

RSU를 위한 WAVE 네트워킹 서비스 프로토콜 개발

종신회원 주 흥 택^{*o}, 준회원 고 강 민^{**}, 권 동 우^{***}

Development of WAVE Networking Service Protocol for RSU(Road-side Unit)

Hongtaek Ju^{*o} *Lifelong Member*, Kangmin Ko^{**}, Dongwoo Kwon^{***} *Associate Members*

요 약

이 논문에서는 지능형 교통 시스템(ITS)을 위해 IEEE 1609.3 표준 문서에서 규정한 WAVE 네트워킹 서비스를 위한 표준 프로토콜 개발 결과를 제시한다. 그 중에서 특히 WAVE Management Entity(WME)와 WAVE Short Message Protocol(WSMP) 개발 결과에 중점을 두고 설명한다. 검증을 위해서 Traffic Management System(TMS)과 RSU 관리자의 개발 결과와 이를 이용한 교통사고 발생 시나리오에서의 TMS의 동작을 통해 개발 결과에 대한 검증 결과도 제시한다.

Key Words : WAVE, Networking Service Protocol, WME, WSMP, RSU

ABSTRACT

In this paper, we analyze and develop standard protocols for WAVE Networking Services described by IEEE 1609.3 which is one of standards for Intelligent Transportation Systems. In particular, we develop 'WAVE Management Entity(WME)' and 'WAVE Short Message Protocol(WSMP)'. For verification of development, we also develop Traffic Management System and RSU Manager, then we present results of verification by Traffic Management System in the car accident scenario.

I. 서 론

증가하는 교통 혼잡에 대비하기 위해서 지능형 교통 시스템(Intelligent Transportation Systems, ITS)이 보급되고 있다. ITS는 교통의 운영과 효율을 도모하고 차량 이용자에게 안전과 편의를 제공한다.

지능형 교통 시스템에서 단거리 통신을 위해 사용되는 Dedicated Short-Range Communications(DSRC)는 노변기지국(Road-Side Equipment, RSE)와 차량탑재단말(On-Board Equipment, OBE) 사이

또는 차량탑재단말들 사이의 근거리 무선통신을 통해 메시지 교환을 위한 시스템이다. DSRC의 핵심 부분인 Wireless Access in Vehicular Environments(WAVE)는 차량 간의 MAC과 네트워크 계층의 통신을 구현하는 표준 프로토콜을 지칭한다.

WAVE는 IEEE 1609 표준 문서를 통해 표준화 되어 있다. 이 중 1609.1은 리소스 매니저, 1609.2는 애플리케이션과 Management 메시지의 보안 서비스, 1609.3은 네트워킹 서비스, 1609.4 멀티채널 오퍼레이션에 관해 기술되어 있다³⁻⁶⁾. 이 논문에서

※ 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2011-0027438)

* 계명대학교 컴퓨터공학과 교수 (juht@kmu.ac.kr), ** 계명대학교 컴퓨터공학과 컴퓨터네트워크실습실 (koko4808@kmu.ac.kr),

*** 포항공과대학교 정보전자융합공학부 박사과정 (dwkwon@postech.ac.kr), (° : 교신 저자)

논문번호 : KICS2011-12-625, 접수일자 : 2012년 12월 21일, 최종논문접수일자 : 2012년 5월 2일

는 Road-Side Unit(RSU)를 위하여 IEEE 1609.3 표준 문서에 기술 되어 있는 WAVE 네트워킹 서비스 프로토콜을 구현하고 그 결과를 제시한다. 그 중에서 특히 WAVE Management Entity(WME)와 WAVE Short Message Protocol(WSMP)을 분석하고 구현하였다. 구현된 프로토콜의 검증은 RSU 관리자와 도로의 상황을 시뮬레이션 하는 Traffic Management System(TMS)을 개발하여 테스트 보드에 탑재한 후 실시하였다.

현재 지능형 교통 시스템이 보급 단계에 있기 때문에 아직 다양한 애플리케이션이 개발되지 않은 상태이다. WAVE 프로토콜 스택의 경우는 차후 상용화를 위해 일부에서 이미 개발이 완료된 상태이지만 연구적 관점에서 WAVE 표준 프로토콜의 구조와 기능을 분석하고 구현한 결과는 아직 제시되지 않았다.

이 논문의 2장에서는 WAVE 관련 연구를 소개하고, 3장에서는 WAVE 프로토콜의 구현 범위 및 개발 환경을 소개한다. 4장에서는 WAVE 네트워킹 서비스 프로토콜 개발 내용을 서술하고, 5장에서는 Traffic Management System 개발 내용에 관해서 서술한다. 6장에서는 프로토콜 개발 결과를 보이고 개발 결과를 검증한다.

II. 관련 연구

Morgan은 DSRC와 WAVE를 소개하고, WAVE 표준을 구현하기 위한 지침을 제시하고 있다^[1]. WAVE와 관련된 용어를 정립하고, RSU와 OBU 등의 구성요소에 대한 설명과 WAVE의 기본 구조, WAVE 통신 스택에서의 MAC 계층 서비스, WAVE 서비스 관리 등에 대한 내용을 서술하고 있다.

Clausen은 WAVE의 실질적인 데이터 전송을 위한 데이터 평면의 WSMP와 IPv6에 관해서 논의하고 있다^[2]. 하지만 이 논문에서 WSMP를 중심으로 분석하고 개발한 것에 비해 Clausen은 IPv6를 사용하여 WAVE 네트워킹을 구축하는 것에 대해서 여러 방향으로 접근하여 논의한다.

박현문은 IP 기반 교통 시스템에서 V2I 통신에 있어서 데이터 신뢰성을 위해 TCP, UDP가 아닌 DCCP의 적용을 제시하고 있다^[7]. 본 논문에서 WAVE의 WSMP를 개발한 것과 달리 IP 기반에서의 차량 통신을 다루고 몇 가지 문제점을 보완하기 위해 DCCP를 제안했다.

