

프로비저닝을 통한 효율적인 서버 가상화 메커니즘

김동욱¹, 정갑현², 김강석¹, 손태식¹

¹아주대학교 대학원 지식정보공학과

²(주)유플렉스소프트

¹{suitbertus, kangskim, tsshon}@ajou.ac.kr

²kaphyeon@gmail.com

Mechanism for Effective Server Virtualization with Provisioning

Dongwook Kim¹, Kaphyeon Jung², Kangseok Kim¹, Taeshik Shon¹

¹Dept. of Knowledge Information Engineering, Graduate School of Ajou
University

²uPlexsoft Co., Ltd. Seoul Korea

요 약

IT 기술의 급격한 발달로 인해 개인의 생활뿐만 아니라 국내 기업들의 업무 환경까지도 많은 변화가 일어나고 있다. 이러한 변화 중 가상화 기술을 이용하여 업무의 효율성 증대와 경제적이며 관리 능력의 향상을 기대하며 가상화 환경을 도입하려는 기관이 많이 생겨나고 있다. 하지만 가상화 기술은 안정적인 서버의 운영이 뒷받침 되지 않는다면 막대한 피해를 줄 수 있다. 따라서 본 논문에서는 데스크탑 가상화 환경에서 프로비저닝(Provisioning) 과정을 이용하여 가상 서버의 시스템 자원을 최적화시키고, 최적화된 가상 서버에 사용자VM(Virtual Machine)을 할당하는 부하 분산 방안을 제안한다.

1. 서론

인터넷의 급성장으로 서버가 받는 사용자들의 요청을 급격하게 증가 시키면서 서버에 심각한 부하를 일으키고 있다. 이러한 상황으로 인해 기업에서는 좀 더 높은 가용성과 확장성을 지닌 고성능 서버를 요구하고 있다. 그러나 기존 서버의 하드웨어 자원조차도 불필요하게 자원을 낭비하거나 사용하지 않고 있기 때문에 기업의 입장에서는 하드웨어 자원의 낭비 문제까지도 신경 쓸 수밖에 없게 되었다[1].

또한 IT 기술의 발전은 개인의 생활 방식뿐만 아니라 국내외 기업들의 업무 방식에도 많은 변화를 불러오고 있다. 특히, 기업의 경우 각 구성원마다 컴퓨터를 필요로 하기 때문에 기업의 중요 정보 유출을 원천적으로 막고자 서버 가상화를 이용한 데스크탑 가상화 환경을 도입하고 있는 추세이다. 데스크탑 가상화 환경은 관리자의 관리 능력의 향상과 PC환경 뿐만 아니라 Thin Client, Zero Client 환경에서도 적용이 가능하기 때문에 경제적이다. 또한 원격지에서도 지속적으로 업무를 가능하게 함으로써 업무의 효율성 증대와 같은 긍정적인 효과를 가져오며, 2장 관련 연구에서 가상화 서비스에 대해서 좀 더 자세히 서술한다.

서버 가상화란 서버에 응용프로그램과 데이터를 두고 필요 할 때마다 접속해서 사용하는 방식이며, 데스크탑 가

상화 환경이란 한 대의 물리적 서버에 여러 개의 VM을 운영하고 사용자의 요구 조건에 맞는 운영체제나 응용프로그램을 설치 한 후 사용자에게 VM을 배포하여 사용자가 RDP(Remote Desktop Protocol) 내부 망을 통해 서버에 접속을 한 후 해당 VM을 이용함으로써 기업의 입장에서는 정보 유출을 차단하고 관리자 입장에서는 편리함을 제공하는 환경을 말한다[2, 3, 4, 5].

하지만 가상화 환경은 특성상 가상 서버에 의존하는 비중이 높다. 따라서 서버의 안전성을 보장하기 위해서는 서버에 문제가 생겼을 때, 중단 없이 가상머신의 이동을 지원하는 실시간 이동(Live Migration), 부하 분산 그리고 서버의 자원을 할당 또는 적절하게 배치해서 운영 할 수 있도록 하는 프로비저닝[6] 기술이 그러한 것들이다.

