

# LOCO 프로토콜 분석과 구현

HeXA

김태훈

# LOCO 프로토콜

- TCP/IP를 바탕으로 작동하는 request and response 프로토콜
- 패킷 경량화를 목표로 제작된 비공개 프로토콜
- '겁나 빠른 황소' 프로젝트의 일환으로 개발됨
- 2011년 10월에 첫 번째 버전 공개 및 사용됨
- 카카오 톡 팀에서 자체적으로 디자인 및 구현

# 분석 과정

- 구 버전(1.0.0.0)의 윈도우 모바일 어플리케이션 분석
  - 대략적인 패킷 생성 과정 분석
- 안드로이드 패킷 분석
  - 현재 어플리케이션에서 사용되고 있는 프로토콜 분석
- 최근 버전(1.5.0.0)의 윈도우 모바일 어플리케이션 데 변조

\* 구 버전 1.0.0.0, 1.2.0.0 사용 불가능

\* 최신 버전 : 1.9.0.0 (2013.05.26)

# App에서 LOCO 서버와 통신 방법

- LOCO 서버에 대한 TCP 소켓이 연결되어 있는지 확인
  - 그렇지 않다면 재 접속 시도
- **FillBuffer** 함수를 통해 현재 커맨드에 해당하는 패킷 생성
- if(isSecureMode)
  - True & Login : handshake 패킷 생성 후 암호화된 커맨드 패킷 앞에 붙이고 전송
  - True & !Login : 커맨드 패킷을 암호화 하고 전송
  - False: 그대로 전송

## 윈도우 KakaoTalk 1.0.0.0 버전 KakaoTalkSLLib.dll

```
public void Send(LocoPacket sendPacket, string host, int port, bool isConnectionLessApi)
{
    DebugEx.Assert(sendPacket != null);
    DebugEx.Assert(host.Length > 0);
    DebugEx.Assert(port > 0);
    DebugEx.WriteLine(string.Format("LocoClientAgent.Send Host: {0}, {1} Method: {2} PacketId: {3} ", new object[] { host, port, sendPacket.Method, sendPacket.PacketId }));
    if ((!(this._TcpSocketClient.IsConnected || (this._TcpSocketClient.Host != host)) || ((this._TcpSocketClient.Port != port) || isConnectionLessApi)))
    {
        _SendPendingPacket_Info item = new _SendPendingPacket_Info {
            SendPendingPacket = sendPacket,
            Host = host,
            Port = port,
            IsConnectionLessApi = isConnectionLessApi
        };
        this._SendPendingPacket_Infos.Add(item);
        this._TcpSocketClient.ConnectAsync(host, port);
    }
    else
    {
        sendPacket.FillBuffer();
        _SendPacket_Timer state = new _SendPacket_Timer {
            SendPacketObj = sendPacket
        };
        this._SendPacket_Timer_Map[sendPacket.PacketId] = state;
        TimeSpan dueTime = TimeSpan.FromSeconds((double) KakaoLibModel.Current.LocoRetryItv);
        state.TimerObj = new Timer(new TimerCallback(this._SendPendingTimerCallBack), state, dueTime, dueTime);
        if (this.IsSecureMode)
        {
            if (sendPacket.Method == "LOGIN")
            {
                LocoSecureHandShakePacket packet = new LocoSecureHandShakePacket();
                packet.FillBuffer();
                LocoSecureNormalPacket packet2 = new LocoSecureNormalPacket();
                packet2.DecryptedDataBlock = sendPacket.Buffer;
                packet2.FillBuffer();
            }
        }
    }
}
```

LOCOS 서버에 대한 TCP 소켓이 연결되어 있는지 확인

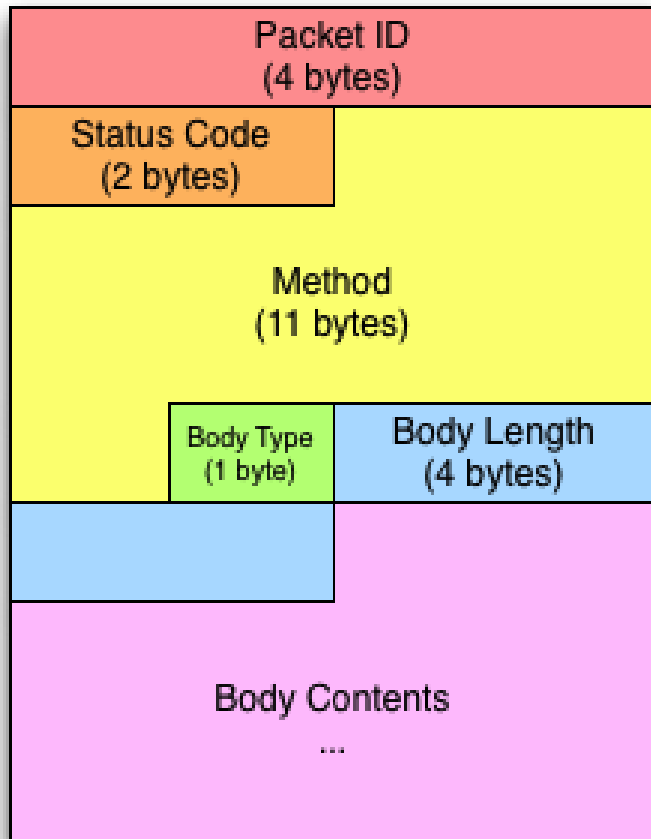
FillBuffer 함수를 통해 현재 커맨드에 해당하는 패킷 생성

True & Login : handshake 패킷 생성 후 암호화된 커맨드 패킷 앞에 붙이고 전송

# LOCO 패킷 종류

- **LocoPacket** : non-secure 패킷
  - 암호화 되어있지 않음
- **LocoSecureNormalPacket**
  - AES encryption 적용된 secure 패킷
- **LocoSecureHandShakePacket**
  - LOGIN 커맨드 패킷에 사용되는 handshak 패킷

# LocoPacket



- 가장 기본이 되는 패킷
- Packet ID
- Status Code : 0
- Method
  - Ex) LOGIN, ADDMEM , ACHATLIST
- Body Type : 0
- Body Length
- Body Contents
  - bson 형태로 전송됨

