두 부분키 전송 메시지 수가 증대되는 것을 볼 수 있다. 그런데 DCAS에서 전송되는 메시지 수는 CCAS에 비해 전체적으로 적었으며, 약 39% 정도의 메시지 오버헤드 효율이 있었다. 이는 호스트 수가 증가되더라도 새로 진입한 호스트가 멤버 노드로서 기존의 클러스터에 소속되는 경우가 존재하여 전체적으로 교환되는 부분키 메시지 수에는 적게 영향을 미치기 때문이다.

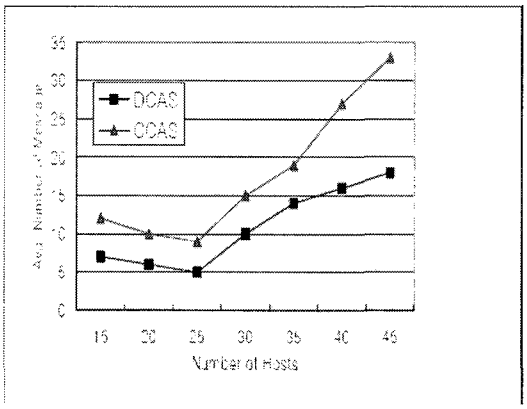

그림 5. 호스트 수에 따른 부분 키 전송 메시지 수의 비교
Fig. 5. Comparison of Partial Key Transmission Message Number According to Number of Host

2) 전송 범위에 따른 메시지 오버헤드의 비교

다음 그림은 통신에 참여하는 Ad-Hoc 노드들의 전송거리를 변화시켰을 경우의 영향을 그래프로 도시한 것이다.

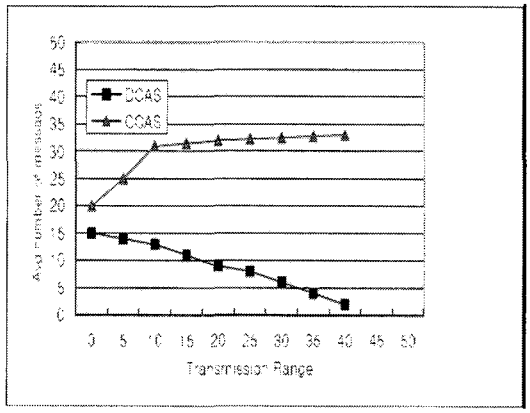

그림 6. 전송범위에 따른 부분키 전송 메시지 비교
Fig. 6. Comparison of Partial Key Transmission Message Number According to Transmission Range

전송거리가 증대될수록 각 노드는 통신할 수있는 통신 링크수가 더욱 증대되므로 CCAS에서는 전체적으로 전송되는 메시지 수가 증대된다. 그러나 DCAS의 경우 각 노드의 전송 거리가 증대될수록 클러스터 수가 감소하고 따라서 클러스터 헤드 수가 감소되므로 전체적으로 교환되는 부분키 메시지 수가 감소된다.

3) 시뮬레이션 공간에 따른 메시지 오버헤드의 비교

Ad-Hoc 통신이 수행되는 시뮬레이션 공간의 크기의 변화가 미치는 영향을 실험해 보았다. 공간의 크기가 증대될수록 통신 노드들은 임의의 위치에 불규칙하게 분포될 가능성이 많아진다.

CCAS의 경우 전체적으로 공간 범위가 증대될수록 전송 메시지의 수가 감소되는 것을 볼 수 있다. 이에 반해 DCAS의 경우 300X300의 경우를 기점으로 메시지 수 가증가 상태에서 감소 상태로 전환된다. 이것은 CCAS의 경우 공간이 증대될수록 노드간의 통신 링크수가 감소되어 노드가 단절되는 경우가 증대되기 때문이다. 따라서 부분키 메시지가 전달되지 못하는 경우가 증대되어 전송 메시지 수는 감소된다. 이에 반해 DCAS의 경우 300X300 경우 전에는 통신하기 위한 클러스터 수가 점차 증대되므로 헤드 수 또한 증대되어 부분키 메시지 수가 증대된다. 그러나 300X300이상 증대되는 경우 클러스터의 고립현상이 증대되어 메시지가 전송되지 않는 경우가 증대되어 메시지 수가 감소되는 현상을 볼 수 있다. 그러나 이 실험헤서도 마찬가지로 DCAS이 CCAS 보다 메시지 전송 효율이더욱 좋았다.

