主題

유비쿼터스 환경과 보안

강원대학교 제어계측공학과 김 형 중

차 례

요 약

유비쿼터스 컴퓨팅 환경에서만 볼 수 있는 보안문제를 가용성, 인증성, 익명성 관점에서 살펴보았다. 이 분야의 보안기술은 기존의 방법과 여러 면에서 다르다. 무선 애드학 네트워크 또는 센서 네트워크의 예에서 보는 것처럼 제한된 계산 성능과 메모리, 전력소모를 최소화하기 위한 통달거리 제한 및 아이들 상태 유지, 온라인 서버가 존재하지 않으므로 인해 발생하는 집중형 기술 구현이 불가능한 한계로 인해 새로운 이슈들이 등장하고 있다. 이런 문제들과 해결책 일부를 소개한다.

1. 서 론

통신과 컴퓨터 기술이 급속히 발전함에 따라 근래 유비쿼터스 컴퓨팅 (ubiquitous computing) 기술이 중요하게 부각되고 있다. 언제 어디서나 누구라도 어떤 대상이든 네트워크를 형성할 수 있게 하는 유비쿼터스 컴퓨팅은 우리의 삶을 크게 변화시키고 있다. 그렇지만 그 핵심 기술 가운데는 보안이 포함되어 있다. 기존의 네트워크에서도 보안이 중요하다. 그렇지만 유비쿼터스 컴퓨팅 환경에서 흔히 볼 수 있는 애드학 (ad-hoc) 네트워크는 동적인 분산시스템을 형성하게 되고 사용자 이동성을 보장해야 하기 때문에 보안관리가 훨씬 어려워진다. 그렇지만 통신 당사자들간의 보안과 프라이버시는 적어도 이전과 동일한 수준으로 유지되어야 한다. 따라서 일부 보안 기술은 이전의 기술로도 충분하지만 유비쿼터스 컴퓨팅 환경에서만 나타나는 새로운 기술도 존재한다. 이 기고문은 주로 후자에 초점을 맞추어 기술하고자 한다.

유비쿼터스 컴퓨팅 환경은 그 영역이 명확하지 않아 약간의 개념상 혼란이 있을 수 있다. 그렇지만 보안 관점에서 동배 (peer-to-peer) 통신,

무선 애드학 네트워크, 센서 네트워크 등이 이 범주에 속한다고 할 수 있다. 일반적으로 계산 성능이 엄청난 컴퓨터를 기반으로 이루어지는 환경을 유비쿼터스 컴퓨팅이라고 부르지 않는다 [9]. 그 대신 문고리, 커튼, 전등, 냉장고, 심지어 구두 등에 설치된 미약한 계산 성능을 지닌 기기들을 유비쿼터스 컴퓨팅 환경이라고 부르는 데는 이견이 없을 것이다. 이런 기기들은 케이블에 의해 연결되지 않을 것이다. 무선으로 연결되는 것은 가능성이 매우 높다. 구두를 케이블에 연결한다면 얼마나 우스꽝스러울지 상상이 될 것이다.

상황에 따라 특정한 목적을 달성하기 위해 긴급히 만들어지는 것이 애드학 네트워크이다. 긴급구난을 위해 출동한 구급대원과 앰뷸런스, 병원의 의사들 사이에 만들어지는 네트워크가 이 범주에 속한다. 애드학 네트워크는 기존의 네트워크와 큰 차이가 있다. 이 네트워크는 짧은 통달거리를 갖는 무선 채널로 주로 구성된다. 또 인프라 지원을 전혀 또는 거의 받지 못한다. 이런 인프라의 부족으로 인해 라우팅, 이름확인 (name resolution), 서비스발견 (service discovery), 보안 등의 문제가 발생한다 [2][11]. 특히 온라인 서버가 없는 것이 보안에 큰 영향을 미친다. 인프라가 없다는 것은 인증기관 (certificate authority) 같은 장치가 없음을 의미한다.

