情報保護學會論文誌
第16卷 第5號, 2006. 10

모바일 Ad-hoc 네트워크에서 Hamming Distance를 이용한 인증프로토클

이 석 래,† 송 주 석‡
한국정보보호진흥원,† 연세대학교 컴퓨터과학과‡

Authentication Protocol Using Hamming Distance for Mobile Ad-hoc Network

SeokLae Lee,† JooSeok Song‡
Korea Information Security Agency,† Yonsei University Computer Science‡

요 약

모바일 Ad-hoc 네트워크는 인프라에 관계없이 동작하고 제3의 신뢰기관 없이 운용되어야 한다. 또한 네트워크 크기, 노드 이동성, 기기의 배터리 용량 및 메모리 크기 등에 의하여 구현상 제약을 받는다. 그럼에도 불구하고 유선 네트워크처럼 인증, 기밀성, 무결성, 부인방지, 접근통제, 그리고 가용성 등 보안문제에 대한 고려가 필요하다. 특히, 이러한 보안문제들 중에서 인증은 네트워크 특성에 상당한 영향을 받기 때문에 본 논문에서는 인증에 초점을 맞추었다. 본 논문은 Hamming Distance의 개념을 도입하여 모바일 Ad-hoc 네트워크에서 공개키 인증서 생성·갱신·폐지 등 공개키 인증서 관리를 위한 프로토콜 및 공개키 인증서 검증을 위한 경로구축 알고리즘을 제안하고 그 성능을 평가한다. 본 논문에서 제안하는 경로구축 알고리즘은 각 노드의 공개키 인증서 저장소 크기를 log_2N보다 작아도 공개키 인증서 경로구축이 가능하도록 하였다.

ABSTRACT

Mobile Ad-hoc networks have various implementation constraints such as infrastructure-free, no trusted authority, node mobility, and the limited power and small memory of mobile device. And just like wired networks, various security issues such as authentication, confidentiality, integrity, non-repudiation, access control, availability and so on have been arisen in mobile Ad-hoc networks. But we focus on authentication of these security issues because it is quitely affected by the characteristics of networks. In this paper, we propose the authentication protocol that can limit the size of certificate repository as log_2N and assures to make a trusted certificate path from one node to another, adopting the concept of Hamming distance. Particularly, our protocol can construct a trusted certificate path in spite of decreasing or increasing the number of nodes in mobile Ad-hoc network.

Keywords : *Hamming Distance, Authentication, PKI, Certificate*

Ⅰ. 서 론

모바일 Ad-hoc 네트워크에서 소형 네트워크 장치

접수일: 2006년 6월 12일; 채택일: 2006년 8월 11일
† 주저자, sllee@kisa.or.kr
‡ 교신저자, jssong@emerald.yonsei.ac.kr

들은 인프라에 무관하게 무선 주파수를 사용하여 상호통신이 가능한 P2P(peer-to-peer) 네트워크 모델에 따라 동작한다.[1] 최근에 모바일 Ad-hoc 네트워크는 군용 네트워크, PAN(Personal Area Network), 센서 네트워크, 재난관리 네트워크 등에 적용되고 있다.[2][3] 그러나 모바일 Ad-hoc 네트워크는 인프라에 관계없이 동작하고 제3의 신뢰기관 없이 운용되어야 하며 네트워크 크기, 노드 이동성, 기기의 배터리 용량 및 메모리 크기 등에 의하여 구현상 제약을 받고, 네트워크 침입, 정보 가로채기, DOS(Denial of service) 공격 등 보안공격에 영향을 받을 수 있다. 이에 따라 그 동안 모바일 Ad-hoc 네트워크의 특성을 고려하면서 보안 문제를 해결하기 위한 많은 시도가 있었다.[4][5][6][7][10][11]

Luo 등은 각 노드가 비밀 공유키를 소유하고 근접해 있는 노드들이 공동으로 인증서비스를 제공하는 다중 서명 메커니즘 및 임계 치를 이용한 비밀분산방법을[8][9] 통해 공개키 인증기관의 역할을 대신하는 방법을 제안하였다.[4][5][6] Capkun 등은 모바일 Ad-hoc 네트워크에서 제3의 신뢰기관 또는 중앙관리 서버 등을 필요로 하지 않을 뿐만 아니라 하위 노드들에게 특별한 역할을 부여하지 않고도 상호인증이 가능한 완전한 자체 공개키 관리 시스템(fully self-organized public key management system)을 제안하였다.[10][11] 이 시스템은 네트워크 분할 및 중앙 서비스에 관계없이 공개키-개인키 쌍 생성, 인증서 발급 및 검증 등을 수행한다. 또한, 이 논문에서는 지엽적으로 관리할 수 있는 공개키 인증서의 저장소 구축방법을 제안하였고 통신 오버헤드를 줄이면서 인증 메커니즘을 수행하는 알고리즘을 제안하였다.

지금까지의 모바일 Ad-hoc 네트워크에서 보안문제를 해결하기 위한 많은 노력들은 안전성과 신뢰성을 유지하면서 인프라에 관계없이 동작, 네트워크에 대한 유연성 제공, 제3의 신뢰기관 없이 운용, 노드의 이동성 보장, 공개키 인증서 저장소 구축방법 등에 초점을 맞추어 진행되었다. 그러나 모바일 노드의 물리적 제약 중의 하나인 각 노드에서 관리하여야 하는 공개키 인증서의 생성 및 관리 부하를 줄이면서 공개키 인증서 경로가 항상 존재하도록 하는 문제 등에 대해서는 아직까지 연구가 미흡한 부분이 있다.

본 논문에서는 *HD*(Hamming Distance)[12]의 개념을 도입하여 공개키 인증서 저장소의 크기를 log_2N보다 작도록 하면서도 신뢰할 수 있는 노드의 인증서를 바탕으로 다른 노드의 인증서를 검증하기 위해 필요한 인증서 경로구축[13]을 보장할 수 있는 인증프로토콜을 제안한다. 이를 위해 본 논문에서는 *HD* 기반으로 노드들 사이의 인증서 발급, 갱신, 폐지 등을 위한 메커니즘을 제시하고, 인증서 저장소의 크기가 log_2N으로 제한되어도 인증 경로구축이 가능함을 증명하기 위해 인증서 경로구축 알고리즘을 제안하고 성능을 분석한다. 더불어 본 논문에서는 모바일 Ad-hoc 네트워크에서 노드 수의 변화에 관계없이 신뢰할 수 있는 인증서 경로구축이 가능함을 보인다.