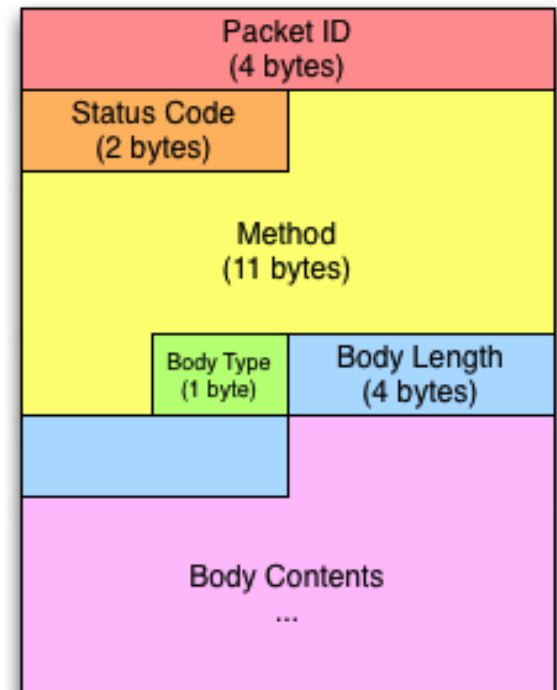
## 윈도우 KakaoTalk 1.0.0.0 버전 KakaoTalkSLLib.dll

```
public class LocoPacket : PacketBase, INotifyPropertyChanged
{
    // Fields
    private byte[] _Body;
    private int _BodyLength;
    private byte _BodyType;
    private static int _CurrentPacketId;
    private string _Method;
    private int _PacketId;
    private short _StatusCode;
    private static StringToAsciiEncodedBytesConverter _StringToAsciiEncoded;
    private PropertyChangedEventHandler PropertyChanged;

    // Events
    public event PropertyChangedEventHandler PropertyChanged;

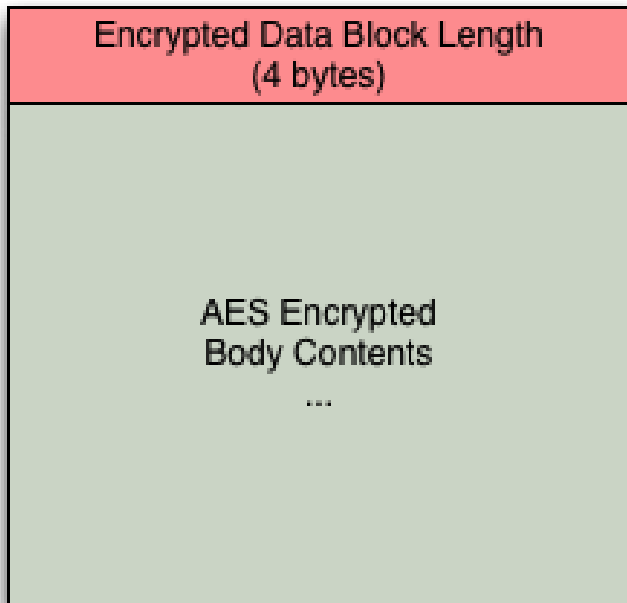
    // Methods
    static LocoPacket();
    public LocoPacket();
    public override void FillBuffer();
    public override int Parse(List<byte> recvData);
    private void This_PropertyChanged(object sender, PropertyChangedEventArgs e)

    // Properties
    public byte[] Body { get; set; }
    public int BodyLength { get; private set; }
    public byte BodyType { get; set; }
    public string Method { get; set; }
    public int PacketId { get; set; }
    public short StatusCode { get; set; }
```





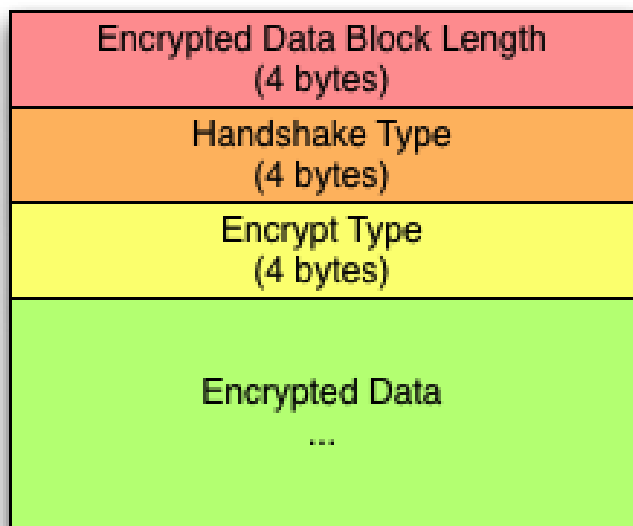
# LocoSecureNormalPacket



- secure 모드일 때 LocoPacket(주로 커맨드)을 AES 암호화 한 형태
- Data Length
- Encrypted Body contents

출처 : <http://www.bpak.org/>

# LocoSecureHandShakePacket



출처 : <http://www.bpak.org/>

- LOGIN 커맨드 앞에 붙여서 보내지는 패킷
- LOGIN 커맨드 : LOCO 서버와 세션을 열기 위해 필요한 커맨드
- AES 암호화에 사용되는 shared key를 서버에 전달하기 위함
- 이후 secure 패킷은 전달된 key로 암호화 되어 보내짐

# 커맨드 종류

- ADDMEM
- NOTIREAD
- LEAVE
- READ
- **BUY # non-secure**
- **CWRITE**
- **LOGIN**
- PING
- BLOCK
- NCHATLIST
- CHATON
- CHATOFF
- UPDATECHAT
- UNBLOCK
- UPSEEN
- CHATLIST
- **WRITE**

