논문 15-40-06-11 The Journal of Korean Institute of Communications and Information Sciences '15-06 Vol.40 No.06
http://dx.doi.org/10.7840/kics.2015.40.6.1070

Science DMZ 적용을 위한 SDN 기반의 네트워크 침입 방지 시스템

조 진 용◆, 장 희 진*, 이 경 민*, 공 정 욱*

SDN-Based Intrusion Prevention System for Science DMZ

Jinyong Jo◆, Heejin Jang*, Kyungmin Lee*, JongUk Kong*

요 약

본 논문은 Science DMZ(Demilitarized Zone) 적용을 위한 SDN(Software Defined Networking) 기반의 네트워크 침입 방지 시스템을 소개한다. 제안된 시스템은 침입 탐지 기능을 침입 방지 장치로부터 분리하고 SDN 기술을 확장해 탐지 기능과 방어 기능을 상호 연동시킴으로써 네트워크 보안 장비의 유연성(flexibility)과 확장성(extensibility)을 높이고 패킷 검사(packet inspection) 등으로 야기되는 패킷 손실을 방지하는데 목적이 있다. 본 논문에서는 제안한 프레임워크의 한 응용 시나리오를 소개하고 국가과학기술연구망에 구축된 네트워킹 DMZ환경에 시험 적용함으로써 활용 가능성을 검증한다.

Key Words : Software defined networking, science DMZ, intrusion detection and prevention

ABSTRACT

In this paper, we introduce an SDN-based intrusion prevention system for more secure Science DMZ with no performance limits. The proposed system is structured with intrusion-prevention, intrusion-detection, and prevention-decision subsystems which are physically distributed but informationally connected by an SDN interface. The functional distribution and the application of SDN technology increase the flexibility and extensibility of the proposed system and prevent performance degradation possibly caused by network security equipments on Science DMZ. We verified the feasibility and performance of the proposed system over a testbed set up at KREONET.

Ⅰ. 서 론

인터넷 백본 대역폭의 폭발적 증가에도 불구하고, 복잡도가 증가된 일반 목적 망(general-purpose network)은 데이터 전송성능의 병목 지점으로 작용하고 있다. 특히, 상태기반 방화벽(stateful firewall)으로 인한 사용자 망(last-mile network)의 병목은 패킷손실을 유발해 대용량 과학기술데이터(scientific data)의 전송 성능을 크게 저하시킨다[1].

방화벽 등으로 인한 사용자 망의 성능병목 문제를 완화하고 대용량 과학기술데이터의 고속 전송을 담보하기 위해 Science DMZ와 같은 네트워크 완충 영역에 대한 연구가 진행되어 왔다[1]. 특히 근래에는 DMZ 내·외부 네트워킹 자원을 유연하게 연동하기 위해

※ 본 논문은 한국과학기술정보연구원 '첨단연구망 기반 협업플랫폼 서비스 기술 개발 및 적용(K-15-L01-C04-S03)' 과제의 지원을 받아 수행되었음
◆ First Author : Korea Institute of Science and Technology Information, jiny92@kisti.re.kr, 정회원
* Korea Institute of Science and Technology Information, {jhj, tsoc, kju}@kisti.re.kr
논문번호 : KICS2015-04-113, Received April 6, 2015; Revised June 12, 2015; Accepted June 12, 2015

SDN(Software Defined Networking[2]) 기술을 DMZ 환경에 적용하는 방안이 모색되고 있다[3-5]. 하지만, SDN의 적용 범위가 가상회선(virtual circuit)의 설정 등 네트워킹 유연성 확보에 초점을 두고 있는 단계로써 DMZ 환경의 보안 강화 방안에 대해서는 고려되지 않고 있다.

본 논문은 DMZ 환경에서 유연한 보안 적용을 가능하게 하는 SDN 기반의 침입 방지 시스템, SAFE(SDN-enabled science-DMZ gAteway to realizae light-weight FirEwall)를 제안한다. 제안된 시스템은 침입 방지 장치(IPS, Intrusion Prevention System)의 침입 탐지, 방지 결정, 방지 시행 등의 기능을 물리적으로 분산하고 SDN을 이용해 상호 연동시키는 것을 특징으로 한다. 해당 기능들을 분산시키고 SDN을 통해 연동함으로써 1) 상태기반 방화벽(stateful firewall) 사용으로 인한 성능병목을 해소하고, 2) 침입 방지 시스템의 구축비용 절감 및 확장성(extensibility) 향상을 기대할 수 있으며, 3) 보안시스템의 빈번한 설정 변경으로 야기되는 운영비용(operational cost)을 절감하는 효과가 있다.

본 논문이 기여하는 바는 다음과 같다. 첫째, DMZ 환경 또는 소규모 네트워크 환경에서 오픈소스 침입 탐지시스템들과 연동해 활용 가능한 침입 방지 시스템을 소개했다. 둘째, 확장성과 유연성 향상을 고려해 침입 방지 시스템의 구성 요소 및 구조를 설계하고 구성 요소들 간 연동을 위해 SDN 기술을 적용했다. 마지막으로, 침입 방지를 위한 응용 시나리오를 시제품으로 구현하고 국가과학기술연구망에 테스트베드를 구성해 제안된 시스템의 DMZ 환경에의 적용 가능성을 검증했다.

본 논문의 2장에서는 Science DMZ와 SDN 기술에 대해 살펴보고 DMZ 환경에서 침입 방지 시스템이 갖춰야 할 기능적 요구사항을 설명한다. 3장에서는 제안된 시스템의 구성요소 및 구조에 대한 세부 설계 내용을 기술하고 관련된 연구들을 소개한다. 4장에서는 응용 시나리오를 예시하고 응용의 구현사항에 대해 기술하며 5장에서 제안된 시스템의 적용 가능성을 검증한다. 마지막으로, 6장에서 결론을 맺는다.

Ⅱ. 배경 기술 및 관련 연구

본 절에서는 제안한 시스템의 적용 환경과 기반 기술인 Science DMZ[4]와 SDN[2]에 대해서 살펴본다. 본 논문에서 Science DMZ의 적용 범위는 대용량 과학기술데이터의 고속 전송에 국한되지 않으며 방화벽으로 인해 서비스가 제약될 수 있는 다양한 온라인 협업 환경[8]을 포함한다.

2.1 Science DMZ

Science DMZ는 독자적인 전송 장비, 네트워크 구성 및 보안 정책을 갖는 일종의 부분망으로써 기업 등의 사설망(intranet)과 인터넷 망의 중간 지역에 구축되어 데이터 전송 성능을 가속시키는 역할을 한다. 대용량 과학기술데이터의 전송 속도를 향상시키기 위해 상태기반 방화벽의 사용이 지양되며 광역 데이터전송(wide area data transfer) 시 전송 성능 가속을 위해 전용 데이터전송노드(DTN, Data Transfer Node[6])를 이용한다.

