www.osia.or.kr

OSIA S&TR Journal

ISSN 1738-9887 Vol. 31, No. 3, September 2018
• 발행인 : 정성호 회장/OSIA • 편집위원 : 한연희 교수/한국기술교육대

Editorial

02 IETF 국제 표준 최신 기술 동향
한연희 교수/한국기술교육대

Article

04 IETF Hackathon – I2NSF WG을 중심으로
김영한 교수/숭실대

08 IETF suit과 teep WG 표준화 동향
이재훈 교수/동국대

14 IETF/IRTF ICNRG 기술 동향
홍정하 박사/ETRI

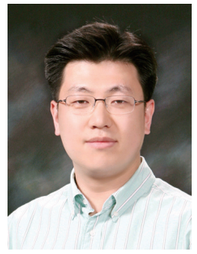
21 IETF 및 IRTF에서의 인터넷 효율성 향상을 위한 표준화 연구 동향
김평수 교수/한국산업기술대

27 BitString 기반 RPL 확장 기술
윤주상 교수/동의대

33 IETF/IRTF 에서의 블록체인 기술
이종혁 교수/상명대

(사)한국컴퓨터통신연구회
Open Standards and ICT Association

IETF 국제 표준 최신 기술 동향



한국기술교육대 교수

한 연 희

인터넷에 관하여 이해관계가 있는 기업, 단체, 혹은 개인이 중심이 되어 사실(De Facto) 표준을 제정해온 국제 표준화 기구인 IETF는 1986년 1월, San Diego에서 첫번째 회의를 개최한 이후 약 30여년 동안 인터넷 구조의 개선, 인터넷의 원활한 운영 및 기술적인 문제에 대한 해결방안을 활발히 모색해 오면서 어느덧 2017년 11월에 100차 IETF 회의를 넘어서 2018년 가을에는 103차 IETF 회의가 방콕에서 열린 예정입니다. 최근에 대부분의 사물들(Things)에게 인터넷 통신 기능을 부여하려는 노력이 사물인터넷(Internet of Things, IoT) 기술로 부각되면서 IETF에서 발행하는 RFC 표준 문서들이 많은 조명을 받고 있으며, 미래지향적인 새로운 인터넷 구조, 데이터 전송 관리 방안, 서비스 관리 방안과 관련된 국제 표준화가 활발히 이루어지고 있습니다. 이에 본 호에서는 "IETF 국제 표준 최신 기술 동향"을 주제로 최근 IETF에서 활발히 논의가 되고 있는 중요 기술들에 대해 정리하여 소개하고자 합니다.

첫번째 원고는 IETF회의와 함께 개최되는 IETF Hackathon에 대한 소개 및 최근 IETF Hackathon행사에서 발표된 I2NSF (Interface to Network Security Functions) 기술을 구현 및 구축 관점으로 자세히 설명합니다. 두번째 원고는 사물인터넷 기술이 보편화됨에 따라 요구되어지는 부가 기능인 SUIT (Software Updates for Internet of Things)과 TEEP (Trusted Execution Environment Provisioning) 기술에 대하여 관련 워킹그룹 활동을 중심으로 소개합니다. 세번째 원고는 정보 중심 네트워크(Information-Centric Networking, ICN)기술을 소개하고 본 기술을 논의하는 IETF/IRTF ICNRG 활동을 자세히 설명합니다. 네번째 원고는 IETF/IRTF에서 인터넷 효율성과 관련된 QUIC (Quick UDP Internet Connections), MAP (Measurement and Analysis for Protocols), ICC (Internet Congestion Control), NWC (Network Coding for Efficient Network Communications), MPTCP (Multipath TCP) 기술들을 관련 그룹들의 표준화 활동과 함께 소개합니다. 다섯 번째 원고는 저전력 디바이스로 구성된 네트워크에서 효율적으로 사용 가능한 라우팅 프로토콜로 주목받고 있는 BitString 기반 RPL (Routing Protocol for Low Power and Lossy Networks) 확장 기술을 다양한 그림과 함께 그 동작 방식을 비교적 자세히 설명합니다. 마지막으로, 여섯 번째 원고는 최근 이슈가 되고 있는 Blockchain 기술이 IETF에서도 여러 그룹 및 다양한 방법으로 논의가 되고있어서, 이러한 IETF에서의 Blockchain 기술 논의 사항을 정리하여 소개합니다.

이번 호에 훌륭한 원고를 제출하여 주신 저자분들께 진심으로 감사드리며, IETF의 최신 기술, 특히 사물인터넷, 인터넷 효율성 및 미래지향 인터넷 관련 기술을 연구하시거나 현업에 도입하려는 학계/산업계 분들에게 본 호가 많은 도움이 되기를 기대합니다.

IETF Hackathon – I2NSF WG을 중심으로

김영한[1], 양현식[2], 정재훈[3]
[1]숭실대학교 전자정보공학부, [2]숭실대학교 정보통신공학부, [3]성균관대학교 정보통신대학 소프트웨어학과

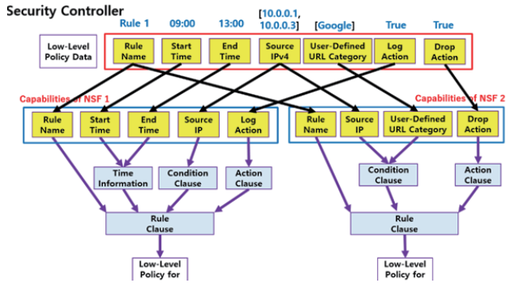
Abstract

국제 인터넷 표준화 기구 IETF (Internet Engineering Task Force)에서는 매 회의마다 Hackathon 행사를 진행하고 있다. Hackathon 은 IETF에서 논의되고 있는 프로토콜이나 프레임워크를 검증하기 위해 오픈소스로 구현하는 행사이다. 매 회의 전에 Hackathon 관련 주제를 모집하여 공지하며 참여를 원하는 사람은 Hackathon 공지사항을 통해 확인하고 사전 신청이 가능하다. 최근에 진행된 회의에서는 SUIT, NETCONF, YANG, ACE, I2NSF, MLS 등 약 24개 주제를 가지고 Hackathon이 진행되었으며 약 300명이 참가하였다. Hackathon 행사는 2일 동안 진행되며, 마지막 날에는 Hackathon 결과를 공유하고 발표하는 시간을 갖는다. I2NSF WG에서는 지속적으로 Hackathon에 참여하였는데 이번 Hackathon에도 참여하여 표준화된 인터페이스 및 컴포넌트를 구현하고 기능 및 시나리오를 검증하였다. 본고에서는 IETF 회의에서 진행된 Hackathon을 I2NSF WG을 중심으로 소개하고 Hackathon 결과 및 다음 Hackathon에 대한 논의사항을 소개하고자 한다.

I. IETF Hackathon

IETF Hackathon은 IETF에서 논의되고 있는 프로토콜이나 프레임워크를 검증하기 위해 오픈소스로 구현하는 행사로 IETF 회의 시작 전에 먼저 진행된다. Hackathon은 각 그룹에서는 현재 진행중인 문서에서 정의하고 있는 내용을 검증하기 위한 행사로 실제 구현 이슈가 있거나 구현을 원하는 그룹에서는 누구나 참여 가능하다.

Hackathon을 진행하기 원하는 단체에서는 구현 주제와 내용에 대해 Hackathon 사이트에 공지를 올리며, 원하시는 사람은 Hackathon 등록을 통해 참여 가능하다. 2일동안 진행되는 Hackathon 행사에서는 표준 문서를 기반으로 구현이 진행되며 각 구현 물 및 Hackathon 결과는 마지막 날 발표를 통해 공개된다. 이때, 투표를 통해 많은 호응을 얻은 팀에게는 상을 수여하기도 한다. 이번 102차 회의 에서도 약 24개의 워킹 그룹에서 약 300명이 참가하였다. 이번 Hackathon에 참가한 그룹 중 I2NSF WG은 NFV(Network Function Virtualization) 환경에서 다양한 보안 기능을 관리하고 연결하기 위한 프레임 워크 및 인터페이스 등을 정의하고 있다[1]. 최근 회의에서도 정의된 인터페이스에 대한 인포메이션 모델 및 데이터 모델에 대한 WG 문서의 진행사항을 중심으로 논의가 진행되었다. 이번 Hackathon에서도 I2NSF 프레임 워크에 정의된 각 기능들과 인터페이스를 중심으로 두가지 시나리오를 구현하였다. 첫번째 시나리오는 웹필터 시나리오로 사용자가 정의한 특정 사이트에 대한 접속 및 차단을 사용자 요구사항에 맞게 설정하는 시나리오이다. 클라이언트를 통해 특정 사이트에 접속하는 경우, 정해진 NSF를 지나서 가도록 경로가 설정되어 정책에 따라 접속을 허용하거나 차단되는 것을 확인할 수 있도록 하였다. 두 번째 시나리오는 특정 웹 메일에 대한 허용 및 차단에 대한 시나리오이다. 이 시나리오는 클라이언트에서 메일을 송수신하는 경우, 사용자가 정의한 정책에 따라 도메인을 허용 및 차단할 수 있다. 앞서 소개된 시나리오 구현을 위해 Mininet을 이용하여 환경을 구성하고 정의된 데이터 모델을 중심으로 인터페이스 및 컴포넌트를 구성하고 검증하였다. 특히 이번 Hackathon에서는 Security Controller의 Policy Translation Model을 추가 구현하여, 사용자의 요청이 low level policy로 해석되어 각 NSF에 전달되는 기능의 동작을 확인하였다[2]. 그림1은 구성된 Translator의 Policy 적용 방법을 보여준다. 사용자가 내린 High level Policy를 분리하여 low level policy에 알맞게 해석한 후 각 기능이 포함된 NSF와의 매핑을 통해 각 NSF(Network Security Function)에 수행해야 할 정책을 전달한다. 정책을 받은 NSF는 전달된 정책에 따라 기능을 수행하게 된다. 시나리오 상에서 High level 정책은 웹페이지를 통해 입력 받았으며, 구현된 Translator에 전달되어 low level policy로 변경 후 각 NSF에 전달하도록 구성하였다. 앞서 논의된 두가지 시나리오 역시 웹을 통해 정책을 받아 Translator를 통해 해석한후 NSF에 전달되는 절차를 적용하여 검증하였다.

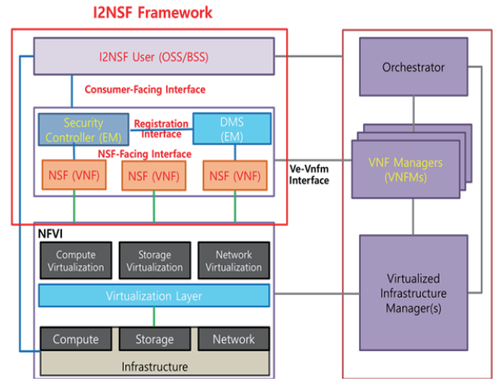


〈그림 1. I2NSF Security Controller 내 Translator〉

IETF Hackathon에서는 행사를 종료하기 전에 좋은 결과를 보여준 그룹에 대하여 시상을 진행한다. 이번 회의에서 진행된 Hackathon에서는 DNS / DNSSEC/ DNS Privacy에서 Multi-provider 모델 및 관련 API를 구성하여 상을 받았으며, PERC(Privacy Enhanced RTP Conferencing) 와 SUIT(Software Updates for Internet of Things)에서도 진행 중인 시나리오 구현 및 multiple OS 구성을 하여 상을 받았다. 이와 관련된 자세한 진행사항은 Hackathon Github 및 IETF 미팅 기록을 통해 확인할 수 있다.

II. I2NSF architecture on NFV reference architecture

I2NSF WG에서 현재 진행 중인 WG 그룹 문서인 Applicability of Interfaces to Network Security Functions to Network(draft-ietf-i2nsf-applicability-04)에서는 NFV 환경에서 I2NSF 프레임워크 및 인터페이스를 적용하기 위한 방법에 대한 논의가 진행 중이다[3]. 위 문서에는 NFV 환경에서 제공할 수 있는 I2NSF 유즈케이스 및 구조에 대해 설명하고 있으며, 최근에는 NFV Reference Architecture에서의 I2NSF 기능 배치를 위한 방법을 제안한 I2NSF on the NFV Reference Architecture(draft-yang-i2nsf-nfv-architecture-02) 문서와 통합되어 배치 방법이 추가되었다[4]. I2NSF on the NFV Reference Architecture(draft-yang-i2nsf-nfv-architecture-02)문서에서 정의한 I2NSF 환경을 NFV 환경에서 제공하기 위해 제안된 구조는 그림 2와 같다.



〈그림 2. I2NSF architecture on NFV reference architecture〉

본 문서에서는 NFV 환경에 정의된 기능들을 기준으로 I2NSF 컴포넌트를 분류하였다. 이후 NFV 특성과 기능적으로 차이가 없는지 확인하는 절차를 통해 구성 가능한 모델을 제시하였다. 먼저 Developer's Mgmt System은 NSF를 제공하는 벤더로 NSF를 Security Controller에 등록하는 역할을 한다. NFV 환경에서 Developer's Mgmt System은 EM(Element Management)의 역할로서 각 NSF 구성을 위한 관리 구조도 적용 가능하다. 각 벤더는 Developer's Mgmt 시스템을 통해 NSF의 생성 및 관리를 요청할 수 있으며, 요청에 따라 Developer's Mgmt는 요청 메세지를 전달할 수 있다. 예를 들어 Developer's Mgmt에게 NSF 생성 메세지가 전달되면 Developer's Mgmt System은 VNFM(VNF Manager)에게 NSF 생성 요청 메세지를 전달할 수 있다. 생성 요청에 의해 구성된 NSF는 Developer's Mgmt System을 통해 Security Controller에 등록될 수 있다. 다음으로 Security Controller는 정책을 받아 각 NSF에 전달하는 역할로 NFV 환경에서는 특별한 VNF를 관리하는 VNFM이나 독립된 기능 혹은 특별한 VNF를 관리하는 EM 기능으로 고려될 수 있다. Security Controller는 NSF 내 정책 관리를 위해서 NSF와 통신이 가능해야 한다. 또한 NSF의 정책을 관리하는 역할을 해야 한다. 즉, Security Controller는 NSF와의 통신이 필수적이다. 기능적 측면에서 Security Controller는 EM에 해당한다. 하지만, 배포 측면에서는 NSF와의 통신만 보장되면 VNF나 독립적인 구성요소로 배포되어 동작할 수 있기 때문에 EM 혹은 하나의 독립된 컨트롤러로서 구성될 수 있다. 각 인터페이스는 기본적으로 컴포넌트 배치에 따라 달라질 수 있지만 앞서 논의된 구조에 따르면 Consumer-Facing Interface, Registration Interface 및 NSF-Facing Interface는 기존 모델과 동일하게 정해진 구성요소 간의 통신을 위해 사용할 수 있다. 이번 102차 회의에서도, NFV 적

용 방안에 대한 논의가 진행되었는데 그림2와 같은 구조가 MANO와 독립적이며, 각 벤더가 각 NSF를 동적으로 관리할 수 있는 구조로 적합하다는 의견이 나왔다.

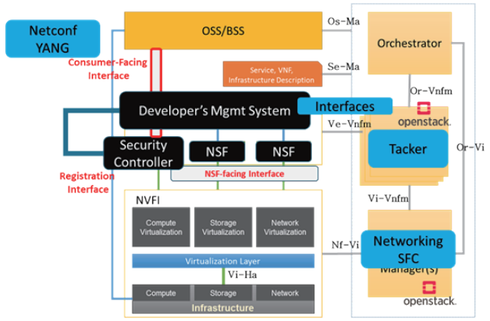
그러나 실제 NFV 환경 적용을 위해서는 좀 더 다양한 시나리오 및 실제 구현 모델을 기반으로 한 설계가 필요할 것이다.

III. NFV에서의 I2NSF 환경 적용 구조 설계

A. NFV에서의 I2NSF 기본 환경 구축

이번 102차에서 진행된 I2NSF WG Hackathon은 NFV 환경이 아닌 일반적인 서버로 구성된 환경에서의 I2NSF의 프레임워크 및 시나리오를 검증하였다. 앞서 논의한 것처럼, I2NSF 프레임워크를 NFV 환경에서 제공하기 위한 방법들이 고려되어야 하며 각각의 기능들을 배포하기 위한 구조들이 고려되어야 한다. 본 장에서는 NFV 환경에서 I2NSF 프레임워크를 구성하기 위해 필요한 구성 환경 및 설계 기능들에 소개하고자 한다. 그림 3은 NFV 환경 구성을 위해 필요한 오픈소스들을 나타낸 것이다. NFV 환경을 구성하기 위해서는 먼저 NFV 환경을 제공하는 오픈소스 플랫폼이 필요하다. NFV 환경을 제공하는 대표적인 오픈소스로는 OpenStack이 있다[5]. OpenStack은 대표적인 NFV 플랫폼으로 Host OS 환경에서 KVM을 이용하여 서버의 리소스를 가상화 하여 NFV 환경을 제공한다. OpenStack을 통해 환경을 구성하게 되면 기본적인 클라우드 환경을 이용할 수 있다. MANO 기능 제공을 위해서는 추가적인 기능을 설치해야 한다. MANO (network functions virtualization management and orchestration) 기능과 관련된 오픈소스로는 OpenStack 프로젝트인 Tacker, Fraunhofer의 Open Baton, ETSI OSM 등이 대표적이며, 본고에서는 OpenStack Tacker 프로젝트를 사용한 방법에 대해 소개하고자 한다[6][7][8]. I2NSF 환경을 구성하는 컴포넌트는 위에서 언급한 대로 배치 가능하며 배치를 위한 고려 사항은 다음과 같다. 먼저 NSF는 VNF를 통해 구성 가능하다. 각 보안 기능들은 개별적인 설치가 필요하며, VNFM을 통해서도 설치할 수 있다.

Developer's Mgmt System은 NFV 환경에서 EM으로 배치할 수 있으며 VNF 관리를 위해 VNF와 EM간의 통신이 가능해야 한다. Security Controller는 하나의 독립된 개체로 설치하거나 EM으로서 설치할 수 있다. 두 경우 모두 NSF와 통신이 가능하도록 환경을 구성해야 한다. VNFM은 I2NSF 프레임 워크 자체에는 명시되어 있지 않지만 NFV 환경을 고려하는 경우, VNFM을 통해 필요한 NSF를 생성할 수 있다.



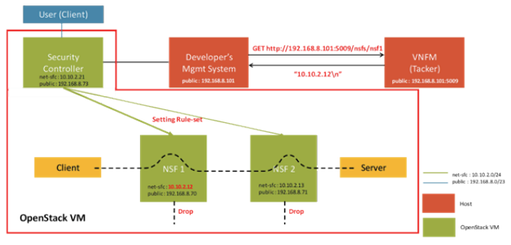
〈그림 3. NFV 환경에서 I2NSF 구현을 위한 기능 구성도〉

이를 위해서는 Security Controller와 VNFM 사이에 통신이 가능하도록 인터페이스를 설정해 줘야 한다. 인터페이스를 통해 새로운 NSF에 대해 요 해야 하며, 요청 후 생성된 정보를 받아 Security controller에 등록하는 역할을 해야 한다.

B. NFV에서의 I2NSF 유즈케이스 환경 구축

I2NSF에서 프레임워크를 통해 유즈케이스를 제공하기 위해 그림 4와 같이 환경을 구축하였다.

먼저 사용자 역할을 하는 클라이언트 VNF를 한 개 구성하였고, 도메인 차단 기능과 웹 필터 기능이 제공되는 NSF를 구성하였다. 패킷의 흐름을 제어하기 위해 각 기능들 간의 패킷 전달은 Networking SFC를 통해 구성하였다[9]. 구성한 환경에서 웹 필터 유즈케이스 시나리오는 다음과 같다. Security Controller에서 특정 사이트에 대해 보안 서비스를 요청하면, 이 정보는 Translator를 통해 low level로 변환되고 구성된 네트워크를 통해 NETCONF 프로토콜로 각 NSF에 전달된다.



〈그림 4. 유즈케이스 검증을 위한 환경 구성〉

이후 클라이언트가 룰이 생성된 특정 사이트로 접속을 시도하면 룰이 설정된 NSF에서 접속을 차단한다. 도메인 차단 시나리오의 경우, 클라이언트는 이메일 전송 기능을 제공하도록 구성하였으며, 도메인 차단을 위한 룰을 네트워크를 통해 NETCONF 프로토콜로 각 NSF에 전달하였다. 클라이언트에서 차단된 도메인으로 메일을 전송하는 경우, 특정 도메인에 대한 차단룰을 가진 NSF가 패킷을 확인하고 차단하도록 설정하였다.

IV. 결론

IETF Hackathon은 IETF에서 논의되고 있는 프로토콜이나 프레임워크를 검증하기 위해 오픈소스로 구현하는 행사이다. Hackathon은 정의된 문서에 대한 검증기능도 제공하지만 구현을 통해 신빙성을 확보할 수 있어 앞으로 진행될 표준화 기능들을 사전 검증도 가능하다. 최근 IETF에서도 다양한 WG 그룹에서 실제 구현 이슈들도 함께 논의가 되고 있으며, 점점 많은 WG에서 Hackathon 행사에 참여하는 만큼 그 의의는 점점 더 중요해지고 있다. I2NSF WG도 기존의 문서들을 실제 구현하여 검증하였으며, 차후 진행될 문서 내용에 대한 구현을 통해 지속적으로 Hackathon에 참여하여 표준화를 위해 노력하고 있다. 다음 Hackathon에서는 NFV 환경을 고려한 I2NSF 환경 구현을 위해 Hackathon에 참여할 예정이다. 이를 위해 최적화된 환경 구성 및 기능 배치를 위한 연구를 지속적으로 진행할 예정이다.