조국래는 차량 간의 GPS 정보교환을 위해 지국을 이용해서 효율적인 인증 프로토콜을 제시하고 있다^[8]. 본 논문에서는 보안과 관련된 부분은 구현하지 않고 있는 것과 달리 이 논문은 통신 방법은 다르지만 정보들을 보내거나 받을 때의 암호화를 통해 보안을 피하고 있다.

Shrestha는 VANET(Vehicular Ad hoc Network)에서의 V2I 통신에서 멀티홉 통신에 대해 언급하고 있다^[9]. V2I 통신 시에 직접적으로 RSU와 OBU가 통신하지 않고 중간에 다른 차들을 중간 노드로 이용해서 멀티홉 통신을 할 때의 라우팅에 대해 설명하고 있다.

Davron은 VANET에서의 V2V 통신을 할 때, 차량들이 브로드캐스팅 할 때 발생할 수 있는 Broadcast Storm 현상을 보완하기 위한 방법을 제시하고 있다^[10].

장성식은 VANET에서 이동 차량을 고려한 메시지 전송 방법을 제시하고 있다^[11]. 이와 비슷하게 김현숙은 VANET에서 긴급 메시지 전달을 위한 중계노드 선정 기법을 제시하고 있다^[12, 15]. 이 두 논문에서는 V2V 통신에서 메시지를 어떻게 효율적으로 전달할 것인가에 대한 방법을 제시한다.

이상우는 차량통신(VANET)에 대한 특징을 설명하고 WAVE 기술에 대한 소개를 하고 있다^[13]. 그 중 WAVE에 대해서는 구조, 주파수 대역, WAVE MAC에 대한 설명을 하고 있다. 본 논문에서는 WAVE 네트워킹 서비스 프로토콜의 개발에 대한 내용이 주를 이루지만, 이 논문에서는 전반적인 WAVE에 대한 설명을 볼 수 있다.

김정훈은 VANET의 V2I 통신에서 효율적인 라우팅 알고리즘에 대한 설명을 하고 있다^[14]. RSU와 차량 사이에 전송 경로 상의 홉 수를 줄이고 전송 경로의 환경을 통해 End-to-End 지연시간을 예측하여 가장 짧은 경로를 선택하는 알고리즘을 설명하고 있다.

변태영은 Mobile Access Point를 이용하여 차량 간 통신 시스템의 프로토타입을 보여주고 있다^[16]. 여기에서는 차량간 통신에 모바일 액세스 포인트가 이용가능한지 알아보기 위해 실제 도로에서 간단한 V2I2V 통신 시스템을 만들어 테스트하고 결과를 제시하고 있다. 또 다른 논문에서는 어디서나 무선 인터넷 접근이 가능한 무선 액세스 포인트의 프로토타입을 보여주고 있으며 실제 테스트에서의 성능을 측정하고 있다^[17]. 그 다음 논문에서는 무선 WAN 기반의 차량 통신 시스템을 설계하고 구현하

고 있으며 실제 도로에서 테스트를 수행하여 성능 평가를 실시하고 있다¹⁸⁾. 이 논문들¹⁶⁻¹⁸⁾은 테스트를 실제 도로 상에서 실시하여 그에 따른 성능 평가를 하고 있는데, 프로토콜을 구현하여 간단한 가상의 상황에서 테스트를 하는 본 논문과는 차이가 있다.

위와 같이 기존의 몇몇 논문들을 살펴보면 대부분 WAVE 네트워킹 서비스 프로토콜을 개발한 본 논문과는 다르게 VANET에서 V2I나 V2V 통신을 할 때 메시지를 보내는 방법이나 기존의 방식을 개선하거나 보안과 관련된 내용을 제시하고 있다^{7-12,14)}. 그 외에는 WAVE 자체의 구현이 아닌 WAVE에 대한 소개나 설명을 하고 있다^{1,13)}. 본 논문처럼 WAVE 구현, 특히 WSMP의 구현을 다루는 논문은 거의 찾아볼 수 없었다. 그렇기 때문에 본 논문에서 WSMP, WME를 구현하는 것이 WAVE 네트워킹 서비스 프로토콜을 구현에 대한 정리의 의미를 가질 수 있다.

III. 프로토콜 구현범위 및 개발환경

3.1. 프로토콜 구현범위

WAVE 표준 프로토콜 스택은 그림 1과 같이 IEEE 1609.2 WAVE 보안 서비스, IEEE 1609.3 WAVE 네트워킹 서비스, IEEE 1609.4 WAVE 멀티채널 오퍼레이션으로 구성된다⁵⁾. WAVE 보안 서비스는 전송하는 통신 메시지의 보안을 위한 서비스이고, WAVE 네트워킹 서비스는 네트워크 계층과 전송 계층에서 WAVE 시스템의 애플리케이션에 계 데이터 전송 서비스를 제공한다. WAVE 멀티채널 오퍼레이션은 다중 채널을 사용하여 네트워크

계층의 데이터 전송을 효율적으로 제어한다.

이 논문에서는 RSU를 위해 1609.3 WAVE 네트워킹 서비스 부분을 구현하였다. WAVE 네트워킹 서비스는 관리 평면(Management plane)의 WAVE Management Entity(WME)와 데이터 평면(Data plane)의 WAVE Short Message Protocol(WSMP), IPv6, UDP/TCP, LLC 부분으로 구성되어 있다. 이 중에서 이미 구현되어 널리 사용되고 있는 IPv6, UDP/TCP 부분을 제외한 WME와 WSMP를 개발하고 WSMP를 지원하기 위해서 기존의 LLC를 확장하였다. 그림 1에서 음영 처리된 부분이 이 논문에서 구현하는 부분이다.

