초증가 수열과 Knapsack 알고리즘을 이용한 MANET에서의 은닉 라우팅 프로토콜 설계

천준호[0] 박재성 이상훈 장근원 전문석
숭실대학교 컴퓨터학과 대학원
{choun[0], sestar, accomp, jaques, mjun}@ssu.ac.kr

Design of Covered Routing Protocol using Super Increasing Sequence and Knapsack Algorithm in MANET

Junho Choun[0], Jaesung Park, Sanghun Lee, Kun-Won Jang, Munseog Jun
Department Computing, Soongsil Univ.

요 약

현재까지의 보안 라우팅 프로토콜은 유무선에 관계 없이 페이로드 부분은 암호화가 되더라도 패킷 헤더의 내용이 평문 형태로 무방비하게 노출되며 라우팅 경로가 안전하게 보장되더라도 악의적인 노드에게 경로가 알려지는 것을 차단 할 수 없다. 또한 유선 환경과는 달리 Ad-hoc 네트워크와 같은 무선 상황에서는 전파의 전방향성 때문에 송수신 범위 내에 있는 노드들이 평문 형태의 라우팅 정보 및 송수신 노드의 정보를 수집하는 것을 방지 할 수 없다.

본 논문에서 제안하는 은닉 라우팅 프로토콜은 한쌍의 노드가 비대칭키 암호화 알고리즘을 통해 공유한 초증가 수열을 통해 송수신 노드를 은닉하면서도 정당한 수신 노드만 자신이 수신 노드임을 알 수 있는 기법을 제공함으로서 악의적인 노드가 라우팅 경로에 대한 정보를 수집하는 것을 원천적으로 차단한다.

1. 서 론

네트워크의 가입과 탈퇴가 자유로운 ad-hoc 네트워크의 특성이 사용자의 편의를 증가시키는 만큼 보안에 대한 위협요소가 될 수 있다. 또한 ad-hoc 네크워크의 구성요소가 되는 모바일 기기의 연산능력이 상대적으로 떨어지므로 기존의 암호화 시스템이나 인증 시스템을 그대로 적용하기 어렵다.

지금까지 안전한 ad-hoc 네트워크를 위해 고안된 방법으로는 Secure AODV에[1] 사용되는 hash-chain 기법과 PKI를 이용한 double signed message 기법, TESLA에[2] 사용된 ACK-chain과 같은 방법으로서 암호화 과정의 경량화를 위해 각 노드를 연쇄적으로 구성하여 오류가 생기면 이를 상호검증을 통해 문제가 된 노드를 색출하는 방법이 주로 사용되었다.

그러나 어떤 기법을 사용하더라도 무선 통신이 전방향성이므로 이웃 노드들에게 송수신 주체가 어느 노드인지 알려지는 것이 불가피하며 악의적인 노드의 간섭을 허용하는 최초의 단초가 된다.

본 논문에서는 위와 같은 위협을 제거하기 위해 초증가 수열을 이용하여 암호화 과정 없이 송수신 노드를 은폐하며 동일한 의미의 데이터라도 잦은 변화를 통해 악의적인 노드의 재사용 공격과 정황유추를 통한 정보 수집을 대비한다. 또한 Knapsack 알고리즘을[3] 사용하여 정당한 수신 노드만 자신에게 데이터를 전송하고자 하는 노드를 알게 하는 메커니즘을 설계한다.

2. 관련연구

2.1 Knapsack 암호화 알고리즘

Knapsack 암호화 알고리즘은 공개키 기반 암호화 알고리즘으로서 공개키를 생성하기 위한 초증가 수열과 복호화를 위한 Knapsack 알고리즘을 사용한다.

초증가 수열(super increasing sequence) $S^*=\{d_1,d_2,d_3,\cdots,d_n\}$은 임의의 $j > 1$에 대하여 $d_j > d_1+d_2+\cdots+d_{j-1}$를 만족하는 수열을 말한다. 이렇게 만들어진 초증가 수열은 임의의 항 d_j와 d_k을 더한 값은 d_j와 d_k 외에는 어떤 조합으로도 얻을 수 없다는 특성을 갖는다.

공개키를 생성하기 위해서는 임의의 초증가 수열 S^*와 소수 w와 정수 $n(1<n<w)$을 선택한 후, w를 법으로 한 n의 역원 n^{-1}을 계산한다. 공개키 $S=\{e_1,e_2,e_3,\cdots,e_i\}$는 $e_i = w \times d_i \bmod p$가 된다. 생성된 공개키는 키 서버로 전송

하여 공개하고, 이 공개키로 암호화된 암호문은 Knapsack 알고리즘에 의해 복호화 된다.