Acknowledgement

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터육성지원사업의 연구결과 (IITP-2018-2017-0-01633) 및 IETF 미러포럼의 지원을 받음

Reference

[1] "Interface to Network Security Functions (i2nsf)", I2NSF WG, (Accessed 08.16.18).
[2] "Security Policy Translation in Interface to Network Security Functions", I2NSF WG draft-yang-i2nsf-security-policy-translation-01, July, 2018.
[3] "Applicability of Interfaces to Network Security Functions to Network-Based Security Services", I2NSF WG draft-ietf-i2nsf-applicability-04, July, 2018.
[4] "I2NSF on the NFV Reference Architecture", I2NSF WG draft-yang-i2nsf-nfv-architecture-02, July, 2018.
[5] OPENSTACK: https://www.openstack.org/, (Accessed 08.16.18).
[6] TACKER: https://wiki.openstack.org/wiki/Tacker, (Accessed 08.16.18).
[7] OPEN BATON: https://openbaton.github.io/, (Accessed 08.16.18).
[8] ETSI OSM: https://osm.etsi.org/, (Accessed 08.16.18).
[9] Networking SFC: https://docs.openstack.org/networking-sfc/latest/, (Accessed 08.16.18).

Biographies



김영한

1984년: 서울대학교 전자공학 학사
1986년: 한국과학기술원 전기전자공학 석사
1990년: 한국과학기술원 전기전자공학 박사
1994년 ~ 현재: 숭실대학교 교수
2017년 ~ 현재: 한국통신학회 부회장
주요 관심분야: SDN/NFV, 오픈소스 네트워킹, 엣지/포그컴퓨팅



양현식

2013년: 숭실대학교 정보통신전자공학 학사
2013년 3월 ~ 현재: 숭실대학교 정보통신공학부 석박통합과정
주요 관심분야: SDN/NFV, 클라우드, 오픈소스 네트워킹, IoT 클라우드



정재훈

1999년: 성균관대학교 정보공학 학사
2001년: 서울대학교 컴퓨터공학 석사
2009년: 미네소타대학교 컴퓨터공학 박사
2004년: 한국전자통신연구원 표준연구센터 연구원
2012년: Brocade Communications Systems 소프트웨어 엔지니어
2012년 ~ 현재: 성균관대학교 정보통신대학 소프트웨어학과 조교수
주요 관심분야: Internet of Things (IoT), Security Services for Software-Defined Networking (SDN) and Network Functions Virtualization (NFV), Cyber-Physical Systems (CPS), Indoor/Outdoor Navigation, Vehicular Ad Hoc Networks, Wireless Sensor Networks, Mobile Ad Hoc Networks

IETF suit과 teep WG 표준화 동향

이재훈
Dongguk University

Abstract

인터넷이 우리의 실생활에 밀접하게 파고들면서, 인터넷/모바일을 통한 금융 서비스가 보편화되었을 뿐만 아니라 개인 정보의 클라우드 서비스를 통하여 언제 어디서나 어떤 기기를 이용해서도 자신의 정보를 액세스하는 것이 가능하게 되었다. 또한 센서나 홈 라우터, 가전 등도 인터넷에 접속되면서 센서를 통하여 감지되는 비 정상적인 상황에 대해서 사람의 관여 없이 능동적으로 대처하거나 유명 요리사의 요리법에 따라 요리할 수 있고 또한 내가 가고자 하는 목적지까지 자동차가 알아서 운전하는 자율 운전 시스템과 같이 Internet of Thing (IoT) 기술은 사람들에게 보다 편리한 세상을 제공하게 되었다. 그렇지만 IoT 기술이 보편화됨에 따라 악의적인 목적을 가진 사람들이 타인의 개인 정보를 이용하여 금융 계좌에 접속하거나, 타인을 사칭하여 마치 자신이 친구인 것처럼 가장하거나, 집안의 방범 장비에 접속하여 개인의 사생활을 엿보거나 또는 도로나 전력과 같은 사회 인프라 망에 접속하여 제어 시스템을 조작하여 사회를 혼란에 빠뜨리는 경우가 발생할 수 있다. 이에 IETF에서는 IoT 환경에서 보안 문제를 해결할 수 있는 방법을 모색하고 있다. 본 고에서는 현재 IETF에서 진행 중인 Software Update for Internet of Things (suit)과 Trusted Execution Environment Provisioning (teep) 워킹 그룹의 표준화 동향에 대해서 살펴본다.

I. 서론

2016년 10월 21일에 '미라이 봇넷'이라고 명명된 봇넷이 약 10만대의 관리자 계정 설정이 취약한 CCTV와 NAS와 같은 IoT 장비에 접속하여 악성코드를 전파하여, 악성코드에 감염된 IoT 장비가 명령에 따라 DNS 제공 업체인 DYN에 DDoS 공격을 수행하여, 미국의 인터넷이 약 3시간 정도 마비되는 사건이 발생하였다[1]. 이를 방지하기 위하여 인터넷에 연결된 IoT 장비의 초기 관리자 모드의 패스워드를 복잡도가 높은 패스워드로 변경하도록 권고하였는데, 그 이후에 이러한 '미라이 봇넷'보다 최소 20배는 더 강한 'IoT 리퍼'라고 하는 IoT 좀비 네트워크가 확산되었다[2]. IoT 리퍼는 미라이 봇넷보다 더욱 진화하였는데, 미라이 봇넷이 취약한 비밀번호를 사용하는 IoT 기기를 대상으로 하는 것에 비하여 IoT 리퍼는 IoT 자체의 보안 취약점을 이용하여 IoT 기기를 감염시켜 좀비로 만든다. 이와 같은 취약점에 노출된 장비는 디링크와 고어헤드, JAWS, 넷기어, 바크론, NVR, 링크시스, AV텍 등에서 만든 IoT 기기들이며, 보안 취약점이 발견된 장비 제조사들은 이를 해결하기 위하여 자신들이 개발한 IoT 장비들의 코드를 수정하고 있다. 이렇게 수정된 코드들은 이미 시중에 설치되어 있는 IoT 장비들에 업데이트되어야 하는데 이러한 IoT 장비들에 새로운 펌웨어를 업데이트하는 것은 상당히 어려운 일이라 할 수 있다. 이는 IoT 장비에 보안 취약점이 발견되는 경우에 펌웨어 업데이트가 수행되어야 하지만, IoT 장비 사용자는 자신의 장비를 업데이트하는 것에 많은 관심을 기울이지 않고 초기 사용 모드를 그대로 고수하는 경우가 대부분이며, 이미 수 많은 IoT 장비가 설치되어 있는 환경에서 이러한 IoT 장비를 수동으로 업데이트한다는 것은 막대한 시간과 인력 그리고 비용이 들어갈 수 있기 때문에 장비 제조사/운용자는 이러한 오버헤드가 들어가는 것을 원하지 않는 경우가 있을 수 있기 때문이다. 또한 IoT 장비가 사람의 손에 미치지 않는 곳에 설치되어 동작하고 있는 경우에는 수동으로 펌웨어를 업데이트하는 자체가 거의 불가능한 경우도 발생하게 된다. 만일 장비 제조자/운용자가 업데이트하고자 하는 펌웨어를 서버에 업로드하고 IoT 장비 스스로가 펌웨어를 다운로드하여 업데이트할 수 있다면 시간, 비용, 인력과 같은 막대한 오버헤드의 지출을 줄일 수 있을 것이다. IoT 장비에 펌웨어를 자동으로 업데이트하기 위해서는 (1) IoT 장비가 신뢰할 수 있는 소스로부터 새로운 펌웨어 이미지를 수신하고, (2) 새로운 이미지를 설치한 후, (3) 새로운 이미지를 구동해서, (4) 새로운 이미지가 완벽하게 동작하면 모든 것이 원활하게 동작하는 것처럼 단순하게 보일 수도 있을 것이다. 그렇지만 실제로는 IoT 장비에 펌웨어를 업데이트하는 것이 그렇게 간단한 문제가 아닌 것이 IoT 장비는 (1) 때때로 네트워크와 연결이 단절되고, (2) 전력이 끊기고, (3) 동작이 되지 않는 펌웨어를 수신하고, (4) 자신이 아닌 다른 장비를 위한 펌웨어를 수신하고, (5) 신뢰할 수 없는 소스로부터 펌웨어를 수신하고, (6) 펌웨어 이미지를 설치하는 도중에 Flash 메모리에 고장이 발생할 수 있고, (7) 새로운 펌웨어 이미지를 구동하는데 실패하고, (8) 새로운 이미지를 구동해서 장비를 제어하는데 실패하는 등의 문제가 발생할 수 있다[3]. IETF Software Update for Internet of Things (suit) 워킹 그룹은 이와 같은 문제를 해결하고 신뢰성 있는 방법으로 펌웨어를 업데이트할 수 있도록 하기 위한 표준화를 진행 중에 있다.

또한 메인 프로세서를 일반 영역과 보안 영역으로 분리하여 일반 영역에서는 소위 Rich Execution Environment (REE)라고 하는 안드로이드, 리눅스 또는 윈도우와 같은 기존 운영체제에서 동작하는 응용이 저장/설치/동작하는 반면에 Trusted Execution Environment (TEE)라고 하는 보안 영역은 인증 정보와 보안 문서와 같은 파일이나 응용에게 향상된 보안을 제공하기 위하여 설계되고 사용되는 것으로서 삼성의 KNOX에 적용된 ARM의 TrustZone과 Microsoft의 CoCo Framework에 적용된 Intel의 SGX가 대표적인 TEE의 예라 할 수 있다[4,5]. 향상된 보안 기능을 제공하는 TEE는 다음과 같은 분야에 적용될 수 있다[6].

(1) 결제

휴대폰과 같은 모바일 기기에서 결제 관련 응용들은 높은 보안과 상호 신뢰를 필요로 한다. 기기 내의 TEE 내에서 구동되는 Trusted Application(TA)을 이용하여 결제를 수행하면 좀 더 강력한 식별과 트란젝션의 증명을 제공할 수 있다. 또한 홍채 인식이나 지문 인식과 같은 생체 식별 정보 등이 TEE 내에 저장될 수 있으며 이러한 생체 식별 정보를 이용하여 인증을 수행할 수 있다. 또한 보안 사용자 인터페이스를 이용하여 악의적인 소프트웨어가 사용자의 민감한 정보를 훔치는 것을 방지할 수 있다. 이러한 응용은 사용자 입력 정보의 보호를 위하여 TEE 내에 구현될 수 있다.

(2) 인증

기기는 좀 더 높은 수준의 보안 기반의 인증을 제공하기 위하여 민감한 인증 관련 키를 TEE 내에 저장함으로써 하드웨어에 의해서 보호되는 보안 키 강도 및 신뢰할 수 있는 실행 코드를 제공할 수 있다.

(3) IoT

IoT는 기기의 보안 취약성으로 인하여 네트워크와 국가 인프라에 위협을 가할 수 있다. 따라서 악의적인 공격자가 IoT 장치 내에 있는 인증 관련 신분증 (Credential)과 같은 민감한 정보를 훔치거나 또는 수정하지 못하도록 하는 것이 필수적이라 할 수 있다. TEE는 이러한 IoT의 보안 기능을 구현하기 위한 가장 좋은 방법이라 할 수 있다.

(4) 기밀 클라우드 컴퓨팅

Tenant가 클라우드 컴퓨팅 서버에 개인 정보와 같은 민감 데이터를 저장하는 경우에, tenant 만이 자신의 데이터에 접근할 수 있도록 하고 클라우드 호스트 제공자는 이러한 데이터에 접근하지 못하도록 하는 것이 필요하다 할 수 있다. Tenant는 서버 TEE에 TA를 구동함으로써 tenant에게는 민감 데이터에 대한 안전한 운영과 좀 더 높은 수준의 보안을 제공할 수 있을 뿐만 아니라 클라우드 호스트 제공자에게는 자신이 tenant의 데이터를 검색하거나 노출할 수 없다는 것을 강조함으로써 tenant의 민감 데이터의 유출 가능성에 대한 위험성과 그로 인한 책임감을 최소화하고 또한 안정성 제공이라는 장점을 부각함으로써 클라우드의 사용을 활성화시키는 것이 가능하다.

그런데 TEE의 향상된 보안 제공과 같은 많은 장점에도 불구하고 TEE에서 동작하는 TA의 개발 및 사용은 활성화되지 못한 실정이다. 이는 다양한 환경에서 TEE가 적용될 수 있음에도 불구하고 기존의 앱 사용자가 앱스토어를 통하여 일반적인 응용을 등록하고 앱 사용자는 앱스토어에 등록되어 있는 응용을 다운로드하여 설치하고 사용하는 것과 같이 응용의 등록과 배포가 용이함에 비하여 TEE의 기능을 이용하여 TA를 개발하고자 하는 개발자는 TEE에 접근하는 것이 상당히 어렵기 때문에 TA를 등록하고 배포하는 것도 또한 상당히 어려우며, 따라서 현재에는 개발자/서비스 운용자들 간에 독자적인 프로토콜을 이용하여 TA가 설치되고 동작되어 확장성이 떨어진다는 단점이 있다. 이는 TEE 자체가 인증된 신뢰할 수 있는 코드로 간주되기 때문에 보안의 관점에서 새로 설치되는 TA가 신뢰할 수 있는 응용인지 또는 TA에 대한 업데이트 코드가 신뢰할 수 있는 코드인지를 판단하기가 어렵기 때문이다. 즉 (스마트폰과 같은) 기기 개발자 및 소유자, TEE, 그리고 TA 개발자 간에 "상호 신뢰"를 구축하기 위해서는 확장성을 제공하면서도 다음과 같은 문제를 해결할 수 있어야 한다.

(1) TEE 제공자는 TA 개발자와 TA를 신뢰할 수 있는지를 어떻게 검증할 것인가?
(2) 기기 소유자는 어떤 TA 개발자를 신뢰할 수 있을 것이며 또한 어떤 TA를 설치하도록 할 것인가?
(3) 기기 개발자는 TA의 업데이트를 어떻게 신뢰할 것인가?
(4) TA 개발자는 서로 다른 형태의 TEE가 구현된 수 많은 기기들에게 어떻게 자신이 개발한 TA를 업데이트하도록 할 것이며, 이 경우 TA가 설치 또는 업데이트되는 기기가 신뢰할 수 있는지를 어떻게 판단할 것이며, 또한

원격에 있는 기기 자체를 어떻게 식별할 것인가?

IETF Trusted Execution Environment Provisioning (teep) 워킹 그룹에서는 상호 신뢰할 수 있는 방법으로 TEE에 TA를 설치 또는 업데이트할 수 있는 개방화된 프로토콜 개발을 목표로 하고 있다.

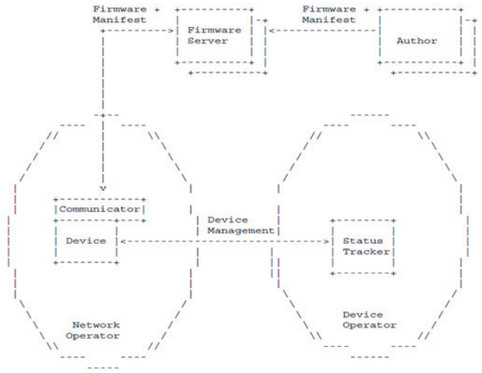
본 고에서는 앞에서 언급한 IETF의 suit과 teep 워킹 그룹의 표준화 동향을 살펴보고자 한다. 본 고는 다음과 같이 구성되어 있다. 2장에서는 IETF의 suit 워킹 그룹의 표준화 동향에 대해서 살펴보고, 3장에서는 teep 워킹 그룹의 표준화 동향에 대해서 살펴본다. 그리고 4장에서 결론을 내린다.

II. IETF suit 워킹 그룹 표준화 동향

Suit 워킹 그룹은 100차 IETF 회의에서 BoF가 열렸고, 101차 IETF 회의에서 정식 워킹 그룹으로 활동을 시작하였다[7]. Suit 워킹 그룹에서는 RFC 7228에서 정의된 약 10킬로 바이트의 RAM과 약 100킬로 바이트 정도의 Flash 메모리를 장착하고 있는 Class 1 IoT 장치에 적용 가능한 안전하고 표준화된 펌웨어 업데이트 기법을 정의하는 것을 목표로 하고 있다[8]. 이 워킹 그룹에서 표준화하고자 하는 펌웨어 업데이트 기법은 다음과 같은 사항을 고려해야 한다.

(1) 펌웨어 업데이트 기법은 펌웨어 이미지가 배포되는 방식과는 무관해야 한다.
(2) 브로드캐스트 전달 방식에 친화적이어야 한다.
(3) 최신의 암호 방식을 사용할 수 있어야 한다.
(4) Rollback 공격을 방어할 수 있어야 한다.
(5) 전력이 끊기거나 업데이트를 인증하는데 실패하더라도 기기에 장애가 발생하지 않도록 높은 신뢰성이 제공되어야 한다.
(6) 작은 부트로더 환경에서도 동작할 수 있어야 한다.
(7) 기존의 펌웨어의 형식에 변화를 요구해서는 안된다.

그림 1은 펌웨어 생성자가 펌웨어 이미지를 생성하고 펌웨어 서버에 업로드하기 위한 통신 구조를 보여준다. 펌웨어 이미지와 (펌웨어 이미지에 대한 메타-데이터를 포함하는) 매니페스트(manifest)는 기기에 있는 통신자(communicator)를 이용하여 push 또는 pull의 방식으로 기기에게 배포된다. 기기 운용자는 상태 추적기 (status tracker)를 이용하여 업데이트 과정을 추적한다. 이를 통하여 기기 운용자는 어떤 기기가 업데이트를 수신하고 또한 어떤 기기는 업데이트를 보류하고 있는지에 대한 정보를 알 수 있다. 여기에서 펌웨어 이미지와 매니페스트는 종단간 암호 메커니즘을 이용하여 암호화된다. 기기는 펌웨어를 업데이트하기 전에 수신한 펌웨어 이미지와 매니페스트가 진본인지를 검증할 수 있어야 한다. 즉, 펌



〈그림 1. IETF suit 워킹 그룹에서 고려하는 네트워크 구조[8].〉

웨어 업데이트를 수신하기 전에 기기는 매니페스트에 대한 서명을 검증할 수 있어야 한다. 기기는 trust anchor를 이용하여 서명을 검증한다. 또한 매니페스트를 생성하고 서명하는 모든 엔터티들이 모두 동일한 권한을 갖지는 않는다. 기기는 매니페스트에 서명된 기관의 허가권에 따라 실행을 할 것인지를 결정할 수 있어야 한다. 또한 펌웨어 이미지의 기밀성을 유지하기 위하여 펌웨어 이미지 개발자(author)는 기기의 공개키와 인증서 또는 둘 사이에 미리 설정된 키를 가지고 있어야 한다. 다음은 개발자가 새로운 펌웨어를 펌웨어 서버에 업로드하고 관련 매니페스트를 만들어 기기에게 펌웨어 이미지를 배포하는 메시지 플로우의 예를 보여준다.

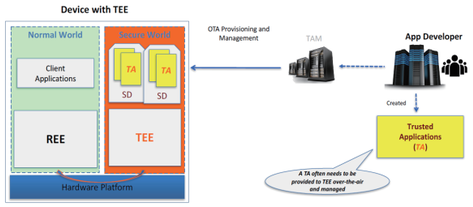
(1) 개발자는 펌웨어를 생성한 후 펌웨어 서버에 업로드한다. 또한 개발자는 매니페스트를 생성하여 서명한 후 펌웨어 서버에 업로드한다.
(2) 기기의 통신자는 펌웨어 서버에 새로운 매니페스트가 있는지를 질의하고, 펌웨어 서버는 매니페스트를 전송한다. 기기의 통신자는 수신한 매니메스트를 검증하고 만일 유효하다고 판단되면 펌웨어 서버에게 펌웨어 전송을 요청하고, 펌웨어 서버는 펌웨어를 전송한다.
(3) 기기의 통신자는 수신한 펌웨어를 검사한 후 유효하다고 판단되면 펌웨어를 저장한다. 그런 후에 기기에게 재부팅을 요청한다.
(4) 기기는 수신한 펌웨어를 검증한 후, 새로운 펌웨어를 실행하여 새로운 펌웨어로 부팅한다.

그런데, 기기의 통신자는 펌웨어 서버에 새로운 펌웨어가 업로드되어 있는지 어떻게 알 수 있는가? 기기의 통신자가 펌웨어 서버에 새로운 펌웨어 이미지가 있는지를 확인하는 클라이언트-기반 업데이트 방식과 상태-추적기의 제어 하에 펌웨어 업데이트의 존재 여부를 결정하는 서버-기반 업데이트 방식, 그리고 통신자가 기기에게 업데이트가 있다는 것을 알려주면, 기기가 가능한 시간에 펌웨어 서버에 접속하여 이미지

를 다운로드받는 하이브리드 방식 등 세 가지 방식이 있을 수 있다.

IETF suit 워킹 그룹에서 현재 개발 중인 표준 기술은 다음과 같다.