모니터링해야 할 전장에 뿌려지는 스마트 더스트 (smart dust) 장치[6]는 센서 네트워크의 한 예이다. 스마트 더스트는 자율적인 센서 노드, 배터리, 솔라 셀, 송신기, 수신기 등으로 구성된 군용 칩의 일종으로 첩보 수집을 위해 전장에 대량으로 뿌려진다. 이 장치는 매우 소형이고 통신 능력이 미약하므로 아주 근접한 스마트 더스트와 서로 대화하면서 데이터를 릴레이한다. 이들은 성능이 비슷해서 어느 것도 마스터가 될 수 없다. 설혹 하나가 마스터 지위에 있다 해도 그것이 호수에 빠지거나 탱크에 깔려 뭉개지면 마스터가 사라져 통신이 완전히 두절될 수 있다. 그래서 어쩔 수 없이 동배 통신을 하지 않으면 안 된다. 그런데 만일 적군이 동일한 장치들을 뿌린다면 어떻게 될까? 수집한 정보가 믿을만한 것이라고 어떻게 단언할 수 있을까?

이런 애드학 네트워크나 센서 네트워크는 이동성 보장을 위해 링크는 무선으로 구성된다. 이동 호스트들은 고정된 인프라가 없기 때문에 어떤 형태로든 서로 협력해야 한다. 네트워크 토폴로지는 상황에 맞게 다이나믹하게 변화된다. 전력제어는 중요한 설계기준이 된다. 그런데 이동 호스트들에 대한 물리적 보호가 충분하지 못하고 링크 접속이 느슨해서 보안에 매우 취약하게 된다. 여기서도 온라인 서버가 존재하지 않음으로 해서 기존의 보안 모델에서는 볼 수 없는 특별한 상황이 출현하게 된다.

이 기고문은 이런 문제를 다룬다. 우선 가용성 (availability), 인증성 (authenticity), 익명성 (anonymity) 관점에서 새로운 문제점과 기술을 다룬다. 이 밖에도 기밀성 (confidentiality), 무결성 (integrity), 부인봉쇄 (non-repudiation) 등이 함께 고려되어야 하지만 참고문헌 [9]이 도움이 될 것으로 보인다. 기밀성은 수동적 공격으로부터 데이터를 보호하는 것으로 정당한 권한이 부여된 사용자만이 데이터 내용을 파악할 수 있게 하는 것이다. 무결성은 수신된 메시지가 불법적으로 재생된 것인지 또는 전송과정에서 변조되었거나 재구성되었는지 등에 대한 확인을 해준다. 부인봉쇄는 송신자와 수신자 사이에 전송된 메시지에 대한 분쟁을 해결해준다.

2. 가용성 (availability)

가용성이란 사용자가 원할 때 서비스가 제공되는 것을 의미한다. 원하는 시점에서 서비스가

제공되지 않는 것은 사용자에 대한 공격이 될 수 있다. 무선 재밍 (radio jamming) 또는 배터리 소진 (battery exhaustion) 등이 소위 서비스거부 (denial of service) 공격 유형에 속한다.

무선으로 접속해야 하는 상황에서 공격자가 특정 밴드에서 재밍을 실시하면 사용자가 그 밴드를 사용할 수 없게 되므로 서비스거부 상황이 발생한다. 이에 대한 방어기술로 군용통신에서는 확산대역(spread-spectrum) 또는 주파수 옮기기 (frequency hopping) 기법이 활용되고 있다. 따라서 공격자는 더 넓은 주파수 대역에 걸쳐 높은 출력으로 전파를 송출해야 한다. 그렇지만 일반 통신환경에서 이런 공격은 실제 이루어지기 힘들다. 이런 공격은 다수의 사용자들을 불편하게 만들어 결국 국가기관이 나서서 공격자를 처벌하므로 위험을 무릅쓰고 무선공격을 감행할 가능성은 매우 낮다고 할 수 있다.

수명의 제한이 있는 배터리는 유비쿼터스 컴퓨팅 환경에서 풀어야 할 숙제 가운데 하나이다. 공격자가 악의적으로 특정기기에 계속해서 접속을 시도하면 배터리가 급속히 소모될 수 있다. 공격자가 배터리를 닳게 만들면 사용자의 기기는 무용지물이 된다. 이런 것을 수면박탈고문 (sleep depravation torture) [11] 공격이라고 부른다.

