

# 가상 네트워크를 위한 Xen 기반 가상 라우터 설계

이신형\*, 황재현\*, 유혁\*

\*고려대학교 컴퓨터학과

e-mail : shlee@os.korea.ac.kr

## Design Xen Based Virtual Router for Virtual Network

ShinHyoung Lee\*, Jae-Hyun Hwang\*, Chuck Yoo \*

\*Dept. of Computer Science and Engineering, Korea University

### 요 약

미래 인터넷은 다양한 형태의 네트워킹 기술이 공존할 수 있고 이를 위해 네트워크 가상화 기술이 각광받고 있다. 네트워크 가상화의 핵심 기술은 라우터 가상화 기술이며, 이를 위해 소프트웨어 기반으로 많은 연구들이 진행되어 오고 있다. 본 논문에서는 가상 네트워크를 위한 가상 라우터가 갖추어야 할 요구사항을 정리하고 이를 위해 오픈 소스 가상화 솔루션인 Xen 기반으로 독립성, 성능, 그리고 유연성을 지원할 수 있는 가상 라우터를 설계한다.

### 1. 서론

최근 전 세계적으로 앞으로 다가올 미래의 인터넷 환경에 대해 논의가 뜨겁게 이루어지고 있다. 이러한 미래 인터넷 환경은 Clean Slate[1]라는 사명 아래 어떠한 고정된 정의도 거부하고 있다. 즉, 미래에는 현재의 인터넷 환경과는 달리, 기존 기술들의 한계점을 뛰어 넘는 다양한 형태의 네트워킹 기술이 공존할 수 있으며, 새로운 기술 역시 쉽게 수용하고 실용화할 수 있는 구조가 제공되어야 한다. 현재 인터넷이 갖는 근본적이고 구조적인 문제는 지금까지 구축된 인터넷 환경을 바닥부터 새롭게 재설계 및 재구축하는 것이 시간적, 비용적 등의 측면에서 볼 때 현실적으로 불가능하다는 점이다. 따라서 미래 인터넷은 기존의 인터넷 환경을 수용하면서 새로운 네트워킹 패러다임으로 확장 가능하게 해야 하며, 이러한 유연성을 제공하는 기술을 연구 개발하는 것이 절실하다.

네트워크 가상화 기술은 미래 인터넷에 유연성을 제공하는 가장 적합한 해결책으로 고려되고 있다. 또한, 미래 인터넷뿐 아니라 클라우드 컴퓨팅, CCN(Content-Centric Network) 등 새로운 컴퓨팅 패러다임 측면에서도 네트워크 가상화는 핵심 기술로 자리 잡고 있다. 네트워크 가상화란 하나의 물리 네트워크 상에 독립적인 서로 다른 가상 네트워크를 운용할 수 있는 기술을 의미한다. 원래 가상화 개념은 1970년대 IBM에서 여러 개의 다른 명령어들을 하나의 기계 상에서 돌리기 위하여, 가상머신을 설계했던 것에서 출발하였고, 요즘에는 여러 영역 -운영체제 가상화, 네트워크 가상화-에 적용되고 있다.

최근 네트워크 가상화 기술에 대해 국내 . 외적으로 많은 관심이 쏟아지고 있으나, 현재까지는 대부분 테스트베드 구축을 위한 용도로 활용되어 왔으며, 특정 하드웨어에 의존하는 경향을 보여 왔다. 그러나 하드웨어의 교체에 따른 비용을 감안할 때 궁극적으

로 네트워크 가상화 기술은 소프트웨어적으로 구현되어야 하며, 더불어 라우터 등 네트워크 구성요소의 특성을 고려한 가상화 구조의 설계가 요구된다. 또한 네트워크 가상화 기술은 현재 네트워크의 근간을 이루는 IPv4 와 미래 인터넷 프로토콜을 동시에 적절한 성능으로 지원함으로써 실제 환경에서 사용할 수 있도록 한다.

