

통합 보안 관리자를 이용한 이동 에이전트 이주 성능 향상 연구

유 응 구*

A Study on Improving the Migration Performance of Mobile Agent using Integrated Security Manager

Eung-Gu You *

요 약

이동 에이전트는 분산 응용을 위한 효율적인 통신 패러다임으로 기존의 통신 패러다임보다 높은 유연성과 성능을 제공하는 것으로 알려져 왔다. 그러나 이런 장점에도 불구하고 이동 에이전트가 널리 사용되지 않는 이유는 이동 에이전트가 갖는 보안 취약성 때문이다. 이를 해결하기 위해 이동 에이전트 보안에 대한 연구가 진행되었지만 대부분의 경우 높은 보안성 제공만을 목표로 하여 성능 저하를 유발하였다. 따라서 이동 에이전트가 분산 응용 개발에 적용되기 위해서는 보안성과 성능이 모두 고려되어야 한다. 본 논문에서는 높은 보안성을 제공하는 통합 보안 관리자를 이용하여 이동 에이전트의 이주 성능을 향상시키는 방법을 제안하고 그 성능을 평가하였다. 제안한 방법에서 통합 보안 관리자는 자신이 관리하는 신뢰 도메인 내의 이주 계획을 모두 수행한 후에 작업 결과를 반환하도록 여행 안내 서비스를 제공한다. 제안한 방법이 기존의 방법보다 33 ~ 82% 향상된 성능을 제공함을 알 수 있다.

Abstract

Mobile Agent has been known that it is an efficient communication paradigm for distributed applications and that a mobile agent provides higher flexibility and performance than existing communication paradigms. Despite these benefits, mobile agent is not used widely in the market because it is very vulnerable to a variety of attacks. To be applied to develop distributed applications, a mobile agent paradigm must consider both security and performance. In this paper, we propose and evaluate an improving method of a migration performance for a mobile agent model using ISM(Integrated Security Manager), which provides high-level security services and travel plan guide. In the proposed method, ISM offers the travel plan guide service which replies the accumulated results when a mobile agent has executed all migration plans related the trusted domain to be managed by ISM. Our method improved about 33 ~ 82% of total execution time than the existing method.

▶ Keyword : mobile agent, Integrated Security Manager, migration plan, security

• 제1저자 : 유응구

• 접수일 : 2007. 9.17, 심사일 : 2007. 10.8, 심사완료일 : 2007. 11.10.

* 인덕대학 컴퓨터소프트웨어과

I. 서론

지금까지 분산 응용을 위한 효율적인 통신 기법으로 클라이언트-서버(Client-Server) 모델, REV(Remote Evaluation) 모델, COD(Code on Demand) 모델 및 이동 에이전트 모델 등이 연구되어 왔다. 이중 이동 에이전트 모델은 많은 연구에서 기존의 모델보다 높은 유연성과 우수한 성능을 제공하는 것으로 알려져 왔다[1-3].

그러나 실제 분산 응용 개발에 이동 에이전트가 적용된 사례는 그다지 많지 않은데 그 이유는 대부분의 이동 에이전트 성능 연구가 보안성을 고려하지 않았기 때문이다. 실제 분산 환경은 보안 위협에 노출되어 있기 때문에 중요 자원과 이동 에이전트를 보호하기 위한 방법에 대한 연구는 반드시 수행되어야 한다.

이동 에이전트 보안 연구들을 살펴보면 높은 보안성을 제공하기 위해 고비용의 보안 연산을 사용함으로써 이동 에이전트의 성능이 저하되고, 이동 에이전트 시스템에 다양한 보안 기법이 적용되면서 복잡도가 증가하여 유연성이 감소하는 문제가 발생되고 있다. 따라서 기존의 성능 평가를 통해 입증된 이동 에이전트의 장점을 기대하기 어렵다[4].

따라서 실제 분산 응용 프로그램 개발에 이동 에이전트가 적용되기 위해서는 성능과 보안성이 모두 고려되어야 하며, 보안을 고려한 이동 에이전트 모델의 성능에 관한 연구가 진행되어 왔다[5-6]. 그러나 기존의 성능 연구에서는 이동 에이전트가 수행한 결과의 누적으로 인한 문제가 고려되지 않았지만[7-10] 보안을 고려한 이동 에이전트 성능 모델에서는 누적을 고려하지 않고 있기 때문에 일반적으로 누적되는 분산 응용에는 적용하기 어렵다는 문제가 있다.

