이동 에이전트의 통신 보안 메카니즘

임동주*, 오창윤*, 배상현*

A Mechanism for Protecting a Mobile Agent's Communication

Dong-Ju Im, Chang-Yun Oh, Sang-Hyun Bae

요 약

이동 에이전트 세계에서는 에이전트들이 악의적인 호스트들로부터 어떻게 보호될 수 있는가를 중심으로 보안측면이 광범위하게 토론되어지고 있다. 본 논문은 에이전트의 소유주의 프로필을 획득하려는, 혹은 그 소유주의 고유의 목적을 방해하려는 경로상의 사이트들에 의해 남용되는 것으로부터 에이전트의 경로 정보를 보호하는 방법들을 토론한다. 우리의 방법들은 방문된 사이트들에게 최소한의 경로 정보를 제공하되 사이트들이 이전 사이트들의 공격을 탐지토록 해준다. 비록 충돌되지 않는 공격하에서, 제시된 모든 방법들이 유사한 보안방법들을 제시하지만, 공격이 탐지될 때 수행과 시점이 다르다.

ABSTRACT

In the world of mobile agents, security aspects are extensively being discussed, with strong emphasis on how agents can be protected against malicious hosts and vice versa. This paper discusses methods for protecting an agent's route information from being misused by sites en route interested in gaining insight into the profile of the agent's owner or in obstructing the owner's original goal. Our methods provide visited sites with just a minimum of route information, but on the other hand allow sites to detect modifying attacks of preceding sites. Though, under noncolluding attacks, all methods presented provide a similar level of protection, the performance and the points of time differ when an attack can be detected.

* 조선대학교 전산통계학과

접수일자 : 1999년 11월 20일

Ⅰ. 서 론

이동 에이전트들은 인터넷 기반 전자시장에서 더욱더 중요시되어져 가고 있다. 많은 시나리오에서 이동 에이전트들은 이동 에이전트들은 정보, 상품, 서비스를 위한 고객, 판매원, 혹은 중재자로 표현되고 있다[1]. 에이전트들은 독립된 프로그램들인데 그들의 소유주를 대신해서 임무를 수행하거나 명령을 받기 위해 경로를 따라다니며 사이트들의 네트워크를 통해 이동한다. 어떠한 보안계획도 없이 방문된 사이트들은 에이전트의 데이터를 찾아내어 에이전트의 소유주에 대한 정보, 예를 들어 서비스, 고객, 서비스 전략, 수집된 데이터 등등을 수집할지도 모른다. 그러한 상황을 피하기 위해 방문된 사이트가 접근 가능한 데이터의 양이 가능한 한 많이 제한되어져야 한다[2].

본 논문에서 우리는 에이전트의 경로 정보, 즉 여행동안에 방문할 사이트들의 주소 리스트에 초점을 맞춘다. 소유주는 에이전트에게 최초의 경로를 제공한다. 에이전트는 여행을 하면서 경로를 통해서 단계별로 작업한다. 경로 보안을 통해서 방문된 어느 사이트도 악의적인 방법으로 경로를 조작할 수 없고 혹은 그 에이전트의 소유주가 접속하고있는 다른 사이트들의 개략적인 정보도 얻을 수 없음을 보장한다. 조작으로부터 에이전트 경로를 보안함으로써 에이전트가 모든 의도된 사이트들을 확실하게 방문한다. 예를 들어 방문된 사이트는 다음에 방문할 사이트를 경로에서 제거할 수 없다. 다른 사이트로부터 에이전트의 경로를 숨김으로써 모든 참여 사이트들의 통신과 상업관계가 다른 사이트들로부터 확실하게 숨겨지고 사이트는 그러한 종류의 정보에 따라 자신의 제안이나 서비스 질을 변경할 수 없다. 우리는 합법적 경로 확장을 통해서 보안 시나리오를 확장한다. 어떤 요청된 서비스를 제공하기 위해서 사이트는 에이전트의 소유주가 예견하지 못한 다른 사이트들과 서로 협력할 필요가 있을지도 모른다. 그러한 상황에서 사이트는 통제되고 안전한 방법으로 원래의 경로를 확장할 수 있다. 다음 예는 우리의 개념에 대한 근거를 제공한다. 소유주가 어느 전자 장치의 가장 좋은 가격 제안을 찾기 위해 기설정된 일련의 가게들을 방문토록 에이전트를 보낸다고 하자. 우리의 방법은 어떠한 가게도 에이전트의 경로에 어느 가게가 포함되고 어느 가게가 포함되지 않았는지에 대한 지식에 근거해서 가격제시를 할 수 없다는 것을 보장한다. 더 나아가 우리의 방법은 어느 경쟁적인 가게도 다른 가게가 혹은 소유주가 알아차리지 못한 채 경로상에서 제거되어질 수 없다는 것을 보장한다. 만약 어느 가게가 일본어로 기술되어진 그 재고품을 가지고 있다면 그 가게는 다른 사이트에게 독일어로 된 기술을 혹은 일본어를 독일어로 번역하도록 요청할지도 모른다. 따라서 그 가게는 에이전트에게 가격을 제안하고 동시에 번역기 사이트를 경로에 포함시킴으로써 원래의 경로를 확장시킨다. 경로확장은 우회 사이트로 제한된다. 즉, 우회 사이트는 다시 확장시킨 사이트로 돌아간다.

