

IPv6 기반의 모바일용 가상 저장장치 드라이버 구현 및 응용

임효택[†], 최세봄^{**}, Vinay Singh^{***}, 차경환^{****}

요 약

IPv6는 현재 사용하고 있는 IPv4의 단점을 개선한 차세대 인터넷 프로토콜이다. Mobile IPv6는 이러한 IPv6의 기능을 그대로 가지면서 모바일 장치들의 이동성을 효과적으로 제공한다. iSCSI(Internet Small Computer System Interface)는 TCP/IP 네트워크를 통해 원격의 스토리지를 마치 자신의 로컬 상에 있는 블록 장치처럼 접근하는 기술이다. iSCSI는 이 같은 접근을 위해 표준 이더넷 스위치와 라우터를 사용하므로 기존의 수많은 이더넷 기술의 적용은 물론 거리 제한이 없는 스토리지 확장을 가능하게 한다. 본 논문에서는 이 점에 착안, 유선 기반의 원격 스토리지 접근을 목적으로 설계된 iSCSI를 PDA와 같은 임베디드 리눅스 장치에 탑재하고자 한다. 이를 통해 IPv6무선 네트워크상에서 모바일 기기가 가지는 저장 공간의 한계점을 극복하고자 하며 이에 대한 응용으로 가상 저장장치 기반의 모바일 학습 서비스 시스템을 구현하고 성능을 분석 하고자 한다.

Implementation of IPv6 based Virtual Storage System for Mobile Devices and Its Application

Hyotaek Lim[†], Sae-bom Choi^{**}, Vinay Singh^{***}, Kyung-Hwan Cha^{****}

ABSTRACT

IPv6 is the internet protocol for next generation which can solve the problem of IPv4 protocol. Mobile IPv6 efficiently provides the mobility for mobile devices holding the properties of IPv6. iSCSI(Internet Small Computer System Interface) is a block-oriented storage access protocol that enables a user to recognize a remote storage as their own local block device through general TCP/IP networks. Since iSCSI uses a standard Ethernet switch and router for this kind of access, it can not only be applied to Ethernet technologies, but can also be used to create a storage networking system without any distance restrictions that can equally be applied to a wireless network environment. Accordingly, this paper presents an alternative approach to overcome the limited storage space of mobile devices based on the IPv6 iSCSI initiator driver which was originally designed for wired networks. Also, this paper describes about the implementation and performance analysis of mobile education service system using virtual storage devices in IPv6 wireless networks.

Key words: iSCSI(iSCSI), IPv6(IPv6), Mobile(모바일), Storage(스토리지), Embedded Linux(임베디드 리눅스)

※ 교신저자(Corresponding Author) : 임효택, 주소 : 부산광역시 사상구 주례2동 산69-1(617-716), 전화 : 051)320-1718, FAX : 051)327-8955, E-mail : htlim@gdsu.dongseo.ac.kr
접수일 : 2006년 7월 20일, 완료일 : 2006년 10월 23일

[†] 정회원, 동서대학교 컴퓨터정보공학부

^{**} 한국산업기술평가원 정보화기획단

(E-mail : saebom@itep.re.kr)

^{***} 동서대학교 디자인&IT전문대학원

(E-mail : vinay@dit.dongseo.ac.kr)

^{****} 동서대학교 컴퓨터정보공학부

(E-mail : khcha@dongseo.ac.kr)

1. 서 론

인터넷 사용인구의 폭발적인 증가로 인해 기존의 IPv4주소가 향후 몇 년 뒤에는 고갈될 것이라 예상되고 있다. 이러한 문제점을 해결하기 위해 차세대 인터넷 프로토콜인 IPv6를 1996년 IETF(Internet Engineering Task Force)가 표준으로 제정하였다. 이와 더불어 기존 네트워크상에서의 서비스를 IPv6 망에서도 적용하는 연구가 지속해서 이루어지고 있다.

현재 무선 네트워크를 기반으로 하는 모바일 장치들은 제한된 전력공급과 이동성을 목적으로 한 물리적인 제약으로 인해 그 성능에 있어서 어느 정도 한계를 가지고 있다. 이러한 제약사항을 가지고 있음에도 불구하고 모바일 장치는 지속적인 무선 네트워크 보급과 기반 기술 발달에 힘입어 다양한 분야에서 활용되고 있으며, 그 수요 역시 지속적인 증가를 보이고 있다. 모바일 장치가 가지는 근원적인 제약성을 극복하고자 많은 연구가 활발히 이루어지고 있으며, 그 중 사용자층으로부터 지속적인 요구를 보이고 있는 것이 충분한 전력공급과 저장 공간 제약의 극복이다. 저장 공간 확장을 위한 대안으로는 CF(Compact Flash)나 SD(Smart Drive)와 같은 장치가 이용되고 있으나 이 같은 장치들은 상대적으로 고가이고 대용량의 스토리지를 필요로 하는 분야에서는 한계가 있으며 전력 사용면에서도 그리 효율적이지 못하다[1]. 또한 IPv6를 이용하여 이동성을 제공하고자 하는 연구가 활발히 진행되고 있다. Mobile IPv6는 IPv6의 기능들을 그대로 이용하면서 이동성을 제공하고자 하기 때문에 Mobile IPv4 보다 효과적으로 이동성을 지원할 수 있으며 탁월한 규모 확장성을 지니고 있다. 즉, Neighbor Discovery와 Address auto-configuration 기능이 IPv4에서의 외부 에이전트 기능을 대신함으로써 IPv4에서는 존재해야만 했던 일부 시그널 메시지와 에이전트를 제거하였다. 또한 경로 최적화를 위한 프로토콜이 기본 기능으로 제공되고 있다.