선진 연구교육망(NREN, National Research and Education Network)의 경우 대역폭이 충분한 광역 통신망(WAN) 보다는 방화벽 등으로 병목이 발생하는 근거리 통신망(LAN)에서 IP 패킷의 손실 가능성이 커진다[1]. WAN과 LAN의 접점에 DTN을 설치하고 전송 구간을 분리해 데이터를 수신 받는다면 LAN 구간에서 발생하는 패킷 손실이 WAN 구간에 영향을 주지 않으므로 TCP 성능의 간접적 향상을 기대할 수 있다.

2.2 Software Defined Networking

SDN[2]은 네트워크 제어와 패킷 포워딩 기능을 분리하고 오픈플로우[7] 등 개방형 API(Application Programming Interface)를 통해 제어 기능과 포워딩 기능을 상호 연동시키는 것을 특징으로 갖는 네트워킹 접근방식이다. 중앙 집중화된 컨트롤러에서 패킷 플로우에 대한 제어(control)와 네트워크 인프라에 대한 환경 설정(configuration)이 가능하기 때문에 네트워크 상황 변화에 민첩히 대응할 수 있다. 또한, 네트워크 자원에 대한 설정, 관리, 보호, 최적화 등의 프로세스들을 SDN 프로그램화함으로 네트워크 운영 업무를 자동화할 수 있다.

SDN에서 패킷 플로우에 대한 출력 포트 검색 절차는 다음과 같다. 패킷이 도착하면 네트워크 스위치는 로컬 플로우 테이블에서 플로우 엔트리를 조회하고 지정된 처리지침(action)에 따라 패킷을 처리한다. 플로우 엔트리가 존재하지 않으면 수신된 패킷을 캡슐화한 후 컨트롤러에게 전달함으로써 일종의 플로우 테이블 조회(flow table lookup)과정을 수행한다. 컨트롤러의 SDN 프로그램은 전달받은 패킷의 처리지침을 결정한 후 개방형 API를 통해 네트워크 스위치에 해당 플로우에 대한 플로우 엔트리를 설치한다.

2.3 관련 연구

우리가 아는 한, 본 논문은 Science DMZ 환경에 적용 가능한 최초의 SDN 기반 침입 방지 시스템이다. DMZ 환경과 직접적 관련성은 낮지만 네트워크 보안 강화 및 보안정책 적용의 유연성 향상을 위해 SDN 기술이 적용된 연구들을 중심으로 내용을 살펴본다.

Resonance[10]는 캠퍼스 망 자원에 대한 유연한 접근제어 방안을 소개했다. 오픈플로우 네트워크와 SDN 기술을 이용하여 캠퍼스 망 자원에 대한 접속 인증을 강화하기 위해 설계되었다. 사전 수락 제어(pre-admission control)의 일환으로 모든 호스트들은 웹 기반 접속인증 시스템으로부터 접속 인증을 받는다. 또한, 보안 모니터링 시스템은 사후 수락(post-admission)의 목적으로 호스트들의 보안 허점을 상시적으로 조사한다. 인증의 성공 여부와 모니터링 결과는 SDN 제어기에게 전달되어 동적 접근제어를 수행하는데 활용된다.

SIMPLE[11]은 미들박스들 간 소스라우팅 정책 시행(policy enforcement)을 자동화하기 위해 SDN 기술을 활용했다. 방화벽, 프록시, 침입탐지시스템 등 다양한 미들박스들을 상호 연결시켜 논리적 체인(chain)을 구성함으로써 보안의 강화 및 전송 성능의 향상을 도모할 수 있다. 기존 미들박스들의 기능변경이 요구되지 않으며 SDN 스위치들만을 제어해 소스라우팅 정책을 실현한다. 정책 시행의 자동화는 체인 구성을 위해 필요한 네트워크 설정 비용과 관리·운영의 복잡도를 낮추는 장점이 있다.

Flowguard[12]는 SDN 기술의 세부 특성들로부터 초래될 수 있는 잠재적 보안 위협에 체계적으로 대처하기 위해 설계된 SDN 방화벽 프로그램이다. SDN이 적용된 내부 네트워크에서 방화벽 정책의 위반 사항을 탐지·해결하는 점에서 기존 인터넷 방화벽과 차이가 있다. SDN 네트워크의 빈번한 상태변화로 인해 발생하는 방화벽 정책과 플로우 정책 간의 비일관성 문제, 오픈플로우 프로토콜의 SET_FIELDS 처리규칙을 악용한 방화벽 회피 가능성 등에 대해 SDN 컨트롤러 측면의 프레임워크를 제시하고 있다.

본 논문은 기존 미들박스들을 활용하고 SDN 기술을 적용했다는 점에서 SIMPLE과 유사점을 갖는다. 하지만, 네트워크 인프라가 SDN의 적용 대상인 SIMPLE에 비해 SAFE는 미들박스들을 대상으로 하기 때문에 관리·운영이 상대적으로 용이하고 네트워크 인프라에 대한 재구축 비용이 필요치 않기 때문에 Science DMZ 환경에 효과적으로 적용할 수 있다. 또한, SIMPLE은 SDN이 적용된 네트워크에서 일반 목적 응용(예, e-mail, 웹 브라우징 등)의 지원을 위해 설계되었기 때문에 종단 응용의 전송 성능에 대해서는 고려되지 않았다. SAFE는 SDN 기술을 이용해 시제품을 구현하고 Science DMZ 환경에 적용해 실증 분석 결과를 제시하는 최초의 논문으로써 보안 관리의 유연성(flexibility)과 확장성을 높을 높이며 패킷 검사 등으로 야기되는 패킷 손실을 방지하는데 효과적이다.

Ⅲ. 침입 방지 시스템 개요

본 절은 제안한 침입 방지 시스템의 고려 사항, 구성 요소 및 구조를 살펴본다.

3.1 침입 방지 시스템의 기능 고려 사항

Science DMZ의 활용 목적 등을 고려해 볼 때 DMZ 내 침입 방지 시스템이 가져야 할 특성은 다음과 같다.

- 데이터 처리성능 및 저비용 구조: DMZ 내·외부 간 데이터 전송 속도와 침입 방지 시스템의 구축비용 간에는 트레이드오프 관계가 존재한다. 고성능 침입 방지 시스템의 적용을 통해 전송 속도 저하와 관련된 문제를 일부 해결할 수 있지만 관련 장비가 고가인 단점이 있다. DMZ 환경에 적용되는 침입 방지 시스템은 데이터 전송 속도의 저하를 초래하지 않아야 하며 시스템 확장성(extensibility)의 확보를 통해 초기 구축비용을 줄일 수 있어야 한다.
- 정책규칙(policy rule) 설정의 유연성 및 관리의 용이성: 방화벽 사용으로 인한 추가적인 문제는 네트워크 자원에 대한 접근 제어가 데이터 통신의 단절을 야기할 수 있다는 점이다. 예를 들어, 방화벽에서 접근제어 규칙을 정적(static)으로 적용하면 임의의 포트번호를 사용하는 UDP 응용의 통신 단절을 야기[9]한다. 다수 응용의 통신 단절문제를 해결하기 위해서는 방화벽 설정의 빈번한 변경이 요구되지만 이는 시스템 관리비용의 증가를 초래한다. 침입 탐지 시스템은 관리·운영 측면의 유연성 확보를 통해 운영비용을 줄일 수 있어야 한다.