4.2.2 부분 키에 대한 키 일치율 비교

1) 호스트 수에 따른 키 일치율 비교

Ad-Hoc 통신에 참여하는 호스트의 수를 증가시킴에 따라 CA 비밀키 구축을 위해 전송되는 부분키들이 각 노드에서 안전하게 전송되는 확률, 즉 키 일치 확률을 실험해 보았다. 다음 그림은 호스트 수에 따른 부분키 키 일치율을 도시한 것이다. 실험 결과 호스트 수가 증대됨에 따라 DCAS과 CCAS 모두 전체적으로 키 일치 확률이 증대되는 것을 알 수 있다. 이는 일정한 전송 범위와 일정한 이동 범위, 즉 일정한 실험 공간에서 호스트 수가 증대될수록 노드간에 서로 통신할 수 있는경로 또한 증대되기 때문이다. 따라서 일부 경로에서 단절이 이루어지더라도 다른 경로를 통해 부분키가 전송될 확률이 높다. 특히, CCKN이 CCAS보다 키 일치 확률이 더욱 높은 것으로 나타났다. 이는 서로 키 일치를 이루어야 할 호스트의수는 클러스터 헤드의 수로 CCAS보다 더욱 적기 때문이다.

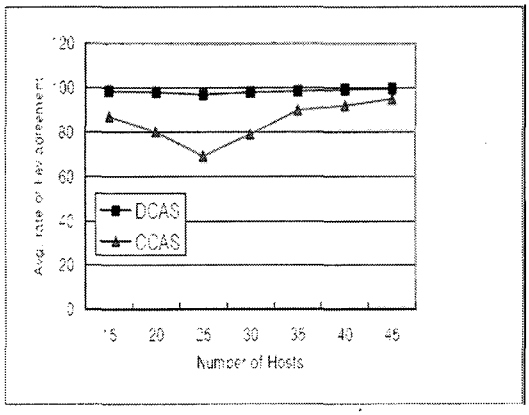

그림 7. 호스트 수에 따른 부분키 키 일치율
Fig. 7. Key agreement rate for number of hosts

2) 전송범위에 따른 키 일치율 비교

Ad-Hoc통신에 참여하는 노드들의 전송거리를 변화시켰을 때의 영향을 살펴 보았다. 다음 그림과 같이 전송거리가 증대될수록 각 노드는 통신 할 수 있는 통신 링크 수가 더욱 증대되므로 CCAS에서는 전체적으로 키 일치 확률이 매우 높아지는 것을 볼 수 있다.

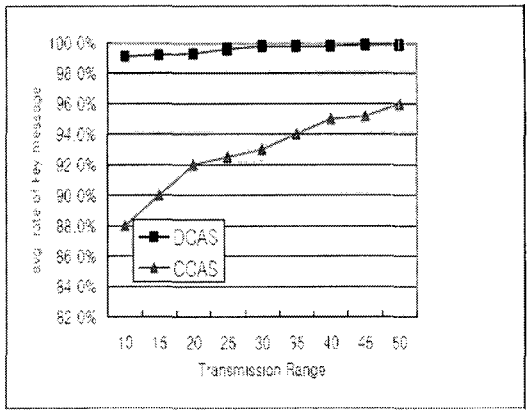

그림 8. 전송범위에 따른 부분키 키 일치율
Fig. 8. Partial key agreement rate according to transmission rang

링크 수가 증대될수록 다른 다양한 경로를 통해 다른 노드가 전송한 부분키들을 전송 받을 수 있는 확률이높아지기 때문이다. DCAS의 경우 전송거리에 크게 영향받지 않고 대부분 100%에 가까운 키 일치율을 보였다.

3) 시뮬레이션 공간에 따른 키 일치율 비교

Ad-Hoc 통신이 수행되는 시뮬레이션 공간의 크기의 변화가 미치는 영향을 실험해 보았다. 공간 크기가 증대될수록 통신 노드들은 임의의 위치에 불규칙하게 분포될 가능성이 많아진다.