본 논문의 구성은 다음과 같다. 2장 관련연구에서는 가상화 기술에 대해서 살펴보고 가상 서버의 안정성을 위한 부하 분산과 프로비저닝에 대해서 관련 연구 한다. 3장에서는 윈도우 운영체제 환경에서 가상 서버의 과부하를 최소화하고 VM의 신속한 배포를 위한 프로비저닝 방안에 대해서 서술하고, 부하 분산 방안을 제안한다. 4장에서는 제안하는 방안에 대한 기대효과를 서술하고 마지막으로 5장에서 향후 연구 과제에 대해서 알아본다.

2. 관련연구

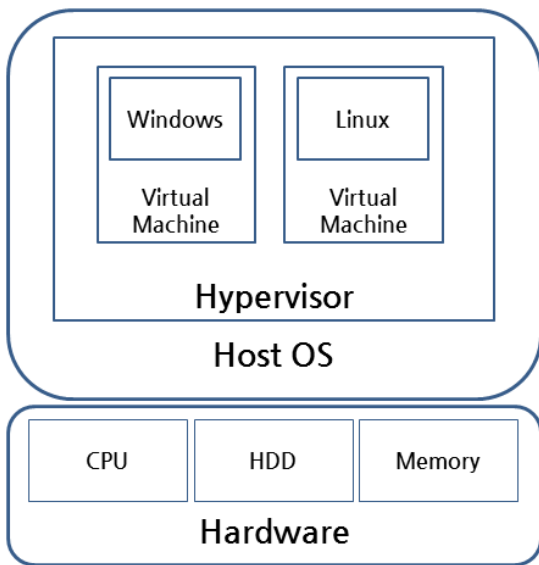
2.1 가상화

가상화 기술은 거의 모든 IT 자원에 대해서 개발되고

※ 본 연구는 지식경제부 및 한국인터넷진흥원의 “고용계약형 지식정보보안 석사과정 지원사업”의 연구결과로 수행되었음.

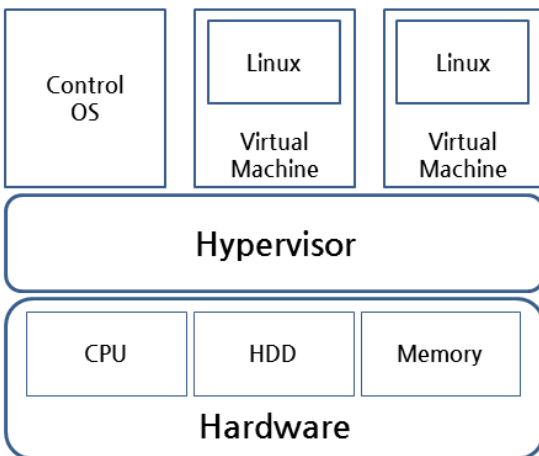
있다. 특히 하드웨어, 네트워크 성능이 향상되기 시작하면서 스토리지, 네트워크, 서버 등 IT 자원의 모든 분야에서 사용되고 있다. 가상화는 하나 혹은 여러개의 물리적 시스템에서 여러개의 논리적 시스템을 제공하는 기술로써 소규모 조직보다 기업이나 정부 등과 같은 대규모 환경에서 비교적 적합하며, 최근에는 시스템의 신뢰성, 확장성을 보장하기 위해 널리 활용되기 시작하였다[7, 8].

가상화 기술의 종류로 CPU가상화 기능의 직접 사용 여부에 따라 Hyper-V와 같은 전 가상화(Full Virtualization)[9]와 Xen[10]에서 사용하는 반 가상화(Para Virtualization)[11] 환경이 있다.



(그림 1) 전 가상화(Full-Virtualization)

전 가상화는 하드웨어를 완전히 가상화하는 것으로 게스트 운영체제를 수정 할 필요가 없고, 다양한 운영체제를 사용할 수 있다. 반 가상화 환경은 하드웨어에 대한 가상화 없이 하이퍼바이저가 제공하는 API를 통해 OS를 제어한다.



(그림 2) 반 가상화(Para-Virtualization)

2.2 부하 분산

부하 분산 기술로는 정적 부하 분산(Static Load Balancing), 라운드 로빈 부하 분산(Round Robin Load Balancing), 확산 부하 분산(Diffusive Load Balancing), Shortest Queue Load Balancing, Local Approach, Global Approach 등이 있다.