본 논문에서는 이동 에이전트의 누적으로 인한 성능 저하 문제를 해결하기 위해 누적을 고려한 이주 계획 수정 기능을 제공하는 통합 보안 관리를 이용한 이동 에이전트 모델을 제안하고 성능을 평가하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구로 보안을 고려한 이동 에이전트 성능 연구와 누적 문제 해결을 위한 이주 방법 연구를 살펴본다. 3장에서는 누적된 결과를 반환하도록 이주 계획을 수정하는 통합 보안 관리자 기반 이주 방법을 살펴본다. 4장에서는 제안한 이주 방법의 성능을 평가해본다. 5장에서는 본 연구를 통해 얻게 된 결론과 향후 연구를 살펴본다.

II. 관련 연구

2.1 보안을 고려한 이동 에이전트 성능 연구

보안 서비스를 고려한 이동 에이전트 모델의 성능에 대한 연구가 발표되었고, 보안 서비스를 제공하는 분산 환경에서도 이동 에이전트 방법이 기존의 RPC 방법보다 우수한 성능을 제공한다는 것을 입증하였다[5]. 그러나 제안한 보안 서비스가 중앙집중형이기 때문에 대규모 분산 환경에 적용하기 어렵다는 단점이 있다.

〈그림 1〉은 중앙집중형 보안 서비스 문제를 해결하기 위해 제안된 TDGM을 이용한 이동 에이전트 보안 모델을 나타낸다.

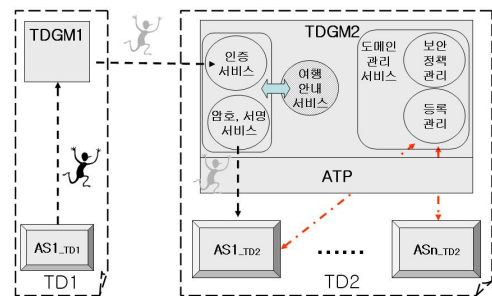


그림 1. TDGM을 이용한 이동 에이전트 보안 모델
Fig 1. Mobile Agent Security Model Using TDGM

TDGM을 이용한 이동 에이전트 보안 모델은 신뢰 도메인 중심으로 이동 에이전트가 이주할 수 있도록 이주 계획을 수정하여 상호 인증을 최소화함으로써 기존의 방법과 동일한 수준의 보안성을 제공하면서도 성능 면에서 32 ~ 53%의 향상됨을 평가를 통해 입증하였다[6].

그러나 기존의 성능 연구와는 달리 이동 에이전트가 방문한 호스트에서 수행한 작업 결과가 누적되지 않고 동일한 크기로 처리되고 있기 때문에 순위를 고려한 분산 검색에는 적용할 수 있지만, 일반적인 분산 정보 검색이나 분산 시스템 관리 등과 같이 작업 결과가 누적되는 분산 응용에는 적용하기 어렵다는 문제가 있다.

2.2 누적 문제 해결을 위한 이주 방법 연구

이동 에이전트의 성능은 인터넷 정보 검색, 네트워크 및 분산 시스템 관리 분야를 중심으로 최근까지 연구되어 왔으

며, 기존의 연구들을 살펴보면 크게 세 가지로 분류된다. 첫째는 이동 에이전트와 클라이언트-서버 모델을 비교하는 연구들이고, 둘째는 이동 에이전트와 REV나 COD 모델을 비교하는 연구들이며, 셋째는 이동 에이전트와 이동 에이전트를 비교하는 연구들이다[3]. 대부분의 연구에서 이동 에이전트의 성능은 통신 횟수, 이동 에이전트의 크기 및 작업 결과 데이터의 크기 등의 요소에 의해 결정되는 것으로 가정하여, 통신 횟수나 전송량을 줄임으로써 성능을 향상시키려 하였고, 이를 통해 기존의 통신 패러다임보다 이동 에이전트가 우수한 성능을 제공함이 입증되었다.

일반적으로 이동 에이전트는 이주 계획에 따라 각 호스트를 방문하면서 작업을 수행하고, 작업의 결과를 누적하면서 이동하기 때문에 방문하는 호스트의 수가 많아지면 이동 에이전트의 크기가 증가하고 이로 인하여 네트워크 소요시간이 크게 증가하는 문제가 발생한다.

[7]에서는 누적되는 작업 결과를 임계값에 따라 반환하고 이동함으로써 에이전트의 크기의 증가 문제를 해결하는 방법을 제안하고 평가하였다.