CCAS의 경우 키 일치 확률이 현격하게 감소되는 것이 나타났다. 이것은 전체적으로 키 일치를 이루어야 할 노드들이 불규칙하게 분포되어 서로 키를 전달 받지 못할 만큼 통신으로부터 고립될 수 있기 때문이다. 그러나 이에 반해 DCAS의경우 키 일치를 이루어야 할 노드 수가 클러스터 헤드 수로서 적다. 뿐만 아니라 노드들이 임의로 흩어진다하더라도 대부분 어느 한 클러스터에 속하거나 새로운 클러스터를 구성해 게이트웨이를 통해 다른 헤드와 키 일치를 이룰 확률이 높으므로 시뮬레이션 공간 크기에 크게 영향 받지 않고 높은 키 일치율을 보였다.

V. 결 론

본 논문에서는 Ad-Hoc 네크워크의 특성을 고려하여, 신뢰 된 기관(trusted)으로부터 오프라인 상으로 인증서

서명키를 분배 받지 않고 통신에 참여하는 노드들이 온라인 상에서 협력적으로 비밀키를 공유하고 인증서를 발행 할 수 있는 모델을 제안하였다. 본 연구결과 제안한 모델은 오프라인 상으로 키 관리자로부터 부분 서명키를 사전에 분배 받거나 중앙 집중적인 인증기관에 의존 문제를 완전히 해결하였다. 또한 CA노드수가 임계치 밑으로 떨어졌을 경우CA 서명키를 재구축하거나 새로 진입한 노드에게 부분키를 동적으로 발행하여 Ad-Hoc 네트워크 토폴로지 변화를 능동적으로 방영 할 수 있었다. 향후 센서 Ad-Hoc 네트워크나 WPAN Ad-Hoc 네트워크의 특성을 구체적으로 고려하여 제안한 모델을 응용 할 수 있는 방안을 연구해 보는 것도 매우 의미 있는 일이 될 것이다.

참고문헌

[1] L. Zhou, Z J. Haas, "Securing Ad Hoc Network", IEEE Network, 13(6), pp.24-30, 1999

[2] H. Luo, S. Lu, "Ubiquitous and Robust Authentication Service for Ad Hoc Wireless Network, " Technical Report TR-2000, 30, Dept. of Computer Science, UCLA, 2000

[3] Y. Frankel, Y. G. Demedt, "Parallel Reliable Threshold Multi-signature", Univ. of Wisconsin-Milwaukee. Technical Report TR-92-94-02. Apr. 1992

[4] A. Shamir, "How to Share a Secret" Communication of the ACM, 22(11), pp.612-613, 1979

[5] S. Basagni, K. Herrin, E. Rosti, and D. Bruschi, "Secure Pebble-nets", Proc. of the ACM Symposium on MobiHoc, pp.156-163, 2001.

[6] J. Kong, P. Zerfos, H. Luo, S. Lu, L.Zhang, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Network," IEEE 9th International conference on Network Protocols(ICNP'01), pp.251-260, 2001

[7] H.Luo, J.Kong, P. Zerfos, S. Lu, L.Zhang, "Self-securing Ad Hoc Wireless Networks," 7th IEEE Symposium on Computer and Communications, pp.1627-1637, 2000

[8] T. P Pedersen, "A Threshold Crypto System without a Trusted Party," In Advances in Cryptolgy Eurocypt 91, pp. 522-526, 1991

[9] K. Sanzgiri, B. Dahill, B Levine, C Shields, E. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Network,"Proc. of ICNP02, pp.78-87, 2002

저자소개

이 대 영(Lee Dae-Young)

1993.3~1999.2 조선대학교 학사
1999.3~2001.2 조선대학교 전산통계학과 석사
2002.3~2006.2 조선대학교 전산통계학과 박사

※관심분야 : MANET protocol, 유무선 네트워크 보안, 유비쿼터스 컴퓨팅

송 상 훈(Song Sang-Hoon)

조선대학교 컴퓨터통계학과 교수
공학박사

※관심분야 : 인공지능, 지리정보시스템, 멀티미디어시스템, 에이젠트통신시스템

배 상 현(Bae Sang-Hyun)

1999~2002 ㈜멀티정보
2002~2003 ㈜모노
2004 조선대 컴퓨터통계학과 학사
현재 조선대 컴퓨터통계학과 석사

※관심분야 : 웹디자인, 미디어서버