정적 부하 분산은 요청이 오면 정적으로 서버에 연결하여 서비스를 할 수 있도록 하여, 한 클라이언트는 언제나 같은 서버에서 서비스를 받게 되고 다른 서버에는 연결되지 않는다. 즉, 각 서버들은 서로 독립적으로 동작하게 된다. 라운드 로빈 부하 분산은 클라이언트가 라우터로 서버 연결 메시지를 보내게 되면 라우터는 라운드 로빈 방식으로 서버에 메시지를 전달하여 연결한다. 확산 부하 분산 방식은 클라이언트 서비스 요청 메시지가 라운드 로빈 방식으로 라우터에게 전달되고 서버로 요청이 들어간다. 요청 받은 서버는 부하가 적은 다른 서버로 전달된다[12].

Shortest Queue Load Balancing은 클라이언트의 요청을 부하가 가장 적은 서버로 이동시킨다. 앞의 3개의 기술과 다른 점은 클라이언트가 서비스를 받는 동안에도 이동이 이루어 질수 있다는 것이 특징이다[13, 14].

Local Approach 방식은 부하가 많아진 서버의 서비스를 다른 서버로 옮겨서 서버의 데이터 부하를 분산시킨다. 이 방식에서 서비스를 이동시킬 서버에는 다른 서버의 정보만 필요로 하기 때문에 서비스 이동에 대한 오버헤드가 적지만 다른 서버의 부하가 많아지면 분산 시킬 수 없다는 단점이 있다[15, 16].

Global Approach 방식은 모든 서버의 정보를 한 시스템에 유지되며, 서버는 그 정보를 바탕으로 부하를 분산 시켜야겠다고 판단되면 서버의 부하를 전체 서버에 분산시킨다. 서비스 전체에 대해 부하 분산을 할 수 있지만 부하를 분산 시 오버헤드가 커지는 단점이 있다[17].

2.3 프로비저닝

프로비저닝 기술은 데이터 저장 장치의 사이즈가 낭비되는 것을 막아주고 실제 사용 영역만 할당함으로써 스토리지 소유 비용을 극대화 할 수 있는 기술이다. 즉, 컴퓨팅 자원 및 서비스를 사용자에게 제공하기 위한 단계로써, 사용자의 요구 사항(하드디스크 용량, 메모리 자원 등)을 가상머신에 반영하여 ‘할당’하고 ‘배포’하며 일정 기간이 지나면 가상머신을 가상 서버에서 ‘삭제’하여 시스템 자원을 효율적으로 사용할 수 있도록 지원하는 일련의 행위를 주도하는 기술이다[18].

프로비저닝 기술을 도입하면 자원 활용률을 향상 시킬 수 있을 뿐만 아니라 가상화 환경 도입 시 초기 과다 용량 할당으로 인한 낭비요소를 최소화 할 수 있다.

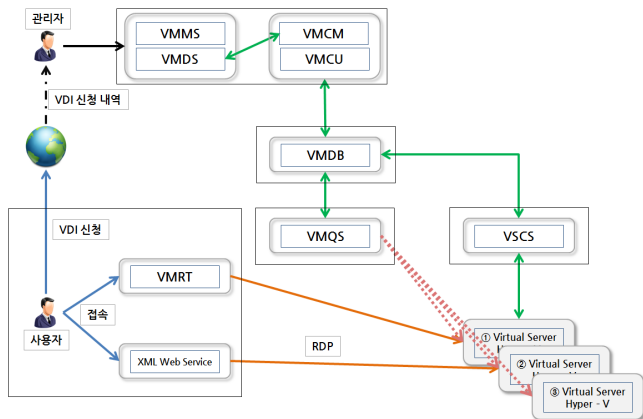
따라서 본 논문에서는 윈도우 환경에서 별도의 모듈에 프로비저닝 기술을 통해 가상 서버를 최적화 시키는 방법에 대해서 서술하고, 최적화된 가상 서버에 가상머신을 할당 하여 서버의 부하를 최소화 시키는 부하분산에 대해서

제안하고자 한다.

3. 설계 및 제안하는 방안

제안하는 환경은 Windows 기반에서 적합하며 (그림3)에 있는 VMQS(Virtual Management Quota System)는 Hyper-V Server에 할당되어 있는 가동VM의 수를 파악하여 최적화된 Hyper-V Server에 사용자VM을 할당하고, VMCM(Virtual Machine Copy Master)과 VMCU(Virtual Machine Copy User)를 통해 VM의 신속한 배포, 가동 그리고 삭제하는 프로비저닝 과정을 통해 Hyper-V Server의 부하를 최소화 할 수 있다.

