Designing Secure IoT (Internet of Things) Environments with SDN and NFV

김연근, 신승원
한국과학기술원

요 약

오늘날 우리 사회는 정보통신기술의 발전으로 다양한 가능성들을 갖게 되었다. 그 중 다양한 사물들에 센싱 기술을 부여하고 네트워크에 연결시켜, 해당 사물들이 처리하는 데이터들을 손쉽게 활용 및 관리 하고자 하는 기술인 사물 인터넷(Internet of Things)이 활발하게 거론되고 있다. 현재 사물 인터넷은 스마트 홈, 헬스 케어 등의 형태로 우리 생활에 깊게 들어왔고, 그곳에서 개인 정보와 같은 매우 민감한 데이터를 다루기 때문에, 이러한 기기 및 데이터들에 대한 보안은 매우 중요하다. 하지만 현재의 사물 인터넷은 보안이 충분히 고려되지 않은 상태이다. 본고에서는 현재 보안 기술을 적용함에 있어서 사물 인터넷 환경이 어떠한 문제점을 가지고 있는지 알아보고, 이러한 문제점들을 해결하고 안전한 IoT 환경을 구축하기 위해 본고에서 제안하는 SDN/NFV 기술을 적용한 IoT 게이트웨이 디자인을 알아본다.

Ⅰ. 서 론

최근 네트워크의 지속적인 발전으로, 사물의 지능화 및 네트워크화를 통한 유비쿼터스 사회의 실현이 빠르게 다가오고 있다. 특히 정보통신기술 (ICT; Information and Communication Technology)의 비약적인 발달과 함께 등장한 사물 인터넷(이하 IoT; Internet of Things)은 스마트 홈, 헬스 케어 등의 다양한 형태를 통해 우리 사회에 깊숙하게 들어와 실제로 우리의 생활을 크게 바꾸고 있다. IoT란 다양한 사물들이 네트워크에 연결되어 다른 사물이나 다른 시스템과 통신하고 데이터를 공유하는 생태계를 의미한다. 미국의 시장 조사 기관인 가트너(Gartner)에 따르면, IoT는 향후 10년간 유망 기술 중 하나이며[1], 통신/미디어 전문 시장조사 기관인 IDATE는 2020년까지 글로벌 커넥티드 단말의 수가 약 800억대에 이를 것으로 전망하였다[2].

이렇듯 끊임없이 성장하고 있는 IoT 환경에는 수많은 IoT 기기들이 존재하며, 이러한 IoT 기기들은 지속적인 센싱을 통해 수집한 개인 정보나 사생활 정보와 같은 민감한 데이터들을 처리하기 때문에 관리자는 보안을 위하여 각 기기들의 동작과 데이터들의 흐름을 파악 및 관리 할 필요가 있다. 하지만 관리자가 분산 되어있는 다수의 IoT 기기들을 하나하나 관리하기란 매우 어렵기 때문에, 분산된 IoT 기기들을 통합 관리할 수 있는 시스템이 필요한 실정이다. 실제로 IoT 환경을 통합 관리하기 위한 연구 및 제품들이 계속해서 나오고 있다. 예를 들어, 세계적인 네트워킹 기업인 프리스케일과 시스코에서는 분산된 IoT 기기들을 통합 관리하는 시스템의 구축을 위한 기기인 IoT 게이트웨이를 제작하고 있다[3][4]. 비록 관리자는 이러한 IoT 게이트웨이와 같은 기기들을 통해 분산된 IoT 기기들을 통합 관리할 수 있지만, 아직까지는 IoT 기기 자체에도, IoT 게이트웨이에도 보안이 충분히 고려되지 않았기 때문에, 현재의 IoT 환경은 여전히 보안 위협에 노출되어 있다.

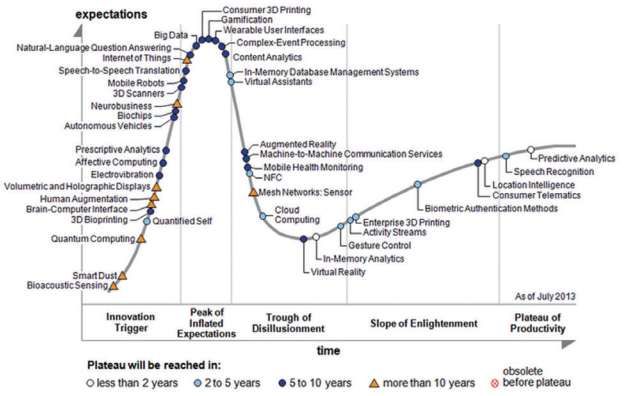


그림 1. 가트너의 미래 기술 하이프 사이클 (2013년)[1]

본고에서는 현재 보안 기술을 적용함에 있어서 IoT 환경에는 어떠한 문제점들이 있는지 기술하고, SDN/NFV 기술이 이러한 문제점들을 해결하고 안전한 IoT 환경을 구축하기 위해 어떻게 적용되는지 알아본다. 본고의 구성은 다음과 같다. 먼저 2장에서는 IoT 환경이 가지는 문제점과 이유를 살펴보고, 3장과 4장에서는 SDN과 NFV의 개념 및 구성을 이해한다. 5장에서는 안전한 IoT 환경의 구축을 위해 본고에서 제안하는 SDN/NFV 기