3.2. 개발환경

WAVE 프로토콜은 Xen 서버를 이용하여 각각 노면기지국 장치(Road-Side Unit, RSU)와 차량 단말기(On-Board Unit, OBU) 역할을 하는 두 대의 가상 머신을 사용하여 개발하였다. 이와 달리 검증환경은 개발환경과 다소 차이가 있는데, RSU는 두 개의 테스트 보드에 각각 설치하고 두 대의 노트북을 이용하여 여러 개의 OBU의 역할을 하도록 사용한다. RSU와 OBU에서 사용되는 다양한 관리 정보(Management Information Base, MIB)는 SNMP 에이전트에 WME MIB를 추가하여 SNMP를 통해 관리할 수 있도록 하였다. RSU와 OBU 간에는 802.11p를 에뮬레이션 한 Pseudo WAVE를 사용하여 통신이 이루어진다. 표 1은 RSU와 OBU 가상 머신의 세부적인 개발 환경을 나타내고 있다.

표 1. RSU 및 OBU 개발환경
Table 1. Development Environment of VM

| | |
|-----------|--------------|
| 운영체제 | CentOS 5.6 |
| 개발언어 | C |
| 컴파일러 | GCC 4.1.2 |
| SNMP 에이전트 | Net-SNMP 5.7 |

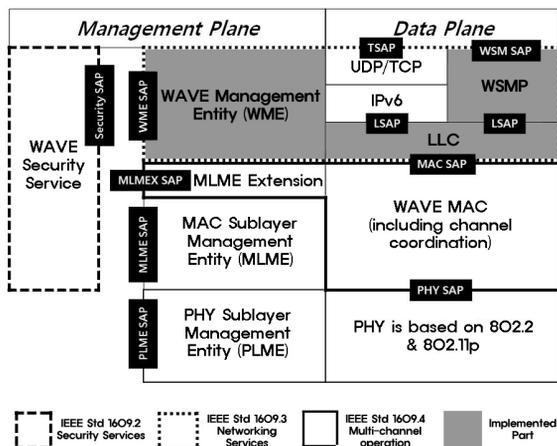


그림 1. WAVE 표준 프로토콜 스택
Fig. 1. Standard Protocol Stack for WAVE

IV. WAVE Networking Service 개발내용

4.1. WAVE Management Entity 구현

WAVE Management Entity(WME)는 RSU 또는 OBU에서 서비스를 광고하거나 요청하고, WME MIB를 통해서 관리 정보를 관리한다. RSU는 자신이 제공하고자 하는 서비스를 WME MIB의 Provider Service Table에 등록하고 WAVE Service Advertisement(WSA) 메시지를 통해서 OBU에게 자신의 서비스 유형을 알리게 된다.

RSU가 보내는 WSA 메시지는 크게 네 가지 부분으로 나눌 수 있다. 헤더 부분과 Service Info, Channel Info, WAVE Routing Advertisement(WRA)로 나눌 수 있다. 헤더에는 WAVE 버전과 Change Count 정보와 함께 확장 필드들이 있다. 확장 필드에는 광고(Advertisement) 반복률, Transmit Power Used, 위치를 의미하는 2DLocation, 3DLocationAndConfidence 등이 들어갈 수 있다. 헤더를 제외한 부분들은 공통적으로 각각과 관련된 WSA WAVE element ID가 들어가며, Service Info에는 광고하려는 서비스와 관련된 정보들이 들어가고 Channel Info에는 Service Info와 연관된 채널의 정보들이 들어가며 WRA에는 라우팅에 대한 정보들이 들어가게 된다. 그림 2는 WSA 메시지의 포맷을 간략하게 나타내고 있다.

RSU가 서비스를 요청하고 광고하기 위해서는 먼저 RSU의 WAVE 어플리케이션이 WME-ProviderService.request 프리미티브를 통해 서비스를 요청한다. 그 다음에 WaveSecurityServices-SignedWsa.request와 MLMEX-VSA.request

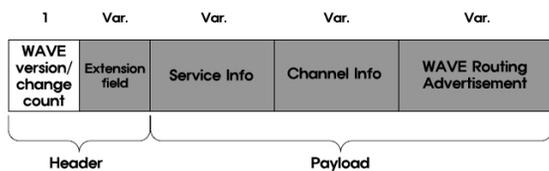


그림 2. WSA 포맷
Fig. 2. WAVE Service Advertisement(WSA) Format

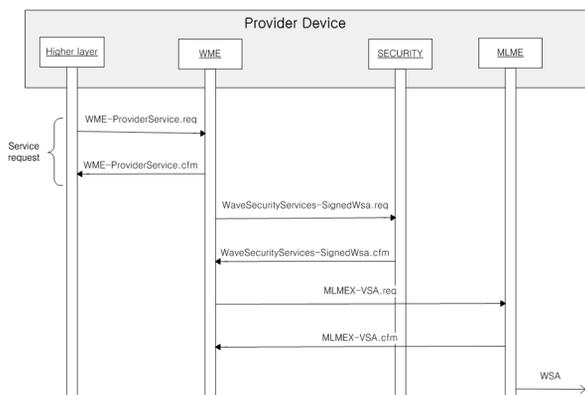


그림 3. Provider Service Request 시퀀스 다이어그램
Fig. 3. The Sequence Diagram for Provider Service Request

를 통해 각각 IEEE 1609.2의 메시지 보안 서비스와 IEEE 1609.4의 멀티채널 오퍼레이션을 수행하고 WSA 메시지를 전송한다. 그림 3은 RSU가 서비스

를 요청하고 광고하는 시퀀스 다이어그램을 나타낸다.

OBU는 RSU로부터 수신한 WSA 메시지를 WME MIB의 User Service Table과 비교하여 자신이 지원하는 서비스인지 확인한 후 WAVE 애플리케이션에 이를 알리고, 서비스를 시작한다^[5].

4.2. WAVE Short Message Protocol 구현

WAVE Short Message Protocol(WSMP)은 간단한 메시지 전송 프로토콜로서, RSU와 OBU 사이 또는 OBU들 사이에 WSM 전송을 요청하고 수신하는 프로토콜이다. 최대 512 바이트의 사용자 데이터를 전송할 수 있다.