본 논문은 다음과 같이 구성된다. 제2장에서는 새로운 인증 프로토콜에 *HD*를 도입하기 위한 개념, 특성 및 정의 등을 설명하고 제3장에서는 제안된 인증 프로토콜을 적용한 공개키 관리 방법을 제시한다. 제4장과 제5장에서는 새로운 인증서 경로구축 알고리즘을 제안하고 시뮬레이션을 통해 성능을 평가한다. 마지막장에서는 본 연구의 결론을 설명하고 향후 연구 방향에 대해 설명한다.

II. 개념 및 정의

본 논문에서는 각 노드의 공개키 인증서 저장소(이하, 저장소) 크기를 줄이고 공개키 인증서 경로구축이 가능하도록 하기 위하여 Hamming Distance(이하, *HD*)의 개념을 도입하였기 때문에, 우선 두 정수사이에 *HD*가 가지는 특성을 설명한다. 위에서 제기된 임의의 두 정수사이의 *HD*는 다음과 같은 특성을 가진다.

(정리 1) $a \in Z_N$, 이고 $R_a = \{x \in Z_N | HD(a,x) = 1\}$ 이면 $|R_a| = \log_2 N$, 여기서, $Z_N = \{0,1,...,N-1\}$, $N = 2^m$, $m > 0$.

(증 명) 임의의 정수 $a \in Z_N$에 대하여 Z_N내에 a와 HD가 "1"인 정수의 개수를 찾는 것은 Z_N내의 정수 중에 "0"과 *HW*(Hamming Weight)가 "1"인 정수의 개수를 찾는 것과 같다. 정수 "0"과 *HW*가 r인 정수의 수는 ${}_mC_r$(combinations)로 계산될 수 있다. 여기서 m은 $\log_2 N$을 의미한다. 따라서 임의의 정수 a에 대하여 *HD*가 "1"인 정수의 수는 ${}_mC_1$ 로 $m(=\log_2 N)$과 동일하다.

또한, 본 논문에서 제안하는 프로토콜에 대한 설명의 편의를 위하여 다음과 같은 정의를 사용한다.

(정의 1) 모바일 Ad-hoc 네트워크내의 임의의 인증서 ID(Identification Number)는 Z_N의 원소이고 네트워크의 최대 크기를 N으로 정의한다. 그러나 모바일 Ad-hoc 네트워크내의 노드들은 유동성을 가지기 때문에 실제 네트워크에 참여하는 노드의 수는 N보다 작을 수 있다. 그러므로 모바일 Ad-hoc 네트워크의 실제 참여하는 노드의 수를 N_r로 정의한다.

(정의 2) 두 노드 a와 b의 공개키 인증서 ID를 각각 $a_{ID}(\in Z_N)$와 $b_{ID}(\in Z_N)$로 정의하고 이들 인증서 ID간의 HD를 $HD(a,b)$ 또는 HD로 정의한다.

(정의 3) 모바일 Ad-hoc 네트워크내의 임의의 노드 a는 인증경로 구축을 위하여 네트워크내의 노드들 중 일부의 공개키 인증서를 자신의 저장소에 관리하게 된다. 이때, 노드 a를 PN(Parent Node)이라 정의하고 PN의 저장소에 저장되어 관리되는 노드들의 집합을 CN(Child Nodes)이라고 정의한다. 그리고 CN 중의 하나를 CN_i라 정의한다. PN과 CN_i의 공개키 인증서사이에 $HD(PN,CN_i)=1$인 관계가 성립하도록 각 노드의 ID를 구성한다.

(정의 4) 노드 a가 노드b를 인증하고자 할 때, 노드 b에서 노드 a까지 인증경로를 $Auth_p(a \rightarrow b)$로 정의한다. 이때, 노드 a를 Tn이라 하고 노드 b를 Vn이라 정의하자. 또한, Tn과 Vn 사이에 인증경로가 찾으려는 시도를 $Auth_p(Tn \rightarrow_{\text{trial}} Vn)$로 표시하고, 만약 인증경로가 존재한다면 $Auth_p(Tn \rightarrow_{\text{success}} Vn)$로 정의한다.

우선, 본 논문에서는 모든 노드를 식별할 수 있도록 Z_N내 임의의 원소를 인증서 ID로 부여하였다. 이때, 인증서 ID는 공개키 인증서(x.509 v3)의 일련번호와 다른 개념이며 인증서(x.509 v3)내의 "Subject Name" 혹은 "Subject Alternative Name" 필드를 이용하여 ID를 관리한다. 인증수단 및 인증서경로의 존재 문제를 해결하기 위하여 모든 노드의 공개키 인증서에 특정 ID(Identification Number)를 부여하고 일부 노드가 존재하지 않을지라도 인증경로 구축이 가능하도록 이들 인증서간에 HD를 이용하여 유기적인 연관성을 가지도록 하였다. 이에 대해서는 제4장과 제5장의 인증서경로검색 알고리즘 및 시뮬레이션을 통하여 설명하도록 하겠다. 또한, 정리 1과 정의 3을 이용하여 하나의 노드가 관리하여야 하는 모바일 Ad-hoc 네트워크내의 노드 수를 $\log_2 N$으로 제한하였다. 이에 대해서는 그림1에서 예를 들어 설명하도록 하겠다. 이러한 개념에 따라 각 노드의 공개키 인증서를 생성, 갱신, 폐지 등을 위한 기본적인 동작을 3장에서 설명하겠다. 본 논문에서 사용한 인증서경로검색 알고리즘을 이용할 때 인증서 경로의 길이는 $2\sqrt{N}$보다 작게 된다. 이에 대해서는 제5장에서 시뮬레이션 결과를 통하여 확인한다. 마지막으로 노드의 이동성으로 인한 모바일 Ad-hoc 네트워크의 유연성을 보장하기 위하여 다음과 같은 방법을 이용하였다.

