

유무선 통합망에서의 SDN 기반 미들박스 관리 프레임워크

이 기 원*, 장 인 선*, 김 원 태*, 주 석 진*, 김 명 수*, 백 상 현°, 강 철 희*

SDN-Based Middlebox Management Framework in Integrated Wired and Wireless Networks

Giwon Lee*, Insun Jang*, Wontae Kim*, Sukjin Joo*, Myungsoo Kim*,
Sangheon Pack°, Chul-Hee Kang*

요 약

최근 네트워크가 점점 복잡해짐에 따라 많은 수의 미들박스를 동적으로 유연하게 관리할 필요성이 증가하고 있으며, 미들박스 설정이 사업자의 정책과 다르게 잘못 설정되는 경우가 빈번하기 때문에 미들박스를 효율적으로 관리할 수 있는 기법이 절실한 상황이다. 이러한 미들박스 관리의 어려움은 SDN (Software Defined Networking)의 중앙 집중화된 컨트롤러 구조와 유연한 프로그래밍 능력을 통해 해결할 수 있다. 즉, 네트워크 상황에 맞춰 동적으로 미들박스 정책을 적용하고, 데이터/컨트롤 평면의 분리를 통해 기존 미들박스 구조는 그대로 유지한 채 새로운 컨트롤 평면을 추가하는 것이 가능하다. 또한 클라우드와 분산 네트워크 기능 가상화 (NFV : Network Function Virtualization) 기술을 통해 보다 유연하게 미들박스를 관리하는 방안도 가능하다. 본 논문에서는 유선망과 무선망이 통합된 네트워크에서 SDN 기반의 미들박스 관리 기법과 클라우드 기반의 미들박스 관리 방안에 대해 알아보고 향후 연구 이슈에 대해서 살펴본다.

Key Words : Middlebox, software defined networking (SDN), network function virtualization (NFV), wired-wireless integrated networks, fixed mobile convergence

ABSTRACT

Recently, middleboxes play a key role in many network settings such as firewalls, VPN gateways, proxies, intrusion detection and prevention systems, and WAN optimizers. However, achieving the performance and security benefits that middleboxes offer is highly complex, and therefore it is essential to manage middleboxes efficiently and dynamically. In this respect, Software-Defined Networking (SDN) offers a promising solution for middlebox policy enforcement by using logically centralized management, decoupling the data and control planes, and providing the ability to programmatically configure forwarding rules. Also, cloud computing and distributed Network Function Virtualization (NFV) can enable to manage middleboxes more easily. We introduce SDN-based middlebox management framework in integrated wired and wireless networks and discuss the further issues.

* 본 연구는 2014년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(과제번호: NRF-2014R1A2A2A010 04832 및 NRF-2014K1A3A1A21001357)

♦ First Author : Giwon Lee, Korea University Department of Electrical Engineering, goodkw@korea.ac.kr, 학생회원

° Corresponding Author : Sangheon Pack, Korea University Department of Electrical Engineering, shpack@korea.ac.kr, 종신회원

* 고려대학교 전기전자공학과, {zerantoss, munte2, choo0128, myeongsukim, chkang}@korea.ac.kr

논문번호 : KICS2014-05-197, Received May 27, 2014; Revised June 9, 2014; Accepted June 9, 2014

I. 서 론

현재 네트워크에서는 보안 유지, 성능 향상 등을 위해 다양한 형태의 미들박스 (Middlebox)를 운영하고 있으며 이러한 미들박스는 통신 사업자의 목적/정책에 맞추어 올바르게 동작해야 한다. 미들박스란 특정 지능적 기능을 가진 네트워크의 중간자로서 서비스를 수행하는 장비이다. 하지만 현재 사용되고 있는 NAT 나 Firewall, IDS, Proxy 등의 많은 미들박스 기능은 네트워크 변화에 따라 동적으로 정책을 변경하기 어렵다는 문제가 존재한다. 즉, 네트워크가 점점 복잡해짐에 따라 많은 수의 미들박스를 유연하게 동적으로 관리할 필요성이 증가하고 있지만 기존의 미들박스 관리 기법들은 관리자가 수동적으로 정책을 미들박스에 반영하기 때문에 동적인 상황에 대처하기 어려우며, 높은 복잡성과 관리비용을 초래한다는 문제점이 있다. 최근 57개 네트워크 사업자에 대한 조사 결과에 따르면^[1], 미들 박스를 수십개-수백개, 심지어 수천개를 유지하는 네트워크 사업자가 많고 2000개 정도의 미들박스를 유지하기 위해서는 관리 인력만 500명이 넘는 경우가 많다. 그럼에도 불구하고 미들박스의 설정이 사업자의 정책과 다르게 설정되는 비율이 50%가 넘는다고 한다. 따라서 미들박스를 효율적으로 관리할 수 있는 기법이 절실한 상황이다. 이러한 미들박스 관리의 어려움은 바로 SDN (Software Defined Networking)^[2-4]을 통해 해결할 수 있다. 즉, SDN의 중앙 집중화된 컨트롤러 구조와 유연한 프로그래밍 능력을 통해 네트워크 상황에 맞춰 동적으로 미들박스 정책을 적용하는 것이 가능하며, 데이터/컨트롤 평면이 분리되어 있으므로 기존 미들박스 구조는 그대로 유지한 채 새로운 컨트롤 평면을 추가하는 것도 가능하다.

