Development of an Over-the-Top Network Protocol for Pervasive, Secure and Reliable Data Transmission over GSM Short Messaging Service

William Emmanuel Yu, Ateneo de Manila University, and Pierre Tagle, Member, IEEE

Abstract—This paper outlines a new protocol for secure and reliable transmission of data over the GSM Short Messaging Service (SMS). The development of the protocol begins with the definition of allowable payload and the constraints due to mobile network conditions. Only particular types of applications will have low cost and pervasive connectivity requirements. A scheme for ensuring that the data transmission can be properly authenticated and encrypted will be applied. This is to provide the security requirements. For this protocol, EAX, an AE mode of operation, will be used for both authentication, integrity and privacy requirements. After that a scheme for protocol compression and encoding of the data will also be specified. This is to circumvent limitations inherent in the SMS protocol. LZ77 will be used for compression and encoding of the data. CRC16 will be used for per message integrity checking. In this document, an overview of the proposed protocol will be covered.

Index Terms—GSM, SMS, AES, EAX, LZ77, Security, Cryptography, Encryption

I. INTRODUCTION

Mobile communications networks have significantly grown to cover a significant portion of the globe. There are now currently close to over four (4) and half billion mobile connections as of 2009 (market penetration of over 65%) [21]. In addition to this, the average sale price of mobile handsets have significantly dropped through the years from over $250 to under $50 [12]. 80% of all mobile technology is provided using GSM networks [21]. It is even common to see $30 mobile handsets in the market today. A particular growing field of interest is machine-to-machine (M2M) technology. In a study for the GSM Association (GSMA), 89% of corporate respondents indicated that innovation in embedded mobile technology is very important to their organizations competitive future [5]. In another area, SMS is also predominantly used to enable mobile value added services and has the potential to function reliability despite the remoteness of certain locations. For this particular property, added safeguards and technology used to increase the reliability of data transmission despite the limitations of the actual network transport.

II. STATEMENT OF THE OBJECTIVES

This project aims to outline a protocol that takes advantage of the pervasiveness of GSM and at the same time propose remedies to its weaknesses. The central assumption is the use of the GSM mobile network as transport network of choice due to its pervasiveness and low cost. The potential uses in the machine-to-machine (M2M) are plentiful. The focus will be M2M applications that require a pervasive, secure and reliable data data transmission protocol such as Smart Meters, Tele-medicine, Remote ATM machines and Remote Voting Systems.

The objective can be summarized by these four (4) key properties:

- **Low cost.** The cost of the technology will significantly affect the cost of the entire M2M solution. If the solution is to be pervasive then it should be at an acceptable price point.
- **Wide Coverage.** The area in which the technology can be used is also crucial. In order to deploy a pervasive service such as Smart Meters, it is required that the technology is available in a wide geographical area.
- **Secure.** For particular M2M applications, it is crucial that the data being transmitted is kept secure. Security requirements may vary depending on the type of service and business requirements. Applications like remote banking would require privacy, authenticity and integrity while navigation systems would only need authenticity and integrity.
- **Reliable.** Finally, it is crucial that these system are able to function reliability despite the remoteness of certain locations. For this particular property, added safe guards and technology is used to increase the reliability of data transmission despite the limitations of the actual network transport.

The rest of the paper will describe the definition of the proposal protocol in terms of these four (4) key properties and the safe guards and technology used to provide each.

III. SIGNIFICANCE OF THE STUDY

By defining a protocol for low cost, pervasive, secure and reliable data communications, the scope and reach of M2M applications can be increased. For majority of the world where GSM is the most pervasive telecommunications technology, this opens up a new avenue of enablement. Efficiency can be obtained from intelligent telemetry and telematic applications (meters, health care, weather). Additional financial services
can reach far flung areas of a country where traditional banking services and voting systems cannot make their way.

IV. SCOPE AND LIMITATION OF THE STUDY

The study is limited to defining a network protocol to be used against the four (4) properties described above. GSM is assumed as the technology of choice due to coverage reasons. This study does not attempt to compare this protocol against any other protocols running on top of mobile data technologies such as GPRS, EDGE, 3G and HSPA.