① 노드들 간에 사전 조율(negotiation)을 통해 현 네트워크에 존재하는 노드의 수(N_r)를 결정하고, 이에 따라 네트워크의 크기 N을 결정한다. 이때, N과 N_r은 $N/2 \leq N_r \leq N-1$의 관계가 성립하도록 한다.

② 네트워크내의 실제 존재하는 노드들을 대상으로 순차적으로 ID를 부여받는다.

③ 노드의 이동으로 인하여 네트워크내의 실제 존재하는 노드의 수(N_r)가 $N/2$ 이하로 되면 최대 네트워크 사이즈를 $N/2$ 으로 변경한다. 이때, $N/2$ 이상의 ID를 가지고 있는 노드는 $N/2$ 이하의 ID로 재 할당받아야 한다. 이 과정은 인증서 갱신 프로세스를 통하여 이루어진다.

④ 만약 네트워크내의 노드의 크기가 N 이상이 되면 해당 노드는 $(N,N+1,....)$ 순으로 ID를 부여하고 해당 ID와 HD가 "1"인 노드들과 공개키 인증서를 교환한다. 물론 네트워크의 크기도 N에서 $2N(=2^{m+1})$로 변경된다.

위에서 설명한 특성들에 대하여 그림1과 그림2를 통하여 간략하게 설명한다. 그림 1은 $N=8$인 정수의 집합에 대하여 HD가 "1"인 정수들 사이의 관계를 나타낸다. 즉, 정수 "001"과 HD가 1인 정수는 ("000", "011", "101") 등 3개($=\log_2 8$) 이다. 이는 "정리1"과 일치한다. 이 때, PN의 ID는 "001"이라면 CN의 ID들은 {"000", "011", "101"} 등 3개가 존재한다. 즉, R_{001}은 {"000", "011", "101"}이 된다. 정리 1과 정의3에 의하여 임의의 PN은 $\log_2 N$ 개의 노드에

노드ID	000	001	010	011	100	101	110	111
000		•	•		•			
001	•			•		•		
010	•			•			•	
011		•	•					•
100	•					•	•	
101		•			•			•
110			•		•			•
111				•		•	•	

그림 1. HD=1인 ID들의 집합

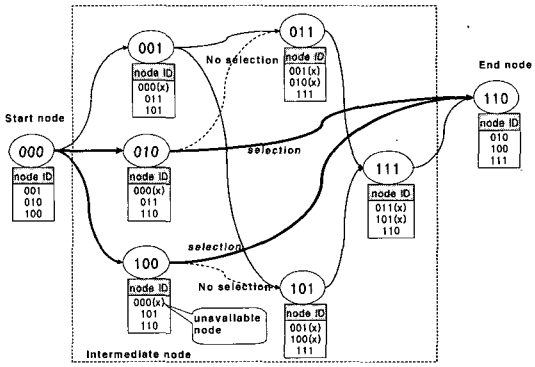

그림 2. 경로구축 개념

해당하는 공개키 인증서만을 자신의 저장소에 관리한다. 즉, 그림 1에서 인증서 *ID*로 "001"을 부여받은 노드는 {"000", "011", "101"}에 해당하는 노드의 공개키 인증서만을 관리한다. 그림 2에서 "unavailable node"는 *CN*의 *ID*들 중에서 인증서 경로상에서 이미 존재했던 *ID*를 의미하고 "selection"은 인증서 경로를 가장 짧게 하는 노드의 선택을 의미한다.

(가 정) 본 논문에서 모바일 Ad-hoc 네트워크내의 모든 노드들이 활성화되어 있다는 가정에서 인증서 생성, 갱신, 폐지 등의 동작 및 인증서 경로 구축 알고리즘을 구현한다.

표 1에서는 모바일 Ad-hoc 네트워크에서의 노드(또는 디바이스, 사용자)간 인증(authentication)을 제공하기 위하여 고려해야할 사항을 설명하고 본 논문에서의 각각의 요구사항을 어떻게 반영하였는지를 제시하였다.

Ⅲ. 공개키 인증서 관리 방법

제2장에서 *HD*를 이용하여 하나의 노드가 Ad-hoc 네트워크상에서 관리하여야 하는 노드의 수를 정하는 방법을 인증서 경로구축 관점에서 설명하였다. 전 장에서 설명하였듯이 Ad-hoc 네트워크상의 한 노드는 자신의 저장소에 $\log_2 N$ 개에 해당하는 노드에 대해서만 공개키 인증서를 생성, 갱신, 폐지, 관리 등을 수행하면 된다. 본 장에서는 전 장에서 설

표 1. 모바일 Ad-hoc 네트워크에서의 인증 요구사항 및 본 논문에서의 방식

인증 요구사항	본 논문의 방식
· 모든 노드의 인증서는 식별 가능하여야 함	· 모든 노드의 인증서에 유일한 식별자(ID) 부여
· 모든 노드들은 제3의 신뢰기관 또는 중앙서버의 도움없이 인증이 가능하여야 함	· HD(Hamming Distance)의 개념을 이용하여 노드 간에 인증수단 제공
· 모바일 노드의 통신범위에 관계없이 모바일 Ad-hoc 내의 모든 노드는 상호간에 항상 인증수단을 제공하여야 함	· HD의 개념을 이용하여 모든 노드 간에 인증경로가 존재하도록 설정
· 하나의 노드가 관리하는 노드 수(혹은 인증서의 수)를 최소화하여야 함	· HD의 개념을 도입하여 관리대상 노드의 수를 $log_2 N$ 으로 설정
· 인증하고자 하는 노드까지의 인증서 경로를 최소화하여야 함	· 시뮬레이션을 통하여 인증서 경로의 길이가 약 $2\sqrt{N}$ 임을 확인
· 임의의 노드는 이동성을 가지기 때문에 네트워크의 유연성을 제공하여야 함	· 노드 인증서의 생성, 갱신, 폐지 제공 · 노드의 수에 따라 네트워크 크기 변경 가능
· 전체적인 네트워크의 안전성과 신뢰성을 보장하여야 한다. 일반적으로 명시된 기밀성, 무결성, 인증, 그리고 부인방지 등의 보안기능을 제공하여야 함	· 공개키 인증서를 도입함으로서 보장