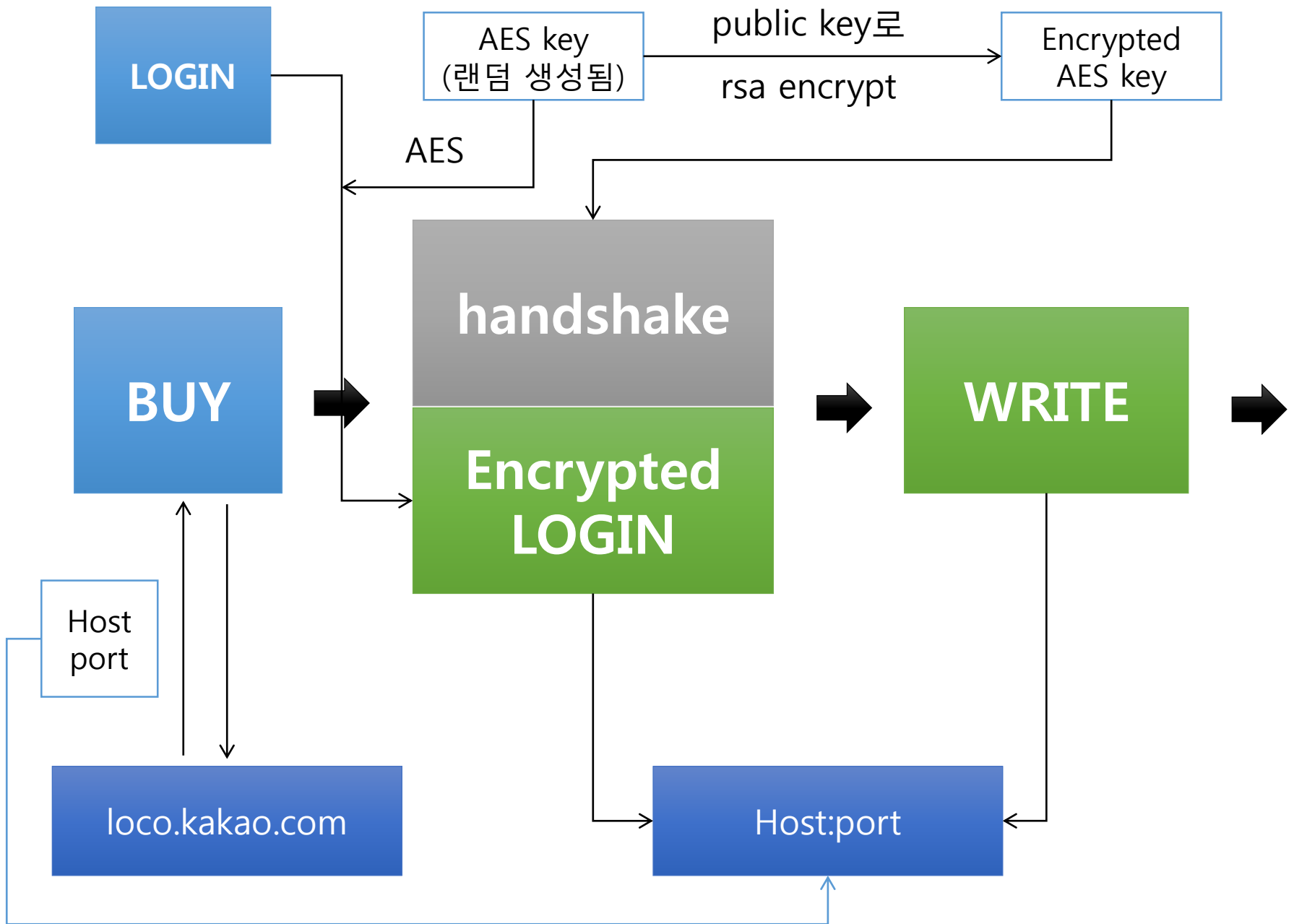
## 윈도우 KakaoTalk 1.0.0.0 버전 KakaoTalkSLLib.dll

MakeRequestSetting

Member	Declaring Type	Assembly
_MakeRequestSetting	KakaoTalkSLLib.LOCO.Base.LocoApiBase	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoAddMemApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoNotiReadApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoLeaveApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoReadApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoBuyApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoCWriteApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoLoginApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoPingApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoNChatListApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoChatOnApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoChatOffApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoUpdateChatApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoUnBlockApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoUpSeenApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoChatListApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoWriteApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
_MakeRequestSetting	KakaoTalkSLLib.LOCO.V1.LocoBlockApi	KakaoTalkSLLib, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null

← → 🔍 L...akeRequestSetting(LocoPacket) : Void ×

```
protected override void _MakeRequestSetting(LocoPacket requestLocoPacket)
{
    base.Host = KakaoLibModel.Current.LocoCarriageServerHost;
    base.Port = KakaoLibModel.Current.LocoCarriageServerPort;
    requestLocoPacket.Method = "LOGIN";
}
```



# BUY 커맨드

- LOCO 서버 정보 받음
  - loco.kakao.com 에서 소켓을 할당함
- non-secure 모드
  - 암호화 되지 않고 LocoPacket 형태로 그대로 전송됨

00000000	01 00 00 00	00 00	42 55 59 00 00 00 00 00 00 00 00 00 00	.....BU Y.....
00000010	00 00	64 00 00 00	64 00 00 00 12 75 73 65 72 49	..d...d. ...userI
00000020	64 00		02 6f 73 00 03 00	d..... ...os...
00000030	00 00 77 70 00 10 6e 74	79 70 65 00 03 00 00 00		..wp..nt ype.....
00000040	02 61 70 70 56 65 72 00	06 00 00 00 31 2e 30 2e		.appver. ....1.0.
00000050	31 00 02 4d 43 43 4d 4e	43 00 01 00 00 00 00 02		1..MCCMN C.....
00000060	63 6f 75 6e 74 72 79 49	53 4f 00 03 00 00 00 55		countryI SO.....U
00000070	53 00 08 76 6f 69 70 00	00 00		S..voip. ..

출처 : <http://www.bpak.org/>

# BUY 커맨드의 구현

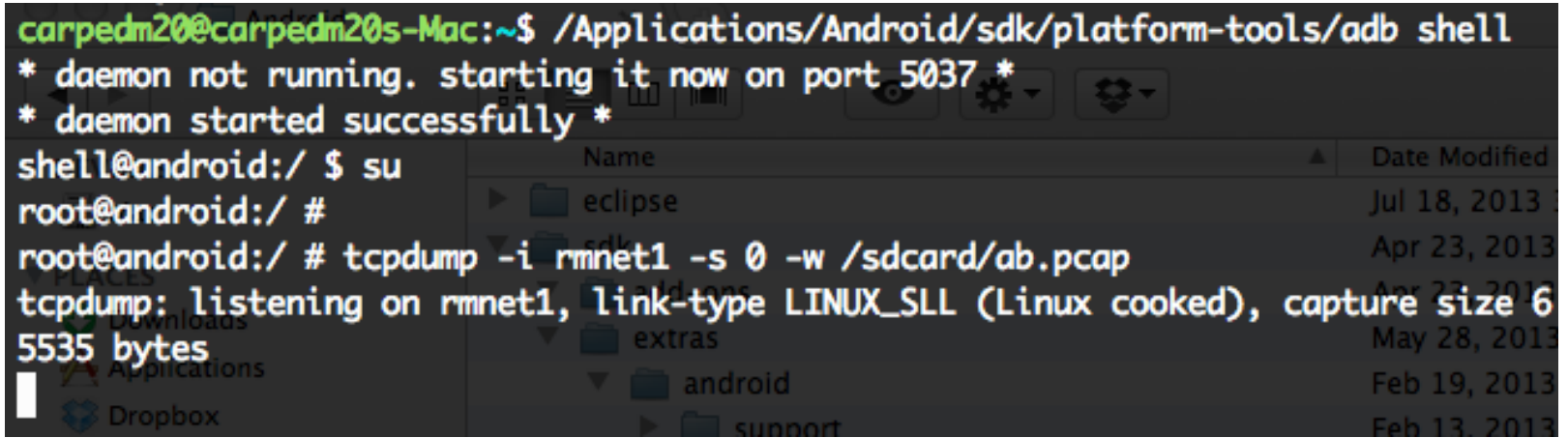
```
def buy():  
    # Alpha: 192.168.77.33:5555  
    # Sandbox: 110.76.140.115:9290  
    # Beta: 110.76.140.165:9282  
    # Real: loco.kakao.com  
  
    # port number when using 3G: 9282, 8080, 5223, 5242, 10009  
    # port number when using WIFI: 80, 8080, 5223, 5242, 10009  
  
    new = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
    new.connect(('110.76.141.20', 5228))  
  
    data = '\x01\x00\x00\x00' # Packet ID (4)  
    data += '\x00\x00' # Status Code (2)  
    data += 'BUY\x00\x00\x00\x00\x00\x00\x00\x00' # Method (11)  
    data += '\x00' # Body Type (1)  
  
    body = BSON.encode({'nType': 3, u'countryISO': u'KR', u'userId':  
6L, u'MCONEC': u'45005', u'appVer': u'1.0.1', u'os': u'android', u'voip': Fa  
lse})  
  
    data += body[:4] # Body Length (4)  
    data += body # Body Contents  
  
    new.sendall(data)  
    reply = new.recv(4096)
```

```
carpedm30@HeXA:~/python/kakao_command$ python test.py  
[*] BUY  
{u'status': 0, u'pingItv': 1200, u'host': u'110.76.141.202', u'reqTimeout':  
30, u'cacheExpire': 1800, u'lazyWaitItv': 1200, u'port': 5228, u'deepSleepIt  
v': 600}  
carpedm30@HeXA:~/python/kakao_command$
```