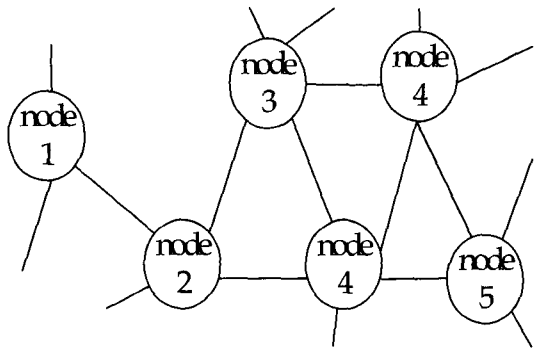

그림 1. 애드학 네트워크의 예

그림 1의 애드학 네트워크에서 인접한 노드들 사이의 협력이 매우 중요하다. 기존의 애드학 라우팅 방법은 수신자의 네트워크 인터페이스가 패킷을 수신하는지 확인할 뿐 그들이 악의적 행동을 하지 않을 것으로 믿는다. 그러나 미지의 노드, 믿을 수 없는 노드들로 구성된 네트워크에서는 이 가정을 그대로 받아들이기 어렵다. 애드학 네트워크는 멀티 홉 라우팅 프로토콜에 의존하며 노드들은 인접한 노드의 패킷을 전송해주어야 제 기능을 발휘할 수 있다. 때로 이기적인 노드를 만드는 정책을 채택하고 패킷 포워딩을 거부함으로써 자신의 배터리를 절약하고 가용한 채널 대역 확보하게 만들 수 있다. 그런데 많은 노드들이 이기적 정책을 채택해 협력을 거부하면 네트워크는 제 기능을 수행할 수 없고 사용자들은 서비스를 받을 수 없게 된다. 그림 1의 노드 2가 협력을 거부한다면 노드 1이 다른 노드에 정보를 보낼 수 있는 방법이 없어진다. 이런 협력거부가 바로 서비스거부로 이어진다. 한편, 악의적인 노드가 노드 2의 자리를 차지하기 위해 이 애드학 네트워크에 잠입할 수 있다. 그리고 그 노드는 패킷의 전송 거부, 엉터리 패킷 투입, 리플레이 (replay) 공격 등을 감행할 수 있다 [7][13]. 따라서 이런 악의적 노드의 참여를 배제하기 위해서는 합당한 인증절차가 요구된다.

현실적으로는 네트워크 상에서 특정 서버를 집중적으로 공격하는 서비스거부 공격은 여전히 문제점으로 남아있다. 악의적인 공격에 의한 서비스거부도 그렇지만 비의도적 서비스거부도 심각한 장애를 초래할 수 있다. 관리자는 누구나 쉽게 서버에 접속할 수 있게 해야 하면서도 악의적 공격에 다운되지 않게 만들어야 하는 서로 양립하기 어려운 목표를 설정하고 있다. 따라서 서비스거부 공격은 언제라도 일어날 수 있다. 가능성은 낮지만 사용자에게 끊임없이 과도한 접속을 시도하는 것도 분산서비스거부 공격에

해당할 수 있다.

3. 인증성 (authenticity)

정보통신체계가 잘 갖추어진 병원에서 만나게 될지도 모르는 체온계와 관련한 시나리오를 하나 살펴보자. 의사나 간호사는 체온계의 출력을 PDA에서 받아 처리할 수 있게 되어있다. 이때 누구나 그 데이터를 받아볼 수 있다면 보안이나 프라이버시 차원에서 큰 문제가 야기될 수 있다. 저명인사의 정보를 전송할 때는 더욱 각별한 권한의 통제가 필요하다. 따라서 어떤 형태로든 접근권한이 부여된 사람들만 데이터를 받아볼 수 있게 할 필요가 있다. 접근제어 목록에는 체온계 데이터를 받을 수 있는 권한이 부여된 기기 또는 의사의 명단이 포함된다. 또 인증서를 가지고 있는 주체는 이를 보이고 체온계 데이터를 읽을 수 있다.

집중형 관리자가 보통 권한을 관리한다. 관리자와의 상호작용은 과다한 부하의 집중으로 인해 시간이 많이 소요되고 효율이 떨어질 수 있다. 접근제어나 인증서 등을 통해 이런 방식을 어떻든 구현할 수 있다. 접근제어 목록이나 인증서 등의 유효기간이나 허용된 권한은 사용자의 편의성에 영향을 미친다. 그래서 적당한 타협이 필요하다. 그런데 보안 관점에서 볼 때 애드학 네트워크는 집중형 온라인 서버를 지니고 있지 않다는 근본적인 문제점을 안고 있다 [2][11].

