PairPhone:
the effort for P2P private talk
over GSM-FR compressed voice channel

Van Gegel*

About. PairPhone is the crossplatform testing software provides p2p speech encryption duplexes over GSM compressed voice call. The modern cryptography with 128-bit protection level is use Triple Diffie-Hellman initial key exchange with elliptic curve 25519 and SHA3 Keccak 1600/576 Sponge Duplexing based cipher, hash, MAC, SPRNG. Each call's setup provide PSF for encryption and IDs protecting, UKS and KCI resistance. Primary authentication using shared password and comparing of secret's fingerprint, hidden notification of enforcement are available. Devices can be paired during first guest call, certificate for each contact will be added to the address book.

Speech compresses using efficient and robust MELPE codec allowing 1200 bps data rate with BER up to few percents. Streaming encryption use CTR mode for 67.5 mS speech frames and resistant to the error spreading. Voice active detector uses speech pauses for signalling to restore the loss of frame synchronization after communication was temporary failure.

The goal of the project is specially designed pseudo speech modem with small distortion of baseband signal passing over tandem of GSM FR codecs uses in real GSM voice communication. Modem achieves 1200 bps data rate with less then 1% bit error rate and quick on-the-fly synchronizing after less than 200 ms of streaming.

Keywords: mobile communications, vocoders, p2p secure communications.

1 General

GSM voice communications are the widely used in whole world. Unfortunately GSM network not provides p2p security and has significant drawback of authentication allowed call interception. Secure voice conversation over GSM requires additional point-to-point encrypting and assumes the using of appropriate equipment on both sides.

* gegel@umsa.edu.ua, Ukraine

1.1 Data transferring

The applying of strong p2p security requires data communication between participants instead voice because scramblers [1–3] do not provides security level sufficient for the modern requirements. GPRS is most popular way for data transferring via GSM network but have a low priority (unstable access) and high latency (over 800 mS) uncomfortable for implement the duplex voice. Existing solutions GSMK Cryptophone series [4] and Enigma E2 [5] uses the data calls (CSD technique) allowing the transmission of encrypted data at 9600 bps. Unfortunately most of providers are not supported CSD by default requiring the activation of services, special tariff plan and even a dedicated number for these purposes. This profiles the users of such systems initiating a more thorough collection and analysis of their metadata.

Our solution [6] is an attempt to use a general voice connection over a cellular phone for transmitting encrypted data. The main problem is the compression of GSM voice channel is strictly optimized for human speech but not for the signals of conventional modems. For relatively efficient data transmission over voice cellular connection needs a pseudo voice modem with baseband can pass the GSM compressed channel with minimal distortion. A similar project JackPair [7] had serious challenges in the implementation of such a modem and still under development. It seems that the great goal and both defeat of JackPair is an attempt to universality (trying use any communicators with headset analog audio interface). Unfortunately operators usually sets low-bitrate AMR codec is the default and activate some additional speech processing (noise and echo suppression, sound loop, etc.), these functions can not be adjusted in most common phones. We sacrificed the universality and chose a distinct solution in the phone form-factor based on GSM module M66 manufacturers by Quectel [8], intended for telemetry. This module typically allow flexible configuration via the engineering menu and can force operators to use GSM-FR codec (this is the only one codec that the operator is must maintains in any cell). Thus it is possible to design a modem in accordance with the algorithm of the certain codec having significantly increases the quality of the data connection. Also the module has the digital audio bus synchronize the internal codecs with modem implemented in the external processor. The proposed software is a only prototype and still used analog audio interface for communication with the module (for example, mounted on the debugging board from the manufacturer), can be useful for testing and debugging. The pure C code [9] in low level have embedded style and ready for embedding into suitable hardware (e.g. SM1000 HAM Radio adaper [10] based on Cortex F4 etc.).

1.2 Security

PairPhone provides a strong point-to-point block-streaming encryption (like CTR mode) using Keccak 1600/576 Sponge Duplexing [11], Perfect Forward Secrecy (PFS) for each session (implicitly authenticated initial key exchange with Triple Diffie-Hellmann (TripleDH) key exchange [12] with Elliptic Curve 25519 (EC25519) [13]), strong authentication with resistance to Unknown Key Share

(UKS), week Perfect Forward Secrecy (wPFS) and Public Key Impersonation (PKI) attacks. Long-term keys adds to address books during the individual pairing of devices, primary authentication of the first guest call before pairing provides by comparing the session ID (for example, in direct contact of subscribers) or via shared password (for example, using PGP in deniable way). Hidden notification of forced collaboration using shared passphrase with a zero-knowledge protocol are available. Achieves the protection of identity against the passive and active attackers and PFS for this protection for calls acceptor (queries resistance).