[8]과 [9]에서는 이동 에이전트가 실행 코드만을 가지고 이동하고, 생성된 데이터는 이동하기 전에 이동 에이전트의 소유자에게 반환하는 방법을 제안하였다.

[10]에서는 네트워크 소요시간을 최소화하기 위해 이동 에이전트의 멀티캐스트 이주 방법을 제안하고 평가하였다. 이동 에이전트 코드만을 복제하고 이를 각 호스트에 멀티캐스트로 전송하기 때문에 네트워크 소요시간을 최소화할 수 있고 호스트나 네트워크 결합에도 안정적으로 동작할 수 있다. 그러나 이동 에이전트를 생성하는 호스트의 경우 에이전트를 복제하기 위해 지역 계산 시간이 증가하게 되고, 네트워크 트래픽도 증가하기 때문에 네트워크 소요시간은 최소화될 수 있지만 지역 계산 시간의 증가를 고려하지 않고 있다는 단점이 있다.

또한 이전의 성능 연구와 마찬가지로 보안성을 고려하지 않고 있기 때문에 실제 분산 응용 개발에 적용하기 어렵다는 문제가 있다.

III. 제안한 이주 방법

3.1 시스템 구성

일반적으로 이동 에이전트 모델은 이동 에이전트, 이동 에이전트 시스템과 호스트로 구성된다. 그러나 보안을 고려

한 경우 복잡한 보안 연산 수행으로 이동 에이전트 시스템의 복잡도가 증가하게 되어 이동 에이전트의 성능이 저하되는 문제가 발생하게 된다. 이를 해결하기 위해 신뢰 도메인을 이용한 방법이 연구되었다. 관련 연구에서 살펴본 바와 같이 TDGM을 이용한 이동 에이전트 보안 모델은 높은 보안성과 우수한 성능을 제공하지만 작업 결과의 누적으로 고려하지 못하고 있다는 단점이 있다[5-6].

본 논문에서 이동 에이전트의 작업 결과가 누적되는 분산 정보 검색에 적용할 수 있도록 안정적인 성능과 높은 보안성을 제공하는 통합 보안 관리자를 이용한 이동 에이전트 모델을 제안한다.

제안한 이동 에이전트 모델은 이동 에이전트 시스템들, 신뢰 도메인 및 통합 보안 관리자(Integrated Security Manager)로 구성된다. 이동 에이전트는 통합 보안 관리자의 여행 안내 서비스를 통해 수정된 이주 계획에 따라 이동 에이전트 시스템을 방문한다. 이동 에이전트는 신뢰 도메인 내에서 상호 인증 과정 없이 이동할 수 있다.

3.2 통합 보안 관리자

본 논문에서 통합 보안 관리자는 신뢰 도메인 단위로 존재하고, 기존의 이동 에이전트 시스템에 기본적인 도메인 등록과 도메인 보안 정책 관리 등 도메인 관리 서비스를 제공할 뿐 아니라 인증, 서명 및 암호 등의 보안 서비스 및 통합 감사 기능을 제공하는 요소를 의미한다. 또한 통합 보안 관리자는 여행 안내 서비스를 제공하는데 이는 기본적으로 이동 에이전트 이주 계획을 수정하는 하는 기능을 의미한다.

각 에이전트 시스템에 존재하는 지역 감사 모듈은 이동 에이전트의 주요 행위와 자원 접근에 관련된 행위에 대한 감사 정보를 수집하여 통합 보안 관리자에 전송하고, 전송된 정보는 통합된 감사 정보로 유지된다. 이를 이용하여 다중 호스트 공격이나 분산 서비스 거부 공격 등을 발견할 수 있고, 각 에이전트 시스템에서 발생 가능한 감사 정보의 변조 및 삭제로부터 감사 정보를 안전하게 보호할 수 있다[11-14].

또한 통합 보안 관리자는 신뢰 도메인 간 이동 에이전트의 이동을 담당하며, 효율적이고 안정적인 에이전트 이주를 위해 누적 문제를 고려한 여행 안내 서비스를 제공한다. <그림 2>는 제안한 이동 에이전트 모델의 통합 보안 관리자를 나타낸다.