따라서 본 논문에서는 상대적으로 저비용의 특성을 가지면서도 높은 확장성과 대용량 서비스가 가능한 IPv6무선 네트워크 기반의 원격 스토리지 접근 방법과 이러한 모바일 장치들의 이동성에 초점을 두고 진행하였다. 2003년 4월 IETF에 의해 정식으로 표준화가 이루어진 iSCSI 프로토콜을 이용하여 모바

일 장치의 주된 제약사항인 저장 공간에 대해 보다 근본적인 해결방안을 모색하고, 이 기술을 Mobile IPv6무선 네트워크를 이용하는 모바일 장치에 원격 스토리지 시스템을 연결하여 서비스하는 동영상 학습 서비스 시스템의 가능성을 가늠해 보겠다.

2. 관련 연구

2.1 IPv6

현재의 주소체계인 IPv4는 사용자의 증가로 인한 주소 부족현상과 보안 등의 문제로 이에 대한 해결책이 필요한 시점이다. IPv4에서도 동적 IP 할당을 이용한 유휴 IP 자원의 활용과 사용하지 않는 주소의 재활당, NAT(Network Address Translator)의 이용 등 나름대로 해결책을 가지고 있다. 하지만 동적 IP 할당의 경우에는 IP 번호가 바뀔므로 인터넷 연결 통로가 고정되지 못한다는 단점을 가지고 있다. 사용하지 않는 IP 번호의 재사용 역시 사용하지 않는 IP의 수보다 더 많은 IP 수요가 발생함으로 근본적인 한계를 지닌다. NAT의 경우는 사설IP 라는 특징 때문에 양방향 인터넷 통신에 많은 제약이 생긴다. 많은 인터넷 서비스, 온라인 게임 등은 사설 망에서 아예 사용 불가능 할 수도 있다. 양방향 인터넷 서비스기술인 P2P 역시 NAT 하에서는 제약이 가해진다.

차세대 인터넷 프로토콜인 IPv6는 기존 32비트 주소 체계로 표현되는 IPv4주소를 128비트 주소 체계로 확장하여 기존 인터넷이 안고 있는 근본적인 주소 고갈 문제를 해결하고, IPv6기본 헤더를 단순화하여 패킷 처리의 성능 개선과 확장 헤더의 정의를 통한 이동성 지원, 보안 기능 제공, 서비스 품질 기능 등이 개선될 수 있도록 설계된 프로토콜이며, 특히 주소 자동설정 기능을 통한 플러그 앤 플레이 기능을 제공함으로써 손쉽게 이용자의 단말을 네트워크에 접속시킬 수 있도록 디자인되어 있다. IPv4와 IPv6의 차이점 비교는 [표 1]에서 쉽게 알 수 있다.

2.2 Mobile IPv6

Mobile IPv6의 표준화 작업은 IETF Internet area의 mip6 워킹 그룹에서 진행하고 있다. Mobile IPv6는 IPv6 네트워크 계층에서의 모바일 단말의 이동성을 지원하고자 하는 프로토콜로써, 자신의 영구적인

표 1. IPv4와 IPv6 차이점 비교

	IPv4	IPv6
주소체계	32비트 (4 octet)	128비트 (16octet)
주소갯수	Over 109 (42억개)	Over 1038 (3.40E38 개)
패킷헤더	Variable Size	Fixed Size (40 octet)
주소할당방법	A,B,C,D(멀티캐스트) 클래스 CIDR	CIDR 기반 계층적으로
주소유형	유니캐스트 (unicast) 멀티캐스트 (multicast) 브로드캐스트 (broadcast)	유니캐스트 - link-local address - site-local address - global address 멀티캐스트 애니캐스트(anycast)
QoS	Defined(ToS), but not generally used	Flow Label Traffic Class
Security	Limited	IPsec built-in
Configuration	Manual Configuration	uto-configuration

홈 주소를 갖고 인터넷을 움직일 수 있도록 하는 라우팅 기술을 제공한다. Mobile IPv6기술은 IP layer 상위의 프로토콜에 투명하게 동작하며, 활성화된 TCP 연결과 UDP 포트 바인딩의 끊임없이 IPv6 호스트의 이동성을 제공하는 기술이다.