WAVE Short Message는 WSMP 헤더와 데이터 부분으로 나눌 수 있다. 헤더 부분에는 버전과 PSID, 확장 필드, WSMP WAVE element ID, Length의 정보가 들어간다. 버전은 WSM 프로토콜의 버전 정보를 담고 있고, PSID는 Provider Service의 ID로 해당하는 Provider의 서비스의 식별

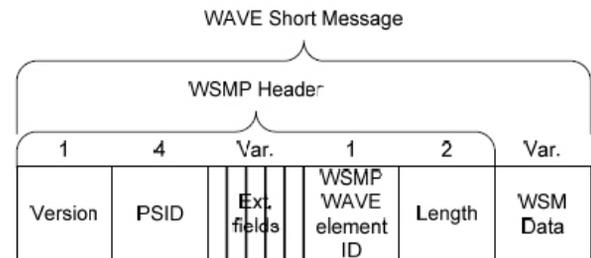


그림 4. WAVE Short Message 포맷
Fig. 4. WAVE Short Message(WSM) Format

자가 들어간다. 확장 필드에는 선택적으로 들어가는 요소들이 있는데 채널 번호, 전송률, Transmit Power Used가 들어갈 수 있다. WSMP WAVE element ID는 WSMP에 사용되는 WAVE 요소들의 식별자가 들어가고, Length에는 뒤에 붙는 WSM Data의 길이 정보를 담고 있다. 데이터 부분인 WSM Data는 상위 계층에서 사용되는 데이터를 담는다. 그림 4는 간략하게 WSM의 포맷을 보여주고 있다.

RSU에서 WSM을 요청하기 위해서는 먼저 WSM-WaveShortMessage.request 프리미티브를 통해 WSM 전송을 요청한다. 이 때, WSM 헤더의 최대 길이(WsmMaxLength)를 확인한 후 DL-UNITDATA.request 프리미티브를 통해 WSMP 헤더를 생성한다. 앞의 두 단계가 성공적으로 수행 되면 WSM-WaveShortMessage.confirm 프리미티브

를 통해서 WAVE 애플리케이션에게 WSM 요청이 성공했음을 알리고, 해당하는 주소의 RSU 또는 OBU에게 WSM 메시지를 전송한다. 그림 5는 RSU에서 WSM 전송을 요청하는 시퀀스 다이어그램을 나타낸다.

WSM 요청이 올 때, WSM 메시지를 생성하고 전송하기 위해서는 일단 WSMP 헤더의 길이를 WSM 헤더 최대 길이보다 작은지 비교한다. 이 때, 결과가 참으로 나타나면 WSMP 헤더를 생성하고 WSM 데이터를 추가 한다. 그 후 Transmit Power Level에 값이 있으면 TxPwr_Level로 변환한다. 모든 과정이 성공적으로 실행되면 DL-UNITDATA.request 프리미티브를 호출하여 LLC 계층으로 데이터를 전달하게 된다^[5]. 그림 6은 WSM 메시지를 생성하고 전송하는 WSM-WaveShortMessage.request 프리미티브의 순서도를 나타낸다.

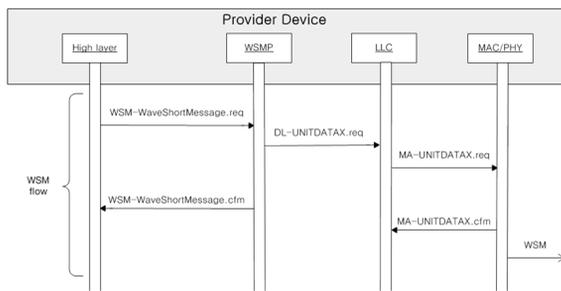


그림 5. WAVE Short Message Request 시퀀스 다이어그램
Fig. 5. The Sequence Diagram for WAVE Short Message Request

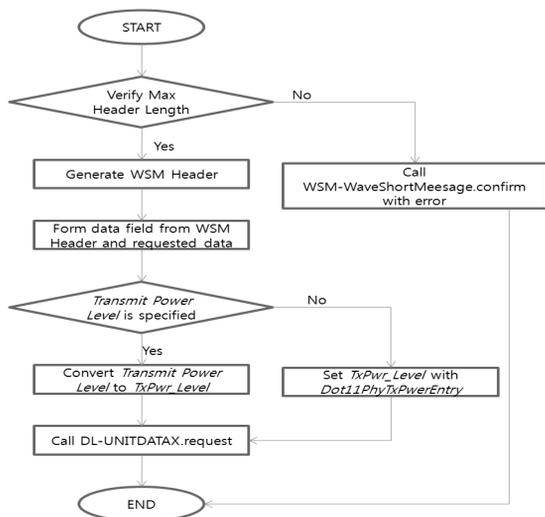


그림 6. WSM-WaveShortMessage.request 순서도
Fig. 6. The Flow Chart of WSM-WaveShortMessage.request

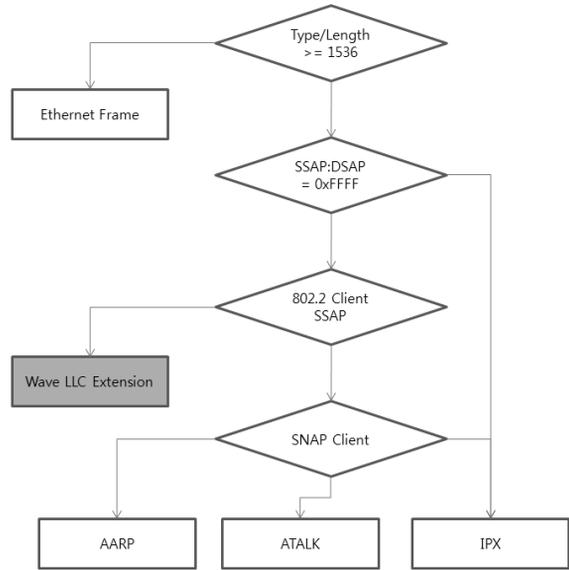


그림 7. LLC/SNAP 순서도
Fig. 7. The Flow Chart of LLC/SNAP

4.3. WAVE LLC Extension 구현

WSMP의 송/수신을 지원하기 위해서 LLC 타입1을 지원하는 기존의 LLC를 확장하여 구현하였다. DL-UNITDATA.request와 함께 확장 LLC 프리미티브인 DL-UNITDATA.request를 지원하도록 확장하였다.