그림 1은 SDN/OpenFlow^[5] 기반의 미들박스 관리 구조를 보여준다. 중앙 집중식 관리가 가능한 SDN 구조를 활용하여 유연한 미들박스 관리가 가능하지만 미들박스 관리는 기존의 OpenFlow에서 정의하는 플로우 경로 설정 이상의 기능이 필요하기 때문에 이에 대한 추가적인 보완/개선이 필요하다.

한편 SDN/OpenFlow 기반의 미들박스 관리 방법 이외에도 최근에는 네트워크 기능 가상화 (NFV : Network Function Virtualization)^[6] 및 클라우드^[7]를 이용한 미들박스 관리 방법이 이슈가 되고 있다. 여러 기능의 미들박스들을 클라우드에 위치시켜 통합적으로 관리하거나 분산 NFV를 통해 필요에 따라 범용 서버에 미들박스 기능을 위한 소프트웨어 모듈을 동

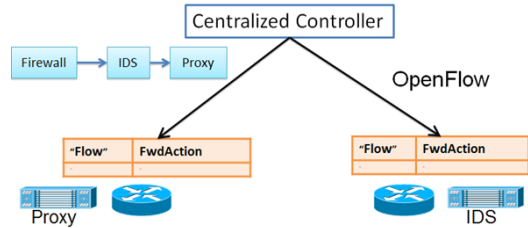


그림 1. SDN 기반의 미들박스 관리 구조
Fig. 1. SDN-based middlebox management framework

적으로 설치하여 운용할 수 있다. 이러한 환경에서는 트래픽 증가에 따른 미들박스 처리 부하 증가를 유연하게 동적으로 처리할 수 있지만, 이를 위한 효과적인 미들박스 동적 관리 알고리즘 및 미들박스 인스턴스 생성 및 관리 등의 연구/개발이 필요하다.

최근 SDN이나 클라우드/NFV를 활용하여 미들박스를 효과적으로 처리하기 위한 방안들이 다수 제안되었지만 대부분 유선망 또는 무선망과 같은 단일망만을 고려하고 있다. 하지만, 대부분의 네트워크 사업자들을 유선망과 무선망을 모두 운영하고 있으며 사용자 서비스 역시 유선망과 무선망이 통합된 환경에서 제공할 필요가 있다. 따라서 SDN 기반의 유무선 통합망 구조에서 전체적인 미들박스의 활용도를 높일 수 있는 새로운 구조가 필요하다. 따라서 본 논문에서는 유선망과 무선망이 통합된 환경에서 SDN 기반의 미들박스 관리 프레임워크와 클라우드 기반의 미들박스 관리 프레임워크에 대해 살펴보고 향후 연구 이슈에 대해 살펴본다. 본 논문의 구성은 다음과 같다. 2장에서는 유선망과 무선망이 통합된 환경에서 SDN 기반의 미들박스 관리 프레임워크와 클라우드 기반의 미들박스 관리 프레임워크에 대해 살펴본다. 3장에서는 2장에서 설명한 기법들에 대한 분석과 향후 연구 이슈에 대해 소개하고 4장에서 결론을 맺는다.

II. 유무선 통합망에서의 SDN 기반 미들박스 관리 프레임워크

본 장에서는 세 가지의 접근방법에 따른 유무선 통합망에서의 SDN 기반 미들박스 관리 프레임워크에 대해 알아본다. 우선 대표적인 SDN 기반의 미들박스 관리를 위한 프레임워크로는 SIMPLE (Software defined Middlebox Policy Enforcement)^[8]과 FlowTags^[9]가 있다. SIMPLE은 기존 미들박스의 구조를 변경하지 않는 접근 방법이며 반면 FlowTags는 기존의 미들박스를 확장하는 접근 방법이다. 또한 클

클라우드를 기반으로 하는 접근 방법도 있다. 이 방법은 미들박스들을 클라우드에 중앙집중적으로 아웃소싱하여 관리하는 방식과 분산 NFV를 활용하여 관리하는 방식으로 나눌 수 있다.

2.1 기존 미들박스 기반의 접근 방법

기존 미들박스 기반의 접근 방법은 미들박스들은 수정하지 않고 컨트롤러의 제어구조를 변경하는 방식을 의미한다. 그림 2는 기존의 유무선 통합망을 관리하는 컨트롤러에 미들박스를 제어하기 위한 별도의 제어 프레임워크를 추가하는 구조를 보여준다. 이 제어 프레임워크는 크게 세 가지 기능을 수행한다. 첫째, 네트워크 및 미들박스의 상태를 모니터링하여 미들박스 관리를 위한 전체적인 뷰를 구성한다. 그 예로 토폴로지/트래픽 부하/스위치와 미들박스의 용량 등이 있다. 둘째, 앞서 구성한 뷰를 바탕으로 네트워크 상황에 따라 미들박스를 제어하기 위해 능동적으로 미들박스/스위치들의 제어방식을 결정한다. 마지막으로, 결정된 제어방식에 따라 동작할 수 있도록 미들박스/스위치의 스펙에 맞게 제어신호를 미들박스/스위치에 전달한다.