그래서 "소생하는 오리 새끼 (resurrecting duckling)" [11] 보안정책이 좋은 시사점을 제시한다. 이 논문에서 슬레이브는 "각인된 (imprinted)" 상태와 "각인할 수 있는 (imprintable)" 상태의 두 배타적인 상태를 가진다고 규정한다. 마스터는 그의 슬레이브를 제어하며 최초로 마스터가 슬레이브에게 전송한 공유된 비밀에 의해 그들은 서로 연계된다. 일반적으로 이 비밀은 직접 접촉에 의해 은밀하게 각인된다. 물론 무선이나 유선 채널을 통해서도 이루어질 수 있다.

여기서 사용되는 중요한 개념이 각인이다. 오스트리아의 동물행동학자 Lorenz Konrad는 인공으로 부화된 갓 태어난 새끼 오리들이 자기를 졸졸 따라다니는 것을 발견했다. 그는 이처럼 생후 초기에 나타나는 새끼 오리의 본능적 행동을 각인 (imprinting) 현상이라고 불렀다. 마찬가지로 포장지에서 막 꺼낸 기기에게 비밀키를 보내는 최초의 객체를 기기는 주인으로 인식한다. 이 기기가 바로 새끼 오리라고 보는 것이다.

각인 단계에서 새끼 오리와 어미 사이에는 공유된 비밀이 형성된다. 슬레이브는 마스터에 의해 각인되며, 마스터는 슬레이브를 각인할 수 있다 (그림 2 참조). 각인이 가능한 상태일 때 공유된 비밀이 마스터로부터 슬레이브에게 보내진다. 일단 각인되면 그 기기는 더 이상 신생 기기가 아니며 평생 주인을 섬기는 충실한 기기로 남는다. 이런 관계는 명확하게 마스터-슬레이브 관계에 속한다. 새로 각인할 수 있게 만들려면 일단 오리 새끼는 죽어야 한다. 죽었다 소생하면 다시 오리 새끼는 각인될 수 있는 상태가 되는 것이다.

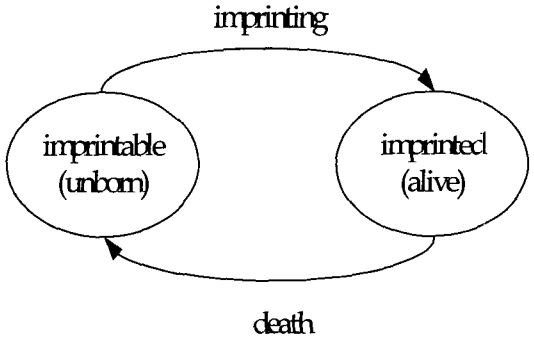

그림 2. 새끼 오리 모델을 위한 상태도

원래의 새끼 오리 보안정책은 동배 상호작용을 지원할 수 있게 확장되었다 [8]. 확장된 정책은 센서 네트워크와 같은 동배 통신을 지원하기 위한 것이다. 동배통신에서 마스터가 존재할 수 없기 때문에 확장된 보안정책에서는 마스터가 유일하지 않아도 된다. 슬레이브는 다른 마스터로부터 각인될 수 있다. 그런데 다른 마스터는 그 시점에서 슬레이브에게 제시할 수 있는 유효한 신용증서를 지니고 있어야 한다. 그 슬레이브는 으뜸 마스터를 모시지만 다른 마스터로부터도 몇몇 명령을 수령할 수 있다. 확장된 새끼 오리 모델에서는 마스터가 새로운 정책을 슬레이브에게 업로드할 수도 있다.

온라인 서버가 존재한다면 서버로부터 커베로스 (Kerberos) 스타일의 티켓을 받아 기기를 인증하거나 공개키를 사용할 경우 서버를 통해 인증서가 믿을 수 있는 것인지 확인 가능하다. 그러나 애드학 네트워크에서는 온라인 서버가 존재하지 않기 때문에 다른 방법으로 이 문제를 풀어야 한다. 현재 제안된 방법으로는 분산신뢰모델 (distributed trust model) [1] [12], 패스워드 기반 키 관리 [2], 새기 오리 모델[8][11], 분산 공개키 관리 [5][14] 등이 있다.