Speed was removed as a consideration for this protocol due to the nature of the transmission requirement. In these particular cases, reliability and security are more important than transmission speed. In order to increase reliability and security additional mechanisms must be integrated into the protocol to provide these guarantees. These additional mechanisms typically consume additional resources and have a consequential effect on the speed of the entire system. When speed is a requirement for the application another protocol must be used.

V. REVIEW OF RELATED LITERATURE

A number of key technology components are necessary to provide the four (4) properties stated above. This section aims to quickly describe each of the principal component technologies used in the development of the protocol.

A. Mobile Cellular Networks

Global System for Mobile Communications (GSM) is a second generation cellular mobile technology that was developed by the European Telecommunications Standards Institute (ETSI) in 1987. GSM is a mobile cellular network that makes use of time division multiplexing and distributed antennas called Base Transceiver Stations (BTS also called cell sites) that cover a particular area called a cell[30]. This systems allows people to make voice calls and send small messages using a technology aptly called Short Messaging Service (SMS) which is a wildly popular service[19].

Various improvements in telecommunication have been developed since then. General packet radio service (GPRS) is an enhancement to GSM that allows sending of small amount of packet data. Enhanced Data rates for GSM Evolution (EDGE) is another enhancement to GSM that allows greater data rates than currently supported by GPRS. Currently, a technology called Wideband Code Division Multiple Access (WCDMA) is now being positioned and deployed as the next evolution in GSM technology. High Speed Packet Access (HSPA) is a set of protocols that aims to further increase performance of existing WCDMA networks[20].

Up to today, GSM and its descendants are world’s most deployed telecommunications technology. Of the GSM family, it is still vanilla GSM that is still the most pervasive covering 80% of mobile networks[21].

B. Information Security

Advanced Encryption Standard (AES) is an encryption standard that defines a 128-bit block cipher with key sizes of 128, 256 or 512-bit keys designed to replace the original Digital Encryption Standard (DES). AES was announced by National Institute of Standards and Technology (NIST) as Federal Information Processing Standard (FIPS) 197 on November 26, 2001[18]. Block ciphers are designed to provide data privacy by encrypting data (converting plaintext into ciphertext) using a secret key (applying symmetric key cryptography principles). Only parties with the secret key are able to decrypt the data (convert the ciphertext back to the plaintext).

Block ciphers can be used in various modes of operation. Each mode of operation can provide different information security benefits to the plaintext being processed[15]. The two (2) most popular schemes are Electronic Code Book (ECB) and Cipher Block Chaining (CBC). Both ECB and CBC modes of operation above were designed to only provide privacy guarantees. Hence, there was a need to develop additional modes of operation.

Authenticated Encryption (AE) modes of operation were then created to provide a way to protect both privacy and integrity capabilities while using block ciphers. In the study by Bellare[6], an Encrypt-then-MAC AE schemes is shown to provide the most guarantees over composition schemes and provides both integrity and privacy. Composite AE schemes are encouraged due to performance and security purposes. A well studied AE scheme can provide solid privacy and integrity guarantees. Like any crypto-system that depends on symmetric keys, the strength of the system resides primarily on the secrecy of the keys (secure key management).

The National Institute of Standards and Technology (NIST) has released a guideline describing an authenticated encryption mode for use by US Federal Agencies in NIST SP 800-38C[16]. The described block cipher mode of operation is called Counter with Cipher Block Chaining-Message Authentication Code (Counter with CBC-MAC or simply CCM) that was no intended for stream processing and defined to work around IPR issues with older AE schemes. Another simpler scheme gaining attention is EAX which was submitted by Bellare, Rogaway and Wagner as an alternative replacement to CCM[7].

C. Securing Mobile Networks

Being a widely available protocol for data transmission, there are a good number of studies that cover the security aspect of SMS and mobile data transmission. Each study primarily covers different approaches to overlay security on-top of presumably insecure SMS due to the compromise of GSM crypto-systems[8].