Ⅱ. 보안구조

에이전트는 소유주를 대신해서 수행하는 자율적인 프로그램이다. 경로에 따라서 에이전트는 통신 네트워크를 통해서 같이 연결된 사이트들을 방문한다. 한 에이전트가 그 소유주의 홈(home context)에서 생성되고, 송신되고, 마침내 수신되고 평가된다. 방문된 사이트에서 에이전트는 작업환경(working context)에서 실행된다. 비용을 줄이기 위해 에이전트는 일반적으로 경로를 이탈하기 전에는 홈(home context)으로 돌아오지 않는다. 따라서 에이전트가 여행하는 동안에 홈 사이트는 통신 네트워크와 줄곧 연결되어질 필요가 없다. 에이전트를 다음 사이트로 혹은 에이전트의 경로를 확장하기 위해 각 방문된 사이트는 경로의 어떤 부분에 접근할 필요가 있다. 사이트는 최초의 경로로부터 아직 방문안된 사이트를 제거하지 못하도록 되어있고 따라서 에이전트에게 서비스 제공을 배제할 수 없도록 되어 있다. 방문된 사이트가 경로를 확장할 때 추가된 모든 사이트들은 최초의 경로에 있질 않아 예를 들어 전자 현금(cashing)에 대한 에이전트의 접근 권한과 기능을 제한할 수도 있다는 사실을 알아야 한다. 사이트가 경로를 수정할 때 반드시 수정한 사이트하고만 연관이 지워져야 한다. 의심을 받지 않으려면 사이트가 에이전트에 대한 공격을 탐지하자마자

즉시 에이전트를 홈(home context)으로 다시 보내야 한다.

다음에서 제시컨대, 방문된 사이트에게 보이는 경로정보를 최소한으로 줄여서 경로를 악의적인 수정으로부터 보호하는 개념이다. 게다가 그 개념은 융통성이 있어 에이전트의 여행동안에 경로확장을 조종할 수 있다. 두 번째 단계에서 수행문제에 초점을 맞추는데 예를 들어 본 논문에서 결합된 대칭적 공용키 부호화 계획(combined symmetric/public-key encryption schemes)에 의해 제시된 대로 순수한 공용키 부호화 계획으로 대체한다. 그러한 대체는 제시된 방법의 수행능력을 증가시키지만 경로보안에는 아무런 영향을 미치지 않는다. 각 사이트에겐 전임자와 후임자 사이트 주소에 대한 접근만이 주어진다[3].

경로에 부가적으로 에이전트는 여행마다 프로필, 이진코드, 이동 데이터, 그리고 여행 표시기(marker)와 같은 다른 구성요소를 포함한다. 에이전트는 수동적, 읽기 공격, 그리고 에이전트의 기능을 수정하거나 에이전트를 완전히 파괴하기조차 하는 적극적 공격으로부터 보호되어져야 한다[4]. 본 논문은 경로와 무충돌 공격에 대한 경로 보안에 초점을 맞춘다.

