

# Internet Protocol Term Project

*SNMPV2 client implementation as an Android application*



2018년 1학기

2013210006

컴퓨터학과

장 용 수

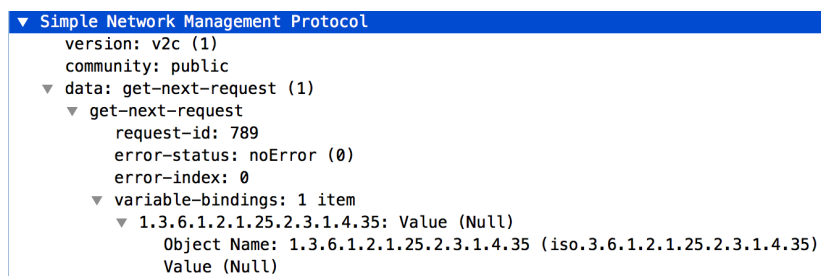
# 인터넷 프로토콜 리포트

2018년도 1학기 인터넷 프로토콜 과제로 SNMP(simple network management protocol)의 Client 역할을 안드로이드 앱으로 구현해 보았다. SNMP는 IP 네트워크 상의 장치로부터 정보를 수집 및 관리하고, 정보를 수정하여 동작을 변경하는 데에 사용하는 프로토콜이다. 우리는 고려대학교 Widen 랩실에 있는 IP 카메라의 정보를 얻고, 수정하였다. SNMP는 UDP에 패킷을 실어서 통신한다. 자바의 `java.net` 라이브러리를 이용하여 전송 및 수신을 하였다. BER 인코딩 및 디코딩은 과제에 주어진 `asn1_java` 라이브러리를 이용하였다.

## SNMPv2 패킷 포맷 설명

SNMP 패킷은 BER 규칙으로 인코딩 된다. BER은 Tag, Length, Value 구조이다. Tag는 데이터 타입, Length는 Value의 길이, Value는 보내고자 하는 내용이 된다. 전체적으로 보면 마치 양과 껍질이 계속 덮여 있듯이, 끝에는 타입과 길이가 정보가 있고 그 속은 내용인데, 그 내용도 또 타입과 길이가 정보 내용으로 구성되어 있다. 인코딩을 하려면 안에서부터 내용을 점점 감싸서 전체를 만들어야 하고, 디코딩을 하려면 겉을 보고 안의 내용을 분석해 나가면 된다.

SNMP패킷은 크게 버전, 커뮤니티, 데이터로 이루어져있다. 버전은 SNMP 버전을 나타낸다. 버전 2는 값 0x01이다. 커뮤니티는 마치 사용자 아이디와 패스워드를 나타내는 것인데 우리는 get일 때는 “public”, set일 때는 “write” String을 이용한다. 데이터는 PDU타입(‘GetRequest’, ‘SetRequest’, ‘GetNextRequest’ 등)과 error-status, error-index, variable-bindings로 구성된다. variable-bindings에는 실제로 우리가 요청하는 oid와 값의 쌍으로 들어있다. Get을 할때는 요청하는 oid와 값 null을 쌍으로 넣어주면 되고, Set을 할때는 변경하려는 oid와 해당 값을 넣어주면 된다. 아래 스크린샷은 WireShark로 SNMP 패킷을 잡았을 때 분석된 모습이다.



## SNMP Client 제작기

SNMP 패킷을 생성하고, UDP에 실어 전송하는 역할은 각각 `SNMPManager`, `UDPManager` 객체가 담당한다. `SNMPManager`에서는 사용자가 요청한 명령에 맞게 패킷을 만들고 반환한다. `MainActivity`에 반환된 전송할 SNMP 패킷은 `UDPManager`의 `sendAndReceiveMessage()` 함수 인자로 넘겨져 UDP에 실려져 목적지까지 간다. IP 카메라는 그에 상응하는 메시지를 전달해

주는데 처음에 보냈던 포트번호로 오기 때문에, 보냈던 소켓을 그대로 사용해 메시지를 받는다. 메시지를 받으면 메시지와 크기정보를 MainActivity에 반환하고, MainActivity에서는 다시 SNMPManager의 parseReceiveMessage() 로 메시지를 전달하여 내용을 확인한다.

보다 자세한 함수 콜 요청은 다음과 같다.

SNMP 패킷 이 만들어 지기 까지.