Chikomo el al[11] discusses secure mobile communications in the context of electronic mobile banking. Mobile banking application require a high degree of security due to their nature as highly regulated financial transactions. Chikomo’s approach to secure SMS transport uses one time pad (OTP) to alleviate the key exchange problem and AES (presumably in ECB/CBC
mode) and SHA1 (as an integrity hash) to provide privacy and integrity. This is a non-composite scheme[6].

Schmidt[32] provides a good overview of the built-in security infrastructure of mobile networks today. He notes that existing security mechanisms such as A5 and A3 potentially weak and suspect. He also notes that certain security guarantees such as non-repudiation are not provided by the existing crypto-system. He recommends supplementing it with an additional security infrastructure such as TLS/SSL which is also a non-composite approach.

Lo et al[26], discusses the vulnerabilities of the SMS and propose a PKI-based approach to securing SMS communications. Their approaches is an over the top approach that uses Java Wireless Messaging Architecture (JWMA). They use PKI (particularly RSA) as the key exchange mechanism to ensure secure transport of keys. Furthermore, they use AES in CTR mode and HMAC with SHA256 for integrity and privacy. This is an example of a non-composite scheme according to Bellare et al[6].

Croft et al[14] on the other hand, proposes a derived one-time pad (OTP) approach that involves various network elements in the crypto-system. This study aims to consider a purely over the top approach to security and focuses on the key exchange mechanism.

There are also other studies on SMS communications that involve denial of service attacks such as signaling starvation such as those by Traynor et al[33][34]. This study does not aim to address this particular aspect of security as it is principally involves the infrastructure layer. This is also the scenario painted by the network efficiency article[17]. This study focuses on building a composite over the top protocol assuming a pre-shared key.

D. Compression

The Lempel-Ziv Compression Algorithm (LZ77) is a sequential data compression algorithm developed by Abraham Lempel and Jacob Ziv in 1977. The algorithm is a universal coding scheme which can be applied to any discrete source and whose performance is comparable to certain optimal fixed code book schemes designed for completely specified sources [25]. The key benefit of this algorithm is the ability to run sequentially without the need to pre-process data. This significantly improves the performance versus algorithms that require a priori knowledge of the data.

E. Transport Integrity

Cryptographic integrity is already provided with the use of an AE scheme as discussed in a previous section. However, transport level integrity is typically used to check against accidental loss of data. It is therefore critical a mechanism be put in place to address transport level integrity concerns. The most common mechanism used is called a checksum. A common checksum is the Cyclic Redundancy Check (CRC) a form of which is used in the IEEE 802.3 standard[13] defined in Section 3.2.9 in the standard.

F. Mobile Network Performance

1) Measuring SMS Performance: Younis[37] gives an overview of the process in which SMS is transmitted and received in a GSM mobile network and the various factors that affect the performance of SMS networks in terms of failure rates. In the paper, four methods are discussed with each method varying in terms of assumptions on where the Mobile Station (MS) is currently camped on. Due to the nature of signaling overhead for establishing an SMS connection compared to GPRS, SMS setup times are expected to be higher than GPRS.

In a study by Meng et al[27][38], the authors measured the performance of SMS in terms of a failure ratio and latency. In this study, it was determined using network trace files from a Mobile Network Operator that SMS message delivery failure ratio was 5.1%. This refers to the number of messages that failed to deliver over the total number of messages sent. Of these messages that failed, 1.6% are due to timeouts and 3.5% are due to maximum retries reached. In this study, only 50% of users experience latencies below five (5) minutes.

2) Mobile Data Reliability: There are a number of studies relating to mobile data network reliability. The key common areas between these various studies is their measure of network reliability and performance which typically involve throughput, latency and failure ratio.

Korhonen[23] compares two (2) types of mobile data connectivity system: General Packet Radio Service (GPRS) and High Speed Circuit Switched Data (HSCSD). Korhonen demonstrated in this experiment that HSCSD performs better in terms of throughput than GPRS. Both services then had similar observed reliability (in terms of failure ratio).