새끼 오리 보안모델은 인증 문제를 안전한 과도연계 (secure transient association) 방식으로 해결한다. 이 방식은 기기와 기기를 마스터-슬레이브 관계로 만든다. 이 방식이 안전한 것은 마스터와 슬레이브가 비밀을 공유하기 때문이다. 연계가 마스터에 의해서만 종결될 수 있기 때문에 과도연계라고 부른다. 마스터만이 비슷한 기기들 가운데서 슬레이브를 식별할 수 있다. 기기를 구입했을 때 최초로 각인시키는 것도 과도상태에 속하지만, 그 기기를 새 주인에 의해 각인이 이루어지게 하려면 마스터는 새끼 오리를 죽였다 다시 살려야 한다. 이때도 잠시 과도상태에 놓인다.

분산신뢰모델[1]은 신뢰관리를 위해 분산처리 방식을 사용하며 신뢰 관련 정보를 교환하기 위해 추천 모델을 사용한다. 이 모델에 따르면 두 객체 사이에서 신뢰관계가 형성되며 그 관계는 일방적 관계로 표현된다 (그림 3 참조). Bob이 Alice를 신뢰하는 정도와 Alice가 Bob을 신뢰하는 정도가 다를 수 있으며, 둘 가운데 하나는 신뢰하는데 다른 하나는 신뢰하지 않을 수도 있다. Alice가 Bob을 신뢰하면 둘 사이에는 직접신뢰관계가 형성된다. Cathy가 Eric을 신뢰하여 다른 객체의 신뢰도에 대한 추천을 할 수 있다면 Cathy와 Alice 사이에는 추천인신뢰관계가 형성된다. 신뢰관계는 각 에이전트의 데이터베이스에만 존재할 뿐이므로 (기기들의 성능제약으로 인해) 신뢰관계에 대한 전체지도는 없다고 가정한다. 객체들은 각자의 정책에 따라 관계의 질적 수준을 판단한다. 즉, 모든 관계는 신뢰관계에 대한 정량적 값을 지닌다. 예를 들어 서로 불신하면 1을, 완전히 신뢰하면 +4의 값을, 그리고 그 중간이면 그에 알맞는 적절한 값을 배정한다 [1]. 또한 신뢰는 절대적이지 않다. 다시 말해 추천된 신뢰정도 수치를 자신의 판단에 따라 변경할 수 있다. 정책은 서로 교환되지 않는다. 이로 인해 객체들은 상대의 정책을 이해할 수 없게 되며 이에 따라 정책이 오용되는 것을 방지할 수 있다. 추천 프로토콜은 신뢰할 수 있는 객체에게 현재 잘 알지 못해서 신뢰할 수 없는 객체 가운데서 믿을만한 노드를 추천해주도록 요청할 수 있게 해준다. 이 프로토콜은 추천의 철회나 변경을 허용한다. 그렇게 함으로써 추천인은 추천한 동일한 기기에 대해 이전과 다른 신뢰의 정도를 표시하거나 중립적인 값을 보내 추천을 철회하게 할 수 있다. 서로의 인증서를 신뢰하기 어려운 상황이라면 인증기관 (certification authority) 기반의 시스템은 비효율적이다. 그래서 분산신뢰모델은 믿을 수 있는 객체들과 분산된 인접 인증기관을

연계한다. 그림 3과 같은 신뢰관계에서 Alice는 Eric을 신뢰할 수 있을지를 이 프로토콜에 의해 결정할 수 있다.

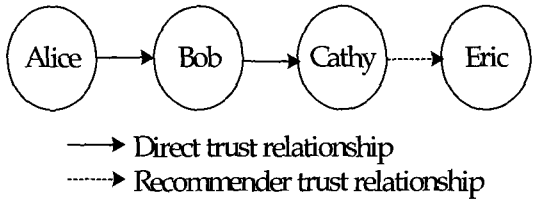

그림 3. 신뢰관계의 예

4. 익명성 (anonymity)