Ⅲ. 최초 경로 보안

에이전트가 작업환경 c_i로부터 c_{i+1}로 이동할 때 에이전트의 모든 객체는 부호화되어 수동적 공격으로부터 보호된다. 에이전트를 출발시키기 전에 작업환경(working context)은 에이전트의 부분들의 암호를 해독해서 적극적 혹은 수동적 공격으로부터 노출될 수밖에 없지만 에이전트의 경로는 가능한 한 많이 보호되어야 한다. 보호되지 않은 경로 $r = ip(c_1) \| \ldots \| ip(c_n)$은 에이전트의 여행동안에 방문되어져야 하는 작업환경의 인터넷 주소들 $ip(c_i)$의 연결된 리스트이다. 에이전트의 여행을 중지시키기 위해서 각 사이트는 홈 환경 h의 인터넷 주소 $ip(h)$를 알아야 하는데 그 인터넷 주소가 보호된 경로와는 별도로 알기 쉬운 텍스트형태로 저장되어 있다. 다음 섹션에서 암호와 서명계획의 여러 조합들을 제시하고 그들의 속성들을 비교한다. 모든 제안된 방법에 있어서, 에이전트를 출발시키기 전에 홈 환경은 보호된 경로를 생성한다. 홈 환경은 모든 방문된 작업 환경이 암호해독으로 후임자의 주소와 같은 각 작업환경에 적절한 데이터만을 노출시킬 수 있는 방법으로 공용키 암호를 사용하고 연결 혹은 캡슐화 테크닉을 적용한다. 더 나아가 방문된 작업환경은 수신된 데이터가 에이전트의 홈 환경이 생성한 서명으로 인증됐는지 증명한다. 에이전트를 다음작업환경으로 보내기 전에 실제 작업환경이 자신을 경유하는 경로를 삭제한다.

공용키 암호영역에서 인증된 키들을 제공하는 일반적 문제는 잘 알려져 있다. 본 논문에서 이 문제에 대해 접근하진 않지만 다음에서 공용키 등의 생성, 증명과 분배를 포함하여 안전한 공용키 환경이 있다는 것과 각 작업 환경은 모든 다른 작업환경들의 인증된 공용키들을 알고 있다는 것을 가정한다.

1. 원자식 암호와 서명

홈 환경이 공용키 암호방법 E와 서명방법 S를 사용하여 에이전트의 경로를 원자식 암호와 서명을 한다면 실제 작업 환경은 후임자의 인터넷 주소를 해독하고 경로수정을 확인할 수 있을 뿐이다. 다음에서 우리는 먼저 방정식 (1)에서 보여준대로 홈 환경이 생성한 보호된 경로 r의 구조를 제시한다. 기호 ‖은 데이터의 연결을 의미한다. 보호된 경로를 전체로서 제시하고 경로구조를 단계별로 설명하고 정당화한다.

$$
\begin{aligned}
r = \; & E_{e_1}[ip(c_2), S_h(ip(h), ip(c_1), ip(c_2), t)] \| \\
& \quad \vdots \qquad\qquad\qquad \vdots \\
& E_{e_{n-1}}[ip(c_n), S_h(ip(c_{n-2}), ip(c_{n-1}), ip(c_n), t)] \| \\
& E_{e_n}[EoR,\ S_h(ip(c_{n-1}), ip(c_n), EoR, t)] \|
\end{aligned}
$$

이러한 경우에 경로는 방문되어져야하는 모든 작업환경들의 암호화된 주소와 그에 따른 서명을 포함한다. 암호화될 최소한의 데이터수로 인하여 이러한 변경은 비용측면에서 효과적이다. 작업환경 c_i는 공용키 e_i에 해당하는 사설키 d_i를 사용함으로써 후임자 c_{i+1}의 인터넷 주소 $ip(c_{i+1})$를 해

독한다. 각 사이트는 에이전트의 경로로부터 해독된 주소와 서명을 제거한다. 모든 다른 경로정보는 실제 작업환경으로부터 은닉된다.

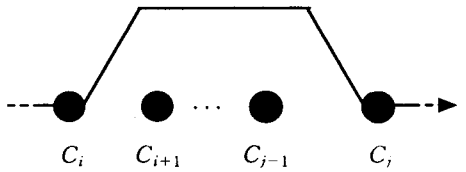
디지털서명의 도움으로 적극적 공격이 탐지될 수 있다. 에이전트의 홈 환경 h가 서명하고 실제 작업환경 c_i에게 제시된 후임자의 주소 $ip(c_{i+1})$의 서명은 c_i를 위한 경로정보가 수정되지 않았다는 것을 증명한다. 인터넷 주소 $ip(c_i)$와 여행 마커(marker) t로 인하여 방문된 작업환경 그 자체가 최초 경로의 일부분임을 보장한다. 여행 마커 t는 각 여행에 대해 유일하다. t는 에이전트의 여행을 유일하게 구분하고 재연(replay)공격을 예방한다. 그렇치않으면 악의적인 작업환경이 실제 에이전트의 완전한 경로를 에이전트의 이전 여행의 복사본 경로로 대체할 수 있다[4].