Ruohonen[31] looks into the performance of GPRS networks in terms of quality (throughput and latency) and reliability (retransmission rate). A key item to note from this study is the behavior of the Subnetwork Dependent Convergence Protocol (SNDCP) which is responsible for compression and data transfer in GPRS. This study demonstrated concerns when using small MTUs which cause unnecessary triggering of the networks ARQ mechanism.

Meyer[28] studies the performance of TCP on GPRS. His studies places focus on the fact that layered protocols that implement their own Automatic Repeat Request (ARQ) functionality tend to perform poorly. In this case, both TCP and GPRS implement ARQ mechanisms. Meyers results were obtained through event simulation. In his simulation, he was able to show that the GPRS ARQ mechanism was sufficiently reliable due to the absence of time-outs and re-transmissions (failure scenarios). The main metric used by this study is round trip time (RTT) and block error rate (BLER) to measure performance. A similar study by Othman[29] looks into TCP performance in GPRS primarily with respect to throughput and latency. An interesting item to note from this study is that the IDLE mode mechanism of mobiles cause unwanted ARQ activity that possibly affect TCP performance.

The studies on SMS and GPRS performance above typically use three (3) measures of network performance: Throughput, Latency and Failure Ratio.
\((\text{LOWC}) \land (\text{COVER}) \land (\text{SEC}) \land (\text{REL}) \rightarrow \text{PROT} \)  \( (1) \)

Fig. 1. Network Protocol High Level Requirements

\[ \text{AE}(\text{COMP}(\text{DATA})) \]  \( (2) \)

Fig. 2. Initial Protocol Envelope

VI. PROTOCOL OVERVIEW

We start by first going back to the original principles discussed earlier in the paper which are low cost \( \text{LOWC} \), wide coverage \( \text{COVER} \), secure \( \text{SEC} \) and reliable \( \text{REL} \) as defined in Equation 1. We break each principles down further into actual requirements for the protocol.

A. low cost (\( \text{LOWC} \))

First, we need to design our protocol to be low cost to ensure broad acceptance. Our definition of low cost will be in terms of processing resources \( \text{LOWC} \text{PROC} \) used, bandwidth/storage \( \text{LOWC} \text{BAND} \) used and cost of transport facility \( \text{LOWC} \text{TRANS} \). There are numerous ways to minimize \( \text{LOWC} \text{BAND} \) and one of them is the use of compression as discussed in a previous section. Compression, however, is a time-space trade-off. Reducing \( \text{LOWC} \text{BAND} \) will cause an increase in \( \text{LOWC} \text{PROC} \).

We can then model the relationship as \( (\text{LOWC} \text{PROC}) \land (\text{LOWC} \text{BAND}) \land (\text{LOWC} \text{TRANS}) \rightarrow (\text{LOWC}) \). For this particular protocol, we will be assuming the use of a GSM network without assuming the availability of GPRS or any enhanced mobile data connectivity (3G, HSPA, HSPA+). We can, therefore, assume \( \top \rightarrow (\text{LOWC} \text{TRANS}) \) and thus \( (\text{LOWC} \text{PROC}) \land (\text{LOWC} \text{BAND}) \land \top \rightarrow (\text{LOWC}) \). In order to make \( \top \rightarrow (\text{LOWC}) \), we must make \( \top \rightarrow (\text{LOWC} \text{BAND}) \land (\text{LOWC} \text{PROC}) \). It can be assumed that \( \text{LOWC} \text{BAND} \) will be the more important requirement to be obtained. In a study by Carus and Mesut[10], it is shown that LZZ77 performs generally well and is the most stable with natural language data. Therefore, if compression (\( \text{COMP} \)) is applied before authenticated encrypted (AE) then we can obtain these benefits. If \( \text{COMP}(\text{AE}(\text{DATA})) \) then the data input to the \( \text{COMP} \) algorithm will not be a natural language. Therefore, our initial envelope will look like the Figure 2. With these constraints, we can assume to comply with the low cost requirement \( \top \rightarrow (\text{LOWC}) \).