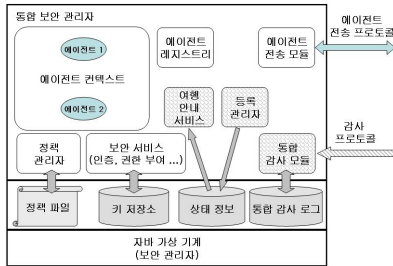


그림 2. 통합 보안 관리자
Fig 2. Integrated Security Manager

3.3 누적을 고려한 여행 안내 서비스

통합 보안 관리자를 이용하는 이동 에이전트 모델에서는 승인된 에이전트 시스템으로부터 인증을 받은 이동 에이전트는 다른 에이전트 시스템을 방문하기 위해 통합 보안 관리자로 전송된다. 통합 보안 관리자는 전송되어온 이동 에이전트를 복호화하고, 서명을 검증하여 무결성을 확인한 후 여행 안내 서비스를 적용한다.

본 논문에서 통합 보안 관리자는 이동 에이전트가 자신이 관리하는 신뢰 도메인 중심으로 이주하도록 하고, 통합 보안 관리자로 회귀할 때 소유자에게 누적된 결과를 반환하도록 이주 계획을 수정한다. <그림 3>은 통합 보안 관리자가 이주 계획을 수정하는 과정으로, 이동 에이전트가 다른 에이전트 시스템을 방문하기 위해서는 먼저 해당 신뢰 도메인을 관리하는 통합 보안 관리자로 이동하게 된다. 이 때 통합 보안 관리자는 {as1, as2, as3, as4, as5, as6, as7, as8, as9}라는 초기 이주 계획을 자신이 관리하는 신뢰 도메인에 포함되어 있는 이주 대상 에이전트 시스템을 먼저 방문하고, 통합 보안 관리자로 회귀할 때 결과를 반환하도록 이주 계획을 {as1, as4, as7, reply, as2, as3, as5, as6, as8, as9}로 변경한다. 이동 에이전트 수정된 이주 계획에 따라 이동하고, 통합 보안 관리자로 회귀할 때 결과를 소유자에게 반환한다.

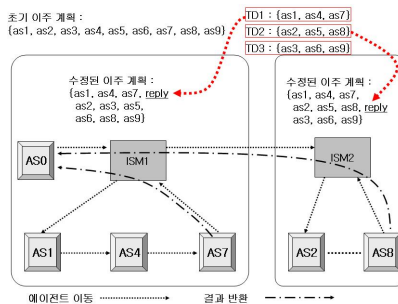


그림 3. 여행 안내 서비스
Fig 3. Travel Guide Service

다른 신뢰 도메인으로 이동할 때 그 동안 누적된 데이터를 소유자에게 반환하기 때문에 다음 신뢰 도메인에서 이동 에이전트 크기는 초기 이동 에이전트 크기와 같게 된다. 따라서 지속적인 누적으로 인한 이동 에이전트 크기 증가로 인한 성능 저하 문제를 해결할 수 있다.

IV. 성능 평가

이동 에이전트의 성능 모델을 살펴보면 이동 에이전트 성능 향상은 이동 에이전트의 네트워크 소요시간을 감소를 의미하기도 하지만 이동 에이전트가 주어진 작업을 수행하는 데 소요된 모든 시간을 의미하기도 한다. 본 논문에서 이주 성능은 이동 에이전트가 주어진 작업을 수행하는 소요 총 시간을 대상으로 한다.

[5]와 [6]에서 성능을 분석하기 위해 고려된 매개변수를 살펴보면 <표 1>과 같다.

표 1. 사용되는 매개변수의 값
Table 1. Parameters used to compute

이름	의미	값
N	방문 에이전트 시스템의 수	50
D_{Code}	이동 에이전트의 코드 크기	39 KB
D_{State}	이동 에이전트의 상태 크기	4 KB
D_{Data}	결과 데이터 크기	512 KB
R_{Se}	정보 검색 처리율	4 회/s
R_{Rf}	정보 기공 처리율	4 회/s
R_{Th}	네트워크 처리율	1 Mbps
R_E	암호화 처리율	1.6 MB/s
R_D	복호화 처리율	1.6 MB/s
R_S	서명 처리율	4.36 MB/s
R_V	서명 검증 처리율	2.03 MB/s
T_{ReqPK}	공개키 요청 시간	0.0000625 sec
T_{ResPK}	공개키 전송 시간	0.2500297 sec
$T_{ReqAuth}$	인증 요청에 소요되는 시간	2.3663329 sec
$T_{ResAuth}$	인증 요청을 처리하고 응답하는 시간	2.3661502 sec
$T_{EndAuth}$	상호 인증이 완료되는데 소요되는 시간	4.76837E-06 sec