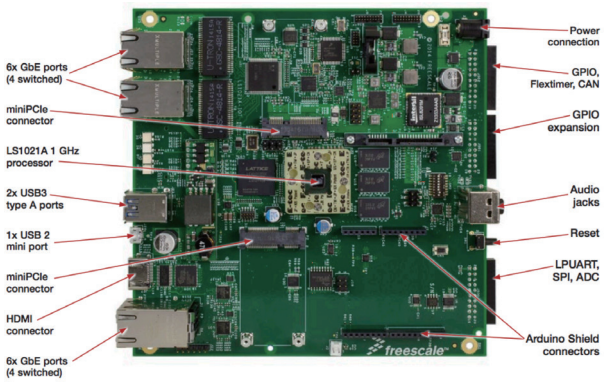


그림 2. Freescale의 IoT 게이트웨이[3]

반의 IoT 게이트웨이 디자인을 살펴본다.

Ⅱ. IoT 환경의 문제점

IoT는 홈/가전, 건강, 산업, 헬스케어 등의 다양한 분야에서 활용되고 있다. 각 분야에서 IoT 기기들은 각자의 목적을 가지고 설치되며, 자신이 부착된 주변 환경으로부터 자신의 목적에 맞는 데이터를 끊임없이 수집하고, 수집한 데이터를 네트워크를 통해 다른 기기 또는 관리자에게 전송한다. IoT 환경에서 다뤄지는 데이터들은 대부분 개인의 건강정보 또는 개인정보와 같은 민감한 데이터이기 때문에 보안의 중요성이 굉장히 크다. 예를 들어, 헬스 케어 기기의 경우 개인의 건강 정보 등을 지속적으로 기록하고 전송하며, 스마트 가전 기기의 경우 전력 사용량을 측정하거나 영상 녹화 및 전송 등의 개인 정보를 처리한다. 하지만 주목할 점은, 실제로 많은 IoT 기기들이 보안기능을 가지고 있지 않거나 아주 간단한 암호화 기술만을 지원한다는 것이다. 이것은 반드시 해결되어야 할 문제점이지만, IoT 환경의 몇 가지 특징들을 고려했을 때 현재의 보안 기술을 IoT 환경에 그대로 적용하기란 매우 어려우며, 그 이유들을 살펴보면 다음과 같다.

1. 관리의 한계

IoT 환경에는 무수히 많은 종류의 IoT기기들이 존재한다. 안전한 IoT 환경을 구축하기 위해서는 이러한 많은 수의 기기들이 어떻게 동작하고 있는지, 어떠한 데이터들이 언제, 어디로 전송 되고 있는지 모니터링할 필요가 있다. 하지만 관리자가 많은 수의 IoT 기기들을 하나하나 확인하고 관리하는 것은 매우 어려운 일이다. 심지어 같은 종류의 기기라 하더라도, 서로 다른 목적을 가지고 설치될 수 있기 때문에, 이에 대한 고려 또한 필요하다. 다시 말하자면, IoT 환경에는 서로 다른 보안 요구사항들을 가지는 수많은 기기들이 공존하고 있으므로, 관리자는 각 IoT 기기들이 가지는 보안 요구 사항을 파악하고 해당 요구사항에 맞는 보안 기능들을 제공해야 한다. 이는 관리자가 자신의 IoT 환경에 설치된 수많은 기기들의 목적과 보안 요구사항을 일일이 확인하고, 이에 적합한 보안 기술을 찾아 일일이 제공해야 하는 엄청난 부하를 가지고 있음을 뜻한다.

설령, 관리자가 모든 기기들이 각기 필요로 하는 보안 기술들을 파악하고 제공하고자 하더라도, 현재의 보안 장비들을 IoT환경에 설치하는데 있어서 한계를 가지기 때문에 실현하기 어렵다. 예를 들어, IoT 기기들에게 원치 않는 네트워크 접근을 막을 수 있는 네트워크 방화벽 기능을 제공하고자 하는 경우, 구축된 IoT환경과 외부 네트워크가 연결되는 지점인 게이트웨이에 방화벽 장비를 설치함으로써 외부 네트워크로부터의 원치않는 접근을 막을 수 있지만, 이것 만으로는 네트워크 내부로부터의 원치않는 접근을 막을 수 없다. 결국 관리자는 방화벽 기능을 필요로하는 수많은 IoT 기기들 각각에 방화벽 장비를 설치하여 기능을 제공해야 하지만, 이는 비용적 문제를 야기할뿐더러 관리자가 개별적으로 관리해야 하는 네트워크 구성 요소들이 증가함을 뜻한다. 또한, 보안 장비를 설치하더라도, 이동성을 가지는 IoT 기기들의 경우 해당 장비의 관리 범위를 벗어날 수 있기 때문에 이에 대한 대처 또한 필요한 실정이다.