t의 의미를 예로써 제시하고자 한다. 소유주가 표준경로를 따라 사이트를 방문하기 위해 정기적으로 에이전트를 보낸다고 가정하자. 여행 마커 t가 없으면 공격자는 표준경로의 복사본을 저장할 수 있고 소유주가 표준경로를 바꾸면 새로운 경로를 이전 경로로 대체할 수 있다. 유일한 여행 마커가 없는 시나리오에서는 방문된 사이트는 실제 경로가 이전에 생성된 유효한 경로의 복사본으로 대체됐는지 그리고 실제 경로가 동일한 홈 환경에 의해 인준됐는지 탐지할 수가 없다. 이러한 공격하에서 복사된 경로에 있어서 인준의 증명은 타협된 경로가 정말로 실제 홈 환경에서 생성되었음을 의미한다. 또한 인준을 통해서 방문된 사이트는 실제 경로에서 자신이 포함되어 있다는 것을 믿고 요청된 서비스를 제공해도 좋다는 것을 확신한다. 여행 마커가 없으면 경로를 유효한 복사본으로 대체한 것이 탐지될 수가 있는 시점이 에이전트가 홈 환경에 도착했을 때 예측하지 못한 사이트들로부터 수신된 데이터를 발견했을 때이다. 이제 홈 환경이 해야 할 일은 방문된 사이트에게 경로가 조작되었음을 알리는 것이다. 특히 전자 상거래 환경에서는 그러한 공격들이 모든 수단을 동원해서라도 예방되어져야 한다. 유일한 여행 마커 t, 예를 들어 에이전트의 출발시간으로 모든 방문된 사이트는 최초의 에이전트의 경로를 입증할 수 있다. 사이트가 동일한 홈사이트로부터 동일한 여행 마커를 가진 에이전트들의 방문을 받았을 때 즉시 재연(replay) 공격을 탐지한다[5].

각 인준에 있어서 전임자 주소 $ip(c_{i-1})$의 중요성은 다음 시나리오에서 분명해진다. EoR entry의 도움으로 작업환경 c_n은 자신이 에이전트 경로의 마지막 entry라는 것을 알아챈다. 인준에 있어서 EoR과 t로 인하여 작업환경 c_n은 이러한 데이터들이 h에 의해 생성되었는지 그리고 자신이 정말로 최초 경로에 포함되었는지 확인할 수 있다.

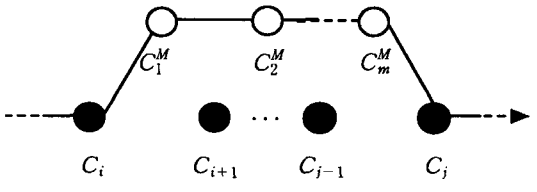
서명은 또한 홈 환경에 의해 암호화되어야 한다. 그렇치 않고 에이전트가 알기 쉬운 텍스트로된 서명을 가지고 다닌다면 t를 알고 있는 공격자는 알려진 주소들의 조합을 배치하고 테스팅함으로써 완전한 경로를 다시 만들 수 있을 수 있을 것이다. 적절한 사이트의 수가 적다면 그러한 공격은 가능하다.

다음에서 에이전트의 기능을 수정하는 적극적 공격에 대한 방법을 토론한다. 악의적인 작업환경 c_i가 에이전트의 경로로부터 entry들을 제거한다면 후임자 c_j가 서명을 확인할 때 entry들의 제거를 탐지할 것이다. 그러한 경우에 c_j가 에이전트를 다시 홈 환경으로 돌려보낸다. 전임자의 주소는 서명 속에 포함되어 있어야 한다. 그렇치 않고 악의적인 작업환경 c_i가 다른 유효한 작업환경 c_j, $i<j\leq n$과 암호화된 경로에서 그의 위치를 예측할 수 있다면 c_i는 에이전트의 경로로부터 이러한 엔트리들을 제거해서 $c_{i+1},\cdots,c_{j-1}$을 뛰어 넘을 수 있다. c_i는 수정된 경로와 더불어 c_j에게 이러한 공격을 알아차리지 못하게 에이전트를 직접 보낼 수 있다.