각 호스트에서 이동 에이전트가 작업을 수행한 결과로 발생하는 데이터를 처리하여 실제로 전송되는 데이터로 변환하였을 때 감소된 데이터의 비율을 에이전트 선택도, σ 라 한다. 일정 순위에 속하는 정보만을 제공하는 정보 검색의 경우 [5]와 [6]에서처럼 이동 에이전트 코드와 함께 이동하

는 결과 데이터의 크기가 일정할 수 있다. 다음은 초기 이동 에이전트 크기, 결과 데이터 및 이동 중 이동 에이전트 크기를 나타낸다.

$$\begin{aligned} D_{Init} &= D_{Code} + D_{State} \\ D_{Sel} &= D_{Data} * (1 - \sigma) \\ D_{Mig} &= D_{Init} + D_{Sel} \end{aligned}$$

그러나 검색 결과가 누적되는 정보 검색이나 분산 시스템 관리 등에서는 결과 데이터가 누적되기 때문에 전송되는 이동 에이전트의 크기가 이동할수록 증가할 수 있다. 누적을 고려하는 경우 i 번째 이동하는 이동 에이전트의 크기는 다음과 같이 변경되어야 한다.

$$D_{Mig-i} = D_{Init} + D_{Sel} * (i - 1)$$

정보를 검색하고, 가공하는데 소요되는 시간은 다음과 같다.

$$T_{IR} = \left\{ \left(\frac{1}{R_{se}} + \frac{1}{R_{rf}} \right) \left(\frac{1}{1-p} \right) \right\} \quad (0 \leq p < 1)$$

참고로 p 는 동일한 시스템에서 연속적으로 검색 작업을 수행할 확률을 의미한다.

단위 데이터를 암호화, 복호화, 서명 및 서명 검증하는데 소요되는 시간은 다음과 같다.

$$T_{SI} = \left\{ \frac{1}{R_S} + \frac{1}{R_E} + \frac{1}{R_D} + \frac{1}{R_V} \right\}$$

상호 인증에 소요되는 시간은 다음과 같다.

$$\begin{aligned} T_{Au} &= 2T_{ReqPK} + 2T_{ResPK} + T_{ReqAuth} \\ &\quad + T_{ResAuth} + T_{EndAuth} \end{aligned}$$

결과의 누적을 고려하는 경우 수식을 간소화하여 성능 비교를 손쉽게 하기 위해서 "각 호스트에서 누적되는 결과 데이터의 크기, D_{Sel} 는 일정하다"라고 가정한다.

4.1 누적을 고려한 중앙집중형 모델

누적을 고려한 중앙집중형 모델은 [5]에서 제안한 중앙집중형 모델에 작업 결과를 누적으로 이동 에이전트의 크기가 증가되는 것을 적용한 모델이다.

누적을 고려한 중앙집중형 모델에서 정보 검색에 소요되는 총 작업 시간은 다음과 같다.

$$\begin{aligned} T_{RSMATh} &\leq N * \frac{D_{Init}}{R_{Th}} + (N-1) * T_{IR} \\ &\quad + \frac{N * (N+1)}{2} * \frac{D_{Sel}}{R_{Th}} \end{aligned}$$

무결성과 비밀성을 제공하기 위해 소요되는 총 시간은 다음과 같다.

$$\begin{aligned} T_{RSMASc} &\leq \frac{D_{Init}}{R_S} + \frac{D_{Init}}{R_E} + \frac{D_{Init}}{R_D} \\ &\quad + N * \left\{ \frac{D_{Code}}{R_V} + T_{SI} * D_{State} \right\} \\ &\quad + \frac{N * (N+1)}{2} * T_{SI} * D_{Sel} \end{aligned}$$

누적을 고려한 중앙집중형 모델의 총 수행 시간은 다음과 같다.

$$T_{RMA} \leq T_{RSMATh} + (N+1) * T_{Au} + T_{RSMASc}$$

4.2 누적을 고려한 TDGM 모델

누적을 고려한 TDGM 모델은 [6]에서 제안한 모델에 작업 결과를 누적으로 이동 에이전트의 크기가 증가되는 것을 적용한 모델이다.

