# 클러스터 기반의 이동 Ad Hoc 네트워크에서 안전하고 확장성 있는 키 관리 메커니즘

송지은° 조기환
전북대학교 컴퓨터정보학과
{jeusong°, ghcho}@dcs.chonbuk.ac.kr

# A Secure and Scalable Key Management Mechanism in the Cluster Based Mobile Ad Hoc Networks

Jieun Song° Gihwan Cho
Dept. of Computer and Information, Chonbuk National University

**요 약**

이동 Ad Hoc 네트워크는 무선 채널 특성으로 인해 많은 보안상 위협에 노출될 수 있으므로 인증, 무결성, 기밀성 등의 보안 서비스를 제공하기 위한 키 관리 메커니즘이 요구된다. 그런데 Ad Hoc 네트워크는 중앙 집중적인 관리 노드의 부재, 동적인 위상 변화, 통신 자원의 제약 등의 특성을 고려하여 키 관리 메커니즘을 설계하도록 요구하고 있다. 따라서 본 논문에서는 클러스터 기반의 Ad Hoc 네트워크에서 ID 기반의 임계치 암호화 기법을 사용하여 공개키의 사전 분배 가정을 제거함으로써 토폴로지의 확장성과 유연성을 반영하고, 분산적이며 확장성있는 키분배 서비스가 가능하게 하였다. 또한 클러스터간에 교환되는 메시지의 무결성을 보장하기 위한 클러스터 MAC키 협상 방안, 헤드와 클러스터 내 특정 멤버간의 보안 채널 구축, 통신하고자 하는 클러스터 멤버 호스트간에 안전하게 세션키를 생성, 분배하는 메커니즘 등을 제안하였다.

## 1. 서 론

이동 Ad hoc 네트워크는 공유된 무선 채널을 통해 AP나 기지국과 같은 유선 기반 구조의 도움 없이 각 이동 노드가 호스트이자 라우터 역할을 동시에 수행함으로서 원활하게 통신이 이루어질 수 있도록 하는 구조이다. Ad Hoc 네트워크는 전쟁터나 재난 구조 상황 혹은 교실이나 회의실과 같은 곳에서의 통신 뿐 아니라 WPAN (Wireless Personal Area Network)이나 홈 네트워킹, 센서 네트워크 등의 기반 라우팅 기술로서 더욱 다양한 응용 방안이 연구될 것으로 보인다.

그러나 Ad Hoc 네트워크는 무선 고유의 특성인 잦은 연결단절, 이동 호스트들의 움직임으로 인한 빈번한 위상 변화, 중앙 집중적인 관리 노드 부재 그리고 Ad hoc 통신의 자원 제약 등의 특성을 지니고 있다. 위와 같은 특성으로 인하여 이동 Ad Hoc 네트워크는 무선 트래픽의 도청이나 스푸핑, 중간자 공격 혹은 DoS(Denial of Service) 공격과 같은 다양한 보안상 위협에 노출되기 쉽다. 따라서 이와 같은 공격으로부터 안전한 통신을 보장하기 위해 인증, 기밀성 그리고 무결성 등을 제공할 수 있고 이동 Ad Hoc 네트워크의 특성을 고려한 키 관리 메커니즘이 필요하다.

본 논문은 클러스터 구조를 이용하여 동적으로 클러스터 헤드 노드의 공개키와 개인키, 클러스터 MAC (Message Authentication Code) 키를 생성 및 분배하는 기법을 제시한다. 또한 클러스터 내에서 인증 받지 않은 노드로부터 중간자 공격이나 DoS 공격을 방지하기 위해 데이터 전송노드를 인증하고 정당한 호스트가 기밀성과 무결성을 보장받으며 데이터를 수신할 수 있도록 호스트와 클러스터 헤드간에 보안 채널을 형성하는 메커니즘을 제안하다. 또한 단 대단 대칭기반 데이터 암호화를 통해 두 Ad Hoc 노드가 안전하게 통신할 수 있도록 세션키를 생성 및 분배하는 방안을 제시하였다.