- A Firmware Update Architecture for Internet of Things Devices (draft-ietf-suit-architecture-01)
- Firmware Updates for Internet of Thins Devices – an Information Model for Manifests (draft-ietf-suit-information-model-01)

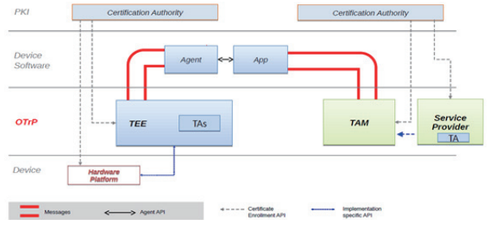


〈그림 2. IETF teep WG에서 고려하고 있는 네트워크 토폴로지[10].〉

III. IETF teep 워킹 그룹 표준화 동향

IETF teep 워킹 그룹은 98차 IETF 회의에 처음으로 BoF가 열린 후에 101차 IETF 회의에서 정식 워킹 그룹으로 활동을 시작하였다[9]. 이 워킹 그룹에서는 기기와 TEE, 그리고 TA 개발자 사이에 상호 신뢰를 구축하면서도 개방적으로 TEE 내에 TA를 설치 및 업데이트를 할 수 있는 프로토콜 개발을 목적으로 한다. 그림 2는 IETF teep 워킹 그룹에서 고려하고 있는 네트워크 토폴로지이다[10]. teep에서는 각각의 엔터티들이 어떻게 서로 간에 상호 신뢰를 구축하는 가가 중요한 문제라 할 수 있다. 이 위킹 그룹에서는 각 엔터티들이 상호 신뢰를 구축하기 위하여 공개키 기반 구조(Public key infrastructure: PKI)가 사용된다. 예를 들어, TEE는 TEE 내에 설치하고자 하는 (또는 업데이트하고자 하는) TA를 어떻게 신뢰할 것인가? TA 개발자는 자신의 비대칭키를 가지고 있어서, 자신의 개인키를 이용하여 TA를 서명하고 자신의 공개키에 대한 인증서를 첨부한다. 그러면 TEE는 TA의 개발자의 공개키에 대한 인증서를 신뢰할 수 있는가? TA 개발자의 인증서는 인증 기관(Certificate Authority: CA)이 발행한 인증서이다. 그러면 TA 개발자의 인증서를 발행한 인증 기관은 신뢰할 수 있는 인증 기관인가? TEE는 인증 기관에 대한 화이트리스트를 가지고 있어서, 이 화이트리스트에 속한 인증 기관으로부터 발행된 인증서 만을 신뢰한다. 즉, TA 개발자는 TEE가 가지고 있는 화이트리스트에 속하는 인증 기관으로부터 발급받은 인증서를 이용하여 TEE와 상호 신뢰를 구축할 수 있다.

실제로 teep 워킹 그룹에서는 TA 개발자는 TA를 앱스토어와 유사한 기능을 하는 TA 관리 장치인 Trusted Application Manager (TAM)에 업로드하며, TAM이 TA의 신뢰성을 책임진다. 즉, TEE는 TAM에 대한 CA의 화이트리스트를 가지고 있어서 이를 이용하여 TAM의 신뢰성을 확인한다. 그림 2에서 TA 개발자는 (1) REE에서 동작하는 클라이언트 응용과 (2) TEE에 설치하고자 하는 보안 기능을 제공하는 TA를 개발한다. 그런 후에 TA 개발자는 클라이언트 응용을 일반적인 앱스토어에 업로드하고 또한 TA를 TAM에 제공한다. TA 사용자는 앱스토어에서 클라이언트 응용을 다운로드하여 설치한다. 그러면 클라이언트 응용은 TAM에 접속하여 TEE에 설치하고자 하는 TA를 다운로드한다. 여기에서 TEE 자체는 기기의 외부와 통신하는 기능을 가지고 있지 않기 때문에, 클라이언트 응용이 TEE와 외부와의 통신을 담당한다. 그런데 TAM의 입장에서는 앞에서 언급한 것과 같이 ARM의 TrustZone이나 Intel의 SGX와 같이 서로 다른 형태를 가진 여러 가지의 TEE가 있을 수 있기 때문에 클라이언트 응용이 서로 다른 형태의 TEE와 통신하기 위하여 각각의 TEE에 따라 서로 다른 형태로 동작할 수 있도록 구현되기 보다는 공통의 인터페이스를 통하여 TEE와 메시지를 교환하는 것이 필요하다고 할 수 있다. 이러한 기능을 수행하기 위하여 teep 워킹 그룹에서는 에이전트라는 기능이 정의된다[11]. 에이전트는 클라이언트 응용을 통하여 TAM과 TEE 사이의 통신이 가능하도록 하며, 안드로이드 기반의 기기의 경우에 안드로이드 서비스 콜의 형태로 클라이언트 응용이 TA에 대한 질의 또는 설치와 같은 정보를 TEE에게 제공할 수 있는 인터페이스를 제공한다. 즉, 클라이언트 응용은 TAM과 통신하며, TAM으로부터 수신된 메시지를 에이전트를 이용하여 TEE에게 전달하고 또한 TEE가 에이전트를 통하여 전송하는 메시지를 TAM에게 전달한다. 다음 2 개의 에이전트의 서비스 콜이 클라이언트 응용을 위해 정의된다.



〈그림 3. 시스템 구성 요소 및 프로토콜의 동작 범위.〉

(1) ProcessOTrPMessage call

클라이언트 응용은 이 서비스 콜을 이용하여 TAM으로부터

수신한 요청 메시지를 기기 내의 TEE로 전달한다. 또한 TEE로부터 생성된 응답 메시지를 클라이언트 응용에게로 전달한다.

(2) GetTAInformation call

TA의 무결성 검사를 위하여 클라이언트 응용이 TA의 식별자와 공개키를 가지고 있는 경우에 TAM의 요청 없이 기존에 설치된 TA에 대한 정보를 이 서비스 콜을 이용하여 TEE에게 질의한다.

그림 3은 IETF teep 워킹 그룹에서 고려하고 있는 시스템 구성 요소와 프로토콜 동작 범위를 보여준다[12]. TAM과 TEE는 Open Trust Protocol (OTrP)를 이용하여 통신한다[11]. 여기에서 OTrP는 TAM과 TEE 간에 상호 신뢰를 구축하기 위한 메시지 프로토콜을 정의하며, JSON Web Encryption (JWE), JSON Web Signature (JWS), JSON Web Key (JWK)와 같은 IETF에서 정의된 종단간 보안 메커니즘을 이용한다. OTrP가 동작하기 위해서는 먼저 (스마트폰과 같은) 기기에 TEE가 구현되어 있어야 하며 또한 기기 고유의 공개키/개인키 쌍이 있어야 한다. 기기의 공개키/개인키 쌍을 "root of trust"라고 한다. OTrP 메시지는 이 메시지가 진본임을 확인할 수 있도록 서명되어야 한다. 여기에서 서명은 펌웨어의 개인키나 TEE의 개인키, 또는 TAM의 개인키와 같이 메시지를 생성하는 생성자의 개인키를 이용하여 이루어진다. OTrP에서 현재 정의된 메시지는 다음과 같다.

(1) GetDeviceState

TAM이 이 메시지를 이용하여 기기의 현재 상태를 질의하며, 기기의 TEE는 버전과 펌웨어 버전, TAM에 의해서 관리되는 TA 리스트 등에 대한 정보를 전송한다.

(2) CreateSD, UpdateSD, DeleteSD

TAM은 서비스 제공자를 위한 기기의 TEE가 보안 도메인 (Security Domain: SD)를 개설(create), 업데이트(update), 또는 삭제(delete)하시 위한 요청 메시지를 전송한다. TEE는 이 요청 메시지에 대한 결과를 (에이전트와 클라이언트 응용을 통하여) 응답 메시지를 이용하여 TAM에게 전송한다.

(3) InstallTA, UpateTA, DeleteTA

TAM은 서비스 제공자를 위한 SD 내에 TA를 설치(install), 업데이트(update), 또는 삭제(delete)하기 위한 요청 메시지를 전송하며, TEE은 이 요청 메시지에 대한 결과를 응답 메시지를 이용하여 TAM에게 전송한다.

IETF teep 워킹 그룹에서 현재 개발 중인 표준 기술은 다음과 같다.

- Trusted Execution Environment Provisioning (TEEP) Architecture (draft-ietf-teep-architecture-00)
- The Open Trust Protocol (OTrP) (draft-ietf-teep-opentrustprotocol-01)

IV. 결론

본 고에서는 IoT 기기에 보안 취약점이 발견되는 경우 사용자/개발자/운용자가 막대한 시간과 인력, 그리고 비용을 들여서 수동으로 펌웨어를 업데이트하는 대신에 펌웨어 개발자가 서버에 새로운 펌웨어를 등록하고, IoT 장비가 안전하게 자동으로 펌웨어를 업데이트할 수 있도록 하기 위하여 IETF suit 워킹 그룹에서 논의 중인 표준화 동향과, 스마트폰, IoT 기기 또는 클라우드 서버와 같은 장치의 메인 프로세서에 향상된 보안 기능이 제공되는 TEE가 구현되어 있는 경우에, TAM이라고 하는 일종의 앱스토어로부터 응용을 다운로드하고 이 응용을 TEE에 설치하고자 하는 경우에, TEE와 응용, 그리고 TAM 간에 상호 신뢰를 제공하면서도 개방화된 방법으로 메시지를 교환할 수 있도록 하기 위한 Open Trust Protocol(OTrP) 개발을 위하여 IETF teep 워킹 그룹에서 논의 중인 표준화 동향에 대해서 살펴보았다. Suit과 teep 워킹 그룹에서는 모두 원격으로 신뢰성 기반 하에서 응용을 설치하는 것과 같은 동일한 목적을 가지고 있는 것처럼 보이지만, suit은 시스템 부팅을 위한 펌웨어 업데이트를 위한 표준화를 목표로 하는데 반해 teep는 부팅 후에 "신뢰할 수 있는 응용" 설치에 초점을 맞추고 있다. 또한 suit은 (TEE이던 아니던) IoT 장비에의 코드 설치를 목표로 하고 있지만 teep은 (IoT 장비이건 아니건) TEE 내에의 코드 설치를 목표로 하고 있다. 또한 suit는 미리 설치된 코드의 업데이트에 초점이 맞추어 있지만, teep은 초기의 코드 설치라는 곳에 초점이 맞추어 있어서 적용 분야가 서로 다르다 할 수 있다. 그렇지만 이 2 개의 워킹 그룹에서도 상호 신뢰를 구축하는 방법은 동일하게 사용될 수 있어서 이를 위하여 상호간에 협력 방안이 논의되고 있다.

현재와 같이 모바일 금융과 IoT 장비의 사용이 확산되어 있는 상태에서는 개인 정보와 사생활 보호, 그리고 사회 인프라의 안전적인 운용을 위해서 좀 더 강력한 보안을 제공하는 것이 필수적이라 할 수 있다. IETF suit과 teep 워킹 그룹에서 개발 중인 표준화 결과물은 향후 엔터티들 간에 상호 신뢰를 구축하고 또한 향상된 보안을 제공할 것이라 생각한다.

Acknowledgement

이 논문은 2018년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. 2018-0-00846, 시간 민감형 트래픽을 위한 Gated 패킷 스케듈링 및 동적 큐 관리 기능을 갖는 프로그램 가능한 Open Platform TS-SDN 스위치 연구)

References

[1] "CCTV, 공유기가 인터넷을 다운시켰다. 미라이 봇넷 설명", ITWORLD, 2018. 3. 13, available at http://www.itworld.co.kr/news/108506.

[2] "거대 IoT 좀비 군단 형성 중… 미라이보다 20배 더 크다", 전자신문, 2017.10.25, available at http://www.etnews.com/20171025000162.

[3] B. Moran, M. Meriac and H. Tschofenig, "Firmware Updates for Internet of Things Devices", available at https://datatracker.ietf.org/meeting/100/materials/slides-100-suit-draft-moran-suit-architecture-and-draft-moran-suit-manifest-00.

[4] "ARM Security Technology - Building a Secure System using TrustZone Technology", available at http://infocenter.arm.com/help/topic/com.arm.doc.prd29-genc-009492c/PRD29-GENC-009492C_trustzone_security_whitepaper.pdf.

[5] Intel Software Guard Extensions, available at https://software.intel.com/sgx.

[6] M. Pei, H. Tschofenig, A. Atyeo and D. Liu, "Trusted Execution Environment Provisioning (TEEP) Architecture", draft-ietf-teep-architecture-00 (work in progress), July 2, 2018.

[7] https://datatracker.ietf.org/wg/suit/about/.

[8] B. Moran, M. Meriac, H. Tschofenig and D. Brown, "A Firmware Update Architecture for Internet of Things Devices", draft-ietf-suit-architecture-01 (work in progress), July 2, 2018.

[9] https://datatracker.ietf.org/wg/teep/about/.

[10] "TEEP BoF Problem Statement", available at https://datatracker.ietf.org/meeting/100/materials/slides-100-teep-problem-statement-01.

[11] M. Pei, A. Atyeo, N. cook. M. Yoo and H. Tschofenig, "The Open Trust Protocol (OTrP)", draft-ietf-teep-opentrustprotocol-01 (work in progress), July 2, 2018.

[12] H. Tschofenig, "Architecture", available at https://datatracker.ietf.org/meeting/101/materials/slides-101-teep-architecture-00.

Biographies



이재훈

1985년: 한양대학교 전자공학과 학사
1987년: 한국과학기술원 전기및전자공학과 석사
1995년: 한국과학기술원 전기및전자공학과 박사
1987년 3월 ~ 1990년 4월: 데이콤(현 LGU+) 연구원
1990년 3월 ~ 1999년 2월: 삼성전자 선임연구원
1990년 3월 ~ 현재: 동국대학교 정보통신공학과 교수
주요 관심분야: IP 이동성, 다중 액세스 프로토콜, 인터넷 프로토콜 등
e-mail: jaehwoon@dongguk.edu

IETF/IRTF ICNRG 기술 동향

홍정하, 유태완, 홍용근
Electronics and Telecommunications Research Institute (ETRI)

Abstract

정보 중심 네트워킹(Information-Centric Networking, ICN) 기술에 대한 표준은 2012년 4월부터 IETF/IRTF ICNRG에서 활발히 진행되고 있다. ICN은 정보 자체를 식별하도록 Name을 정의하고 Name 기반으로 라우팅과 포워딩을 지원하며, 캐싱 기능을 통해 인터넷의 코어에 부담 없이 효과적이며 빠른 형태의 전송 서비스 및 정보의 무결성을 지원해 주는 암호화 기법을 제공하는 새로운 형태의 통신 기술이다. 본 고에서는 현재 진행중인 IETF/IRTF ICNRG의 표준화 현황 및 ICN 관련 기술의 연구 개발 동향에 대해서 알아보고 향후 기술 발전 방향에 대해 논의 한다.

I. 서론

인터넷은 2020년경 200억개 이상의 디바이스들이 연결될 것으로 예측되고 있으며[1], 웹(World Wide Web)을 통해 수 많은 정보를 서로 공유하며 사용하기 위한 거대한 사회적 인프라와 같은 형태로 발전하고 있다. 이와 같은 인터넷의 급격한 발전은 확장성, 보안성, 이동성 등과 같은 현재 인터넷의 문제점을 야기 시켰고[2], 이를 해결하기 위하여 처음부터 인터넷 구조를 새롭게 설계하고자 하는 미래 인터넷(Future Internet)에 대한 연구가 2000년대 초부터 본격적으로 시작되었다.

이동 단말의 수가 급격히 증가하고, 이 단말을 활용해 다양한 정보를 수신 받는 구조로 사용자의 서비스 활용 방식이 달라짐에 따라, 현재의 사용자들은 더 이상 원하는 정보의 위치가 아니라 정보 자체에 집중하는 방식으로 정보 요청 방식이 변화되었다. 이러한 통신 패러다임의 변화에 따라서 통신 대상 정보에 네임(Name)을 부여하고 이 네임을 통해 정보의 위치와 독립적으로 정보를 획득하는 구조인 정보 중심 네트워킹(Information-Centric Networking, ICN)은 미래 인터넷의 대표적인 기술로 부각되었으며 현재 전 세계적으로 관련 연구가 진행되고 있다[3].

ICN 기술의 표준화를 위하여 short-term의 엔지니어링 이슈를 표준화 하는 인터넷 관련 국제표준화기구인 IETF(Internet Engineer Task Force)와 병렬적으로 구성되어 long-term 연구 이슈를 다루는 IRTF(Internet Research Task Force) 내 Information-Centric Networking Research Group (RG)이 2012년 4월 12일에 공식적으로 설립되었고, 2012년 7월에 열린 IETF 84차 회의에서 정식 1차 회의를 시작으로 표준화를 위한 연구 활동이 활발히 진행되고 있다. ICNRG는 표준화를 위한 연구 활동뿐 아니라 ICN 관련 프로젝트 및 기술 이슈 등을 논의하는 포럼으로도 활용되고 있다. 그래서, IETF/IRTF 회의가 개최되는 첫날인 일요일에 full day workshop 형식으로 ICNRG interim 회의가 항상 개최되고 있고, 여기에서는 기고서 기반의 ICN 기술 논의뿐만 아니라 현재 진행중인 ICN 관련 프로젝트들의 소개 및 논의도 함께 이루어지고 있다.

따라서, 본 고에서는 이러한 IETF/IRTF ICNRG의 표준화 현황 및 ICN 관련 기술의 연구 개발 동향에 대해서 알아보고 향후 기술 발전 방향에 대해 살펴볼 것이다.

II. ICN 기술 개요

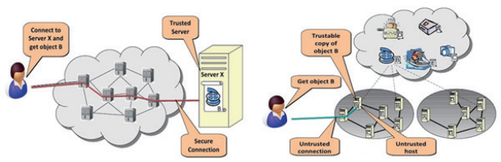
ICN은 현재 인터넷의 단대단(end-to-end) 통신 기법을 중심으로 한 호스트 기반 네트워킹과는 달리 실제 전달하고자 하는 정보를 가장 효과적으로 제공하기 위한 이름 기반 네트워킹 기술을 포함하고 있으며, 전달된 정보 자체의 무결성을 위한 신뢰 기법을 제안하고 있다[4].

호스트 중심 네트워킹과 ICN의 가장 대표적 차이점은 통신의 종단을 호스트에 두지 않고 정보 자체에 둔다는 점이다. 따라서 현재 인터넷과는 달리, ICN에서는 정보 자체를 식별하도록 Name을 정의하였고, 이 Name은 모든 네트워크 상에서 정보를 찾는 것뿐 아니라 기존의 라우팅과 포워딩에도 사용하게 된다. 이러한 특징은 자연스럽게 단대단 통신 세션을 유지하지 않고 한 홉을 건너 갈 때마다 정보를 찾고 다시 포워딩하는 절차를 수행하는 형태로 통신 형태가 바뀌게 되었다.

두 번째 핵심 패러다임의 변화는 전송 장비에서 이루어지는 캐싱(In-network caching) 기능이다. 이는 현재 인터넷의 단대단 통신 방법에서 실시간 및 대용량 미디어 서비스 등을 지원하기 위해 P2P 및 CDN(Content Delivery Networking) 등과 같은 OTT(Over The Top) 기반의 캐시 서버를 별도로 구축하고, 응용 서비스 차원의 분산된 미디어 서비스를 제공하는 형태를 탈피하여, 전송 장비 자체에 네트워크 상의 정보를 직접 캐싱할 수 있는 기능을 탑재한 것이다. 이러한 네트워크 상의 캐싱 기능은 네트워크 상의 미디어 정보들이 자연스럽게 사용자 주변의 장비에 저장 되도록 하며, 결과적으로 인터넷의 코어에 부담이 없이 효과적이며 빠른 형태의 전송 서비스를 제공할 수 있는 토대가 되게 하는 것이다.

세 번째로, 호스트 중심 네트워킹과 달리 ICN에서는 정보들이 네트워크 상에 분산되어 저장되기 때문에 그에 따른 정보의 무결성을 지원해주는 암호화 기법을 제공하는 것이다. 기존 인터넷은 단대단 통신이기 때문에 두 호스트 간 채널의 암호화를 통해 보안을 제공하며, 이는 통신 과정 중 정보의 탈취나 변조가 불가능 하도록 하는 장점을 지니고 있으나, 양 끝단에서 암호화 절차의 복잡성에 의해 IoT 환경에는 적합하지 않으며 네트워크 관리 등을 위해 중간 노드에서 패킷 처리 등이 불가능 하다는 문제점을 가지고 있다. 반면, ICN에서는 중간 어느 노드에서도 정보를 획득할 수 있으므로 획득한 정보가 인증된 게시자(publisher)로부터 생성된 것인지에 대한 암호 정보를 기본으로 탑재하고 있어, 사용자(consumer)는 자신이 원하는 정보의 이름과 함께 유효한 게시자 정보(공개키)를 이용하여 정보의 무결성을 체크할 수 있다.

위의 대표적인 특징들은 모든 ICN 구조 및 관련 연구들에서 공통적으로 보이는 것이며, 그림 1은 호스트 중심 네트워킹과 정보 중심 네트워킹의 이러한 대표적 차이점을 나타낸 그림이다[5].



〈그림 1. 통신 모델 : 호스트 중심 네트워킹 대 정보 중심 네트워킹 [5]〉

이 밖에 각각의 ICN 구조에서는 평면적 이름 (flat name) 지원, 이동성 지원, IP 서비스와의 호환성 지원 등과 같은 차별성을 지니고 개발되고 있으며 최근 IoT 및 5G 등의 새로운 네트워킹 요구사항에 맞춘 연구개발 또한 활발히 이루어지고 있다.