3.2 침입 방지 시스템 개요

SAFE는 Science DMZ 환경에서 데이터전송노드들의 전송성능을 저하시키지 않고 네트워크 보안을 강화하는데 목적이 있다. 보안 강화를 위해 상태기반 방화벽을 적용할 경우 패킷 손실로 인해 데이터전송노드의 파일전송 성능에 부정적인 영향을 줄 수 있다.

SAFE의 구성 요소는 그림 1과 같이 SDN이 적용

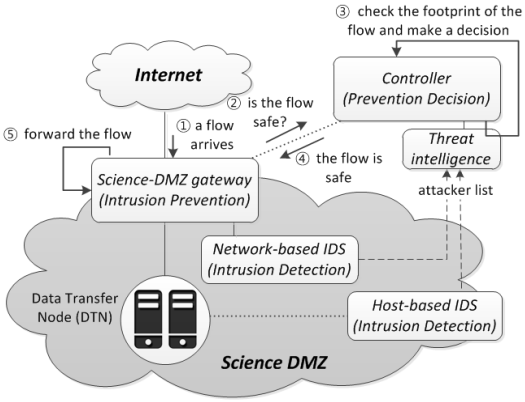

그림 1. 시스템 구성 요소 및 침입 방지 워크플로우
Fig. 1. System components and prevention logic

된 네트워크 장치(Science DMZ gateway 또는 Middlebox), 컨트롤러(Controller), 침입 탐지 시스템(IDS, Intrusion Detection System)을 포함한다. IDS는 상시적으로 침입을 탐지하고 컨트롤러에게 침입자 정보를 제공한다. 컨트롤러는 제공된 침입자 정보를 위협정보(Threat intelligence) 데이터베이스에 저장하고 Science DMZ에 대한 패킷 플로우의 접근제어에 활용한다. IDS의 특성 상, IP 패킷의 헤더 필드값이 변경되어야 하거나 네트워크 기반 IDS로 동작되어야 할 경우 게이트웨이의 이더넷 포트에 직접 연결될 수 있다.

SAFE에서 침입 방지를 위한 패킷처리 절차는 다음과 같다. 게이트웨이는 처리지침이 설치되어 있지 않은 IP 패킷이 도착하면(①) Science DMZ에 대한 접근제어를 위해 컨트롤러에게 플로우 테이블 조회를 요청한다(②). 컨트롤러는 설정된 보안 정책과 위협정보 데이터베이스를 이용해 접근 허락 또는 거부를 결정하고 결과를 플로우 엔트리에 반영한 후 게이트웨이에 설치한다(③, ④). 게이트웨이는 해당 플로우에 속한 IP 패킷들이 도착하면 설치된 플로우 엔트리의 처리지침에 따라 패킷을 처리한다(⑤).

지금부터는 앞서 기술한 Science DMZ 환경에서 요구되는 침입 방지 시스템의 기능 고려 사항에 맞춰 SAFE의 개략적 내용을 살펴본다.

탐지, 결정, 방지 기능의 분리 및 SDN을 통한 연동

SAFE는 침입 탐지, 방지 결정, 방지 시행 등의 방화벽 기능을 물리적으로 분산시킨 후 SDN 기술을 활용해 상호 연동시키는 침입 방지 시스템이다. 비록 SAFE가 네트워크 기반인프라(스위치나 라우터들의 집합)로 볼 수는 없지만 SDN 기술을 적용함으로써 개별 네트워크 장비에 대한 통일된 제어 및 설정이 가능해지는 장점이 있다. 추가적으로 다양한 미들박스 기능들 간에 논리적 체인을 구성해야 할 경우[11] SDN 적용을 통해 미들박스 네트워킹의 유연성과 관리의 용이성을 확보할 수 있다.

SAFE 환경에서는 방지 결정 기능 등을 컨트롤러에게 이관하고 탐지 기능을 분리함으로써 게이트웨이의 기능을 간소화할 수 있다. IPBox의 기능 간소화는 장비 가격을 낮추고 성능병목 문제를 해결하는데 효과적이다. 또한 중앙 컨트롤러에 의한 방지 결정을 통해 탐지된 보안 위협에 대한 신속하고 동적인 대처가 가능하다. 탐지된 공격자 정보의 유지관리를 중앙 집중화하고 유관 컨트롤러, IPS, 방화벽 등과 공격자 정보를 공유함으로써 DDoS(Distributed Denial-of-Service[17]) 등 분산서비스 공격 등에 효과적으로 대응할 수 있다.

SAFE는 탐지 기능을 분리시킴으로써 DPI(Deep Packet Inspection) 등 복잡한 탐지 기능 수행으로 야기되는 게이트웨이 또는 컨트롤러의 성능 저하 문제를 해결한다. 탐지 기능이 IP 패킷의 전송 경로 외부에 존재하는 별개의 시스템으로 분리됨으로써 침입 탐지 중 패킷 손실이 발생해도 데이터전송노드의 파일전송 성능에 영향을 주지 않는 장점이 있다. 또한, Snort[14], OSSEC[15], Modsecurity[16] 등 오픈소스 IDS/방화벽들과의 연계가 용이해지기 때문에 초기 구축비용을 절감하고 시스템 확장성을 높이는 장점이 있다. 확장성 측면에서, 그림 2와 같이 호스트 IDS(HIDS), 네트워크 IDS(NIDS), 또는 혼합 IDS(Hybrid IDS) 등 다양한 형태의 IDS를 조합해 침입 탐지에 활용 가능하므로 각 IDS가 갖는 단점[13]을 상쇄하는데 효과가 있다.