(그림 1) 주소 제거

물론 그러한 공격의 가능성은 잠재적 작업환경의 수에 달려 있다. 동일한 서비스의 많은 제공자를 가진 에이전트 시스템에서 보다 더 적은 제공자를 가진 에이전트 시스템에서 악의적인 작업환경이 어떤 작업환경을 예측한다는 것이 더 쉬울 수도 있다[6]. (그림 2)와 같이 악의적인 작업환경이 경로로부터 $j-1$ entry들을 제거한 후에 새로운 인터넷 주소 $c_1^M, \cdots, c_m^M$의 m개의 암호문을 삽입하려고 한다면 작업환경 c_1^M 서명확인에 의하여 이러한 수정을 탐지할 것이다. 작업환경 c_1^M이 에이전트 시스템의 규칙들을 준수한다면 에이전트의 여행을 중지시키고 홈 환경으로 다시 돌려보낸다. 악의적인 작업환경이 모든 부가된 환경 $c_1^M, \cdots, c_m^M$들과의 충돌을 형성한다면 아마도 이들 중 어느 환경도 에이전트의 여행을 중지시키지 못할 것이다. 어느 경우든 간에 최초 경로상에 있는 c_j가 서명을 확인할 때 그러한 공격은 감지될 것이다. 악의적인 작업환경은 에이전트의 경로로부터 바로 다음에 계속되는 entry들을 제거하거나 교환하는 대신 훨씬 후에 일어나는 entry들을 제거하거나 교환하는데 관심이 있을 수도 있다. 그러한 상황에서 그러한 공격이 빨라야 언제 감지될 수 있느냐 하는 물음이 야기된다. 공격이 늦게 감지되면 될수록 더욱더 홈 환경 관점에서는 작업환경들은 불성실한 환경으로 의심받게 될지도 모른다. 다음 시나리오를 고려해보자. 악의적인 작업환경 c_i가 c_j를 방문하게 되는 entry $E_{ej}[ip(c_{j+1}), S_h(ip(c_{j-1}), ip(c_j), ip(c_{j+1}), t)]$와 경로로부터 $0<m<n-j$를 가진 따라오는 $m-1$ entry들을 제거한다.



(그림 2) 주소 교환

처음에는 이러한 종류의 수정이 어떠한 관심도 이끌지 못한다. 작업환경 c_j만이 이러한 공격을 탐지할 수 있는데 경로 entry가 c_{j+m}에 의해서 해독될 수 있을 뿐이기 때문이다. c_j가 암호를 해독한 후에 해독문에 유효한 인터넷 주소가 포함되지 않았다는 것을 인식한다. 따라서 c_j는 에이전트의 여행을 중지시키고 다시 홈 환경으로 돌려보낸다. c_i가 에이전트의 경로로부터 마지막 $m=n-j$ entry들을 제거한다면 c_j는 마지막 entry를 해독한 후에 EoR 대신에 인터넷 주소 $ip(c_{j+1})$를 얻는다. c_j는 그러한 공격을 인식할 수 있는 첫 번째 작업환경이다. c_i가 기존의 entry를 새로운 entry로 바꾼다면, 예를 들어 최초경로의 $ip(c_{j+1})$를 $ip(c_1^M)$로 바꾼다면 빨라야 c_j에 이르러서야 이러한 공격이 탐지된다. 요약하자면 원자식 암호와 서명이 모든 제시된 공격으로부터 충분히 에이전트를 보호한다. 불행히도 몇몇의 공격이 공격하는 작업환경과는 상당히 멀리 떨어져 있는 작업환경에 의해 탐지될 수 있을 뿐이고 공격하는 작업환경 그 자체가 유일하게 구분되어질 수가 없다. 그러한 상황에서 에이전트는 공격이 탐지되기 전에 최초의 경로에서 예견되지 않은 사이트들에서 작업을 수행할 수 있다.

