이동 에이전트를 이용한 이기종 환경에서의 호스트 부하를 고려한 컴퓨터 바이러스 탐색 시스템의 설계 및 구현

최종욱° 김영균 오길호
금오공과대학교 컴퓨터공학부
{jwchoi, ygkim, gilho}@cespc1.kumoh.ac.kr

Design and Implementation of A Computer-virus Detection System with Host Load Conditions using Mobile Agents on Heterogeneous Environments

Jong-Wook Choi° Young-Gyun Kim Gil-Ho Oh
School of Computer Engineering, Kumoh National University of Technology

요 약

최근 컴퓨터 바이러스와 해킹 기법의 기술적인 향상으로 인하여 바이러스로 인한 피해가 확산되고 있다. 이에 따른 바이러스와 해킹 피해들로부터 시스템과 사용자 데이터를 보호하기 위한 다양한 방법들이 연구 및 적용되어 있다.하지만 기존의 연구는 이기종으로 구성된 서로 이질적인 네트워크 환경에 적용하고 사용하기 위해서는 사용자의 많은 수동적인 노력과 시간을 필요로 하고 있다. 본 논문에서는 이기종으로 구성된 네트워크상에서 이동 에이전트를 이용한 바이러스 탐색 기법에 대해 연구하였다.제시한.방법은 사용자들에게 바이러스 탐색 에이전트와 관련된 해당 호스트상에서의 탐색업무 수행 투명성을 제공하여 호스트의 부하에 크게 영향을 주지 않는 방안으로써 자바 언어 특성인 플랫폼의 독립성이라는 이점을 지원하고 있는 자바 기반의 바이러스 탐색 시스템을 설계하였다. 이는 중앙 집중 관리 형태의 서버기반 방식으로 등록된 지역 네트워크 내의 이질적인 호스트에서 각 호스트의 부하를 고려하여 바이러스 탐색 업무를 수행함으로써 사용자로 하여금 능동성과 자율성,바이러스 탐색 업무에 있어서의 투명성을 제공할 수 있는 컴퓨터 바이러스 탐색 업무를 수행하는 이동 에이전트 기반의 탐색 시스템을 새롭게 제안한다.

1.서론

인터넷 사용의 증가로 개인 정보 유출과 데이터 파괴 및 시스템을 위협하는 신종 컴퓨터 바이러스의 종류와 피해가 날로 확산됨에 따라 바이러스 검사 및 차단을 위한 다양한 방법들이 연구 및 배포 되고 있다. 이러한 바이러스 위협에 있어서 LAN으로 연결된 이기종의 컴퓨터들은 많은 프로그램을 공유함은 물론 많은 자료를 고속으로 교환할 수 있기 때문에, 한 곳에 침투한 컴퓨터 바이러스[1]가 순식간에 다른 컴퓨터를 감염시킬 수 있어 그 위협 또한 증가 되고 있다. 그러나 이러한 바이러스들의 위협으로부터 대처할 수 있는 상업용 백신들은 대부분이 클라이언트용으로 윈도우 시스템에서 주로 사용되어 오고 있으며 리눅스, 유닉스 계열의 서버용 백신들은 고가이면서 그 수 또한 제한되어 있어 바이러스 확산과 치료에 사용자들의 많은 시간과 노력을 요구하여 바이러스 위협에 대해 능동적으로 대처하지 못하는 실정에 있다.이에 본 논문에서는 지역 네트워크 내의 이질적인 환경에서도 수행이 가능한 플랫폼에 독립적인 자바 기반의 이동 에이전트(Mobile Agents)[2]를 이용하여 중앙 집중 관리 방식인 서버 기반 탐색 시스템을 설계,구축하여 지역 네트워크 내의 서로 이질적인 시스템을 사용하고 있는 사용자들이 좀더 자율적으로 바이러스의 위협에 대처할 수 있는 방안을 제시한다. 이와 더불어 바이러스 탐색을 수행하는데 있어서 해당 호스트의 부하를 고려하여 사용자들의 업무에 방해되지 않는 좀 더 지능적이면서 효율적인 방법으로 바이러스를 탐지 및 복구 할 수 있는 방법을 새롭게 연구하였다. 본 논문에서 제안한 방법은 서버 기반의 중앙 집중형 관리방식으로서 지역 네트워크 내의 이기종의 시스템을 사용하는 사용자가 바이러스 위협과 치료에 대한 부담을 줄일 수 있어 좀 더 능동적인 방법과 지능적인 방법으로 네트워크 상에서 유입되는 바이러스들의 진단 및 치료를 수행할 수 있다.