이것은 수신 패킷의 SNAP 헤더의 이더넷 타입을 보고 판단하는데 그 값이 '0x88DC' 이면 WSMP로, '0x88DD' 이면 IPv6 패킷으로 처리한다^[5]. 그림 5의 시퀀스 다이어그램을 보면 WSM을 전송하기 위해 DL-UNITDATA.request를 사용하여 WSM 데이터를 전달하고 있다. 그림 7은 WAVE LLC 확장을 위한 LLC/SNAP의 순서도이다.

V. WAVE Networking Service 개발내용

5.1. Traffic Management System 개요

WAVE Networking Services 프로토콜의 개발 결과를 검증하기 위해서 구현된 프로토콜을 사용하여 도로의 상황을 시뮬레이션 하는 Traffic Management System(TMS)을 개발하였다. TMS는 도로의 교통 정보를 실시간으로 전송해주는 WAVE 애플리케이션으로, RSU가 다수의 OBU로부터 차량들의 속도 데이터를 수집하여 다른 구역의 RSU를 통해 OBU들에게 전달해주는 서비스를 제공한다. WSMP와 WME가 제대로 구현이 되었는지 검증하

기 위해 RSU와 OBU들 간에 WSM 전송이 잘 되는지. WME의 내용을 잘 읽어오는지 확인하여야 한다. TMS 서비스는 RSU와 OBU간의 WSM을 송수신하는 메시지를 볼 수 있고 RSU 관리자를 통해 각 RSU들의 WME MIB를 열람하여 볼 수 있기 때문에 WSMP와 WME가 제대로 구현이 되었는지 충분히 확인 할 수 있다. TMS 서비스 이용자는 차량의 정체, 사고와 같은 도로 상황 정보를 얻을 수 있다.

5.2. Traffic Management System 구성 및 동작

TMS는 TMS 서버와 RSU용 TMS, OBU용 TMS, 그리고 RSU 관리자로 구성된다. 그림 8은 전체적인 TMS의 동작을 설명하고 있다.

5.2.1. OBU 용 TMS

OBU용 TMS는 WME를 사용해서 RSU로부터 전송받은 WSA 메시지를 확인하여 RSU가 TMS 서비스를 제공하는지 확인한 후, 특정 시간 간격으로 현재 차량의 속도 데이터를 WSMP를 사용해서 RSU에게 송신한다. 그리고 RSU로부터 다른 구역에 있는 OBU들의 속도 평균이 담긴 WSM을 수신 받으면 화면에 출력한다. 이를 통해 RSU와 OBU간의 통신이 제대로 이루어진다는 것을 확인할 수 있다.

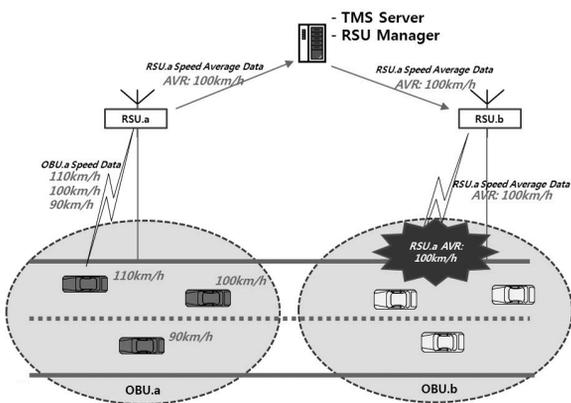


그림 8. TMS 동작 시나리오
Fig. 8. A Service Scenario of TMS

5.2.2. RSU 용 TMS

RSU용 TMS는 OBU들로부터 전송 받은 속도 데이터의 평균값을 계산하여 UDP/IP를 사용해서 TMS 서버에 송신한다. 또, TMS 서버로부터 다른 구역의 RSU와 연결된 OBU들의 속도 평균 데이터를 수신하면 자신에게 접속된 OBU들에게 브로드캐

스팅 하여 광고한다. 이를 통해 RSU 간의 통신이나 RSU와 OBU간의 통신이 제대로 이루어진다는 것을 볼 수 있다.

5.2.3. TMS 서버

TMS 서버는 여러 RSU들로부터 전송 받은 속도 평균 데이터들을 모든 RSU들에게 브로드캐스팅 하여 전달하는 역할을 수행한다. TMS 시스템 내에서 TMS 서버는 단순히 받은 WSM을 브로드캐스팅하는 기능만 존재한다.

5.2.4. RSU 관리자

RSU 관리자는 각 RSU들의 MIB를 열람하고 관리하는 SNMP 기반의 웹 인터페이스로써, Adventnet SNMP API를 사용하여 ASP.NET으로 구현되었다. RSU 관리자를 통해 각 RSU들의 WME MIB의 내용들을 확인하고, Provider Service를 등록함으로써, IEEE 1609.3의 WME가 구현이 잘 되었는지 확인 할 수 있다.

5.3. Traffic Management System 프로토콜 정의

TMS에서 사용되는 프로토콜은 RSU용 TMS와 OBU용 TMS에서 WSMP를 사용하여 TMS 서비스를 제공하기 위해 정의한 텍스트 기반의 간단한 프

표 2. TMS 프로토콜
Table 2. TMS Protocol

| 프로토콜 | 의미 |
|-------------|----------------------|
| OBUSPEED | OBU차량의 현재 속도 |
| RSUSPEEDAVG | RSU가 수집한 OBU들의 평균 속도 |
| OBUEXIT | OBU가 RSU와의 연결을 종료 |
| UNDEFINED | 정의되지 않은 TMS 프로토콜 |