III. IETF/IRTF ICNRG 현황

IETF/IRTF ICNRG의 현재까지의 표준 제정 문서 및 그룹 문서에 대한 현황은 IETF Datatracker (https://datatracker.ietf.org/rg/icnrg/documents)와 ICNRG wiki (https://trac.ietf.org/trac/irtf/wiki/icnrg)에서 모두 관리되고 있으나, ICNRG wiki의 정보가 더 빠르게 업데이트 되고 있어서 IETF Datatracker 보다 더 정확한 최신 정보라 할 수 있다.

3.1 표준 제정 문서 (RFCs)

현재까지의 표준 제정 문서로는 다음과 같이 Informational RFC만 4건이 있다.

- RFC 7476 Information-Centric Networking: Baseline (draft-irtf-icnrg-scenarios) [6]
  본 문서는 서로 다른 접근 방식 간의 성능 비교를 가능하게 하는 reference baseline 시나리오를 정의하고 있다.
- RFC 7927 Information-Centric Networking (ICN) Research Challenges (draft-irtf-icnrg-challenges) [7]
  본 문서에서는 naming, security, routing, system scalability, mobility management, wireless networking, transport services, in-network caching, and network management와 같은 ICN research challenges에 대하여 설명하고 있다.
- RFC 7933 Adaptive Video Streaming over ICN (draft-irtf-icrng-videostreaming) [8]
  본 문서에서는 ICN에서의 비디오 스트리밍에 대한 주요 이슈들을 설명하고 현재 비디오 스트리밍을 ICN 아키텍처에 맞게 어떻게 수정해야 하는지에 대하여 설명하고 있다.
- RFC 7945 Information-Centric Networking: Evaluation and Security Considerations (draft-irtf-icnrg-evaluation-methodology) [9]
  본 문서는 ICN evaluation과 관련하여 여러 가지 고려 사항을 제시하고 ICN이 네트워크 보안에 미치는 영향에 대해 설명하고 있다. 또한 현재 ICN 연구자가 사용할 수 있는 evaluation tool들을 제공하고 방법론(methodology) 및 메트릭(metrics)에 대해서도 제안하고 있다.

3.2 ICNRG 문서 (Current work items)

현재 다음과 같이 10건의 문서가 ICNRG의 그룹 문서로 채택되어 표준 제정을 위하여 논의가 진행되고 있다.

- CCNx Semantics (draft-irtf-icnrg-ccnxsemantics-09) [10]
  본 문서는 CCNx (Content Centric Networking) 아키텍처의 핵심 개념을 설명하고 Interest 및 Content Object라는 두 가지 메시지를 기반으로 네트워크 프로토콜을 제시하고 있다. 이러한 메시지 내의 필수 및 옵션 필드를 지정하고 메시지의 동작과 해석을 설명하고 있다. 또한 이 프로토콜은 InterestReturn이라는 제어 메시지를 사용해서

어떤 오류로 인해 현재의 시스템이 Interest 메시지를 처리할 수 없음을 이전 홉에 알릴 수 있음을 설명하고 있다. 본 문서는 현재 Experimental RFC로의 제정을 위하여 IRSG(Internet Research Steering Group) Review 중이다.

- CCNx Messages in TLV Format (draft-irtf-icnrg-ccnxmessages-08) [11]
  본 문서에서는 CCNx message의 인코딩을 TLV 패킷 형식으로 정의하고 있고, CCNx 메시지의 semantics는 인코딩과 독립적인 CCNx Semantics 문서를 따르고 있다. 위의 문서, CCNx Semantics와 같이 현재 Experimental RFC로의 제정을 위하여 IRSG Review 중이다.
- Research Directions for Using ICN in Disaster Scenarios (draft-irtf-icnrg-disaster-03) [12]
  본 문서에서는 자연 또는 인간이 생성한 대규모 재난에 대처하기 위해 ICN을 적용하는 것과 관련된 ICN 연구 방향을 설명하고 있다.
- File-Like ICN Collection (FLIC) (draft-irtf-icnrg-flic-01) [13]
  본 문서에서는 하나의 ICN 데이터 세트를 큰 파일과 유사한 ICN 컬렉션 (FLIC)으로 구성하기 위한 “index table” approach에 대하여 설명하고 있다. 이 컬렉션의 핵심에는 컬렉션의 루트 노드 역할을 하는 manifest가 있고 이 manifest는 포인터를 갖는 인덱스 테이블을 포함하며, 각 포인터는 최종 데이터 블록 또는 다른 인덱스 테이블 노드를 가리키는 해시 값이다.
  본 문서의 저자들이 현재 계속해서 문서를 업데이트하기 어려운 상황이어서, 문서 개발을 대신해줄 저자를 찾고 있는 중이다.
- Information-Centric Networking (ICN): CCN and NDN Terminology (draft-irtf-icnrg-terminology-00) [14]
  본 문서에서는 NDN 및 CCN의 개념을 설명하는데 사용된 용어 및 정의에 대한 개요를 제공하고 있다. NDN 및 CCN 비전의 일부가 아닌 다른 ICN 아키텍처들은 본 문서의 scope에 해당하지 않는다.
- Deployment Considerations for Information-Centric Networking (ICN) (draft-irtf-icnrg-deployment-guidelines-04) [15]
  본 문서에서는 ICN 기술이 배포(deploy) 될 때 고려되어야 할 사항들을 제공하고 있다. 먼저, 키 오버레이 및 언더레이 방식을 포함하여 ICN의 주요 배포 구성이 설명되고, 그런 다음 네트워크 및 응용 프로그램 마이그레이션과 같은 주요 실제 문제를 해결하기 위해 제안된 배포 마이그레이션 경로가 설명된다. 다음으로, 몇몇 ICN 프로젝트에서 ICN의 배포를 위하여 실제로 어떻게 하였는지에 대하여 설명하고, 마지막으로 향후 표준화가 필요한 프로토콜 영역에 대하여 설명하고 있다.
  본 문서는 현재 Informational RFC로의 제정을 위하여 RG last call을 진행 중이다.
- Native Deployment of ICN in LTE, 4G Mobile Networks (draft-irtf-icnrg-icn-lte-4g-01) [16]
  본 문서에서는 효율적인 멀티미디어 데이터 전송이 가능한 ICN 기술을 3GPP protocol stack에 사용을 해서 cellular mobile networks에 실제로 배포(deploy) 해서 ICN의 장점을 활용하여 네트워크를 최적화 하는 것에 대하여 설명하고 있다.
- Design Considerations for Applying ICN to IoT (draft-irtf-icnrg-icniot-01) [17]
  본 문서에서는 일반적인 IoT 요구사항에 대해서 설명하고, ICN 아키텍처를 기반으로 ICN-IoT 프레임 워크를 실현하기 위한 challenges와 design에 대해서 다루고 있다. 이러한 ICN-IoT는 ICN의 두드러진 기능을 활용하고 네이밍, 보안, 이동성, 효율적인 콘텐츠 및 서비스 제공 지원도 활용하고 있다.
- Requirements for Name Resolution Service in ICN (draft-jhong-icnrg-nrs-requirements-04) [18]
  본 문서에서는 ICN 구조에서 NRS(Name Resolution Service)의 기능을 정의하고 NRS가 왜 필요한지에 대한 motivation을 설명하며, NRS를 디자인할 때 고려되어야 할 요구사항들에 대해서 설명하고 있다.
  본 문서는 IETF 102차 회의 후 ICNRG 메일링 리스트를 통하여 2018년 9월에 그룹 문서로 채택된 것으로써 IETF Datatracker에 곧 업데이트 될 예정이다.
- Architectural Considerations of ICN using Name Resolution Service (draft-hong-icnrg-icnnrs-01) [19]
  본 문서에서는 네임 기반 라우팅을 하는 기본적인 ICN 구조에서 NRS를 사용하고자 할 때 발생할 수 있는 ICN의 구조적 변화와 고려되어야 할 사항들에 대해서 설명하고 있다.
  본 문서도 위의 문서와 마찬가지로 2018년 9월에 채택된 것으로써 IETF Datatracker에 곧 업데이트 될 예정이다.

3.3 개인 기고서 (Internet-Drafts)

현재 활발히 논의 중인 개인 기고서 중심으로 간략히 소개한다.

- MAP-Me: Managing Anchorless Mobility in Content Centric Networking (draft-irtf-icnrg-mapme-01) [20]
  본 문서는 시스코에서 개발 중인 것으로써 대기 시간에 민감한 애플리케이션 지원을 위하여 CCN (Content Centric Networking) 및 NDN (Named Data Networking) 아키텍처에서의 콘텐트 생산자(producer)의 마이크로 이동성을 관리하기 위한 anchor-less 솔루션 인 MAP-Me에 대해 설명하고 있다. MAP-Me는 생산자의 이동에 의해 trigger되고, sequence number를 사용해서 생산자가 위치를 옮길 때 마다 1씩 증가시켜서 FIB 업데이트 메시지를 보내서 빠르게 FIB를 업데이트 해서 생산자를 빨리 찾을 수 있도록 해주는 기술이다.
  본 문서는 현재 IETF Datatracker 상에 ICNRG 문서로 업로드 되어 있지만 실제로 그룹 문서로 채택된 것은 아니고 시스템의 오류라고 ICNRG 의장들이 발표를 하였다.
- ICN Adaptation to LowPAN Networks (ICN LoWPAN) (draft-gundogan-icnrg-ccnlowpan-02) [21]
  본 문서는 유럽 세개의 대학교에서 공동으로 개발 중인 것으로써 IEEE 802.15.4 LoWPAN 네트워크에서의 CCNx 및 NDN 용 convergence 계층을 정의하고 있다. 새로운 프레임 형식은 CCNx 및 NDN 패킷을 IEEE 802.15.4의 작은 MTU 크기에 맞게 줄이는 것이다. LoWPAN 기술과의 호환성을 지원하기 위해 6LoWPAN이 제공하는 dispatch 방식을 CCNx 및 NDN의 새로운 dispatch 유형을 포함하도록 확장하였다.
- ICN Traceroute Protocol Specification (draft-mastorakis-icnrg-icntraceroute-03) [22]
  본 문서에서는 CCN 또는 NDN을 기반으로 하는 ICN 네트워크에서 traceroute에 상응하는 프로토콜 메커니즘을 설명하고 있다. traceroute 요청을 발생시키고 traceroute 응답을 처리하는 ICN traceroute 클라이언트 응용 프로그램에 대해서도 설명하고 있다.
- ICN Ping Protocol Specification (draft-mastorakis-icnrg-icnping-03) [23]
  본 문서에서는 ICN 네트워크에서의 ping에 해당하는 프로토콜 메커니즘을 제안하고 있다. 에코 요청을 발생시키고 에코 응답을 처리하는 IP ping과 비슷한 ICN ping 클라이언트 응용 프로그램에 대해서도 설명하고 있다.
- Enabling ICN in 3GPP’s 5G NextGen Core Architecture (draft-ravi-icnrg-5gc-icn-02) [24]
  본 문서는 Huawei, Cisco, InterDigital, CableLabs에서 공동으로 개발 중인 것으로써, 3GPP에서의 5G core next gen architecture (5GC)에 대한 간략한 설명과 함께 ICN의 패킷 데이터 unit (PDU) sessions을 지원하기 위한 5GC의 제어 및 사용자 확장을 설명하고 있다. 모바일 에지 컴퓨팅 및 이동성 지원에 대한 서비스 시나리오를 기반으로 5GC에서 ICN이 어떻게 적용되는 지에 대한 설명도 포함하고 있다.
- CCNinfo: Discovering Content and Network Information in Content-Centric Networks (draft-asaeda-icnrg-ccninfo-01) [25]
  본 문서는 일본에서 개발 중인 것으로써 CCN의 네트워크 토폴로지 및 네트워크 내 캐시에 대한 정보를 검색하는 "CCNinfo”라는 메커니즘을 설명하고 있다. CCNinfo는 name prefix, device name, and function/application 별 CCN 라우팅 경로 정보, 콘텐트 전달자(forwarder)와 소비자(consumer) 간의 왕복 시간 (Round-Trip Time, RTT), name prefix 별 네트워크 내 캐시 상태를 조사하고 CCN 및 NDN와 같은 다른 프로토콜을 지원하는 게이트웨이도 찾아준다.
- Consideration on Applying ICN to Edge Computing (draft-wang-icnrg-icn-edge-01) [26]
  본 문서는 China mobile에서 개발 중인 것으로써 Edge Computing에 ICN 기술을 적용하는 이유와 기회를 분석하고 Edge Computing에 ICN을 적용할 때 고려해야할 사항들에 대해서 설명하고 있다.
- Internet Protocol Tunneling over Content Centric Mobile Networks (draft-white-icnrg-ipoc-01) [27]
  본 문서는 CableLabs에서 주도적으로 개발 중인 것으로써 CCN 또는 NDN에서 인터넷 프로토콜 트래픽의 터널링을 가능하게 하는 프로토콜에 대해 설명하고 있다. 이 프로토콜은 LTE 네트워크의 Evolved Packet Core(LTE-EPC)에서 사용하는 GPRS 터널링 프로토콜 (GTP)과 같은 IP over IP 터널링을 사용하는 모바일 네트워크 용 IP mobility plane을 제공하는 것을 target use case로 설명하고 있다. 이 프로토콜은 CCN 또는 NDN이 제공하는 built-in 이동성 지원 기능을 활용하여 LTE-EPC와 동등한 성능, 동일한 효율성을 보이고 구현 및 프로토콜의 복잡성을 크게 낮추어 준다.

Ⅳ. ICN 기술 관련 연구 동향

ICN 연구개발은 2000년대 초반의 경우 기존 클라이언트 서버 모델을 탈피하고 정보를 공유하기 위한 네트워킹 기술을 중심으로 진행되었고, 2000년대 중반에 접어들면서 호스트 중심의 네트워킹 컨셉을 완전히 버린 구조로 개발되기 시작했다. 최근엔 IoT를 위한 새로운 네트워킹 기술과 5G 네트워킹 기술 등에 대한 새로운 요구 사항들과 더불어 실제적인 IP 인터넷과의 연동, ICN을 위한 인증된 프로그래밍 코드, 실질적인 서비스를 통한 개발 등이 진행되고 있다.

제록스 팍(Xerox PARC)은 지난 10년 동안 ICN 기술의 대

표적인 구조로 알려진 CCNx를 개발해 왔고, 시스코가 2017년 2월에 이 기술을 인수했다[28]. 이후 시스코는 이 기술을 'Community ICN(CICN)'이라고 불리는 리눅스 재단 FD.IO 커뮤니티 내 오픈 소스 프로젝트에 통합해서 PARC로부터 인수한 CCN 소프트웨어를 포함해 시스코 ICN 소프트웨어까지 모두 오픈하였다[29].

시스코는 ICN Name을 IP 주소에 인코딩함으로써 IP 내에 ICN 배치를 가능하게 하는 'hICN(Hybrid ICN)'을 개발했다. hICN은 IPv4- 또는 IPv6-RFC 호환 패킷 포맷을 지원하며 표준 IP 네트워킹 장비와의 투명한 상호 연결을 보장한다. 기존 IP 인프라에 ICN 기술 삽입을 간소화하고 기존 IP 트래픽과의 공존도 가능하다는 것이 hICN의 주요 특징이다[30].

버라이즌은 자사 연구소에서 hICN 기술을 성공적으로 테스트 완료했다고 2018년 3월에 발표하였다[31]. 버라이즌은 "hICN은 5G 환경에서 불안정한 이동성을 보완하고, 네이티브(Native) 다중 경로 지원을 통한 액세스 이동, 통합된 유니캐스트/멀티캐스트, 엣지(Edge) 내장 캐싱(Caching)/처리 능력 및 유연한 객체 기반 보안 등 다양한 이점을 제공한다"라고 말했다. 버라이즌은 이번 테스트에서는 다중 액세스를 통한 향상된 속도 적응과 동적 패킷 세분화 로드 밸런싱 등의 성능을 확인했고 높은 처리량과 낮은 지연 속도가 필요한 AR/VR 및 인텔리전트 비디오 등의 애플리케이션에서 이 기술을 활용할 수 있을 것으로 기대하고 있다. 버라이즌의 기술 아키텍처 및 계획 전무이사 아닐 건투팔리는 "정확히 예측하기는 어렵지만 1~2년 이내에 hICN 기반 서비스가 실현될 수도 있다"라고 말했다.

미국의 NSF를 통해 개발되고 있는 NDN 역시 오픈 소스 개발과 함께 IoT 관련 실질적인 서비스를 개발하는 등 NDN Community를 토대로 적극적인 기술 확산에 힘쓰고 있다[32].

유럽의 경우 ICN 기술을 개발한 결과를 새로운 POINT(IP over ICN-the Better IP) 프로젝트에 연동시켜 서비스를 가능하도록 하는 구조 (IP over ICN)를 개발하고 있다[33].

일본 역시 유럽과 함께 GreenICN 이라는 프로젝트를 통해 재난상황에서의 구조 등을 위한 통신 기법에 대한 연구 프로젝트를 수행 중이다[34].

최근 미국의 NSF와 인텔은 IoT와 Wireless Edge computing(WEN) 기술을 결합해 5G를 위한 새로운 무선 구간 네트워킹을 ICN 기술을 통해 구축하려는 연구인 ICN-WEN을 시작했다[35].

V. 결론 및 시사점

본 고에서는 통신 패러다임의 변화에 따라서 통신 대상 정보에 Name을 부여하고 이 Name을 통해 위치가 아닌 정보 자체를 획득하는 구조인 정보 중심 네트워킹 (Information-Centric Networking, ICN)의 기술에 대해서 간략히 설명하였고, ICN 기술의 표준화가 진행 중인 IETF/IRTF ICNRG의 표준화 현황 및 ICN 관련 기술의 연구 개발 동향에 대해서 살펴보았다.

현재 인터넷의 단대단 통신 기법을 중심으로 한 호스트 기반 네트워킹과는 달리 실제 전달하고자 하는 정보를 가장 효과적으로 제공하기 위한 네트워킹 기법인 ICN 기술은 2000년대 초반에는 IP 인터넷 기반으로 연구가 진행되었다가 2000년대 중반부터 일어났던 미래 인터넷 연구 개발의 시기와 맞물려 IP 기반 인터넷과 다른 미래 인터넷 구조로 개발이 되어 왔었다. 국내뿐만 아니라 해외에서도 많은 연구소와 학계에서 정보 중심 네트워킹에 대한 연구가 제안되고 개발되었으나 실제 상용 서비스와 네트워킹 기술로써 활용되지 못하였다가, 2010년 중반부터 IoT 및 5G에 대한 요구사항을 만족시키기 위한 기술로서 ICN 기술이 다시 많은 관심을 받고 있다.

그러나 국외의 ICN 연구개발 현황과 달리 국내의 ICN 연구개발 현황에 대해서는 표면적으로 잘 드러나지 않고 있다. 본 고의 저자들은 2014년 7월에 개최된 IETF 90차 회의부터 IETF/IRTF ICNRG에서 Name Resolution Service (NRS) 기술에 대한 기고서를 개발해 왔고, 최근 2018년 9월에 IETF/IRTF ICNRG 문서로 NRS 관련 기고서 2건이 채택되었으며, 이 문서들의 표준 제정을 위하여 앞으로도 계속해서 노력할 계획이다.

현재의 ICN 기술은 단순히 새로운 개념을 제안하는 수준의 기술이 아니라 IoT 및 5G와 같은 실제 분야에 적용 가능한 실질적 서비스를 개발하는 등 ICN 기술의 상용화를 멀지 않은 미래에 앞두고 있다. 국내에서도 예전 미래 인터넷을 연구했던 경험과 노하우를 바탕으로 ICN 기술 개발 현황에 관심을 가지고 자체적인 핵심 기술 확보를 위하여 노력한다면 충분히 국제 연구 개발 수준을 따라 잡을 수 있을 것으로 전망한다.

Acknowledgement

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 국가과학기술연구회 융합연구단 사업(No. CRC-15-05-ETRI)의 지원을 받아 수행된 연구임

References

[1] Gartner's news report for 2015, Garner Inc., https://www.gartner.com/newsroom/id/3165317

[2] D. Meyer et al, "Report from the IAB Workshop on Routing and Addressing," RFC 4984, September 2007

[3] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A Survey of Information-Centric Networking Research," IEEE Communications Surveys and Tutorials, Vol. 16, No. 2, 2014, pp. 1024-1049.

[4] 유태완 외, "ICN 기반 IoT 기술 개발 동향," IITP 주간기술동향, 1799호, 기획시리즈, 2017, pp. 2-15.

[5] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric," IEEE Communications Magazine, Vol. 50, No. 7, 2012, pp. 26-36.

[6] K. Pentikousis et al, "Information-Centric Networking: Baseline Scenarios," RFC 7476, March 2015. [http://www.rfc-editor.org/info/rfc7476]

[7] D. Kutscher et al, "Information-Centric Networking (ICN) Research Challenges," RFC 7927, July 2016. [https://www.rfc-editor.org/info/rfc7927]

[8] C. Westphal et al, "Adaptive Video Streaming over Information-Centric Networking (ICN)," RFC 7933, August 2016. [https://www.rfc-editor.org/info/rfc7933]

[9] K. Pentikousis et al, "Information-Centric Networking: Evaluation and Security Considerations," RFC 7945, September 2016. [https://www.rfc-editor.org/info/rfc7945]

[10] M. Mosko et al, "CCNx Semantics," draft-irtf-icnrg-ccnxsemantics-09, June 2018.

[11] M. Mosko et al, "CCNx Messages in TLV Format," draft-irtf-icnrg-ccnxmessages-08, July 2018.

[12] J. Seedorf et al, "Research Directions for Using ICN in Disaster Scenarios," draft-irtf-icnrg-disaster-03, February 2018.