### 2. 관련 연구

가상 네트워크를 위해 많은 연구가 진행되고 있다. 가상 네트워크는 유연성 지원 때문에 주로 소프트웨어 기반으로 이루어져야 한다.

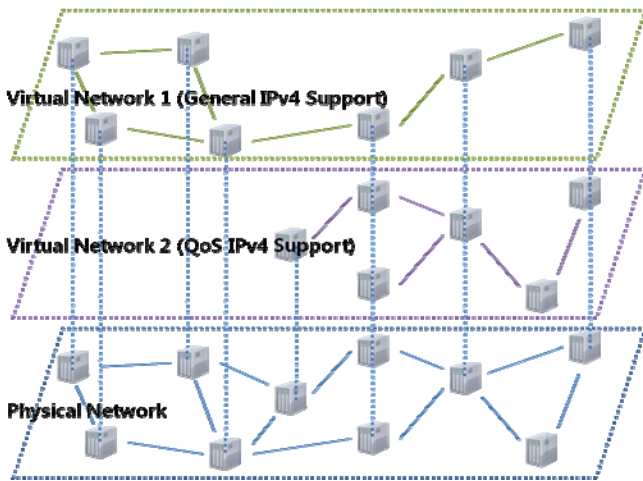
클릭 라우터는 2000년 MIT 와 Mazu network, ICSI 센터, UCLA 가 공동으로 개발한 모듈화된 소프트웨어 라우터로써, 유연성과 확장성이 뛰어나다. 클릭은 엘리먼트 라고 하는 패킷 처리 모듈로 구성되는데, 엘리먼트는 패킷분류, 큐잉, 스케줄링, 네트워크 장비들의 연결과 같은 기능을 하며, C++를 이용하여 프로그래밍되어 있다. 표준 클릭 IP 라우터는 최대 16 개의 엘리먼트를 가질 수 있다. “pull connection” 과 “push connection” 을 통하여 엘리먼트 사이를 연결 하며, 라우터의 목적에 따라 사용자는 엘리먼트들을 재조합하고, 재설정하며, 새로운 엘리먼트도 추가 할 수 있다. 이러한 확장성과 유연성 때문에 가상화 연구의 소프트웨어 구조로써 사용되고 있다[2,3].

Linux-VServer 는 2003년 리눅스 서버를 모니터링, 백업, UPS 등의 공통 태스크를 공유하면서 가상 서버로 분리하기 위한 프로젝트가 Linux 커뮤니티 프로젝트로 발전한 것으로 OS 단계의 가상화를 제공한다 [4,5]. Linux-VServer 의 구조는 하드웨어, 커널, 가상머신의 3 개의 컴포넌트로 구성되어 있으며, 물리적 서버위에 생성 가능한 가상머신의 수에 제한이 거의 없다. 더욱이 리눅스 커널을 공유하는 분리된 사용자

공간(VServer)을 이용하여 서로 다른 정책 적용이 가능하다. VServer(VM)은 커널을 제외한 작은 공간만을 사용하여 다수의 VM 을 탑재할 수 있다는 장점이 있다[9]. 하지만 커널을 공유하기 때문에 VM 들간의 완벽한 격리가 이뤄지지 않기 때문에 보안 문제와 다른 VM 들로부터 간섭을 받을 수 있으며, VM 들은 반드시 리눅스를 사용하여야 한다는 단점이 있다[5,6].

XORP 는 안정적인 라우터용 플랫폼을 위한 오픈 소스 소프트웨어 개발을 목적으로 한다[7]. XORP 는 하이레벨(유저레벨)과 로우레벨의 서브시스템으로 구성되어 있다. 하이레벨의 시스템은 라우팅 프로토콜을 위한 라우팅 정보를 기반으로 라우팅 프로세스를 지원한다. 로우레벨의 서브시스템은 OS 커널 안에서 동작한다. 로우레벨에서는 포워딩 경로를 관리하고 하이레벨과의 연결을 위한 API 를 제공한다. 이러한 구조로써 XORP 는 안정적이고 유연한 라우터를 위한 오픈 플랫폼을 목적으로 개발 되었지만, 기존 소프트웨어 라우터와 마찬가지로 하드웨어 기반라우터에 비해 성능이 떨어진다는 단점과 배포가 쉽다는 장점을 동시에 갖고 있다[8].