2 Get started

PairPhone is completely open source, making with gcc for `*nix` (tested with `Ubuntu` and `Debian`) and mingw for `Windows` (tested on `Win98` up to `Win8`), do not use external libraries except libasound2-dev (Linux alsa audio) and winmm, comctl32 (Windows wave audio).

2.1 Installation

PairPhone is completely portable and can be run from removable disk or a TrueCrypt container and does not write to disk anywhere excepts its working folder.

2.2 Requirements

Since MELPE codec requires 40 MIPS itself the CPU must doing at least 50 MIPS total. Linux/Windows PC based on Intel Celeron 1000MHz 512RAM suitable for software PairPhone as well and must have two analog audio interfaces one of which is connected to the headset and operates on 8 kHz mono and other - to a line of GSM link and operates on 48 kHz mono. External USB audio device or Bluetooth audio interface connected to the BT headset can be used.

2.3 Configuration

Run the program default for obtaining a list of installed audio devices then configure both interfaces in the 'conf.txt' file, for example:

```
#headset
AudioInput=plughw:1,0
AudioOutput=plughw:1,0
#line
_AudioInput=plughw:2,0
_AudioOutput=plughw:2,0
```

3 User guide

To get started just run the executable file from the working folder. PairPhone is running in idle mode and ready to receive incoming connection automatically. To perform certain actions input optional string parameter and press **F1-F10** for applying such functionality:

	F1	show help screen
<name>	**F2**	pair devices during active call (name must be specified)
{mask}	**F3**	search contacts in adressbook (mask can be specified)
{name}	**F4**	originate outgoing call (name can be specified, otherwise 'guest')
	F5	repeat the last outgoing call
{pin}	**F6**	apply pin code access to the book (not supported yet)
{pass}{pass}	**F7**	apply pre-shared passphrase (1 or 2 words or empty defaults)
	F8	terminate active call to idle state (hang up)
	F9	reconnect active call (agreed new keys)
	F10	exit

Other keys used for controlling:

Tab	talk/mute switch
Del	clean inputted string
Back	delete last inputted char

3.1 Talk

After reporting a successful connection is established participants must press **Tab** to activate the voice output. While there are no entered characters the call duration, percentage levels of bit error rate and authentication will be dynamically displayed. Note that authentication level updates only in speech pauses and will be freezing during active conversation. Mute mike to be sure that obtained value is actual.

3.2 Initial authentication

The first call to the new subscriber is performed as a 'guest' call and initiated pressing **F4**. Once the PairPhone is in a 'guest' mode make sure the connection is secure (no MitM occurred). If subscribers are near (pair your device with the personal touch) they can compare the session codes (secrets fingerprint) that are displayed after the connection established. If the pairing is performed remotely subscribers must pre-share one-time password, for example using PGP. To achieve PFS-protected deniability parties can manually implement the Abadi protocol [14]: the initiator pick the first half of password and sends in PGP message entering his name (PGP fingerprint), all without

signing. Acceptor read message, pick the second half of password and answers back the complete, also without signing. Once connected via PairPhone in a 'guest' mode participants submitted to each other, applied shared password and look for authentication level updated in a speech pauses. Reaching the threshold value of 80-90% the participants will be sure the connection is secure and can be pairs their devices.

3.3 Pairing process

Once guest connection was established and verified both participants will choose any unique names for this contact and press **F2**. Long-term key pairs will be generated and added to the 'contacts.txt' address book files for current contact. Later to originate the mutually authenticated call a subscriber enter the specified name and press **F4**. Identification and authentication will be performs automatically and IDs will be protected with PFS.