- 1) MainActivity 에서 snmpGET(), snmpSET(), snmpWALK() 이 호출되면 SNMPManager 의 buildRequestPacket() 를 호출한다. 여기서 인자는 OID, 요청 종류(get, set, walk), 값의 타입, 실제 값 4가지 이다. 반환된 SNMP Packet을 인자로 UDPManager의 sendAndReceiveMessage() 함수가 호출된다.
- 2) SNMPManager의 buildRequestPacket()에서는 요청 종류에 맞게 패킷을 만든다. 그 과정에서 requestData를 만들기 위해 get\_RequestBERDATA() 함수가 호출된다. BER로 인코딩 된 Version, Community 데이터와 반환된 requestData 를 가지고 최종 SNMP 패킷을 만든다.
- 3) get\_RequestBERDATA()에서는 request id, error status, error index 각각의 BER 인코딩된 데이터를 생성한다. 그리고 타입과 값으로 바인딩 된 Request Item 을 얻기 위해 get\_RequestItemBER() 함수를 호출한다. 위의 데이터들과 반환된 Request Item 을 엮어 requestData를 생성하고 반환한다.
- 4) get\_RequestItemBER() 함수에서는 요청 종류를 파악하고, set일 경우 각 값의 타입에 맞게 실제 값을 인코딩하여 OID-값 페어를 만든다. get일 경우 값은 NULL 이다. 만들어진 아이템 페어를 반환한다.

UDP로 전송되고 메시지를 받기까지.

- 1) MainActivity의 snmpGET(), snmpSET(), snmpWALK()에서는 SNMP Packet을 인자로 UDPManager의 sendAndReceiverMessage()함수를 호출한다.
- 2) sendAndReceiverMessage함수는 내부적으로 스레드를 만들어 UDP로 전송할 준비를 한다.
- 3) 생성자에서 정의했던 목적지 주소와, 포트번호, 인자로 받은 메시지를 이용하여 DatagramPacket을 만들고, DatagramSocket 을 통해 send 한다.
- 4) IP 카메라에서는 전송 했던 포트번호를 목적지 포트번호로 이용하기 때문에 만들었던 DatagramSocket 을 그대로 사용하고 receive를 한다. 메시지를 받으면 Handler를 통해 MainActivity로 넘긴다.

수신된 SNMP 패킷을 분석해서 결과를 보여주기 까지.

- 1) MainActivity에서 Handler를 통해 받은 수신된 SNMP 패킷을 SNMPManager의 parseReceiveMessage 함수를 호출하여 사용자에게 보여줄 결과값을 얻어온다.
- 2) SNMPManager의 parseReceiveMessage 함수에서는 메시지를 디코드 한다. variable-binding 아이템을 디코드 하면서 OID값과 값의 타입, 실제 값을 얻는다.
- 3) OID값과 값의 타입, 실제 값을 모두 사용자에게 보여줄 String 문자열로 만들고, 반환한다.

- 4) MainActivity에서는 String으로 된 결과값들을 사용자 화면에 출력한다. 요청 종류가 SNMP Walk일 경우 결과값으로 받은 OID를 가지고 다시 snmpWALK 함수를 호출하여 반복적으로 값을 받아오게끔 한다.

제작 과정 중 문제점.

SNMP Walk를 호출하면 값을 끝까지 받아오지 못하는 문제점이 있었다. 전송한 메시지에 대해 응답을 못받아 receive에서 stuck되는 현상이 발생하였다. 응답을 받지 못하면 다음 요청할 OID값을 얻어오지 못해 더 진행이 안되었다. 정말 문제점은 멈추는 지점이 실행할때 마다 다른데 이것은 SNMP Packet의 문제점이 아닌 것이다. 전송한 패킷이 중간에 유실 되었을 가능성이 있고, 응답 패킷이 유실가능성이 있다. 이를 Socket의 타이머를 설정하여 해결하였다. 메시지를 보낸 후 2초이상 응답이 없으면 다시 재전송 하는 것이다. Socket의 타이머를 설정하는 방법은 <https://stackoverflow.com/questions/10055913/set-timeout-for-socket-receive> 에서 얻을 수 있었다.

## SNMP Client 동작 이미지 캡처

### 1. GET

- OID 1.3.6.1.2.1.1.1.0 에 해당하는 Value 를 얻어옴.

The screenshot shows the SNMP Client application interface. At the top, there's a status bar with 'KT' and various icons. Below it, a blue header bar contains the text 'SNMP'. The main form has two sections. The first section has 'IP Address' set to 'kuwiden.iptime.org' and 'PORT' set to '11161', with a 'UDP SET' button below. The second section has 'OID' set to '1.3.6.1.2.1.1.1.0' and a dropdown menu for 'value' set to 'Integer'. Below this are 'GET' and 'SET' buttons. The 'GET' button is highlighted. Below the buttons, the output shows '[OID] 1.3.6.1.2.1.1.1.0' and '[VALUE] Hânwhã WiseNet IP Camera (OCTET STRING)'. At the bottom, there is a 'WALK' button.