[13] C. Tschudin and C. Wood, "File-Like ICN Collection (FLIC)," draft-irtf-icnrg-flic-01, December 2017.

[14] B. Wissingh et al, "Information-Centric Networking (ICN): CCN and NDN Terminology," draft-irtf-icnrg-terminology-00, December 2017.

[15] A. Rahman et al, "Deployment Considerations for Information-Centric Networking (ICN)," draft-irtf-icnrg-deployment-guidelines-04, September 2018.

[16] P. Suthar et al, "Native Deployment of ICN in LTE, 4G Mobile Networks," draft-irtf-icnrg-icn-lte-4g-01, July 2018.

[17] R. Ravindran et al, "Design Considerations for Applying ICN to IoT," draft-irtf-icnrg-icniot-01, February 2018.

[18] J. Hong et al, "Requirements for Name Resolution Service in ICN," draft-jhong-icnrg-nrs-requirements-04, July 2018.

[19] J. Hong et al, "Architectural Considerations of ICN using Name Resolution Service," draft-hong-icnrg-icnnrs-01, July 2018.

[20] J. Auge et al, "MAP-Me : Managing Anchorless Mobility in Content Centric Networking," draft-irtf-icnrg-mapme-01, July 2018.

[21] C. Gundogan et al, "ICN Adaptation to LowPAN Networks (ICN LoWPAN)," draft-gundogan-icnrg-ccnlowpan-02, July 2018.

[22] S. Mastorakis et al, "ICN Traceroute Protocol Specification," draft-mastorakis-icnrg-icntraceroute-03, August 2018.

[23] S. Mastorakisl et al, "CN Ping Protocol Specification," draft-mastorakis-icnrg-icnping-03, August 2018.

[24] R. Ravindran et al, "Enabling ICN in 3GPP's 5G NextGen Core Architecture," draft-ravi-icnrg-5gc-icn-02, July 2018.

[25] H. Asaeda and X. Shao, "CCNinfo: Discovering Content and Network Information in Content-Centric Networks," draft-asaeda-icnrg-ccninfo-01, June 2018.

[26] L. Wang and L. Geng, "Consideration on Applying ICN to Edge Computing," draft-wang-icnrg-icn-edge-01, July 2018.

[27] G. Whitel et al, "Internet Protocol Tunneling over Content Centric Mobile Networks," draft-white-icnrg-ipoc-01, June 2018.

[28] Cisco Blogs, Cisco Announces Important Steps toward Adoption of Information-Centric Networking, http://blogs.cisco.com/sp/cisco-announces-important-steps-toward-adoption-of-information-centric-networking.

[29] FD.io CICN project. [Online]. Available: https://wiki.fd.io/view/Cicn.

[30] "Mobile Video Delivery with Hybrid ICN: IP-integrated ICN solution for 5G," Cisco white paper.

[31] 기사, "Verizon Completes Successful Test of Cisco's Hybrid Information-Centric Networking in

its Network," https://globenewswire.com/news-release/2018/03/16/1441592/0/en/Verizon-Completes-Successful-Test-of-Cisco-s-Hybrid-Information-Centric-Networking-in-its-Network.html
[32] NSF Named Data Networking project. http://www.named-data.net/
[33] HORIZON 2020 POINT project. https://www.point-h2020.eu/
[34] FP7 GreenICN project. http://www.greenicn.org/
[35] NSF ICN-WEN project. http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=505310

Biographies



홍정하

1999년: 고려대학교 수학과 이학사
2001년: 고려대학교 수학과 이학석사
2006년: University of Missouri – Kansas City, Computer Science 석사
2010년: University of Missouri – Kansas City, Telecommunications Networks and Computer Networks 박사
2010년 ~ 현재: 한국전자통신연구원 선임연구원
주요 관심분야: ICN, Internet architecture design, Queueing system, 네트워크 성능 분석
e-mail: jhong@etri.re.kr



유태완

2001년: 전북대학교 컴퓨터과학과 학사
2004년: 서울대학교 컴퓨터공학부 석사
2010년: 서울대학교 컴퓨터공학부 박사수료
2004년 ~ 2007년: 한국전자통신연구원 연구원
2011년 ~ 현재: 한국전자통신연구원 선임연구원
주요 관심분야: 미래인터넷, 이동성, ICN, Edge computing
E-mail: twyou@etri.re.kr



홍용근

1997년: 경북대학교 컴퓨터공학과 공학사
1999년: 경북대학교 컴퓨터공학과 공학석사
2013년: 경북대학교 컴퓨터공학과 공학박사
1998년 ~ 2001년: MJL Technology 근무
2001년 ~ 현재: 한국전자통신연구원 지능형IoE네트워크 연구실 실장
2015년 ~ 2016년: 한국전자통신연구원 전문위원
주요 관심분야: IPv6, Mobile IP, M2M/IoT, 네트워크 머신러닝
e-mail: yghong@etri.re.kr

IETF 및 IRTF에서의 인터넷 효율성 향상을 위한 표준화 연구 동향

김평수
한국산업기술대학교 전자공학부

Abstract

본 고에서는 급격한 데이터 트래픽 증가에 따른 인터넷 성능을 향상시키기 위해 더욱 정교한 방법을 찾고 해결하기 위한 최근 연구 동향을 살피고자 한다. 연구 동향은 인터넷 전 분야의 표준화 기술을 다루고 있는 세계적인 국제표준화 기구 IETF(Internet Engineer Task Force)는 물론 IETF 표준화 사전연구기구인 IRTF(Internet Research Task Force) 기반으로 한다. IETF 및 IRTF를 기반으로 성능 및 효율성 향상을 다루는 주요 기술들의 최신 동향을 워킹그룹(Working Group, WG) 및 연구그룹(Research Group, RG)별로 정리하여 기술한다.

I. 서론

2017년 6월 글로벌 네트워크 장비 업체 시스코가 8일(현지시각) 글로벌 인터넷 트래픽 예상 보고서를 공개했다[1]. 2016년 글로벌 인터넷 트래픽은 96엑사바이트(960억 기가바이트)이지만 5년 뒤인 2021년에는 278엑사바이트(2,780억 기가바이트)로 3배 가량 증가할 것으로 예측했다. 이처럼 빠를 증가의 요인으로는 동영상 콘텐츠를 꼽았다. 인터넷 속도가 안정적으로 빨라지고 고해상도 디스플레이 보급이 가속화 되면서 고화질 영상 수요가 증가할 것이라는 전망이다. 고화질 영상은 그만큼 많은 용량이 필요하다. 넷플릭스, 유튜브, 인터넷 TV 등의 동영상 콘텐츠가 전체 트래픽에서 차지하는 비율은 2016년 73%지만 2021년에는 82%로 더 많은 비중을 차지할 것으로 예상했다.

하지만, 이러한 급격한 데이터 트래픽 증가 속도에 모바일 네트워크 접속 속도가 미치지 못할 것으로 예상되기 때문에 사용자가 느끼는 네트워크 속도에 대한 만족도는 점차 하락할 수 있다[2]. 국제표준화 기구인 IETF(Internet Engineering Task Force)의 발표에 따르면 인터넷 트래픽의 약 90% 이상이 TCP를 이용하여 전송되고 있다[2]. 지난 수십 년간 많은 연구 개발을 통해 TCP/IP의 속도를 향상시키기 위한 다양한 노력이 진행되었으며, 주로 TCP가 트래픽 혼잡에 대응하는 방법에 중점을 두어왔다.

본 고에서는 급격한 데이터 트래픽 증가에 따른 인터넷 성능을 향상시키기 위해 더욱 정교한 방법을 찾고 해결하기 위한 최근 연구 동향을 살피고자 한다. 연구 동향은 인터넷 전 분야의 표준화 기술을 다루고 있는 세계적인 국제표준화 기구 IETF는 물론 IETF 표준화 사전연구기구인 IRTF(Internet Research Task Force) 기반으로 한다. IETF 워킹그룹(Working Group: WG)의 경우 QUIC WG[3][4], MPTCP WG[13]-[15], IPPM WG[16], LMAP WG[16], RMCAT WG[17], TCPM WG[18] 등, IRTF 연구그룹(Research Group: RG)의 경우 MAP RG[5][6], ICC RG[7]-[10], NWC RG[11][12] 등을 기반으로 성능 및 효율성 향상을 다루는 주요 기술들의 최신 표준화 동향을 소개한다.

II. 인터넷 효율성 향상을 위한 표준화 연구 동향

2.1 IETF QUIC WG

최근 구글에서 연구 개발하고 있는 QUIC(Quick UDP Internet Connections)은 상당히 가볍고 성능과 보안성을 모두 고려해서 설계되었으며, 암호화된 전송을 통해 다중화된 스트림을 제공하는 UDP(User Datagram Protocol) 기반 전송 프로토콜이다[3][4]. QUIC는 독립적인 TCP 연결과 거의 같지만 대기 시간이 훨씬 단축되고 스트림 다중화 지원이 향상된다.

QUIC 기술에 대한 표준화는 IETF 전송 분야 내 QUIC WG에서 다루어지고 있다. 현재 QUIC 기술은 구글을 통해 실제로 서비스 되고 있지만, 표준화가 완료되지 않아 현재까지는 RFC로 출판되지는 않았다. 따라서, 프로토콜 이해를 위해서는 QUIC WG의 첫 번째 WG Draft 문서인 "QUIC: A UDP-Based Multiplexed and Secure Transport(draft-ietf-quic-transport-13)"를 참조하면 된다. 지난 2018년 7월 캐나다 몬트리올에서 개최되었던 102차 IETF 총회에서는 13번 문서에서 새롭게 추가되거나 변경된 이슈들에 대한 논의가 이루어졌다. 그 외에, QUIC 기술의 적용 및 확산 가능

성에 대한 문서는 또 다른 WG Draft 문서인 “Applicability of the QUIC Transport Protocol(draft-ietf-quic-applicability-02)”를 참조하면 된다. 참고로, QUIC WG의 첫 번째 워킹그룹 회의는 2016년 11월 서울에서 개최된 97차 IETF 총회에서 열렸으며 현재 구글, 마이크로소프트 및 모질라 등에서 주도적으로 표준화 추진 중이며 다음의 목표를 이루기 위해서 다양한 주제의 주요문서들이 다루어지고 있다.

- HTTP/2를 포함한 애플리케이션의 연결 설정 및 전반적인 전송 대기 시간 최소화
- HOL(head-of-line)블록 현상 없이 다중화 제공
- 확산을 가능하게 하기 위해 경로 끝단에서의 변경만 요구
- 다중 경로 및 전달 오류 수정 확장 기능 사용
- TLS 1.3을 디폴트로 사용하여 항상 전송계층의 보안을 제공

일반적인 TCP 기반 통신에서는 기본적으로 연결 수립을 위해 일련의 Handshake 과정(Three-Way Handshake)이 필요하며, SSL/TLS 암호화 통신을 적용한다면 추가적인 일련의 Handshake 과정이 필요하다. 다시 말해서, 실제 데이터 통신 전에 여러 단계의 시그널링 교환 과정이 발생하게 된다. 신뢰성 있는 통신을 제공하기 위해 TLS와 TCP가 결합되는 경우 연결이 설정되기 위해서는 약 4.5 RTT(Round-trip time, 전송 왕복시간)의 시간이 필요하다. 반면, QUIC은 UDP 프로토콜 기반에서 동작하기 때문에 기본적으로 Zero-RTT이 원칙이며, 따라서 당연히 연결 수립을 위한 오버헤드가 획기적으로 감소한다. 또한, 일반적인 TCP는 여러 개의 스트림이 다중화 되어 있는 환경에서 서로 다른 스트림으로부터 트래픽이 전송된다 하더라도 하나의 스트림으로부터 전송되는 트래픽이 혼잡으로 인해 전송되지 않는다면, 이 트래픽으로 인해 다른 스트림에 속하는 트래픽도 전송되지 않는 HoL(Head-of-line) 블록 현상이 발생할 수 있다. 구글은 QUIC 프로토콜을 이용하여 이러한 통신과정에서 지연문제 및 패킷 손실에 따른 HoL 블록 현상 방지하고자 추진중이다.

구글은 크롬 브라우저와 구글 서비스의 일부에 QUIC을 적용되어 있어, 단말의 경우 약 10억 개의 단말에 QUIC이 사용되고 있으며 검색 엔진 쇼단(Shodan, https://www.shodan.io)의 QUIC 설치 동향을 보면 전 세계적으로 약 20억 개의 서버에 QUIC 서버 모듈이 구동되고 있다. 국내의 경우 네이버 및 일부 게임 회사에서 QUIC을 이용한 실시간 트래픽 전송 테스트중이다. 구글에 따르면, QUIC을 통한 구글 검색에서 평균 페이지 로딩 시간이 약 3% 개선됨을 보였다. 작은 수치처럼 보이겠지만 구글 검색이 이미 최대한 최적화되어 있는 것을 감안한다면 무시할 수 없는 개선이라고 볼 수 있다. 특히, 대기시간이 긴 웹 애플리케이션의 경우 더 큰 개선이 전망된다. 자사 동영상 서비스인 유튜브(YouTube)에 QUIC을 통해 접속한 사용자는 유튜브 시청 중 재 버퍼링 발생이 약 30% 감소되었다고 밝혔다. 아울러, QUIC의 개선된 혼잡제어 및 UDP 손실 복구를 통해 매우 느린 연결의 사용자가 QUIC 기반 홈페이지 로딩 시간을 개선할 수 있었다.

2.2 IRTF MAP RG (Measurement and Analysis for Protocols Research Group)

현재 우리가 사용하는 인터넷은 초기 개발 당시 예상했던 바와 다르게 성장해 왔다. 아울러, 인터넷 프로토콜은 가끔 설계 당시에 고려했던 환경과 다른 환경에서 동작되기도 한다. 예를 들어, 어떤 네트워크 구성 요소는 어떤 프로토콜을 다른 구성요소와 다르게 다루기도 하며, 이러한 프로토콜은 다시 사용되기도 하지만 반면에 잘못 사용되기도 한다. 비교적 최근인 2016년 새롭게 만들어진 연구그룹 MAP RG는 기존 IETF에서 개발되었던 프로토콜 스택을 제대로 사용할 수 있도록 하기 위해 더 나은 엔지니어링 측면에서의 결정을 내리는데 도움이 되는 측정 기술 및 데이터 소스를 다루고 있다[5][6]. MAP RG은 IETF와 다양한 교류를 통해 프로토콜 엔지니어링과 실제 적용 결과를 알린다. 최근 7월 개최된 102차 캐나다 몬트리올 회의에서는 MAP RG가 IETF의 기술개발과 운영 방향을 자문하며 또한 ISOC(Internet Society)에 대한 기술자문 그룹 역할을 수행 IAB(Internet Architecture Board)로부터 좋은 평가를 받았다는 내용과, MAP RG에서 다뤄지는 관련 연구 내용들이 2017년 7월 아일랜드 더블린과, 2018년 2월 오스트리아 비엔나에서 열린 국제학술대회 TMA(Network Traffic Measurement and Analysis Conference)에서 발표되었다는 내용을 공유하였다. 또한, 다음과 같이 QUIC, IPv6, DNS 등 기존에 잘 알려진 IETF 프로토콜 기반 여러 기술 이슈들을 SIDN Lab, Google, ETH Zurich 등에서 발표하였다.

- Dmap: Automating Domain Name Ecosystem Measurements and Applications (G. C. M. Moura et al., SIDN Labs)
- Monitoring DNS with open-source solutions (F. Espinoza, NIC Chile Research Lab)
- Packet Reordering in QUIC (Ian Swett, Google)
- Is Bufferbloat a Privacy Issue? (B. Trammell, ETH Zurich)
- Clusters in the Expanse: De-Aliasing IPv6 Hitlists (Q. Scheitle, Technical University of Munich)
- Measuring the usable maximum packet size across Internet paths (G. Fairhurst, University of Aberdeen)
- When the Dyke breaks: dissecting DNS Defenses during DDoS (G. C. M. Moura, SIDN Labs)
- Finding the source of DNS resolver users that were using old DNSSEC keys (W. Hardaker, USC/ISI)

2.3 IRTF ICC RG (Internet Congestion Control Research Group)

패킷 네트워크는 내재적인 설계 특성 때문에 일시적인 혼잡을 야기하는데 이러한 문제를 해결하기 위해 IRTF 연구그룹인 ICC RG은 혼잡제어 및 인터넷 혼잡 관리와 관련된 최신 연구주제를 논의하고 있다[7]. ICC RG가 고려해야 할 혼잡제어에는 여러 가지 측면이 있다. 예를 들어, 다른 실시간 미디어 애플리케이션은 동적 혼잡제어 메커니즘으로 인해 어려움을 겪으며, 특히 짧은 시간 단위로 변경해야 하는 경우 더욱 그렇다. 따라서 인터넷 전화의 등장과 고대역폭 인터넷 TV의 등장으로 혼잡제어 환경에 큰 영향을 미칠 것이다. 따라서 ICC RG는 다양한 응용 프로그램 클래스를 최상의 상태로 지원하는 방법과 이것이 새롭게 부상하는 혼잡제어 아키텍처에 어떤 영향을 미치는지 평가할 때 이러한 응용 프로그램을 고려하고 있다.

ICC RG 연구그룹에서 최근에 논의 했던 대표적인 내용 중의 하나는 네트워크 속도를 향상시킬 수 있는 구글의 새로운 혼잡제어 알고리즘인 BBR(Bottleneck Bandwidth and RTT)이라 할 수 있다[8]-[10]. BBR은 병목(Bottleneck) 대역폭(Bandwidth)과 왕복 전파 시간(Round-trip propagation time)에서 각각의 첫 알파벳에서 따온 것으로, 구글이 새롭게 자체 개발한 혼잡제어 알고리즘이며 네트워크 속도를 향상시킬 수 있는 기술이다. 기존 TCP에서 전통적으로 사용해 왔던 혼잡제어 알고리즘은 인터넷에 사용자가 몰리거나 대규모 다운로드/업로드 등 네트워크 트래픽이 갑자기 발생하는 경우 패킷 손실이 발생하면 자동적으로 네트워크 속도를 반으로 줄여 패킷의 손실을 방지하게 된다. 다시 말해서, 기존의 이진 증가 혼잡제어인 BIC나 리노는 패킷 손실을 이용하여 혼잡 여부를 판단하는 것이다. 반면, 구글에서 개발한 BBR 알고리즘은 여러 라우터에 걸쳐 지속적으로 쓰루풋과 트래픽 처리시간을 체크한다. 즉, 손실될 패킷의 현재 위치와 이동 경로를 미리 예측하여 네트워크에 트래픽이 갑자기 증가하더라도 전송 속도가 느려지는 것을 방지할 수 있다. 이를 기반으로 데이터가 네트워크를 통해 전송되는데 어느 정도 시간이 걸리는 지를 확인하고 네트워크가 감당할 수 있는 속도로 트래픽을 전송한다. 이와 같은 방식으로 기존 TCP 혼잡제어보다 더 효과적인 성능을 얻는다. 결국, BBR은 패킷 손실과 상관없이 데이터 패킷을 전송하는 최적 경로를 결정하기 위해 트래픽 속도를 실제로 측정하는 유일한 TCP 알고리즘이라 할 수 있다[8][10].

구글은 BBR 알고리즘을 유튜브에 적용했으며 전세계적으로 약 4%의 속도가 증가되는 효과를 확인한 것으로 알려졌으며, 일부 국가에서는 14% 이상 향상된 것으로 조사되었다. 특히, BBR은 앞서 설명한 전송 프로토콜인 QUIC와 호환된다. 구글의 최종적인 목표는 BBR을 IETF의 표준으로 만드는 것이다. 구글 서비스를 사용하는 몇몇 기업들은 이미 BBR의 효과를 체험하고 있다고 조사되었다. 구글 클라우드에서 웹사이트 50만 개를 호스팅하는 워드프레스의 경우 BBR을 이용했을 때 기존의 패킷손실 기반 혼잡제어 기법과 비교할 때 쓰루풋이 약 2,700배 향상되었다고 조사되었다. 또한, 큐잉 지연(Queuing Delay) 역시 약 25배 줄어들었다[9].

2.4 IRTF NWC RG (NetWork Coding for Efficient Network Communications Research Group)

NWCRG는 네트워크 코딩 기술에 대한 연구 분야를 다루고 향후 IETF에서 다룰 수 있는 표준화 이슈를 도출하는 연구그룹이다[11][12]. 네트워크 코딩 기술은 기본적으로 기존의 라우터 방식을 사용했을 때와 비교하여 통신 네트워크의 전송 속도 향상을 목적으로 통신 자원을 효율적으로 사용하는 것이며, 네트워크 쓰루풋(Throughput), 효율성(Efficiency), 지연(Latency), 확장성(Scalability) 및 각종 공격으로부터의 회복력(Resilience)등에 대한 향상을 위한 기술이라고 할 수 있다. 특히, IPTV와 같은 멀티캐스트 기반 미디어 스트리밍 응용 서비스를 위한 효율적인 자원 할당, 최대 전송 효율, 전송 에러의 최소화를 목적으로 사용되는 기술이기도 하다. 네트워크 코딩은 최초로 멀티캐스트 유선 상황에 적용하기 위해 고안하였으며, 무선 상황에서도 구현 가능하다는 것이 몇몇 연구 결과를 통해 확인되고 실제적으로 증명되었다. 네트워크 코딩은 소스노드에서 목적노드까지 다중 홉으로 전송되는 네트워크에 적용되는 기법이며 통신 네트워크에서 중계 역할을 담당하는 노드에서 네트워크 코딩이 적용된다. 최근 들어, 네트워크 코딩 기술은 무선 네트워크 (Wireless networks), 콘텐트 배포(Content distribution) 및 P2P 설계(Peer-to-Peer design)등의 분야에서 통신 성능을 향상시키고 있다.