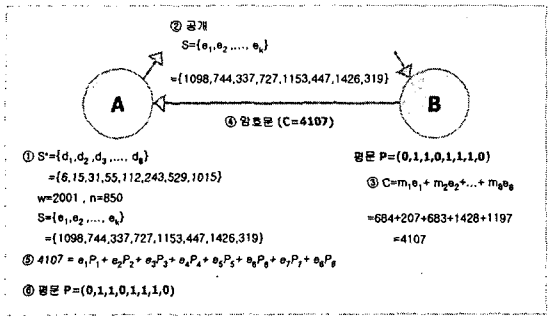

그림 1 Knapsack 알고리즘의 예

3. 은닉 라우팅

3.1 초기 설정

은닉 라우팅을 하고자 하는 노드들은 이미 초증가 수열 $S^*=\{d_1,d_2,d_3,\cdots,d_n\}$를 공유하고 있다는 것을 전제로 한다. 이웃한 한쌍의 노드가 [그림 2]와 같은 형태로 서로의 테이블을 동기화 한다.

Ad-hoc 네트워크 내의 각 노드는 초기화 작업으로서 2.1의 n, w와 별도의 비대칭키 암호화 알고리즘으로 공개/개인키 쌍을 생성한다(P_A/S_A, P_B/S_B).

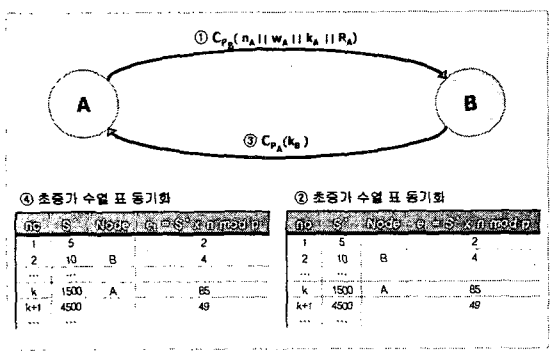

그림 2 초증가 수열 동기화

① 노드 A는 자신의 n_A, w_A와 자기 자신을 지칭하게 될 k번째 항을 지정하는 k_A와 의사난수 R_A을 이웃한 노드 B의 공개키, P_B로 암호화하여 전송한다.

② 노드 B는 자신의 개인키, S_B로 ①을 복호화 하여 n_A, w_A, k_A, R_A을 얻은 후, 이미 공유되어 있는 $S^*=\{d_1,d_2,d_3,\cdots,d_n\}$를 $n_A \times d_n \bmod w_A$하여 초증가 수열을 일치시키고 k_A를 자신의 표에 기록한다.

③ 노드 B는 노드 A가 지정한 것 이외의 항, k_B을 지정하고 노드 A의 공개키, P_A로 암호화 하여 전송한다.

④ 노드 A는 자신의 개인키, S_A로 ③을 복호화한 후, 상호 교환된 정보를 바탕으로 이미 공유되어 있는 $S^*=\{d_1,d_2,d_3,\cdots,d_n\}$를 $n_A \times d_n \bmod w_A$하여 상대와 초증가 수열을 일치시키며 k_B를 자신의 표에 기록한다.

3.2 은닉 라우팅 데이터 송수신

본 논문에서 제안한 은닉 라우팅은 3.1을 바탕으로 서로 초증가 수열을 일치시킨 한쌍의 노드 단위로 이루어진다. 라우팅 정보를 전달하기 위해 노드 A는 [그림 3]의 e_i 중에서 송신자인 자신과 목적지 노드에 해당하는 항을 더한 후, 의사난수 R_A을 더하여 해쉬한다. 만일 동일한 해쉬 결과물이 자주 탐지될 경우 정황 유추를 통해 제 3의 노드가 송수신 노드를 알 수 있으므로 그림 3과 같이 의미 없는 e_i를 하나 이상 더하여 해쉬함으로서 같은 송수신 노드를 의미하지만 매번 다른 해쉬 결과물을 갖게한다.

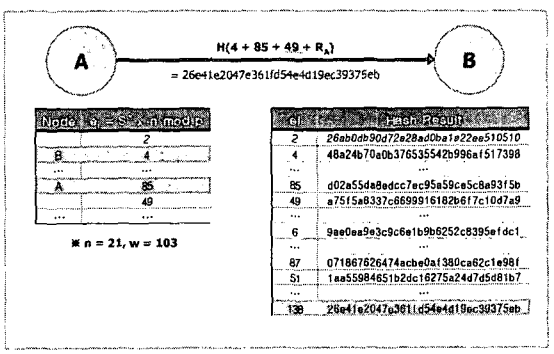

그림 3 은닉 라우팅 데이터 송수신

해당 패킷을 수신한 노드는 자신의 초증가 수열 S^*의 항을 무작위로 더하여 나올 수 있는 모든 조합에 의사난수 R_A을 더한 표를 작성하여 각각을 해쉬한다. 전송 받은 패킷의 헤더에 포함된 해쉬 결과물이 자신의 표에 있다면 송신 노드는 그림 3의 수신측 표를 만들기 위해 사용된 n_A과 w_A값을 공유한 노드가 되므로 1차 검증이 이루어진다. 2차 검증으로는 Knapsack 알고리즘으로 어떤 항들을 더하여 만들어진 수인지 복원하여 송신자인 노드 A와 사전에 약속된 초증가 수열의 항, k_A와 수신자인 노드 B 자신을 지칭하는 항, k_B가 모두 검출되면 해당 데이터는 자신에게 온 것으로 간주한다.