IPv6 노드(MN)가 홈 네트워크를 벗어나 다른 네트워크에 접속되면 해당 노드는 방문한 네트워크에서 임시로 사용할 주소인 CoA(Care of Address)를 얻게 된다. 이 주소를 자신의 홈 네트워크상에 위치한 홈 에이전트(HA)에게 등록한다. CoA를 홈 에이전트에 등록한 후에 이동노드의 홈 주소를 목적지로 가지는 패킷이 전달되면 홈 에이전트가 이동노드를 대신하여 패킷을 수신한다. 홈 에이전트는 수신한 패킷을 CoA를 목적지로 터널링하여 이동노드가 위치한 네트워크로 전달하며, 이로써 이동노드는 원래 패킷을 얻어낸다. 반대로 이동노드가 상대노드(CN)로 패킷을 전송할 경우에 최초의 패킷전송은 홈 에이전트의 역터널링을 거쳐서 상대노드로 전달된다. 이때 상대노드는 RR(Return Routability)라는 방식을 통해 이동노드의 CoA를 등록하며, 상대노드에 CoA가 등록된 이후에는 두 노드가 홈 에이전트를 거치지 않고 직접 통신을 수행할 수 있다[2,3]. [그림 1]은 Mobile IPv6의 동작을 시스템 개발환경의 예를 들어 간략하게 보여주고 있다.

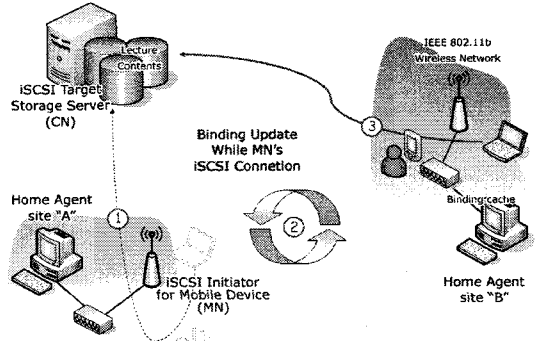


그림 1. Mobile IPv6의 동작

2.3 iSCSI

현재 가장 널리 쓰이고 있는 스토리지 접근 프로토콜은 SCSI이다. SCSI는 수 미터로 제한된 전송 거리와 개별 스토리지 장치의 연결 개수가 한정되어 있다. 이 같은 점을 극복하고자 파이버 채널 네트워크를 이용한 SAN(Storage Area Network)이 등장하였다. FC기반의 SAN은 자체적인 FC Switch와 FC-AL(Fibre-Channel Arbitrated Loop)과 같은 방법을 이용하여 연결 가능한 장치 노드의 개수를 상당히 증가시켰고 낮은 지연율을 가지는 파이버 채널을 이용함으로써 전송거리 역시 수 킬로미터의 영역까지 확장시켜 놓았다[4]. 하지만 스토리지의 확장 영역은 제한되어 있으며, 시스템이 복잡함과 높은 유지비로 인하여 유연성이 떨어진다. 이를 상대적인 저비용과 함께 스토리지 네트워킹의 영역을 더 넓은 곳으로 확장시킨 기술이 바로 iSCSI(Internet SCSI)이다 [5,6,7]. iSCSI는 IETF의 작업 그룹인 IPS(IP Storage)에서 개발된 프로토콜 중의 하나로써 스토리지상의 SCSI 블록 데이터를 기존의 물리적인 파이버 채널 네트워크 대신 널리 퍼져있는 TCP/IP네트워크를 이용하여 데이터를 전송하는 SCSI 전송 프로토콜이다[1,8,9,11].

[그림 2]은 표준 규약을 따르는 iSCSI의 프로토콜 스택을 IPv6를 적용시켜 나타낸 것이다. 본 논문에서는 이 같은 IPv4무선 대신 IPv6무선환경을 이용하여 Target과의 통신을 수행하였다.

3. 시스템 구현

본 단원에서는 동영상 학습 서비스를 위해서 모바일

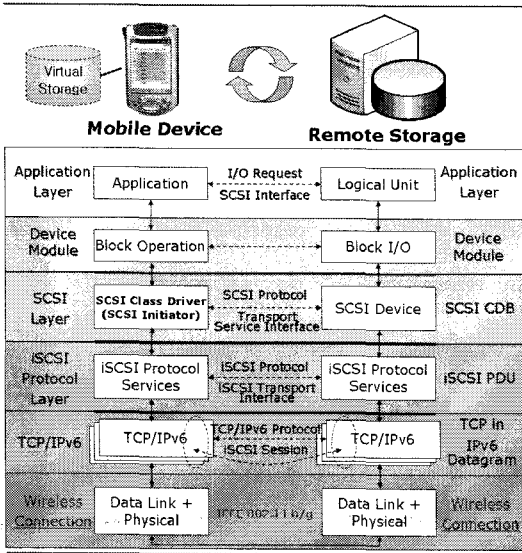


그림 2. iSCSI 프로토콜 스택

일 장치에 탑재할 iSCSI Initiator 드라이버 개발환경과 시스템의 구현 사항에 대해 서술한다.

3.1 시스템 개발환경

3.1.1 Mobile IPv6 무선 네트워크 구성

Mobile IPv6 무선 네트워크 환경을 구성하기 위해서 [그림 3]과 같이 교내에 IPv6 Testbed를 구축하였다.

3.1.2 모바일 장치 및 운영체제 선택

모바일 장치로는 개인용 휴대 단말기인 PDA를 선택 하였으며 이는 IPv6무선 네트워크 장치를 통해

범용 IP 네트워크에 접근하게 되고 이를 통해 원격의 서버측 스토리지인 iSCSI Target을 제어할 수 있게 된다. 서버 측에 해당하는 iSCSI Target Driver는 UNH-iSCSI Target[10]을 IPv6 무선 네트워크에 맞게 수정하였고, PDA는 COMPAQ사의 iPAQ-h3850 모델을 사용하였다.