2. 자원의 한계

이러한 관리상의 한계를 극복하기 위한 가장 좋은 해결책은 각각의 IoT 기기들이 자신의 보안 요구사항에 맞는 보안 기능을 스스로 지니는 것이다. 이것은 관리자가 각 IoT 기기들의 보안 요구사항을 확인하는 부하를 줄여주고, 보안 장비의 추가적인 설치를 필요로 하지 않기 때문에 비용상의 문제 또한 해결해 준다. 하지만 이 해결책은 저전력, 저성능이라는 또 다른 IoT 환경의 특징을 상기했을 때 실현하기 어렵다는 것을 알 수 있다. 대부분의 IoT 기기들의 경우 낮은 컴퓨팅 파워와 작은 메모리를 가지고 제작되는데, 이는 센싱 및 데이터 전송 등에 필요한 최소한의 자원만을 제공하여 낮은 단가로 제작하기 때문이다. 하지만 서비스 거부 공격(Denial of Service) 탐지기나 침입 탐지 시스템(IDS)과 같은 대다수의 보안 기능들은 높은 컴퓨팅 파워와 큰 메모리 등의 충분한 자원을 필요로 하기 때문에, 저성능의 IoT 기기들에 이러한 고성능의 보안 기능을 추가하는 것은 매우 어렵다.

설령 충분한 자원을 가지고 있는 IoT 기기들에 보안 기능들을 추가하고자 하더라도, 대부분의 IoT 기기들은 현재 구현된 보안 기능들과는 다른 컴퓨팅 환경에서 설계되었기 때문에, 해

당 환경에 맞게 기능을 새로 구현해야 하는 문제가 있다. 심지어 IoT 기기들 간에도 설계된 컴퓨팅 환경이 다르기 때문에, 추가하고자 하는 보안 기능들을 IoT 기기마다 새로 구현해야 하는 포팅(Porting) 문제 또한 존재한다. 이러한 문제를 해결하기 위해 IoT 기기 제작 단계에서 보안 기능들을 넣고자 하더라도, 제작자가 해당 기기가 활용되는 범위를 모두 예측하여 그에 적합한 모든 보안 기능을 넣는 것은 어려우며, 관리자가 자신의 IoT 환경에 특화된 새로운 보안 기능을 추가하고자 할 때 똑같은 문제에 부딪히게 된다.

3. 접근 제어의 한계

IoT 기기들은 데이터를 지속적으로 수집하고 처리하며, 네트워크를 통해 서로 데이터를 주고 받는다. IoT 기기들 간에 전달되는 데이터의 종류는 매우 다양하며, 각 종류에 따라 데이터의 중요성 또한 다르다. 다시 말하자면, IoT 기기가 어떠한 데이터를 어느 기기와 주고 받는지에 따라 보안 요구사항이 달라지게 된다. 이는 침입 탐지 시스템(IDS) 등의 악의적인 접근을 차단하는 보안 기능뿐 아니라, 정상적인 IoT 기기와의 통신이라 할지라도 처리되는 데이터의 중요성과 해당 IoT 기기의 역할에 따라 접근이 제어되는 접근 제어 기능(Access control) 또한 필요함을 의미한다.

예를 들어, 일반 사용자(환자)의 헬스 케어 기기가 병원 네트워크에 연결된 상황을 가정하자. 해당 기기는 환자의 신체에 부착되어, 내장된 센싱 기술을 통해 심박수를 체크하거나 혈압을 재는 등의 기능을 수행함으로써 사용자의 건강 데이터를 수집한다. 이후 해당 기기는 수집된 데이터를 병원 네트워크를 통해 환자의 전담의에게 전달함으로써 환자가 수집된 데이터를 바탕으로 원격진료를 받을 수 있게 한다. 이 데이터는 민감한 개인정보이기 때문에, 해당 데이터로의 접근이 허용된 사용자, 즉 환자의 전담의로부터의 접근만을 허용하고, 다른 환자나 의사와 같이 접근이 허용되지 않은 사용자로부터의 접근을 제어해야 한다. 또한, 전담의라 하더라도 허용된 데이터 이외의 개인정보로의 접근을 제어할 필요가 있다.

하지만 현재 IoT 환경에서는 관리자가 사용자들간의, 데이터들간의, 혹은 사용자와 데이터간의 연관성을 알기 어려울뿐더러, 각 사용자가 자신의 보안 요구사항을 전달할 수 있는 방안 또한 존재하지 않아 세밀한 접근 제어가 이뤄지지 않고 있다. 설령 관리자가 이러한 연관성을 파악하더라도, 현재 상황에 맞는 세밀한 접근 제어 규칙을 생성하여 보안 장비들 또는 IoT 기기들에 동적으로 설치하는 것 또한 실현하기 어려울 뿐 아니라, 대부분의 IoT 기기들이 이러한 접근 제어 기능을 지니고 있지 않다.