# 안드로이드 패킷 분석

- ADB (Android Debug Bridge) 사용
- tcpdump를 사용해 패킷 덤프
  - 설치에 super user 권한 필요
- Wireshark로 패킷을 hex 형태로 분석

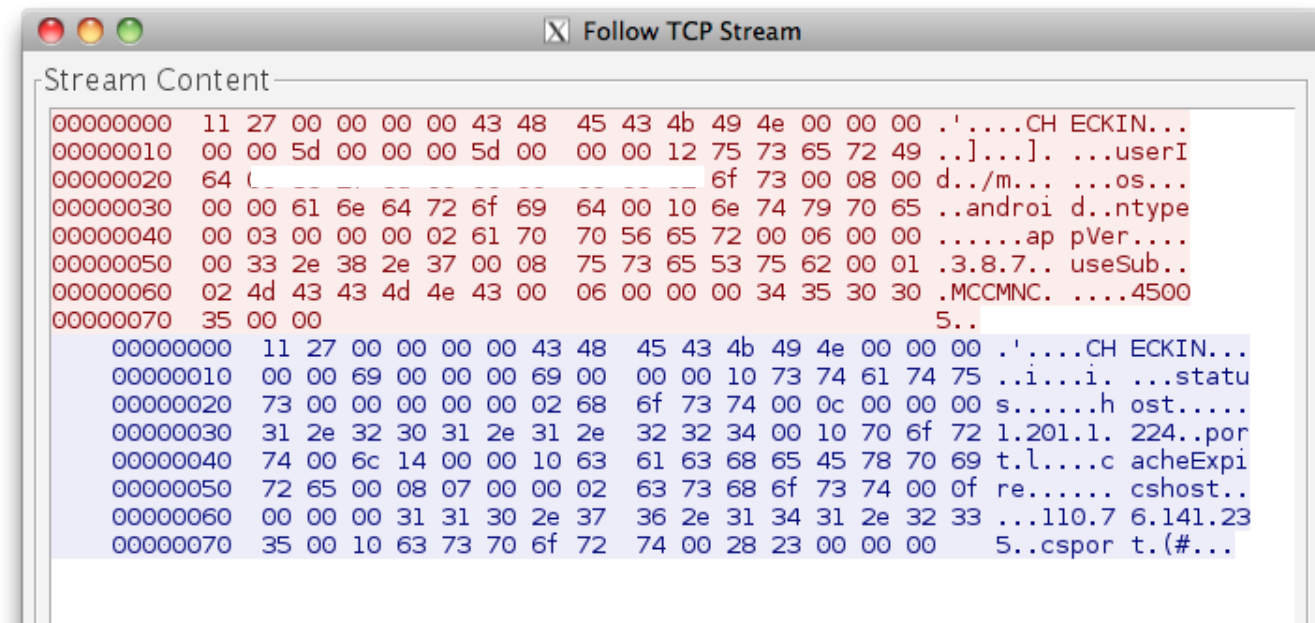
```
carpedm20@carpedm20s-Mac:~$ /Applications/Android/sdk/platform-tools/adb shell
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
shell@android:/ $ su
root@android:/ #
root@android:/ # tcpdump -i rmnet1 -s 0 -w /sdcard/ab.pcap
tcpdump: listening on rmnet1, link-type LINUX_SLL (Linux cooked), capture size 65535 bytes
```

A screenshot of a terminal window on a Mac. The terminal shows the execution of ADB shell, su, and tcpdump commands. A file explorer window is overlaid on the terminal, showing a directory structure with folders like eclipse, sdk, extras, android, and support. The terminal output shows that tcpdump is listening on rmnet1 with a capture size of 65535 bytes.



# CHECKIN 커맨드

- 이미 세션이 열려있는 경우 and 세션이 만료되지 않은 경우 사용됨
- 형태는 BUY 커맨드와 유사함



```
00000000 11 27 00 00 00 00 43 48 45 43 4b 49 4e 00 00 00 .'....CH ECKIN...
00000010 00 00 5d 00 00 00 5d 00 00 00 12 75 73 65 72 49 ..]...]. ...userI
00000020 64 00 00 00 00 00 00 00 00 00 6f 73 00 08 00 d../m... ..os...
00000030 00 00 61 6e 64 72 6f 69 64 00 10 6e 74 79 70 65 ..androi d..ntype
00000040 00 03 00 00 00 02 61 70 70 56 65 72 00 06 00 00 .....ap pVer....
00000050 00 33 2e 38 2e 37 00 08 75 73 65 53 75 62 00 01 .3.8.7.. useSub..
00000060 02 4d 43 43 4d 4e 43 00 06 00 00 00 34 35 30 30 .MCCMNC. ....4500
00000070 35 00 00                                     5..

00000000 11 27 00 00 00 00 43 48 45 43 4b 49 4e 00 00 00 .'....CH ECKIN...
00000010 00 00 69 00 00 00 69 00 00 00 10 73 74 61 74 75 ..i...i. ...statu
00000020 73 00 00 00 00 00 02 68 6f 73 74 00 0c 00 00 00 s.....h ost.....
00000030 31 2e 32 30 31 2e 31 2e 32 32 34 00 10 70 6f 72 1.201.1. 224..por
00000040 74 00 6c 14 00 00 10 63 61 63 68 65 45 78 70 69 t.l....c acheExpi
00000050 72 65 00 08 07 00 00 02 63 73 68 6f 73 74 00 0f re..... cshost..
00000060 00 00 00 31 31 30 2e 37 36 2e 31 34 31 2e 32 33 ...110.7 6.141.23
00000070 35 00 10 63 73 70 6f 72 74 00 28 23 00 00 00 5..cspor t.(#...
```

