World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:9, No:7, 2015

Independent Encryption Technique for Mobile Voice Calls

Nael Hirzalla

***Abstract*—**The legality of some countries or agencies' acts to spy on personal phone calls of the public became a hot topic to many social groups' talks. It is believed that this act is considered an invasion to someone's privacy. Such act may be justified if it is singling out specific cases but to spy without limits is very unacceptable. This paper discusses the needs for not only a simple and light weight technique to secure mobile voice calls but also a technique that is independent from any encryption standard or library. It then presents and tests one encrypting algorithm that is based of Frequency scrambling technique to show fair and delay-free process that can be used to protect phone calls from such spying acts.

***Keywords*—**Frequency Scrambling, Mobile Applications, Real-Time Voice Encryption, Spying on Calls.

I. Introduction

SPYING on your personal or even business calls by a third party, no matter who this part may be, is very annoying and gives the feeling of privacy invasion. On October 23rd, 2013, BBC news in its article [1], mentioned that Mrs. Merkel, the German Chancellor, had told Mr. Obama: "Such practices must be prevented immediately." This was after Mrs. Merkel received information that the US may have spied on her mobile phone.

On May 23rd of that year, many News channels such as RT News, [2] reported what WikiLeaks' Julian Assange has said about the fact that the National Security Agency (NSA) has recorded almost all domestic and international phone calls in some countries such as the Bahamas. The Observer in its Sunday 27th of October 2013 issue also mentioned that Merkel's phone may have been monitored for over 10 years.

The Guardian in its May 4, 2013 issue mentioned that a former FBI counterterrorism agent claims that all telephone calls are recorded and they are made available to the US government to access them at its wish. On January 23rd, 2014, Bloomberg mentioned [3] that the U.S. privacy-policy board concluded that the National Security Agency's collection of bulk telephone data was illegal and should be stopped. While a federal appeals court ruled on May 7th, 2015 that the National Security Agency's collection of millions of Americans' phone records violates the USA Patriot Act. This came just before Congress decided after a contentious debate not to reauthorize the statute that underpins the NSA program. The question now is whether we would believe that such a ruling would be respected at all. Moreover, why does this ruling mention only Americans' phone records and not any other records?

Without hacking, one should know that the network provider can listen to your phone call directly. Moreover, in 2009, hackers published a detailed guide on how to intercept mobile phone calls made over GSM networks. In other words, your phone calls can be easily spied on by not only big agencies such as NSA but also individuals.

Since research is made to help the public, we decided to research ways to secure personal phone calls in a handy and easy way. Many applications do exist in hardware or software to encrypt voice calls off-line as well as real-time; however, they are not made easy to use or portable to carry during our everyday call. With most calls are made through mobile devices, one should carry either a big encrypting device or even a laptop that does the encryption/decryption and plug it in when making a call. This paper proposes a solution to offer such a tool in an easy and more portable way in order to make it available for all to use.

With the availability of smartphones and their low prices, the proposed idea is to develop a mobile light weight application, in terms of speed and overhead, that secures (encrypt and decrypt) voice communications in real-time. This is because the process should not only work in real-time, but also on smartphones. The paper will first give an overview to existing voice encryption systems. It then talks about the technical issues related to securing phone calls before proposing a simple and portable mechanism that can be implemented on portable or mobile phones.

II. Related Work Overview

Audio or voice encryption was one of the earliest digital applications. The first approaches for a voice encryption system were invented in the 1920th mainly at the Bell Labs between World War I and World War II. In the 1940th a far more secure system by Bell Labs was invented and used which was called Sigsaly [4]. This system weight 50 tons and was a very complex system but was considered the first real secure voice encryption system.

In the following decades (beginning in the 1960th) a big range of different systems has been developed to protect the privacy of the voice. First analogue and then, with the improvements of the digital signal processors (DSP), digital techniques were invented leading into pure software applications.

Although there were few applications that targeted real-time voice encryption, most of such inventions targeted recorded audio files. In non-real-time systems, the caller needs to record the message first. The recorded file then gets encrypted. The output file will then be sent digitally through normal telephone

N. Hirzalla is with the Najran University: 966-530-789237; (e-mail: nbhirzallah@nu.edu.sa.)

lines, or digital Mobile Networks. Voice encrypting applications do exist in either hardware or software.