4. 전문 지식의 한계

일반적으로 네트워크 보안 기능들은 관리자에 의해 정의된 규칙에 따라 보안 기능을 수행한다. 예를 들어, 침입 방지 시스템(IPS)의 경우, 관리자가 차단 하고자 하는 침입들에 대한 규칙을 정의하면, 정의된 규칙에 따라 패킷들을 검사하여 침입 시도를 탐지하고 차단한다. 방화벽의 경우, 관리자는 차단하고자 하는 기기 (IP)와 서비스 (Port)별로 규칙을 설정하고, 전체 네트워크의 구조를 파악한 후 우회 가능한 경로까지도 차단해야 한다. 기존의 네트워크의 경우 전문적인 지식을 가진 관리자들에 의해 이러한 규칙들이 정의되었다. 하지만 IoT 환경의 경우 우리의 일상 생활과 굉장히 밀접해 있기 때문에, 스마트 홈과 같은 환경을 고려한다면 전문적인 지식을 가진 관리자가 아닌 일반 가정과 같은 비전문적인 관리자들이 충분히 존재할 수 있으며, 이러한 관리자들은 전문적인 지식을 필요로 하는 네트워크 보안 기능들을 제대로 활용하기 어렵다는 문제가 있다.

5. 해결 방안

안전한 IoT 환경을 구축하기 위해서는 위와 같은 문제점들을 해결해야 하지만, 앞서 설명한 바와 같이 현재의 보안 장비들은 IoT 환경에서 한계점들을 지니기 때문에, 해당 보안 장비들을 IoT 환경에 그대로 적용하는 것은 바람직하지 않다. 이를 해결하기 위해, 본고에서는 1) 수많은 네트워크 장비들을 중앙 집중화된 컨트롤러에서 관리 및 제어 하는 SDN기술과, 2) 다양한 네트워크 기능들을 가상화 하여 유연한 관리를 가능하게 하는 NFV 기술을 활용함으로써, SDN/NFV 기반의 IoT 게이트웨이를 제안하고, 이를 통해 안전한 IoT 환경을 구축하고자 한다.

Ⅲ. SDN의 개념 및 구성

SDN/NFV 기술을 활용한 안전한 IoT 환경의 구축 방향을 제안하기에 앞서, 먼저 SDN이 어떤 기술인지를 알아보고, 4장에서는 또 다른 핵심 기술인 NFV에 대해 알아보고자 한다.

1. 기존 네트워크 환경의 문제점

SDN의 개념을 설명하기 앞서 기존 네트워크가 어떻게 동작하는지 살펴보자. 스위치와 같은 기존의 네트워크 장비의 경우, 패킷을 처리하기 위한 결정을 내리는 제어 평면(control plane)과 내려진 결정에 따라 패킷을 처리하는 데이터 평면(data plane)으로 구성된다. 패킷이 스위치에 도착하면, 먼저 스위치의 데이터 평면은 해당 패킷에 대한 처리 규칙(이하 플로우 룰;

flow rule)이 존재 하는지 확인하고, 만약 플로우 룰이 존재하지 않는다면 해당 패킷의 정보를 제어 평면에 전달한다. 제어 평면은 사전에 정의된 결정 알고리즘 또는 결정 규칙에 따라 전달된 패킷에 대한 플로우 룰을 생성하여 설치하고, 스위치는 새로 설치된 플로우 룰에 따라 이후 패킷들을 처리한다.

이처럼 패킷에 대한 처리를 결정하는 부분이 각각의 네트워크 장비들에 분산되어 있기 때문에, 네트워크 관리자는 자신의 네트워크에 맞는 결정 규칙을 설치하기 위해 네트워크를 구성하는 장비들에 일일이 접근해야 하는 단점이 있다. 게다가, 대부분의 네트워크 장비들은 공개되어 있지 않을뿐더러 복잡한 구조를 가지고 있기 때문에, 관리자가 자신의 네트워크에 맞게 기존의 기능들을 수정하거나 새로운 기능을 추가하기 매우 어렵다는 단점 또한 존재한다. 이러한 단점을 극복할 수 있는 기술이 SDN이다.

2. SDN의 개념

SDN이란 소프트웨어 정의 네트워크 (Software-defined Networks)의 약자로, 기존의 분산된 네트워크 구조가 아닌 논리적 중앙 집중화된 네트워크 구조를 일컫는다. 논리적 중앙 집중화된 네트워크 구조란, 기존의 네트워크 장비들의 제어 평면을 논리적으로 하나의 장비에 집중화시킨 구조를 의미하며, 이 논리적 중앙 집중화된 제어 평면은 SDN 컨트롤러라 불린다. 결국, SDN 구조에서는 네트워크 장비들이 전체 네트워크의 하나의 논리적 데이터 평면이 되며, SDN 컨트롤러 가 전체 네트워크의 하나의 논리적 제어 평면이 된다.