B. Coverage (\( \text{COVER} \))

With the use of GSM technology and the additional assumption that we do not require stable GPRS connectivity, we will assume that \( \top \rightarrow (\text{COVER}) \).

C. Secure

The next step is to ensure that the message is secure. Security here is defined as ensuring the privacy and integrity of the messages. Also the intention is to use a mechanism with provable security. With the use of a composition-based encrypt-before-MAC authenticated encryption (AE) mode, Bellare and Namprempre[6] have shown that guarantees can be provided with respect to privacy (indistinguishability and non-malleability) and integrity.

The following guarantees when using this type of composite mode of operation[6]:

- **IND-CPA.** Indistinguishability under a chosen plaintext attack
- **IND-CCA.** Indistinguishability under a chosen ciphertext attack
- **NM-CPA.** Non-malleability under a chosen plaintext attack
- **INT-PTXT.** Integrity of plaintext
- **INT-CTX.** Integrity of ciphertext

This means that the data will remain private given the encryption (IND-CPA) and decryption (IND-CCA) algorithms as long as the secret key is not compromised (indistinguishability), messages cannot be changed without knowing the original plaintext (non-malleability), and it is computationally unfeasible to generate a ciphertext that decrypts to a message never encrypted by the sender or not produced by a sender (integrity). So as long as the key remains secret, we can assume that the message was encrypted by the sender (INT-PTXT and INT-CTX), the message was not tampered (NM-CPA) and the message cannot be read by people without the key (INT-CPA and INT-CCA).

We can select either CCM or EAX schemes that comply with the definition of composition-based encrypt-before-MAC mode. Both schemes support AES as their block cipher of choice. For this purposes of this study, we will be using EAX which has a readily available open source implementation using the Botan Cryptography Library[1]. We can then assume to comply with the security requirement \( \top \rightarrow (\text{SEC}) \). Our definition of secure is therefore \( (\text{IND-CPA}) \land (\text{IND-CCA}) \land (\text{NM-CPA}) \land (\text{INT-PTXT}) \land (\text{INT-CTX}) \rightarrow (\text{SEC}) \).

D. Reliability

Any network protocol will need to reliable to ensure that critical data gets to its intended recipient. However, it is the nature of data communications and medium to have a certain degree of unreliability. This is particularly true for mobile telecommunications networks. Another restriction is that this protocol is intended to be using SMS as transport. Binary SMS messages have a payload limit of 160 7-bit characters or 140 8-bit characters[4]. This requires our protocol to break the resulting payload of the first envelope into smaller packets. Other parameter such as encoding scheme, addresses and other fields are fixed making this protocol inflexible[4]. This necessitates the development of a protocol on top of SMS. The section above does describe some mechanism to determine integrity and non-malleability of the data sent. However, it does not cover how to manage the integrity of each of these component messages. In the review section, a technology that can be applied is a checksum. A checksum can be applied to each resulting smaller message. In addition, a header must be added to each message to enable the receiving element to
determine the part number and the total number of parts to expect. Retransmissions can be requested after a system defined time-out value.

The checksum used can be the IEEE 802.3 CRC which has been selected for its balance of performance and has enough resolution for the target payload[13]. Given the controls above, we can then assume to comply with the reliability requirement $T \rightarrow (REL)$.

E. Overview of Proposed Protocol

This section now aims to summarize the various aspects of the protocol described above. This section describes the steps taken to transform the data to allow it to be transmitted and received in line with the requirements specified in the previous parts of this paper.

1) Transmitting: The process flow for the transmitting leg of the protocol is shown in Figure 3.

![Fig. 3. Transmitter Protocol Flow](image)

Here is a short description of the key steps in the transmitting end of the protocol:

- **Get Datagram to be Transmitted.** This step involves getting the datagram to be transmitted from an input system. For this particular project, it is assumed that datagrams are read from a filesystem for simplicity. However, this section can be wired into another network protocol.