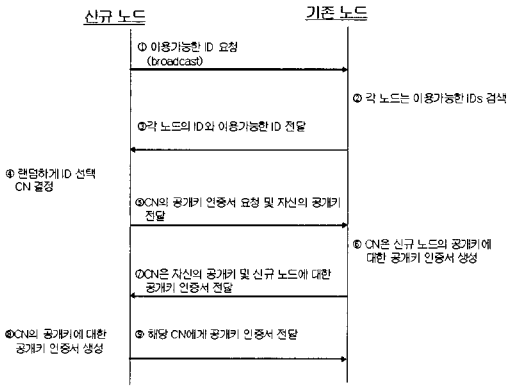

그림 3. 신규 노드와 CN간의 공개키 인증서 교환절차

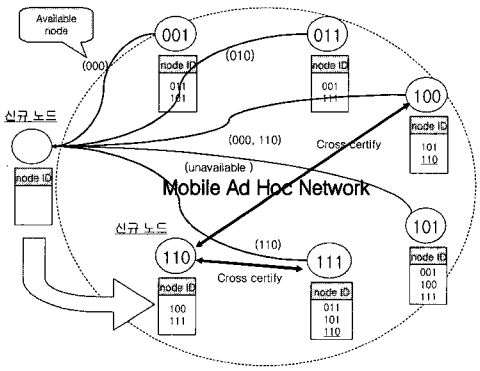

그림 4. 신규 노드에 노드 ID 부여

명한 개념을 토대로 임의의 노드에 대한 공개키 인증서를 생성, 갱신, 폐지하는 방법에 대하여 자세히 설명하고자 한다.

1. 공개키 인증서 획득을 위한 모바일 Ad-hoc 네트워크 초기화

모바일 Ad-hoc 네트워크는 다음 절차를 통하여 초기화된다. ① 모바일 Ad-hoc 네트워크에 참여하는 노드들 간에 사전 조율을 통하여 네트워크 크기 N을 결정한다. ② N에 따라, 각 노드는 Z_N내의 원소 중의 하나를 자신의 ID로 선택한다. 이때, 각 노드는 사전 조율 단계에서 교환한 노드의 IP (Internet Protocol) 어드레스들을 참조하여 IP 어드레스의 순서대로 ID를 선택한다. ③ 각 노드의 ID 할당이 완료되면, 각 노드는 HD=1인 노드와 공개키 인증서를 교환하고 이렇게 교환된 공개키 인증서를 자신의 저장소에 보관하여 관리한다.

2. 신규 노드를 위한 공개키 인증서 생성

모바일 Ad-hoc 네트워크에 신규로 참여하고자 하는 노드는 이미 네트워크에 존재하는 노드들로부터 $ID(\in Z_N)$를 부여받고, 이 ID와 HD가 "1"인 노드들과 공개키 인증서를 교환한다. 다음은 신규로 참여하는 노드에게 ID를 부여하고 공개키 인증서를 교환하는 과정에 대하여 설명한다.

우선, 모바일 Ad-hoc 네트워크에 신규로 참여하고자 하는 노드(신규 노드)는 자신이 Ad-hoc 네트워크에 참여하고자 하는 의사를 Ad-hoc 네트워크내의 모든 노드들에게 전달한다. 이를 수신한 노드들은 자신의 ID와 사용 가능한 ID 정보(만약 사용 가능한 노드 ID가 없는 경우에는 자신의 ID 만 전송)를 신규 노드에 송신한다. 신규 노드는 사용가능한 ID 중의 하나를 랜덤하게 선택한다. 두 번째로 자신의 ID와 HD가 "1"인 ID의 집합(CN)을 결정하여 그들과 공개키 인증서를 교환한다. 이 단계에서 우선 신규 노드는 CN에게 그들의 공개키에 대한 인증서를 요청한다. 그리고 자신의 개인키로 CN의 공개키에 대한 인증서를 발행한다. 그림 3은 신규 노드에 대한 공개키 인증서 생성 절차를 나타낸 것이다.

그림 4는 신규 노드가 네트워크내의 노드들로부터 이용 가능한 ID("000", "010", 110") 정보를 획득하여 자신의 ID로 "110"을 선택함으로써 Ad-hoc 네트워크의 구성원이 되는 것을 나타내었다. 또한, 신규 노드는 자신과 HD가 1인 노드("100", "111")들과 공개키 인증서를 교환하여 자신의 저장소에 공개키 인증서를 보관하게 된다. 물론, 그림 4에서는 공개키 인증서 자체보다는 어느 노드의 공개키 인증서를 관리하느냐가 관심사이기 때문에 공개키 인증서 자체를 표시하지 않고 해당 노드의 ID만을 표시하였다.

3. 공개키 인증서 갱신

Ad-hoc 네트워크내의 공개키 인증서를 갱신하고자 하는 경우는 다음 두 가지 경우로 나누어 고려할 수 있다. 하나는 공개키 인증서의 유효기간이 만료되기 전에 유효기간 연장차원의 갱신이고 다른 하나는 Ad-hoc 네트워크의 크기가 작아지는 경우이다. 첫 번째는 공개키 인증서의 유효기간을 연장하는 경우로 유효기간이 곧 만료될 공개키 인증서를 가지고 있는 노드(PN)는 유효기간 만료 전에 해당하는 CN_i에게

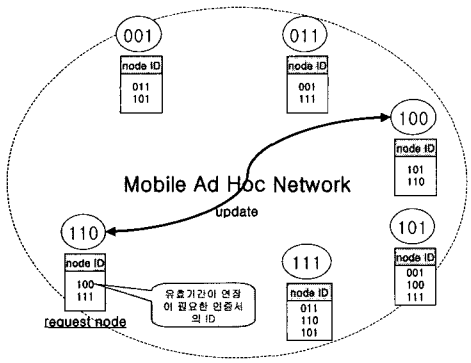

그림 5. 공개키 인증서 갱신 방법(1)

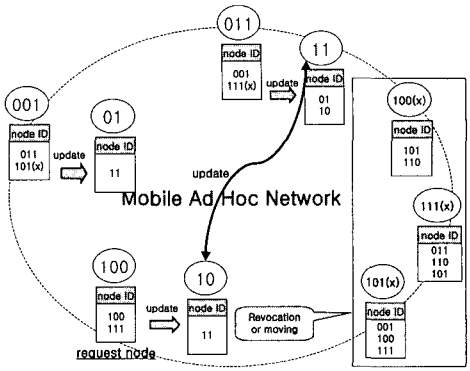

그림 6. 공개키 인증서 갱신 방법(2)