모바일 장치의 OS로는 공개 임베디드 리눅스 프로젝트로써 진행 중인 Familiar 배포판을 사용하였다.

[그림 4]는 이 같은 장비를 이용하여 IPv6 무선 네트워크상에서 원격 스토리지에 접근하는 환경을 도식화 한 것이다.

3.1.3 개발환경 구성

임베디드 장치의 특성상 개발환경을 목적 시스템에 직접 구성할 수는 없으므로 해당 플랫폼에 종속적인 교차 개발환경 구축이 필요하다. 이를 위해 교차 개발환경을 binutils-2.9.5.0.22와 gcc-2.95.2 그리고 glibc-2.1.2을 이용하여 구성하였다.

3.2 시스템 구조 및 동작과정

UNH-iSCSI Initiator 드라이버는 IP 네트워크를 통한 커널 레벨의 저수준 블록 입출력을 지원하기 위해 LKM(Loadable Kernel Module)방식을 사용하였다. LKM은 커널레벨에서 필요로 하는 기능을 모듈의 형식으로 적재가 가능하므로 커널의 동적인 구성이 가능하다. 이 같은 사항은 리눅스 기반의 임베디드 장치에서도 적용되며 아래의 [그림 5]과 같은 시스템 계층 설계가 가능하다. 시스템은 크게 사용자 레벨과 커널 레벨로 구분되어 진다. iSCSI Initiator

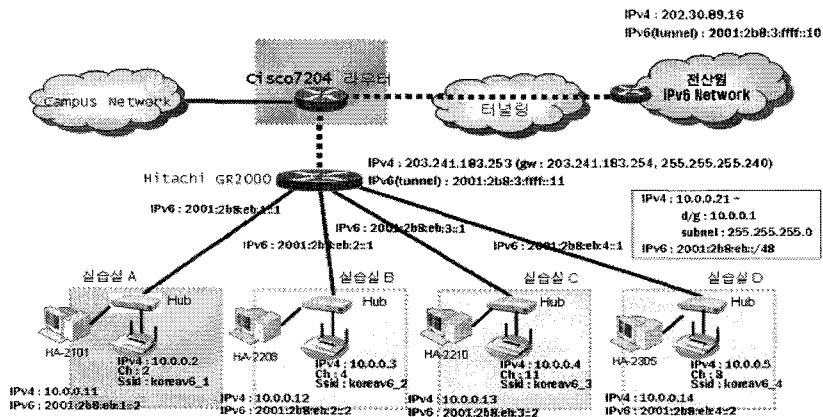


그림 3. IPv6 Testbed

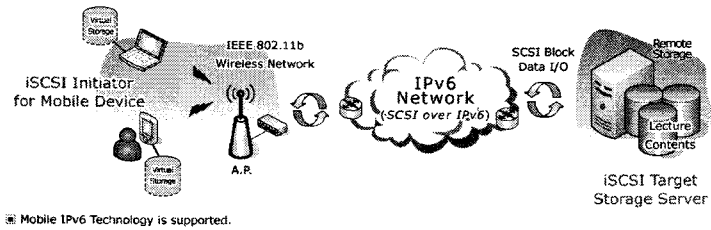


그림 4. 모바일 장치의 원격 스토리지 시스템 구현

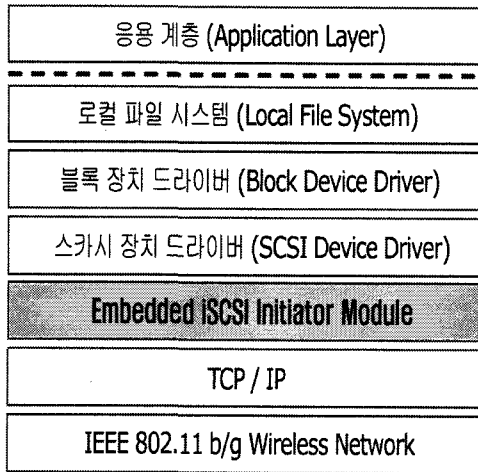


그림 5. 모바일 장치상의 시스템 계층 구조

의 장치 모듈은 사용자 계층에서 발생하는 SCSI 명령을 기반으로 IP 네트워크에서 SCSI 데이터를 동기화하기 위해 iSCSI PDU(Process Data Unit)를 생성하게 되고, 이는 하위의 TCP계층을 거쳐 원격지의 iSCSI Target module과 통신이 이루어지게 된다. 이 같은 방법으로 사용자 계층에서는 마치 자신의 로컬 상에 SCSI 장치가 존재하는 것처럼 인식하게 되는 것이다.

3.2.1 동작 기반 환경

일반적으로 모바일 장치는 SCSI 드라이버를 필요로 하지 않는다. 또한 File System 역시 JFFS와 같이 자신의 하드웨어에 특화된 파일시스템만이 존재한다. 그러므로 모바일 장치는 원격 스토리지의 직접적인 제어를 위해서 ext2나 ext3와 같은 데스크탑용 파일시스템 모듈이 적재될 필요성이 있다. 가상 스카시 호스트를 등록하기 위해서는 이러한 의존 모듈인 파일시스템 모듈과 SCSI 모듈이 커널 상에서 전제되어야 한다. 이와 같은 모듈이 바탕이 되어야만 모바일 장치상에서도 데스크탑의 HBA(Host Bus Adapter)

와 같은 역할을 수행하게 되는 가상의 iSCSI 호스트 등록이 가능하며 이를 통해 실제적인 SCSI 명령을 수행할 수 있는 기반이 마련된다.