위에서 언급했듯이 네트워크 코딩 기술은 기본적으로 성능향상에 목표를 두기 때문에 그 동안 연구를 살펴보면 표준화 보다는 실제 구현 과정에서 성능향상을 위한 알고리즘에 중점을 두고 진행되어 왔다. NWCRG에서는 인터넷 통신에 도움이 되는 네트워크 코딩 원칙과 기법을 연구를 하며, 첫 번째 목표는 다양한 연구 결과를 수집하고 인터넷 통신을 향상 시키는 네트워크 코딩의 실제적인 응용을 개발하

기 위한 다양한 의견을 상정하고 논의하는 것이다. 두 번째 목표는, 기존의 실제적인 구현 결과를 공유하고, 일반적인 기능(Functionalities)을 정의하고, 네트워크 코딩 기반 통신을 표준화 논의의 필요성을 도출하는 것이다. 결국, 실제 IETF WG에서 논의될 표준화 항목을 도출하는 것이 궁극적인 목표라 할 수 있다. NWCRG에서 주로 다뤄지는 연구 분야는 다음과 같다

- 성능 및 효율성(Performance and Efficiency)
- 응용 계층(Application layer)
- 데이터 계층과의 연관성(Implication to Data Link Layer)
- 공격에 대한 보안/보호/견인성(Security, privacy, robustness to attacks)
- 경제적인 측면(Network coding implications on pricing and economics)

2.5 IETF MPTCP WG (Multipath TCP Working Group)

이미 앞에서 언급했듯이, 현재 가장 널리 쓰이는 TCP는 종단 간 하나의 경로를 통해 신뢰성 있는 전송을 수행 할 때 사용된다. 다시 말해서, TCP는 기본적으로 단대단(End-to-End) 통신 경로(Path)가 하나로 구성된다. 하지만, 최근 유무선 인터넷 단말들은 다수의 유무선 네트워크 인터페이스를 탑재함으로써 다중 경로(Multipath)를 통한 전송이 가능해졌고, 따라서 이를 활용하는 다중 경로 전송 프로토콜이 요구되었다. MPTCP(Multipath TCP) WG은 이러한 문제를 해결하기 위해서 구성된 표준화 워킹그룹이며 전송 분야에 속해 있다[13]-[15].

MPTCP 개발 및 구현을 위한 구조 측면에서의 가이드라인을 위해서는 RFC 인 “Architectural Guidelines for Multipath TCP Development, RFC 6182”를 참조하면 된다. 또한, 최근 WG에서 진행되는 대표적인 흥미로운 Internet Draft 문서로는 최근 이슈가 되고 있는 5G 환경에서의 MPTCP 운영에 관한 고려사항을 다루고 있는 “Considerations for MPTCP operation in 5G(draft-defoy-mptcp-considerations-for-5g-01)”와 TCP Fast Open을 다루고 있는 “TFO support for Multipath TCP(draft-barre-mptcp-tfo-03), 단방향 지연에 대한 고려사항을 다루고 있는 “One Way Latency Considerations for MPTCP(draft-song-mptcp-owl-04) 문서들이 있다.

MPTCP 기술의 핵심 개념은 통신 경로를 여러 개로 늘리고자 하는 전송계층 프로토콜로서, 인터넷 단말에 할당된 복수 네트워크 주소를 활용하여 호스트간 TCP 서브플로우를 다중으로 동시에 구성하여 다중 경로 TCP(MPTCP) 연결을 가능하게 한다. 사용자는 이를 이용하여 동시에 다중 경로 전송을 수행하거나, 기존 전송 경로에 추가적인 보조 전송 경로를 더해 주 전송 경로에 이상 발생 시 보조 경로를 통한 전송을 수행하여 전체 데이터 전송 속도를 높이거나 보다 안정성 있는 전송을 가능하게 한다. MPTCP는 TCP를 기반으로 구성되었기 때문에 기본적으로 TCP를 하위 지원하여 현재의 네트워크 구조를 바꾸지 않고서도 손쉽게 적용이 가능하다. 또한, 단일 전송 경로 중 가장 전송 성능이 우수한 경로 이상의 전송 속도를 낼 수 있도록 설계되어 성능면에서도 우수하다고 할 수 있다.

최근 들어, LTE와 같은 이동통신 네트워크는 발전을 거듭하고 있는데 규모 또한 확장되고 있으며 5G로의 발전을 준비하고 있다. 위에서 이미 언급한 Internet Draft “Considerations for MPTCP operation in 5G(draft-defoy-mptcp-considerations-for-5g-01)” 문서가 그 예라 할 수 있다.

2.6 그 외 성능 측정 및 TCP 확장 표준화 동향

인터넷 트래픽을 효율적으로 관리하기 위해 가장 기본적으로 선행되어야 하는 것은 인터넷 트래픽을 측정하는 것이다. 이를 위한 IETF 워킹그룹으로는 전송 분야(Transport Area) 내 IPPM(IP Performance Metrics) WG과 운영 및 관리 분야(Operations and Management Area) 내 LMAP(Large-Scale measurement of broadband performance) WG이 있다[16]. IPPM WG은 인터넷 트래픽 측정에 관련된 표준 기술을 다루기 위한 워킹그룹으로 주요 목표는 인터넷 트래픽 전송 관련된 표준 메트릭(Metrics)을 개발하는 것이며, 품질(Quality), 성능(Performance) 그리고 안정성(Reliability) 등이 표준 메트릭 개발에서 고려 한다. LMAP WG은 대규모 광대역 성능 측정 인프라 표준화에 대한 중요한 작업을 진행하고 있다. 홈 또는 엔터프라이즈 엣지 라우터, 개인용 컴퓨터, 모바일 장치, 셋톱박스와 같은 광대역 액세스 장치의 유·무선 성능 측정을 위한 LMAP 측정 시스템을 표준화 한다. 대규모 인터넷 부분 측정은 시간과 지리적으로 성능을 정확하게 특성화하고 ISP와 사용자가 네트워크 진단을 조사하여 공공 정책 개발을 지원하기 위해 정보를 수집하는데 있어 필수적이다. LMAP WG는 IPPM WG이 정의한 표준 메트릭을 사용한다.

TCP 기반 전송 기법의 확장 기술을 다루는 IETF 워킹그룹으로는 전송 분야 내 RMCAT(RTP Media Congestion Avoidance Techniques) WG, TCPM(TCP Maintenance and Minor Extensions) WG이 있다[17][18]. RMCAT WG은 실시간 오디오 및 비디오 통신을 위한 새로운 인터넷 사용을 처리하기 위해 혼잡제어 알고리즘을 개발하고 평가하고 있다. TCPM WG은 TCP의 작동에 대한 적은 범위 내에서의 수정 및 확장을 다루고 있다.

III. 결론

본 고에서는 급격한 데이터 트래픽 증가에 따른 인터넷 성능을 향상시키기 위해 더욱 정교한 방법을 찾고 해결하기 위한 최근 연구 동향을 살피고자 인터넷 전 분야의 표준화 기술을 다루고 있는 세계적인 국제표준화 기구 IETF는 물론 IETF 표준화 사전연구기구인 IRTF 기반으로 조사 분석하였다. IETF 워킹그룹의 경우 QUIC WG, MPTCP WG, IPPM WG, LMAP WG, RMCAT WG, TCPM WG 등, IRTF 연구그룹의 경우 MAP RG, ICC RG, NWC RG 등을 기반으로 성능 및 효율성 향상을 다루는 주요 기술들의 최신 표준화 동향을 소개하였다.

본 고에서 조사 분석한 바와 같이, IETF는 물론 IRTF에서는 기존에 표준화가 완료된 네트워크/전송 계층 프로토콜 및 인터페이스 기반에 실제 다양한 형태의 트래픽 발생을 고려하고, 측정, 성능분석, 효율성 향상 등에 대한 일종의 검증 및 고도화 과정을 다루는 주제를 많이 다루기 시작했다. 향후 5G 및 사물인터넷 시대를 고려하면 발생하는 인터넷 트래픽의 양은 지금 보다 더욱 폭발적으로 증가할 것으로 예상된다. 이러한 상황에서 인터넷 성능 및 효율성 향상을 위한 표준화 개발은 물론 표준화 가능 이슈에 대한 연구는 매우 중요하다고 판단되므로 국내의 개인 연구자는 물론 및 관련 기관/기업은 관심을 갖고 대응을 해야할 것이다.

Acknowledgement

이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2017R1D1A1B03033024)

References

[1] The Zettabyte Era: Trends and Analysis, Cisco, Jun 7, 2017.
[2] 김평수, “인터넷 성능 및 효율성 향상을 위한 최근 기술 동향”, ICT 신기술, 주간기술동향 1825호, 2017년12월6일.
[3] QUIC(Quick UDP Internet Connection) 기술, TTA 표준화 포럼 기술문서(forum.tta.or.kr), 2017.
[4] S. Jero, “QUIC: Performance and Security at the Transport Layer”, IETF Journal, Vol.12, No.2, Nov. 2016.
[5] Measurement and Analysis for Protocols Research Group(MAPRG), IRTF, https://irtf.org/maprg
[6] M. Kühlewind and D. Plonka, “A New MAP: Measurement and Analysis for Protocols Research Group (MAPRG), ACM Internet Measurement Conference 2016, Nov. 2016
[7] Internet Congestion Control Research Group(ICCRG), IRTF, https://irtf.org/iccrg
[8] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, V. Jacobson, “BBR: Congestion-Based Congestion Control”, ACM QUEUE, Vol.14, No.5, 2016, pp.20-53.
[9] Brandon Butler, “구글, 인터넷 속도 14% 높이는 새 기술 개발”, Network World, CIO번역 기사(www.ciokorea.com/news/35292), 2017년8월23일
[10] V. Jain, V. Mittal, M. P. Tahiliani, “Design and implementation of TCP BBR in ns-3”, Proceedings of the 10th Workshop on ns-3, 2018, pp.16-22.
[11] NetWork Coding for Efficient Network Communications Research Group(NWCRG), IRTF, https://irtf.org/nwcrg
[12] Privacy Models, Architectures and Tools for the Information Society, Activity Report 2017, Inria, pp. 12, 2017
[13] LTE와 Wi-Fi간의 연동 및 병합 기술(2): Non-3GPP기반, 넷매니아즈, 2014년11월8일
[14] Q. Peng, A. Walid, J. Hwang, S. H. Low, “Multipath TCP: Analysis, Design, and Implementation”, IEEE/ACM Transactions on Networking, Vol.24, No.1, Dec. 2016, pp.596-609.
[15] X. Liu, D. Shan, R. Shu, and T. Zhang, “MPTCP Tunnel: An Architecture for Aggregating Bandwidth of Heterogeneous Access Networks”, Wireless Communications and Mobile Computing, Vol.2018, Article ID 2045760, 2018, pp.1-11.
[16] V. Bajpai, J. Schönwälder, “A Survey on Internet Performance Measurement Platforms and Related Standardization Efforts” IEEE Communications Surveys & Tutorials, Vol.17, No.3, April 2015, pp.1313-1341.
[17] L. D. Cicco, G. Carlucci, S. Mascolo, “Congestion Control for WebRTC: Standardization Status and Open Issues”, IEEE Communications Standards Magazine, Vol.1, No.2, July 2017, pp.22-27.
[18] TCP Maintenance and Minor Extensions Working

Group(TCMP WG), IRTF, https://datatracker.ietf.org/wg/tcpm/about/

Biographies



김평수

2001년: 서울대학교 전기컴퓨터공학부 공학박사
2001년 ~ 2005년: 삼성전자 디지털미디어연구소 책임연구원
2005년 ~ 현재: 한국산업기술대학교 전자공학부 교수
2012년: University of California, Irvine, CECS, Visiting Scholar
2005년 ~ 2014년, 2018년 ~ 현재: ICT 국제표준화전문가
주요관심분야: 차세대 유무선 네트워크, IPv6 운용 및 전환 기술, 미래인터넷 및 IoT 네이밍, 시스템 소프트웨어 솔루션, 최적 추정 필터링

BitString 기반 RPL 확장 기술

윤주상[1], 홍지연[2], 이경헌[3]
[1]동의대학교 산업ICT기술공학, [2]동의대학교 디지털미디어공학과, [3]동의대학교 응용소프트웨어공학과

Abstract

본 고에서는 IETF에서 제안된 BIER(Bit Index Explicit Replication) 도메인에서 사용되는 비트스트링(Bitstring) 기법을 RPL 라우팅 프로토콜에 확장하는 RPL-BIER 기술을 소개한다. RPL은 자원 제약적 특징을 가진 저전력 디바이스로 구성된 저전력 손실 네트워크에서 사용하기 위해 개발된 라우팅 프로토콜이다. 따라서 데이터 전달 과정에서의 오버헤드를 줄이는 기술이 필요하다. 특히, RPL 라우팅 프로토콜은 유니캐스트 측면에서는 기존 라우팅 프로토콜에 비해 효율적이나 멀티캐스트 서비스 측면에서는 데이터 전달 과정의 오버헤드가 크다. 본 고에서는 RPL-BIER 기술 소개와 함께 이 기술이 멀티캐스트 데이터 전달 과정에서 라우팅 오버헤드를 어떻게 감소 시키는지 상세히 기술한다.

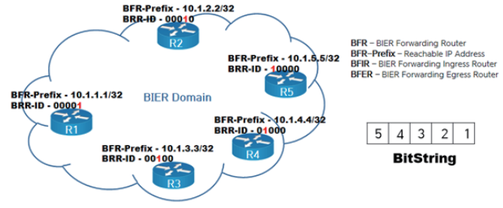
I. 서론

사물 인터넷 환경은 모든 사물들이 IP(Internet Protocol) 기반의 인터넷에 연결되어 필요한 서비스를 제공하는 환경을 말한다. 또한 사물 인터넷 기술이 발전함에 따라 저전력 손실 네트워크 정의된 LLNs(Low Power and Lossy Networks)에서 효율적인 데이터 전달 서비스를 위한 다양한 기술이 연구되고 있다. 특히, RPL(Routing Protocol for Low Power and Lossy Networks) 라우팅 프로토콜[1]은 LLN에 최적화된 프로토콜이다. 또한 RPL 은 데이터 전달을 위해 네트워크 트리를 구성하고 트리를 기반으로 데이터를 전달한다. 하지만 이런 트리 기반 데이터 전달의 경우 유니캐스트 서비스 제공에는 장점이 있지만 멀티캐스트 서비스 제공 시 오버레이 유니캐스트 형태로 서비스를 제공해야 하기 때문에 LLN을 구성하는 노드 입장에서는 큰 오버헤드로 작용한다. 본 고에서는 멀티캐스트 서비스 제공 시 효율적인 방법인 BIER 기술을 LLN에 적용하는 기술인 RPL-BIER 방법에 대해서 자세히 살펴본다. 우선, 2장에서 BIER 아키텍쳐 모델과 비트스트링 기반 데이터 포워딩 방법을 살펴보고, 3장에서는 RPL-BIER을 자세히 기술한다. 마지막으로 결론을 기술한다.

II. Bit Index Explicit Replication(BIER) Architecture

우리가 현재 사용하고 있는 IP 네트워크에서 데이터 전달 방법은 주어진 패킷을 운반할 때 특정 목적을 위해 구성된 경로에 따라 전달된다. 이를 위해서는 중계 라우터가 경로 흐름 상태를 유지해야 하며, 관련 경로를 결정하기 위한 라우팅 프로토콜이 필요하다. 본 고에서 소개하는 BIER[2]은 데이터 전달 과정에서 경로를 미리 설정하기 위한 라우팅 프로토콜이 필요 없는 기술로 평가되고 있다. 이번 장에서는 BIER 기술 소개와 BIER 기술이 가진 장점 등을 기술한다.

2.1 BIER 도메인



〈그림 1. BIER 도메인〉

BIER 도메인은 주소 체계를 IP가 아닌 연속된 비트 값으로 정의되는 비트스트링 (BitString)을 라우터의 ID로 정의하며 이를 통해 데이터를 전달하는 네트워크 도메인으로 정의된다. 따라서 BIER 도메인은 비트스트링 식별자를 기반으로 데이터 포워딩을 수행하는 비트포워딩 라우터 (Bit-Forwarding Router:BFR)로 구성된다. 여기서 BFR은 고유한 Bit Position(BFR-ID) 값을 할당 받게 된다. BFR-ID는 “1” 값의 비트 위치를 결정하는 비트 포지션(Bit Position) 할당 방법으로 결정된다. BIER 도메인에서 사용할 수 있는 비트스트링 사이즈는 256, 512, 1024, 2056 등의 단위로 설정할 수 있다.

2.2 비트 포지션 전달 (Bit Position Advertisement)

BIER 도메인 안에 자신의 BFR-ID와 BFR-Prefix 값을 네

트워크 내 라우터에 전달하기 위해 플러딩 기법을 통해 자신의 비트 포지션 정보를 전달한다.

2.3 BIER 도메인 포워딩 기술

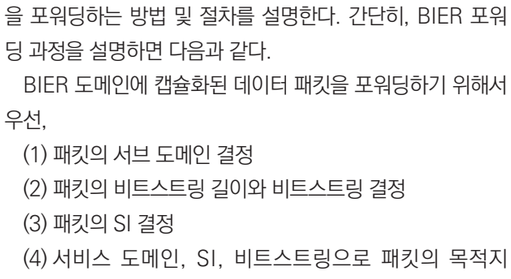
이번 서브절에서는 BIER 도메인에서 캡슐화된 데이터 패킷을 포워딩하는 방법 및 절차를 설명한다. 간단히, BIER 포워딩 과정을 설명하면 다음과 같다.

BIER 도메인에 캡슐화된 데이터 패킷을 포워딩하기 위해서 우선,

(1) 패킷의 서브 도메인 결정
(2) 패킷의 비트스트링 길이와 비트스트링 결정
(3) 패킷의 SI 결정
(4) 서비스 도메인, SI, 비트스트링으로 패킷의 목적지 BFER 집합을 결정
(5) 패킷의 목적지 BFER의 인접한 다음 홉 결정
(6) 패킷의 비트스트링은 전달되는 데이터의 비트스트링은 각 라우터에서 업데이트 후 결정

2.4 Bit Index Forwarding Table

BIER 도메인에서는 BFR에 할당되어 있는 BFR-ID와 BFR-Prefix를 이용하여 비트스트링 데이터 패킷을 라우팅하기 위한 BIFT(Bit Index Forwarding Table)을 BFR내부에 구성한다. 그림 2에서 A 라우터는 자신에 연결된 두 개의 인접 라우터에 정의된 BFR의 BFR-ID 값을 OR 연산을 통해 비트마스크 (BitMask)를 만들고 이 값을 BFR-Prefix로 정의하여 들어오는 데이터 패킷의 비트스트링 값에 비교하여 데이터 포워딩을 수행하도록 한다. 그림 2에는 D 노드의 BIFT을 예로 도시하고 있다.

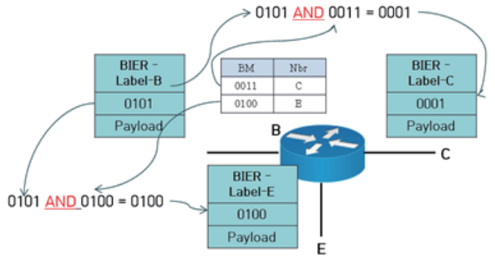


〈그림 2. Bit Index Forwarding Table 구성 예〉

2.5 BIER Forwarding Semantic

BIER 도메인에서는 데이터 포워딩을 위해서 각 라우터의 BIFT을 이용한다. BIER 포워딩 방법은 다음과 같다. 먼저 수신된 패킷으로부터 전달된 목적지 노드의 비트스트링을 축출한다. 여기 B 노드는 자신의 BIFT 내에 비트마스크와 추출된 비트스트링 값의 AND 연산을 수행한다. 연산 후의 결과를 전달할 데이터의 헤더에 업데이트 후 데이터를 전달한다. 이 과정을 비트스트링 값이 모든 0 값을 가질 때까지 반복하면 라우터를 이동한다.

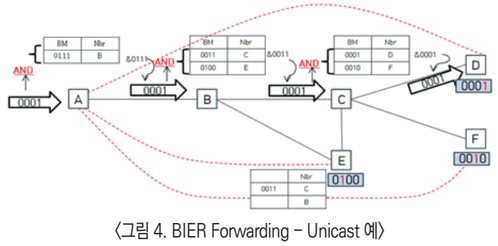
그림 3은 B 링크를 통해 전달된 데이터가 AND 연산 후 C 링크와 E 링크를 통해 비트스트링 값의 업데이트 후 전달되는 과정을 도시하고 있다.



〈그림 3. BIER Forwarding Semantic 예〉

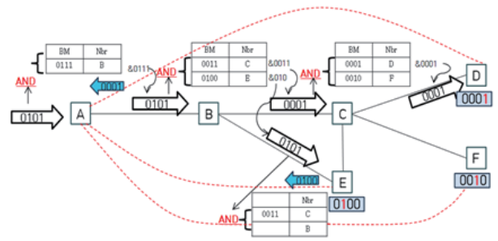
2.6 BIER Packet Forwarding (Unicast vs Multicast)

BIER 도메인에서 유니캐스트를 위한 데이터 포워딩의 예를 그림 4에 도시하고 있다. 2.5 절에 기술한 것처럼 데이터 포워딩 기술은 전달 받은 데이터 패킷 내부의 비트스트링과 각 라우터 내부에 정의된 BIFT의 비트마스크와 AND 연산을 수행하고 하나의 비트값이라도 1 값을 가지는 비트마스크 링크로 AND 연산 결과 값을 비트스트링 값으로 업데이트 후 데이터를 전달한다. 그림 4에 있는 예를 들어 설명하면 "0001" 값을 가진 비트스트링 패킷이 수신되면 인접한 라우터에 A 노드에서 인접 노드인 C 노드의 비트마스크 "0111" 와 AND 연산을 수행한다. 그 후 연산 결과인 "0001"을 데이터의 비트스트링 값으로 업데이트 후 데이터 패킷을 B 노드로 전달한다. 이 과정을 "0001" 값을 BFR-ID로 가진 D 노드까지 전달될 때까지 반복하여 수행한다.