2. 이동 에이전트

이동 에이전트란 자율성을 가지고 비동기적으로 수행이 가능한 개체로,선택적으로 지능을 가지고 특정 작업을 수행하기 위해 네트워크 상의 한 호스트에서 다른 호스트로 이동할 수 있는 개체라고 정의 할 수 있다. 네트워크 상의 호스트는 이러한 이동 에이전트가 실행 할 수 있는 환경을 제공하여 이질적인 하드웨어와 운영체제로 구성된 분산 컴퓨팅 환경에 동적인 이식을 가능케 하여(Portability) 서로 이질적인 분산 환경에서도 에이전트가 실행할 수 있는 환경을 제공하여 주는 에이전트 시스템의 역할을 수행한다. 이러한 에이전트는 특정 작업을 수행하기 위해 자신을 복제(Clone)하여 작업 부하를 분산시킬 수 있으며 자신의 현재 실행 상태를 저장하여 네트워크를 통해 이동하며(Code mobility) 상태를 다시 복원함으로써 해당 작업(본 논문에서는 바이러스 패턴 코드 탐색 작업)에 대한 실행을 재개할 수 있어 과부하 된 호스트의 부하를 줄여 수행의 효율성을 높이는 기능을 제공 할 수 있다[3][4].

본 논문에서는 이러한 이동 에이전트의 특성들을 응용하여 이기종으로 구성된 지역 네트워크 내의 호스트들에 대한 부하를 중앙의 백신 이동에이전트가 가지고 있는 특정 호스트에 대한 바이러스 발견 빈도에 따라서 부하 측정 에이전트를 파견, 측정하여 그 결과값에 따라 바이러스 탐색 업무를 수행하는 이동 에이전트의 파견 개수를 중앙의 백신 서버에서 조절, 파견하여 지역 네트워크 내의 사용자들에게 바이러스 탐색 에이전트의 수행 투명성을 제공할 수 있어 효율적인 작업 환경을 지원하는 장점을 제공하는 동시에 사용자가 좀 더 자율적으로 이기종의 환경에서 바이러스에 대해 대처할 수 있는 장점을 제공한다.

3.바이러스 탐색 시스템 설계 및 구현

3.1 기본 정책

이동 에이전트를 이용한 이기종 환경에서의 부하를 고려한 컴퓨터 바이러스 탐색 시스템은 지역네트워크 내의 호스트에 탐색 에이전트의 수행을 최적화된 호스트 성능을 감안하여 바이러스 탐색 업무를 수행할 수 있어야 한다. 또한 등록된 모든 지역 네트워크 내의 Host들에는 플랫폼에 독립적인 자바 기반의 Agent System인 IBM Aglets system[5]이 설치 되어 있어야 한다.기본 정책은 호스트의 부하와 등록된 각각의 호스트 내에서의 바이러스 발견 빈도에 따라서 부하측정 이동 에이전트(LMMA) 파견 주기와 탐색 에이전트(SMA)의 파견 수를 중앙의 백신 이동 에이전트 서버(VMAS)에서 관리함으로써 조절할 수 있다. 본 논문에서의 호스트의 부하 측정 기준은 지역 네트워크 내의 이기종 환경에서의 호스트 성능을 고려하여 일정 크기의 정수 행렬 곱셈 연산을 하는데 걸리는 시간을 구해 각 호스트간의 부하를 측정하였다. 단, 지역네트워크 내의 네트워크 트래픽은 없는 것으로 간주하여 고려하지 않았다.각 호스트에 대한 부하 지표(Load Index) 값은 주어진 작업(행렬 계산)을 수행하는데 있어서 임의의 지역 네트워크 내의 호스트에서 걸리는 최소 수행시간(P_{Mintime})을 부하 측정 에이전트가 측정하였을 때의 시간(P_{time})을 최소 수행 시간으로 나눈 배수 비례식으로서 나타내었다. 즉 해당 호스트의 LoadIndex가 1의 값을 가질 때 호스트의 부하는 없는 것으로 간주하였다. 단, 구현에 있어서 여러 번 실험을 바탕으로 한 결과로 LoadIndex의 값이 10을 초과하였을 경우 그 호스트는 과부하 상태로 간주하고 실험 및 성능평가를 하였다[6].