본 논문의 구성은 다음과 같다. 2장에서는 ID 기반의 임계치 암호화 기법과 클러스터 기반의 메시지 인증에 관한 기존 연구들을 살펴본다. 이어 3장에서는 본 논문에서 제안하는 클러스터 기반의 안전하고 경량화 된 키 관리 방안을 기술한다. 마지막으로 4장에서는 결론을 내리고 향후과제를 살펴본다.

## 2. 관련 연구

### 2.1. ID-based threshold cryptography

Khalili[1]는 ID 기반 암호화 기법[2]의 편리성과 효율성, 임계치 암호화 기법[3]의 유연성 및 안전성의 이점을 결합하여 Ad Hoc 네트워크에서 각 노드의 공개키와 개인키를 생성하는 기법을 제안하였다. 대부분의 Ad Hoc 네트워크는 사전에 각 노드가 공개키와 개인키 혹은 공개키에 대한 인증서를 분배받아 보유하고 있거나 대칭적인 비밀키를 소유하고 있다는 가정에서 출발한다. 그러나 이와 같은 가정은 Ad Hoc 네트워크의 확장성을 저해하는 매우 바람직하지 못한 가정이다. 따라서 Khalili는 노드가 네트워크에 참여할 때 공통적으로 분배받는 마스터 공개키와 공개되어 있는 호스트의 ID로 해당 노드의 공개키를 유도하고 임계 개수만큼 주변 노드들로부터 ID에 해당하는 부분 개인키를 얻어내어 완전한 개인키를 획득하는 메커니즘을 제안하였다.

그러나 이 메커니즘은 비밀키를 요청하는 주체를 분명히 인증하지 못하므로 중간자 공격에 매우 취약하다.

### 2.2 클러스터 기반의 메시지 인증 방법

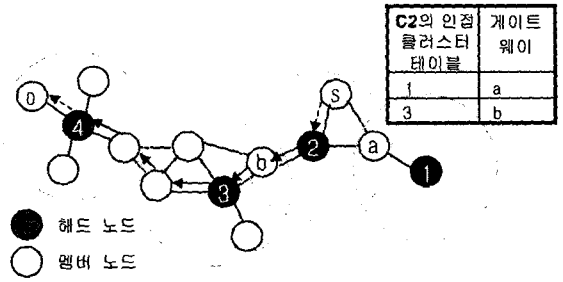
참고문헌 [4]에서는 두 통신주체사이에 키를 공유하기 전에 클러스터 헤드가 자신의 멤버 호스트들을 대신하여

인증을 수행하는 방법이 제안하였다. 이 방법에서는 임의의 두 클러스터 헤드가 각자 상대편 클러스터 헤드의 공개키를 이용하여 상호인증을 수행한다. 따라서 클러스터 헤드의 공개키가 먼저 모든 클러스터 헤드에 분배되어 있어야 한다. 클러스터 헤드 간 인증 후에 대칭키 기반의 세션키가 분배되고, 이는 다시 통신주체인 멤버 호스트에게 분배된다.

이 방법은 클러스터 헤드들이 자신의 공개키를 모든 클러스터 헤드에게 분배해야 하므로 통신 오버헤드가 크다. 또한 두 멤버 호스트간 비밀키인 세션키 분배 시 헤드의 개인키로 암호화되어 해당 노드에 분배함으로써 세션키가 클러스터내의 모든 호스트들에게 노출될 수 있다.

## 3. 제안 방법론

제안 방법은 다음 [그림 1]과 같이 클러스터 구조의 Ad Hoc 네트워크를 기반으로 한다. 클러스터 기반 구조는 경로 발견 패킷의 flooding 범위를 최소화하기 위해 계층화한 것이다. 각 클러스터는 노드들의 가중치(weight) 값에 따라 그룹화 된 것으로 클러스터 멤버와 멤버 노드들을 관리하는 클러스터 헤드 노드로 구성된다. 각 클러스터 헤드는 인접 클러스터 테이블을 통해 근접 클러스터 헤드의 존재를 인지하고 관련된 상태 정보 등을 관리할 수 있다.