3. SDN의 구조

SDN의 개념에서 언급 하였듯이, SDN은 크게 네트워크 장비들의 계층(infrastructure layer)과 SDN 컨트롤러의 계층으로 나뉜다. 또한 SDN 컨트롤러는 크게 2개의 계층으로 나뉜다. 먼저 컨트롤 계층(control layer)은 SDN 컨트롤러의 코어 부분으로, 오픈 플로우 (OpenFlow) 프로토콜[6]을 통해 네트워크 장비들을 제어하는 기능과, 전체 네트워크의 정보를 추상화하는 기능을 제공한다. 이러한 제어 및 접근 기능들은 API 형태로 정의되어 있으며, 프로그래밍 가능한 상위 계층인 어플리케이션 계층(application layer)에 존재하는 프로그램들은 컨트롤 계층에 정의된 API들을 이용하여 전체 네트워크를 관리한다. 이러한 프로그램들은 SDN 어플리케이션 이라 불리며, 네트워크 관리자는 컨트롤 계층에서 제공하는 API들을 이용하여 손쉽게 SDN 어플리케이션을 제작할 수 있다.

이처럼 전체 네트워크의 정보가 하나의 제어 평면으로 전달되고, 하나의 제어 평면에서 전체 네트워크의 모든 데이터 평면을

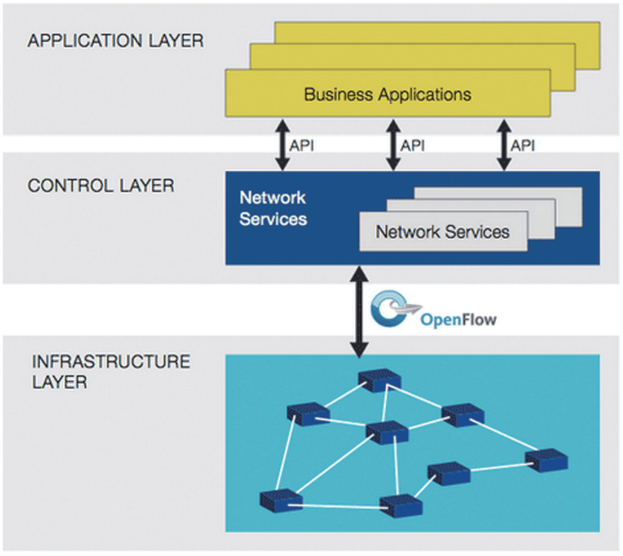


그림 3. SDN의 구조 [5]

관리하기 때문에, SDN은 기존의 네트워크에 비해 각 네트워크 장비들을 관리하기 용이할 뿐 아니라, SDN 어플리케이션을 통한 플로우 단위의 세밀한 네트워크 제어까지도 가능하게 한다. 또한 네트워크를 제어하는 SDN 어플리케이션은 프로그래밍이 가능하기 때문에, 네트워크 관리자는 언제든지 자신의 네트워크에 맞게 새로운 기능을 추가하거나 기존의 기능을 수정할 수 있다.

Ⅳ. NFV 의 개념 및 구성

1. 기존 네트워크 환경의 문제점

기존의 네트워크 환경의 경우, 네트워크 관리자는 특정 네트워크 기능을 제공하는 네트워크 장비를 구입하고 설치함으로써 해당 기능을 네트워크에 제공한다. 이때 네트워크 관리자는 자신의 네트워크의 구성을 고려하여, 해당 기능을 필요로 하는 패킷들이 적절한 장비를 지나갈 수 있는 위치를 찾아 장비를 설치해야 한다. 이는 네트워크가 커질수록, 복잡해질 수록 더 많은 수의 장비를 필요로함을 의미하며, 비용적 문제를 야기함을 뜻한다. 뿐만 아니라, 새로운 네트워크 기능이 필요하다면, 네트워크 관리자는 해당 기능을 제공하는 또 다른 네트워크 장비를 구입하여 설치해야하며, 이 또한 비용적 문제로 이어진다. 또한 이러한 장비들의 경우 대부분 공개되지 않았기 때문에 네트워크 관리자가 자신의 네트워크에 맞게 기능을 수정할 수 없다는 한계점 또한 존재한다. 이러한 문제들을 풀 수 있는 해결책이 NFV 기술이다.

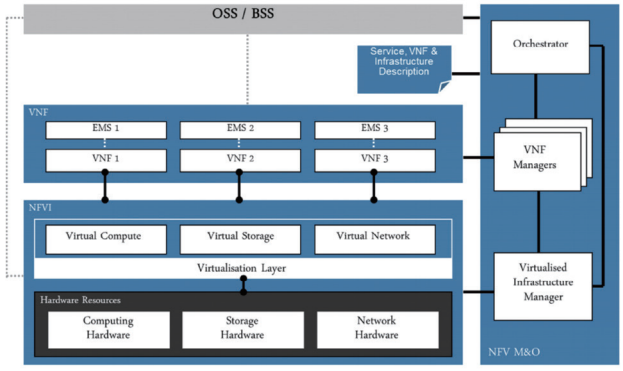


그림 4. NFV의 구조 [7]

2. NFV의 개념

NFV는 네트워크 기능 가상화 (Network Function Virtualization)의 약자로, 다양한 네트워크 기능들을 사용하기 위해 전용 하드웨어 장비들을 설치하는 대신에, 각각의 네트워크 기능들을 가상화하여 서버 OS등에 설치하여 사용하는 기술을 의미한다. 네트워크 관리자는 필요한 네트워크 기능이 설치된 하드웨어 장비를 구입하여 설치하는 대신에, 해당 기능이 구현된 소프트웨어를 서버나 스위치 등에 설치함으로써 네트워크에 해당 기능을 제공할 수 있다. 이러한 가상화된 네트워크 기능들은 하이퍼바이저 (HyperVisor) 위에서 독립적으로 작동하기 때문에 서로 다른 네트워크 기능들 간의 충돌을 방지하며, 다양한 네트워크 기능들이 하나의 하드웨어 기기에서 작동하기 때문에 서비스 및 자원 관리가 매우 유연하다는 장점이 있다.