UCL 에서는 젠 기반 라우터를 위해 VROUTER 연구를 진행하고 있다. VROUTER 연구는 젠 기반 라우터의 성능을 향상시키는 것을 목적으로 삼고 있다. 그러나 너무 성능에 초점을 맞추다 보니 가상화의 중요한 기능인 독립성과 유연성을 제대로 지원하지 못하는 치명적인 문제가 있다.



(그림 1) 하나의 물리 네트워크를 공유하는 두개의 가상 네트워크

### 3. 라우터 가상화의 요구사항

라우터 가상화 기술이 완전히 구현되기 위해서는 1) 독립성(isolation), 2) 성능(performance), 3) 유연성(flexibility)을 만족시켜야 한다.

#### 3.1 독립성

물리적으로 한 개의 라우터 위에 다수의 가상 머신을 올려서 다수의 가상 네트워크를 지원하는 라우터 가상화 기법을 사용할 때 각각의 가상 네트워크는 서로

영향을 주지 않아야 한다. 예를 들어 그림 1 에서 일반적인 IPv4(Virtual Network 1)와 QoS 를 보장하는 IPv4(Virtual Network 2)가 각각 가상 네트워크를 구축하고 있을 때 Virtual Network 1 에 문제가 생기더라도 Virtual Network 2 의 동작 및 성능에 영향을 미쳐서는 안되고 마찬가지로 그 역으로도 영향을 미쳐서는 안된다. 이것이 가상 네트워크간의 독립성이다.

#### 3.2 성능

현재 소프트웨어 기반의 라우터는 7Gbps 정도의 성능을 제공한다. 하지만 하드웨어 기반의 라우터는 수십 Gbps 에서 수십 Tbps 의 성능을 제공한다. 이는 소프트웨어 기반 라우터가 갖는 가장 큰 문제이기도 하다. 라우터 가상화 기법이 실용성을 갖기 위해서는 성능 향상을 통해 기존의 하드웨어 라우터에 준하는 성능을 제공해야 한다.

#### 3.3 유연성

기존의 하드웨어 라우터에 비해 소프트웨어 라우터가 갖는 가장 큰 장점은 다양한 프로토콜을 간단한 소프트웨어 패치를 통해서 적용 가능하다는 점이다. 라우터 가상화 기법을 사용할 경우 가상 머신 위에서 동작하는 각각의 도메인마다 모두 다른 프로토콜을 사용할 수 있고, 또한 다른 정책을 사용할 수도 있어야 한다.

## 4. 젠[11]을 이용한 가상 라우터의 설계

앞서 이야기한 가상 라우터의 요구사항을 만족시키기 위해서 다음과 같은 젠 기반 가상 라우터 구조를 제안한다.

#### 4.1 링크 가상화

하나의 물리적 링크를 다수의 가상 네트워크가 공유하여 사용하기 위해서는 물리적 링크를 가상화 시켜야 한다. 이를 링크 가상화 기법이라 부른다. 링크 가상화는 네트워크 인터페이스 카드를 가상화시킴으로써 가능하다.

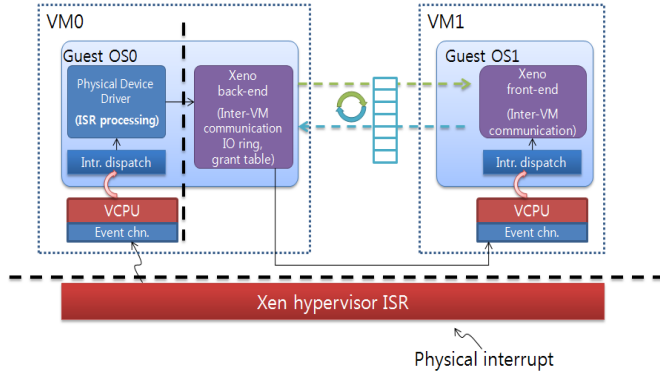
링크 가상화의 중요한 점은 각 가상 네트워크간의 간섭을 최소화 시키는 것이다. VServer 와 같은 기술은 커널을 공유함으로써 인해 각 가상 네트워크간의 독립성이 완전히 제공되지 않는다. 이는 곧 하나의 가상 네트워크에 문제가 생길 때 그 문제가 다른 가상 네트워크로 전이될 수 있음을 뜻한다.