3.4 Hidden notification on forced collaboration

Parties can ensure there are no forced collaboration on other side at the current call. For such possibility the parties should pre-share a passphrase composed two words, swaps them on the one side and keep in mind for this contact. Once connected the phrase is applied by pressing **F7** and follow up the level of authentication that is updated in the speech pauses. In the case of forced collaboration the participant changes the second word on any other suitable meaning. Wherein the authentication result on this side will appear as normal but on the other side the authentication value will be low indicating failure. The attacker has no way to ensure validity of phrase before and after using despite any previous passive and active queries.

4 Modem design

The solution for transferring data over any voice channel is the acoustic modem. The main problem is GSM compression developed for human speech and significantly distorts the carrier [15]. There are three ways for develop modems useable for the GSM compressed channels. The first is using of standard carriers with PSK, FSK, QAM or FDMDV adapted for the GSM channel [16–22]. This way is have simplest complexity but usually not achieves required bitrate with acceptable error rate. Second way is using synthetic pseudo speech symbols (finite alphabet) [23–25] and achieves acceptable bitrate even with GSM HR and AMR HR codecs but has a relatively high complexity of demodulator. The third way is pulse coding in accordance with the channel codec engine [26–28]. This way is seems best for secure streaming data over voice link compressed by known GSM codec.

Our research has shown that the compressed channel, not an AWGN channel, is characterized by a significant increase of error rate causing data rate increase above 1200 bps. So data redundancy with any overhead seems unefficient despite any coding schemes. The main cause of error is a phase jitter due asynchronous ACELP codebooks of the coders tandem. The small difference of sampling

rates (fewer hertz) leads to a slow changes of the channel response and periodically produces fading causing a significant increase the BER of a few seconds duration. Close to real-time requirements for voice assumes the impossibility of any efficient correction of such periodic regressions using redundant coding schemes. In contrast, the use of antipodal modulation (BPSK) with the minimal necessary bit rate is more preferably in comparison with the any M-ary schemes and significantly stabilize the channel response.

4.1 Proposed solution

Modem described here [29] achieves bitrate 1200 bps with bit error rate less then 0.1% passed over model of GSM EFR and GSM FR compressed channel, fast self-synchronization after less then 200 mS of streaming, no training sequences and low complexity of modulator and demodulator. We combines pulse coding technique in time domain with special waveform style in frequency domain view and actually use BPSK of a 1333Hz carrier (exactly six 8KHz samples per period coded one bit) with coherent demodulation, frame synchronization and simplest FEC. To select the carrier parameters we considered the ACELP codebook search mechanism of GSM EFR codec [30], AMR family [31] and RPE engine of the GSM FR codec [32]. Each period of carrier (6 samples) have maximum energy in 2nd and 5th samples ensure accurate GSM transcoding and good phase locking. This allowed to achieve the lowest BER at a given bitrate for GSM FR coded channel in comparison with known previous research.

Modulator uses wave tables and produce 1333 Hz BPSK modulated carrier with some emphasis empirically optimized for tandems of GSM FR codecs. Thus, the second halftime of waveform has twice the amplitude. Moreover, applies elementary controlled inter-symbol interference: once bit changed (phase jumped to 180 degree) this waveforms period is distorted from the sinusoidal (shaper filter used). This improves the baseband signal to pass through tandem of codecs in the real GSM connection and lowers resulting bit error rate. Also for prevention the rejection by the in-builded into GSM equipment voice active detector uses a trick with periodical changes the signal level (decreasing by half over 67.5 ms time).

The demodulator is coherent type with adaptive equalizing, compensation of the controlled inter-symbol interference (ISI) and dynamical adjusting under the current channel response. The updating time choose in accordance with the characteristics of GSM FR coder.

4.2 Synchronization

Synchronization is implemented as a phase locking typically for BPSK, frequency adjusting in a way of skipping/doubling some baseband samples and frame alignment. Sync inset is functionally combined with the sequence of parity bits are also used to FEC. The modem frame 67.5 mS contains 9 subframes: 9 bits of payload and parity bit each. The stream is first the nine LSBs of all subframes, nine next bits, up to nine MSBs. Nine parity bits transmits as a tail on the end

of the frame and uses for obtaining the frame boundary due their predictability. The goal of our modem is the relatively fast synchronization (less then 200 mS stream can be lost after starting) with minimal data overhead required to reduce overall bitrate causing a decrease of BER.