〈그림 4. BIER Forwarding - Unicast 예〉

BIER 도메인에서 멀티 캐스트 데이터 전달의 예는 그림 5에 도시하고 있다. 전달 방법은 유니캐스트 방법에서 기술한 것처럼 데이터 비트스트링 값과 각 라우터에 정의된 BIFT의 비트마스크와 AND 연산을 수행하고 "1" 비트값을 가진 이웃 노드의 링크로 데이터를 전달한다. 그림 5에 도시된 예를 보면 B 노드에 전달된 "0101" 비트스트링 데이터는 B 노드의 BIFT 비트마스크와 AND 연산 수행 결과 두 개 모두 "1" 비트값을 가진 비트스트링이 생성되었다. 이 경우 두 개의 이웃 노드에 각 AND 연산 수행 결과 값을 비트스트링 값으로 정의한 데이터를 전달한다. 따라서 C 노드에 전달되는 데이터의 비트스트링은 "0001" 이며 E 노드에 전달되는 데이터의 비트스트링 값은 "0101" 이다. 이와 같이 BIER 도메인에서는 데이터 전달 과정에서 멀티캐스트 서비스를 제공하기 위한 별도의 멀티캐스트 트리를 형성하지 않고도 비트스트링을 통해 네트워크 오버헤드가 작은 효율적 멀티캐스트 서비스를 제공할 수 있다.

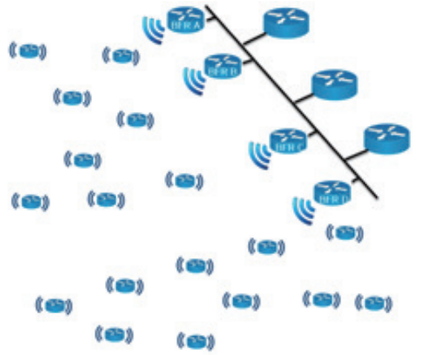


〈그림 5. BIER Forwarding - Multicast 예〉

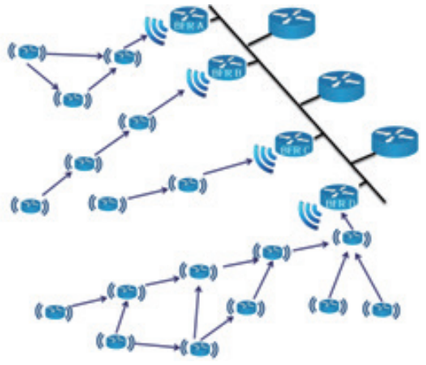
III. RPL-BIER

RPL라우팅 프로토콜은 저전력 손실 무선 네트워크 환경에서 사용하기 위해 개발된 경량화 된 IP 기반 라우팅 프로토콜이다. 특히, RPL은 라우터들끼리 데이터 전달을 위해 DODAG(Destination-Oriented Directed Acyclic Graph)를 구성하고 구성된 트리에서 데이터 전달 방법으로 플러딩 기법을 사용한다. RPL은 네트워크 사이즈, 라우팅 상태, 패킷 사이즈 등에 따라서 라우팅 오버헤드에 대한 트레이드오프가 존재한다. 본 고에서는 LLN 환경에서 RPL을 통한 멀티캐스팅 서비스 제공 시 네트워크 오버헤드를 격감 시킬 수 있는 RPL-BIER 기법을 소개한다. BIER은 위에서 기술한 것처럼 IP 주소를 대신해서 트리 기반의 위치에 기반하여 비트스트링 값을 각 노드에 정의한다. 이를 BIER에서는 비트 포지셔닝이라 하며 비트 포지셔닝 후 각 노드는 비트맵을 형성하고 데이터 전달을 위한 포워딩 테이블을 구성한 후 데이터를 전달한다. 또한, BIER 적용 방법은 RPL 저장 모드(Storing Mode), 비저장 모드(Non-Storing Mode)로 구분하여 적용된다.

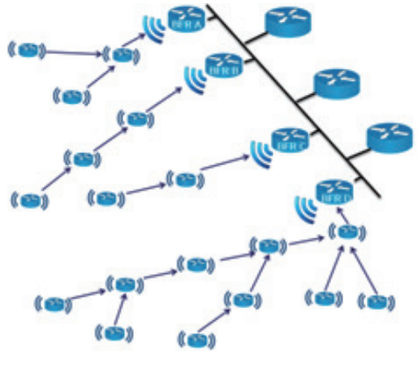
3.1. RPL DODAG 트리



〈그림 6. 네트워크 토폴로지〉



〈그림 7. RPL DODAGs〉



〈그림 8. RPL DODAG Tree〉

RPL은 데이터 전달을 위한 트리를 형성할 때 루트를 중심으로 트리를 구성한다. 이때 DODAG를 형성하고 이를 기반으로 DODAG 트리를 구성하다. 그림 6, 7, 8은 LLN에서

DODAG 트리를 구성하는 과정을 그림으로 도시하고 있다.

3.2. RPL-BIER 확장 기술

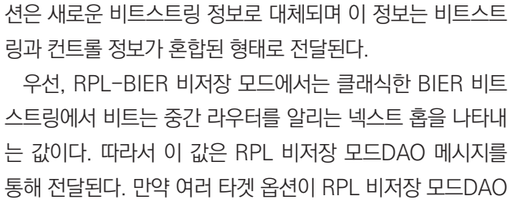
RPL-BIER[3] 확장은 RPL 저장 모드(Storing Mode), 비저장 모드(Non-Storing Mode)로 구분되어 적용된다. RPL에 비트스트링을 전달하기 위한 컨트롤 메시지 옵션이 추가되어 있다. 추가적으로 RPL-BIER 동작 모드에서 RPL 타겟 옵션은 새로운 비트스트링 정보로 대체되며 이 정보는 비트스트링과 컨트롤 정보가 혼합된 형태로 전달된다.

우선, RPL-BIER 비저장 모드에서는 클래식한 BIER 비트스트링에서 비트는 중간 라우터를 알리는 넥스트 홉을 나타내는 값이다. 따라서 이 값은 RPL 비저장 모드DAO 메시지를 통해 전달된다. 만약 여러 타겟 옵션이 RPL 비저장 모드DAO 메시지에서 Transit Information Option(TIO)을 각 타겟으로 분해할 수 있다면 여러 타겟을 통합한 비트스트링을 하나의 BIO로 대신할 수 있는 장점을 가지고 있다.

RPL-BIER 저장 모드에서 클래식한 BIER 비트스트링에서 비트는 마지막 최종 목적지를 나타내는 값이다. RPL-BIER 저장 모드도 또한 타겟 모드에 대한 전달을 RPL 저장 모드 DAO 메시지를 통해 전달한다. DAO 메시지 내에 비트스트링 정보 옵션은 수집된 통합 비트스트링 값을 포함하고 있으며 이 값은 트리 구조에서 자신의 하위에 존재하는 모든 노드의 비트스트링 값을 포함하고 있다. 이 모드에서는 TIO을 사용하지 않는다.

3.2.1 비트스트링 정보 정의

RPL-BIER에서는 새로운 컨트롤 메시지 옵션으로 Bit-String Information option(BIO)이 추가되어 있다. BIO는 한 개 이상의 타겟의 주소를 대신하여 사용된다. 그림 9는 비트스트링 정보 옵션 포맷을 도시하고 있다. 여기서 옵션 길이는 옵션 필드의 길이를 정의하고 비트스트링 타입은 비트스트링 필드의 길이를 정의한다. 그림 10은 비트스트링 타입 별 비트스트링 길이를 정의하고 있다.



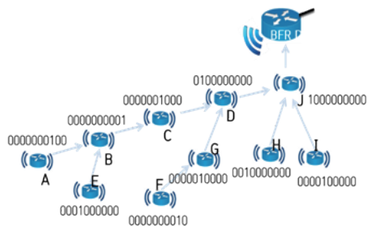
〈그림 9. 비트스트링 정보 옵션 포멧〉

BitString Type	BitString Size
15	8 bits
16	16 bits
17	48 bits
18	96 bits
19	160 bits

〈그림 10. 비트스트링 타입 별 비트스트링 길이 정의〉

3.2.2 비트 포지셔닝 (Bit Positioning)

RPL에서 IP기반으로 주소를 지정할 때는 DODAG는 상향 트래픽을 지원하는 용도로만 사용할 수 있다. 그렇기 때문에 hop-by-hop 형식으로 루트를 향하는 트래픽을 전달하기 위해 다음 홉 노드만 라우팅 테이블에 저장한다. 또한, 하향 트래픽을 지원하기 위해서는 RPL 노드의 목적지 정보를 목적지 알리기 위해 DAO(Destination Advertisement Object) 메시지를 생성하여 전달하는 과정을 수행한다. 이때 트리는 형성하기 위한 노드는 자신에 연결된 노드들을 대상으로 DAO 메시지를 DODAG 루트에 전달한다. 이 경우, DAO 메시지는 메시지를 생성 한 RPL 노드의 부모 집합에 대한 정보도 포함한다. 그런 다음 경로를 따라 모든 RPL 노드로부터 DAO메시지를 수신하면 DODAG 루트는 DAO 상위 정보를 보고 경로를 구성하는 방법을 사용한다. RPL-BIER에서는 RPL 루트 탐색과 경로 생성 과정에 있어 RPL 기법을 그대로 사용한다. 단, IP 기반이 아닌 비트 단위의 비트스트링 기반으로 데이터 전달이 이루어지기 때문에 트리 구조에 기반한 각 노드의 비트 값을 결정하는 비트 포지셔닝을 수행한다. 그림 11은 RPL 트리에서 각 노드의 비트 값을 결정하는 비트 포지셔닝 수행의 예를 보여주고 있다. 우선, 루트에 인접한 J 노드의 1비트 위치를 1로 지정하고 J 노드에 인접한 D, H, I 노드의 1비트 위치를 2, 3, 4 값을 비트로 지정하고 이와 같은 과정을 자신에 하위 노드에 반복적으로 수행하여 각 노드의 비트 값을 결정한다. 이때 설정된 10진수로 표기된 값은 비트 단위로 변화되어 지정된다. 이때 트리에 있는 각 노드는 같은 비트 값을 가질 수 없다. 이 결과는 그림 11에는 도시된 테이블처럼 결정되며 이 테이블은 루트에 비트 포지션 테이블 (Bit Position Table) 형태로 저장된다. 그림 11에는 노드의 주소 별 비트포지셔닝 수행 후 결과를 표로 도시하고 있다.

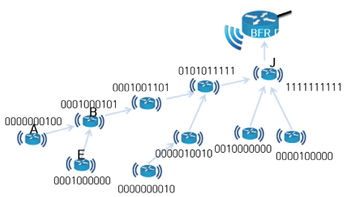


Address	Bit
A	8
B	10
C	7
D	2
E	4
F	9
G	6
H	3
I	5
J	1

〈그림 11. 비트포지셔닝 수행 및 각 노드의 비트스트링 값 결정의 예〉

3.2.3 비트스트링 (BitStrings)

RPL 트리를 구성하는 노드들은 자신에 할당된 비트 값을 가지고 있으며 각 노드는 자신의 하위 노드에 할당된 비트 값을 가지고 병합된 비트 값인 비트스트링 값을 결정하게 된다. 이때 각 노드가 가진 비트스트링 값을 비트맵 값으로 정의한다. 하위 노드는 자신의 비트맵 값을 상위 노드에 전달하기 위해 DAO 메시지를 사용한다. 각 노드는 자신의 하위 노드로부터 전달된 DAO 내에 비트맵 값과 자신의 비트 값을 OR 연산을 통해 자신의 비트맵 값을 결정한다. 이를 비트맵 연산이라 한다. 이 과정을 하위 노드로부터 상위 노드까지 반복되어 수행하고 루트 노드까지 비트스트링 값을 결정한다. 예를 들면 그림 12에 도시된 것처럼 B 노드는 자신의 하위 노드로 A 노드와 E 노드를 가지고 있으며 각 노드의 비트맵 값은 “0000000100”, “0001000000” 이다. DAO 메시지를 통해 B 노드는 A 노드, E 노드의 비트맵 값을 전달 받아 자신의 비트 값, “0000000001”와 OR 연산인 비트맵 연산을 수행한 후 자신의 비트맵 값으로 “0001000101”을 결정한다. 이후 B 노드는 DAO 메시지를 통해 자신의 상위 노드인 C 노드에 비트맵 값을 전달하고 이 값을 전달 받은 C 노드는 자신의 비트 값과 비트맵 연산을 통해 자신의 비트맵 값을 결정한다. 이 과정은 루트까지 반복되어 각 노드는 비트맵 값을 결정한다.



Child	BitMap
A	0000000100
E	0001000000

〈그림 12. B 노드의 Bit DAO Operation의 예〉

3.3. 비트 포워딩 동작 (Bit Forwarding Operation)

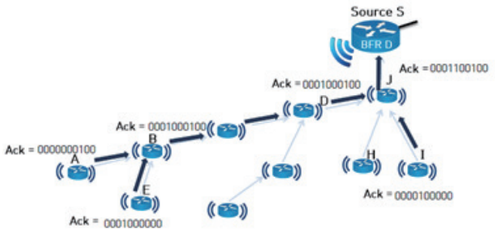
RPL-BIER에서 비트스트링을 통해 데이터를 전달한다. 전달 방법은 2가지 방법인 신뢰적 비트 포워딩 방법과 비신뢰적 비트 포워딩 방법으로 구분된다. 데이터 전달 과정에서 데이터 전달 결정은 데이터가 가진 목적지 비트스트링 값과 각 노드가 가진 비트맵 값 과의 AND 연산을 통해 데이터 전달 유무를 결정한다.

3.3.1 신뢰적 비트 포워딩 동작 (Reliable Bit Forwarding Operation)

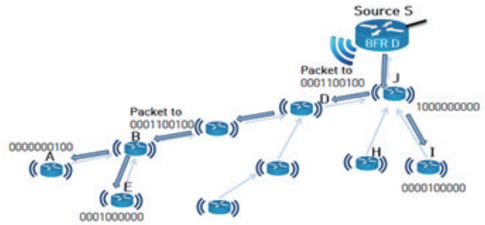
신뢰적 비트 포워딩 방법은 데이터 전달을 위해 각 노드에서 병합된 목적지 정보인 목적지 비트스트링 값을 확인 후 데이터 포워딩 유무를 결정하는 것이다. 그림 13에서처럼 A 노드, E 노드, I 노드를 목적지로 하는 비트 스트링 값은 3개의 노드가 가진 비트 값의 OR 연산을 통해 결정된다. 결정된 비트스트링 값은 “0001100100” 이다. 이 값은 루트를 통해 전달될 데이터의 목적지 비트스트링으로 결정되며 데이터 전달 과정에서 각 노드는 “0001100100”과 각 노드가 가진 하위 노드의 비트맵 값의 AND 연산을 통해 전달 유무를 결정한다. 각 비트 별 AND 연산 결과가 1이 나오면 노드는 1이 나온 하위 노드로 데이터를 전달한다. 예를 들어 J 노드에서 목적지 비트스트링 값과 각 하위 노드 간 AND 연산 과정은 아래와 같다.

{Dest bitmap = 0001100100
(Dest bitmap AND I’s bitmap) = 0000100000 –〉
I 노드 전달
(Dest bitmap AND H’s bitmap) = 0000000000 –〉
H 노드 전달 안함
(Dest bitmap AND D’s bitmap) = 0001000100 –〉
D 노드 전달 }

D 노드에 전달된 데이터 패킷은 트리를 구성한 하위 노드들에서 AND 연산 과정을 통해 목적지까지 전달된다. 그림 14는 신뢰적 비트 포워딩 동작으로 데이터를 전달 받은 노드가 상위로 ACK 메시지를 전달하는 과정이 도시되어 있다. ACK 메시지 과정 또한 상위로 전달되는 과정에서 병합ACK 메시지를 비트스트링 값으로 결정하며 상위로 전달된다. 이때 과정은 OR 연산이 수행된다. 따라서 두 개로 전달되는 ACK 메시지가 하나로 전달되는 효과를 볼 수 있다. 또한 RPL-BIER는 유니캐스트 서비스 보다 멀티캐스트 서비스 제공 시 큰 효과를 볼 수 있다.



〈그림 13. 신뢰적 비트 포워딩 동작 노드의 데이터 전달 예〉



〈그림 14. 신뢰적 비트 포워딩 동작 노드의 ACK 전달 예〉

3.3.2. 비신뢰적 비트 포워딩 방법 (Unreliable Bit Forwarding Operation)

비신뢰적 비트 포워딩 방법은 신뢰적 비트 포워딩 방법과 데이터 전달 방법은 같으나 데이터 전달 후 ACK 메시지를 전달 받지 않는 방법이다.

IV. 결 론

IETF에서 제안된 BIER은 IP을 대신하여 각 라우터에 고유한 비트 값을 정의하고 데이터 패킷에 IP을 대신하는 비트스트링을 정의하여 데이터를 전달하는 기술이다. 이 기술은 IP 네트워크에서 멀티캐스트 서비스 제공 시 발생하는 오버헤드를 격감 시키는 기술로 개발되었다. 특히, 멀티캐스트 서비스를 제공하는 데이터 포워딩 기술인 유니캐스트 서비스 제공 방법과 유사한 형태로 이루어지기 때문에 멀티캐스트 서비스 제공 시 빠르게 데이터를 전달 할 수 있는 장점을 가지고 있다. 본 고에서는 멀티캐스트 서비스 제공 시 장점을 가진 BIER 기술을 LLN에 적용하기 할 수 있는 RPL-BIER 기술을 소개하였다. LLN을 구성하고 있는 자원 제약적 노드에서는 멀티캐스트 서비스 제공에 대한 한계가 있다. 하지만 RPL-BIER 기술은 이런 한계를 극복할 수 있는 기술로 평가받고 있다. 따라서 이 기술은 사물인터넷의 로컬 네트워크로 정의되고 있는 LLN에서 멀티캐스트 서비스를 제공할 수 있는 기술로 평가될 수 있다.

Acknowledgement

이 논문은 2018년도 BB21+ 사업에 의하여 지원되었음

References

[1] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP. and R. Alexander, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks”, RFC 6550, March 2012.

[2] Wijnands, IJ., Rosen, E., Dolganow, A., Przygienda, T. and S. Aldrin, “Multicast Using Bit Index Explicit Replication (BIER)”, RFC 8279, November 2017.

[3] Thubert, P. , “RPL-BEIR”, draft-thubert-roll-bier-01, January 22, 2018.

Biographies



윤주상

2001년: 고려대학교 전기전자전파공학부 공학사
2003년: 고려대학교 전자공학과 공학석사
2008년: 고려대학교 전자컴퓨터공학과 공학박사
현재: 동의대학교 산업ICT기술공학전공 부교수
주요 관심분야: 미래인터넷, 사물인터넷, 엣지/포그 컴퓨팅
e-mail: jsyoun@deu.ac.kr



홍지연

2018년: 동의대학교 멀티미디어공학부 공학사
현재: 동의대학교 대학원 디지털미디어공학과 재학중
주요 관심분야: 빅데이터, 사물인터넷, 엣지/포그 컴퓨팅
e-mail: anasis6095@gmail.com



이경헌

현재: 동의대학교 응용소프트웨어공학과 재학중
주요 관심분야: AI, 사물인터넷, 엣지/포그 컴퓨팅
e-mail: typhoon33759@gmail.com

IETF/IRTF 에서의 블록체인 기술

이종혁[1], 박정수[2]
[1]상명대학교 프로토콜공학연구실, [2]한국전자통신연구원 표준연구본부

Abstract

비트코인(Bitcoin)으로 시작된 암호화폐(cryptocurrency) 붐은 아직도 현재 진행형이다. 이와 더불어 암호화폐의 근간을 이루는 블록체인(blockchain)을 다방면에서 활용하고자 하는 연구가 이루어지고 있다. 본 논문에서는 국제 인터넷 표준화 기구인 IETF에서의 블록체인 기술에 관한 표준화 및 IRTF에서의 분산된 인터넷 아키텍처 다

I. 서론

10년 전인 2008년 사토시 나카모토(Satoshi Nakamoto)라는 필명을 가진 누군가가 혹은 단체는 Bitcoin: A Peer-to-Peer Electronic Cash System 논문을 인터넷에 공개한다 [1]. 9 페이지짜리 이 논문은 현재 우리나라를 포함해 전세계적으로 불고 있는 암호화폐 열풍의 주인공인 비트코인의 백서(white paper)이다. 인터넷 환경에서 은행과 같은 제 3자의 도움 없이 개인간 지불이 가능케 하는 이 새로운 기술은 우리가 이미 잘 알고 있는 Peer-to-Peer (P2P) 네트워크, 공개키 기반 서명(signature), 해쉬(hash), 머클 트리(merkle tree) 등과 같은 기술을 이용한다. 인터넷 환경에서 사용 가능한 암호화폐 비트코인이 세상에 나온 이후 많은 아류 코인들이 속속 개발되고 있기도 하다. 또한, 암호화폐 투기에 대한 우려로 인해 암호화폐 규제가 늘고 있지만, 2018년 1월 암호화폐 시가총액은 이미 8,000억 달러를 돌파 할 정도로 하나의 산업으로 발전하고 있다.

