분산 네트웍 환경에서 실시간 모니터링시스템(NetCop)의 설계 및 구현

윤치영[0] 정천복 황선명
대전대학교 컴퓨터공학과 소프트웨어공학 연구실
mrmeg@zeus.taejon.ac.kr
chunbokj@hanmail.net
sunhwang@dragon.taejon.ac.kr

A Design and Implementation of Real Time NetCop on Distributed Network Environment

Chi-Young Yoon[0] Chun-Bok Jung Sun-Myung Hwang
Dept. of Computer Engineering, Daejeon University

요 약

본 논문에서는 분산된 Host들의 실시간 모니터링 및 원격지 Host의 제어,관리를 수행함으로써 네트웍 활용의 효율을 높이기 위한 분산 네트웍 환경에서 실시간 모니터링시스템(NetCop)을 구현 연구하였다.

이 시스템은 분산된 Host들의 효율적인 네트웍 활용 및 관리를 위해 그것의 성능을 저해하는 사용자의 유무를 체크하고 경고 메시지 및 허용되지 않은 프로그램의 실행 방지 등 사용자의 편리를 위해 제공·비치된 컴퓨터나 어떤 형태로든지 네트웍 망에 연결된 단말기의 사용범위를 실시간으로 감시, 제어 및 관리할 수 있는 기능을 제공한다. 또한 그것으로 인해 네트웍 사용의 낭비를 막을 수 있고 더 나아가 네트웍의 전체적인 성능을 높일 수 있으며 사용자에게 더 효율적인 사용 환경을 제공할 수 있는 장점이 있다.

* 외곽의 빨간 선은 메뉴의 보기/안내선을 선택하면 보이며, 출력되지는 않습니다.

1. 서론

인터넷의 대중적인 보급과 더불어 컴퓨터 네트워크 구성 또한 빠르게 발전, 확산되고 있는 시점에서 어느곳이든지 이러한 기류에 못지않게 크게 뒤지지 않는 컴퓨터 네트워크 환경을 구축했거나 앞을 다투어 네트워크 환경구성에 힘을 쏟고 있다. 정보 사회에 적극적으로 적응하기 위해서는 정보의 신속한 입수와 체계적 관리 및 활용 능력을 함양하는 것이 매우 중요하다. 하지만 이러한 환경을 효율적으로 활용하지 못함으로써 그 활용의 가치를 크게 얻지 못하고 있는 실정이다.[6] 바로 특정 사용자의 무분별한 컴퓨터사용이 가져오는 결과이다. 한사용자의 무분별한 사용은 여러 다른 사용자에게까지 그 피해를 확산시키고 있다. 이러한 때에 무분별한 사용자의 호스트 및 네트웍 사용을 막고 그 활용의 효율을 높이기 위해서는 효율적인 네트웍관리가 필요하고 네트웍 활용의 효율을 저해하고 방해하는 사용자로부터 지켜낼 수 있는 안전장치가 필요한 시점이다.[5]

본 연구에서는 무분별한 사용자의 Host 및 네트웍 사용을 원격에서 감시 및 제어할 수 있는 분산 네트웍 환경에서 실시간 모니터링시스템(NetCop)을 구현 연구 하였다. 이러한 시스템은 사용자의 편리를 위해 제공·비치된 컴퓨터나 어떤 형태로든지 네트웍 망에 연결된 단말기의 사용범위를 실시간으로 감시, 제어 및 관리할 수 있는 기능을 제공함으로써 네트웍 사용의 낭비를 막을 수 있고 더 나아가 네트웍의 전체적인 성능을 높일 수 있으며 사용자에게 더 효율적인 사용 환경을 제공할 수 있는 장점이 있다.

본 논문의 구성은 다음과 같다. 2장에서는 제안한 실시간 모니터링 시스템(NetCop)의 구조와 특성을 소개하고 NetCop(Network Cop)의 동작환경과 구현내용은 3장에서 소개한다. 마지막으로 결론과 향후 연구방향에 대해 논의한다.