3. NFV의 구성

〈그림 4〉에서 보여지듯이, NFV는 크게 3개의 부분으로 구성된다. 먼저 VNF(Virtualized Network Function)은 네트워크 기능을 수행하는 소프트웨어를 지칭한다. 설치된 VNF들은 NFVI 위에서 실행되며, NFVI(Network Function Virtualization Infrastructure)는 하드웨어 자원과 가상화 자원을 NFV M&O의 지시에 따라 VNF에 제공하는 역할을 한다. NFV M&O (Management and Orchestration)는 하드웨어 자원과 가상화 자원을 관리 및 전달하고, VNF들을 관리하는 역할을 수행한다.

4. SDN과의 관계

앞서 설명한 두 개념은 매우 중요한 관계를 가지고 있다. 먼저 NFV는 SDN에 매우 보완적이다. 하지만 NFV는 SDN에 의존하지 않으며, 이것은 SDN 또한 마찬가지이다. NFV는 SDN 기술 없이도 구현이 가능하지만, 두 개념을 합친다면 더 큰 가치를 발휘할 수 있다.

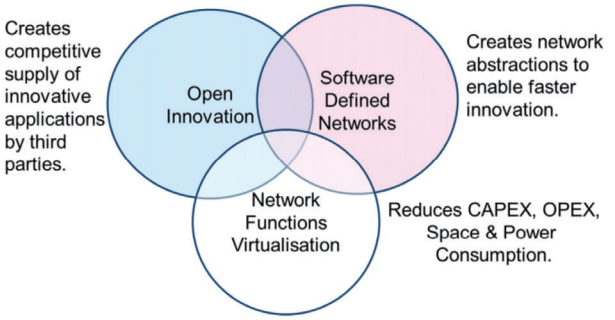


그림 5. SDN과 NFV의 관계도 [7]

V. SDN/NFV 기반의 IoT 게이트웨이

본고에서는 안전한 IoT 환경의 구축을 위해 현재 IoT 환경을 관리하기 위한 제품인 IoT 게이트웨이에 SDN/NFV 기술을 적용함으로써, IoT 게이트웨이가 SDN 컨트롤러의 역할을 수행하게 하여 전체 IoT 환경을 관리하게 한다. 더불어 다양한 네트워크 보안 기능들을 SDN 어플리케이션 형태로 제작하고 설치하여, SDN 컨트롤러, 즉 IoT 게이트웨이가 해당 기능들을 관리함으로써 IoT 기기들이 가지지 못하는 보안 기능들을 유연하게 제공하고자 한다. 이외에도 SDN/NFV 기술을 함께 활용함으로써 기존의 IoT 환경에서 해결할 수 없었던 보안에서의 문제점을 해결하고자 한다. 본고에서 제안하는 SDN/NFV 기반의 IoT 게이트웨이의 구조는 〈그림 6〉과 같으며, 해당 디자인을 통해 얻을 수 있는 보안 기능들은 다음과 같다.

1. 중앙 집중화된 네트워크 제어 및 모니터링

기존의 분산된 IoT 환경의 경우, 관리자가 수많은 기기를 일일이 관리하기 매우 어렵다는 문제가 있다. 하지만 SDN 기반의 중앙 집중화된 구조를 현재의 IoT 환경에 적용한다면, 각각의 기기들에 대한 관리뿐 아니라 전체 네트워크 내부의 패킷들에 대한 관리 및 모니터링까지도 가능하다.

앞서 SDN 개념에서 살펴본 바와 같이, SDN 에서는 패킷에 대한 처리 규칙, 즉 플로우 룰이 각각의 스위치에서 생성되지 않고, 논리적 중앙 집중화된 SDN 컨트롤러에서 일괄적으로 생성되고 설치된다. 이는 전체 네트워크에 대한 제어권이 중앙의 SDN 컨트롤러에게 있음을 의미한다. 또한 플로우 룰이 설치된 패킷들의 경우 스위치는 해당 패킷들의 흐름 정보를 자신의 플로우 통계 정보에 기록하고, 이후 오픈 플로우(OpenFlow) 프로토콜을 통해 SDN 컨트롤러에 전달함으로써, SDN 컨트롤러의 전체 네트워크 내의 패킷들에 대한 모니터링을 가능하게 한다.