본 논문에서는 이를 위해 젠 기반의 라우터 가상화 기법을 제안한다. 젠은 하이퍼바이저 기반의 가상화 기술로써 각 가상머신간의 독립성이 보장된다는 장점이 있다.

#### 4.2 효율적인 I/O

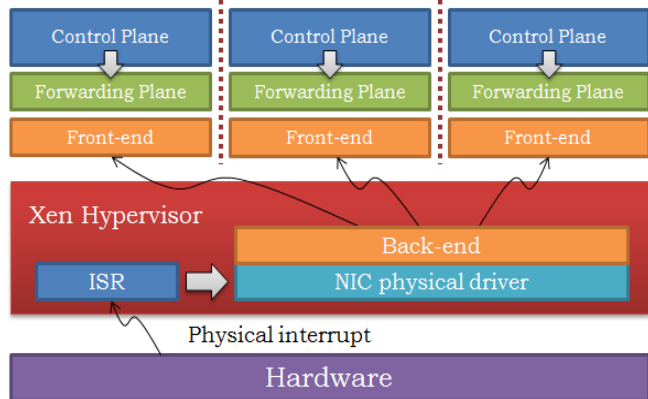
기존의 젠 I/O 구조는 드라이버 도메인을 분리하여 제공함으로써 장치 드라이버를 하이퍼바이저에서 분리하였다. 이를 통해 젠은 드라이버 도메인 가상 장치에 대한 추상화 인터페이스를 제공하고, 하이퍼바이저는 두 도메인 간의 통신을 제공한다. 또한 가상

장치의 동기적 처리로 인한 처리 시간 지연을 줄이기 위해 도메인 간 비동기 통신 기법을 사용한다. 데이터 전달을 위한 비동기 통신은 그림 2 와 같이 백엔드와 프론트엔드 드라이버 간의 링버퍼를 통해 이루어진다. 이 때, 요청과 응답이 어떤 도메인에 쌓여 있는지를 알려주기 위한 방법으로 이벤트 채널을 둔다. 이러한 도메인간 비동기 통신 기법은 처리량을 높이는 데 큰 기여를 한다.