2. 원자식 암호와 중첩된 서명

이러한 접근은 첫 번째보다 더 고도의 복잡한 계산을 의미한다. 실제 작업환경 c_i에 대해서는 c_i에 도달하는 모든 비밀 주소와 서명은 암호화되어 있다. 서명은 암호 EoR를 포함해서 후에 발생하는 사이트들이 사용하는 암호들뿐만 아니라 실제 작업환경 주소, 전임자 및 후임자 주소, 그리고 여행 마커 t를 포함한다.

$$
\begin{aligned}
r = \\
&E_{e_1}[ip(c_2), S_h(ip(h), ip(c_1), ip(c_2), E_{e_2}[\ldots], \ldots, E_{e_n}[\ldots], t)] \| \\
&E_{e_2}[ip(c_3), S_h(ip(c_1), ip(c_2), ip(c_3), E_{e_3}[\ldots], \ldots, E_{e_n}[\ldots], t)] \| \\
&\qquad\vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\vdots \\
&E_{e_{n-1}}[ip(c_n), S_h(ip(c_{n-2}), ip(c_{n-1}), ip(c_n), E_{e_n}[\ldots], t)] \| \\
&E_{e_n}[EoR, S_h(ip(c_{n-1}), ip(c_n), EoR, t)] \|
\end{aligned}
$$

방문된 작업환경은 후임자 주소와 구체적 서명을 해독한다. 유일하게 이러한 환경 그 자체가 사

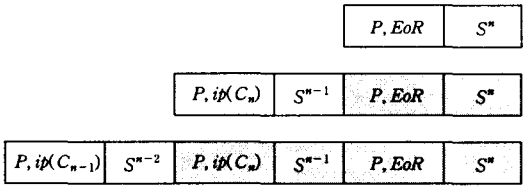
설키로 이러한 데이터들을 해독할 수 있다. 서명을 통해서 작업환경은 경로 entry가 수정되었는지 확인하고 자신의 인터넷 주소에 의존하는 서명을 통해서 자신이 최초경로의 일부분인지 확인한다. 서명이 성공적으로 입증되면 작업환경은 에이전트의 경로로부터 암호화된 후임자의 주소와 서명을 제거하여 에이전트의 다음 이동을 위한 경로를 생성시킨다. 성공적으로 확인된 서명을 통해서 작업환경이 에이전트의 최초 경로에 속한다는 것이 보장된다. 서명에 있는 전임자의 주소를 통해서 (그림 1)에서 제시된 대로 공격이 탐지될 수 있다. 그렇지 않고 악의적인 작업환경 c_i가 에이전트 경로의 나중에 발생하는 entry c_j를 정확히 예측한다면 entry c_j 전에 있는 모든 entry들은 제거될 수 있다. 첫 번째 접근에 있어서처럼 (그림 2)에 따른 공격은 환경 c_1^M에 의해서 탐지될 수 있다. 지금까지 토론된 두 가지 접근방식은 똑같이 기술된 공격에 대하여 에이전트의 경로를 보호하지만 악의적인 작업환경 c_i가 자신과 인접하지 않은 entry들을 제거하거나 교환하는 경우에는 다르다. 두 번째 접근방법에서는 모든 그러한 공격들이 서명입증으로 c_{i+1}에서 항상 직접 탐지될 수 있다. 따라서 첫 번째 접근방법의 몇 가지 약점은 좀더 고도의 복잡한 계산으로 피할 수 있다.

3. 중첩된 암호와 원자식 서명

이러한 접근방법으로 실제 작업환경 c_i는 전임자의 환경으로부터 유일한 암호를 수신한다. 이러한 암호를 해독함으로써 실제 작업환경은 후임자의 주소, 홈 환경이 c_i에게 보낸 서명, 그리고 후임자의 작업환경을 위한 암호문을 받는다(3). 작업환경이 두 개의 첫 요소들을 확인하고 평가한 후 에이전트의 경로로부터 이러한 데이터를 제거하고 남아있는 암호문을 후임자에게 보낸다.

$$r = E_{e_1}[\,ip(c_2), S_h(ip(c_1), ip(c_2), t), E_{e_2}[\,ip(c_3), \ldots, E_{e_{n-1}}[\,ip(c_n), S_h(ip(c_{n-1}), ip(c_n), t), E_{e_n}[EoR, S_h(ip(c_n), EoR, t)]]\ldots]],$$

두 개의 접근방법과는 대조적으로 이러한 접근에 있어서 서명은 전임자의 주소를 포함할 필요가 없다. 이전 접근의 기술(description)로 보아 실제 주소, 전임자의 주소, 그리고 여행마커 t가 서명에 왜 포함되어야 하는지 명백하다. 중첩된 암호 때문에 (그림 1)로부터 공격이 서명 속에 전임자를 포함시키지 않고 조차도 탐지될 수 있다. 악의적인 작업환경 c_i가 자신과 c_j 사이의 모든 사이트들을 제거하려 한다면 c_j는 기술된 프로토콜에 따라 유효한 정보를 제공받을 수가 없다. 자신의 선택에 따라 사이트를 대체해도 마찬가지이다(그림 2). 공격 둘 다 에이전트가 방문하는 다음 작업환경에 의해 탐지된다.