2. NetCop(Network Cop)의 구조 및 특성

2.1 NetCop의 기본동작

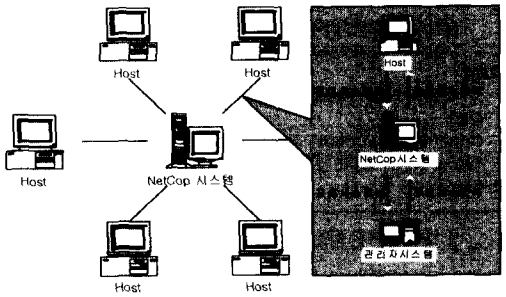

그림 2.1 NetCop 의 기본 동작

그림 2.1에는 NetCop의 기본적인 동작이 나타나 있다. 그림에서 알 수 있듯이 분산된 Host들로부터 등록된 프로세스는 NetCop시스템 서버를 통해 관리되고 제어할 수 있으며 또한 관리자는 1차로 서버에서 제어된 각 Host들에 대한 프로세

스 정보를 모니터링하고 이를 분석하여 불필요한 사용자의 행위에 대해 서버를 통한 권고 메시지 및 실행된 프로세스의 정지 등 직접적이고도 더 강력한 제어기능을 행사할 수 있다.

2.2 NetCop의 구조

NetCop의 구조도는 그림 2.2 와 같다.

분산 네트웍 환경에서 실시간 모니터링시스템(NetCop)은 Host 등록관리, Host 상태관리, Host 연결관리, Host 제어관리, Host 프로세스관리 등 크게 다섯 개의 모듈로 구성되었다.

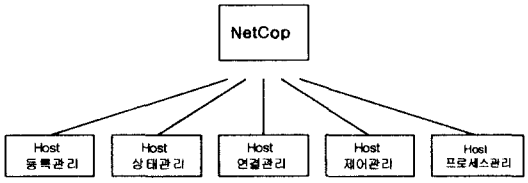

그림 2.2 NetCop 의 구조도

각 모듈별 기능은 다음과 같다.

▶Host 등록관리

관리하고자 하는 Host의 IP주소 또는 Host명을 등록하여 Host를 감시, 제어할 수 있고 또한 Host명으로도 등록가능하기 때문에 유동 IP를 채택한 네트웍 시스템에서도 사용 가능하다. 등록된 Host의 IP주소 및 Host명은 데이터베이스에 저장·관리되고 저장된 Host의 정보는 추후 허용되지 않는 프로세스의 등록 시 함께 연결되어진다.

▶Host 상태관리

각 Host의 네트웍 참여여부와 사용여부를 체크하고 또한 클라이언트 프로그램의 실행 유무를 관리하는 모듈이다.

▶Host 연결관리

클라이언트 프로그램에서 서버로의 시작을 알리는 Start 정보를 전송하고 등록된 Host의 모니터링을 위해 Host의 화면을 서버로 연결하는 등, 각 Host와의 연결을 담당하는 모듈이다.

▶Host 제어관리

1차로 서버에서 제한된 프로세스에는 포함되지 않으나 네트웍 활용을 저해하는 프로그램의 실행이나 사용자의 행위에 대해 관리자가 권고 메시지 및 마우스의 권한을 획득하여 2차로 제어를 할 수 있는 기능을 담당하는 모듈이다.

▶Host 프로세스관리

제한하고자 하는 프로그램의 프로세스를 서버에 등록하여 관리자의 모니터링을 통한 제어가 아니더라도 서버를 통해 1차 제어를 할 수 있다. 서버에 등록된 프로세스에 대해서 관리대상 Host는 그 프로세스를 실행 시 자동 종료되도록 하는 기능을 갖는 모듈이며, 여기에 등록된 프로세스 또한 등록된 Host의 정보와 연결되어 데이터베이스에 저장·관리되어 진다.

3. NetCop(Network Cop)의 구현

본 장에서는 2장에서 언급된 설계에 따라 구현된 분산 네트웍 환경의 실시간 모니터링시스템(NetCop)의 구현에 대하여 소개하기로 한다.

실시간 모니터링 시스템은 서버와 클라이언트 모두 Visual C++ 로 구현되었다. 클라이언트 프로그램은 특정 Host에서 백그라운드 데몬 형태로 실행되며, 서버는 관리대상 Host의 시작 시 Host로부터 서버로 보내는 Start 신호와 Host의 연결을 체크하기 때문에 서버 프로그램은 관리대상 Host가 구동되면 자동 실행되도록 되어 있다.