$$LoadIndex_{Host} = \frac{P_{time}}{P_{Mintime}} \quad [\text{식 1}]$$

3.2 부하 측정 방법

3.2.1 부하 측정 에이전트 파견주기 정책

부하 측정 에이전트의 파견주기 정책(LMMA_dispatch_policy)은 지역 네트워크 내 각각의 등록 호스트에서 일정 시간동안(Δt) 의 바이러스 감염 빈도(Virus_infection_frequency$_{Host}$)와 비례한다[식 2]. 중앙의 백신 에이전트 서버는 각각의 등록 호스트들에 대해서 탐색 에이전트가 수집한 등록 호스트들에 대한 일정한 시간동안의 바이러스 감염 빈도 정보를 저장하고 있으며 이 정보에 따라 부하 측정 에이전트의 파견 주기를 결정할 수 있다. 즉 바이러스가 자주 발견 된 호스트에 대해서는 부하측정 에이전트의 파견주기를 다른 호스트에 비해 짧게 설정하여 그 호스트에 대해서는 바이러스 탐색 에이전트의 파견 수(SMA_dispatch_Numbers$_{Host}$) 또한 함께 조절하여 관리할 수 있어 좀 더 지능적인 방법으로 등록된 호스트들과 지역 네트워크 환경에 있어서 부하 측정 작업 자체에 대한 부하를 줄이면서 탐색 작업을 행할 수 있다.

$$(\text{LMMA\_dispatch\_frequency}_{Host} \ \& \ \text{SMA\_dispatch\_Numbers}_{Host}) \propto \text{Virus\_infection\_frequency}_{Host} / \Delta t \quad [\text{식 2}]$$

3.2.3 이동 에이전트(Mobile agent)를 이용한 부하 측정

중앙의 백신 이동 에이전트 서버에서는 부하측정 에이전트 파견주기 정책에 따라 부하 측정 에이전트를 지역 네트워크 내에 파견하여 도착 호스트에 대한 성능지표 값을 중앙의 백신 서버 에이전트에게 전송한 후 다음 호스트로 이동하여 부하 측정 업무를 수행한다.

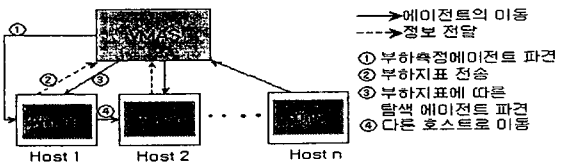

그림 2. 이동 에이전트를 이용한 부하 측정 절차

3.3 제안한 시스템의 구조

본 논문에서 제안한 시스템의 구성은 백신 이동 에이전트 서버,부하측정 에이전트 바이러스 탐색 에이전트,백신 에이전트,바이러스 정보 데이터베이스로 구성되어 있으며 각각의 구성별 모듈의 역할은 다음 표 1과 같다.

표 1. 시스템 구성별 모듈의 역할

•VMAS(Vaccine Mobile Agent Server)	*LAN상의 등록된 호스트에 대한 바이러스 탐색/치료 관리 *SMA/VMA/LMMA hosting, *VIDB update
•LMMA(Load Measure Mobile Agent)	*Host 부하 측정
•SMA(Scanner Mobile Agent)	*바이러스 탐색 , *VMAS에게 VMA요구
•VMA(Vaccine Mobile Agent)	*발견된 바이러스 치료
•VIDB(Vaccine Information DataBase)	*이기종 시스템에 따른 바이러스 정보(바이러스 패턴코드, 감염 증상,치료 방법)와 부하지표(LI)정보테이블,백신 이동 에이전트,부하 측정 에이전트 저장

표 2. 부하 지표(LI) 정보 테이블 항목

	등록된 호스트의 위치정보
	등록된 호스트들의 바이러스 감염 빈도
	감염빈도에 따른 부하측정 에이전트 파견 수
	등록된 호스트들의 최소 부하지표 정보
	등록된 호스트들의 부하 지표
	부하지표에 따른 탐색 에이전트 파견 수

3.4 시스템의 동작구조

제안한 시스템은 바이러스 탐색 기능을 모듈화 하여 차후에 확장이 용이하도록 설계하였다.