3.1 제안하는 VDI 환경



(그림 3) 제안하는 VDI환경과 부하 분산

① 사용자는 먼저 홈페이지를 통해 자신이 이용하고자 하는 하드디스크의 용량, 메모리, 운영체제 그리고 설치될 응용프로그램 등 VDI 환경 정보를 신청한다.

② 관리자는 VDI 신청 정보에 따라 가상머신 배포 시스템(VMDS;Virtual Machine Distribution System)의 가상머신 복사 마스터 (VMCM) 모듈을 통해 사용자VM을 생성한다.

③ 사용자VM이 생성 되면 가상머신 복사 사용자(VMCU) 모듈을 통해 가동VM이 생성된다. ①②③은 모두 단일 서버에서 동작 할 수도 있고 모듈을 나누어서 여러 개의 서버에서 작업을 처리 할 수 있다. 또한 이러한 작업 정보는 모두 VMDB(Virtual Machine Database)에 정해진 스키마에 따라 저장이 된다.

④ 이렇게 해서 생성된 사용자VM은 가동 대기 중에 있다가 사용자가 자신의 클라이언트에서 홈페이지를 통해 미리 설치한 가상머신 원격 터미널(VMRT:Virtual Machine Remote Terminal)이나 홈페이지에서 RDP(Remote Desktop Protocol)를 이용하여 Hyper-V Server에 자신의 가상머신으로 최초 접속 시도를 한다.

⑤ 사용자가 자신의 가상머신으로 접속을 시도 할 때 가

상머신 할당 시스템(VMQS)은 가상머신 제어 시스템(VSCS;Virtual System Control System)을 이용하여 VMDB에 저장되어 있는 Hyper-V Server의 상태정보를 파악하여 가동 대기 중인 사용자VM을 최적의 Hyper-V Server에 가동시킴으로써 사용자 최초 접속이 이루어진다.

⑥ 마지막으로 사용자가 가동 머신의 접속을 종료하면 Hyper-V Server에서 운영 중인 가동VM을 사용자VM 상태로 되돌린다.

각 모듈의 상세한 기능은 <표1> 표시되어 있으며, 가상머신 관리 시스템(VMMS;Virtual Machine Management System)은 Hyper-V Server위에 운영 중인 Hyper-V를 관리하고 그 위에서 운영 중인 가동VM을 관리하는 응용 프로그램이다.

<표 1> 각 모듈 별 기능

명칭	기능
VMCU	사용자VM을 가동VM으로 생성.
VMRT	클라이언트 원격 접속 터미널.
VMMS	Hyper-V Server의 VM을 관리하며, 특히 마스터VM 관리에 사용.
VMDS	마스터VM을 이용한 사용자VM을 관리.
VMCM	마스터VM을 사용자VM으로 생성(환경파일 수정).
VMQS	가동할VM을 최적의 Hyper-V Server로 할당.
VSCS	가동할VM을 등록하고 시작하며, 가동VM을 제거하고, Hyper-V Server의 가동VM의 목록 및 상태를 전송.

4. 기대효과

제안하는 데스크탑 가상화 환경에서 마스터VM, 사용자VM, 가동VM으로 나누어 가상머신을 배포한 이유는 다음과 같다. 사용자가 최초 Hyper-V Server와 연결을 종료하면 가동VM은 Hyper-V Server에서 삭제된다. 사용자VM 상태로 돌아가게 되면 사용 중에 있던 데이터 정보들이 최초 배포 상태로 돌아간다. 즉, 사용자VM으로 돌아가게 함으로써 VM의 상태를 보존하고 다시 접속을 시도할 때 사용자 대기 시간을 감소시킬 수 있다. 그리고 운영 중인 가동VM에 설치되는 에이전트 없이 별도의 모듈이 가상서버의 상태를 VMDS에 제공하고 그 정보를 바탕으로 VMQS가 사용자VM을 최적화된 Hyper-V Server에 할당하기 때문에 가동VM의 부하를 줄이면서 전체적으로 효율적인 부하 분산 기능도 제공할 수 있다.