For hardware Based Encryption Systems the following are just a snapshot that is used to secure voice calls:

- Digital Voice Protection (Motorola)
- STU III (Secure Telephone Unit, Generation III)

While for software Based Encryption Systems, the following are examples:

- PGPfone
- Nautilus

With the development of smartphones, it became much easier to develop a light weight encryption and decryption techniques that could run on a smartphone. The following are examples on VoIP secure calls:

- RedPhone
- Speak Freely
- SecureVoice

Unfortunately, none can be found for pure GSM voice calls. Furthermore, most of the encryption applications use one of following libraries for their algorithms:

- Microsoft CryptoAPI
- Microsoft SChannel
- OpenSSL library

These libraries use some pseudorandom number generator that is approved by the National Institute of Standards and Technology (NIST). Unfortunately, NIST itself is not clear from the National Security Agency friendly or unfriendly attacks. To elaborate more, take for example the story of Dual-EC. Dual-EC is a pseudorandom number generator proposed by NIST in 2006. Since then it was used by many systems for encryption techniques. Few months later Shumow and Ferguson from Microsoft claimed that NSA has inserted a backdoor into the Dual-EC. They showed the diagram shown in Fig. 1 in their presentation during CRYPTO convention of 2007. The thick green arrows of Fig. 1 mean 'this part is easy to reverse', while the thick red arrows should mean the opposite. However, they claimed that NSA can reverse them via the backdoor key they have from NIST.

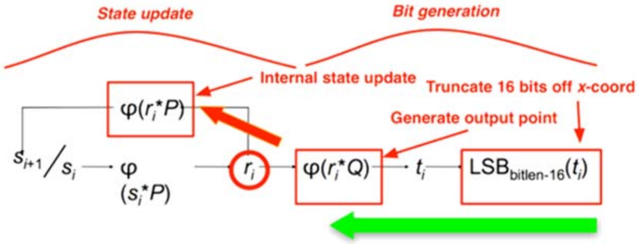


Fig. 1 Shumow and Ferguson diagram

This backdoor will allow the NSA to break nearly any cryptographic system that uses this pseudorandom number generator. Surprisingly after almost six years, and just around the 2013 year, NIST took the unprecedented step of warning implementers to avoid this pseudorandom number generator altogether [5].

Although some commercial voice encrypting applications vendors claim that they do not and never have used Dual_EC_DRBG in any of their products, such as SecureGSM™, but the fact that NIST took six years to acknowledge the backdoor, make one worry about other future acknowledgments regarding other functions and libraries. Hence, this paper explains few issues that are related to voice encryption so that one may develop his/her own voice encryption technique that may be simple yet independent from any other tools provider. The more techniques we have the harder it will be on hackers or spies to spy on calls.

III. Calls Over VoIP or GSM Need Encryption?

GSM stands for Global System for Mobile Communications. It is the dominant mobile phone communication protocol. In Europe, GSM broadcasts on 900 MHz and 1800 MHz. However, in the United States, GSM transmits on 850 MHz or 1900 MHz. GSM 2G forced the networks to change from an analog network to digital. The latest version of GSM has been named LTE (4G networks). GSM remains the most widely used protocol in the world; approximately 80 percent of all mobile phones operate on the GSM standard. The 2G technology is structured around the Time division multiple access (TDMA) method. While the 3G service is based on two primary protocols: Global System for Mobile communication (GSM) and Code Division Multiple Access (CDMA). Finally, the 4G technology is structured around OFDMA Internet Protocol (IP) packet-switched networks.

On the other hand, Voice over Internet Protocol (or VoIP) telephony has become increasingly popular. It was first introduced in 1995. Fig. 2 (taken from [6]) shows the three main technologies used to perform voice calls: the old PSTN, the GSM, and the VOIP.