3.2.2 FFP(Full Feature Phase) 동작 과정

iSCSI 프로토콜의 세션 연결과정의 여러 단계 중 본 단원에서는 모바일 장치 상에서 실제 SCSI 명령이 수행되는 iSCSI FFP 내부적인 수행 구조에 대해 기술한다. 이것은 유선상의 iSCSI 수행 구조와 거의 일치한다.

(1) iSCSI Read Process

모바일 장치 상에서의 데이터 읽기 명령에 의해 가상 iSCSI 호스트는 원격의 iSCSI Target에게 읽기 요청 명령 셋을 보내게 된다. 이에 iSCSI Target은 정해진 규약에 따라 데이터를 처리하고 이에 대한 응답을 하게 된다. [그림 6]는 그 동작 과정을 나타내며 데이터를 읽기 위한 요청에 대한 허용명령(iSCSI R2T) 전달부터 시작됨을 보인다. 모든 입출력에서는 무선 네트워크 전송환경을 이용한다.

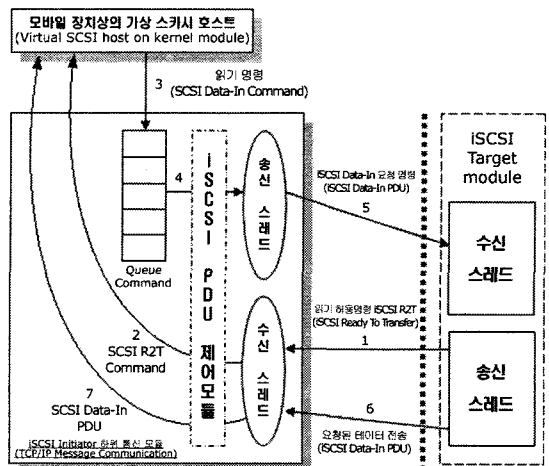


그림 6. iSCSI Read 동작 과정

(2) iSCSI Write Process

쓰기 명령 역시 수행을 위한 준비 단계로써 쓰기 요청 명령 셋을 Target 측에 전달한 뒤에 실제 입출력이 진행되어 진다. [그림 7]에서 iSCSI Write의 동작 과정을 그림으로 도식화 하였다.

3.3 구현 사항

무선 네트워크 기반의 iSCSI Initiator는 물리적 통신 구조를 제외하고는 x86 기반의 그것과 기본적인 동작과정이 동일하다. UNH-iSCSI의 Initiator 드라이버 소스를 참조하여 작업을 진행하였고, 기존의 IPv4무선 네트워크 기반의 iSCSI Initiator를 IPv6무선 네트워크 환경에 맞게 수정하여 이를 앞서 구축되어진 교차 개발환경을 이용하여 구성하였다.

여기서 선택한 목적 디바이스는 Strong-ARM (Advanced RISC Machine) 계열의 장치이므로 Familiar 프로젝트에서 사용한 v7.0.2의 해당 커널 버전인 linux-2.4.19-rmk6-pxa1-hh30을 이용하여 iSCSI Initiator 모듈이 동작할 수 있는 커널로 재구성하였다. 제어를 위한 파일시스템과 임베디드 장치 상에서 표준 SCSI 명령 집합을 생성하기 위한 환경이 구성되어 iSCSI Initiator를 탑재하였고 해당 장치는 원격지의 스토리지를 마치 자신의 로컬상의 존재하는 SCSI 디스크로 인식하였다[1]. 이로써 임베디드 리눅스가 탑재된 모바일 장치 PDA에서 원격지의 SCSI 스토리지 에 저장된 동영상 강의를 볼 수 있다. [그림 8]는 PDA에서 iSCSI 프로토콜을 이용하여 8기

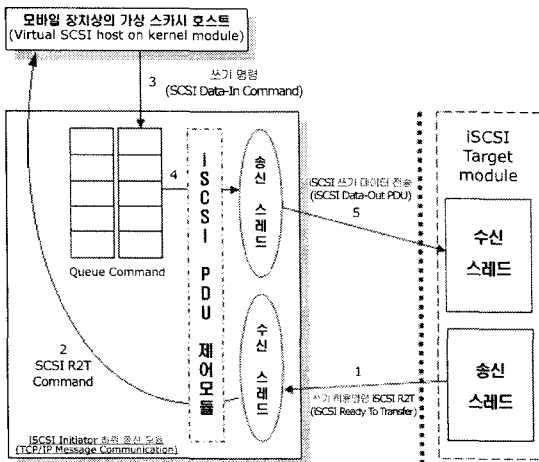


그림 7. iSCSI Write 동작 과정

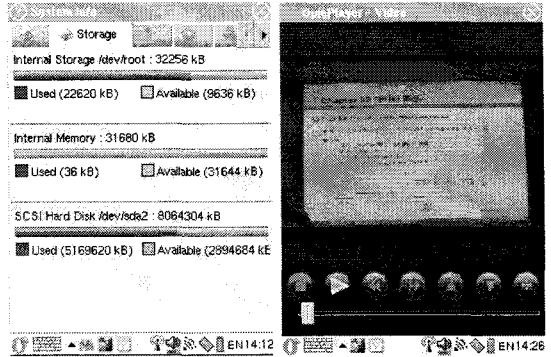


그림 8. iSCSI 로컬 디스크 인식 및 모바일 동영상 강의 서비스

가 SCSI 디스크를 로컬 스토리지로 인식한 화면과 이 디스크에 저장된 동영상 강의 파일을 이용하여 동영상 강의 서비스 중인 화면을 캡처한 모습이다.