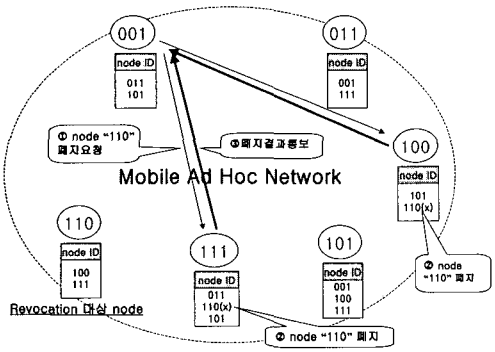

그림 7. 공개키 인증서 폐지

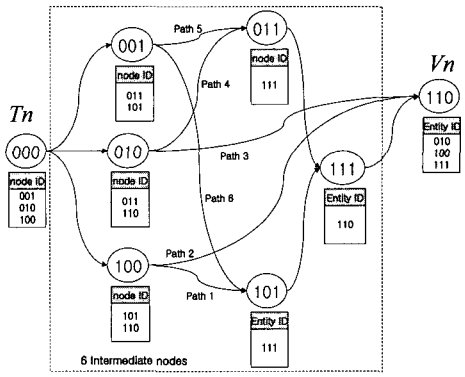

그림 8. T_n과 V_n사이의 신뢰 경로

인증서 갱신을 요청한다. 이를 수신한 CN_i는 PN의 인증서를 갱신하여 PN에게 전송한다. PN은 갱신된 인증서를 자신의 저장소에 보관한다.

그림 5는 노드 "110"의 인증서 갱신을 간략하게 나타내고 있다. 두 번째는 Ad-hoc 네트워크의 크기가 작아지는 경우에 해당하는데 이는 제2장에서 간략히 설명한 것처럼 네트워크의 사이즈가 $N/2$보다 작아지는 경우에 해당하고 이때 갱신 대상 노드는 ID가 $N/2$보다 크거나 같은 경우에 해당한다. 만약, 그림 5에서 일부 노드가 폐지 혹은 다른 Ad-hoc 네트워크로 이동하게 되어 남아 있는 노드가 "001", "011", "100" 일 때, 노드 "100"은 $N/2(=4)$ 보다 작기 때문에 남아 있는 ID("00" 또는 "10") 중에 하나로 갱신하여야 한다. 만약 "10"으로 갱신하는 경우 그림 6과 같이 HD=1인 노드 "11"과 인증서를 갱신한다.

4. 공개키 인증서 폐지

임의의 노드가 Ad-Hoc 네트워크내의 특정 노드의 부정행위 등을 감지하였을 경우 이 노드에 대한 공개키 인증서를 폐지한다. 이때 부정행위를 감지한 노드는 대상 노드의 공개키 인증서를 Ad-hoc 네트워크내의 모든 노드들에게 통지한다. 이를 수신한 노드가 폐지 대상 노드의 공개키 인증서를 관리하는 경우 자신의 저장소에 있는 폐지대상 공개키 인증서를 삭제하거나 폐지목록에 등록하여 폐지절차를 수행한다. 그리고 폐지 결과를 폐지요청 노드에 통지한다.

Ⅳ. 노드 ID와 HD를 이용한 경로구축 알고리즘 제안

1. 인증서 경로구축 알고리즘

신뢰경로란 PN이 CN에게 공개키 인증서를 발급하고 이 CN은 각각의 하위 CN에게 공개키 인증서를 발급하여 모든 노드 간에 계층적으로 신뢰구조를 구성하는 것을 의미한다. 인증서 경로구축 알고리즘은 Tn부터 Vn까지 신뢰경로를 찾는 것을 말한다.

즉, 그림 8에서 보듯이 Tn(000)부터 Vn(110)까지는 6개의 경로가 존재한다. 이들 중 하나의 경로가 Tn부터 Vn까지 구축되고, 경로에 존재하는 인증서에 이상이 없는 것으로 확인된다면 Tn은 Vn을 신뢰하게 된다.

본 논문에서는 이러한 신뢰경로를 구축하기 위하여 각 노드의 저장소 정보(CN)와 제2장에서 설명한 HD의 특성을 이용하여 Tn부터 Vn까지 인증서 경로를 찾는 알고리즘을 그림 9와 같이 제안한다. 우선, 그림 9에서 사용하고 있는 용어들에 대하여 간단히 설명한다. path_nodes는 인증서경로구축과정에서 Tn부터 Vn까지 인증서경로에 존재하는 공개키 인증서들의 집합을 나타낸다. Ad-hoc 네트워크에서 실제 노드의 수는 N_r로 표시되고, empty_nodes는 모바일 Ad-hoc 네트워크에 사용하지 않는 ID들의 집합을 나타낸다. 본 논문에서 제안한 알고리즘은 PN의 CN에 대하여 Vn과 HD를 계산하는 부분과 HD가 "1"이 아닌 경우 PN의 CN_i들 중의 하나를 선택(SN)하여 다음 PN으로 삼는 부분으로 나누어진다. 우선, $HD(PN, Vn)=1$이면 경로구축 알고리즘은 성공적으로 종료된다. 그렇지 않은 경우, PN의 CN에 대하여 $HD(CN_i, Vn)=1$을 만족하는 CN_i가 존재해도 역시 알고리즘은 성공적으로 종료된다. 여기서 $i=0,1,...,\log_2 N-1$. 만약, 모든 CN_i에 대하여 $HD(CN_i, Vn)\neq 1$이라면 CN_i들 중의 하나를 랜덤하게 선택하여 이를 새로운 PN으로 삼고, 이 PN의 CN을 이용하여 $HD(CN_i, Vn)=1$을 만족하는지 확인한다. 이러한 계산을 반복하여 수행하면 Vn까지의 인증서 경로를 찾을 수 있다. 물론 경로를 구축할 수 없는 경우도 존재한다. 또한, Ad-hoc 네트워크의 크기는 $N(=2^m)$이지만 실제 존재하는 노드의 수(N_r)는 N보다 작기 때문에 경로가 존재하지 않거나 경로가 존재함에도 불구하고 경로를 찾지 못하는 경우에 대비하여 다시 경로를 찾는 방법 등을 포함한다. 즉, 그림 9에서 경로가 존재하지 않는 경우를 대비하여 exit_count를 정의하였고, 경로를 찾지 못하는 경우 경로구축 알고리즘을 반복적으로 수행하기 위하여 rep_count와 중지조건 C를 정의하였다.