4. 은닉 라우팅의 안전성

4.1 은닉성 보장

그림 3에서 해쉬된 결과물은 평문으로 전달되므로 송신 노드 A의 전파가 도달 가능한 범위내의 모든 노드들이 수신 할 수는 있으나 IP나 Sequence Num.와 같이 노드에 대한 직접적인 정보를 일체 포함하지 않으면 n_A과 w_A, 의사난수 R_A을 공유하지 않는 노드는 수신 노드를 알 수 없으며 무선 통신의 특성상 전방향성을 가지므로 해당 데이터가 어느 노드로부터 왔는지도 알 수 없다.

4.2 사전공격에 대한 강인성

악의적인 노드가 다른 노드 사이의 송수신 정보를 알기 위해서는 아래와 같은 (1),(2),(3)을 곱한 만큼의 경우의 수에 해당하는 목록을 만들고 각각을 해쉬해야 한다. 또한 같은 의미의 해쉬 결과물일지라도 매번 다른 형태를 가지므로, 어떤 항이 어떤 노드를 의미하는지 알기 위해서 최소 (1)의 절반 이상의 패킷을 엿들어야한다. 즉 악의적인 노드가 초고속의 연산과 대용량의 데이터 저장이 가능할지라도 패킷 하나의 전송에 필요한 시간, t의 (2) /2 배 만큼의 지연이 불가피하다.

- 초증가 수열의 항의 개수를 k로 할 때, 무작위로 더하여 조합 가능한 모든 경우의 수를 n으로 곱하고 w로 나머지 연산을 한 수:
  $\{{}_kC_2 + \cdots + {}_kC_k\} \times m \bmod w$ …(1)
- 의사난수 R의 최대범위, R_M …(2)
- 이웃 노드의 수 n이라 할 때 무작위로 두 노드를 선택하는 경우의 수: ${}_kC_2$ …(3)

한편 서로의 공개키로 초증가 수열 표를 동기화 시킨 두 노드는 (2)와 (3)의 경우의 수는 이미 알고 있으므로 계산 과정에서 생략 할 수 있으며 (1)의 경우에도 송신 노드는 초증가 수열로 만들 수 있는 모든 경우의 수를 계산할 필요 없이 단 한번의 해쉬 연산을 거치면 되고 수신 노드는 Birthday Problem을 이용해 전체 목록의 1/7만으로도 50%이상의 적중률을 가질 수 있으며 모든 목록을 유지하더라도 나머지 연산을 하는 w의 최대값 이하의 목록만 해쉬하면 된다. 따라서 악의적인 노드에 비해 정당한 노드는 적은 연산으로도 복원이 가능하다.

그러나 악의적인 노드가 모든 경우의 수를 계산하는데 걸리는 시간을 무한대라 가정할 수는 없으므로 안전성을 위해 본 논문의 은닉 기법을 정적 라우팅 프로토콜에 적용 할 경우에는 주기적인 갱신이 필요하며 동적 라우팅 프로토콜이나 요구기반 라우팅 프로토콜에 응용 할 경우 일정 횟수를 사용한 후에는 초증가 수열을 재동기화 시킴으로서 악의적인 노드의 사전공격을 차단해야 한다.

또한 재사용 공격을 차단하기 위해 이미 사용된 해쉬 결과물은 다시 사용하지 않기 위한 폐기 목록을 만들어야한다.

5. 결론

본 논문에서 제안한 은닉 라우팅 기법은 암호화를 하더라도 노출 될 수밖에 없는 송수신 노드와 패킷 헤더의 정보를 은닉함으로서 악의적인 노드의 간섭을 원천적으로 봉쇄하는데 효과적이다. 그러나 은닉성의 기초가 되는 초증가 수열과 n, p의 크기가 작을 경우 중복이 발생해 송수신 노드를 오인하거나 정상적인 인식이 불가능할 우려가 있으며 지나치게 커질 경우 캐쉬해야 할 정보의 양과 해쉬 연산의 횟수도 그에 비례한다. 또한 노드가 일정 수준 이상 밀집된 경우가 아니라면 송수신 노드를 은폐하는 것이 보안에 큰 영향을 주지 않으며 악의적인 노드의 사전 공격에 취약해진다는 단점을 갖는다.

본 논문의 은닉 기법으로는 평문으로 전달되는 패킷 헤더를 보호할 수는 있지만 기존의 보안 프로토콜과는 달리 패킷의 페이로드를 암호화하여 전달하는 기밀성을 제공하지는 못하므로 본 논문에서 제안한 은닉 라우팅 기법을 단독으로 사용하는 것은 바람직하지 않다.

6. 참고문헌

[1] M.G. Zapata, "Secure Ad Hoc On-Demand Distance Vector Routing," <draft-guerreromanet-saodv-00.txt>.

[2] Adrian Perrig, Ran Canetti, J.D. Tygar, and Dawn Song. "Efficient authentication and signing of multicast streams overlossy channels," In IEEE Symposium on Security and Privacy, May 2000.

[3] Merkle, R., Hellman, M., "Hiding information and signatures in trapdoor knapsacks,"Information Theory, IEEE Transactions on , Volume: 24 Issue: 5 , Sep 1978Page(s): 525 - 530