4. 성능분석

본 장에서는 앞서 구현 되어진 모바일 장치용 iSCSI Initiator 드라이버의 성능 분석과 평가에 대하여 서술한다.

4.1 환경 설정

성능 측정은 구축되어진 IPv6 테스트베드에서 수행하였고, 성능 측정 시에 사용된 스토리지 서버의 장치 정보를 [표 2]에서 나타내었다. iSCSI와 NFS의 성능측정 비교의 과정에서 사용한 서버의 환경은 동일하다.

성능 측정은 PDA상에서 수행하였고 측정시의 데이터의 흐름은 최초 PDA상의 입출력 요구를 시작으로 하여, [그림4]에서 보여진 바와 같이 IPv6 테스트베드 내의 원격 스토리지 장치로 전달되는 과정을 거치게 된다. 모든 성능 측정에 있어서 별도의 보안 설정은 하지 않았다. 이 같은 환경 구성을 이용하여

표 2. 서버 측 장비 정보

CPU	Intel Xeon 1.0GHz CPU
OS / Kernel	Redhat Linux 9.0 / 2.4.22-18
SCSI Interface	Adaptec aic7899 Ultra160
Disk Model	IBM-DDYS-T18350M
Disk capacity	18 GB (10,025 rpm)
iSCSI Target Driver	UNH-iSCSI(ref15)

무선 환경 상에서 파일 기반으로 입출력을 수행하는 NFS와 블록 기반 입출력을 제공하는 iSCSI의 성능 비교를 할 수 있었으며[12], IPv4에서의 성능분석[1]과 마찬가지로 현저한 성능상의 차이는 볼 수는 없었지만 iSCSI가 NFS보다 좀 더 나은 성능을 보이는 것을 볼 수 있었다.

4.2 성능 분석

IPv4 네트워크 에서의 성능분석과 마찬가지로 CPU의 점유율과 데이터의 처리량에 두고 진행하였으며 두 가지의 분석 도구를 사용하였다. 첫 번째로 순차적 데이터 입출력의 성능측정이 용이한 Bonnie++(1.03a)를 선택하였으며, 두 번째로는 다중 스레드를 이용한 입출력 성능 측정 도구인 Tiobench(0.3.3)를 사용 하였다. Bonnie++로는 미리 지정된 크기를 가지는 단일 데이터에 대한 처리량을 비교해 보았고 Tiobench로는 입출력시에 모바일 장치의 CPU 점유율과 데이터의 처리량을 측정하였다. 먼저 [표 3]에서 Bonnie++를 이용한 iSCSI와 NFS의 성능 측정 결과를 나타낸다.

[표 3]을 보면 전반적으로 iSCSI가 NFS보다 좀 더 나은 성능을 보여주는 것을 알 수 있다. 특히 랜덤 Seek Time은 NFS보다 3배정도의 시간차가 났다. 이 것의 결과는 IPv4에서의 성능분석과 마찬가지로 iSCSI는 NFS와는 다르게 파일시스템을 거치지 않고 절대적인 블록 위치 값으로써 데이터를 접근하므로 이 같은 성능상의 차이가 나타나는 것으로 보여진다.

다음으로는 Tiobench의 스레드 방식을 이용한 다중 입출력 성능을 측정해 보았다. 성능분석의 조건으로 2개의 스레드를 이용하였고 스레드당 파일의 크기는 10MB를 주고 블록사이즈는 512B 부터 64KB까지 주어진 뒤 성능분석을 실시하였고 다음과 같은 결과가 나왔다.

표 3. 단일 파일에 대한 처리속도 측정결과 (NFS vs iSCSI)

	File Size (MB)	Sequential Write			Sequential Read		Random Seeks (ms)
		char (KB/s)	block (KB/s)	rewrite (KB/s)	char (KB/s)	block (KB/s)	
iSCSI	50	532	961	335	546	718	86.2
NFS		544	665	211	574	531	
iSCSI	100	536	879	334	544	701	52.6
NFS		548	666	215	565	518	

먼저 [그림 9]의 결과는 iSCSI와 NFS의 입출력 연산에 따른 모바일 장치의 CPU 점유율을 나타낸다. NFS와 iSCSI의 점유율 차이가 그렇게 크게 나지는 않았지만 전체적인 결과를 놓고 볼 때, 모바일 장치 상에서 차지하는 CPU 점유율이 NFS보다는 iSCSI가 좀더 낮은 것을 알 수가 있다. 이는 iSCSI Initiator가 원격에 존재하는 스토리지에 대하여 물리적인 입출력을 위한 연산을 자신이 직접 하지 않고 Target측에 존재하는 실제 SCSI Host가 대신 수행하기 때문에 비롯된 결과로 볼 수 있다. 그러나 읽기 연산의 결과에서 높은 CPU 점유율이 나타난 이유는 모바일 장치 상에서도 자신의 파일 시스템상의 데이터를 iSCSI PDU로 읽어오기 위해 부가적인 연산을 수행하여야 하는 이것은 PDA의 낮은 성능에 의한 것으로 분석된다.