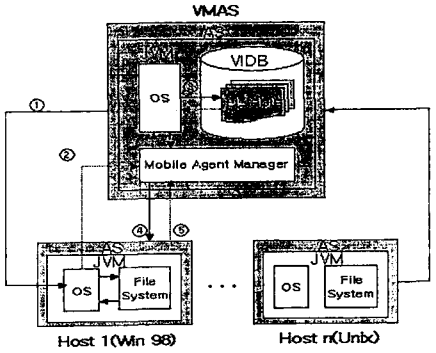

그림 3. 시스템 동작 구조

① VMAS는 VIDB내의 LI_table 에서 등록된 호스트의 최소 부하지표 정보(H_min)와 바이러스 감염 빈도 정보를 참조하여 해당하는 호스트에 LMMA를 파견하여 부하 측정

② LMMA는 자신이 가지고 있는 작업을 도착 호스트에서 수행한 시간과 해당 호스트의 부하지표를 이용해 결과값을 VMAS에게 전송

③ VMAS는 전송된 결과값을 가지고 자신의 VIDB내의 LI_table에서 참조하여 SMA파견 수를 결정하여 해당 호스트로 전송

④ 도착한 SMA들은 해당 호스트에 대한 바이러스 탐색 업무를 수행

⑤ 바이러스의 탐색 업무가 종료되면 그 결과값을 VMAS에게 보고 , VMAS는 탐

색 업무의 결과에 따라 등록된 해당 호스트내의 바이러스 감염 빈도 정보 유지 및 갱신하며 해당 호스트에 VMA를 파견하여 발견된 바이러스 치료

4.결과 및 성능평가

본 논문의 구현에서는 이기종의 LAN으로 구성된 16대의 호스트를 대상으로 등록 호스트 수를 점차 늘려가면서 각 호스트들에 대한 부하측정을 고려한 방법과 고려하지 않은 방법으로 평균 바이러스 탐색 시간을 측정 하였다.부하 측정을 하기 위해 사용한 작업은 동일한 N × N 정수 행렬 곱셈을 여러 번 반복적으로 수행한 시간을 측정하였다.표3은 부하지표에 따른 탐색 에이전트 파견 개수를 나타내었다.

표 3. 부하지표에 따른 탐색 에이전트 파견 개수

1 ≤ i < 5	4	i = 실수
5 ≤ i ≤ 10	2	
10 < i	1	

구현 환경으로는 에이전트 시스템으로서는 플랫폼에 독립적인 자바 기반의 다중 에이전트 시스템인 IBM의 ASDK(Aglets Software Development Kit) 1.1 beta3 버전과 자바 개발도구 JSDK(Java Standard Development Kit) 1.1.8 버전을 사용하여 구현하였으며 각각의 지역 네트워크 내의 등록 호스트의 부하를 고려 하였을 때의 바이러스 평균 탐색 시간 측정과 부하를 고려하지 않았을 때의 바이러스 평균 탐색 시간을 측정하였다. 즉 부하를 고려하지 않았을 때의 지역 네트워크 내의 호스트로 파견할 기본적인 탐색 에이전트의 개수를 2로 고정시키고 이에 따라서 호스트의 부하를 고려하였을 때 탐색 에이전트의 파견개수를 부하지표에 따라 조절하여 파견하였을 때 바이러스 탐색의 전체 평균시간을 측정하였다. 단 각 호스트의 파일 시스템내의 바이러스 파일 감염 수는 무작위로 설정하여 성능평가를 하였으며 바이러스 탐색 기법은 지역 네트워크의 트래픽에 크게 영향을 받지않는 순환탐색기법[7]을 사용하였다. 표 4는 지역 네트워크 내의 서로 다른 이기종으로 구성된 호스트 수에 따라서 부하를 고려하였을 경우와 고려하지 않았을 경우의 바이러스 평균 탐색시간을 나타내었다.