누적을 고려한 TDGM 모델에서 정보 검색에 소요되는 총 작업 시간은 다음과 같다.

$$\begin{aligned} T_{RTh} &\leq (N+2 * C_{TD}) * \frac{D_{Init}}{R_{Th}} + (N-1) * T_{IR} \\ &\quad + \frac{N * \{N+1+2 * (C_{TD}+1)\}}{2} * \frac{D_{Sel}}{R_{Th}} \end{aligned}$$

무결성과 비밀성을 제공하기 위해 소요되는 총 시간은 다음과 같다.

$$\begin{aligned} T_{RSc} &\leq \frac{D_{Init}}{R_S} + \frac{D_{Init}}{R_E} + \frac{D_{Init}}{R_D} \\ &\quad + (N+2 * C_{TD}) * \left\{ \frac{D_{Code}}{R_V} + T_{SI} * D_{State} \right\} \\ &\quad + \frac{N * \{N+1+2 * (C_{TD}+1)\}}{2} * T_{SI} * D_{Sel} \end{aligned}$$

누적을 고려한 TDGM 모델의 총 수행 시간은 다음과 같다.

$$T_{RTDGM} \leq T_{RTh} + C_{TD} * T_{Au} + C_{TD} * T_{Guide} + T_{RSec}$$

4.3 ISM을 이용하는 모델

누적으로 고려한 여행 안내 서비스를 제공하는 ISM을 이용한 모델에서 정보 검색 작업을 수행하는데 소요되는 총 시간, T_{Th} 는 다음과 같다.

$$T_{Th} \leq (N+2 * C_{TD}) * \frac{D_{Mit}}{R_{Th}} + (N-1) * T_{IR} + \frac{N * \left(\frac{N}{C_{TD}} - 1 \right)}{2} * \frac{D_{Sel}}{R_{Th}}$$

무결성과 비밀성을 제공하기 위해 소요되는 총 시간, T_{Sec} 는 다음과 같다.

$$T_{Sec} \leq \frac{D_{Mit}}{R_S} + \frac{D_{Mit}}{R_E} + \frac{D_{Mit}}{R_D} + (N+2 * C_{TD}) * \left\{ \frac{D_{Code}}{R_V} + T_{Sl} * D_{State} \right\} + \frac{N * \left(\frac{N}{C_{TD}} - 1 \right)}{2} * T_{Sl} * D_{Sel}$$

통합 보안 관리자간의 상호 인증에 소요되는 총 시간, T_{Auth} 은 다음과 같다.

$$T_{Auth} = C_{TD} * T_{Au}$$

통합 보안 관리자가 여행 안내를 하는데 소요된 시간을 $T_{ACGuide}$ 라 할 때 이주 계획 안내에 소요되는 총 시간, T_{MP} 는 다음과 같다.

$$T_{MP} = C_{TD} * T_{ACGuide}$$

누적된 결과 데이터를 에이전트 소유자에게 전송하는데 소요되는 총 시간, T_{Reply} 는 다음과 같다.

$$T_{Reply} \leq C_{TD} * \left\{ 3 * \frac{N}{C_{TD}} * \left(\frac{1}{R_{Th}} + T_{Sl} \right) * D_{Sel} + T_{RAuth} \right\}$$

통합 보안 관리자를 이용하는 에이전트 모델의 총 수행 시간, T_{ISM} 은 다음과 같다.

$$T_{ISM} \leq T_{Th} + T_{Auth} + T_{Sec} + T_{MP} + T_{Reply}$$

4.2 실험 및 평가

본 논문에서는 펜티엄IV 1.4Mhz-640MB와 펜티엄IV 2.4GHz-1GB PC에 자바 기반 이동 에이전트 시스템 Aglet 2.0.2를 사용하여 자바 벡터로 구현된 이주 계획을 수정하는 소요된 시간을 실험하였다. 실험 결과 참여 에이전트 시스템의 수, N 이 50인 경우 여행 안내에 소요되는 시간, T_{AGuide} 는 60~85 ms로 측정되었다.

본 논문에서는 <표 1>의 매개변수와 측정된 실험값을 이용하여 총 수행 시간 측면에서 각 모델을 해석적 방법으로 평가하였다. 누적을 고려한 중앙집중형 모델과 누적으로 고려한 TDGM 모델의 경우 통합 감사 기능을 제공하지 않기 때문에 성능 비교 시 이를 배제하였다.

<그림 4>는 동일한 에이전트 시스템에서 연속적으로 작업할 확률, p 는 0.1이고, 네트워크 처리율이 10Mbps이며, 신뢰 도메인 당 방문한 에이전트 시스템의 수 4인 경우 에이전트 선택도 변화에 따른 각 모델간의 작업 수행 시간의 변화를 나타낸다.