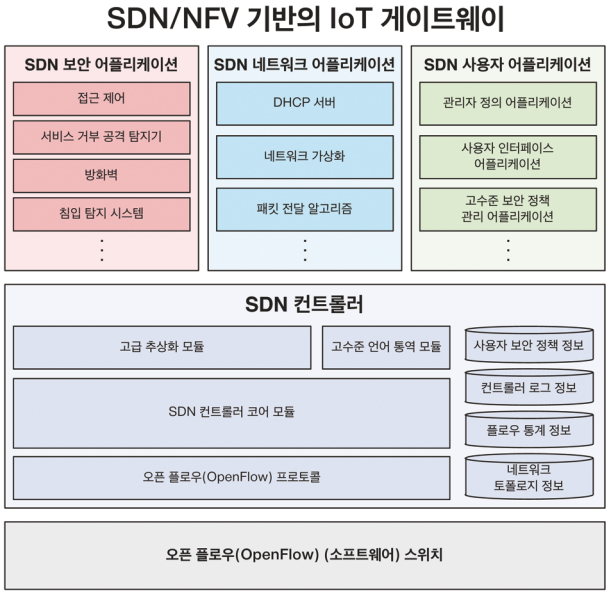


그림 6. SDN/NFV 기반의 IoT 게이트웨이 디자인

SDN은 이러한 네트워크 중앙 제어 및 모니터링 기능을 실현하기 위해 중앙 집중화된 형태의 장비를 필요로 한다. 본고에서는 현재 IoT 기기들의 통합 관리하기 위해 연구 및 개발되고 있는 IoT 게이트웨이를 활용함으로써 이러한 필요성을 충족하고자 한다.

2. 네트워크 보안 어플리케이션

IoT 기기들은 최소한의 자원만을 가지고 있기 때문에 고성능의 네트워크 보안 기능들을 내장하기 어렵다. 또한 보안 장비를 구입하더라도, 많은 수의 IoT 기기들 때문에 설치상의 한계점이 존재한다. 하지만 SDN/NFV 기술을 통해 다양한 네트워크 보안 기능들을 SDN 어플리케이션의 형태로 구현함으로써 이러한 문제를 해결할 수 있다.

SDN 어플리케이션은 프로그래밍이 가능한 소프트웨어이기 때문에, 관리자들은 IoT 기기들에게 다양한 보안 기능들을 제공하기 위해 별도의 장비를 따로 구입할 필요 없이 SDN컨트롤러가 제공하는 다양한 API들을 활용하여 방화벽, 침입 탐지/방지 시스템 (IDS/IPS), 서비스 거부 공격(Denial of Service) 탐지기 등과 같은 여러 보안 어플리케이션을 구현할 수 있다. 이렇게 구현된 보안 어플리케이션들은 SDN 컨트롤러에 의해 관리되며,네트워크에 새로운 패킷이 들어왔을 때 구현된 알고리즘에 따라 악의적인 패킷들을 탐지 및 차단함으로써 IoT 기기들에게 다양한 보안 기능을 제공할 수 있다. 또한 앞서 설명한 바와 같이, SDN 컨트롤러는 모든 패킷들에 대한 제어권을 가지며, 이는 SDN 어플리케이션 또한 모든 패킷들을 제어할 수 있음을 의미하기 때문에, 기존의 네트워크 장비 설치상의 한계점 또한 극복 가능하다.

3. 세밀한 네트워크 접근 제어 기능

기존의 IoT 환경에서는 관리자가 사용자 및 데이터간의 연관성을 알기 어렵고, 현재 상황에 맞는 접근 제어 규칙을 동적으로 생성하고 설치할 수 없기 때문에 세밀한 접근 제어가 잘 이루어지지 않는다. 하지만 SDN/NFV 기술의 플로우 단위의 네트워크 제어 기술과 프로그래밍 가능성을 통해 이러한 문제점을 해결할 수 있다.

SDN은 네트워크 전체에 대한 전망을 제공하기 때문에 관리자는 현재 네트워크 상황을 쉽게 파악할 수 있다. 더불어, IoT 기기들로부터 보안 요구사항을 전달 받는 SDN 어플리케이션을 통해 관리자는 IoT 기기들의 서로 다른 보안 요구사항을 알고, 이를 바탕으로 플로우 단위의 세밀한 네트워크 접근 제어 규칙을 동적으로 생성하고 설치함으로써, 하나의 기기에서도 다양한 상황에 맞는 다양한 종류의 접근 제어를 가능하게 한다.

4. 고급(High-level) 추상화 기능

대부분의 네트워크 보안 장비들은 전문적인 지식을 필요로 하기 때문에, 그렇지 않은 관리자들이 해당 장비들을 활용하기가 어렵다. 하지만 SDN 기술은 고급 추상화 기능을 제공하기 때문