[그림 1] Ad Hoc 네트워크에서 클러스터 기반 구조

기본적인 가정 사항은 다음과 같다. 첫째, Ad Hoc 네트워크 통신에 참여하는 모든 노드들은 시스템 마스터 공개키와 부분 마스터 개인키를 부여받는다. 둘째, 각 노드의 ID는 공개되는 값이다. 셋째, 선출된 클러스터 헤드는 다른 클러스터 헤드에 대해 혹은 멤버 노드에 대해 데이터 변조 및 중간자 공격과 같은 악의적인 공격을 수행하지 않는다. 다음은 논문에서 사용되는 표기어이다.

- $Pu_{SM}/Pr_{SM}$ : 시스템 마스터 공개키/개인키
- $Pu_{Ci}/Pr_{Ci}$ : 클러스터 헤드 Ci의 공개키/개인키
- MK : 클러스터 헤드 MAC 키
- SK : 세션키
- K{X | Y..} : X, Y 등 데이터들을 연속적으로 암호화
- MAC(MK |X | Y..) : X, Y 등 데이터들에 키 MK를 이용해 메시지 인증코드 생성

### 3.1. 1단계 : 클러스터 헤드의 공개키/개인키 획득

네트워크에 참여하는 모든 노드들에게는 시스템 마스터 공개키와 시스템 마스터 부분 개인키가 분배된다. 공개키는 시스템 마스터 공개키와 ID를 사용하여 다음과 같이 생성된다.

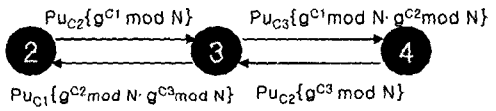
- $Pu_{Ci} = Pu_{SM}$ + head_ID

Ad Hoc 네트워크의 구성 노드들은 다른 노드의 공개키를 사전에 보유하고 있을 필요 없이 위와 같은 방법으로 공개키를 획득하여 통신할 수 있다. 따라서 기존 공개키 기반 구조에서 야기되는 공개키 분배 및 저장에 관한 오버헤드를 감소시킬 수 있다.

클러스터 헤드 노드의 공개키에 대응되는 개인키는 헤드 노드만이 획득할 수 있어야 한다. 다시 말해, "role = 헤드"인 헤드 노드들만이 임계값인 K개의 이웃 헤드 노드들에게 부분 개인키에 대한 요청을 할 수 있어야 한다. 부분 개인키 요청을 받은 이웃 클러스터 헤드 노드들은 자신의 인접 클러스터 테이블을 검사하여 요청 노드가 진짜 헤드 노드인지 확인한 후 요청 노드의 ID에 해당하는 부분 개인키를 전달한다. 일단 헤드로 선정된 노드들은 다른 헤드에 대해 ID 스푸핑이나 중간자 공격 등을 단행하지 않는다는 가정이 성립했을 때 클러스터 헤드 개인키는 멤버 노드나 다른 헤드 노드에 의해 유출되지 않고 해당 노드에 의해 [1]에서 제안한 연산에 의해 안전하게 생성될 수 있다.

### 3.2. 2단계 : 클러스터 헤드간의 MAC키 유도

클러스터간에 전달되는 메시지는 클러스터 헤드노드의 공개키와 개인키에 의해 암호화/복호화 됨으로써 기밀성을 보장받을 수 있다. 그러나 기밀성이 보장된다고 하여서 데이터 무결성이 반드시 보장되는 것은 아니다. 따라서 서로 다른 클러스터를 거쳐 전달되는 메시지의 무결성을 보장하기 위해 클러스터간에 메시지 인증키가 생성 및 공유되어야 한다. 각 클러스터 헤드 노드는 Diffie-Hellman 키 교환 알고리즘을 기반으로 다음 [그림 2]와 같이 클러스터 헤드간에 대칭적인 메시지 인증키 즉, $MK = g^{C2C3C4} \bmod N$를 생성하여 공유한다.



