

# USB/IP를 이용한 원격장치공유에 대한 연구

유진호<sup>1\*</sup>

<sup>1</sup>백석대학교 정보통신학부

## A Study on Sharing the Remote Devices through USB over IP

Jinho Yoo<sup>1\*</sup>

<sup>1</sup>Division of Information Communication, Baekseok University

**요약** 본 논문은 원격장치를 공유하는 방법에 의해 자료를 공유하는 방법을 제시하는 연구에 대한 것이다. 원격장치 공유는 실제로 서버 가상화 환경에서 개인화된 하나의 접근방법이 되는 USB장치의 연결기술을 그 기반으로 한다. 가상화된 서버 환경에서 서버를 자유롭게 할당받아 사용하는 사용자는 원격컴퓨팅 환경을 로컬환경의 저장장치로의 이동을 원할 수 있다. 또한 USB장치 같은 경우는 파일시스템 수준이 아니라 장치 드라이버 수준에서 공유를 구현하여 장치접근에 관한 문제를 해결할 수 있다. 이에 본 논문에서는 서버가상화 환경에서 개인화된 서비스의 기반이 되는 USB장치의 가상화 기술에 대한 구현연구이다. 이와 같이 장치 드라이버 수준에서 장치를 가상화하여 네트워크에서 공유하게 된다. USB가 장치수준에서 가상화되기 때문에 USB의 파티션 위의 파일시스템 수준에서 데이터를 저장하는 것에 대한 처리 뿐만아니라 장치수준에서 기록한 데이터에도 접근할 수 있게 된다.

**Abstract** This paper is related to the method for sharing some remote devices through USB/IP on the connected system environment. Sharing the remote devices is actually based on the connection technology using USB which is one way of access methods on virtualized server environment. The users who receive server computing unit might want to connect to remote allocated server with the local devices. We can solve the problem of the access methods related to the sharing devices through USB device emulation. This paper discusses the implementation of USB emulation which is based on personalized services on virtualized server environment. Like this, this paper will share devices on the level of USB device. This research can write to the device directly due to virtualizing the device level.

**Key Words** : USB, USB/IP, virtualization

### 1. 서론

하드웨어 환경의 급속한 발전에 따라 컴퓨터 시스템의 하드웨어 구성은 고성능화 되었고 범용 소프트웨어 수행을 위해 요구되는 컴퓨팅 능력을 충분히 처리할 수 있다. 이에 컴퓨터 시스템 상에 하드웨어를 소프트웨어로 가상화하여 구현하게 되었다. 한 컴퓨터 하드웨어 상에 가상화된 여러 소프트웨어 머신을 탑재할 수 있게 되었다. 컴퓨터 주요기능수행에 있어서는 전혀 지장없이 하나의 컴퓨터 상에 소프트웨어 머신들을 탑재할 수가 있었다. 하

드웨어가 분리된 것처럼 컴퓨팅요소로 채워지게 되었다. 이에 따라 서버 가상화 환경에서 개인의 컴퓨팅을 위해 컴퓨팅을 인가하는 환경에서 새로운 개인화에 관련한 서비스가 필요하게 되었다. 하드웨어 환경이 소프트웨어의 발전보다 훨씬 앞서는 형태는 결국 다시금 고성능서버로부터 개인에게 필요에 따라 컴퓨팅을 빌려주는 형태로 발전하게 되었다. 즉 개인이 필요에 의해 컴퓨팅을 요구하면 그 요구 스펙에 맞는 소프트웨어 컴퓨터인 가상화된 컴퓨터를 구성하여 빌려주게 된다. 이와 같은 컴퓨팅 환경의 변화와 유비쿼터스 환경의 결합에 의해 어느 곳

\*교신저자 : 유진호(yoojh@bu.ac.kr)

접수일 10년 10월 28일

수정일 10년 11월 16일

게재확정일 10년 11월 19일

에서나 컴퓨팅 서버로부터 자원을 할당받는 형태의 컴퓨팅 모델로 발전하게 되었다. 필요로 되는 시스템의 구성요구에 의하여 주변 컴퓨팅 환경을 조립하여 요구에 의한 시스템을 구성하게 되는 것이고, 시스템의 구성요소가 가상화 되어 있기 때문에 가상환경으로 구성된 컴퓨팅요소를 필요에 의해 공유하거나 빌려줄 수 있다. 이러한 시스템 구현 기술에 따라 본 논문에서는 가상 시스템의 주된 컴퓨팅 구성요소 뿐만아니라 장치를 빌려주는 방법에 대해서 논한다. 본 논문에서는 요구에 의해 구성되는 시스템 상에서 원격장치를 사용하는 원격장치제어의 기술적인 구현에 대해 논할 것이다.