[그림 10]에서는 입출력 블록 크기에 따른 데이터의 평균 지연율을 나타낸다. 여기서 주목할 점이 iSCSI와 NFS의 쓰기연산의 지연율 차이이다. 이것은 앞서 Bonnie++에서 분석된 결과를 보면 알 수 있다. 모든 측정 결과에서 쓰기 지연율이 읽기 지연율

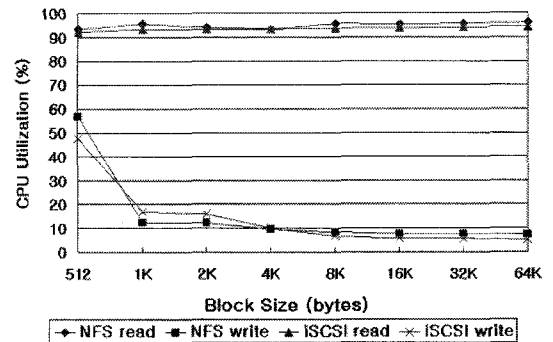


그림 9. CPU 점유율 측정 결과 (NFS vs iSCSI)

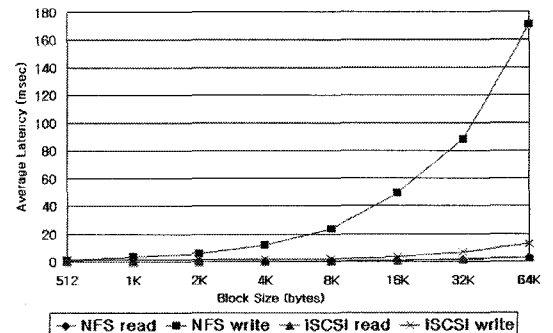


그림 10. 입출력 지연율 측정 결과 (NFS vs iSCSI)

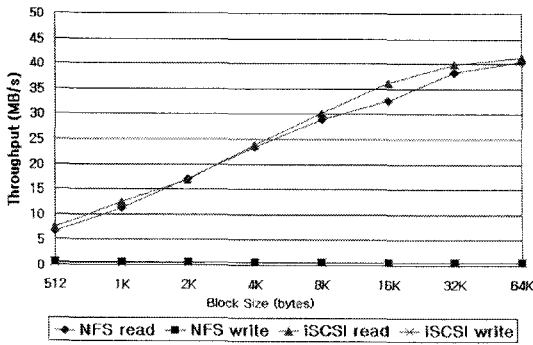


그림 11. 입출력 처리량 측정 결과 (NFS vs iSCSI)

보다 더 높게 측정되는 것은 모바일 장치상의 성능문제에서 기인한다고 볼 수 있다.

[그림 11]에서는 입출력에 대한 데이터 처리량을 나타낸다. 두 방법 모두 그렇게 큰 차이는 나지 않지만 전반적으로 NFS보다 iSCSI가 좀더 나은 결과를 보여주었다. 그리고 두 방법 모두 데이터의 쓰기 보다는 읽기에 대한 데이터 처리량이 높았다.

이런 성능분석의 결과들이 나타나게 되는 것은 iSCSI는 비록 네트워크상으로의 데이터 전송을 위해 패킷으로 변환하는 과정을 거치게 되지만 원격지에 도착한 후에는 저수준의 블록 입출력만을 통해 Storage에 접근하게 된다. 즉, 해당 목적지의 파일 시스템을 거칠 필요 없음을 의미 한다. 이에 반해 NFS는 입출력데이터를 RPC(Remote Procedure Call)를 통해 목적지 NFS 데몬에게 전달하고 데이터를 전달받은 데몬은 자신이 받은 데이터를 목적지의 파일시스템으로 다시 전달하게 되고 해당 파일시스템은 내부의 블록장치로 데이터를 전달하는 과정을 거치게 되는 점 때문에 이러한 결과가 나타난 것으로 분석이 된다. 이와 같은 점은 차세대 인터넷 환경에서 iSCSI가 모바일 장치가 가지는 제약사항 극복의 대안이 될 수 있다는 가능성이 있음을 보여주는 것을 나타낸다.

5. 결 론

본 논문에서는 저장 공간의 확장과 이동성이 필요한 모바일 시장에서 점차적으로 시장영역을 형성하고 있는 임베디드 리눅스를 이용하여 IPv6 무선 네트워크 환경에 맞는 iSCSI Initiator Driver를 구현하였고, 이를 동영상 학습 시스템에 적용하여 학생들이

동영상 강의의 시청할 수 있도록 하였다. 제한된 무선 네트워크 환경에서 IPv4 환경에서의 결과와 마찬가지로 NFS와 같은 원격 스토리지 접근 방법보다는 나은 성능을 가지는 것을 알 수 있었다.