## 2. SET

- OID 1.3.6.1.6.3.16.1.5.2.1.3.5.95.97.108.108.95.1.1 의 String을 “Hello world” 에서 “Good Morning!”으로 바꾼 결과

The screenshot shows the SNMP application interface. At the top, there's a status bar with 'KT' and various icons. Below it, a blue header bar contains the text 'SNMP'. The main area has two input fields: 'IP Address' with the value 'kuwiden.iptime.org' and 'PORT' with the value '11161'. Below these is a grey button labeled 'UDP SET'. A horizontal line separates this from the next section. In the next section, the 'OID' field contains '1.3.6.1.6.3.16.1.5.2.1.3.5.95.97.108.108.95.1.1'. Below the OID is a 'value' field with a dropdown menu set to 'String' and a text input field containing 'Hello world'. Below the input field are two grey buttons: 'GET' and 'SET'. At the bottom of this section, the text '[OID] 1.3.6.1.6.3.16.1.5.2.1.3.5.95.97.108.108.95.1.1' and '[VALUE] Hello world (OCTET STRING)' is displayed. A final horizontal line is followed by a large grey button labeled 'WALK'.

This screenshot shows the same SNMP application interface as the previous one, but with the 'value' field now containing 'Good Morning!'. The 'SET' button has been pressed, and the status at the bottom now shows '[VALUE] Good Morning! (OCTET STRING)'. All other elements, including the IP Address, PORT, and OID, remain the same.

### 3. WALK

[illegible]

## <안드로이드 로그 파일>

- 시작부분

```

06-21 17:19:45.228 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.1.0 OBJECT STRING Hanwha WiseNet IP Camera
06-21 17:19:45.272 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.2.0 OBJECT IDENTIFIER 1.3.6.1.4.1.36849.1.2.377
06-21 17:19:45.311 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.3.0 TimeTicks 292383018
06-21 17:19:45.357 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.4.0 OBJECT STRING Me<=somewhere.oryg=
06-21 17:19:45.403 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.5.0 OBJECT STRING XNP-6040H
06-21 17:19:45.447 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.6.0 OBJECT STRING Right here, right now.
06-21 17:19:45.484 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.8.0 TimeTicks 2
06-21 17:19:45.521 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.2.1 OBJECT IDENTIFIER 1.3.6.1.6.3.1
06-21 17:19:45.557 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.2.2 OBJECT IDENTIFIER 1.3.6.1.6.3.16.2.2.1
06-21 17:19:45.609 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.2.3 OBJECT IDENTIFIER 1.3.6.1.6.3.11.3.1.1
06-21 17:19:45.654 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.2.4 OBJECT IDENTIFIER 1.3.6.1.6.3.15.2.1.1
06-21 17:19:45.699 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.2.5 OBJECT IDENTIFIER 1.3.6.1.6.3.10.3.1.1
06-21 17:19:45.756 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.3.1 OBJECT STRING The MIB module for SNMPv2 entities
06-21 17:19:45.809 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.3.2 OBJECT STRING View-based Access Control Model for SNMP.
06-21 17:19:45.857 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.3.3 OBJECT STRING The MIB for Message Processing and Dispatching.
06-21 17:19:45.927 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.3.4 OBJECT STRING The management information definitions for the SNMP User-based Security
06-21 17:19:45.984 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.3.5 OBJECT STRING The SNMP Management Architecture MIB.
06-21 17:19:46.027 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.4.1 TimeTicks 2
06-21 17:19:46.063 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.4.2 TimeTicks 2
06-21 17:19:46.103 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.4.3 TimeTicks 2
06-21 17:19:46.146 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.4.4 TimeTicks 2
06-21 17:19:46.193 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.1.9.1.4.5 TimeTicks 2
06-21 17:19:46.237 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.1.0 INTEGER 5
06-21 17:19:46.283 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.1.1 INTEGER 1
06-21 17:19:46.326 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.1.2 INTEGER 2
06-21 17:19:46.377 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.1.3 INTEGER 3
06-21 17:19:46.426 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.1.4 INTEGER 4
06-21 17:19:46.477 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.1.5 INTEGER 5
06-21 17:19:46.525 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.2.1 OBJECT STRING lo
06-21 17:19:46.571 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.2.2 OBJECT STRING eth0
06-21 17:19:46.616 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.2.3 OBJECT STRING teq10
06-21 17:19:46.660 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.2.1.2.2.2.4 OBJECT STRING sit0

```

## - 끝부분

```

06-21 17:20:52.568 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.108.110.111.101.95.1.2 INTEGER 4
06-21 17:20:52.616 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.97.108.108.1.1 INTEGER 1
06-21 17:20:52.668 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.97.108.108.95.1.0 INTEGER 1
06-21 17:20:52.721 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.97.108.108.95.1.1 INTEGER 1
06-21 17:20:52.772 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.97.108.108.95.1.2 INTEGER 1
06-21 17:20:52.822 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.108.110.111.101.95.1.0 INTEGER 1
06-21 17:20:52.874 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.108.110.111.101.95.1.1 INTEGER 1
06-21 17:20:52.927 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.108.110.111.101.95.1.2 INTEGER 1
06-21 17:20:52.977 813-813/com.example.jang_yongsu.snmp I/RESULT: 1.3.6.1.6.3.16.1.5.2.1.6.95.108.110.111.101.95.1.2 END

```