그림 8에서 Tn은 {"001", "010", "100"} 등 3개의 노드에 공개키 인증서를 발행하였고 이들을 자신의 저장소에 관리한다. 이때, Tn이 PN이 되고 {"001", "010", "100"}이 CN이 된다. 경로 구축을

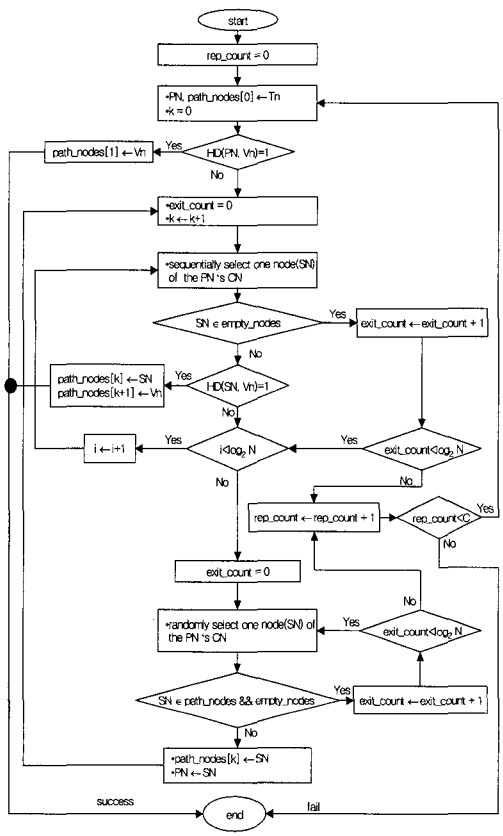

그림 9. 인증서 경로구축 알고리즘

위해 Tn는 자신이 직접 Vn(110)에게 공개키 인증서를 발행하지 않았다면, 매개노드를 이용하여 Vn까지 경로를 찾는다. 이때, 다음 프로세스를 진행하기 위한 PN은 자신의 CN_i들 중에 하나를 랜덤하게 선택한다. 즉, {"001", "010", "100"}의 노드 중에 하나를 다음 PN으로 선택한다. 만약 다음 PN으로 "010"을 선택하게 되면 경로구축(경로 : "000" → "010" → "110")은 완료되지만, 다음 PN로 "001"을 선택한다면 경로는 완료되지 못하고 해당 CN_i("011" 또는 "101")들 중의 하나를 그 다음 PN으로 선택하여야 한다. 이때 PN로 "011"이 선택되면 경로는 "000" → "010" → "011" → "111" → "110" 로 완성된다.

일반적인 경우에 경로구축 알고리즘은 Ad-hoc 네트워크에서의 라우팅 알고리즘과 비슷하다. 본 논문에서는 경로구축 알고리즘이 얼마나 짧은 인증서 경로를 획득하느냐 보다는 얼마나 정확하게 인증서 경로를 찾는가 하는 측면에서 알고리즘 성능을 정의하

였다. Luo 등이 제시한 알고리즘 성능평가방법[4]을 이용하면 본 논문에서 제시된 알고리즘 성능은 다음과 같이 정의된다.

$$P_b(A_{HD}, r, Z_N) = \frac{|\{(Tn, Vn) \in Z_N \times Z_N : Tn \rightarrow_{\text{success}} Vn\}|}{|\{(Tn, Vn) \in Z_N \times Z_N : Tn \rightarrow_{trial} Vn\}|}$$

여기서, A_{HD} : HD를 이용한 인증서 경로구축 알고리즘

r : $(N - N_r)/N$

제5장에서 알고리즘의 성능을 평가한 시뮬레이션 결과를 제시한다.

2. Multi-Hop 문제

모든 노드의 통신범위(power range)는 1 홉(hop)으로 제한되어 있기 때문에 1홉 이상 떨어져 있는 노드 간에 인증서 경로를 구축하기 위해서는 위에서 제안한 인증서경로구축 알고리즘의 개선이 요구된다. 이를 위해 본 논문에서는 인증서경로구축 알고리즘을 인증서경로 검색과 인증서경로 획득 알고리즘으로 나누어 제안한다. 인증서경로검색은 Ad-hoc 네트워크에 존재하는 모든 노드 ID만을 이용하여 Tn이 자신의 ID로부터 Vn까지 인증서 경로를 찾는 것을 말한다. 이를 위하여, 모든 노드들은 자체적으로 Ad-hoc 네트워크내의 노드 ID 목록을 관리한다. 인증서경로검색 알고리즘을 통하여 인증서 경로검색이 완료되면 Tn는 Ad-hoc 네트워크내의 노드들과 실질적인 통신을 통하여 Vn을 검증하기 위해 필요한 인증서를 획득한다. 이 과정을 인증서경로 획득이라 정의한다. 인증서경로검색 알고리즘은 위에서 제안한 인증서 경로구축 알고리즘과 유사하기 때문에 이에 대해서는 별도로 기술하지 않고 다중 홉에서의 인증서 경로획득 알고리즘에 대하여만 설명하고자 한다.