기존 미들박스 기반의 접근 방법의 대표적인 예로 SIMPLE이 있다. SIMPLE은 SDN 구조를 가정하고 있으며 그림 3과 같이 제어 평면에 Resource Manager, Dynamic Handler, Rule Generator로 이루어진 Policy Enforcement layer를 도입하였다. Resource Manager는 네트워크 토폴로지, 스위치와 미들박스의 용량 같은 정적인 요소 뿐만 아니라 트래픽 부하 같은 동적인 요소들을 종합적으로 고려하여 최적의 미들박스 처리 방안을 제안한다. 예를 들어, 미들박스로 유입되는 트래픽의 양을 고려하여 최적화 문제를 정의한다. 최적화 문제를 CPLEX와 같은 툴을

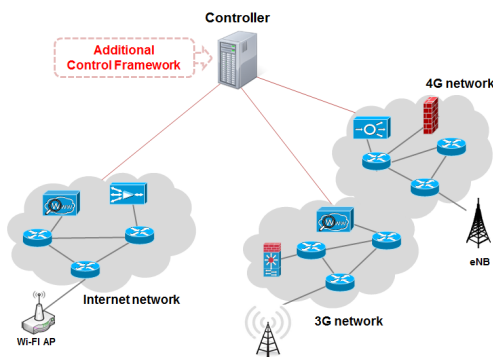


그림 2. 기존 미들박스 기반의 접근 방법
Fig. 2. Approach using existing middlebox function

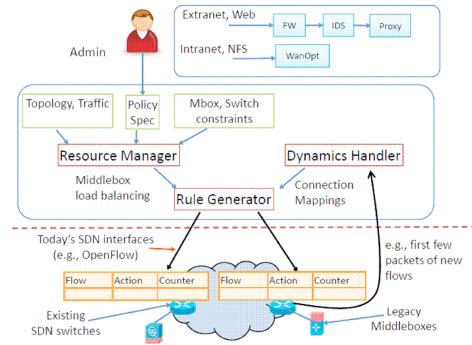


그림 3. SIMPLE 구조^[8]
Fig. 3. SIMPLE structure

이용하여 솔루션을 도출해 미들박스에게 부하를 할당함으로써 로드밸런싱 효과를 얻을 수 있다.

Dynamic Handler는 헤더나 페이로드가 변경된 패킷에 대해 유사성 검증을 통해 정책을 적용시키는 역할을 한다. 미들박스 중 일부(NAT, WAN Optimizer 등)는 패킷의 헤더나 페이로드를 변경한다. 이러한 변경이 있는 경우 어떤 플로우에 속하는지 확인하기 어려워 올바른 정책을 적용시키지 못할 수도 있다. 이 문제를 해결하기 위해 Dynamic Handler는 미들박스의 입력과 출력 패킷의 유사성을 검증하는 알고리즘을 수행한다. 유사성 검증 알고리즘을 통해 패킷이 어떤 플로우에 속하는지 판단하고 결과에 따라 적절한 정책을 적용한다.

Rule Generator는 Policy Enforcement layer에서 결정된 정책을 데이터 평면에 적용하는 역할을 한다. Rule Generator는 기존의 SDN에서 사용하는 OpenFlow를 지원하는 인터페이스를 기반으로 동작하며 Flow table을 정책에 맞게 수정한다. 스위치는 플로우들이 도착하면 플로우에 명시된 대로 플로우를 처리하여 Policy Enforcement layer에서 결정한 정책대로 플로우가 전달될 수 있도록 한다. SIMPLE은 유선 네트워크에 초점을 맞추고 있지만, 무선 네트워크에서도 SDN 구조, 즉 OpenFlow를 제공하는 인터페이스를 갖출 수 있다면 Policy Enforcement layer를 적용하여 유무선 통합망에 활용하는 것이 가능하다. 하지만 SIMPLE은 유선망의 토폴로지와 논리적인 뷰만을 제공하고 있으므로, 유무선 통합망에서의 미들박스 관리를 위해서는 유무선이 통합된 뷰를 제공할 수 있어야 한다.

2.2 미들박스 확장 기반의 접근 방법

미들박스 확장 기반의 접근 방법은 기존의 미들박

스 기능을 확장하여 컨트롤러가 제어하는 방법이다. 그림 4와 같이 유무선 통합망의 기존 미들박스들을 관리하는 중앙 컨트롤러가 존재하고 네트워크 내부의 미들박스들과 SDN 스위치들의 기능을 확장하여 컨트롤러가 별도의 부가적인 제어 기능을 수행하는 구조이다.