## 2. 연구배경

### 2.1 배경과 대상장치

본 절에서는 원격장치를 공유하는 기술에 대한 기술적인 배경을 설명하도록 한다. 본 시스템의 배경은 기존의 USB장치를 원격에서 접근하고자 하는 요구에서 비롯되었다. USB의 경우 특성별로 접근하는 것보다 장치자체로 접근하는 것이 유용한 사용을 제공하므로 유익하다. 즉 USB특성별로 그 기능에 따라 공유하는 것이 아니라 USB 장치의 모든 하드웨어에 대한 접근방법이 가능하게 구성하는 것이다. 접근에 대한 구현이 하위수준에서 이루어지기 때문에 상위에서 일어나는 것보다 모든 접근에 대해 가능하다. 즉 USB장치의 하부에서 수정이 일어나므로 수정되는 하부 이상의 부분에 대한 접근이 정상적으로 인가되는 것이다.

### 2.2 가상화 시스템

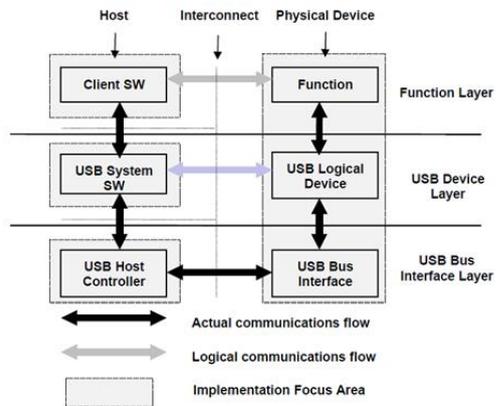
본 논문에서는 논문의 기본적인 수행환경이 가상화 시스템인 것을 가정한다. 이 시스템 구성 상에는 장치들이 혼재해 있다. 네트워크 상에 장치들이 흩어져 있고 로컬 시스템에 로컬장치로 설치가 되어 있다. 가상화 시스템은 가상 컴퓨터와 또 다른 컴퓨터에 연결된 다양한 주변장치로 구성된다. 가상화 컴퓨터는 자신의 Guest 영역을 관리하며 다음과 같은 기능을 제공한다.

- 컴퓨팅 자원의 관리
- 사용자와 주변장치들의 제구성
- 자원의 동적인 할당
- 협업화된 가상머신 환경 수행

가상화 영역 내의 모든 컴퓨팅 자원과 주변장치들은 가상화 컴퓨터의 자원관리자에 등록된다. 주변장치를 제외한 모든 컴퓨팅환경은 Xen에 연관하여 구성된다.

### 2.3 USB와 USB/IP

USB는 가장 널리 사용되는 컴퓨터 주변장치 중에 하나이다. 거의 모든 주변장치가 USB 인터페이스를 지원한다[3]. 즉 USB 장치를 통해 입출력장치를 정의할 때 가장 일반적으로 사용되는 것이 될 수 있다는 뜻이다. USB는 물리적 장치 뿐만아니라 소프트웨어 장치스택을 가지고 있으며 주종관계를 통해 상당히 통제되고 예리없는 사용방법을 제공한다. 아래 그림은 USB 전체의 스택에 관한 것이다.



[그림 1] USB stack

그림에서 보는 것과 같이 가장 하부에는 호스트제어기가 USB 버스 인터페이스를 통해서 주종관계로 연결되어 있다. 이를 위해 호스트는 USB 장치를 구동시키는 시스템 소프트웨어와 그 위의 각 장치별 장치구동 소프트웨어로 스택이 구성된다. 이러한 호스트 스택은 여러 개의 USB 장치를 제어할 수 있는 형태로 제공된다. 종속적인 역할을 하는 USB장치는 소수의 기능단위 구현이 된다. 위의 스택과 같이 구성되는 USB의 관계 중 가장 하위단인 USB 버스 인터페이스 수준을 소프트웨어 버스로 연결한다. 이를 통해 원격지에 있는 USB 장치를 마치 자신의 로컬장치에 부착된 장치인 것처럼 사용할 수 있다. 실제적인 연결을 IP네트워크를 통해 USB/IP를 구현한다 [1,2].