결과적으로 우리는 iSCSI와 Mobile IPv6무선 네트워크를 이용하여 모바일 장치가 가지는 여러 제약 사항 극복방안 중 전원 절감, 저장 공간의 확장과 이동성 측면에서 보다 근본적인 해결 방안을 찾을 수 있었다. 특히, 아주 큰 내용량의 저장장치를 필요로 하는 모바일 환경에서는 기존의 방법들에 비해 더욱 효과적인 해결 방안이 될 수 있을 것이며, 다른 분야에서도 응용이 가능할 것이라 생각한다.

감사의 글

본 연구는 산업자원부의 지역혁신 인력양성사업의 연구결과로 수행되었음.

참 고 문 헌

- [1] 최세봉, 임효택 “iSCSI 기반의 무선 네트워크 연결형 가상 저장장치 드라이버 구현 및 성능 분석,” 한국통신학회논문지, 제30권, 제6A호, pp. 517- 525, 2005. 6.
- [2] 권혁찬, 나재훈, 정교일, “Mobile IPv6 표준화 및 기술동향,” IITA 주간기술동향, 2004. 5.
- [3] 유준석, 나재훈, 손승원, “Mobile IPv6 표준화 및 기술동향,” IITA 주간기술동향, 2003.
- [4] R. Telikepalli, T. Drwiega, and J. Yan, “Storage area network extension solutions and their performance assessment,” *IEEE Communications Magazine*, Vol. 42, No. 4, pp. 56-63, 2004.
- [5] K. Meth, and J. Satran, “Features of the iSCSI protocol,” *IEEE Communications Magazine*, Vol. 41, No. 8, pp. 72-75, 2003.
- [6] Y. Zhou, A. Bilas, S. Jagannathan, D. Xinidis, C. Dubnicki, and K. Li, “VI-attached database storage,” *IEEE Transactions on Parallel and Distributed Systems*, Vol. 16, No. 1, pp. 35-50, 2005.
- [7] J. Lee, J. Hwang, S. Lim, and K. Park,

"i2SCSI: Intelligent iSCSI disk providing cache consistency in TCP/IP-based storage area network," *16th International Conference on Parallel and Distributed Computing Systems(PDCS)*, Aug. 2003.

- [8] S. Narasimhamurthy, P. Gurumohan, S. Sreenivasamurthy, and J. Hui., "Quanta data storage: an information processing and transportation architecture for storage area networks," *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 10, pp. 2032-2040, 2005.
- [9] Y. Lu and D. Du, "Performance study of iSCSI-based storage subsystems," *IEEE Communications Magazine*, Vol. 41, No. 8, pp. 76-82, 2003.
- [10] UNH iSCSI Consortium, <http://iol.unh.edu/consortiums/iscsi/>
- [11] M. LoBue, "Surveying today's most popular storage interfaces," *IEEE Computer*, Vol. 35, No. 12, pp. 48-55, 2002.
- [12] 박수라, 문보석, 김대근, 김주호, 김경훈, 박성순, 이진구, "CDMA-2000 1x 방식을 이용한 무선 네트워크에서의 CIFS, NBD, iSCSI 프로토콜 성능분석," 스토리지 시스템 학술대회 논문집, pp. 241-245, 2003.



임 효 택

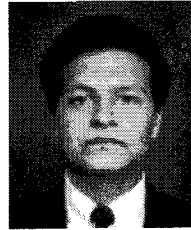
1988년 홍익대학교 전자계산학과 졸업(이학사)
 1992년 포항공과 대학원 전자계산학과 졸업(공학석사)
 1997년 연세 대학교 컴퓨터과학과 졸업(공학박사)
 1988년~1994년 한국전자통신연

구소 연구원

2000년~2002년 Univ. of Minnesota(미) 컴퓨터공학과 연구교수

1994년~현재 동서대학교 컴퓨터공학과 부교수

관심 분야: Computer Network, Protocol Engineering, Storage Networking, IPv6, Mobile Application



비나이 쿠마르 싱

2004년 인도 UP 기술대학교, Lucknow 전자통신공학과(공학사)

2005년~현재 동서대학교 컴퓨터 네트워크공학과 석사 과정

관심 분야: u-헬스케어, USN, 디지털이미지프로세싱



차 경 환

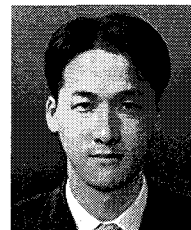
1985년 부경대학교 전자통신공학과(공학사)

1990년 부경대학교 전자통신공학과(공학석사)

2006년 부경대학교 전자공학과(공학박사)

1990년~1995년 LG전자 LS연구소 주임연구원

1995년~현재 동서대학교 컴퓨터정보공학부 부교수
 유비쿼터스 어플라이언스 지역혁신센터 소장
 관심분야: 임베디드 시스템, 스마트 홈 시스템



최 새 bom

2005년 2월 동서대학교 컴퓨터공학과(공학사)

2007년~현재 연세대학교 컴퓨터공학과 재학

2005년~현재 산업기술평가원 정보화기획단 근무

관심분야: Embedded Systems, IPv6, Mobile IPv6, SDR