이러한 접근 방법의 대표적인 예로 FlowTags가 있다. FlowTags는 미들박스들의 구조를 일부 변경하여 패킷 헤더에 적절한 Tag를 추가하는 방식이다. 이는 유무선 통합망에 배치되어 있는 미들박스들의 구조만 일부 변경할 수 있다면 적용할 수 있다. 그림 5는 FlowTags를 지원하는 중앙의 컨트롤러가 스위치들과 미들박스들을 제어 및 관리하는 구조를 보여주고 있다. 이 때 FlowTags를 지원하는 컨트롤러는 해당 패킷을 어떻게 미들박스들 상에서 효율적으로 처리할 것인지를 결정하여 Tag값을 부여하는 기능을 수행한다. 또한 각각의 미들박스들과 컨트롤러는 FlowTags API를 통하여 FlowTags 설정 정보를 교환하고, 이를 통해 각각의 미들박스들은 Tag값에 따라 적절하게 패킷을 처리하는 기능을 수행한다. 반면에 스위치들은 Tag를 처리하는 기능은 없고 단지 Tag값에 따라 패킷을 전달하는 기능을 수행한다. 이때 컨트롤러와 스위

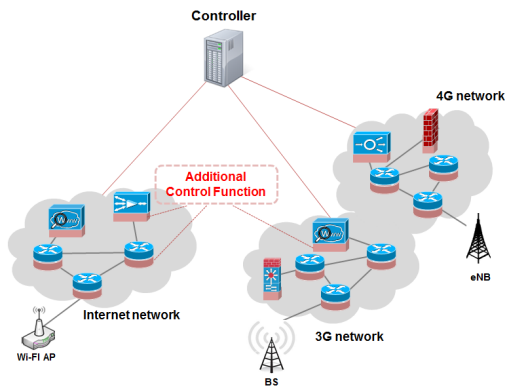


그림 4. 미들박스 확장 기반의 접근 방법
Fig. 4. Approach using extension of middlebox function

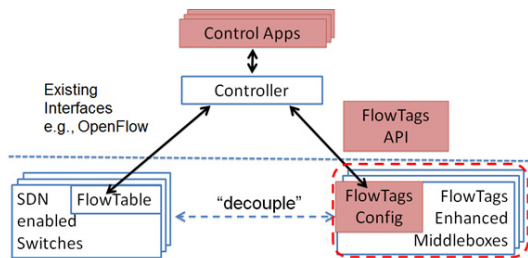


그림 5. FlowTags 구조^[9]
Fig. 5. FlowTags structure

치들은 기존에 OpenFlow와 같은 인터페이스를 통해 Flow table 설정 정보를 교환한다.

그림 6에 FlowTags의 자세한 동작과정이 나타나 있다. 먼저 패킷이 Proxy에 도착한 경우 Proxy는 해당 패킷과 Proxy의 상태정보를 FlowTags API를 통해 중앙 컨트롤러에게 전달하며 적합한 Tag 할당 및 처리를 요청한다. 이로 인해 컨트롤러는 전달받은 정보들을 바탕으로 해당 패킷을 미들박스들 상에서 어떻게 처리할 것인지 판단을 하여 패킷의 헤더에 적절한 Tag를 할당하고 Tag에 대한 정보를 미들박스들과 스위치들에게 알린다. 이를 통해서 Proxy는 TagFlowTable을 설정하여 패킷을 처리하고, 스위치는 Flow table을 설정한 뒤 Tag값에 따라 적절하게 패킷을 전달한다. 그 이후 Tag값을 지닌 패킷이 ACL(Access Control List)로 전달된 경우 ACL은 패킷과 Tag 정보를 다시 중앙 컨트롤러에게 전달하여 해당 Tag를 어떻게 처리해야 하는지에 대한 Action을 질의를 하고, 컨트롤러는 해당 Tag에 대한 적절한 Action을 미들박스들과 스위치에게 알린다. 이를 바탕으로 ACL은 TagsActionTable을 생성하고 스위치는 FlowTable을 설정하여 해당 패킷을 처리하게 된다. 이와 같이 미들박스들의 기능을 확장하여 중앙의 컨트롤러가 제어한다면 복잡하고 다수의 미들박스들이 구성되어 있는 유무선 통합 환경에서도 패킷을 보다 효율적으로 전달 및 처리할 수 있다. 하지만, FlowTag를 그대로 유무선 통합망에 적용하기에는 각 단일망에 대해 여러 컨트롤러가 존재할 수 있기 때문에, 각 컨트롤러 간에도 해당 플로우의 Tag 식별 정보를 교환하거나 처리할 수 있는 기능이 추가되어야 한다. 따라서 컨트롤러끼리 정보를 주고 받을 수 있는 표준화된 API가 필요하다.