모니터링 세션의 분리 및 시스템 통합의 지원

게이트웨이는 SDN의 개방형 API인 FORWARD

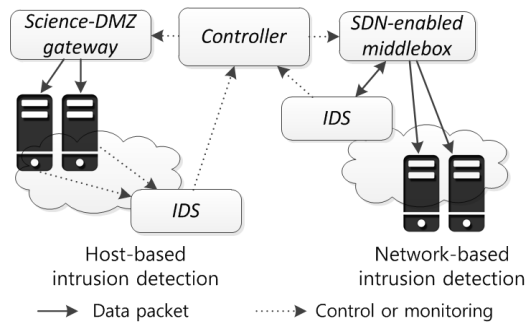

그림 2. 시스템 구성 요소의 연동 예시
Fig. 2. Exemplification of interlocking system components

와 DROP 명령을 이용해 Science DMZ에 대한 접근 제어를 수행한다. IP 헤더 필드값의 변경을 요구하는 IDS와 연동될 경우, 게이트웨이는 SET_FIELDS 명령을 이용해 네트워크 주소와 포트 주소를 변경한다. SET_FIELDS는 NAT(Network Address Translation) 기능 및 소스라우팅 정책의 실현에 이용 가능하다. 추가적으로 IP 계층에서 패킷 무결성(integrity)을 고려한 소스라우팅 정책의 실현을 위해 SAFE는 SET_TUNNEL 명령을 지원한다.

그림 3은 SET_FIELDS, FORWARD 및 DROP 명령을 이용해 오픈소스 IDS/방화벽과 제안된 Science DMZ 게이트웨이를 연동하는 한 방법을 예시한다. SDN의 프로그램가능성(programmability)은 예시된 방법 이외의 다양한 설정을 가능하게 한다. SAFE는 패킷의 데이터 경로(data path)와 모니터링 경로(monitoring path)를 물리적으로 분리하고 기존 투명(transparent) 또는 불투명 방식의 IDS/방화벽[14],[16]를 모니터링 경로에 연동시킴으로써 침입 탐지 시 패킷 손실이 발생해도 데이터전송노드의 파일전송 성능에는 영향을 주지 않는다. 모니터링 경로는 공격자 탐지를 위해 사용된다.

모니터링 경로에서 IP 헤더 필드값 등의 변경을 통해 기존 오픈소스 IDS/방화벽들의 운용환경을 모사할 수 있기 때문에 해당 장비들의 소프트웨어를 수정하거나 종단 호스트·서버의 네트워크 설정을 변경할 필요가 없다. 예를 들어, Modsecurity[16]와 같은 웹 방화벽은 역프록시(reverse proxy) 방식으로 구성되므로 서버 측에서 프록시 환경을 설정해야 하는 불편함이 있다. 하지만 SAFE를 적용하면 SET_FIELDS 명령을 통해 IP 패킷의 주소를 변경할 수 있기 때문에 서버의 설정변경 없이 역프록시 방식의 운용환경을 모사할 수 있다. 결론적으로, 수신되는 패킷을 모니터링 경로와 데이터 경로로 동시에 FORWARD하고 모니터링 경로로 포워딩된 패킷에 대해 추가적으로 SET_FIELDS 명령을 적용함으로써 모니터링 경로와 데이터 경로를 분리할 수 있다.

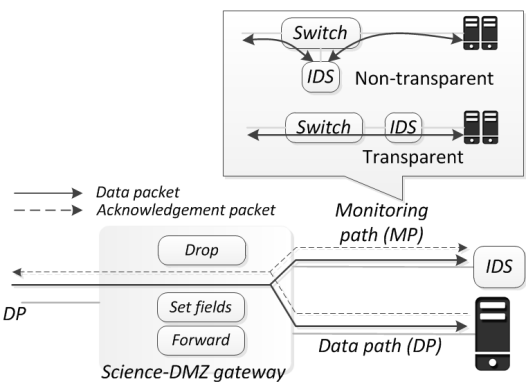

그림 3. 네트워크 IDS와 연동을 위해 제안된 Science-DMZ 게이트웨이의 기능
Fig. 3. Functions of proposed Science-DMZ gateway to accommodate network IDSs

플로우의 국지적 처리(Localization)

SDN에서 제어평면의 분리는 HOF(Head-Of-Flow delay) 문제와 플로우 테이블 조회 병목(lookup bottleneck)을 유발한다. SDN 스위치는 플로우에 속한 최초 패킷에 대해 플로우 테이블 조회를 수행하게 되는데 SDN 메시지의 왕복시간 및 처리시간 등으로 인해 HOF 문제가 발생한다. 조회 병목은 네트워크 스위치에서 발생하는 요청 병목과 컨트롤러의 SDN 메시지 처리 병목으로 구분할 수 있는데 요청 병목이 플로우 처리율에 미치는 영향이 큰 것으로 알려져 있다[19]. HOF와 조회 병목은 TCP의 RTO(Retransmission TimeOut)에 부정적인 영향을 주기 때문에 TCP 성능 저하를 초래할 수 있다.

SAFE는 인가된 패킷의 시그니처(signature)[1]를 게이트웨이에 미리 저장한 후 해당 패킷의 처리를 컨트롤러로부터 게이트웨이에게 위임함으로써 HOF 문제와 조회 병목을 완화시킨다. 시그니처 기반의 패킷 처리 국지화를 위해 블룸 필터(Bloom filter[20])를 적용했다. 블룸 필터는 위양성(false positive)과 공간-효율성 간에 상반관계를 갖는 확률적 데이터 구조이다. 위양성 확률은 식 (1)과 같이 계산된다.

$$p_f = (1 - e^{-kn/m})^k \quad (1)$$

여기서, 위양성 확률 p_f는 해쉬 함수의 수 k, 블룸 필터의 비트 수 m, 시그니처를 남긴 요소의 수 n의 관계식으로 표현된다. 일반적으로 위양성 확률 0.01에 대해 개별 요소 1개를 저장하는데 10 비트 이하의 저장 공간이 요구된다[21].

본 논문에서는 블룸 필터를 이용한 정합 방법을 시그니처 정합으로 정의한다. SAFE에서 정합의 우선순위는 완전일치 정합(exact match), 시그니처 정합, 와일드카드 정합(wildcard match)의 순이다. 즉, 패킷이 수신되면 게이트웨이는 각 정합 테이블의 우선 순위에 따라 플로우 엔트리를 검색하고 할당된 처리지침

1) 본 논문에서 시그니처는 IP 주소와 포트 정보 등을 포함하는 해쉬 기반의 비트 배열 값이다. 패킷의 텍스트 탐지 패턴을 의미하지는 않는다.

에 따라 패킷을 처리한다.

시그니처 정합을 지원함으로써 정합 규칙 적용의 유연성을 높일 수 있고 하드웨어 자원을 효율적으로 활용할 수 있다. 예를 들어, 와일드카드 정합 또는 완전일치 정합에 이용되는 플로우 엔트리를 패킷의 도착 전에 미리 할당(proactive allocation)하기 위해서는 값비싼 TCAM(Ternary CAM)을 이용하거나 정합규칙(rule, 예를 들어 TCP/IP 5-tuple)을 미리 알아야 하는 문제가 있다.

Ⅳ. 침입 방지 시스템 구현 및 성능 평가

본 절에서는 SAFE 구성요소의 세부 기능과 구현 내용을 살펴보고 호스트 기반 IDS를 SAFE에 적용한 성능 평가 결과를 소개한다.