# Handshake 패킷

```
00000000 80 00 00 00 01 00 00 00 01 00 00 00 52 bf 59 05 .....R.Y.
00000010 0c 99 c0 a0 4b af 35 ba 66 3e de 6e 6f 72 b1 c5 ....K.5. f>.nor..
00000020 1e d6 b3 48 23 54 b7 01 0a e9 fd e4 e0 11 ed bb ...H#T..
00000030 4c e5 04 4f 32 d6 29 d4 fa 55 9a 9b 62 7b ed 7b L..02.). U..b{.
00000040 7e 06 f6 a9 f1 0a 4a 2f 4a 28 fb 24 14 e4 6d 4d ~....J/ J(.$..mM
00000050 30 2e 43 d7 ff 3f 6d 95 5f c2 3f 1a 1a 62 f7 08 0.C..?m. _.?..b..
00000060 cd 50 c4 00 23 d8 9c be a0 e8 46 5d 21 af 5c f5 .P..#... ..F]!.\.
00000070 ee 1c 24 43 a5 3c 10 2a 25 62 5d 4c 1c 17 1f a5 ..$C.<.* %b]L....
00000080 d9 15 63 a1 b5 f5 8c 91 1e ae cd b8 ..C.....
0000008C 10 01 00 00 75 b7 25 ac 8f d7 b9 53 f1 4d 5a 76 ....u.%. ...S.MZv
0000009C 31 b7 8e 9c 9f 8d 5e 7a b5 a6 22 19 f7 3b 3f d3 1.....^z .."?;?.
000000AC 6c cc 54 65 d5 7b c3 5f 58 aa ac 61 59 8a 25 0c l.Te.{_ X..aY.%.
000000BC d8 c7 85 46 6f 05 2d b2 86 dc 55 e6 87 0f 40 f2 ...Fo.-. ..U...@.
000000CC 1e d0 9a fd 3a e0 a4 29 0c e8 25 dc 8d a9 b5 d4 .....) ..%.....
000000DC d5 ac e9 57 d6 23 2d 4f 07 13 97 7c 15 81 a9 36 ...W.#-0 ...|...6
000000EC 1d 78 4a 6d 99 14 db c9 59 3d f8 c8 87 36 76 02 .xJm.... Y=...6v.
000000FC c1 23 6c 05 26 f3 24 e7 52 f9 59 39 33 98 b0 55 .#l.&$. R.Y93..U
0000010C 6e cd 71 73 ef 49 22 f7 f6 73 12 e9 9e a9 c7 cb n.qs.I". .s.....
0000011C 5f d6 6d 75 f7 c7 d3 c0 21 72 51 6d f7 b0 17 ff _mu.... !rQm....
0000012C 08 8f 23 e3 a0 66 3d d1 2e 93 41 61 67 23 8b 99 ..#...f=. ..Aag#..
0000013C 55 7c 80 c4 1f b5 54 9d dd 2c de d5 1b d4 14 ba U|....T. ,.....
0000014C aa 79 98 f5 b8 b6 60 96 19 01 d8 3a 21 59 e1 6a .y.....` ....!Y.j
0000015C d1 58 92 76 ce a3 e7 14 3d 20 ac a1 76 91 3f 42 .X.v.... = ..v.?B
0000016C b4 38 f4 3a 25 57 05 30 b1 f2 7f e3 3f 48 16 78 .8.:%W.0 ....?H.x
0000017C 7f d2 21 eb d0 3f 62 88 34 7a 76 c7 b6 8e b3 03 ..!...?b. 4zv.....
0000018C 77 97 53 cb 3c 90 eb f6 22 06 a9 05 a5 66 12 23 w.S.<... "....f.#
0000019C e7 9f 94 e0 .....
00000000 60 00 00 00 75 b7 25 ac 8f d7 b9 53 f1 4d 5a 76 `...u.%. ...S.MZv
00000010 31 b7 8e 9c 7f b1 f4 37 b6 38 37 da 44 c2 fe bb 1.....7 .87.D...
00000020 56 5b ba d0 b5 09 37 03 b1 2e 8b 85 9d 78 5c 5b V[....7. ....x\[
00000030 57 bf 43 74 18 e1 9f 98 ab 54 37 37 10 8f df f6 w.Ct.... .T77....
00000040 f9 17 21 3c 22 57 6d 39 3f d1 a7 ca 1c 97 06 50 ..!<"Wm9 ?.....P
00000050 08 b4 81 44 f3 42 1f b4 ff 42 f0 ad 09 cd 36 39 ...D.B.. .B....69
00000060 3b 45 51 75 ;Equ
000001A0 60 00 00 00 7f 9c a3 31 f0 7a dd 5d 17 e6 c2 a7 `.....1 .z.]....
000001B0 63 96 1e ba b6 d6 43 ba de de 98 06 62 95 d7 ed c.....C. ....b...
```

Handshake  
with RSA  
encrypted  
AES key

AES  
Encrypted  
Login

AES  
Encrypted  
Response

# LOGIN 커맨드의 구현

```
def handshake():  
    hand = '\x80\x00\x00\x00'  
    hand += '\x01\x00\x00\x00' # RSA = 1, DH = 2  
    hand += '\x01\x00\x00\x00' # AES_CBC=1, AES_CFB128=2, A  
    hand += rsa(aes_key)  
    return hand  
  
def login():  
    data = '\x04\x00\x00\x00' # Packet ID  
    data += '\x00\x00' # Status Code : when sending command  
    data += 'LOGIN\x00\x00\x00\x00\x00\x00' # Method  
    data += '\x00' # Body Type : when sending command -> 0  
    body = RSA_encrypt(hand, aes_key)
```