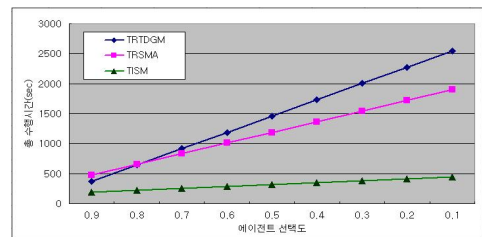


그림 4. 에이전트 선택도에 따른 총 수행시간
Fig 4. Total execution time versus the selectivity of mobile agent

에이전트 선택도가 높아 누적되는 결과 데이터가 작은 경우 신뢰 도메인을 이용한 모델이 성능이 중앙집중형보다 우수한 성능을 제공하지만, 에이전트 선택도가 낮은 경우 누적되는 결과 데이터의 크기가 커지고, 따라서 총 수행시간이 크게 증가한다. 제안한 방법은 기존의 방법과 비교하여 누적되는 결과 데이터가 큰 경우 최대 82%의 성능 향상을, 작은 경우에도 최소 48%의 성능 향상을 제공하며, 안정적인 성능을 제공함을 알 수 있다.

〈그림 5〉는 p 는 0.1이고, 에이전트 선택도가 0.5이고, 신뢰 도메인 당 방문한 에이전트 시스템의 수 4인 경우 네트워크 처리를 변화에 따른 각 모델간의 작업 수행 시간의 변화를 나타낸다.

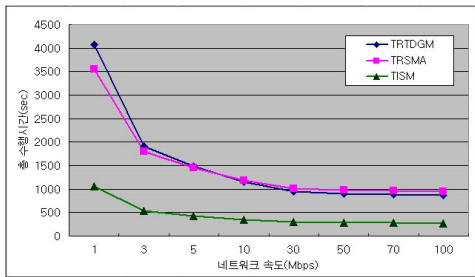


그림 5. 네트워크 처리를 변화에 따른 총 수행시간
Fig 5. Total execution time versus the network throughput

네트워크 처리율이 높은 경우 이동 에이전트를 전송하는데 소요되는 시간이 감소하기 때문에 보안 연산을 적게 수행하는 모델이 우수한 성능을 나타내고, 네트워크 처리율이 낮은 경우 이동 에이전트를 전송하는데 소요되는 시간이 증가하기 때문에 이동 에이전트의 이동이 적은 모델이 우수한 성능을 나타낸다. 제안한 방법은 두 가지 장점을 모두 고려하기 때문에 68 ~ 75%의 향상된 성능을 제공할 수 있다.

〈그림 6〉은 p 는 0.1이고, 에이전트 선택도가 0.9이며, 네트워크 처리율이 10Mbps인 경우 신뢰 도메인 당 방문한 에이전트 시스템의 수의 변화에 따른 각 모델간의 작업 수행 시간의 변화를 나타낸다.

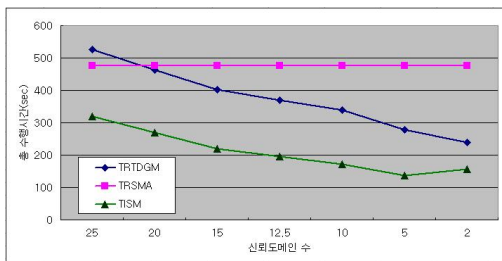


그림 6. 신뢰 도메인 수에 따른 총 수행시간
Fig 6. Total execution time versus the number of trusted domain

제안한 방법은 신뢰 도메인 당 방문한 에이전트 시스템의 수의 많을수록 보안 연산에 소요되는 시간이 감소되기 때문에 33 ~ 72%의 성능이 향상됨을 알 수 있다. 제안한

방법의 경우 신뢰 도메인의 크기가 커지면 누적되는 결과 데이터가 크게 증가하기 때문에 성능 향상이 둔화되지만 기존의 방법 보다는 향상된 성능을 제공할 수 있다.

제안한 모델의 보안성을 살펴보면 [5]와 [6]에서는 인증, 서명 및 암호 등의 보안서비스만을 제공하지만, 지역 감사 및 통합 감사 기능을 제공하도록 설계하였기 때문에 향상된 보안성을 기대할 수 있다.