- **Compress.** This step compresses the content of the datagram according to the discussion in Section VI-A. As discussed in the mentioned section, compression occurs before encryption to take advantage of regularity of information data. Encrypted data will have less properties that compression algorithms can exploit. In the case above, the algorithm selected takes advantage of natural language properties.

- **Encrypt.** This step encrypts the content of the previous step according to the discussion in Section VI-C.

- **Breakdown into Smaller Packets.** This step breaks down the packet since each SMS can only contain a limited amount of data as discussed in Section VI-D.

- **Insert Envelope Per Small Packet.** This step discusses the envelop used on the resulting packets to maintain flow control as discussed in Section VI-D.

- **Add Checksum.** Then a checksum is added to each resulting packet as part of the added envelope also discussed in Section VI-D.

- **Transmit.** Then the resulting enveloped packets are transmitted.

2) Receiving: The process flow the receiving leg of the protocol is shown in Figure 4. This is more or less the reverse of the previous sub-section’s process.

![Fig. 4. Receiver Protocol Flow](image)

Here is a short description of the key steps in the receiving end of the protocol:

- **Receive.** Enveloped packets are initially by the receiving system. These packets can potentially be missing or out of order.

- **Validate Checksum**. The checksum of each packet is checked to determine if the packet was not damaged during transmission. This is also described in Section VI-D.

- **Remove Envelope.** Packets are initially collected and their envelopes removed to determine if they are complete. Envelope is described in Section VI-D.

- **Re-assemble Segment.** The header information contained in the enveloped of each packet is then used to determine if the packets are complete and then ordered appropriately. If some packets are missing after a defined time-out period then a retransmission of the missing packet is requested as described in Section VI-D.

- **De-cryption.** The re-assembled packet is then validated and decrypted according to Section VI-C. This provides a number of information security guarantees include the actual integrity check against intentional tampering.

- **De-compress.** The data is then decompressed according to Section VI-A.

- **Return Restored Datagram.** This step involves returning the transmitted datagram to the calling system. For this particular project, it will simply be written to the filesystem for subsequent processing. Like the first transmission step, this can be wired into another network protocol.

All key elements discussed previously are included in this
transmission and receiving cycles. Security, in terms of Privacy and Authentication, is provided using EAX mode of operation with an AES-128 cipher block cipher as specified in Section VI-C. Reliability is provided with the use of CRC (described in Section VI-D) as well as the integrity guarantees provided by EAX-AES (described in Section VI-C). Compression is used to further reduce the payload size given the low bandwidth nature of mobile networks (described in Section VI-A). Rudimentary flow control (described in Section VI-D) is provided to allow for packet retransmissions. Datagrams are compressed and encrypted before they are fragmented. In an on-going study[36], preliminary results show that there is a significant benefit to encrypting payload before fragmenting data. In another on-going study[24], preliminary results also show that fragmenting payload into smaller packets generally improve the overall throughput of the system even when used in a non-SMS based mobile data system.

VII. IMPLEMENTATION

The test system was built principally using the Python programming language and the following associated tools. The primary reason Python was chosen was for its flexibility as a scripting language and its large number of available libraries and utilities.

A. Physical Setup

The setup uses commodity computer systems running with the Linux operating system (other related projects also use MS Windows systems). These computer systems are provided access to cellular communications facilities with the use of USB dongles. These USB dongles are convenient as they are relatively low cost (under $100) and allow connectivity to a computing platform such as a laptop. These coupled with unlimited SMS and data plans allow use to perform these experiments with a tight budget. The ability to use a commodity computer system allow experiments to be run using sophisticated and flexible tools like Python. Python was selected due to its use as a scripting language that allows us to easily move around components and test different aspects of the protocol.