3.1 NetCop의 이벤트 제어

실시간 모니터링시스템(NetCop)의 이벤트는 크게 서버를 통한 Host의 제어 이벤트와 Host에서 서버로의 모니터링 화면 전송 이벤트 두 가지로 구성된다.

그림 3.1은 서버를 통하여 Host를 제어하기 위한 패킷을 보내는 프로토콜의 모습을 보인 것이다.

그림 3.1 프로토콜 구조

전송되어진 프로토콜은 변수/값의 유무에 따라 분할되어 제어기능을 가르키는 Var 부분은 반환하고 제어기능의 실질적인 값을 가리키는 Val 부분은 포인터로 연결된다. 이렇게 linkedlist에 저장된 값은 이벤트 전송의 완료시점에서 제어를 완료한다.

그림 3.2 는 화면 전송/저장 이벤트 위한 16bit Color 구조를 보이고 있다.

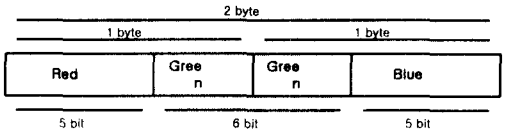

그림 3.2 전송/저장 정보를 위한 16bit Color 구조

그림 3.2에서 보는 바와 같이 본 시스템에서는 화면 전송/저장 정보를 위해 16bit Color를 사용하고 있다. 예를 들어 Server와 연결이 이루어지고 Server화면이 보통 1024*768의 해상도를 가진 시스템이라면 네트워크상에서의 실시간 화면 업데이트를 위하여 100*100으로 나누어 보내게 되고 화면상의 빠른 실시간 업데이트를 위하여 JPEG 압축 코덱을 이용하여 전송한다. 향후 Version-Up 시 16bit Color 이외의 시스템에서도 사용 가능하게 되어야 할 것이다.

3.2 NetCop의 서버 인터페이스

그림 3.3은 실시간 모니터링시스템(NetCop)의 서버 인터페이스를 보인 것이다.

실시간 모니터링시스템(NetCop)은 각 Host의 사용행위에 대해 실시간 모니터링을 통하여 감시, 관리된다. 그림 3.3에서 보는 것과 같이 관리자는 관리대상 Host를 등록관리하고 관리대상 Host의 리스트 화면에서 등록된 Host의 연결여부, 네트웍

참여여부 및 클라이언트 프로그램의 실행여부를 실시간으로 Update하여 보여준다.

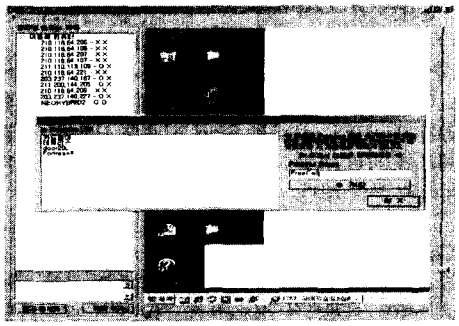

그림 3.3 서버 인터페이스

또한 관리자는 사용자의 Host사용행위에 대해 실행을 제한할 프로세스를 등록·관리할 수 있고 여기에서 등록된 프로세스는 Host에서 실행 시 자동 종료된다. 이와 같은 자동종료 기능은 감시 및 관리되는 많은 Host들에 대해 각각 개별 제한해야 하는 불편함을 없애고 서버에 한번만 등록함으로서 등록된 Host들을 제어할 수 있는 기능이다. 그리고 관리자는 개별 Host의 사용행위에 대한 사용화면을 실시간 모니터링 함으로써 1차로 제한된 프로세스의 목록에는 포함되지 않으나 그 사용범위가 불필요한 사용행위로 간주 될 때에는 직접 Host의 권한을 획득함으로써 제어를 할 수 있다.