Strongly synchronized non-rewinding counter of frames is ready for cryptographic purposes. The 16 bits size allows to pass the appropriate number of frames in a single session and specifies the maximum duration of the call above 1 hour. After this time the connection will automatically reinstalled with the new key exchange and counter reset.

Loss of synchronization is possible only with long fading depending the sampling rate difference between modulators and demodulators audio hardware. The recover is possible in speech pauses while the transmitter sends packets with the actual value of its counter. So the counter is strongly one-way the back synchronization is possible only by freezing the receivers counter. The quick synchronization allows to adjust the counter by only 1 to back (freeze) or 255 to forward after just one good sync packet received. Forward correction to the arbitrary greater value requires few sync packets sequence even with bit errors to avoid a fatal failure due a significant BER.

4.3 Errors correction

Voice channel requires a lowest delay so that is impossible to apply complicated methods of error correction. Fortunately MELPE codec [33] relatively tolerant up to few percent of BER due to in-built FEC protection the most important data in the speech descriptor. This codec operates on a 1200 bps rate and uses 67.5mS frames contains 81 bits each. This is corresponds modem frames 90 bits each. 81 bits payload with 9 parity bits provide r=9/10 overhead suitable only for simplest transporting real-time FEC. Supposedly one of 9 bits can be wrong in the subframe with incorrect parity. Probably this is a bit with lowest likelihood as a soft output of the correlator, and it can be corrected. This will be effective in the small initial BER (up to 2%).

4.4 Controlling

Besides transmitting speech descriptors the controlling blocks are used during silence for the maintain counter synchronization and on the initial key exchange stage. These packets are important for the sessions integrity therefore use more powerful error correction by reducing the payload. The 81 bits of block are split into logical fields: 32 bits payload, 4 bits label, 8 bits tag and 1 bit as a flag. Payload and label (36 bits total) protected with three Golay24/12 codes produced 72 raw bits. The payload field is whitening using a pseudo-random mask derives from the label for some improving the stream statistic with a relatively constant bits in frames (for example, MSBs of the counter). Flag bit is always set to zero and allows to discover inversion of the physical link for the correct further demodulation.

The payload field of sync packet contains two copies of counter value 16 bits each. The label matches 4 LSB of the counter. Tag uses for password authentication. Because tag is not protected

by FEC a few bit errors are possible. A percentage of bit matches during whole session ensures the reliably of the decision even with significant BER.

The packets for initial key exchange contains 32 bits data in the payload field as a part of key or authenticator. The label specifies the data type and the tag contains checksum of the data in this packet. Packets necessary at this protocol stage are repeated in loop. The number of checksum bits matching is define the weight of received data bits. Accepted soft bits are accumulated each replay. The integrity of whole data sequence (public key, authenticator) is checked by CRC32 which is part of them. This allows to speed up the call setup even under high BER.

4.5 Complexity and testing

The computational complexity of demodulator is less than 0.5 MIPS on Floating Points platforms. Modem can be easily embedded into GSM module as a service controlling by AT commands. Since sampling rates difference between some PC audio hardware can be up to 2%, the 48Khz sampling rate uses for a more stable synchronization. If the digital audio bus synchronized with GSM coders is used, sampling rate will be just a 8kHz with 5 times less computational complexity.

Modem tested on framework included 3 PC running modulator, transcoder and demodulator. Analog audio interface was used. BER over one pass GSM FR transcoder is less then 0.1%.

Another test was performed using GSM link over two Quectel M66 evaluations boards, configured as follow:

AT+QAUDCH=1 //use headset audio channel

AT+QSIDET=0 //close side tone

AT+QSFR=1 //set priority for Full Rate encoding

AT+CAGC=0,0 //disable analog auto gain control and echo algorithm

AT+QAPS=1,4,1,"0.224.32512.31.57351.24607.400.132.80.4325.99.0.32513.0.0.0"

The BER was 0.2-0.9% in the network of mobile operators MNC+MCC 25501, 25503, 25506 as well as between this operators. Note that in some cases codec tandem can occasionally include extra transcoding with AMR NB codec causing much worse results. This is due engineering settings of inter-cell link and we can not affect it other than try to recall.