4.1 침입 방지 시스템의 세부 기능 및 구현

본 절에서 소개하는 SAFE 응용 시나리오는 호스트기반 IDS가 제공하는 위협 정보를 이용하고 Science DMZ의 경계선(perimeter)에서 블랙리스트 기반의 침입 방지를 수행하는 특징을 갖는다. 침입탐지 기능을 경계선에서 분리하고 전용 하드웨어를 이용해 침입 방지를 수행함으로써 호스트 기반 방화벽(예를 들어, *iptables*)이 갖는 성능병목 문제[18]를 해결할 수 있다.

SAFE는 그림 4와 같이 IPBox, IPController, IPAgent로 구성된다. IPBox는 SDN이 적용된 Science DMZ 게이트웨이 또는 미들박스이고 IPController는 SAFE의 제어평면으로써 컨트롤러에 해당된다. IPAgent는 IDS에 설치되는 소프트웨어 에이전트이다. SAFE에 적용된 SDN 구조가 오픈플로우 기반 SDN과 유사하므로 SDN 기술과 관련된 구체적 설명은 생략하고 구성요소들의 주요 기능들을 중심으로 기술한다.

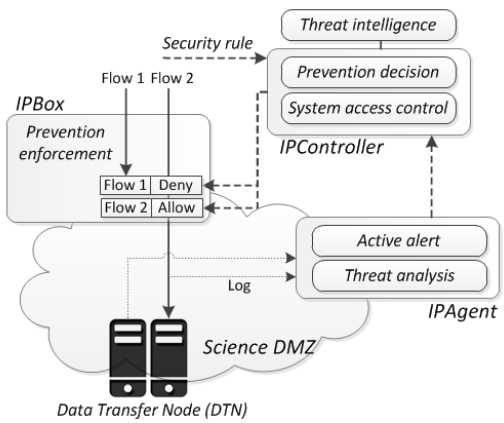

그림 4. SAFE 구성요소 별 기능과 호스트 기반 IDS의 연동
Fig. 4. Features of SAFE components and the application of a host-based IDS

IPAgent가 설치된 IDS는 *rsyslog* 등을 이용해 DMZ 내부 서버들로부터 로그 정보를 취합하고 IDS의 보안 정책에 따라 보안 위협을 분류한다. 수집된 시스템 로그의 분석을 위해 호스트 기반 IDS인 *logwatch*와 *OSSEC*(Open Source SECurity)이 이용되었다. IPAgent는 보안 위협을 단계별로 재분류하고 표 1과 같은 XML 기반의 E2N(End to Network) 메시지에 위협 정보를 기록한 후 컨트롤러에게 전달한다.

IPController는 IPAgent가 보고한 보안 위협 정보와 SAFE 관리자가 입력한 보안 규정(security rule)에 따라 침입 방지를 결정하고 위협정보 데이터베이스를 관리한다. 비인가 IPBox나 IPAgent의 컨트롤러 접근을 차단하기 위해 IP 기반의 접근 제어를 수행한다. 특정 호스트가 DMZ 내 데이터전송노드들을 연쇄적으로 공격할 경우, 보안 위협 정보(예를 들어, 공격자의 IP 주소)를 공유함으로써 다수의 서버에 대한 공격을 사전에 방지할 수 있다.

IPBox의 데이터평면은 성능 가속을 위해 NetFPGA[24] 하드웨어 상에 구현되었다. 데이터평면의 초기 설계 구조[25]에 대한 설명은 생략하고 IPBox에서 출력 포트를 자체적으로 선택하기 위해 추가 구현된 기능들을 중심으로 살펴본다.

그림 5는 SAFE에 적용된 플로우 엔트리의 처리지침을 보여준다. cmd는 FORWARD, DROP 등 패킷 플로우의 처리 방법을 정의한다. f_opt는 SET_TUNNEL 등의 명령에서 패킷이 송신자에게 루핑(looping)되는 것을 막을 목적으로 이용된다. MAC, IP, TCP/UDP 주소 중 변경되어야 할 헤더 필드를 다중 선택하기 위해 t_opt가 이용된다. IPBox는 자체적으로 유지하는 MAC/ARP 테이블의 주소 정보를 활용해 MAC 주소를 자동 설정한다. TCP/UDP 및 IP 주소의 변경과 전송계층에서의 1:1 전송을 위해 flow

표 1. E2N 메시지 예시
Table 1. Example of an E2N message

```
<interface ver="1.0">
<summary><subject>security alert</subject>
<reporter>OSSEC</reporter><address>1.1.1.1</address>
<version>1.0</version><length>1</length></summary>
<threat><category>syslog</category><attacker>2.2.2.2
</attacker><detector>123</detector><level>10</level>
...</threat>
```

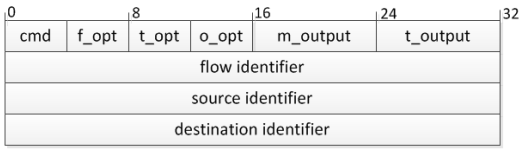

그림 5. 플로우 엔트리의 처리지침 필드
Fig. 5. Action fields of flow entry

identifier, source identifier, destination identifier가 이용되며, 1:N 전송의 경우 위상 테이블(topology table)을 이용해 목적지 주소를 설정한다. o_opt 필드는 패킷의 출력 포트 선택 방법을 정의한다. MAC/ARP 테이블을 이용해 자동 선택하거나 m_output 또는 t_output 필드를 이용해 출력 포트를 수동 선택할 수 있다. IPBox가 주소 값 및 출력 포트 등을 자체적으로 설정함으로써 IPController가 유지해야 하는 IPBox에 대한 상태 정보를 줄일 수 있으며 플로우 엔트리의 설정 과정을 간소화할 수 있다.

IPBox는 지금까지 설명한 플로우 엔트리의 처리지침에 따라 방지 시행(prevention enforcement)을 실행한다. 패킷이 도착하면 IPBox는 MAC 테이블을 참조해 이더넷 포트로 출력시킨 후 IPController에게 플로우 테이블 조회를 요청함으로써 HOF 문제를 완화한다. SET_FIELDS나 SET_TUNNEL 명령을 이용해 위협 트래픽을 소스라우팅하면 Science DMZ 외부에 설치된 다양한 보안 분석 자원들을 활용할 수 있을 것으로 기대한다.