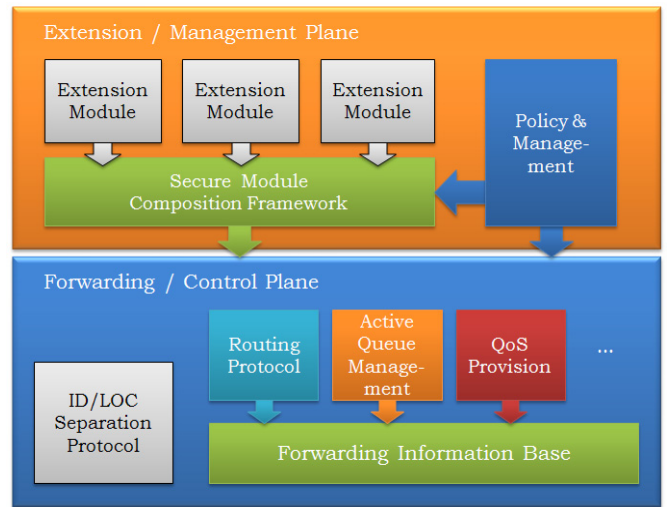
(그림 2) Xen 기반 I/O 가상화 구조

그러나 문제는 가상 머신의 수가 증가함에 따라 백엔드 드라이버를 가진 도메인을 거쳐 해당 도메인으로 패킷이 도착되기까지 많은 지연시간이 발생한다는 점이다. 기존 연구에 따르면[8], 포워딩 테이블을 각 도메인에서 관리하는 일반적인 가상화 구조상에서 가상 네트워크의 수가 10 개 이상이 되면 그 성능이 저하가 됨이 보고되었다. 즉, 현재의 구조는 확장성에 치명적인 약점이 존재하며, 이는 분리 드라이버 모델을 채택하고 있는 Xen 하이퍼바이저의 구조적 특징 때문이라고 볼 수 있다. 이에 대한 하나의 해결책으로 특정 도메인(Dom0)에서 포워딩을 전담하는 구조를 생각할 수 있다. 즉, 각 도메인으로 패킷을 교환하는 오버헤드를 줄임으로써 향상된 성능을 가져온다. 그러나 이러한 구조는 각 도메인의 독자적인 프로토콜 및 서비스 정책을 반영할 수 없다는 한계를 가진다.



(그림 3) 제안하는 Xen I/O 가상화 구조

이에 본 논문에서는 포워딩 테이블의 위치를 개별 도메인에 두고 독립성과 유연성을 확보하면서 확장성을 향상시키기 위한 구조를 제안한다. 제안하는 구조는 라우터에서 가장 중요한 디바이스 드라이버인 네트워크 인터페이스 카드에 대한 백엔드 드라이버가 안전하다는 가정 하에 이를 하이퍼바이저 내에 포함시키는 것이다(그림 3). 즉 물리 인터럽트가 쥘에 도달하면 인텔의 I/O 가상화 기술인 VT-d 기술[12]을 이용하여 별도의 도메인을 거치지 않고 해당 도메인으로 곧바로 전달한다. 따라서 하나의 도메인만 거치면 포워딩이 가능하기 때문에 기존의 쥘이 가지던 지연 시간을 효과적으로 제거할 수 있다.



(그림 4) 유연한 라우터 플랫폼 구조

### 4.3 유연성 있는 라우터 플랫폼

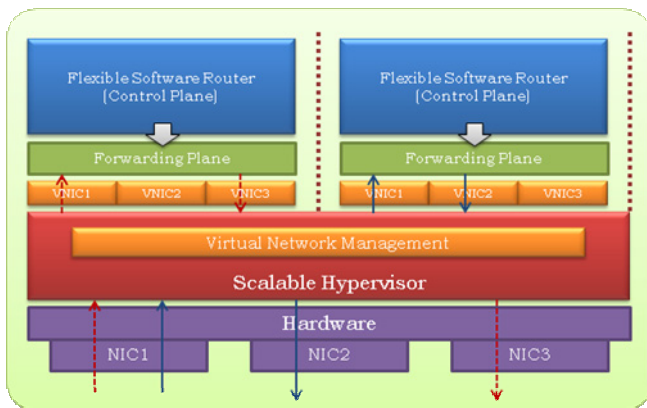
그림 4 는 제안하는 유연한 라우터 플랫폼 구조이다. 기본적으로 각 가상 머신 위에 하나의 독립적인 가상 라우터가 동작하게 되며, 서로 다른 프로토콜, 서비스, 운영 정책 등을 가질 수 있다. 따라서 가상 네트워크 관리자가 쉽게 다양한 기능들을 관리할 수 있도록 유연한 구조를 가져야 한다. 이는 게스트 운영체제 내에서 지원되어야 할 유연한 프로토콜 스택 부분(forwarding 및 control plane 포함)과 사용자 수준에서 지원되어야 할 확장 모듈 실행 환경을 포함한다.

먼저 유연한 프로토콜 스택의 경우 다양한 라우팅 프로토콜, 능동적 큐 관리 알고리즘, 그리고 QoS 정책 등을 제공할 수 있도록 쉽게 확장 및 재구성할 수 있는 구조를 가져야 한다. 또한 최근 미래 인터넷 환경에서 라우팅 테이블의 확장성, 이동성 및 멀티호밍 등의 서비스를 지원하기 위해 새로이 요구되는 ID/LOC 분리를 지원하기 위해, node ID 및 locator 에 대한 분리가 가능한 LISP(Locator Identifier Separation Protocol)과 같은 프로토콜을 포함할 수 있어야 한다 [9,10].