B. Software Components

In this study, the primary programming environment used is Python and the following key libraries and utilities were utilized: (a) pySerial[2] using AT commands[3] to communicate with the USB modems, (b) Botan, a C++ cryptographic library with Python bindings, will be used for the cryptographic routines[1]. We have already made a few patches to fix the current limitations of Botan, (c) pycrc is used to compute for the CRC with varying CRC polynomials.

Flow control was implemented directly using Python programming constructs. Due to the SMS payload limit of 140 bytes, it is necessary to break the data into smaller packets. SMS can be replaced with UDP (and its payload size varied) to allow comparison with regular data transmission on cellular data which is a focus of an on-going study[24]. Botan and pycrc allow us to vary the choice of cryptographic routine and CRC algorithm to create various experiments[36].

VIII. TESTING AND PERFORMANCE PROFILING

Performance of network systems are typically measured in terms of failure ratio and latency. Studies in Section V-F use these measures to quantify the performance of Mobile Data (GPRS) and SMS networks.

A. Failure Ratio

A common measure of network performance is failure rate, failure ratio or mean time between failure (MTBF). Failure rate refers to the number of errors that occur per unit of time. Failure ratio refers to the percentage of messages successfully transmitted versus the total number of messages sent. MTBF refers to the amount of elapsed time that occur between errors.

B. Latency/Response Time

Latency is typically defined as the amount of time it takes for a message to be transmitted. Response time refers to the turnaround time of the dialogue which includes the time it takes for the receiver to acknowledge receipt of the message. The required network latency is dependent on the actual application being supported. Typical monitoring applications can tolerate a higher amount of latency. Others like health care monitoring will require lower amounts of latency.

C. Throughput

Throughput refers to the absolute amount of data that can be transmitted given a particular circuit. For the purposes of this study, the choice of network technologies available will generally have the same amount of maximum available bandwidth (as a control). Therefore, throughput will generally be determined by the amount of data that is sent through the network making this factor entirely dependent on the latency of the network. The higher the latency the lower the general throughput.

IX. NEXT STEPS AND MOVING FORWARD

The current implementations[36][24] are written specifically to execute particular experiments. The next step would be to generalize these routines into a re-usable Python library that can be used by research groups moving forward to try out various aspects of the system by modifying particular parameters. This library can also be used on prototype systems to demonstrate efficacy of particular mechanisms by allowing experimenters the flexibility to vary their choice of cryptography, check sums, flow control and other parameters.

This library is not only valuable for SMS-based bearer. This protocol can be used with a data bearer using technologies such as GPRS, EDGE, WCDMA, HSPA, LTE and others. With the use of a data bearer, the same small packets and UDP transport, this protocol can be re-purposed to avoid the pitfalls currently associated with using mobile data in M2M applications such as excessive signaling and high failure rate[17].
X. CONCLUSION

In summary, this paper shows an outline of a proposed protocol designed to provide the guarantees: Low cost, Coverage, Security and Reliability. In terms of information security guarantees it is designed to provide both privacy and integrity.

REFERENCES


William Emmanuel S. Yu is currently head of Technology Operations at Novare Technologies and is also part of the Strategic Business Development group at Smart Communications. He is working on mobile telecommunication systems integration projects focused on fixed mobile convergence and enterprise mobility applications with mobile network operators and technology providers here and abroad. He is actively into Internet engineering, mobile platforms and information security research. He is also a faculty member at the Ateneo de Manila University and Systems and Network Consultant for the Ateneo Campus Network Group. He is currently taking up his PhD at the Ateneo de Manila.

Pierre U. Tagle is a recognized expert in industry and academy in the field of information and communications technology (ICT). Prior to launching Mohilbane Inc, Pierre has served as the Chief Information Officer of the SVI Group of Companies, taking on various roles from conceptualization, roll-out and support of SVIs various business lines in software services, call center, services and business process outsourcing (BPO) services. He has also served as IT/networking consultant in the conceptualization, design and implementation of various industry projects including IT parks (Northgate-Cyberzone, Fort Bonifacio), call centers and other ICT companies. Concurrent to his industry practice, he also served as Assistant Professor and Campus Network Manager of Ateneo de Manila University.