## 3. 원격장치서비스

### 3.1 개관과 시나리오

본 서비스 시나리오는 원격지의 저장장치를 이용하는 것이다. 예를 들어 스마트폰 사용자가 늘어나는 상황을 생각해 보자. 개인용 컴퓨팅 장치인 스마트폰을 가지고

이동하는 사용자가 방문한 위치 내의 가까운 컴퓨터에 연결되는 방법을 제공하는 것이다. 스마트 폰에서 서버 역할의 다른 컴퓨터에 있는 기억장치에 접근할 수 있다. 반대로 서버에서 계산을 하고 스마트폰에 있는 기억장치에 접근할 수도 있다. 이는 원격지 접근이 어떤 시나리오를 가지는가를 보면 알 수 있다. 본 연구에서 구현하는 원격 저장장치 접근은 USB 장치를 빌려주는 것이다. 단순히 파일시스템 개념의 디렉토리를 보여주는 것이 아니라 USB 장치 자체를 빌려주는 것이다. 본 연구는 원격지의 파일시스템을 접근하는 것이 아니라 원격지의 USB 장치를 접근하는 것이다. 원격지의 파일시스템에 접근하는 것은 하부가 파일시스템으로 이루어져 있어야 하며 하부 파일시스템 아래로 접근은 불가능해진다. 장치를 빌려준다는 것은 USB 장치 수준으로 원격지 접근을 가능하게 하는 것이다. USB 장치의 특정한 위치에 임의로 접근하는 것은 파일시스템으로 접근하는 것으로는 한계가 있다. USB를 장치수준으로 접근해야 해결할 수 있다. 파일시스템은 파티션 위에 올라가는 소프트웨어 구조이다. 디스크 장치는 장치 위에 512바이트의 MBR(Master Boot Record)와 최대 4개의 파티션으로 구성된다. 파티션은 다시 부트레코드블럭, 슈퍼블럭, 아이노드리스트 그리고 데이터블럭으로 이루어진다. 파일시스템이 파티션 위에 올라간다 해서 파일시스템을 통해 파티션의 모든정보에 접근가능한 것은 아니다. 파일시스템에서 인가한 함수에 의해서만 파티션에 접근할 수 있다. 즉 파일시스템의 파일 접근 함수만을 이용할 수 있다.

### 3.2 원격장치파일 생성방법

USB 저장 장치의 파일시스템을 파일로 변화시키기 위해 기존의 기법인 file-backed storage gadget(FSG)를 사용한다. 우리는 스토리지 장치를 대신하는 파일을 만들 수 있다. 우리는 파일시스템에 대한 접근을 하나의 파일로 구성할 수 있다. 그 명령은 다음과 같다.

```
# dd if=/dev/zero of=USB_storage bs=1024K count=64
```

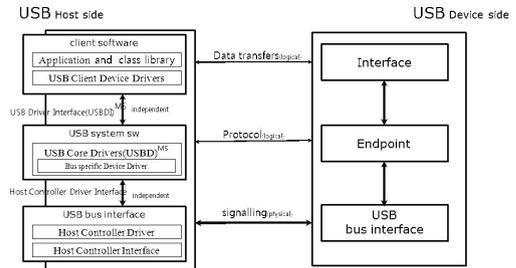
위의 명령은 64M의 파일을 만들게 되며 64M USB 장치를 구성하는 것이다. 우리는 위 파일을 원격접근을 위한 파일시스템으로 사용하기 위한 것이다. 파일을 파일시스템으로 다루기 위해 운영체제에서 몇가지 작업이 필요하다. file-backed storage를 사용하기 위해서는 그 기능을 지원하는 USB 호스트 제어기가 전제되어야 한다. 그 USB 호스트 제어기에 file-backed storage를 USB장치로서 연결하게 된다. 시스템에서 부착된 file-backed storage

는 USB디스크로 보인다. file-backed storage인 파일은 이제부터 USB디스크로 인식된다. fdisk를 수행하여 파티션을 설정할 수 있으며 파일시스템처럼 접근할 수 있다[4].