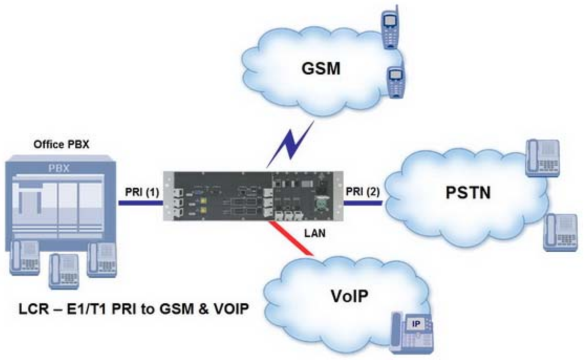


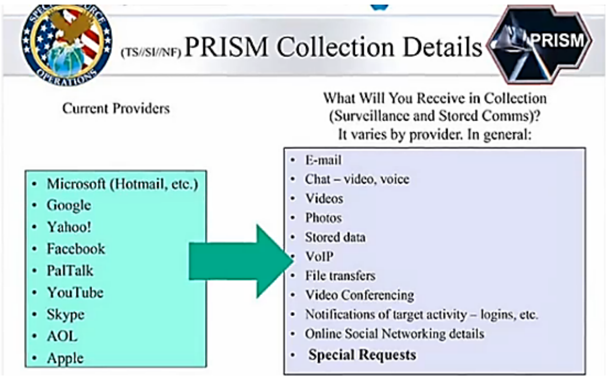
Fig. 2 PSTN, the GSM, and the VOIP

One thing that should be clear is that regardless how the call is initiated, the route of any call depends on the receiver device and how is it connected. It is usually clear for VoIP devices as well as traditional PSTN phones. For the VoIP devices, the call stays over IP. While for traditional PSTN devices, the call will enter the PSTN network through the VoIP-PSTN Gateway. However, for mobile phones the case is different. If the mobile phone device is connected through the internet, whether through a LAN or the Mobile Data, the call stays over IP. While if it is not connected to the internet, then the call will enter the GSM network through the VoIP-GSM Gateway.

Since this paper focuses on mobile phones, then we can easily say the following. VoIP may need GSM to complete a call, but GSM may skip VoIP to complete a call. Worldwide statistics on the usage rate of VoIP calls versus GSM calls are nowhere to be found. It is obvious that the decision to initiate a call over VoIP versus GSM depends on the following:

- Internet availability
- Internet Speed through the call path
- Charging rate for the call, which in turn depends on
- Per minute or per data bytes
- International, Long distance, or Local
- Same Network or different Network
- With Video or without Video

Now the question that is raised, Is VoIP calls more secure? To answer this question let's take you to the PRISM program. PRISM program allows the U.S. intelligence community to gain access from nine Giant Internet companies as shown in Fig. 3, which is taken from Washington Post, June 6, 2013 issue, such as Yahoo, Google, Skype, and Facebook. In other words, the government does not need a case by case court approval to gain access to the packets travelling over IP.



(TS//SI/NF) PRISM Collection Details

Current Providers

- Microsoft (Hotmail, etc.)
- Google
- Yahoo!
- Facebook
- PalTalk
- YouTube
- Skype
- AOL
- Apple

What Will You Receive in Collection (Surveillance and Stored Comms)?
It varies by provider. In general:

- E-mail
- Chat – video, voice
- Videos
- Photos
- Stored data
- VoIP
- File transfers
- Video Conferencing
- Notifications of target activity – logins, etc.
- Online Social Networking details
- **Special Requests**

Fig. 3 PRISM from Washington Post, June 6, 2013 issue

Thus, the answer for this question is No. VoIP calls are not secure as well. This makes voice encryption inevitable in order to secure a mobile call.

The next question that may arise is as follows. Is it better to implement voice encryption technique over VoIP or over GSM? Implementing an encryption technique for GSM calls does not cover pure VoIP calls, and implementing an encryption technique for VoIP calls does not cover pure GSM calls. Hence, it is best to implement an encryption technique between the Hardware sound system and the Mobile OS, as shown in Fig. 4. Unfortunately, while it may be easy to implement this in the case of calls over VoIP, but many mobile OS's prevent the interception between the sound devices and the built-in GSM calling app. A work around could be implemented by running the encryption process on an external device as shown in Fig. 5. In either case, the implemented algorithm must preserve the audio properties to make it independent from the transmission system, else modulation/encoding and demodulation/decoding phases of the transmission system will destroy the original signal and decrypting it will not be able to restore it.