익명성은 위치 익명성 (anonymity of location) 및 데이터 발신지/수신지 익명성 (anonymity of data origin/destination) 관점에 초점이 맞춰지고 있다 [3]. 비록 통화내용의 비밀이 지켜진다고 해도 누가 어디서 누구와 통화했는지가 매우 민감한 문제일 수 있다. 청와대와 관련된 로비 수사에서 통화가 있었는지의 여부, 시기적으로 언제였는지 등이 단서가 되는 것을 흔히 보았을 것이다. 그래서 극단적인 경우 통화 자체의 유무여부도 보호되어야 할 필요가 있다. 위치의 익명성이란 객체의 소재나 행방을 숨기는 것을 의미한다. 데이터 발신지/수신지 익명성은 객체의 행동을 숨겨 보호하는 것이다. 이것은 객체와 객체가 참여한 세션과 주고 받은 메시지를 연관시켜 정보를 추출하려는 것을 방지한다. 마찬가지로 통화한 객체 사이의 관계와 메시지를 연관시켜 정보를 추출하는 것도 방지할 수 있다. 물론 익명성을 어디까지 보호해야 하는지도 중요한 문제가 되고 있다. 개인의 프라이버시 차원에서 익명성을 옹호하는 측과 재난방지나 신변안전을 위해 익명성을 완화해야 한다는 측의 의견이 팽팽하게 대립되고 있다.

전통적인 방법인 유선통신에서 위치 익명성은 그리 심각한 문제가 되지 않았다. 무선통신환경에서는 위치 익명성이 중요해지고 있다. 그럼에도 불구하고 이동성 보장을 위한 프로토콜 (예를 들면, mobile-IP) 자체에서도 아직 이런 문제가 고려되고 있지 않다.

전통적인 보안 프로토콜은 두 객체 사이에서 단계별로 주고 받는 절차를 기술하고 있다. 따라서 기본적으로 이들은 일대일 통신이므로 익명성 보장이 불가능하다. 그래서 기본적으로 일대일 구조는 곤란하고 다수가 참여하는 프로토콜이 되어야 비로소 익명성 보호가 가능하다. 이런 문제를 해결하기 위해 제안된 것이 코카인 경매 프로토콜 (cocaine auction protocol) [10] 개념이다. 이 프로토콜은 전자적으로 조정되는 경매에서 신뢰관계를 다룬다. 이 상황에서 참여자들은 조정자를 무조건적으로 신뢰할 준비가 되어있지 못하다. 이 논문[10]은 신뢰할 수 있는 중재자가 없이도 익명으로 경매에 참여할 수 있게 하며 다양한 공격과 대응방법을 제시하고 있다.

코카인경매 프로토콜이 이루어지는 상황은 그림 4와 같다. 경매에 n명의 객체가 참여하지만 한 사람이 판매자이고 다른 사람들은 입찰자들이다. 입찰자들은 익명으로 참여한다. 기본적인 프로토콜은 매우 간단하며 여러 라운드에 걸쳐 입찰이 시행된다. 라운드 i는 판매자가 그 라운드에서의 경매가 bi를 알린다 (broadcast). 구매자들은 t라는 시간 안에 제안을 할 수 있다 ("예"라고 말함으로써 그 가격에 사겠다는 의사를 표시한다). 누군가 한 사람이 "예"라고 말하자마자 그 구매자 wi가 그 라운드의 당첨자가 된다. 만일 t라는 시간 안에 제안자가 없을 경우 타임아웃으로 경매는 끝나고 이전 당첨자 wi-1이 낙찰자가 된다. 만일 아무도 제안자가 없을 경우이는 라운드 0에서 경매가 종료된다 (그림 5 참조).

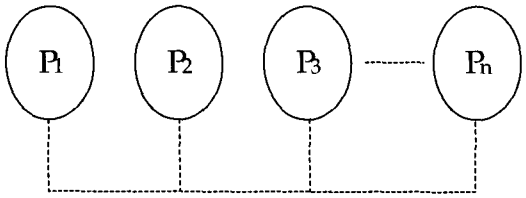

그림 4. 코카인경매 프로토콜이 이루어지는 상황

이 방법을 구현하려면 구체적인 사항에 대한 논의가 추가적으로 필요하다. 우선, 판매자가 낙찰자를 인식할 수 있는 방법이 필요하다. 모든 참가자가 다 "예"라고 말하면 누구나 판매자에게 가서 최종 가격을 주고 실제 낙찰자 대신 물건을 가져갈 수 있다. 모든 참여자는 각 라운드에서의 가격을 알기 때문에 최종 낙찰가격을 알 수 있다. 이것은 바람직한 일이 아니다. 그래서 낙찰자를 식별할 수 있게 각각의 응찰자는 자신의 비밀 식별자를 암호화해서 첨부하게 한다. 즉, 그림 5에서 응찰자의 메시지를

broadcast (yes) broadcast (yes, f(xi))