5 Cryptography design

- Asymmetric cryptography is Diffie - Hellmann on EC25519 used curve25519_donna C-code [34] ;
- Symmetric cryptography is stream encryption used Keccak Sponge Dulpexing library [35] ;
- SPRNG is Keccak Sponge Duplexing RNG [36] with Havege reseeding.

- Hashing, MAC, KDF are Keccak in 576/1600 mode [37] ;

We don't use special MAC and KDF functions i.e. Keccak is RO-PRF and can be directly used for this purpose.

5.1 Operation Mode

Stream cipher based on Keccak is used. The block mode is CTR: both parties maintain synchronized counters separately for sent and received packets. Modem provides synchronization of counters. The symmetric keys are different for both directions (encrypting and decryption) and derives from shared secret. For each packet the Keccak Sponge is initiates and absorbs 128-bits session symmetric key k, 16-bits counter ctr, then squeezes gamma $g = H(k \parallel ctr)$. Gamma XOR-ed with plaintext p (all packets bits) provides ciphertext: $c = g \oplus p$. After this counter is incremented. There is no way to rewind the counter. Decryption is similar to encryption using the appropriate key. This mode prevents the spread of bit errors (one error in an encrypted message will be produce only one error in decrypted).

Message integrity is not checked explicitly since the bit errors happens all the time and must not block the processing the data are mostly still correct. An attacker can arbitrarily change any bit messages but can not obtain its value. Keccak gamma has a PRF property those ciphertext statistics will be random. Any bit changing of the ciphertext cause random bit in the plaintext. It is believed that when the phone the speech itself is the authenticator sufficient in this case.

During the speech pauses obtained by voice active detector no need to constantly transmit voice descriptors, and it is possible to perform explicit authentication using shared password. Authenticator is $m = H(a \parallel ctr \parallel k)$, where ctr is actual counter value, k is corresponding session symmetric key and a is 128 bits key PKDF-derived from password: $a = H(salt \parallel pass \parallel \cdots \parallel pass)$. Only 8 bits of the authenticator transmitted each control packet (67.5 mS duration) but over a time the total number of authenticating bits is enough for secure level even in some bit errors.

5.2 Initial key exchange

The core of implemented Initial Key Exchange (IKE) protocol is known as the TripleDH [38] (modified KEA+ protocol [39]) and provides key agreement (produces fresh session key for each run achieving Prefect Forward Secrecy) with implicit authentication using long-term ECDH public keys stored in adressbook while devices was paired. This IKE looks resistant to UKS , wPFS and KCI attacks.

Let G_1 is additive group of order q on an elliptic curve 25519. Public key $T = g^t$, where $t \in Z_q$ is the private key and $g = 9$ is the base point of G_1. H is a one-way hash function with RO-PRF property (SHA3 Keccak Sponge). Alice and Bob have long-term keys individual for this contact. Here, B, a are Bob's public and Alice's private keys in Alice's adressbook; A, b are corresponding Alice's public and Bob's private keys in Bob's adressbook.

When Alice wants to agreed with Bob, they cooperatively perform the following steps. Participants randomly pick ephemeral secret keys x and y respectively. Beginning they computes and anonymously exchanges corresponding ephemeral public keys X and Y.

At the next step participants agreed common secret $S = Y^x = X^y$, computes and exchanges theirs hidden identificators: $ID_A = A \oplus \lfloor H(S)$ and $ID_B = B \oplus \lceil H(S)$. For this originator (Alice) send his identificator first, acceptor unmask it: $A = ID_A \oplus \lfloor H(S)$ and search match in his adressbook, obtaining identity of the originator and compute own identificator corresponds its contact . Originator checks answer same way and ensure identity of acceptor. Comparing X and Y participants obtain roles and derive encryption and decryption symmetric key depending on them: $k_e \parallel k_d = H(B^x \parallel Y^a \parallel S)$ and $k_d \parallel k_e = H(X^b \parallel A^y \parallel S)$.

5.3 Explicit authentication

While the first call before pairing participants use default 'guest' long-term keys in the TripleDH IKE process. So these keys are hardcoded and published they not provide protection against intercept attack (MitM) thereby IKE degenerates into an ordinary Diffie-Hellman. To check the security of this connection it is necessary to perform explicit authentication using an pre-shared password or directly compare the fingerprints of agreed secret. Four digits of fingerprints as a short authentication string (SAS) displayed after the connection established. Normally, using the short authenticator (only 13 bits in 4 digits of SAS) require commitment of the ephemeral public keys before exchange. Quick call setup at a high BER and low data rate prefers other way for preventing the influence of SAS on MitM: the exchange of 256 bit ECDH ephemeral public keys doing into two steps: the parties exchange the first 224 bits and then remaining 32 bits. Thus, the first stage provides commitment and the second stage outputs 256-bit shared secret and its SAS.