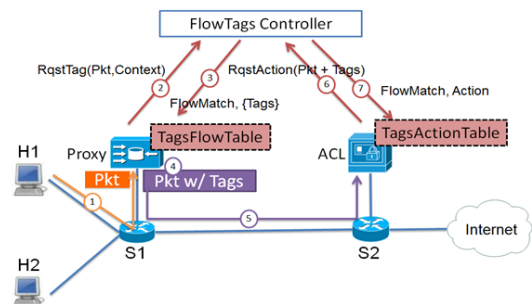


그림 6. FlowTags 동작과정^[9]
Fig. 6. FlowTags operation process

2.3 클라우드 기반의 접근 방법

클라우드 기반의 접근 방법은 크게 중앙집중적으로

아웃소싱하여 관리하는 방식과 분산적인 NFV로 관리하는 방식으로 나눌 수 있다. 중앙집중적으로 관리하는 방식은 그림 7과 같이 기존의 분산 배치되어 있는 미들박스를 클라우드에 집중하여 배치시키고, 유무선 통합망을 관리하는 컨트롤러가 트래픽을 중재하여 미들박스가 있는 클라우드로 향하게 해주는 구조를 가진다. 이러한 구조는 미들박스를 여러 분산된 유무선 네트워크에 배치하는데 드는 설치/운영 비용을 절감하고, 네트워크 전체의 망 복잡도를 낮출 수 있다. 클라우드로 미들박스를 아웃소싱 하는 상태에서 컨트롤러는 각 호스트들의 트래픽에 대해 목적에 맞는 미들박스가 있는 클라우드로 해당 트래픽을 전달할 수 있게 한다. 특히, 유무선 통합 네트워크 및 미들박스의 상태를 모니터링 할 필요 없이 트래픽만을 전달해주며, 미들박스를 추가 및 변경하기 위해서는 클라우드에 있는 미들박스만을 변경하면 된다. 또한, 전달한 트래픽량을 바탕으로 사용한 클라우드의 사용량을 쉽게 측정 할 수 있다. 이를 통해 클라우드 사용량에 기반한 경제적 과금을 손쉽게 체크 할 수 있다.

중앙집중적 클라우드 기반의 접근 방법의 대표적인 예로 APLOMB^[11]와 Stratos^[10]가 있다. APLOMB는 게이트웨이로써, 그림 8과 같이 미들박스를 클라우드에 배치하고 특정 네트워크로 향하는 트래픽을 제어하여 미들박스를 관리한다. 즉, 외부에서 네트워크로 유입되는 트래픽은 클라우드의 미들박스로 전달하고, 미들박스 처리가 완료된 트래픽은 다시 본래의 목적지로 전달하여 준다. 하지만 APLOMB를 통하여 미들박스 처리를 위한 트래픽이 한 클라우드에 몰릴 수 있는 동시에 지연시간이 길어질 수 있다. 이를 해결하기 위해, APLOMB는 미들박스가 여러 클라우드에 위치해 있을 경우에 여러 클라우드 중 가장 작은 지연시간

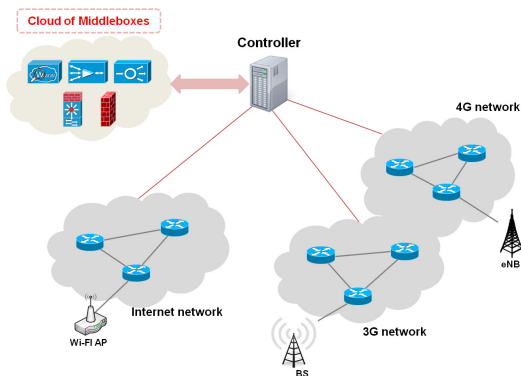


그림 7. 클라우드 기반의 접근 방법
Fig. 7. Approach using cloud computing

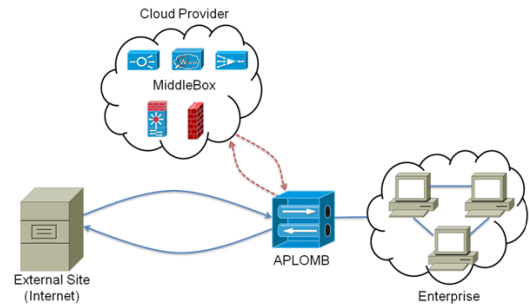


그림 8. APLOMB 구조^[11]
Fig. 8. APLOMB structure

을 갖는 클라우드를 선택하는 알고리즘을 사용한다. 이는 유무선 통합망에서 한 클라우드에서 미들박스를 관리하기 어려운 점에 있어서 각 망에 대해 여러 클라우드가 배치되었을 때 이용 가능하다. APLOMB는 외부에서 클라우드로 들어가는 트래픽 양을 관찰 할 수 있으며, 이에 기반한 클라우드 트래픽 사용량에 대해 과금이 가능케 하는 장점이 있다.

Stratos는 클라우드 내의 미들박스를 효과적으로 관리하기 위한 네트워크 인지형 통합계층이다. 이는 클라우드 내의 미들박스 체인에 대한 논리적인 뷰를 제공하여, 상황에 맞게 응용별로 네트워크 인지형 스케일링 (Scaling) 기능을 제공한다. 이를 이용하여 클라우드 내의 병목 현상을 낮추고 트래픽을 균등하게 분배하여 자원 활용도를 높이고 비용을 낮추는 것이 가능하다. 즉, Stratos는 유무선 통합망에서 각 네트워크에서 필요로 되는 미들박스를 클라우드 내에 배치시켰을 때 효율적인 관리를 가능케 한다. 특히, 이기종 네트워크에 대한 미들박스가 중첩되었을 때, 미들박스 체인에 대한 논리적인 뷰를 통해 미들박스 기능들을 제공할 수 있다.