와 같은 넌스(nounce) 정보를 첨부해 보내면 해결할 수 있다. 물건을 넘겨주기 전 판매자는 낙찰자에게 넌스를 요구할 수 있다. 당연히 그 넌스는 유일해서 다른 사람들은 만들 수 없어야 한다.

둘째, 경매가 끝나고 낙찰자에게 판매자가 은밀한 약속을 하고 싶어할 것이다. 다른 경쟁자들이 시뻘겋게 눈을 부릅뜨고 있는데 현금을 주고받으며 물건을 넘겨주는 것은 위험천만한 일이다 (물론 이런 거래 자체가 불법이고 위험한 것이지만 익명성을 설명하기에 좋아 예로 든 것이다). 한편 물건을 인수할 때까지 낙찰자는 자신의 신분을 판매자에게 노출시키고 싶어 하지 않는다. 낙찰자 신분이 알려질 경우 판매자가 물건 인도를 거부할 수도 있기 때문이다. 이때 낙찰자가 누구인지 모르는 상황에서 판매자가 낙찰자에게만 메시지를 보낼 수 있는 방법이 필요하다.

```
Seller:

set initial bid amount;
bid round: {
  broadcast (bid amount);
  wait for:
    recieve (yes);
      increament bid amount;
      repeat bid round;
    timeout;
      exit bid round;
}
broadcast (commit to winner);
negotiate exchange with winner.
```

```
Bidder:


bid round: {
  wait for:
    recieve (message);
      if message is bid amount?
        broadcast (yes);
      else
        exit bid round;
}
// message is commit to winner
if winner
  negotiate exchange with seller.
```

그림 5. 코카인경매 프로토콜의 골격

어느 프로토콜이나 항상 공격에 노출될 수 있다. 공격 유형이 파악되면 대응책을 마련할 수 있다.이 프로토콜에 대한 공격은 다음과 같은 것이 있다.

1. 최고가 낙찰자에게 물건을 팔지 않는 경우
2. 판매자가 낙찰자가 되는 경우
3. 응찰자가 물건을 구매하지 않는 경우

이상의 공격에 대한 해는 쉽게 만들 수 있다. 문제는 코카인 프로토콜이 쓸모가 있게 되려면 입찰 참여자들이 익명으로 "예"라고 할 수 있는 여건이 마련되어야 한다. 이에 대한 해답의 일부는 "식사하는 암호학자 (dining cryptographers)" 예화[4]가 힌트를 제공해줄 수 있다. 3명의 암호학자가 식탁에 앉아있다 (3명 이상이라도 괜찮다). 웨이터가 식사비는 익명의 인사가 지불했다고 알려준다. 식대를 지불한 사람은 암호학자 가운데 한 명일 수도 있고 암호학자가 아닌 정보기관에서 지불했을 수도 있다. 암호학자는 이웃 자리의 학자가 식대를 냈다면 은근히 기뻐할 것이다. 그렇지만 정보기관에서 냈을지도 모른다고 생각하고 있다. 이들은 다음 프로토콜을 수행함으로써 구가 냈는지에 대한 불확실성을 해소할 수 있다. 이때의 불확실성이란 학자들 가운데 하나가 냈느냐 아니냐에 관한 것이다.

암호학자는 메뉴로 가리고 각자의 동전을 뒤집을 수 있다. 각자는 오른편 암호학자의 동전을 볼 수 있다. 그리고 모든 암호학자는 자신의 동전과 이웃 학자의 동전이 같은 면이었는지 다른 면이었는지를 밝힐 수 있다. 식대를 지불한 학자는 자신이 본 것을 반대로 말한다. 3명의 암호학자가 앉아서 식사를 할 때 발생할 수 있는 경우의 수는 요약하면 그림 6과 같다. 만일 학자 가운데 한 명이 식대를 지불했을 경우 차이(X)의 수가 홀수이고, 모두 동일하다고 답변하거나 차이가 짝수라면 학자들이 식대를 내지 않았음을 의미한다. 이런 방식으로 식대를 지불한 사람은 익명성을 유지하며 식대 지불 사실을 알릴 수 있다. 이런 방식으로 코카인 프로토콜의 익명성을 구현할 수 있다.