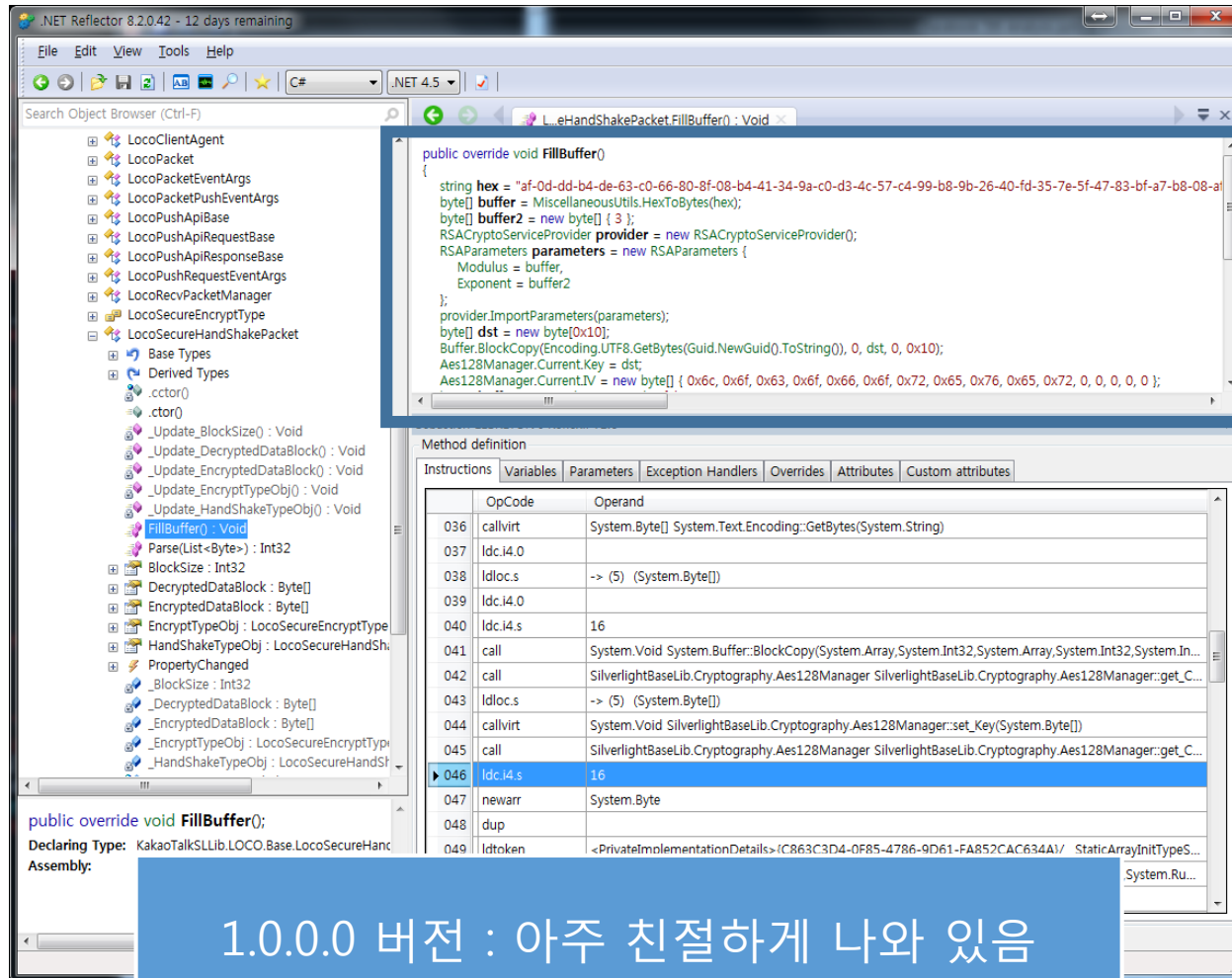
Handshake  
with RSA  
encrypted  
AES key

AES  
Encrypted  
Login

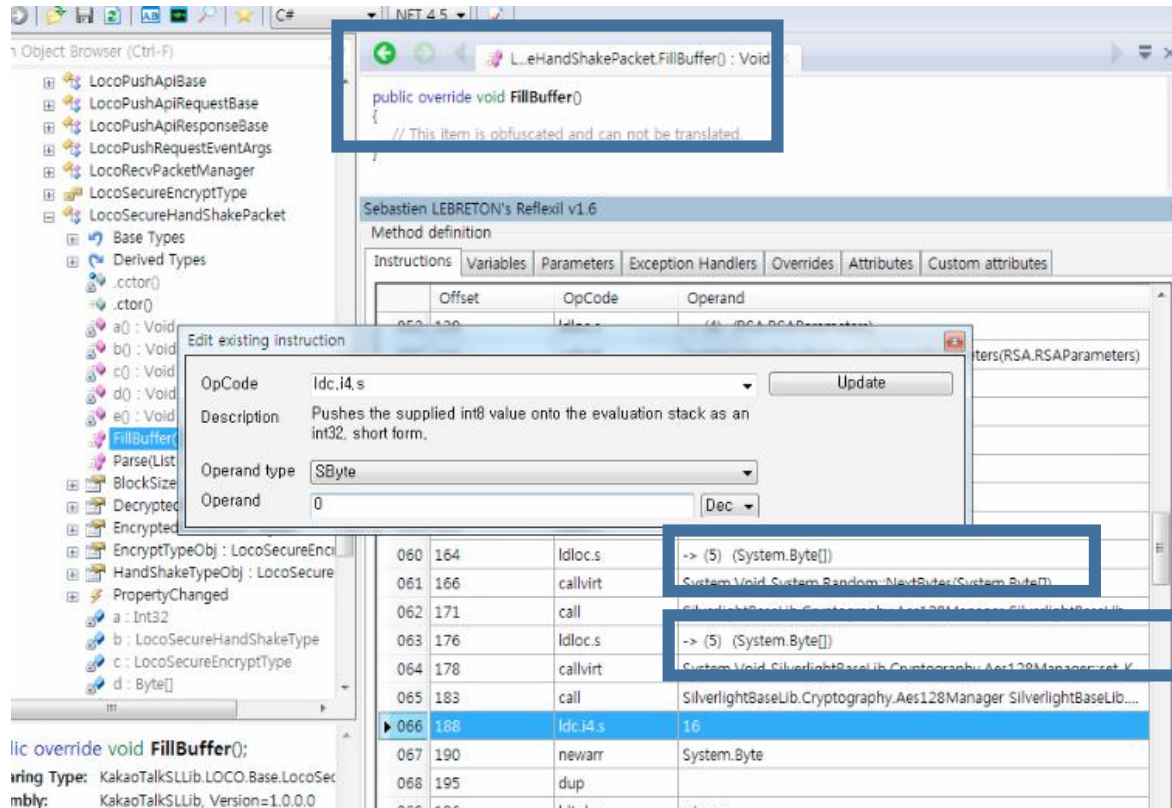
```
carpedm30@HeXA:~/python/kakao_command$ ./main.py  
[*] START  
[*] CHECKIN  
[!] HOST : 1.201.1.234 Jong Kim is now following  
[!] PORT : 5228  
[*] LOGIN  
[LOGIN] {u'status': 0, u'userId': 'Kyu-Yul Lee'}  
[*] WRITE to 42865071710223 : 123
```