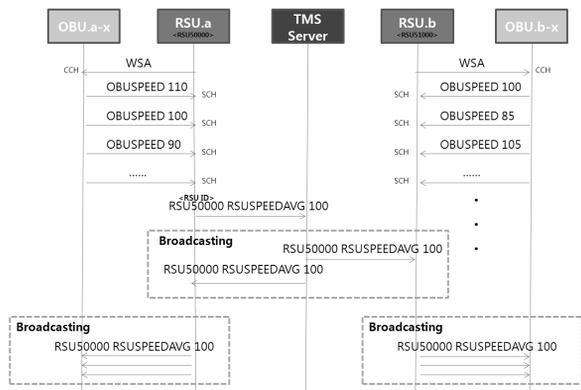


그림 9. TMS 프로토콜 사용 예
Fig. 9. A Usage Example of TMS Protocol

로토콜이다. 이 프로토콜은 단순히 차량의 속도만을 사용하는 RSU의 'Traffic Notification' 서비스를 가정하여 만든 프로토콜이다. 이 서비스에서는 차량의 속도만을 데이터로 사용하기 때문에, 각 OBU 차량의 속도와 해당 RSU 영역 내의 OBU 차량의 평균 속도만 있어도 TMS 내에서 RSU와 OBU의 동작을 볼 수 있다. 표 2는 TMS에서 사용되는 프로토콜과 그 의미를 나타내고 있으며, 그림 9는 그림 8의 시나리오를 바탕으로 TMS 프로토콜의 사용 예를 나타내고 있다.

VI. 프로토콜 개발 결과 및 검증

6.1. 시험 시스템 구성

구현된 WSMP 프로토콜을 검증하기 위해 시험 시스템을 구성했다. 이 시험 시스템은 구현한 프로토콜이 정상적으로 동작하는 지 알아보기 위한 시스템이므로 테스트 보드 상에서 실시하며, 실제 운행 시에 일어날 수 있는 변수나 돌발 상황은 고려하지 않는다.

시험 시스템은 그림 10과 같이 노트북 2대와 Tricomtek MW2443-Control 보드(ARM) 2대, 인터넷 공유기 1대로 구성된다.

노트북A는 OBU.a 그룹의 OBU를 시뮬레이션하고, TMS 서버를 구동한다. 노트북B는 OBU.b 그룹의 OBU를 시뮬레이션하고, RSU 관리자를 실행한다. MW2443-Control 보드 2대는 각각 RSU.a와 RSU.b의 역할을 한다. 그림 11은 시험 시스템의 환경을 간단히 나타낸다.

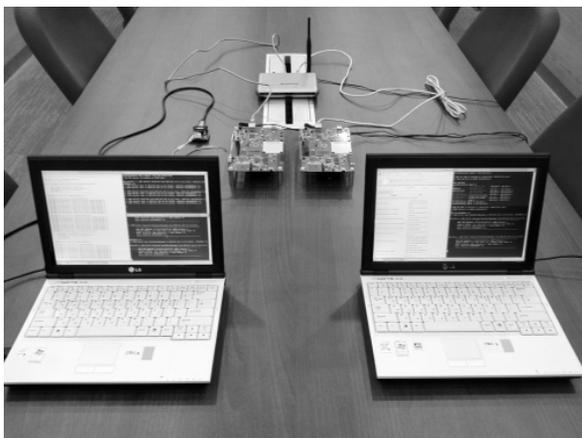


그림 10. TMS 시험 시스템
Fig 10.TMS for Experiment

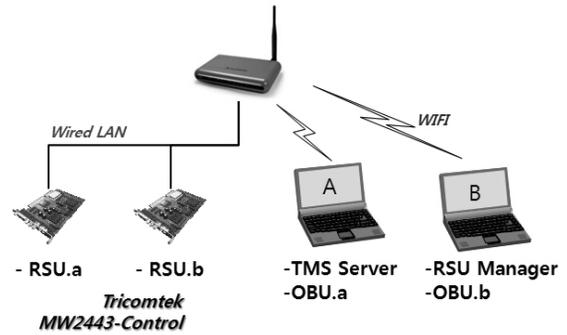


그림 11. 시험 시스템 환경
Fig 11.Environment of Test

6.2. 개발 결과 및 검증

6.2.1. 차량 사고 상황 시뮬레이션 실험

RSU.a와 RSU.b에 속한 OBU를 각각 10대로 설정하고, RSU.a에 속한 OBU.a 그룹 차량들에게 사고가 발생하여 도로가 완전히 정체되는 상황을 시뮬레이션하여 시험했다. 차량들의 속도 및 평균 속도를 이용한 'Traffic Notification' 서비스가 조금 더 명확하게 보이게 하기 위해 사고 상황을 가정하고 여러 대의 OBU의 동작을 보여준다. 하지만 OBU가 너무 많다면 WSM 메시지를 일일이 확인하기가 어려워지므로 OBU의 수를 10대로 설정했다. 그림 12는 사고 상황 시나리오를 나타낸다.

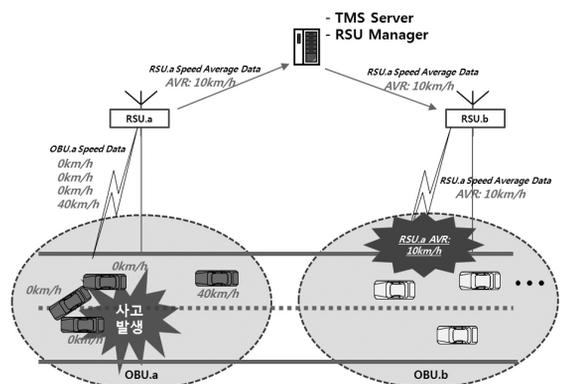


그림 12. 사고 상황 시나리오
Fig. 12.The Accident Scenario

그림 13은 차량 사고 발생 시뮬레이션 시의 RSU.a 통신 화면이다. OBU.a 그룹으로부터 전송받은 WSMP 메시지의 속도 데이터의 평균값(0km/h)을 계산하여 TMS 서버로 전달하고, TMS 서버로부터 전송받은 평균 속도 데이터를 WSMP를 사용하여 OBU.a 그룹에게 전달하고 있다. 수신한 WSMP 패킷을 분석하여 화면에 출력한 결과, 표준 패킷 포

맷에 맞게 정확히 송/수신 되고 있음이 확인된다. 현재 TMS 서비스에서는 WSMP 패킷을 분석하는 것 외에는 WSMP가 잘 구현되었는지 확인할 방법이 없다. 또 모든 RSU와 OBU간에 송수신되는 패킷을 분석하는 것은 어렵기 때문에 RSU와 서버 사이에 송수신되는 패킷을 분석하는 것이 더 효율적이다.