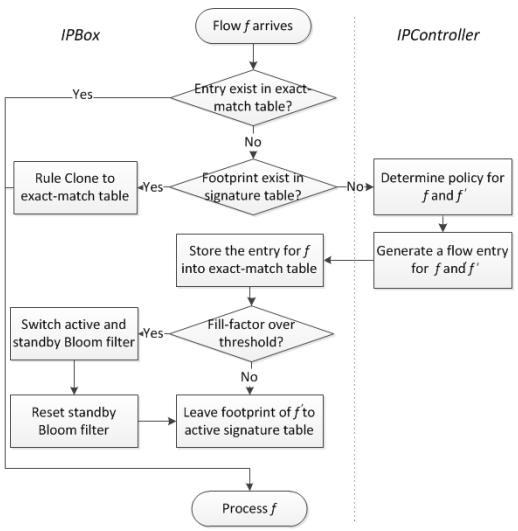

그림 6. 국지적 플로우 처리를 위한 블룸 필터의 적용
Fig. 6. Application of Bloom filter to localize flow lookup process

그림 6은 패킷 플로우의 국지적 처리를 위해 블룸 필터와 규칙 복사(rule clone[26])를 이용하는 절차를 보여준다. 수신된 패킷이 IPBox의 시스템 소프트웨어에서 시그니처 정합될 경우, IPBox는 해당 패킷에 대한 플로우 엔트리를 생성하고 NetFPGA 하드웨어의 완전일치 정합 테이블에 저장한다. 이 과정을 규칙 복사로 정의한다. 상대적으로 복잡한 정합 과정을 시스템 소프트웨어가 담당하고 저가의 하드웨어 메모리에 플로우 엔트리를 복사해 패킷을 처리하게 함으로써 정합 방법의 적용 유연성을 높이고 메모리 자원을 효율적으로 사용할 수 있다.

앞서 기술했듯이 플로우의 국지적 처리는 HOF 문제와 조회 병목 문제를 완화시킨다. 시그니처 정합을 위한 플로우 엔트리는 2개의 블룸 필터를 이용해 구성한다. 시그니처 정합 엔트리(signature-match entry)의 처리지침은 IPController에 의해 결정된다. 위양성 문제를 해결하기 위해 2개의 블룸 필터를 교대로 이용한다. 위양성 확률 p_f를 α이하로 유지하기 위해 블룸 필터에 n개 이하의 플로우 시그니처만 남기는 방식이다.

$$t_r = \frac{-m(\ln 2)^2}{\ln \alpha} \leq n \tag{2}$$

$$\beta \cdot t_r < n < t_r \quad (0 < \beta < 1) \tag{3}$$

식 (2)는 식 (1)을 간소화해 재정리한 것으로 블룸 필터의 교체 조건을 의미한다. 즉 활성 블룸 필터에 t_r개 이상의 플로우가 기록되면 해당 필터를 초기화한 후 대기 상태로 전환하고 대기 상태였던 블룸 필터를 활성 블룸 필터로 교체한다. 필터 초기화는 아직 규칙 복사가 수행되지 않은 플로우의 시그니처 정보를 삭제하는 문제가 있다. 플로우 시그니처의 수 n이 식 (3)을 만족하면 활성 및 대기 블룸 필터 모두에 시그니처를 남김으로써 필터 초기화로 인한 문제를 완화한다. 기대 위양성 확률은 $\alpha = 0.001$로 계수는 $\beta \approx 1$로 각각 설정했다.

플로우 f가 입력되면 IPBox는 IPController에게 테이블 조회를 요청한다. IPController는 플로우 f에 대한 완전일치 정합 엔트리를 SDN 메시지를 통해 전달한다. 피드백 플로우 f'을 IPBox의 시그니처 정합 테이블에 등록할 것인지를 결정하고 SDN 메시지에 피기백(piggyback)해 함께 전달한다. 피드백 플로우 f'이 입력되면 시그니처 정합이 이뤄지고 시그니처 정합 엔트리에서 완전일치 정합 엔트리로 규칙 복사

가 수행된다. 규칙 복사를 통해 제한된 하드웨어 자원을 효율적으로 사용할 수 있다.

하나의 통신 세션이 2개의 플로우(예를 들면, 데이터 플로우와 피드백 플로우)로 구성된다고 가정한다. 블룸 필터와 규칙 복사를 적용하면 N개의 통신 세션에 대해 테이블 조회 횟수가 $2N$에서 N으로 50% 줄고 SDN 메시지의 발생도 $4N$에서 $2N$으로 감소하기 때문에 조회 병목 문제와 메시지 부하를 줄일 수 있다. 한 번의 테이블 조회는 SDN 메시지를 2번(요청과 응답) 발생시킨다.

SDN 환경에서 국지적인 플로우 처리 특히 블룸 필터의 사용은 네트워킹 상태에 대한 일관성(consistency[27]) 문제를 야기할 수 있다. 소개한 SAFE 응용 시나리오에서는 플로우의 타임아웃 시간에 국지적으로 엔트리를 삭제하는 등 약한 일관성 모델(weak consistency model)을 적용했다.

먼저, 블룸 필터가 시그니처 삭제를 지원하지 않기 때문에 플로우의 상태 변화가 요구될 때(예를 들면, 접근 허락된 플로우가 접근 거부로 상태 변화되어야 할 때) 필터에서 시그니처를 삭제할 수 없는 문제가 발생한다. SAFE는 블룸 필터에서 플로우 f의 시그니처를 삭제하는 대신 규칙 복사되어 완전일치 정합 테이블 등에 기록된 플로우 f 및 f'의 처리규칙을 변경함으로써 해당 문제를 해결한다. 보다 강한(strong) 일관성 모델을 적용하기 위해 집계 블룸 필터(Counting Bloom filter[28])의 사용을 고려할 수 있다.

둘째, SAFE는 필터 충전율(fill factor)에 따라 블룸 필터를 초기화시킴으로써 위양성 문제를 완화한다. 블룸 필터가 IPBox 내에서 국지적으로 초기화되기 때문에 IPController와 IPBox의 플로우 테이블에 정보의 일관성 문제가 발생할 수 있다. 블품 필터에 의해 시그니처 정합된 플로우는 규칙 복사되어 최종적으로 완전일치 테이블에 저장된다. SAFE는 완전일치 테이블에 저장된 플로우 엔트리 중 접근 허가된 플로우에 대해서 타임아웃(timeout)시키는 방법으로 일관성 문제를 완화했다.

4.2 성능 평가

소개한 SAFE 시나리오의 타당성을 검증하기 위해 그림 4와 같은 환경에서 성능 평가를 수행했다. IPBox의 프로토타입은 1Gbps 회선 속도에 준하는 패킷 처리 성능을 제공[25]하며 침입탐지 기능을 분리했기 때문에 DPI 등으로 인한 성능저하가 발생하지 않는다. 따라서 본 논문은 IPBox의 패킷 처리 성능에 대한 검증을 생략한다. 동일한 서브넷에 구축된 총 7대의 희생자(victim) 서버가 *OSSEC* 서버에게 로그 정보를 전달하면 IPAgent는 무작위 대입 공격(brute force attack)을 시도한 공격자 IP 주소를 추출해 IPController에게 전달한다.