인증서 경로획득 알고리즘의 기본적인 방법은 인증서 경로를 획득하고자 하는 Tn의 통신범위 내에 있는 중개노드(Intermediate Node)들로부터 인증서 경로검색에서 발견된 경로상의 인증서를 우선 획득한다. 만약, 이렇게 획득된 인증서가 완전한 경로를 구성하지 못한다면, 통신범위에 있는 중개노드들 중의 하나를 선택하여 누락된 인증서를 찾아줄 것을 요청한다. 이 때, 중개노드의 통신범위위 내에 가장

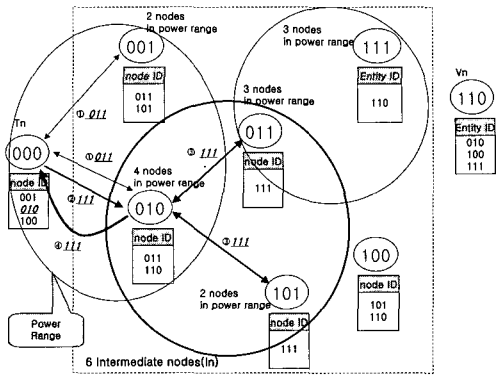

그림 10. Multi-Hop 환경에서의 경로 획득

많은 노드들과 연결되어 있는 노드를 중개노드로 선택한다. 이와 같은 방법을 반복하여 경로상의 인증서를 획득한다. 예를 들어, 그림 10에서 인증서경로검색 알고리즘을 통하여 Tn에 검색한 경로가 "000" → "010" → "011" → "111" → "110"라고 가정할 때, 각 경로상의 인증서 중에서 {"000", "010", "110"}에 해당하는 인증서는 {Tn, Tn의 저장소에 저장된 인증서, Vn}으로 이미 Tn이 획득하였지만 {"011", "111"}에 해당하는 인증서들은 다른 방법으로 획득하여야 한다.. 다행히 "011"은 Tn의 통신범위에 있는 "001" 및 "010"의 저장소에서 관리되고 있기 때문에 쉽게 획득할 수 있지만 "111"은 Tn의 통신범위를 벗어난다. 따라서 다른 노드의 도움을 받아야 한다. 그림에서는 "010"이 자신의 통신범위 내에 가장 많은 노드(4 nodes)를 포함하고 있기 때문에 Tn은 "010"에게 "111"의 인증서를 찾아 줄 것을 요구한다. 이 요청을 받은 중개노드 "010"은 자신의 통신 범위내에 "111"의 인증서가 있는지 확인한다. 다행히, "111" 인증서가 존재하기 때문에 이를 획득하여 Tn에게 전송한다. 이러한 방법을 통하여 인증서 경로상의 인증서가 통신범위를 벗어날지라도 Tn은 Vn의 인증서를 검증하기 위한 경로상의 인증서들을 모두 획득할 수 있게 된다.

V. 제안된 알고리즘을 이용한 경로검색 시뮬레이션

본 시뮬레이션은 그림 9에서 제시된 알고리즘의 완성도를 검증하는 데 목적이 있다. 따라서 본 시뮬레이션은 두 가지 상황을 확인하고자 한다. 하나는 Ad-hoc 네트워크에 참여하는 노드의 수에 따른 인증서 경로의 길이를 분석하는 것이고, 다른 하나는

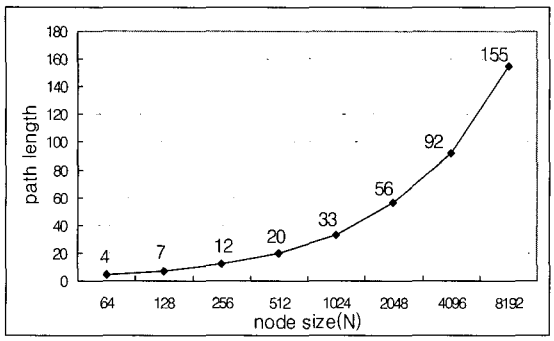

그림 11. 경로 길이 vs. 네트워크 사이즈(N)

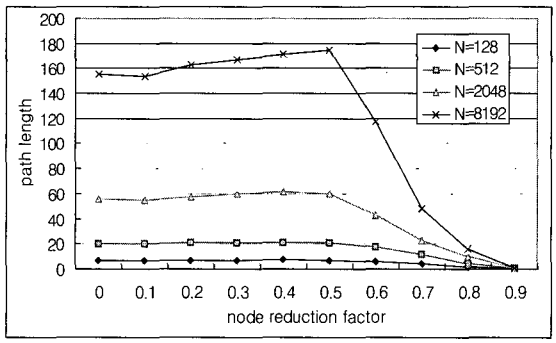

그림 12. 경로 길이 vs. Ad-hoc 네트워크 사이즈(≤N)

Ad-hoc 네트워크의 노드의 수가 $N=2^m$보다 작은 경우 경로의 길이와 경로구축 성공확률을 분석하는 것이다.

그림 11에서는 Ad-hoc 네트워크 크기($N=2^m$, $m=6,7,...,13$)에 대한 경로길이를 나타내고 있다. 그림에서 보는 바와 같이 N=128 까지는 경로길이는 $\log_2 N$ 보다 작은 것으로 나타나고 있지만, N=128보다 큰 경우에는 점차적으로 증가함을 알 수 있다. 따라서 본 논문에서 제시한 프로토콜은 Ad-hoc 네트워크의 크기가 증가할수록 노드의 저장소 크기($\log_2 N$)와 인증서 경로길이 사이의 조율이 필요하다. 비록 N이 증가함에 따라 인증서 경로가 증가할지라도, N=8,192에서 인증경로의 길이는 $2\sqrt{N}$보다 작음을 알 수 있다.

시뮬레이션의 다른 하나는 노드의 이동에 따른 인증서경로구축 가능성 여부이다. 즉, Ad-hoc 네트워크에서의 노드의 감소율 $r(=(N-N_r)/N)$에 따라 경로구축 가능성 여부를 파악하는 것은 제안한 알고리즘의 성능을 파악하는 데 매우 중요한 요소가 된다. 그림 12와 13은 네트워크의 크기 N이 "128, 512, 2048, 8192"이고 노드의 감소율 $r(=0.1, 0.2 \ldots 0.9)$에 따른 경로길이 및 경로구축 성공확률을 나타내고 있다. 그림 12에서 r이 0.4까지는 경로길이가 증가하지만 0.5부터는 감소하는 것으로 나타나고 있다. 또한, 그림 13에서의 경로구축 성공확률(P_b)에서 보는 바와 같이 r이 0.5까지는 신뢰수준 99%에서 경로구축이 97.9%이상 성공하는 것을 볼 수 있다. r이 0.5인 경우는 $N/2=2^{m-1}$ 과 동일하다. 따라서 r이 0.5보다 큰 경우에는 모든 노드의 ID를 $N/2$보다 작도록 조정함으로써 네트워크 사이즈를 $N/2$으로 변경할 수 있다. 이를 통하여 r이 $(N/2-N_r)/(N/2)$ 으로 조정되어 성능을 개선할 수 있다. 노드 ID를 변경하는 것은 제3장의 "공개키 인증서 갱신"에서 설명한 바와 같이 노드 ID가 $0 \leq node\ ID < N/2$ 범위에 존재하도록 공개키 인증서를 갱신한다는 것을 의미하다.