분산 NFV를 활용한 클라우드 기반의 접근 방법은 그림 9와 같이 여러 미들박스들의 기능들을 소프트웨어 모듈 형태로 x86 기반의 범용 서버 (즉, 클라우드)에 설치하고, 유무선 통합망에서 필요로 하는 미들박스 기능을 각 표준 가상화 장비 (예, 가상화 스위치)에 동적으로 설치하여 분산적으로 운영하는 방식이다. 따라서 트래픽이 급격하게 증가하는 경우에도 필요로 하는 미들박스 가상 머신만을 동적으로 설치하여 수행함으로써 분산적인 처리가 가능하다. 이로 인해 유무선 통합망의 복잡하고 다양한 네트워크 환경에서 미들박스를 더욱더 용이하게 관리할 수 있다.

앞서 설명한 APLOMB와 Stratos에서 소프트웨어적으로 가상화하여 동적으로 클라우드에서 운영하는

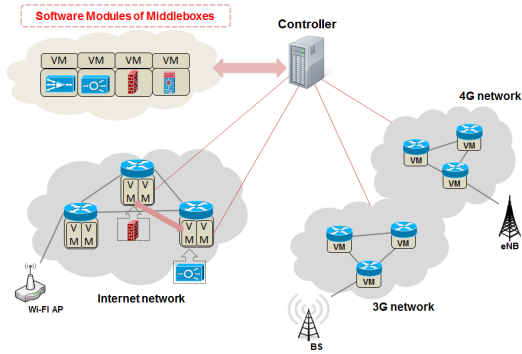


그림 9. 분산 NFV 기반의 접근 방법
Fig. 9. Approach using distributed NFV

방안 (즉, NFV 접근방법)에 대해서도 언급은 하고 있지만 유무선 통합망에 분산 NFV 적용을 위한 세부적인 기술 개발을 위해서는 다음과 같은 이슈를 고려해야 한다. 먼저, 통합된 정보 모니터링 및 유무선 통합망에 대한 전체적인 네트워크 뷰 구성방안과 이를 기반으로 한 효과적인 미들박스 관리 알고리즘이 필요하다. 예를 들어, 각 스위치에 미들박스 기능을 추가로 할당하고 해체하는 미들박스 처리 부하 변화에 따른 동적 관리 알고리즘이 있을 수 있다. 다음으로, NFV 규격 기반의 아키텍처/프로토콜 규격화 및 구체화가 필요하며, 호환성을 위해 전용 장비 기반의 미들박스와 x86 기반의 가상화된 미들박스가 혼용된 환경에서의 효과적인 미들박스 관리 방안이 필요하다. 또한, 기존 미들박스 관리 방법에 비해서 유무선 통합망을 관리하는 통신 사업자 수준의 성능 보장 여부를 확인을 위한 테스트베드를 구축하여 성능 향상 방안을 마련할 필요가 있다.

III. 정성적 분석 및 유무선 통합망에서의 SDN 기반 미들박스 관리의 추가 연구 이슈

본 장에서는 앞서 설명된 유무선 통합망에서의 SDN 기반 미들박스 관리 기법들에 대해 분석하고, 추가적인 연구 이슈들에 대해 알아본다.

SIMPLE과 FlowTags 두 기법 모두 SDN을 이용한 미들박스 관리의 복잡성을 개선하고 효과적인 관리를 위한 방법이지만, SIMPLE은 기존 미들박스 기반의 접근방법이고, FlowTags는 미들박스 확장 기반의 방법으로 접근 방법 측면에서 다른 점이 있다. SIMPLE은 미들박스 또는 OpenFlow (즉, 스위치)의 수정 없이 별도의 제어 평면을 통해 기능을 제공하기 때문에 기존의 미들박스를 그대로 이용할 수 있어 범용성이

있지만, FlowTags는 미들박스와 OpenFlow 자체에 대한 수정 (즉, OpenFlow 기능 확장)이 필요하며, 간단한 미들박스 배치 문제 해결 등의 지원되는 기능이 제한된다. 즉, 보다 다양하고 일반적인 시나리오에 대한 검증이 필요하다. 하지만, FlowTags는 태그 정보를 추가하는 간단한 방법을 통해 미들박스 수의 증가에 따른 복잡성과 오버헤드가 SIMPLE에 비해 낮을 수 있다는 장점이 있다.