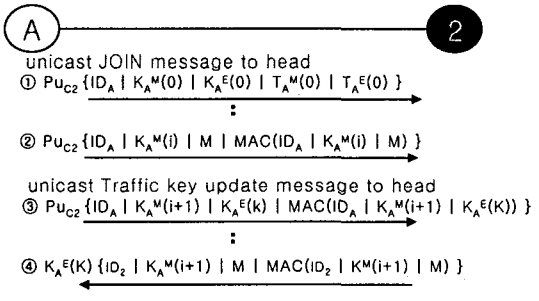

[그림 2] 클러스터 헤드간 MAC 키 유도 방법

만일 새로운 헤드가 선출되거나 기존 헤드 노드의 이동 등으로 위상 변화가 생길 경우 이를 능동적으로 반영하여 다시 새로운 MAC키를 유도한다. 이와 같은 방법은 MAC키를 사전에 공유하지 않고 동적인 분배가 가능하게 함으로써 Ad Hoc 네트워크의 확장성과 유연성을 반영하였다.

### 3.3. 3단계 : 헤드와 클러스터 내 특정 노드 간에 보안 채널 구축

Ad hoc 네트워크에서는 인증 받지 않은 악의적인 노드가 특정 노드에게 메시지를 과도하게 전송하여 대상 노드의 배터리 자원을 고갈시킴으로서 DoS 공격을 시도할 수 있다. 따라서 데이터를 전송하는 노드를 인증하여

인증되지 않은 노드로부터 오는 데이터는 포워딩하지 않고 폐기하도록 하는 메커니즘이 필요하다. 특히 클러스터 구조의 경우 클러스터 헤드 노드에 대한 악의적 공격이 일어날 가능성이 높으므로 멤버 노드에 대한 인증이 반드시 수행되어야 한다. 이를 위해 일방향 해쉬 함수[5]를 이용해 메시지 송신자를 인증하고 헤드 노드가 의도된 특정 노드에게만 안전하게 메시지를 전송하기 위해 검증 가능한 멤버노드의 트래픽 키로 데이터를 암호화해서 전송하는 기법을 제안하였다. 다음 [그림 3]이 일방향 해쉬 함수를 이용해서 클러스터 헤드와 멤버 노드 사이에 보안 채널을 구축하는 기법을 보인 것이다.



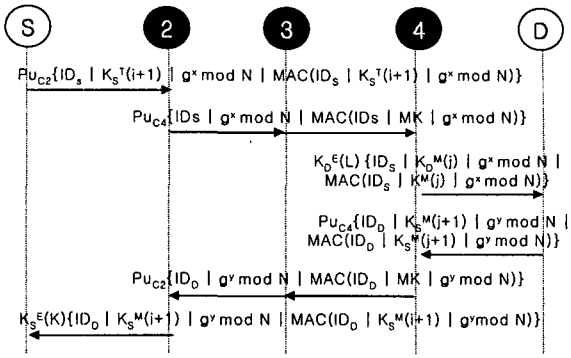

[그림 3] 헤드와 멤버 노드간 보안 채널 구축

네트워크 구성 초기에 클러스터 멤버 노드들은 임의의 난수에 대해 K(0) = F(K(1)) = F(F(F(..F(K(n))))) = $F^n$(K(n))와 같은 방식으로 MAC키와 암호화키에 대해 각각 일방향 함수 체인을 계산하여 ①과 같이 $K_A^M(0)$키와, $K_A^E(0)$ 값을 헤드 노드에 전달한다. 이를 받으면 헤드노드는 $ID_A$에 대해 $K_A^M(0)$, $K_A^E(0)$ 값을 저장해둔다. 이어 ②와 같이 멤버노드 A로부터 $K_A^M(i)$를 이용해 계산된 MAC 값이 전달되었을 때, 헤드 노드는 $F^i(K_A^M(i))$ =? $K_A^M(0)$인지 확인하여 $ID_A$를 인증한다. 이때, 클러스터 멤버 노드로부터 헤드 노드에 전송되는 메시지는 $Pu_{SM}$ + C2_ID 연산을 통해 획득한 공개키로 암호화해서 전달됨으로 기밀성 역시 보장된다.