본 시뮬레이션 결과에 따르면 각 노드의 저장소 크기는 $\log_2 N$에서 r이 0.5까지는 97.9% 이상의 확률로 인증서 경로가 존재함을 알 수 있으나, Ad-hoc 네트워크의 크기가 증가함에 따라 인증서 경로길이가 다소 증가함을 알 수 있다. 따라서 향후 과제는 라우팅 알고리즘 기술 등을 이용하여 인증서 경로길이를 줄이는데 역점을 두어야 할 것이다.

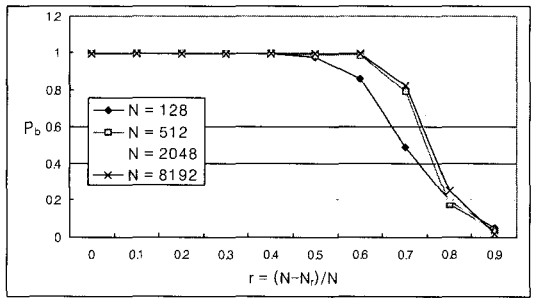

그림 13. 경로구축 성공확률 vs. 노드 감소율r

Ⅵ. 결 론

본 논문은 모바일 Ad-hoc 네트워크상의 노드에서 공개키 인증서의 저장 및 관리에 필요한 저장소 크기를 개선하기 위하여 모든 노드에 ID를 부여하고 각 노드는 자신의 공개키 인증서 ID와 HD가 "1"인 노드에 대해서만 공개키 인증서를 발급, 갱신, 폐지하는 방안을 제안하였다. 그리고 인증경로 구축 시 나

머지 노드를 중개노드로 활용함으로써 각 노드가 자신의 저장소에서 직접 관리하여야 하는 노드 정보의 수를 $\log_2 N$ 으로 줄이고, 인증서 경로의 길이가 N=8,192보다 작을 때 $2\sqrt{N}$보다 작게 할 수 있는 새로운 인증서 경로구축 알고리즘을 제안하여 하였다. 이 인증서 경로구축 알고리즘은 신뢰수준 99%에서 노드감소율이 0.5보다 작은 경우 97.9% 이상의 확률로 인증서 경로를 찾을 수는 있지만, 노드 감소율에 따라 다소 인증서 경로가 길어지는 단점이 있다. 또한, 모바일 Ad-hoc 네트워크내의 모든 노드들이 활성화되어 있고 신뢰할 수 있다는 가정에서 인증서 관리 및 경로구축 알고리즘을 설계하였다. 따라서 향후에 이러한 문제들을 해결하기 위한 노력이 필요할 것이다.

참 고 문 헌

[1] S. Corson and J. Macker, "Mobile Ad hoc Networking(MANET): Routing Protocol Performance issues and Evaluation Considerations", *IETF* RFC2501, Jan. 1999.

[2] K. Fokine, "Key Management in Ad Hoc Networks", LiTH-ISY-EX-3322-2002, 2002.

[3] F. Stajano and R. Anderson, "The Resurrecting Duckling : Security Issues for Ad-hoc Wireless Networks", *Proc. seventh Int'l workshop security protocols*, 1999

[4] H. Luo and S. Lu, "Ubiquitous and Robust Authentication Services for Ad Hoc Wireless Networks", Technical Report 200030, *UCLA Computer Science Department*, Oct. 2000.

[5] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for MANET," *Proc. ninth Int'l conf. Network Protocols*(ICNP), Nov. 2001.

[6] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang, "Self-securing Ad Hoc Wireless Networks", *Seventh IEEE Symp. on Computers and Communications*(ISCC '02), 2002.

[7] A. Weimerskirch and G. Thonet, "A Distributed Light-Weight Authentication Model for Ad-hoc Networks", v. 2288 of LNCS, 2002.

[8] A. Herzberg, S. Jarecki, H. Krawczyk, M. Yung, "proactive Secret Sharing, or: how to cope with perpetual leakage," *Advances in Cryptography - Crypto 95' Proceedings*, LNCS Vol 963, 1995.

[9] Y. Frankel, P. Gemmell, P.-D. MacKenzie, and M. Yung, "Optimal-Resilience Proactive Public-Key Cryptosystems", *IEEE Symp. on Foundations of Computer Science*, 1997.

[10] J.-P. Hubaux, L. Buttyan, and S. Capkun, "The Quest for Security in Mobile Ad Hoc Networks," *Proc. ACM Symp. Mobile Ad Hoc Networking and in Computing(MobiHoc)*, 2001.

[11] S. Capkun, L. Buttyan and J.-P. Hubaux, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks", *IEEE Trans. on mobile computing*, vol. 2, No. 1, Jan./Mar. 2003.

[12] R. Hamming. *Coding and Information Theory*. Prentice-Hall, 1980.

[13] R. Housley, W. Polk, W. Ford, and D. Solo," Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", *IETF* RFC3280, April 2002.

〈著者紹介〉

이 석 래 (SeokLae Lee) 정회원
1992년 2월: 한양대학교 전자통신공학과 졸업
1994년 2월: 한양대학교 전자통신공학과 석사
1994년 1월~1999년 6월: LG전자 주임연구원
1999년 7월~현재: 한국정보보호진흥원 선임연구원
2001년 3월~현재: 연세대학교 컴퓨터과학과 박사과정
〈관심분야〉 정보보호

송 주 석 (JooSeok Song)
1976년 2월: 서울대학교 전기공학과 졸업
1979년 2월: 한국과학원 전기·전자 석사
1988년 2월: Univ. of California at Berkeley 컴퓨터과학 박사
1988년~1989년 Assistant Professor in Naval Postgraduate School
1989년 3월~현재: 연세대학교 컴퓨터과학과 정교수
〈관심분야〉 Information Security, Cryptography, Protocol Engineering