5.4 Long-term keypair

Since PairPhone provide p2p security is meant pairing two devices each other to perform further hidden identification and cryptographic authentication without any additional actions of subscribers. Note the pairing should be performed only during the secure 'guest' session after applying the common password or checking fingerprint. During pairing the independent public and private key will be created and added to the adressbook, and on the other side - by appropriate them as a pair. Comparing session ephemeral public keys X and Y parties obtain roles and derives long term secrets $a \parallel b = H(k_e \parallel k_d)$, where k_e and k_d are encryption and decryption session symmetric keys. Using this keys contacts certificates be computes and adds to adressbooks of participants: A, b and B, a respectively. Each adressbook entry marked by individual name and can be appended by additional information about this contact (physical phone number etc.). The names are identities for this pair and will be used later for originate the call and notify of incoming.

References

[1] Secure voice during conversations over mobile network and in Skype. Mobile Trust Telecommunications manual. (Full text)

[2] Yohan Suryanto, Kalamullah Ramli. Implementation and Performance Analysis of Reliable and Secure End to End Voice Encryption over Public Mobile Network Based on Frequency Domain Using Dual Processor in FPGA Platform. Int.J.Computer Technology and Applications,Vol 5 (1),2014,103-111. (Full text)

[3] S. Islam, F. Ajmal, S. Ali, J. Zahid Secure end-to-end communication over GSM and PSTN networks Electro/Information Technology, 2009. eit '09. IEEE International Conference, 323 - 326. (Full text)

[4] GSMK CryptoPhone: Trustworthy Voice and Message Encryption. (Web link)

[5] E2 the New Enigma Secure Phone. (Web link)

[6] Van Gegel. PairPhone: the effort for P2P private talk over GSM-FR compressed voice channel. (Web link)

[7] Jeffrey Chang (2015). JackPair: secure your voice phone calls against wiretapping. (Web link)

[8] Quectel Wireless Solutions - Dedicated M2M Wireless Module Supplier. (Web link)

[9] Van Gegel (2016) PairPhone: the software testing tool for P2P speech encryption over GSM-FR compressed voice channel (Web link)

[10] David Rowe. SM1000 FreeDV Adaptor for the Digital Ham Radio. (Web link)

[11] Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche. Duplexing the sponge: single-pass authenticated encryption and other applications. Selected Areas in Cryptography. 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers, 2011; 320:337. (Full text)

[12] Trevor Perrin, Moxie Marlinspike (2013). Simplifying OTR deniability. (Web link)

[13] Daniel J. Bernstein Curve25519: new Diffie-Hellman speed records. In Public Key Cryptography (PKC), Springer-Verlag LNCS 3958, 2006. (Full text)

[14] Martin Abadi, Cedric Fournet. Private Authentication In Software Security Theories and Systems. Mext-NSF-JSPS International Symposium (ISSS02); 2003; 317:338. (Full text)

[15] R. Kazemi ,R. Mosayebi, S. M. Etemadi, M. Boloursaz, F. Behnia. A Lower Capacity Bound of Secure End to End Data Transmission via GSM Network. Telecommunications (IST), 2012 Sixth International Symposium, Tehran, 6-8 Nov. 2012. (Full text)

[16] STANAG 4285. Characteristics of 1200/2400/3600 bits per second single modulators/demodulators for HF radio lynks, 1989-1993 (Full text)

[17] Gianluigi Biancucci, Andrea Claudi, Aldo Franco Dragoni. Secure Data and Voice Transmission Over GSM Voice Channel: Applications for Secure Communications. 4th International Conference on Intelligent Systems, Modelling and Simulation, 2013, 230-233. (Full text)

[18] T.Chmayssani, G.Baudoin. Data transmission over voice dedicated channels using digital modulations. Radioelektronika, 2008 18th International Conference, 24-25 April 2008, P.1-4. (Full text)