### 3.3 시스템 명령 절차와 USB/IP구성

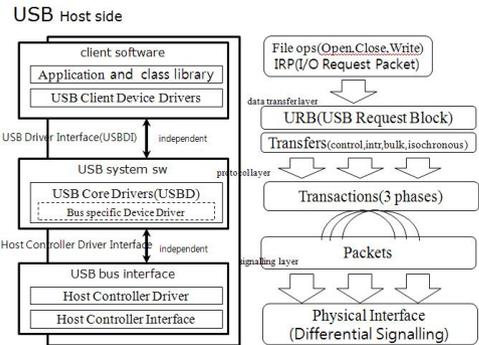
#### 3.3.1 USB/IP구성

시스템은 사용을 위해 기본적으로 USB/IP설정을 해야 한다. USB/IP는 커널영역에서 IP연결을 사용한다[1,2]. USB구성은 아래그림에서 보는 바와 같다.



[그림 2] USB 호스트와 장치 프로토콜 스택

USB는 원래 그림2와 같은 구성을 가지고 있다. USB/IP는 위 부분에서 호스트 스택의 가장 하위단을 하위단 바로 위에 위치한 소프트웨어 수준에서 아랫단 하드웨어를 대신 처리하게 된다. 하드웨어는 전송단에 해당되며 상위스택에서 전달된 데이터를 USB장치로 전달하게 된다. USB호스트의 하위 하드웨어를 소프트웨어로 처리하여 USB장치의 연결을 소프트웨어적으로 임의로 변경할 수가 있게 된다. 그림 2에서 보는 하위단을 물리적인 연결이 아닌 IP연결을 사용하여 구현한다. 또한 이러한 구현을 위해서는 USB호스트단의 하위 소프트웨어 부분인 호스트제어기 드라이버를 새로 작성해주어야 한다. 호스트 제어기단의 상세내역은 그림3과 같다.



[그림 3] 호스트 제어기 상세

그림 3은 USB 호스트단의 상세한 내용을 보여준다. 가장 상위단에서는 파일을 접근하는 것처럼 접근을 하게 된다. 이는 아래로 내려오면서 트랜잭션들로 구성이 되고 다시 패킷으로 구성된다. 이러한 자료구조 지원을 위한 소프트웨어구조는 클라이언트 소프트웨어는 USB시스템 소프트웨어에 연결이 되고 그 아래는 호스트 제어기로 연결되기 위한 호스트 제어기 드라이버로 연결된다. USB/IP는 호스트 제어기 드라이버를 소프트웨어적으로 재구성한다. 그래서 USB 호스트 제어기 드라이버를 커널모듈을 새롭게 구성하여야 한다. 새롭게 구성된 USB 호스트 제어기는 커널에 독립된 USB 호스트 제어기로서 등록된다. 독립적으로 구성된 USB 호스트 제어기에 file-backed storage를 가상의 USB 디스크로 인식되는 파일을 설정할 수 있다. 본 연구는 USB 호스트 제어기로 연결되는 파일을 USB 디스크로 보이게 하여 원격에 위치한 파일을 USB디스크로 부착시킬 수 있다. USB/IP는 커널 모듈형태로 만들어져 커널에 로딩되어 수행된다[1].

### 3.3.2 커널 설정

USB 호스트제어기의 사용과 그에 맞는 소프트웨어 구성을 위한 커널설정을 해야한다.

```
Configure kernel compile option in 'Make menuconfig'
Device Driver -> USB Support -> USB Gadget ->
<+> Support for USB Gadget
Device Driver -> USB Support -> Driver Mode ->
<+> Gadget Stack
Device Driver -> USB Support -> USB Gadget ->
USB peripheral controller
<+> Invertra (M) USB peripheral
<M> USB Gadget driver
<M> File-backed Storage gadget
```

위와 같은 설정을 통해 커널이 가상의 호스트 제어기와 file-backed storage를 구성할 수 있는 환경을 만드는 것이다[4].

```
# modprobe g_file_storage file=USB_storage
```

위 명령에서 USB\_storage는 만들어둔 64M 파일이며, 위와 같은 명령을 사용하여 커널 내의 모듈로 로딩한 후 USB\_storage라는 파일로 USB 디스크를 구성하게 된다.