또한, APLOMB와 Stratos는 모두 미들박스 기능을 클라우드에서 운용, 관리하는 방법을 제안하고 있다. 기존 미들박스 관리 방안의 경우 높은 CAPEX/OPEX를 가지고 있는데 비해서 클라우드 환경에서 미들박스를 관리하게 되면 컴퓨팅 자원을 효과적으로 관리하는 클라우드 컴퓨팅의 특성으로 인해 보다 경제적으로 미들박스를 관리할 수가 있다. 또한, 클라우드 내의 자원만 할당받으면 되기 때문에 네트워크에 분산된 미들박스의 재설정, 오류 복구를 위해 소비되는 시간과 노력을 감소시킬 수 있다. 미들박스의 물리적인 결함이나 과도한 부하가 발생한 경우에도 클라우드 내의 자원을 추가적으로 할당하거나 클라우드 자체의 결함 복구 기법을 통해 쉽게 해결할 수 있다. 즉, 유연한 자원관리가 가능하다. 하지만, 이러한 클라우드 기반의 미들박스 관리 기법도 해결해야 하는 다양한 이슈가 존재한다. 우선 이를 적용하기 위한 유무선 통합망의 복잡도를 최소화해야 하며, 동시에 기존의 미들박스와 동일한 기능을 제공하여야 한다. 또한, 클라우드에 위치한 미들박스를 사용하는 데에 따른 지연시간 및 트래픽 부하등의 성능 오버헤드를 최소화하여야 한다.

유무선 통합망에서의 SDN 기반 미들박스 관리의 추가 연구 이슈는 크게 두 가지로 나눌 수 있으며 다음과 같다. 첫째, 미들박스를 통합관리하기 위해서는 중앙 컨트롤러가 네트워크 및 미들박스의 상태를 모니터링하여 논리적인 글로벌 뷰 형태로 관리자에게 제공해야 한다. 이를 위해서는 다양한 네트워크 및 미들박스들의 상태를 특정 벤더사에 관계없이 통합적으로 중앙 컨트롤러에게 정보를 제공할 수 있어야 한다. 특히, 기존의 단일망에서의 미들박스들을 관리하는 것과는 달리 유무선 통합망의 경우 미들박스가 Internet, 3G network, 4G network 등에 분산되어 있기 때문에 이를 통합하여 관리할 수 있는 방안이 필요로 하다. 또한 각 단일망에서의 별도의 컨트롤러가 존재하는 경우 컨트롤러 간의 통신을 위해 East-Westbound 인터페이스를 활용할 수 있을 것이다. 그리고 유무선 통합망에서의 미들박스들을 관리하는 경우 수많은 미들

박스들을 중앙의 컨트롤러에서 모두 수용할 수 있는지 확장성(Scalability) 측면에서 고려해 보아야 할 것이다. 이를 해결하기 위한 방안으로는 물리적인 컨트롤러를 분산적으로 여러 대 사용하되 논리적으로는 하나의 중앙컨트롤러를 사용하는 것처럼 구성하여 해결할 수 있을 것이다. 또한 중앙컨트롤러에서 논리적인 글로벌 뷰를 기반으로 한 동적인 미들박스 처리를 위해 컨트롤러-스위치 혹은 미들박스에 OpenFlow와 같은 인터페이스를 정의해야 할 필요가 있고, 사업자들 간의 복잡하고 다양한 관리 정책을 낮은 복잡도로 해석하여 처리 할 수 있는 언어 혹은 인터페이스가 필요하다.

둘째, 미들박스를 네트워크 상황에 맞게 동적으로 운용할 수 있어야 한다. 이를 위해 II장 중앙집중적 및 분산 NFV를 활용한 클라우드 접근 방법을 이용할 수 있다. 하지만, 이 방법을 이용하기 위해서는 NFV 규격 기반의 아키텍처 혹은 프로토콜 규격화 및 구체화가 필요하고, 기존의 전용 장비 기반의 미들박스 와 X86 기반의 NFV화 미들박스가 혼용된 환경도 고려해야 한다. 또한, 미들박스를 처리 부하 변화에 따른 동적 관리 알고리즘을 필요로 한다. 예를 들어 미들박스 기능의 추가 할당 및 해제 알고리즘, 과금, QoS 등이 있다. 하지만, 효율적인 적용을 위해 클라우드의 물리적 토폴로지 혹은 네트워크 자원 상황을 종합적으로 고려한 미들박스 인스턴스 생성 및 관리방안이 필요하다. 즉, 체인 형태의 미들박스 처리 및 단말의 이동성을 고려하여 미들박스 VM을 배치하는 방안 등을 고려해야 한다. 또한, 미들박스 기능의 추가 할당 및 해제 등으로 인해 토폴로지가 변화하거나 서비스 요구사항이 달라져 서비스 체인을 새로 구성해야 하는 경우 동적으로 최적의 라우팅 경로를 재설정하는 알고리즘에 관한 연구가 필요할 것이다. 이때 토폴로지 변화에 따라 빠르게 새로운 경로를 설정해야 하며 이를 위해서는 복잡도가 작은 알고리즘과 유연하게 라우팅을 적용할 방안이 필요하다. 최종적으로는 통신 사업자 수준의 유무선 통합망 성능 보장여부 확인을 위한 테스트베드 구축 및 성능 향상 방안이 필요하다.