(그림 3) 블록층에서 암호해독후의 경로

어떤 환경이 여러 경우에 있어서 자신과 인접하지 않은 entry들을 수정한다면 그 공격은 즉시 탐지될 수 없고 후에 발생하는 사이트들에 대한 경로 entry들을 파괴할지도 모른다. 이것을 설명하기 위해서는 기반 데이터 구조를 보는 것이 유용하다. 암호화된 메시지가 변할 때 대응되는 해독된 메시지는 완전히 변한다. RSA [7]는 블록 기반 암호계획인데 암호화된 블록의 변화는 대응하는 해독된 블록에만 영향을 준다. 후임자 주소와 서명을 가진 경로 entry는 두 개의 블록을 커버하는데(그림 3), 거기에서 S^i는 c_i에게 도달되는 서명을 의미하고 P는 padding pattern을 의미한다. 회색 셀들은 암호화된 정보, 흰색 셀들은 텍스트 정보를 의미한다. 제일 위쪽 라인의 정보는 c_n에 도달되고 다음 라인의 정보는 c_{n-1}에 도달된다. 악의적인 작업환경 c_i가 라인 맨 끝에 있는 블록들을 수정하면 c_n만이 그러한 공격을 탐지할 수 있다. 악의적인 작업환경 c_i가 이러한 블록들을 삭제한다면 c_{n-1}이 EoR로 인하여 그러한 공격을 탐지할 것이다. 양쪽 경우 모두다 그러한 공격은 아주 늦

게 탐지된다. 요약하자면 어떤 공격들을 탐지하는 데에 전임자의 주소가 필요 없다는 사실이 상당한 의미가 부여될지는 모르지만 이러한 접근의 좀더 고도의 복잡한 계산조차도 공격이 가능한 한 빨리 탐지된다고 보장 못한다[8].

4. 중첩된 암호와 원자식 서명

이러한 접근방법은 복잡한 계산 대신 이전 두 개의 접근방법의 장점들을 포함한다.

$$r = E_{e_1}[ip(c_2), S_h(ip(c_1), ip(c_2), t, E_{e_2}[...]), E_{e_2}[..., E_{e_{n-1}}[ip(c_n), S_h(ip(c_{n-1}), ip(c_n), t, E_{e_n}[...]), E_{e_n}[EoR, S_h(ip(c_n), EoR, t)]]...]],$$

작업환경 c_i는 후임자 주소, 서명, 그리고 다음 작업환경으로 전송될 암호문을 해독한다. 에이전트를 보내기 전에 모든 알기 쉬운 텍스트 문은 경로로부터 제거된다. 서명은 실제 작업환경 주소, 후임자 주소, 여행마커 t, 그리고 완전한 나머지 경로를 포함한다. 공격을 피하기 위해서는 전임자의 주소는 필요치 않다. 다른 접근에서처럼 유사한 고려를 통해서 모든 공격은 가능한 한 빨리 탐지될 수 있다.

5. 요약

모든 제시된 접근방법들은 기술된 공격들에 대한 에이전트의 경로를 보호한다. 중첩된 서명을 가진 (2)과 (4)만이 공격은 가능한 한 빨리 탐지된다는 것을 보장하고 (2)가 복잡한 계산측면에서는 좀더 효율적이다. 비용이 더 저렴하기 때문에 (1)는 짧은 경로에 혹은 덜 민감한 서비스에 사용될지도 모른다. 중첩된 암호를 가진 (3)과 (4)는 악의적인 작업환경들의 충돌로 인해 발생되는 공격에 대한 에이전트의 경로를 보호하기에 적합하다. 홈 환경이 에러 메시지 없이 작업환경 c_n으로부터 에이전트를 수신했다면 에이전트는 최초 경로의 모든 환경들을 방문했고 가능한 한 많은 주소들이 보안 유지되었다는 것이 확실하다. 다음에서 경로가 에이전트의 여행 동안에 합법적으로 확장되는 사례들을 조사한다.