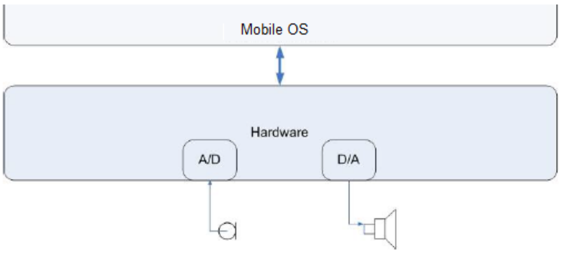


Fig. 4 Sound Hardware in Mobile Phones

Fig. 5 External Encrypting/Decrypting device

IV. Proposed Solution

The paper proposes to implement an encryption/decryption algorithm for voice calls over mobile devices. As explained in the previous section, this implementation must be performed outside the calling phone in order to cover both call types; those over GSM and those that are over VoIP. By taking an advantage of the availability and affordability of smartphones, we suggest running such a process on a secondary smartphone. The secondary smartphone will be connected through the mic/speaker analogue lines of the primary phone that is used to make the call. This mobile application allows the user to speak freely without the need to worry if others are spying on the call or not.

The proposed encryption algorithm is based on Frequency Scrambling technique that is suggested by [7] and implemented on PC's by [8]. The resulting scrambled signal is still an audio. Thus, this makes the technique independent from the transmission system.

The frequency scrambling technique used is to scramble sub-bands of the original frequency itself without adding any extra signals or multiplexing the frequencies on to other signals. By scrambling the frequencies of an audio or a speech signal will result in a signal that sounds different from the original signal. First, the original signal is timely split into buffer size chunks. Each buffer full chunk will be cut into sub-bands by filtrating them into high-pass band (with filter H(z)) and/ or a low-pass band (with filter G(z)), see Fig. 6.

Once the signal is decomposed by the frequency filters they get down-sampled before being scrambled. The down-sampling process can be easily performed by taking every second sample out of the signal, which is not a problem in digital signal processing.

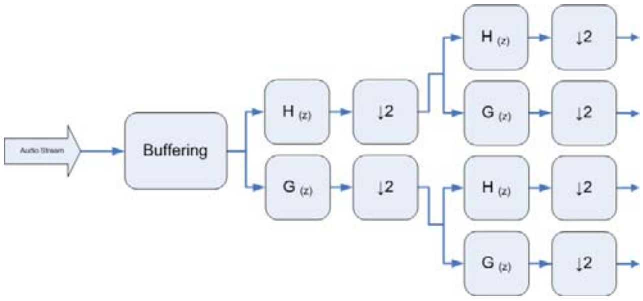


Fig. 6 Encryption System

Finally, the scrambling process is the actual process of making the signal real different from the original signal. There are many possibilities for the scrambling result, but not all of the possible combinations make a good valid code. Otherwise the scrambled spectrum is too similar to the original one and can be understood easily. As a result the sub-bands are shifted to the desired rebuilding path. Fig. 7 shows one possible scrambling option, where the first block goes to third and third goes to second and so on.

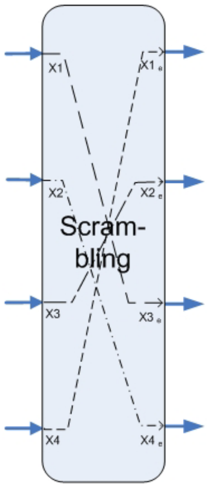


Fig. 7 Scrambling Phase

Rebuilding the signal follows similar operations as in the frequency separation but for different signals since they have been scrambled before. Of course, the signal will not be sampled down again, but up. Low pass filter will be used to reconstruct the original signal and to delete the image frequencies out of the spectrum, thus not worrying about the upper image frequencies.

V. Implementation and Testing

We implement the above technique using Java Language in Android Studio. We developed two modules, one for encryption and another for decryption. For encryption we get the signal from the MIC directly by setting the set Audio Source of the Media Recorder object to Audio Source.MIC and setting the set Audio Encoder to AMR_NB. The other module is for decryption using the MediPlayer class.