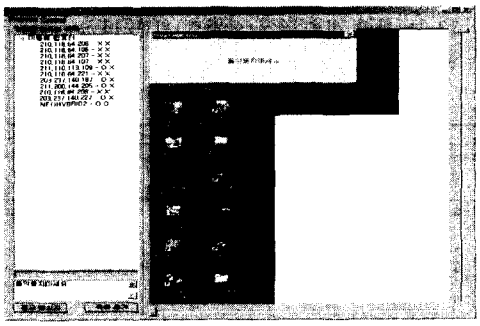

그림 3.4 원격지 Host에 대한 권고메시지

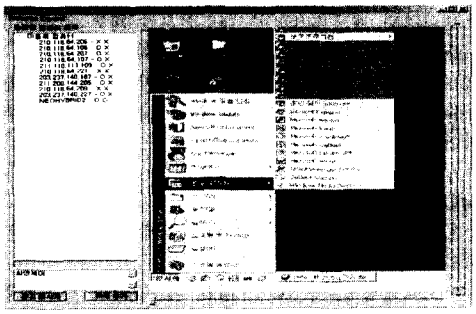

그림 3.5 원격지 Host에 대한 권한 획득

그림 3.4 와 그림 3.5 는 서버로부터 관리대상 Host로의 권고메시지 및 Host에 대한 마우스, 키보드의 권한을 획득하여 제어를 가하고 있는 화면을 보인 것이다. 허용되지 않는 프로세스의 목록에는 포함되지 않으나 관리자의 실시간 모니터링을 통하여 불필요한 행위일 때는 가벼운 권고 메시지 및 Host의 마우스, 키보드의 권한을 획득하여 직접 Host를 제어할 수 있고 특별한 경우에는 Host를 리 부팅 시킬 수 있다.

4. 결론 및 향후 연구 방향

본 논문에서는 무분별한 사용자의 Host 및 네트웍 사용을 원격에서 감시 및 제어할 수 있는 분산 네트웍 환경에서 실시간 모니터링시스템(NetCop)을 구현 연구하였으며 이러한 시스템은 사용자의 편리를 위해 제공·비치된 컴퓨터나 어떤 형태로든지 네트웍 망에 연결된 단말기의 사용범위를 실시간으로 감시, 제어 및 관리할 수 있는 기능을 제공함으로써 다음과 같은 효과를 기대할 수 있다.

첫째, 특정 사용자로 인한 네트웍 사용의 낭비를 막을 수 있기 때문에 다수의 사용자에게 더 효율적인 사용 환경을 제공할 수 있으며 더 나아가 네트웍의 전체적인 성능을 높일 수 있게 될 것이다. 둘째, 다수의 원격지 Host를 서버에서 관리, 제어할 수 있기 때문에 분산된 Host를 관리하는데 소요되는 인적, 물적 자원을 절약할 수 있게 될 것이다. 셋째, 기밀을 필요로 하는 프로젝트를 진행하는 회사나 컴퓨터를 이용한 수업을 진행하는 학교 같은 환경에서의 응용도 기대해 볼 수 있다.

현재 구현된 시스템은 서버와 클라이언트가 수시로 패킷을 교환해야 하기 때문에 오히려 이로 인해 생기는 네트웍의 과부하는 고려하지 않은 실정이며 또한 한번에 하나의 Host 화면만 모니터링하기 때문에 관리자의 계속적인 모니터링이 필요하고 그것을 벗어난 Host는 제한된 프로세스 이외의 행위에 대해서는 무방비 상태에 놓인다는 단점이 있다.

Host들에 대한 계속적인 실시간 모니터링을 하면서도 네트웍에 대한 부담을 줄이고 제한하기 위해 등록하는 프로세스 이름 대신 사용자가 더 쉽게 알 수 있는 것으로 대체할 수 있는 연구를 수행하는 것이 필요할 것으로 생각된다.

[참 고 문 헌]

[1] Dave Bixler, Larry Chambers, Joseph Phil "Implementing and Administering a Microsoft Windows 2000 Network Infrastructure" 삼각형프레스

[2] Donald L. Bailey, Raymond J.A. Buhr "An Introduction to Real-Time Systems from Design to Networking with C/C++" Prentice Hall

[3] Alan Burns, Andrew J. Wellings "Real Time Systems and Programming Languages 3E" Addison-Wesley

[4] John G. Ackenhusen "real-time Signal Processing : Design and Implementation of signal Processing Systems" Prentice Hall

[5] Wayne Wolf, Yiqing Liang, Michael Konzuch, Heathery Yu, and Michael Philips, "A Digital Video Library on the World Wide Web", ACM Multimedia96

[6] 임병학, 방윤학 "효율적 인터넷 트래픽 처리를 위한 공중 가입자 인터넷 액세스 기술" 정보과학회지(99')