Login

# 최근 윈도우 모바일 앱 리버싱



# 최근 윈도우 모바일 앱 리버싱



Todo : AES 키를 random이 아닌 이미 아는 키로 바꿔주어야 함

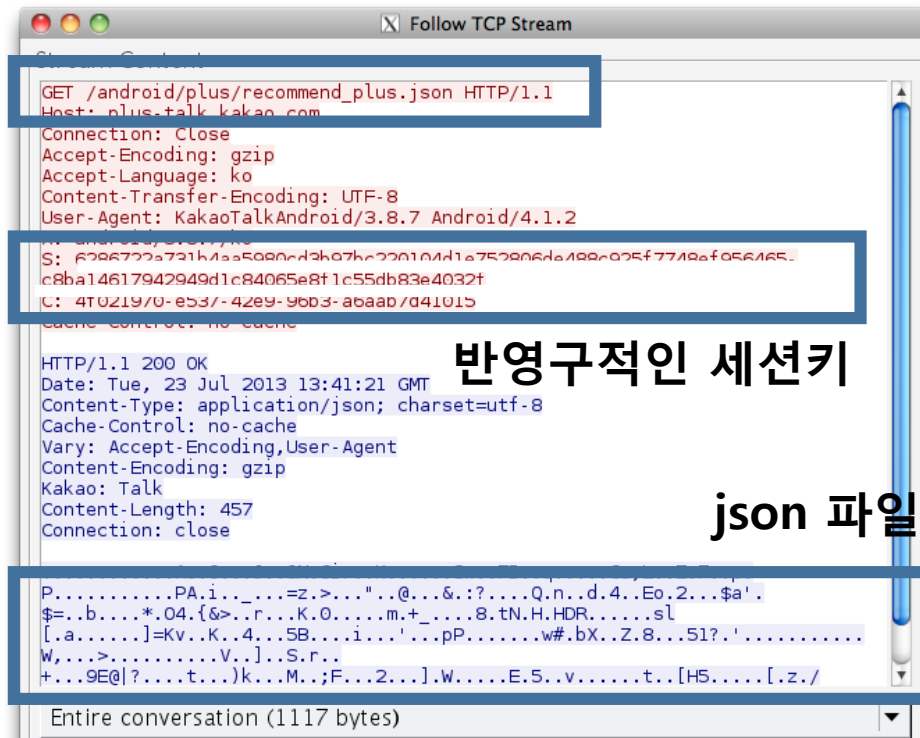
# Write 커맨드의 구현

```
def write(s, chatId = 42805871718223L, msg = u'test'):  
    print "[*] WRITE to " + str(chatId) + " : " + msg  
  
    data = '\x06\x00\x00\x00' # Packet ID  
    data += '\x00\x00' # Status Code : when sending command  
    data += 'WRITE\x00\x00\x00\x00\x00\x00' # Method  
    data += '\x00' # Body Type : when sending command ->  
  
    print msg  
  
    body = BSON.encode({u'opt':
```

```
carpedm30@HeXA:~/python/kakao_command$ ./main.py  
[*] START  
[*] CHECKIN  
[!] HOST : 1.201.1.234 Joong Kim is now following  
[!] PORT : 5228  
[*] LOGIN  
[LOGIN] {u'status': 0, u'userId': 99036740L}  
[*] WRITE to 42805871718223L : 123  
123  
[WRITE] {u'status': 0, u'chatLog': {u'msgId': 0, u'sendAt': 1375017225, u'logI  
d': 452656123451242520L, u'authenId': 88826740L, u'attachmet': None, u'messag  
e': u'123', u'type': 1, u'chatId': 42805871718223L}, u'logId': 452656123451242520L,  
u'chatId': 42805871718223L, u'errMsg': None, u'authorNickname': u'carpedm30',  
u'authorId': 452656123451242520L}
```

# LOCO 프로토콜 외의 통신

- SSL 층에서 통신함 (https)
- But 3g 네트워크를 사용하면 http 형태로 분석 가능



```
GET /android/plus/recommend_plus.json HTTP/1.1
Host: plus.talk.kakao.com
Connection: Close
Accept-Encoding: gzip
Accept-Language: ko
Content-Transfer-Encoding: UTF-8
User-Agent: KakaoTalkAndroid/3.8.7 Android/4.1.2

S: 6786772a721k4aa5080cd3k07kc220104d1e752806da189-025f7718af056165...
c8ba14617942949d1c84065e8f1c55db83e4032f
C: 4T0219/0-e53/-4ze9-96b3-abaab/d4101b

HTTP/1.1 200 OK
Date: Tue, 23 Jul 2013 13:41:21 GMT
Content-Type: application/json; charset=utf-8
Cache-Control: no-cache
Vary: Accept-Encoding,User-Agent
Content-Encoding: gzip
Kakao: Talk
Content-Length: 457
Connection: close

P.....PA.i.....=z.>...".@...&.?:...Q.n..d.4..Eo,2...$a'.
$.b....*.04.{&>...r...K.O.....m.+.....8.tN.H.HDR.....sl
[.a.....]=Kv..K..4...5B....i...'.pP.....w#.bX..Z.8...51?..'.....
W,...>...V...].S.r..
+...9E[?...t...k...M...;F...2...].W....E.5..v.....t..[H5....[.z./

Entire conversation (1117 bytes)
```