블록체인은 비트코인이라는 암호화폐의 출현으로 인해 세상에 나타났지만, 암호화폐와 분리해 논할 수 있는 기술로 받아들여지고 있다. 네트워크로 연결된 컴퓨터가 주고 받는 데이터에 대한 무결성(integrity)과 비가역성(irreversibility)을 제공하기 위한 하나의 방안으로 사용 될 수 있으며 [2], 분산 컴퓨팅 네트워크 아키텍처를 구성하는데 사용 될 수 있는 하나의 플랫폼으로 기대된다. 특히, 특정 조건에 따라 자동화 된 실행이 보장 되는 스마트 컨트랙트(smart contract) 기술이 블록체인에 더해짐에 따라 분산 컴퓨팅 환경에서 자동화 된 동작 수행하는 플랫폼으로 성장하고 있다 [3].

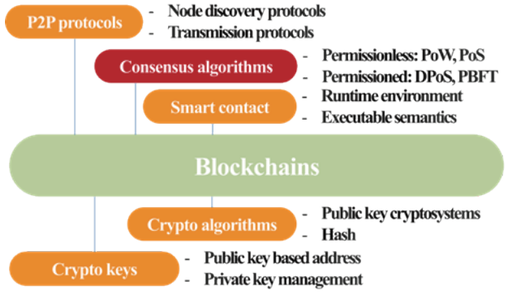
인터넷 아키텍처를 설계하고 다양한 인터넷 서비스를 실현하기 위한 프로토콜을 개발하고 표준화를 하는 IETF 에서도 블록체인을 활용한 인터넷 서비스 확장이나 프로토콜 설계에 관한 새로운 제안이 이루어지고 있다. 또한, 실질적 IETF 표준화 이전에 사전 연구를 진행하는 IRTF 에서도 2017년 9월부터 Decentralized Internet Infrastructure Proposed RG (DINRG) 연구 그룹을 만들고 관련 연구를 시작하고 있다.

본 논문은 IETF/IRTF에서의 블록체인 기술을 살펴 본다. 먼저 2장에서 블록체인의 주요 특징을 살펴본다. 3장에서는 현재 IETF 에서 제안되고 있는 블록체인 관련 기술을 살펴보고, 4장에서 IRTF 에서 진행되고 있는 분산된 인터넷 아키텍처를 위한 연구 내용을 소개한다. 5장에서 결론을 내린다.

II. 블록체인

분산원장기술(Distributed Ledger Technology)로 분류되는 블록체인은 블록(block)과 체인(chain)의 합성어로, P2P 네트워크로 연결 된 노드간 전송하는 트랜잭션(transaction)을 담고 있는 블록들이 체인과 같이 서로 연결되어 저장 된다는 의미를 가진다. P2P 네트워크로 구성되는 블록체인 네트워크내에서 각 노드는 트랜잭션이라고 불리우는 메시지를 브로드캐스트(broadcast) 형태로 전송 한다. 트랜잭션은 공개키 기반의 메시지 서명을 통해 보호 되며, 블록 타임(block time)이라는 특정 시간 동안 모아진 트랜잭션들은 합의 알고리즘(consensus algorithm)을 통해 하나의 블록으로 만들어진다.

비트코인은 자신의 블록체인 플랫폼을 가지며, 이더리움(Ethereum) [4]이라는 암호화폐 또한 자신만의 블록체인 플랫폼을 가진다. 다시 말해, 블록체인 플랫폼은 서로 다를 수 있으며, 서로 다른 플랫폼 위에 어떠한 서비스를 구현하느냐에 따라 암호화폐가 만들어 질 수 있고, 다른 형태의 서비스가 만들어 질 수 있다. 또한 블록체인 플랫폼에 따라 하나의 블록을 만들기 위한 블록 타임이 달라 질 수 있으며, 블록 크기, 트랜잭션의 크기, 합의 알고리즘 등이 달라 질 수 있다.



〈그림 1. 블록체인의 주요 콤포넌트〉

그림 1은 블록체인의 주요 콤포넌트를 나타낸다. 통신을 위한 P2P 프로토콜, 노드간 합의를 도출하기 위한 합의 알고리즘, 자동화 된 실행을 보장하기 위한 스마트 컨트랙트, 암호학적 보안성 제공을 위한 암호 알고리즘, 암호 알고리즘에서 사용되는 암호 키 등이 주요 콤포넌트이다.

블록체인을 구별하는 또다른 요소는 어떠한 아키텍처를 채택하고 있느냐이다. 암호화폐와 같이 블록체인 네트워크에 누구나 참여해 트랜잭션을 주고 받으며, 투명한 형태의 합의 알고리즘이 수행되어야 할 필요성이 있는 경우에는 공개형(public) 블록체인을 도입해야 한다. 대표적으로 비트코인, 이더리움과 같은 암호화폐가 채택하고 있는 블록체인 아키텍처이다. 이에 반해 컨소시엄(consortium)이나 비공개형(private) 아키텍처를 채택하고 있는 블록체인이 존재한다. 컨소시엄 블록체인은 컨소시엄을 맺고 있는 맴버들만이 해당하는 블록체인에 등록되어 합의 알고리즘을 수행하고, 비공개형 블록체인은 특정 운영주체에 의해 합의 알고리즘이 수행되는 형태를 가진다. 공개형 블록체인에 비해 컨소시엄 블록체인과 비공개형 블록체인은 블록체인에 참여하는 노드에 대한 접근 제어(access control)를 수행해야 하며, 이를 위해서 노드에 대한 ID 관리가 요구된다.

블록체인의 특징을 구별하는 또다른 요소는 합의 알고리즘이다. 어떠한 합의 알고리즘을 채택하고 있느냐에 따라 해당하는 블록체인의 특성이 달라진다. 비트코인, 이더리움에서는 작업증명(Proof-of-Work)을 합의 알고리즘으로 채택하여 어떠한 노드가 블록을 생성 할지를 결정한다. 작업증명은 공개형 블록체인에서 가장 널리 사용되는 합의 알고리즘이지만, 마이너(miner)라 불리우는 작업증명 참여 노드의 높은 연산와 전기 사용량으로 인해 지분증명(Proof-of-Stake), 위임 지분증명(Delegated Proof-of-Stake) 등의 합의 알고리즘이 개발되었다. 또한, 기존 비잔틴 장군 문제를 해결하기 위해 개발 된 Byzantine Fault-Tolerant, Practical Byzantine Fault-Tolerant 등을 컨소시엄이나 비공개형 블록체인에 채택하는 사례가 늘어나고 있다.

III. IETF 에서의 블록체인 기술

2018년 9월 현재까지 IETF 에서 블록체인 기술을 활용한 새로운 형태의 인터넷 프로토콜이나 서비스가 표준화 되어 RFC 로 출판 된 사례는 없으며, 특정 작업반(working group)에서 정식 문서로 채택되어 표준화가 이루어지고 있지도 않다. 하지만 2015년 부터 블록체인을 활용한 인터넷 프로토콜과 서비스가 개인 기고서(individual draft)로 제안되고 있으며, 주요한 개인 기고서는 다음과 같다.

- Special-Use Domain Name for Namecoin (June 30, 2015)
- ALTO for the blockchain (July 6, 2016)
- An analysis of the applicability of blockchain to secure IP addresses allocation, delegation and bindings (July 03, 2017)
- Blockchain Transaction Protocol for Constraint Nodes (March 02, 2018)
- SIPCoin: A Cryptocurrency for Preventing RoboCalling on the PSTN (March 2, 2018)

3.1 Special-Use Domain Name for Namecoin

2015년 6월 30일에 제출 된 기고서 Special-Use Domain Name for Namecoin (draft-grothoff-iesg-special-use-p2p-bit-00) [5]는 Namecoin 기반 도메인 네임 서비스를 위해 bit 이라는 도메인 네임을 할당하고 사용하는 방안에 대해서 제안한다. 본 기고서는 현재의 도메인 네임 서비스는 중앙 집중적으로 관리 되고 있기에, 이것을 Namecoin 블록체인을 이용해 분산된 형태로 바꾸기 위한 방안을 제시한다.

기존 도메인을 한번에 Namecoin 을 통한 도메인 서비스를 제공하는 것은 현실적으로 어렵기에 본 기고서에서는 bit 도메인 네임을 특별히 할당하여 서비스 할 것을 제안하였다.

3.2 ALTO for the blockchain

2016년 7월 처음 제안된 본 기고서는 P2P 네트워크 환경에서 블록체인 노드간 안정적인 데이터 통신을 위한 방안을 제안한다 [6]. 이를 위해 RFC 7285에 정의 된 Application-Layer Traffic Optimization (ALTO)를 활용하는 방안을 제시한다.

본 기고서에서 제시하는 방안은 1) 노드가 처음 블록체인 네트워크에 접속하는 부트스트래핑(bootstrapping) 과정이나 통신 중 접속이 끊겼을때 빠르게 접속 가능한 노드 정보를 할당 받는데 ALTO 사용을 제안하며, 2) ALTO를 통해 최소의 라우팅 비용을 제공 받아 궁극적으로 블록체인 네트워크 내에서 브로드캐스트 되는 메시지의 전파 시간(propagation latency)을 줄이고자 한다.

3.3 An analysis of the applicability of blockchain to secure IP addresses allocation, delegation and bindings

2017년 7월 처음 제안된 이후 2번의 갱신을 통해 2018년 6월 28일 기고서An analysis of the applicability of blockchain to secure IP addresses allocation, delegation and bindings (draft-paillisse-sidrops-blockchain-02)를 제출 하였다 [7]. 본 기고서에서는 공개키 기반 서명으로 계층적 주소 할당을 제공하면서도 PKI 구조를 사용하지 않고 블록체인 기반 주소 할당, 위임, 도메인 네임과의 바인딩 방안을 제시한다.

현재 인터넷에서의 계층적 주소 할당과 비슷한 방법으로 주소에 대한 소유권을 주장할 수 있으면서도, PKI 구조를 사용하지 않기에 인증서(certificate)나 인증기관(certificate authority)가 요구 되지 않는다. 블록체인의 트랜잭션 형태로 주소 할당, 위임, 바인딩 정보가 블록체인 내에 저장 되기에 데이터 분산화, 단일 장애점 제거, 데이터에 대한 비가역성 등 블록체인이 제공하는 다양한 장점을 가진다.

3.4 Blockchain Transaction Protocol for Constraint Nodes

2018년 9월 1일 제출 된 본 기고서는 사물인터넷 환경에서 자원이 한정적인 사물인터넷 기기를 위한 블록체인 아키텍처와 전송 프로토콜을 제안한다 [8]. 제안된 아키텍처에서는 사물인터넷 기기는 블록체인 네트워크에 직접 연결 되지 않고, 컨트롤러(controller)라고 불리는 노드에 의해 연결 된다. 컨트롤러는 Provisioning 메시지를 통해 사물인터넷 기기와 CoAP, HTTPS 등을 통해 정보를 전송하고, 사물인터넷 기기는 Transaction 메시지를 통해 컨트롤러에게 정보를 전송합니다.

사물인터넷 기기는 블록체인 네트워크에 직접적으로 접속해 있지 않는 아키텍처를 가지기에 사물인터넷 기기로 부터 전달 받은 데이터를 블록체인에 저장하는 기능, 필요한 데이터를 블록체인으로 부터 읽어 들이는 에이전트 역활을 수행한다.

3.5 SIPCoin: A Cryptocurrency for Preventing RoboCalling on the PSTN

본 기고서에서는 SIP 기반의 인터넷 전화망에서 악의적으로 사용되는 RoboCalling 을 방지하기 위한 수단으로 SIPCoin을 제안한다 [9]. SIPCoin은 기존 암호화폐와 비슷하게 이해 될 수 있지만, SIP 기반의 인터넷 전화망에 특화 된 특징을 가진다.

SIPCoin은 SIP 통화자의 익명성을 보장하면서도, 악의적으로 SIP 전화망을 사용하는 사용자에게 요금을 지불하도록 한다. 이를 위해 제안된 Create 트랜잭션과 Burn 트랜잭션을 통해 SIP 프로토콜과 블록체인을 연동 시킨다. 기고서에서는 SIPCoin이 실제 사용 될 때의 고려사항 분석과 함께 경재학적으로 SIPCoin을 사용함으로써 얻을 수 있는 장점을 분석하고 있다.

IV. IRTF 에서의 블록체인 기술

2017년 9월 시작 된 Decentralized Internet Infrastructure Research Group (DINRG)는 분산된 인터넷 아키텍처를 위한 연구 그룹이다. 초창기 인터넷은 분산된 아키텍처를 지향하였지만, 인터넷의 대중화로 인해 다양한 형태의 상업적 서비스가 개발되고 이용되다 보니 본래의 취지와는 다르게 중앙 집중적이고 계층화 된 서비스 구조를 가지게 되었다. 본 연구 그룹에서는 블록체인이라는 특정 기술을 연구하기 보다는 거시적인 관점에서 어떻게 하면 현재의 인터넷을 분산된 인터넷 구조로 가져 갈 수 있을지에 대한 연구를 진행하고 있다.

DINRG는 Dirk Kutscher과 Melinda Shore가 의장을 맞고 있으며, 현재까지 3번의 오프라인 회의를 가졌다.

- NDSS 2018 컨퍼런스 (February 17, 2018)
- IETF 101 회의 (March 19, 2018)
- IETF 102 회의 (July 20, 2018)

미국 센디에고에서 열린 NDSS 2018 컨퍼런스에서 DINRG는 16명이 모인가운데 처음으로 오프라인 회의를 가졌다. 오전 9시 30분 부터 시작된 회의는 오후 4시 30분까지 이어졌으며, 참석자들이 준비해 온 내용을 바탕으로 어떻게 하면 분산된 인터넷 아키텍처를 만들 수 있을 것이며, 무엇이 현재의 중앙 집중적이고 계층화 된 인터넷을 만들었는지에 대해서 의견을 교류하였다.

2018년 3월과 7월에 IETF 101 회의와 102회의에서 오프라인 회의를 이어갔으며, 참여자들의 높아져가는 관심속에 연구를 계속하고 있다. 현재까지 3개의 개인 기고서가 등록되어 있다.

- The Stellar Consensus Protocol (SCP) (March 5, 2018)
- Delegated Distributed Mappings (July 02, 2018)
- Blockchain-based IoT Infrastructure Functional Requirements (September 14, 2018)

4.1 The Stellar Consensus Protocol (SCP)

인터넷 환경에서 사용 가능한 합의 알고리즘을 제안하는 본 기고서는 2015년 Stellar Development Foundation 에서 발표한 백서 The Stellar Consensus Protocol: A Federated Model for Internet-level Consensus [10]을 바탕으로 한다. 비잔틴 합의 알고리즘을 채택한 블록체인은 선택된 특정 노드들에 의해 합의 과정을 이끌어내기 때문에 빠른 처리가 가능했지만, 중앙 집중화된 합의를 도출하고 있다. 이에 반해, 본 기고서에서 제안하는 스텔라 합의 프로토콜은 합의 과정중에 발생 할 수 있는 단합이나 악의적인 노드의 영향으로부터 최대한 벗어나기 위해 쿼럼(quorum) 설정에 노드들이 참여 할 수 있도록 한다. 이를 통해 탈중앙화된 합의 알고리즘을 이끌도록 하고 있다.

2018년 6월 29일 제출된 최근 기고서(draft-mazieres-dinrg-scp-04) [11]는 스텔라 합의 프로토콜의 전반적인 동작 과정과 메시지 포맷에 대한 명세를 포함하고 있다.

4.2 Delegated Distributed Mappings

본 기고서에서는 현재 인터넷 환경에서 중앙 집중적이고 계층화 된 형태로 제공되고 있는 위임 된 네임스페이스들 (delegated namespaces)에 대한 분산된 매핑 방안을 제안한다. 현재 우리가 사용하고 있는 도메인 네임(domain name), IP 주소 할당 (IP address allocation) 등은 분산된 형태로 제공 되지 않고, 특정 기관이나 서버에 의해 할당되고 관리되는 형태로 운영되고 있다. 이렇게 중앙 집중적이고 계층화 된 형태로 운영되는 이유는 인터넷 초기 단계에 관리의 편의성도 있었지만, 보안의 관점에서 하나의 인가된 기관이나 서버에서 운영하는 것이 유용했기 때문이다.

2018년 7월 2일 제출된 기고서(draft-watson-dinrg-delmap-00) [12]는 공개키 기반 서명을 통해 어떻게 네임스페이스(도메인 네임, IP 주소 등)를 합당한 기관에 매핑하고 위임 될 수 있는지에 대한 방법과 데이터 구조를 정의한다. 합의 알고리즘을 통해 위임 정보(delegation tables)에 대한 보안성과 일관성을 유지하게 되며, 본 기고서에서는 개별적인 합의 알고리즘을 개발하고 제시하기 보다는 기존 합의 알고리즘을 채택해서 사용 할 수 있도록 한다.

4.3 Blockchain-based IoT Infrastructure Functional Requirements

본 기고서에서는 자원이 한정적인 사물인터넷 기기들로 구성되는 환경에서 블록체인이 적용 될 때, 필요한 기능적 요구사항을 명세하고 있다 [13]. 필요한 기능적 요구사항으로 1) 사물인터넷 기기들에 대한 ID 관리, 2) 필요 서비스를 수행 할 사물인터넷 기기를 매칭하기 위한 방안, 3) 분산된 서비스 스케쥴링을 위한 정책과 합의 알고리즘, 4) 자동화 된 실행을 보장하기 위한 스마트 컨트랙트를 제안하고 있다.

V. 결론

IETF 에서는 아직까지 성숙한 단계의 블록체인 기술 표준화가 이루어지지 못하고 있는 실정이다. 블록체인 기술 자체를 고도화 하기 위한 작업반이 만들어 지거나 블록체인 기술을 활용한 인터넷 프로토콜 및 서비스의 개선을 위한 노력이 필요한 시점이다. 이와 더불어, 현재 우리가 알고 있는 블록체인 기술이 아닌 좀 더 거시적인 관점에서 분산된 인터넷 아키텍처를 연구하는 IRTF의 활동을 지켜 볼 필요가 있다.

Acknowledgement

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. 2018-0-00841, IoT 디바이스를 위한 Lightweight 블록체인 표준개발).

References

[1] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2008.
[2] J.-H. Lee, “BIDaaS: Blockchain based ID as a Service,” IEEE Access, vol. 6, pp. 2274-2278, February 2018.
[3] L. Luu et al., Making Smart Contracts Smarter, ACM CCS 2016, October 2016.
[4] G. Wood, “Ethereum: a secure decentralised generalised transaction ledger byzantium version”, June 2018.
[5] C. Grothoff et al., “Special-Use Domain Name for Namecoin,” draft-grothoff-iesg-special-use-p2p-bit-00, June 2015.
[6] S. Hommes et al., “ALTO for the blockchain,” draft-hommes-alto-blockchain-02, January 2017.
[7] J. Paillisse et al., “An analysis of the applicability of blockchain to secure IP addresses allocation, delegation and bindings,” draft-paillisse-sidrops-blockchain-02, June 2018.
[8] P. Urien, “Blockchain Transaction Protocol for Constraint Nodes,” draft-urien-core-blockchain-transaction-protocol-01, September 2018.
[9] J. Rosenberg and C. Jennings, “SIPCoin: A Cryptocurrency for Preventing RoboCalling on the PSTN,” draft-rosenberg-stir-sipcoin-00, March 2018.
[10] D. Mazieres, “The Stellar Consensus Protocol: A Federated Model for Internet-level Consensus,” Stellar Development Foundation whitepaper, 2015.
[11] N. Barry et al., “The Stellar Consensus Protocol (SCP),” draft-mazieres-dinrg-scp-04, June 2018.
[12] S. Li et al., “Delegated Distributed Mappings,” draft-watson-dinrg-delmap-00, July 2018.
[13] H. Ding and Z. Jiao, “Blockchain-based IoT Infrastructure Functional Requirements,” draft-ding-dinrg-biot-00, September 2018.

Biographies



이종혁
2009년 ~ 2012년: 프랑스 국립연구소 INRIA 연구원
2012년 ~ 2013년: 프랑스 그랑제꼴 TELECOM Bretagne 조교수
2013년 ~ 현재: 상명대학교 소프트웨어학과 조교수
주요 관심분야: 프로토콜 분석, 익스플로잇 분석, 블록체인
e-mail: jonghyouk@smu.ac.kr



박정수
1988년 ~ 1992년: 경북대학교 전자공학 석사
2005년 ~ 2013년: 경북대학교 전자공학 박사
1994년 ~ 현재: 한국전자통신연구원 책임연구원
주요 관심분야: 사물인터넷, 네트워크보안, 블록체인
e-mail: pjs@etri.re.kr

Article 6

ETRI
한국전자통신연구원
Electronics and Telecommunications Research Institute



제4차 산업혁명을 선도하는

ICT Innovator

ICT Innovator leading the 4th Industrial Revolution

ETRI의 최첨단 기술은
새로운 영역과 새로운 가치를 만들어내고 있습니다.
꿈꾸는 모든 것을 현실로 구현하는 기술,

ETRI의 첨단 ICT기술은
세상을 더욱 풍요롭게 만들어갑니다.