V. 결론 및 향후 연구

실제 분산 응용 프로그램 개발에 이동 에이전트가 적용되기 위해서는 보안성과 성능이 모두 고려되어야 하며, 이를 위한 연구가 진행되었다. 그러나 보안성과 성능을 모두 고려한 연구에서는 작업 결과의 누적으로 발생하는 문제를 고려하지 않고 있기 때문에 작업 결과가 누적되는 분산 응용에는 적용하기 어려웠다.

본 논문에서는 작업 결과가 누적되는 이동 에이전트 보안 모델에서 누적으로 인한 성능 저하 문제를 해결할 수 있도록 통합 보안 관리자를 이용한 이동 에이전트 모델을 제안하고 그 성능을 평가하였다. 제안한 방법은 통합 보안 관리자가 누적된 작업 결과를 반환함으로써 작업 결과가 계속 누적됨으로써 전송 시간이 크게 증가하는 문제를 해결하였고, 신뢰 도메인을 이용하여 보안 연산을 줄임으로써 기존의 방법보다 33 ~ 82% 향상된 성능을 제공한다.

향후 연구로는 제안한 모델을 구현해보고, 호스트와 네트워크 상태를 고려하여 결과 반환 빈도를 정하는 연구가 필요하다.

참고문헌

- [1] M. Strasser and M. Schwehm, "A Performance Model for Mobile Agent System," Proc. the International Conference Parallel and Distributed Processing Techniques and Applications PDP97, 1997.
- [2] A. Puliafito, S. Riccobene and M. Scarpa, "An analytical comparison of the client-server, remote evaluation and mobile agents paradigms," the First International Symposium on Agent Systems and Applications/Mobile Agents, Palm Springs, California(USA), Oct 1999.

- [3] L. Tang and B. Pagurek, "A Comparative Evaluation of Mobile Agent Performance for Network Management," Proc. the 9th IEEE International Workshop on the Engineering of Computer-Based Systems, Apr 8-11, 2002.
- [4] 박재경, 원유현, "안전한 이동 에이전트 게이트웨이의 설계 및 구현," 정보과학회 논문지 : 컴퓨팅의 실제, 제8권, 제3호, Apr 2002.
- [5] 한승완 외 3인, "보안 서비스를 고려한 이동 에이전트 모델과 클라이언트-서버 모델의 성능 비교," 정보과학회 논문지 : 정보통신, 제29권, 제3호, Jun 2002.
- [6] 유응구, 이금석, "성능을 고려한 이동 에이전트 보안 모델," 정보과학회 논문지 : 정보통신, 제30권, 제5호, Oct 2003.
- [7] 유응구, 이금석, "이동 에이전트를 이용한 대규모 분산 시스템 관리를 위한 이동 모델 설계," 정보과학회 추계 학술발표논문집, 제25권, 제2호(III), Oct, 1998.
- [8] D. Hagimont and L. Ismail, "A Performance Evaluation of the Mobile Agent Paradigm," Proc. OOPSLA' 99, International Conference on Object-Oriented Programming, Systems and Applications, Nov, 1999.
- [9] M. Strasser and K. Rothermel, "System Mechanism for Partial Rollback for Mobile Agent," Proc. of the 20th International Conference on Distributed Computing Systems, 2000.
- [10] 김광중 외 3인, "네트워크 소요시간 최소화를 위한 이동 에이전트의 멀티캐스트 이주 모델 구현," 정보처리학회 논문지 D, 제12권, 제2호, Apr, 2005.
- [11] G. Vigna, B. Cassell and D. Fayram, "An Intrusion Detection System for Aglets," Proc. the International Conference on Mobile Agents(MA'02), Barcelona, Spain, Oct 2002.
- [12] G. Noordende, F. Brazier and A. Tanenbaum, "A Security Framework for a Mobile Agent System," Proc. the Second International Workshop on Security of Mobile Multiagent System (SeMAS 2002), Jul 2002.
- [13] CERT/CC, "Manage logging and other data collection mechanisms," <http://www.cert.org/security-improvement/practices/p092.html>.
- [14] Chl0ie, "Setting up a Linux Log Server to enhance System Security", http://www.linuxsecurity.com/feature_stories/feature_story-65.html, Aug, 2000.

저 자 소 개



유 응 구

2004년 2월 : 동국대학교 컴퓨터공학과 박사(공학박사)

2004년 ~ 현재 : 인덕대학 컴퓨터 소프트웨어과 교수

관심분야 : 이동 에이전트, 시스템 관리, 보안 관리, 가상화