6.2.2. RSU 관리자 웹 인터페이스

RSU 관리자 웹 인터페이스는 크게 세 개의 화면으로 나누어진다. 가장 처음 화면은 RSU의 목록을 보여주는 화면이다. 이 화면에서는 각 RSU가 동작하고 있는지를 On/Off 열을 통해 볼 수 있으며, Detail 링크를 통해 해당 RSU의 WME MIB의 테이블 리스트 화면으로 갈 수 있다. 또, Reg 링크를 통해 해당 RSU의 Provider Service Request 테이블에 서비스를 등록할 수 있다. 그림 14는 첫 화면으로 두 대의 RSU(RSU.a, RSU.b)가 등록되어 관리

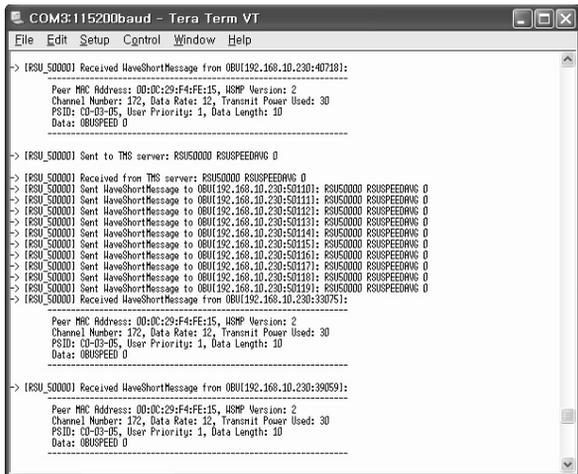


그림 13. RSU 통신 화면
Fig. 13. Communication of RSU on Board

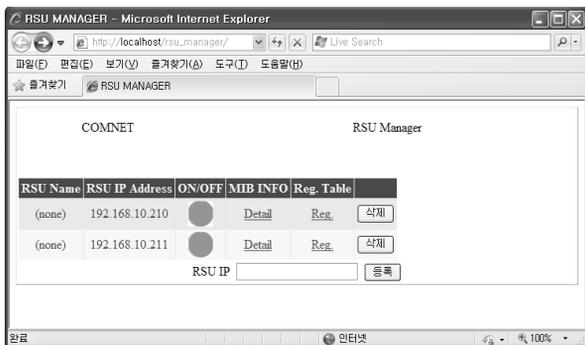


그림 14. RSU 목록
Fig. 14. RSU List

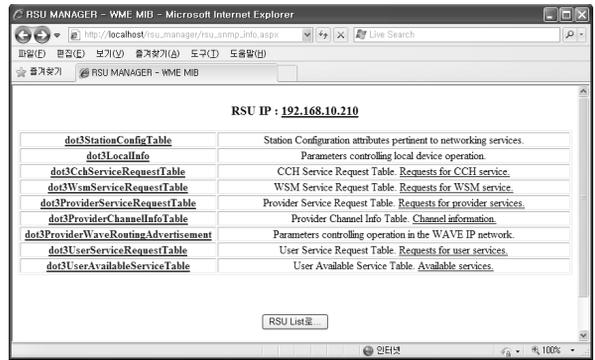


그림 15. RSU의 WME MIB 테이블 목록
Fig. 15. WME MIB Table List of RSU

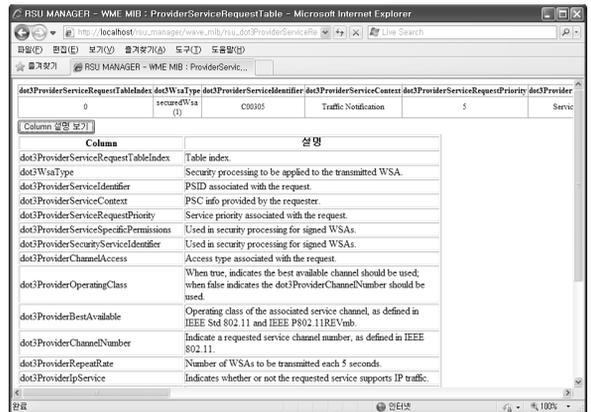


그림 16. Provider Service Request 테이블의 MIB 데이터
Fig. 16. MIB Data of Provider Service Request Table

되고 있으며 모두 동작하고 있는 화면을 보여준다.

두 번째 화면은 첫 화면에서 Detail 링크를 누르면 나타나는 화면으로 해당 RSU의 WME MIB 테이블 목록을 보여준다. 이 화면에서는 Provider Service Request, User Service Request 테이블 등의 MIB 데이터를 열람할 수 있다. 그림 15는 RSU.a의 WME MIB 테이블 목록 화면이다.

세 번째 화면은 각 WME MIB 테이블의 내용을 보여주는 화면이다. 그림 16은 RSU.a의 Provider Service Request 테이블의 MIB 데이터를 보여주고 있다. 현재 TMS에서 제공하는 'Traffic Notification' 서비스가 추가되어 있다.

Ⅶ. 결 론

이 논문에서는 IEEE 1609.3에 기술되어 있는 WAVE Networking Services 표준 프로토콜을 분석하고 구현하였다. 구현된 프로토콜을 검증하기 위해 WAVE 애플리케이션인 TMS와 RSU 관리자를 개발하였고, 특정 시나리오(사고 상황)를 만들어서

TMS를 이용하여 프로토콜이 표준에 따라 올바르게 구현되었음을 검증하였다.

차후 연구 과제로는 IEEE 1609.2에 기술되어 있는 메시지 보안 서비스와 IEEE 1609.4의 MAC 계층의 다중 채널 서비스를 분석하고 구현하는 것이다. 그리고 TMS를 프로토콜 검증의 목적이 아닌 실제 지능형 교통 시스템에 적용하여 사용할 수 있도록 요구사항을 파악하고, 이 요구사항을 만족시킬 수 있는 실용적인 시스템으로 향상시키는 것이다.