IV. 결 론

본 논문에서는 유무선 통합망에서 기존 미들박스 관리의 문제점을 해결할 수 있는 SDN 기반의 미들박스 관리 프레임워크를 소개하였다. SDN의 중앙 집중화된 컨트롤러와 프로그래밍 능력을 통해 네트워크

상황에 맞춰 동적으로 미들박스를 관리할 수 있다. 또한, SDN 이외에도 여러 미들박스 기능을 클라우드에 위치시켜 통합적으로 관리하는 방법과 분산 NFV를 통해 여러 미들박스 기능을 소프트웨어 모듈로 동적으로 설치하여 운영하는 방법들도 존재한다. 이러한 방법들이 트래픽 증가에 따른 미들박스 처리 부하 증가를 유연하게 처리할 수 있다. 하지만 유선망과 무선망이 통합된 논리적 뷰나, 네트워크 혹은 트래픽 상황에 따른 효과적인 동적 관리 알고리즘, 아키텍처/프로토콜 구체화 등의 추가적인 연구가 더 필요하다. 궁극적으로는 SDN 기반의 미들박스 관리를 통해 통신 사업자 수준의 성능과 유연한 자원관리를 모두 보장할 수 있는 네트워크 관리가 가능할 것으로 예측된다.

References

- [1] J. Sherry, et al., "Making middleboxes someone else's problem: Network processing as a cloud Service," in *Proc. ACM SIGCOMM 2012*, Aug. 2012.
- [2] Open Networking Foundation. (2012, April), "Software-defined networking: The new norm for networks," [Online]. Available: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf>
- [3] J. Jo, et al., "A centralized network policy controller for SDN-Based service overlay networking," *J. KICS*, vol. 38B. no. 4, pp. 266-278, Apr. 2013.
- [4] I. Cho, et al., "VIMS: Design and implementation of virtual network integrated control and management framework over national research network," *J. KICS*, vol. 37B. no. 10, pp. 877-888, Oct. 2012.
- [5] Open Networking Foundation. (2011, February), "Open flow switch specification," [Online]. Available: www.opennetworking.org
- [6] "Network functions virtualization (NFV) - an introduction, benefits, enablers, challenges & call for action," *SDN and OpenFlow World Congress 2012*.
- [7] H. Kim, et al., "Control algorithm for virtual machine-level fairness in virtualized cloud data center," *J. KICS*, vol. 38C. no. 6, pp.

512-520, Jun. 2013.

- [8] Z. Qazi, et al., "SIMPLE-fying middlebox policy enforcement using SDN," in *Proc. ACM SIGCOMM 2013*, Aug. 2013.
- [9] S. Fayazbakhsh, et al., "FlowTags: Enforcing network wide policies in the presence of dynamic middlebox actions," in *Proc. ACM HotSDN 2013*, Aug. 2013.
- [10] A. Gember, et al., "Stratos: Virtual middleboxes as first-class entities," Technical Report TR1771, University of Wisconsin-Madison, 2012.

이 기 원 (Giwon Lee)



2009년 2월 : 고려대학교 전기
전자전파공학부 학사
2011년 2월 : 고려대학교 전기
전자전파공학부 석사
2011년~현재 : 고려대학교 모바일
솔루션 학과 박사과정
<관심분야> 미래 인터넷, 다중
무선 인터페이스 네트워크

장 인 선 (Insun Jang)



2011년 2월 : 고려대학교 전기
전자전파공학부 학사
2011년~현재 : 고려대학교 전
기전자공학과 석·박사과정
<관심분야> 미래 인터넷, 다중
무선 인터페이스 네트워크

김 원 태 (Wontae Kim)



2013년 2월 : 중앙대학교 전기
전자공학부 학사
2013년~현재 : 고려대학교 전기
전자공학과 석사과정
<관심분야> 미래 인터넷, 네트
워크 가상화

주 석 진 (Sukjin Choo)



2013년 8월 : 고려대학교 전기
전자전파공학부 학사
2013년~현재 : 고려대학교 전기
전자공학과 석·박사과정
<관심분야> 미래 인터넷, 모바일
데이터 오프로딩

김 명 수 (Myeongsu Kim)



2014년 2월 : 한성대학교 정보
통신공학과 학사
2014년~현재 : 고려대학교 전기
전자공학과 석·박사과정
<관심분야> 미래 인터넷

백 상 현 (Sangheon Pack)



2000년 2월 : 서울대학교 컴퓨
터공학부 학사
2005년 2월 : 서울대학교 전기
컴퓨터공학부 박사
2007년~현재 : 고려대학교 전기
전자전파공학과 부교수
<관심분야> 미래 인터넷, 무선
이동 네트워크

강 철 희 (Chul-Hee Kang)



1975년 2월 : 일본 와세다대학교
전자공학과 학사
1977년 2월 : 일본 와세다대학교
전자공학과 석사
1980년 2월 : 일본 와세다대학교
전자공학과 박사
1980년~1994년 : 한국전자통신
연구원(ETRI) 연구원

1990년~1994년 : 한국전자통신연구원(ETRI) 부원장
1994년 : 미국 워싱턴대학 객원교수
1995년~2011년 : 고려대학교 전자전기전자공학과 교수
<관심분야> 미래 인터넷