표 4. 등록 호스트 수에 따른 평균 탐색 시간

LAN내의 Host 의 수와 평균 부하	실험 횟수	탐색 시간 (Minutes)			
			평균 시간		평균 시간
Win98se(1)	1	1.7	2.4	1.9	2.2
Solaris(1)	2	2.3		2.1	
Win2000pro(1)	4	3.1		2.7	
Win2000xp (1)					
Win98se(2)	1	4.2	5.9	4.9	6.2
Solaris(2)	2	7.1		7.8	
Win2000pro(2)	4	6.3		5.9	
Win2000xp (2)					
Win98se(4)	1	11.2	15.2	15.2	15.9
Solaris(4)	2	14.9		15.5	
Win2000pro(4)	4	19.6		17.1	
Win2000xp (4)					

실험에 있어서 지역 네트워크내의 등록 호스트들의 수가 적을 경우에는 부하를 고려하였을 경우와 부하를 고려하지 않았을 경우 평균 탐색시간은 오히려 부하를 고려하지 않았을 때가 평균적으로 조금 더 나은 결과를 보였다.하지만 지역 네트워크 내의 등록 호스트의 수가 증가하면 증가할 수록 부하를 고려하였을 경우가 부하를 고려하지 않았을 경우보다 평균탐색시간과 그 성능면에 있어서 조금씩 개선되어 진다는 것을 확인할 수 있었다.그림 5는 등록 호스트 수에 따른 평균탐색시간을 그래프로 표현한 그림이다.

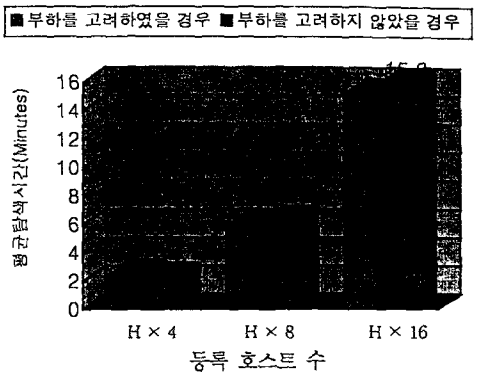

그림 5. 등록 호스트 수에 따른 평균 바이러스 탐색 시간

5.결론 및 향후 계획

본 논문에서 설계 및 구현된 이동 에이전트를 이용한 이기종 환경에서의 바이러스 탐색 시스템은 지역 네트워크 내의 서버 기반 중앙집중 관리 방식 시스템으로서 서로 다른 이질적인 시스템을 사용하는 사용자들의 바이러스에 대한 수동적인 노력과 부담을 줄일 수 있으며 또한 등록된 호스트에 대한 작업 부하를 고려하여 바이러스 탐색 업무를 투명하게 수행할 수 있어 바이러스 탐색 업무와 사용자들의 컴퓨팅 환경을 좀 더 효율적으로 구축하도록 설계 및 구현하였다. 향후 계획으로는 지역 네트워크내의 트래픽과 좀 더 정확한 입력 데이터를 바탕으로 한 최적화된 컴퓨팅 환경에서의 바이러스 탐색 에이전트의 개수 및 파견 주기 시간에 대해 대처할 수 있는 좀더 지능적인 바이러스 탐색 시스템에 대해서 연구할 계획이다.

참고 문헌

[1] 권석철, " 컴퓨터 바이러스 기술 동향 및 대응 전략" , 2001 국제 컨퍼런스 IT21-정보처리학회 pp.287-316, 2001.6

[2] OMG, " Mobile Agent Facility Specification" http://www.omg.org/, pp12-13, 2000.

[3] *anny B.Lange , Mitsuru Oshima* , " Programming and Deploying Java Mobile Agents with Aglets"

[4] D. Chess, C. Harrison, A. Kershenbaum. " Mobile agents : Are they a good idea?, *In Mobile Object Systems: Towards the Programmable Internet*" , Vol. 1222 of Lecture Notes in Computer Science, Spring-Verlag, 1997.

[5] Aglets Software Development kit, http://www.trl.ibm.com/aglets/

[6] 김지균, 김태윤, " 이동 에이전트 기반의 동적 작업 부하균형 프레임워크" , 정보과학회 논문집, 제28권, 제2호, pp196-206, 6월,2001년.

[7] 최종욱, 김영균,오길호 " 이동 에이전트를 이용한 지역 네트워크 단위에서의 컴퓨터 바이러스 탐지 및 치료방법 구현" , 정보 과학회 가을 학술발표논문집, 제28권, 제1호, pp 778-780, 10월, 2001년.