상위 확장 및 관리 영역의 경우, 새로운 네트워킹 기법 및 서비스나 프로토콜을 동적으로 배포하여 활용 가능하도록 하는 기능을 가진다. 이러한 서비스

및 프로토콜의 경우 작은 단위의 확장 모듈로 분리가 가능하며, 필요에 따라 특정 프로토콜에 대한 모듈 조합을 지원한다. 또한 동적으로 배포된 새로운 서비스 등을 안전하게 실행할 수 있는 모듈 조합 프레임워크가 구현되어야 한다. 이렇게 새로이 구성된 프로토콜 및 서비스들을 네트워크 프로토콜 스택에 전달하여 동적으로 재구성하여 사용할 수 있도록 한다. 마지막으로, 서로 다른 서비스 및 프로토콜 정책을 관리할 수 있는 컴포넌트를 포함하여, 현재 보유한 기능들을 동적으로 재구성하거나 필요하면 다운로드 받을 수 있도록 한다.

전체 가상 라우터 구조는 그림 5 와 같다.



(그림 5) Xen 기반 가상 라우터 구조

## 5. 결론

본 논문에서는 가상 네트워크를 지원하기 위해 Xen을 사용한 가상 라우터 구조를 제안하였다. Xen은 하이퍼바이저 기반의 가상머신 플랫폼으로써 가상 네트워크 간 독립성을 지원해줄 수 있다. 한편 부족한 성능은 Xen의 I/O 구조의 특징인 드라이버 도메인을 사용하지 않고 라우터의 네트워크 인터페이스 카드에 대한 드라이버를 하이퍼바이저에 위치시킴으로써 도메인간 통신에 의한 오버헤드를 없앴으로써 완화시키고자 하였다. 마지막으로 유연성을 제공하기 위해 도메인에 올라가는 유연성 있는 라우터 플랫폼 구조를 제안하였다.

추후 제안한 구조를 Xen에서 구현하여 실제 가상 네트워크를 구축할 예정이다.

## 참고문헌

- [1] Clean Slate Design for the Internet, <http://cleanslate.stanford.edu/>
- [2] Click Modular Router Project, <http://read.cs.ucla.edu/click>
- [3] Robert Morris, Eddie Kohler, John Jannotti, "The Click modular router", ACM SIGOPS Operating Systems Vol. 33 Issue 5, 2002.02.08.
- [4] 백은경, 황진경, 최은호, "미래인터넷 서비스 연구 동향", 한국정보통신기술협회(TTA) 논문지 NO. 124, p43~46, 2009.07

- [5] 변성혁, "미래인터넷 아키텍처 연구동향", 전자통신동향분석 제 24 권 제 3 호 p1~12, 2009.06
- [6] 백은경, 신명기, "미래 네트워크 기술 동향" 주간기술 동향 통권 1387 호 2008.03.11
- [7] Mark Handley, Orion Hodson, Eddie Kohler, "XORP: an open platform for network research", ACM SIGCOMM Computer Communication Volume 33, Issue 1, 2003.01
- [8] 이권용, 박성용, "미래인터넷의 라우터 가상화 기술동향", 한국정보통신기술협회(TTA) 논문지 NO. 124, p60~65, 2009.07
- [9] T. Li, "Design Goals for Scalable Internet Routing", draft-irtf-rrg-design-goals-01.txt, IETF Network Working Group, Jul. 2007
- [10] D. Farinacci, V. Fuller, and D. Meyer, "Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-05.txt, IETF Network Working Group, Sep. 2009
- [11] Welcome to xen.org, home of the Xen; hypervisor, the powerful open source industry standard for virtualization, <http://www.xen.org/>
- [12] Intel Virtualization Technology for Directed I/O, <http://www.intel.com/technology/itj/2006/v10i3/2-io/7-conclusion.htm>