Ⅳ. 결 론

제시된 모든 변형들이 기술된 어느 공격에도 견뎌낼 수 있을지라도 중첩된 서명을 가지고 있는 두 개의 변형만이 가능한 한 빨리 공격을 탐지한다. 원자식 암호를 사용하는 변형은 더 적은 계산시간을 요구한다. 원자식 암호와 서명을 가진 변형조차도 어떤 조건 하에서는 성공적으로 사용될 수 있다. 각 변형은 합법적 경로 확장을 허용한다. 어떤 환경이 에이전트의 최초경로를 확장한다면 에이전트는 확장된 경로의 모든 사이트를 방문한 후에 경로를 확장했던 환경으로 돌아가 최초의 경로상에서 계속 진행한다. 이러한 패턴을 따라가면서 몇몇 수준의 경로 확장을 개발하고 공격에 대비한 에이전트의 경로를 보호할 기술된 방법들을 사용할 수 있다.

Reference

[1] F. Mattern: 'Mobile Agenten', Olden bourg Verlag, 1998, 4, pp. 12-17.

[2] W. Ernestus, D. Ermer, M. Hube, M. Kohntopp, M. Knorr, G. Quiring-Kock, U. Schlager, G.Schulz: 'Datenschutzfre- undliche Technologien', DuD 21, 1997, 12, pp. 709-715.

[3] S. Berkovits, J. Guttman, V. Swarup: 'Authentication for Mobile Agents', in 'Mobile Agents and Security', Proceedings, Springer Verlag, LNCS 1419, 1998, pp. 114-136.

[4] D. Chess: 'Security Issues in Mobile Code Systems', in 'Mobile Agents and Security', Proceedings, Springer Verlag, LNCS 1419, 1998, pp. 1-14.

[5] D. Westhoff: 'AAPI: an Agent Application Programming Interface', Informatikbericht 247-12/ 1998, FernUniversitat Gesamthochschule in Hagen 1998.

[6] G. Vigna: 'Cryptographic Traces for Mobile Agents', in 'Mobile Agents and Security',

Proceedings, Springer Verlag, LNCS 1419, 1998, pp.138-153.

[7] W.M. Farmer, J. Guttman, V. Swarup:'Security for Mobile Agents: Authentication and State Appraisal', in 'Proc. of the 4th European Symp. on Research in Computer Security', Springer Verlag, LNCS 1146, 1996, pp.118-130.

[8] U.G. Wilhelm: 'Cryptographically protected Objects'. Technical report, Ecole Polytechnique Federale de Lausanne, Switzerland, 1997.

[9] T. Sander, C. Tschudin: 'Protecting Mobile Agents Against Malicious Hosts', in 'Mobile Agents and Security', Proceedings, Springer Verlag, Lncs 1419, 1998, pp. 44-60.

[10] F. Hohl: 'Time Limited Blackbox Security: Protecting Mobile Agents from Malicious Hosts', in 'Mobile Agents and Security', Procedings, Springer Verlag, LNCS 1419, 1998, pp. 92-113.

임 동 주(Dong-Ju Im)
1985년 전남대학교 영문학과 졸업(문학사)
1993년 미국뉴욕주립대학교 전산학과(이학석사)
1994년~1999년 대불대학교 근무
1999년 조선대학교 대학원 전산통계학과(이학박사)
※관심분야 : 컴퓨터네트워크, 멀티미디어, 소프트웨어엔지니어링, 데이터베이스 등

오 창 윤(Chang-Yun Oh)
1992년 조선대학교 전산통계학과 졸업(이학사)
1994년 조선대학교 대학원 전산통계학과 (이학석사)
1994년~1996년 (주)아시아자동차 근무
1996년~2000년 조선대학교 대학원 전산통계학과 (이학박사)
※관심분야 : 컴퓨터네트워크, 영상처리, 전문가시스템, 멀티미디어 등

배 상 현(Sang-Hyun Bae)
1982년 조선대학교 전기공학과 (공학사)
1984년 조선대학교 대학원 전기·전자공학과 (공학석사)
1988년 일본 동경도립대학 전자정보통신공학부 (공학박사)
1984년~1985년 일본동경공대 객원연구원
1995년~1996년 일본 NAIST 초빙교수
1999년~현재 조선대학교 자연과학대학 전산통계학과 교수
※관심분야 : 대규모 지식베이스, 인공신경망, 퍼지시스템, GIS, 전문가시스템, 지식처리