공격자 정보는 약 50일 간 수집되었다. 수집 기간 동안 관측된 서버 별 총 공격자(IP 주소) 수는 평균 158개였다. 공격자 수는 공격의 횟수 또는 공격 성공의 횟수를 의미하지 않는다.

그림 7은 희생자 서버의 수에 따른 공격 교차율(attack intersection rate)을 보여준다. 본 논문에서 공격 교차율은 임의의 희생자 서버 k에 대한 공격의 집합을 A_k라고 했을 때 k에 대한 공격 a_k가 다른 희생자 서버에서도 탐지된 비율로 정의한다. 즉, 공격 교차율은 $U = \cup A_i (1 \le i \le l)$에서 $\exists a_k$가 $a_k \in A_k$이고 $a_k \in \overline{A_k}$인 비율이다. 희생자 서버의 수가 증가함에 따라 공격 교차율도 높아짐을 확인할 수 있다. 공격 정보를 공유하고 경계선에서 방어함으로써 교차율이 높은 보안 위협에 대해 효과적으로 대응할 수 있다. 바꿔 말하면, SAFE는 교차율이 높은 공격들로부터 Science DMZ 내부 서버들을 효과적으로 방어할 수 있다.

표 2는 침입탐지/방지(ID/IP) 방식에 따른 서버 별 1일 평균 공격자 수를 보여준다. 호스트 기반(Host-based) 방식은 각 희생자 서버에서 *logwatch*를 이용해 무작위 대입 공격을 탐지한 후 로컬 *iptables*를 설정해 추가적인 공격을 차단한다. SAFE를 통해 공격 정보를 공유함으로써 호스트 기반 침입방지 방식에 비해 공격 시도를 60% 이상 감소시켰다. 호스트 기반 방식이 TCP 전송 성능에 부정적인 영향을 주는 점을 고려할 때 소개된 SAFE 시나리오의 타당성이 높음을 확인할 수 있다.

패킷을 국지적으로 처리하기 위해 적용된 블룸 필

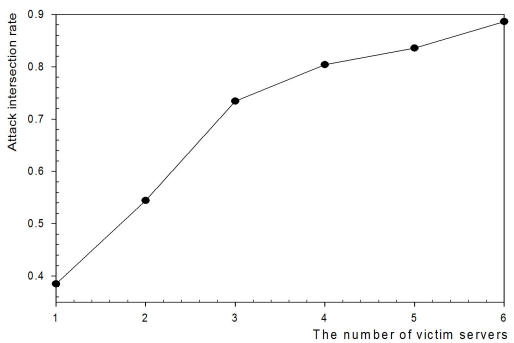

그림 7. 공격 교차율
Fig. 7. Attack intersection rate

표 2. 평균 공격자 수/1일
Table 2. Average number of attackers/day

ID/IP strategy	Attackers/day
Host-based	3.0418
SAFE-based	1.8334

터의 적중률은 표 3과 같다. 블룸 필터의 미적용 시와 비교했을 때 높은 적중률은 상대적으로 낮은 조회 병목과 네트워크 대역폭 소비를 의미한다. 적중률을 측정하기 위해 희생자 서버에서 발생된 웹 플로우(HTTP flow)들을 IPController에서 분석했다. IPController는 외부 접근이 허가된 플로우의 역 플로우 정보를 IPBox의 블룸 필터에 등록했다. 블룸 필터에 의해 국지적으로 처리되지 못한 플로우의 처리지침을 얻기 위해 플로우 테이블 조회가 요구된다.

4회에 걸친 측정 결과 94% 이상의 블룸 필터 적중률을 보였다. 바꿔 말하면 SAFE에 추가적으로 시그니처 정합 엔트리를 적용하면 완전일치 정합 엔트리만 이용했을 때보다 메시지 발생회수가 최소 46.9% 이상 감소시킬 수 있다. 또한 플로우 엔트리의 설치를 위해 IPBox와 IPController 간 SDN 메시지를 교환하지 않아도 되므로 플로우 설정 시간을 단축시키는 장점이 있다. 소개된 SAFE 응용 시나리오에서 메시지 발생회수와 대역폭 소비에 대한 최대 성능이득은 50%이다. 플로우 테이블 조회 요청 메시지와 응답 메시지의 크기가 동일하다고 가정하면 네트워크 대역폭 소비도 동일한 비율로 감소한다.

블룸 필터에 비 적중(missed flows)된 플로우가 발생하는 이유는 역 플로우 정보가 블룸 필터에 등록되기 이전에 해당 역 플로우가 IPBox에 도착했기 때문인 것으로 판단한다. 예를 들어, 희생자 서버와 접속 요청한 웹서버의 왕복지연시간이 플로우 테이블 조회 시간 보다 짧을 경우 비 적중 플로우가 발생한다. IPBox 및 IPController에서 SDN 메시지의 생성, 분석, 설치 등에 필요한 처리 시간이 병목으로 작용한 결과로써 구현 최적화를 통해 처리 시간을 줄일 수 있을 것으로 기대한다.

표 3. 블룸 필터의 적중률
Table 3. Hit-rate of Bloom filter

Total flows	Missed flows	Hit rate (%)
37,953	2,277	94.0
20,269	466	97.7
65,467	916	98.6
56,372	394	99.3

V. 결 론

본 논문은 자동화된 보안 정책의 적용이 가능하며 데이터 전송 성능에 영향을 주지 않는 SDN 기반의 침입방지 시스템을 소개했다. 제안된 시스템은 침입탐지, 방지 결정, 방지 시행 등의 방화벽 기능들을 물리적으로 분산하고 SDN 기술을 활용해 상호 연동시킴으로써 기존 방화벽이 갖는 성능 저하의 문제를 완화시켰다. 또한, 모듈화된 내부 구성요소들은 다양한 오픈소스 침입탐지소프트웨어들과 용이하게 연동될 수 있으므로 침입탐지시스템의 저비용 구축을 가능케 한다. 국가과학기술연구망에 응용 시나리오가 구현되어 제안한 시스템의 적용 가능성과 성능이 간접적으로 검증되었다. 추후 웹 응용에 대한 보안 강화를 위해 프록시 방식의 오픈소스 침입탐지시스템과의 연동 방안을 연구할 예정이다.

References

[1] E. Dart, L. Rotman, B. Tierney, M. Hester, and J. Zurawski, "The science DMZ: A network design pattern for data-intensive science," *Scientific Programming*, vol. 22, no. 2, pp. 173-185, 2014.

[2] N. McKeown, "Software-defined networking," *Keynote Talk at IEEE INFOCOM 2009*, Retrieved Aug., 27, 2014, from http://tiny-tera.stanford.-edu/~nickm/talks/

[3] I. Monga, E. Pouyoul, and C. Guok, "Software defined networking for big-data science - Architectural models from campus to the WAN," in *Proc. High Perf. Comput., Netw. Storage and Anal. (SCC)*, pp. 1629-1635, Salt Lake City, USA, Nov. 2012.