### 3.3.3 시스템 연결

USB/IP연결이 된 후 USB호스트 제어기가 있는 시스

템에서는 원격의 USB장치가 USB호스트에 부착된 것을 아래와 같이 확인할 수 있다.

```
# lsusb
Bus 001 Device 006: ID 03f0:a004 Hewlett-Packard DeskJet 5850c
Bus 001 Device 004: ID 413c:2003 Dell Computer Corp.
Bus 002 Device 001: ID 0000:0000
Bus 002 Device 002: ID 047d:102e Kensington
```

호스트 제어기단에서 위와같은 확인이 이루어지면 성공적으로 USB enumeration에 성공한 것이다. 위의 내용은 USB버스는 2개가 있고 거기에 장치가 몇 개씩 부착된 형태이다.

## 4. 결론

본 논문은 원격 저장장치 접근에 대한 기술적인 접근에 관련한 것이다. 본 논문은 원격지 자원의 상호공유에 관한 것이며, 그 방법의 일환으로 USB장치를 소프트웨어로 이물레이션한 후 장치 간 통신을 위해 IP를 사용한다. 이러한 일련의 과정은 커널 상에서 구현이 되고 이를 USB/IP라 한다. USB/IP를 기반으로 USB장치를 공유하게 된다. 자원공유를 위해서 USB 장치 중에 USB디스크를 사용하여 디스크장치를 공유한다. 디스크장치를 유연하게 사용하기 위해 파일로 연결시켜 접근할 수 있는 방법을 모색한다. 디스크장치를 파일형태로 조작할 수 있게 함으로써 소프트웨어 수준에서 하드웨어 조작을 처리할 수 있게 되어 훨씬 다루기가 용이해지는 장점이 있다. 하드웨어 부착장치를 소프트웨어로 구성하여 조작이 쉽고 원격지에서 접근할 수 있게 하여 자원공유를 편리하게 하였다.

이러한 구성은 다음과 같은 시나리오로 응용이 될 수 있다. 스마트폰을 가지고 있는 사용자가 어떤지역에 접근했을 때 스마트폰에 있는 일정화일을 USB장치로 인식시켜 암호화된 자신의 ID로 인식시켜 호스트 컴퓨터에서 사용할 수 있게 할 수 있다. 또한 휴대용 저장장치의 무선접근이 가능하게 하는 역할을 동시에 하여 그 편리성과 휴대성을 가질 수 있다. 네트워크 연결만 되면 어디서든 USB장치를 로컬이 아닌 곳에서 접근할 수가 있고 무선환경이 제공되면 그 편리함을 그대로 이용할 수가 있다. 무선네트워크를 이용할 경우에는 네트워크의 속도가 USB 장치로서의 시간 요구사항을 만족시킬 수 있어야

한다는 조건이 따른다. 일반 안드로이드 단말에서 성공적으로 접근이 되었다. 이러한 기술적 접근을 통해 맥내의 원격지 USB 장치 접근이 용이하다는 것을 알았다. 이를 응용하는 시나리오들이 편의를 제공할 것이다.

## 참고문헌

- [1] Takahiro Hirofuchi, Eiji Kawai, kazutoshi Fujikawa, and Hideki Sunahara, "USB/IP: A Transparent Device Sharing Technology over IP Network," IPSJ Transactions on Advanced Computing Systems, Vol. 46 No. SIG12 (ACS11), pp.349-361
- [2] Source Forge, "The USB/IP Project," <http://sourceforge.net/projects/usbip/>
- [3] USB Implementers Forum, Inc., USB 2.0 specification, <http://www.usb.org/developers/docs/>
- [4] Backing Storage for the File-backed Storage Gadget, [http://www.linux-usb.org/gadget/file\\_storage.html](http://www.linux-usb.org/gadget/file_storage.html)
- [5] Sun Microsystems, Sun xVM, <http://www.sun.com/software/-products/xvm/index.jsp>

---

유진호(Jinho Yoo)

[정회원]



- 1996년 2월 : 서강대학교 컴퓨터 공학과 (전산학석사)
- 2006년 8월 : 충북대학교 컴퓨터학과 (전산학박사)
- 1996년 1월 ~ 1998년 12월 : 엘지정보통신연구소 전임연구원
- 1999년 1월 ~ 2008년 2월 : 한국전자통신연구원 선임연구원
- 2008년 3월 ~ 현재 : 백석대학교 정보통신학과 교수

<관심분야>

임베디드시스템, HCI, 가상화 시스템