[19] S.Ciornei, I.Bogdan, L.Scripcariu, M.Popa. Sensitivity analysis of Quadrature Amplitude Modulation over AMR-WB voice codec. Second International Conference on Advances in Computing, Communication and Information Technology CCIT-2014, University of Birmingham, UK 16 - 17 November, 2014,P.13-18. (Full text)

[20] Zdenko Mezgec, Amor Chowdhury, Bojan Kotnik, and Rajko Svecko. Implementation of PCCD-OFDM-ASK Robust Data Transmission over GSM Speech Channel. Informatica 20, 1 (January 2009), 51-78. (Full text)

[21] Aditya Dhananjay, Ashlesh Sharma, Michael Paik, Jay Chen, Trishank Karthik Kuppusamy, Jinyang Li, Lakshminarayanan Subramanian. Hermes: data transmission over unknown voice channels. Conference: Mobile Computing and Networking - MOBICOM , pp. 113-124, 2010. (Full text)

[22] Sigurdur Sverrisson, Xiaoyun Liang. Digital Communication over Speech Compressed Channel. Degree thesis, Chalmers University of Technology, Goteborg, Sweden EXE028/2008. (Full text)

[23] Mahsa Rashidi, Abolghasem Sayadiyan. A New Approach for Digital Data Transmission over GSM Voice Channel. 2nd WSEAS Int. Conf. CISST'08, Acapulco, Mexico, January 25-27, 2008 p.193-196 (Full text)

[24] Vitaliy V. Sapozhnykov, Kurt S.Fienberg. A Low-rate Data Transfer Technique for Compressed Voice Channels. J Sign. Process Syst (2012) 68:151170 (Full text)

[25] M.Boloursaz, A.H.Hadavi, R.Kazemi, F.Behnia. A data modem for GSM Adaptive Multi Rate voice channel. East-West Design and Test Symposium, 2013. (Full text)

[26] N. N. Katugampala , K. T. Al-naimi , S. Villette , A. M. Kondoz. Real-time end-to-end secure voice communications over GSM voice channel. 13th European Signal Processing Converence, EUSIPCO 05, 2005. (Full text)

[27] Andreas Tyrberg. Data Transmission over Speech Coded Voice Channels. Degree thesis, Department of Electrical Engineering Linkopings universitet. Linkoping, Sweden, 12 June, 2006. (Full text)

[28] 3GPP TS 26.268 eCall data transfer; In-band modem solution; ANSI-C reference code, 2008-2014 (Download link)

[29] Van Gegel. A low comlexity data modem for GSM FR and GSM EFR compressed voice channel. C Source code and testing software. (Download link)

[30] ETSI EN 300.726 Enhanced Full Rate (EFR) speech transcoding (GSM 06.60), 2000. (Full text)

[31] ETSI TS 126.090. Adaptive Multi-Rate (AMR) speech codec; Transcoding functions, 2011 (Full text)

[32] ETS 300 961. Full rate speech; Transcoding (GSM 06.10), 1998 (Full text)

[33] John S.Collura, Diane F.Brandt, Douglas J.Rahikka. The 1.2Kbps/2.4Kbps MELP Speech Coding Suite with Integrated Noise Pre-Processing. Military Communications Conference Proceedings, 1999. MILCOM 1999. IEEE, P.1449-1453. (Full text)

[34] Adam Langley (2011) Implementations of a fast Elliptic-curve Diffie-Hellman primitive. (Full text)

[35] Van Gegel (2013) Portable C implementation of universal Sponge construction based on compact Keccak source code. (Web link)

[36] Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche. Sponge-based pseudo-random number generators. Cryptographic Hardware and Embedded Systems, CHES 2010. 12th International Workshop, Santa Barbara, USA, August 17-20, 2010; 33:47. (Full text)

[37] Guido Bertoni, Joan Daemen, Michael Peeters, Gilles Van Assche. The Keccak reference. Version 3.0.; January 14, 2011. (Full text)

[38] Andrikopoulos Konstandinos. A triple dh implementation in C. (Web link)

[39] Kristin Lauter, Anton Mityagin. Security Analysis of KEA Authenticated Key Exchange Protocol. PKC 2006, volume 3958 of LNCS; 2006; 378:394. (Full text)