5. 결 론

이 기고문에서는 유비쿼터스 컴퓨팅 환경에서만 볼 수 있는 보안문제를 가용성, 인증성, 익명성 관점에서 살펴보았다. 이 분야의 보안기술들은 기존의 방법과 여러 면에서 다르다. 그래서 우선 유비쿼터스 컴퓨팅 환경에 대해 간략히 소개를 했고 구현상의 문제점을 살펴보았다. 무선

Paid not by cryptographer	Paid by cryptographer
T T T O O O	T T T O O X
T T H O X X	T T H O X O
T H H X O X	T H H X O O
H H H O O O	H H H O O X

H: Head T: Tail O: Same X: Different

그림 6. 3명의 학자가 식사할 때 발생할 수 있는 경우의 예

애드학 네트워크 또는 센서 네트워크의 예에서 보는 것처럼 제한된 계산성능과 메모리, 전력소모를 최소화하기 위한 통달거리 제한 또는 시스템을 아이들 상태로 만들어 전력소모를 최소화하려는 노력, 온라인 서버가 존재하지 않아 집중형 기술구현이 불가능하기 때문에 발생하는 새로운 이슈들이 등장하고 있다. 이런 문제들과 해결책으로 새기 오리 모델과 코카인경매 프로토콜에 대해 살펴보았다.

참고문헌

[1] A. Abdul-Rahman, and S. Hailes, "A Distributed Trust Model," ACM New Security Paradigm Workshop, pp. 48-60, 1997.

[2] N. Asokan, and P. Ginzboorg, "Key Agreement in Ad Hoc Networks," Computer Communications, vol. 23, pp. 1627 - 1637, 2000.

[3] U. Carlsen, "Optimal Privacy and Authentication on a Portable Communication System," Operating Systems Review, vol. 28, no. 3, pp. 16-23, 1994.

[4] D. Chaum, "The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability," Journal of Cryptology, vol. 1, no. 1, pp. 65-75, 1988.

[5] J.-P. Hubaux, L. Buttyn, and S. Apkun, "The Quest for Security in Mobile Ad Hoc Networks," ACM Symposium on Mobile Ad Hoc Networking and Computing, 2001.

[6] J. M. Kahn, R. H. Katz, and K. S. Pister, "Next Century Challenges: Mobile Net- working for 'Smart Dust'," International Conference on Mobile Computing and Networking, 1999.

[7] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Mobile Computing and Networking, pp. 255-265, 2000.

[8] F. Stajano, "The Resurrecting Duckling What Next?," Lecture Notes in Computer Science, vol. 2133, pp. 204-214, 2000.

[9] F. Stajano, Security for Ubiquitous Computing, John Wiley and Sons, 2002.

[10] F. Stajano, and R. Anderson, "The Cocaine Auction Protocol: On the Power of Anonymous Broadcast," Lecture Notes in Computer Science, vol. 1768, pp. 434-447, 1999.

[11] F. Stajano, and R. Anderson, "The Resurrecting Duckling: Security Issues in Ad-Hoc Wireless Networks," Lecture Notes in Computer Science, vol. 1796, pp. 172-182, 1999.

[12] A. Weimerskirch, and G. Thonet, "A Distributed Light-Weight Authentication Model for Ad-Hoc Networks," Lecture Notes in Computer Science, vol. 2288, pp. 341-354, 2001.

[13] K. Wrona, "Distributed Security: Ad-Hoc Networks and Beyond," Workshop on Requirements for Mobile Privacy and Security, 2002.

[14] L. Zhou, and Z. H. Haas, "Securing Ad Hoc Networks," IEEE Network Magazine, vol. 13, no. 6, 1999.

김형중
1978년 서울대학교 전기공학과 졸업
1986년 서울대학교 제어계측공학과 석사
1989년 서울대학교 제어계측공학과 박사
2003년 현재 강원대학교 제어계측공학과 교수

<주관심분야> 멀티미디어 보안, 디지털 워터마킹