5. 결론 및 향후 연구

본 논문에서 VMCM, VMCU를 이용하여 가상 머신 배포 과정을 마스터VM, 사용자VM 그리고 가동VM으로 나누었다. 이는 사용자들의 가상화 환경 접속 요청 시 대기 시간을 최소화 시키고 사용자의 접속 종료 시 가동VM을 Hyper-V Server에서 삭제시킴으로써 가상서버의 쓸모 없는 시스템 자원 낭비를 최소화 시킬 수 있는 방법을 제안하였다. 또한 VMQS가 VMDB 정보를 통해 최적화 된 Hyper-V Server에 사용자VM을 할당함으로써 좀 더 효율적인 부하 분산 방법을 제안하였다.

향후 연구 계획으로는 제안하는 환경에서의 기대효과를 검증한다. 또한 데스크탑 가상화 환경을 제공하는 각 기업들의 솔루션에 동작 방식에 대해서 알아보고, 마지막으로 가상화 환경에서 발생 할 수 있는 보안 이슈에 대해서 연구 할 계획이다.

참고문헌

- [1] 권혁진, 최영규, “분산 개체 기반 로드 밸런싱 서버 시스템의 설계 및 구현”, 한국정보보호학회논문지, 제3권 제4호, 2005.
- [2] 김인혁, 김태형, 김정환, 임병홍, 엄영익, “시스템 보안을 위한 가상화 기술 활용 동향”, 정보보호학회지, Vol.19, No. 2, pp.26-34, 4. 2009.
- [3] 류한석, “Server based computing”, 소프트뱅크미디어랩, 5. 2007.
- [4] Heiser Gernot, Elphinstone, Kevin Kuz, Ihor Klein, Gerwin, Petters, and Stefan M. "Towards Trustworthy Computing Systems: Taking Microkernels to the Next Level", perating System Review 41 (3), pp.3-11, 7. 2007.
- [5] 임세정, 김광준, 강태근, “가상 데스크탑 기반에 응용 프로그램 가상화”, 한국전자통신학회, pp.505-601, 12. 2010.
- [6] Provisioning, <http://en.wikipedia.org/wiki/Provisioning>
- [7] Heiser Gernot, Elphinstone, Kevin Kuz, Ihor Klein, Gerwin, Petters, and Stefan M., " Towards Trustworthy Computing System: Taking Microkernels to the Next Level", Operating Systems Review 41(3), pp.3-11, 7. 2007.
- [8] VMware, Inc., "Understanding Full Virtualization, Paravirtualizaion, and Hardware Assist", White Paper, 11. 2007.
- [9] Full Virtualization, http://en.wikipedia.org/wiki/Full_virtualization
- [10] Xen, <http://en.wikipedia.org/wiki/Xen>
- [11] Para Virtualization, <http://en.wikipedia.org/wiki/Paravirtualization>
- [12] 전상일, “Xen기반 가상 데스크탑 서비스를 위한 부하 분산 정책”, 상명대학교 대학원, 석사학위논문, 12. 2007.
- [13] R. D. Nelson, “An Approximation to the Response Time for Shortest Queue Routing”, ACM SIGMETRICS Performance Evaluation Review, Proceedings of the 1989 ACM SIGMETRICS international conference on Measurement and modeling of computer systems SIGMETRICS '89, Volume 17 Issue 1, pp.181-189, 4. 1989.
- [14] Milan E, "Simulation of Load Balancing Algorithms", ACM SIGCSE Bulletin, Volume 34, Issue 4, pp.138-141, 12. 2002.
- [15] B. Ng., "A Multi-Server Architecture for Distributed Virtual Walkthrough", In Proceedings of the ACM Symposium on Virtual Readlity Software and Technology(VRST'02), pp.163-170, 11. 2002..
- [16] S. Pekkola, "Collaborative Virtual environments in the year of the dragon", In Proceddings of the ACM International Conference on Collaborative Virtual Environments(CVE'00), pp.11-18, 2000.
- [17] J. C. S. Lui, "An efficient partitioning algorithm for distributed virtual environment systems", IEEE Transactions on Parallel and Distributed System, 13(3):193-211, 3. 2002.
- [18] 정순기, 정만현, 조재익, 손태식, 문종석, “클라우드 컴퓨팅 가상화 보안을 위한 아키텍처 구성 및 기능 분석 연구”, 보안공학연구논문지, 제8권, 제5호, 10. 2011.