[4] J. Zurawski, "The science DMZ - introduction and architecture," in *Proc. Operating Innovative Netw. (OIN)*, Oct. 2013.

[5] P. Calym, A. Berryman, E. Saule, H. Subramoni, P. Schopis, G. Springer, U. Catalyurek, and D. K. Panda, "Wide-area overlay networking to manage science DMZ accelerated flows," in *Proc. IEEE Int. Conf. Comput. Netw. Commun. (ICNC)*, pp. 269-275, Feb. 2014.

[6] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, "Software as a service for data scientists," *ACM Commun. Mag.,* vol. 55, no. 2, pp. 81-88, Feb. 2012.

[7] N. Mckeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69-74, Apr. 2008.

[8] K. Curran, "An online collaboration environment," *Edu. Inf. Technol.*, vol. 7, no. 1, pp. 41-53, Mar. 2002.

[9] X. Gou and W. Jin, "Multi-agent system for multimedia communications traversing NAT/firewall in next generation networks," in *Proc. CNSR*, pp. 99-104, May 2004.

[10] A. Nayak, A. Reimers, N. Feamster, and R. Clark, "Resonance: dynamic access control for enterprise networks," in *Proc. 1st ACM Workshop on Research on Enterprise Netw.*, pp. 11-18, 2009.

[11] Z. A. Qazi, C. Tu, L. Chiang, R. Miao, V. Sekar, and M. Yu, "SIMPLE-fying middlebox policy enforcement using SDN," in *Proc. ACM SIGCOMM*, vol. 43, no. 4, pp. 27-38, Oct. 2013.

[12] H. Hu, W. Han, G. Ahn, and Z. Zhao, "FLOWGUARD: building roubust firewalls for software-defined networks," in *Proc. HotSDN*, pp. 97-102, Aug. 2014.

[13] R. Berthier, W. H. Sanders, and H. Khurana, "Intrusion detection for advanced metering infrastructures: requirements and architectural directions," in *Proc. IEEE SmartGridComm*, pp. 350-355, Oct. 2010.

[14] Sourcefire, *Snort*, Retrieved June 2, 2015, from https://www.snort.org/

[15] TrendMicro, *OSSEC(open source host-based intrusion detection system)*, Retrieved June 2, 2015, from http://www.ossec.net/

[16] Trustwave, *Modsecurity*, Retrieved June 2, 2015, from http://www.modsecurity.org/

[17] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 2, pp. 39-53, Apr. 2004.

[18] D. Hoffman, D. Prabhakar, and P. Strooper, "Testing iptables," in *Proc. CASCON,* pp. 80-91, 2003.

[19] B. Astuto, A. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: past, present, and future of programmable networks," *IEEE Commun. Survey & Tutorials*, vol. 16, no. 3, pp. 1617-1634, Feb. 2014.

[20] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: a survey," *Internet Math.*, vol. 1, no. 4, pp. 485-509, 2004.

[21] F. Bonomi, M. Mitzenmacher, R. Panigraphy, S. Singh, and G. Varghese, "An improved construction for counting bloom filters," in *Proc. 14th Conf. Annu. Eur. Symp.*, vol. 14, pp. 684-695, 2006.

[22] K. Bauer, *Logwatch*, Retrieved June 2, 2015, from http://www.logwatch.org/

[23] D. B. Cid, *Log analysis using OSSEC*, Retrieved June 2, 2015, from http://www.ossec.net/ossec-docs/auscert-2007-dcid.pdf

[24] J. W. Lockwood, N. McKeown, G. Watson, G. Gibb, P. Hartke, J. Naous, R. Raghuraman, and L. Jianying, "NetFPGA - An open platform for Gigabit-rate network switching and routing," in *Proc. IEEE Conf. Microelectronic Syst. Edu. (MSE '07)*, pp. 160-161, San Diego, Jun. 2007.

[25] J. Jo, S. Lee, and J. Kim, "Programmable IP service gateway for software-defined networking: assisting easy composition of service overlays," *IEICE Trans. Commun.*, vol. E96-B, no. 7, pp. 1918-1929, Jul. 2013.

[26] R. C. Andrew, C. M. Jeffrey, T. Jean, Y. Praveen, S. Puneet, and B. Sujata, "DevoFlow: scaling flow management for high-performance networks," in *Proc. ACM SIGCOMM*, vol. 41, no. 4, pp. 254-265, Aug. 2011.

[27] L. Dan, W. Andreas, H. Brandon, H. Nikhil, and F. Anja, "Logically centralized?: state

distribution trade-off in software defined networks," in *Proc. Hot Topics in Software Defined Netw.*, pp. 1-6, Jan. 2012.

[28] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, "Summary cache: a scalable wide-area Web cache sharing protocol," *IEEE/ACM Trans. Netw.*, vol. 8, no. 3, pp. 281-293, Jun. 2000.

조 진 용 (Jinyong Jo)

1999년 : 전남대학교 컴퓨터공학과(학사)
2002년 : 광주과학기술원 정보통신공학과(석사)
2013년 : 광주과학기술원 정보통신공학과(박사)
2003년 8월~현재 : 한국과학기술정보연구원
<관심분야> 사용자정의네트워킹, 멀티미디어 시스템 및 서비스, ID 페더레이션

장 희 진 (Heejin Jang)

2001년 : 포항공과대학교 컴퓨터공학과 졸업(학사)
2003년 : 포항공과대학교 대학원 컴퓨터공학과(석사)
2003년~2010년 : (주)삼성전자
2010년~2011년 : 국가보안기술연구소
2012년~현재 : 한국과학기술정보연구원
<관심분야> 네트워크, 정보보안 분야

이 경 민 (Kyoung-Min Lee)

2011년 : 안동대학교 컴퓨터공학과 졸업(학사)
2013년 : 안동대학교 컴퓨터공학과(석사)
2013년~현재 : 한국과학기술정보연구원
<관심분야> 네트워크, 망 제어 및 관리, 웹 서비스

공 정 욱 (JongUk Kong)

1993년 : 한국과학기술원 전기 및 전자공학과 졸업(학사)
1998년 : 포항공과대학교 정보통신대학원 정보통신공학과 졸업(공학석사)
2015년 8월 : 충남대학교 정보통신공학과 졸업 예정(공학박사)
1993년~2001년 : (주)데이콤 중앙연구소 선임연구원
2001년~2002년 : (주)맥스웨이브 책임연구원 및 한국항공우주연구원 초빙연구원
2002년~현재 : 한국과학기술정보연구원 책임연구원
<관심분야> 망 자원 관리 및 제어