표 1. 현재 IoT 환경의 문제점 및 SDN/NFV 기술을 이용한 해결 방안 요약

IoT 환경의 문제점	보안 요구사항	SDN/NFV 기술의 특징	적용 방안
각기 다르게 작동하는 많은 수의 기기로 인한 관리의 한계	모든 기기들에 대한 통합 관리 및 제어 기능	SDN 컨트롤러를 통한 중앙 집중화 된 네트워크 관리 및 제어	개발된 IoT 게이트웨이에 SDN 컨트롤러를 설치
저성능, 저전력 IoT 기기들이 가지는 자원의 한계	다양한 고성능 장비의 설치를 통한 보안 오프로딩 기능	NFV의 소프트웨어 형태로의 다양한 네트워크 보안 기능 가상화 및 관리 기능	다양한 보안 기능들을 SDN 어플리케이션으로 구현하여 SDN 컨트롤러에 설치
세분화된 데이터들에 대한 접근 제어의 한계	보안 정책 인터페이스 및 세밀한 접근 제어 기능	SDN의 플로우 단위의 네트워크 제어 기능 및 프로그래밍 가능성	IoT 기기들의 보안 정책 관리 SDN 어플리케이션 구현
IoT 환경 관리자의 전문 지식의 한계	고급 추상화 기능 및 고급 언어 지원 기능	SDN의 고급 추상화 기능 및 프로그래밍 가능성	사용자 인터페이스 및 고급 언어 통역 기능 구현

에, 일반 가정과 같은 비전문적인 관리자들도 다양한 보안 기능들을 사용할 수 있다.

SDN 컨트롤러는 전체 네트워크의 정보나 전달되는 패킷들의 정보를 그대로 제공하지 않고 추상화하여 제공하기 때문에, 관리자는 저급(Low-level) 네트워크 요소를 알고 있을 필요가 없이, 사용자 인터페이스 등을 통해 SDN 컨트롤러가 제공하는 추상화 및 도식화된 데이터를 얻음으로써 네트워크의 구조 및 흐름을 쉽게 파악할 수 있다. 또한, 이러한 특성을 확장하여 고급 언어로 작성된 보안 정책들을 실제 보안 기능들에 설치되는 저급 언어로 통역해주는 프로그램을 구현하고 컨트롤러에 설치함으로써, 비전문적인 관리자들도 구현된 네트워크 보안 기능들을 충분히 손쉽게 활용할 수 있게 한다.

Ⅵ. 결 론

본고에서는 현재 큰 관심을 받고 활발하게 연구 및 개발이 이뤄지고 있는 사물 인터넷 (IoT) 환경에 현재의 보안 기술을 적용하기에 어떠한 문제점들이 있고, 이 문제점들이 발생한 이유를 기술하였다. 또한 이러한 문제점들을 해결하기 위해 활용하고자 하는 SDN/NFV 기술의 개념을 설명하고, 해당 기술들이 어떻게 IoT 환경에서의 보안 문제점들을 해결할 수 있는지를 기술하고, 이것들을 [표1]에 정리하였다. 또한, SDN/NFV 기술을 활용한 IoT 보안 게이트웨이의 디자인을 제안함으로써 안전한 IoT 환경의 구축 방향을 제안하였다. 비록 IoT 환경에는 본고에서 언급한 보안 문제점 이외의 문제점도 존재하지만, 본고에서 제안한 SDN/NFV 기반의 IoT 보안 게이트웨이를 통해 안전한 IoT 환경 구축에 한 발 다가갈 수 있을 것으로 기대한다.

참 고 문 헌

[1] J. Rivera, R. Meulen, "Gartner's 2013 Hype Cycle for Emerging Technologies Maps Out Evolving Relationship Between Humans and Machines", Gartner Newsroom , 2013, http://www.gartner.com/newsroom/id/2575515

[2] M2M.World.News, "IDATE forecasts 80 Billion things connected in 2020", M2M World News, 2013, http://m2mworldnews.com/2013/09/18/27009-idate-forecasts-80-billion-things-connected-in-2020/

[3] Freescale, "LS1021A-IoT Gateway Reference Design" http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=LS1021A-IoT

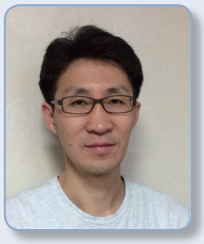
[4] Cisco, "Cisco 910 Industrial Router", 2014, http://www.cisco.com/c/dam/en/us/products/collateral/routers/900-series-industrial-routers/at-a-glance-c45-732146.pdf

[5] ONF, "Software-Defined Networking: The New Norm for Networks", Open Networking Foundation, 2012

[6] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner, "OpenFlow: enabling innovation in campus networks", SIGCOMM Comput. Commun. Rev. 38, 2 (March 2008), 69-74.

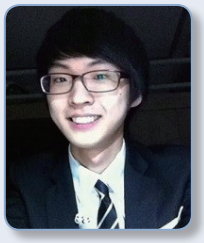
[7] C. Cui et al.,"Network Function Virtualisation", Introductory White Paper, ETSI, 2013

약 력



신 승 원

1998년 한국과학기술원 공학 학사
2000년 한국과학기술원 공학 석사
2000년~2002년 티맥스소프트 책임연구원
2002년~2005년 ETRI 연구원
2005년~2006년 MIT 방문 연구원
2006년~2009년 티맥스소프트 수석보 연구원
2011년 SRI International 인턴
2012년 SRI International 인턴
2013년 Texas A & M University 공학 박사
2013년~현재 한국과학기술원 전산학부 정보보호대학원 조교수
관심분야: SDN, NFV, Security, Android, Cloud



김 연 근

2014년 울산과학기술대학교 공학 학사
2014년~현재 한국과학기술원 석사 과정