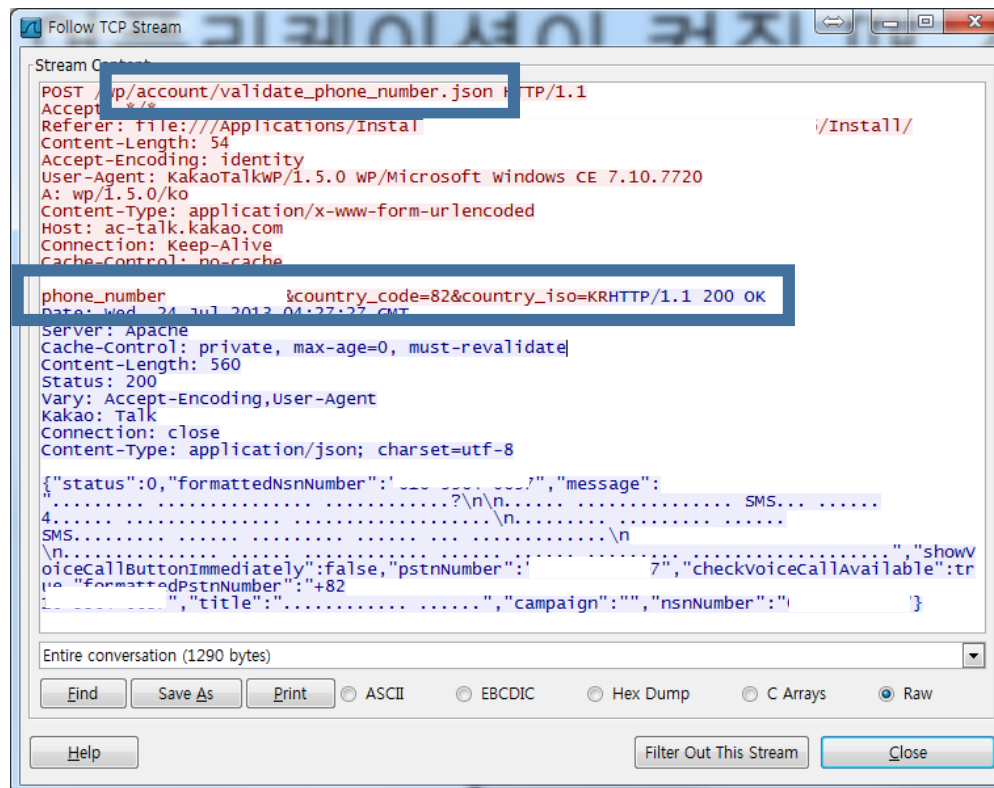
반영구적인 세션키

json 파일



# LOCO 프로토콜 외의 통신

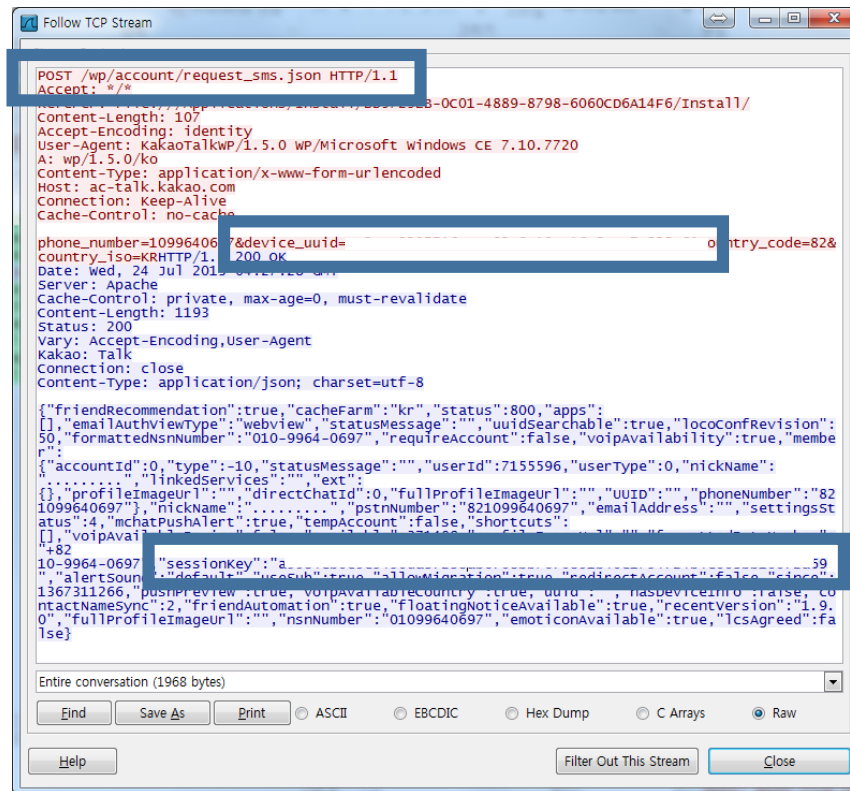
- 어플리케이션이 켜질 때 가장 먼저 보내지는 패킷





# LOCO 프로토콜 외의 통신

- 새로운 세션키를 요청하는 패킷



```
POST /wp/account/request_sms.json HTTP/1.1
Accept: */*
Content-Length: 107
Accept-Encoding: identity
User-Agent: KakaoTalkWP/1.5.0 WP/Microsoft Windows CE 7.10.7720
A: wp/1.5.0/ko
Content-Type: application/x-www-form-urlencoded
Host: ac-talk.kakao.com
Connection: Keep-Alive
Cache-Control: no-cache

phone_number=1099640697&device_uuid=...&country_code=82&country_iso=KRHTTP/1.1 200 OK
Date: Wed, 24 Jul 2012 07:12:00 GMT
Server: Apache
Cache-Control: private, max-age=0, must-revalidate
Content-Length: 1193
Status: 200
Vary: Accept-Encoding,user-Agent
Kakao: Talk
Connection: close
Content-Type: application/json; charset=utf-8

{"friendRecommendation":true,"cacheFarm":"kr","status":800,"apps":
[],"emailAuthViewType":"webview","statusMessage":"","uuidsearchable":true,"locoConfRevision":
50,"formattedNsnNumber":"010-9964-0697","requireAccount":false,"voipAvailability":true,"membe
r":
{"accountId":0,"type":-10,"statusMessage":"","userId":7155596,"userType":0,"nickName":
".....","linkedServices":{},"ext":
{"profileImageUrl":"","directChatId":0,"fullProfileImageUrl":"","uid":"","phoneNumber":"82
1099640697","nickName":"","psnNumber":"821099640697","emailAddress":"","settingsSt
atus":4,"mchatPushAlert":true,"tempAccount":false,"shortcuts":
[],"voipAvailability":true,"nsnNumber":"01099640697","emoticonAvailable":true,"lcsAgreed":fa
lse}
},
"sessionKey":"a...59
",
"alertSound":"default","useSub":true,"allowMigration":true,"redirectToAccount":false,"since":
1367311266,"pushReview":true,"voipAvailability":true,"uid":"","hasDeviceInro":false,"co
ntactNamesync":2,"friendAutomation":true,"floatingNoticeAvailable":true,"recentVersion":"1.9.
0","fullProfileImageUrl":"","nsnNumber":"01099640697","emoticonAvailable":true,"lcsAgreed":fa
lse}
```

# HTTPS 통신의 구현

```
carpedm30@HeXA: ~/python/kakao_command
lastUpdatedAt: 1374640562
newMessageCount: 32
chatId: [REDACTED]
carpedm30@HeXA:~/python/kakao_command$ python get.py
----- chatroom 0-----
LastMessage: ㄷ플로
LastLogId: 450432493804949505
lastUpdatedAt: 1374632937
newMessageCount: 2
chatId: [REDACTED]
----- chatroom 1-----
LastMessage: 책책책책
LastLogId: 450490683196295168
lastUpdatedAt: 1374639874
newMessageCount: 20
chatId: [REDACTED]
----- chatroom 2-----
LastMessage: ㅋㅋㅋㅋㅋ
LastLogId: 450496453199622144
lastUpdatedAt: 1374640562
newMessageCount: 32
chatId: [REDACTED]
carpedm30@HeXA:~/python/kakao_command$ █
```

Q&A