Since we experienced some difficulty to release and allocate the mic and speaker devices from the calling phone during a GSM call, we needed secondary phones. To implement a full duplex communication system, we needed two secondary phones. Thus, the small box shown in Fig. 5 is mainly composed of two secondary phones (S1 and S2) as shown in Fig. 8.

We set the number of sub-bands equal to 4 and for the filters, we chose the following:

IIR Low-pass filters:

- designed as Chebyshev II
- Fs=10 kHz; Fstop= 2.62 kHz
- - 80 dB stopband ripple
- Order 20

IIR High-pass filters:

- designed as Chebyshev II
- Fs=10 kHz; Fstop= 2.37 kHz
- - 80 dB stopband ripple
- Order 20

Testing the application was split into three separate tests. Test one starts with a call between two parties. One party will use both modules to speak through a secondary phone (S1) and listen through (S2) connected via the headphone and mic jacks with the primary (as in Fig. 8), while we use no modules on the other party. In this test, neither party understood each other.

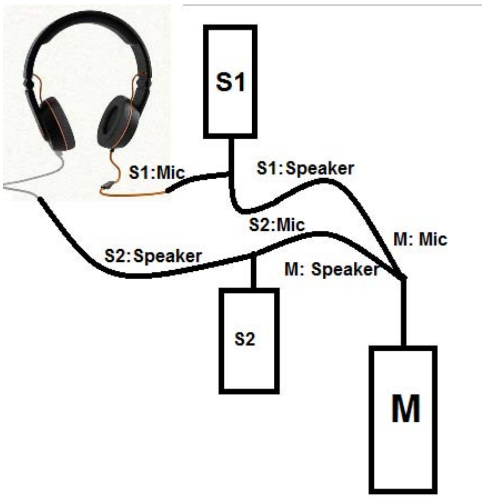


Fig. 8 Proposed System Connection

For test 2, we let the other party run the application for decryption (using S2). He was able to understand what is being spoken by the first party but with some noise of triggering sound.

The final test included a full duplex test in which each party will run the application that includes encrypting the voice going out to the second part as well as decrypting the voice being received by the same party at the same time. The connection was similar to that in Fig. 8 where two secondary smartphones at each end were used. The use of a secondary phone is based on the difficulty to release and allocate the mic and speaker devices from the main phone via GSM application to the encryption/decryption mobile application. The call was made. The quality was fair and the delay was below 90 ns, i.e. barely noticeable.

VI. Conclusion

The need for a simple and portable but independent voice encrypting application became critical with the distrust that people developed towards internet companies, phone network operators and on top of all the government. The paper adopted a light weight independent and simple technique based of frequency scrambling technique to implement on smartphones and use to secure mobile phone calls. The implementation was done on Android phones and the test gave fair quality and insignificant delays.

References

[1] http://www.bbc.com/news/world-us-canada-24647268 visited May 10, 2015
[2] http://rt.com/news/160988-wikileaks-nsa-phone-afghanistan/ visited May 10, 2015
[3] http://www.bloomberg.com/news/2014-01-23/nsa-s-spying-on-phone-calls-illegal-u-s-privacy-board.html visited May 10, 2015
[4] http://www.nsa.gov/publications/publi00019.cfm#N4 visited May 10, 2015
[5] http://www.propublica.org/documents/item/785571-itlbul2013-09-supplemental visited May 10, 2015
[6] http://www.hypermedia-usa.com/solutions/voip_enterprise_solutions.htm visited May 10, 2015
[7] Pere Salvadó Lloveras; “Encriptació de veu per mescla de subbandes” Escola Universitària d'Enginyeria Tècnica Industrial de Terrassa, UPC gen-2006 http://hdl.handle.net/2099.1/3861
[8] Implementation of a real-time voice encryption system, a Mater Thesis of Markus Albert Brandau at Universitat Politècnica de Catalunya, 2008

Open Science Index, Computer and Information Engineering Vol:9, No:7, 2015 publications.waset.org/10001839.pdf