참 고 문 헌

- [1] Yasser L. Morgan, "Managing DSRC and WAVE Standards Operations in a V2V Scenario", *International Journal of Vehicular Technology*, vol.2010
- [2] Thomas Clausen, Emmanuel Baccelli, Ryuji Wakikawa, "IPv6 Operation for WAVE - Wireless Access in Vehicular Environments", *INRIA*, September, 2010.
- [3] IEEE Standards Association, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager", *IEEE Std 1609.1-2006*, October, 2006.
- [4] IEEE Standards Association, "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages", *IEEE Std 1609.2-2006*, July, 2006.
- [5] IEEE Standards Association, "IEEE Standard for Wireless Access in Vehicular Environment (WAVE) - Networking Services", *IEEE Std 1609.3-2010*, December, 2010.
- [6] IEEE Standards Association, "Draft Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-channel Operation", *IEEE Std 1609.4-2010*, June, 2010.
- [7] Hyun-Moon Park, Hae-Moon Seo, Gil-Yong Lee, Soo-Hyun Park, Sung Dong Kim, "Development of The DCCP for Data Reliability in IP Traffic System", *Journal of IEMEK*, 5(1), pp. 7-17, March, 2010 (Written in Korean).
- [8] Kookrae Cho, Jong-wuk Son, HuiSup Cho, "An Efficient Authentication Protocol for GPS Information Exchange between Cars Using the Base Station", *Journal of IEMEK*, 5(3), pp. 118-127, September, 2010 (Written in Korean).
- [9] Raj K. Shrestha, Sangman Moh, Ilyong Chung and Heewook Shin, "Multihop Vehicle-to-Infrastructure Routing Based on the Prediction of Valid Vertices for Vehicular Ad Hoc Networks", *Journal of IEMEK*, 5(4), pp. 243-253, December, 2010.
- [10] Davron Makhmadiyarov, Soojung Hur, Yongwan Park, "Broadcasting Message Reduction Methods in VANET", *Journal of IEMEK*, 4(1), pp. 42-49, March, 2009.
- [11] Seong-Sik Jang, Hyun-Sook Kim, Hyun-Cheol Cha, Tae-Young Byun, "A Message Forwarding Scheme Considering Moving Vehicle's Trajectory over VANET", *2010대한임베디드공학회 추계학술대회 논문집*, 1(1), pp. 432-435, November, 2010 (Written in Korean).
- [12] Hyun-Sook Kim, Seok-Yeol Heo, Won-Ryul Lee, Wan-Jik Lee, Tae-Young Byun, "A Relay Node Selection Scheme for Delivery of Urgent Message Considering Density of Moving Vehicles over VANET", *2010대한임베디드공학회 추계학술대회 논문집*, 1(1), pp. 438-441, November, 2010 (Written in Korean).
- [13] 이상우, 오현서, 광동용, 박중현, "차량통신 네트워킹을 위한 MAC 기술", *The Journal of The Korean Institute of Communication Sciences(Information & Communications Magazine)*, 26(4), pp. 55-60, March, 2009 (Written in Korean).
- [14] JungHun Kim, SuKyoung Lee, "Reliable Routing Protocol for Vehicle to Infrastructure Communications in VANET", *The Journal of The Korean Institute of Communication Sciences*, 34(8), pp. 751-845, August, 2009 (Written in Korean).
- [15] Hyun-Sook Kim, Seong-Sik Jang, Hyun-Cheol Cha, Tae-Young Byun, "A Relay Vehicle

Selection Scheme for Delivery of Emergency Message Considering Density and Trajectory of Moving Vehicles for VANET”, *Communications in Computer and Information Science*, 195, pp. 257-266, 2011.

- [16] Tae-Young Byun, “ICSW2AN : An Inter-vehicle Communication System using Mobile Access Point over Wireless WAN”, *Communications in Computer and Information Science*, 78, pp. 355-366, 2010.
- [17] Tae-Young Byun, “An Experimental Performance Measurements of Implemented Wireless Access Point for Interworking Wi-Fi and HSDPA Networks”, *Communications in Computer and Information Science*, 121, pp. 145-154, 2010.
- [18] Tae-Young Byun, “An Performance Evaluation of Intervehicle Communication System over Wireless WAN”, *International Journal of Control and Automation*, 3(3), September, 2010.
- [19] Net-SNMP 5.7, *Net-SNMP* <http://www.net-snmp.org>, March, 2012.
- [20] Adventnet WebNMS SNMP API .NET EDITION 4, *WebNMS SNMP .NET Edition - Home* <http://www.webnms.com/net-snmp/index.html>, March, 2012.

고 강 민 (Kangmin Ko)

준회원



2011년 2월 계명대학교 컴퓨터공학과 졸업
2011년 3월~현재 계명대학교 컴퓨터공학과 석사과정
<관심분야> WAVE

권 동 우 (Dongwoo Kwon)

준회원



2010년 계명대학교 컴퓨터공학과 학사
2010~2012년 계명대학교 컴퓨터공학과 석사
2012년 3월~현재 포항공과대학교 정보전자융합공학부 박사과정

<관심분야> 인터넷 침입 예측, WAVE, Honeynet, 네트워크 관리

주 흥 택 (Hongtaek Ju)

종신회원



1989년 한국과학기술원 전자계산학과 학사
1989~1991년 포항공과대학교 컴퓨터공학과 석사
1991~1997년 대우통신 종합연구소 선임연구원
1997~2002년 항공과대학교

컴퓨터공학과 박사

2007년~2009년 대구경북과학기술원 겸임연구원

2002년~현재 계명대학교 컴퓨터공학과 부교수

<관심분야> 네트워크 및 시스템 관리, Embedded 웹 서버를 이용한 네트워크 요소관리, SyncML, 인터넷 침입 예측, 네트워크 보안