클러스터 헤드가 전달하는 메시지가 특정 멤버 노드에게만 전달 될 수 있도록 하기 위해 ③과 같이 클러스터 멤버 노드는 주기적으로 헤드노드에게 유효한 트래픽 암호화키를 갱신하여 전송한다. 이 메시지를 받으면 헤드 노드는 ②에서와 마찬가지 방법으로 저장되어있는 $K_A^M(i)$를 이용해 $K_A^M(i+1)$ 키의 정당성을 검증하여 송신자를 인증한다. 또한 전송 받은 트래픽 키에 대해서도 $F^i(K_A^E(K))$ =? $K_A^E(0)$와 같은 확인을 통해 유효성을 검토한다. $K_A^M(i+1)$과 $K_A^E(K)$의 검증 결과가 타당할 경우 이 두 키를 저장하여 두고 이후 클러스터 멤버 노드에게 데이터 전송 시 ④와 같은 형태로 연산하여 전달함으로써 기밀성과 무결성을 보장한다.

### 3.4. 4단계 : 통신하는 멤버 노드간의 세션키 생성

서로 다른 클러스터에 위치해 있는 두 개의 노드가 안전하게 통신할 수 있기 위해서 단 대단 대칭키 기반의 세션키를 생성하고 분배하는 방법이 필요하다. 양자간의 세션키는 위 1·2·3단계에서 생성된 MAC키와 암호화키에 의해 인증, 무결성, 기밀성 등을 보장받으며 Diffie-Hellman 키 교환 알고리즘에 의해 안전하게 구축된다. 그 과정은 다음 [그림 4]와 같다.




[그림 4] 클러스터 멤버 노드간 세션키 생성 메커니즘

## 4. 결론

본 논문에서는 클러스터 구조를 이용하여 동적으로 클러스터 헤드 키와 클러스터 MAC(Message Authentication Code) 키를 생성 및 분배하는 기법을 제안하였다. 또한 클러스터 내에서 데이터 전송노드를 인증하여 헤드에 대한 악의적 노드의 공격을 방지하고 정당한 호스트가 기밀성과 무결성을 보장받으며 데이터를 수신할 수 있도록 호스트와 클러스터 헤드간에 보안 채널을 형성하는 메커니즘을 제안하였다. 그리고 마지막으로 단 대단 대칭키 기반의 데이터 암호화를 통해 두 Ad Hoc 노드가 안전하게 통신할 수 있도록 세션키를 생성 및 분배하는 방안을 제시하였다. 본 논문에서 제안한 키 관리 방안은 Ad Hoc 네트워크의 동적인 위상 변화와 확장성, 중앙 집중적인 관리 노드 부재, 이동 단말의 자원 제약 등의 특성을 반영하여 인증, 기밀성, 무결성 등의 보안 서비스를 제공한다.

향후 Ad Hoc 네트워크의 연산 부하를 줄이기 위해 공개키 연산을 사용하지 않고도 높은 보안 효과를 획득할 수 있는 기법을 고안하고 클러스터 헤드에 부과되는 보안상 책임을 분산시킴으로써 집중적인 공격의 위협을 감소시키는 방안 등도 의미 있는 연구사항이 될 것으로 보인다.

## 5. 참고문헌

[1] A. Khalili, et al., "Toward Secure Key Distribution in Truly Ad-Hoc Networks," *IEEE SAINT'03*, pp. 342-346, Jan. 2003.

[2] D. Boneh, et al., "Identity-Based Encryption from the Weil Pairing," *CRYPTO 2001*, vol. 2139, pp. 213-229, Aug. 2001.

[3] L. Zhou, et al., "Securing Ad Hoc Networks," *IEEE Network Magazine*, 13(6), Nov./Dec. 1999.

[4] L. Venkatraman et al., "A Novel Authentication scheme for Ad hoc Networks," *IEEE WCNC' 2000*, vol. 3, pp. 1268-1273, 2000.

[5] L. Lamport, " Password authentication with insecure communication," ACM, 24(11), pp.770-772, Nov. 1981.