cryptography

MDPI

Technical Note

On the Cryptographic Features of a VoIP Service

Dimitrios Alvanos, Konstantinos Limniotis * and Stavros Stavrou

School of Pure & Applied Sciences, Open University of Cyprus, Latsia 2220, Cyprus;
dimitrios.alvanos@st.ouc.ac.cy (D.A.); stavros.stavrou@ouc.ac.cy (S.S.)
* Correspondence: konstantinos.limniotis@ouc.ac.cy

Received: 27 November; Accepted: 18 January 2018; Published: 19 January 2018

Abstract: Security issues of typical Voice over Internet Protocol (VoIP) applications are studied in this paper; in particular, the open source Linphone application is being used as a case study. An experimental analysis indicates that protecting signalling data with the TLS protocol, which unfortunately is not always the default option, is needed to alleviate several security concerns. Moreover, towards improving security, it is shown that a VoIP application may operate over a virtual private network without significantly degrading the overall performance. The conclusions of this study provide useful insights to the usage of any VoIP application.

Keywords: Linphone; SIP; TLS; VoIP; ZRTP

1. Introduction

The Voice over Internet Protocol (VoIP) constitutes one of the most important technologies in telecommunications, due to the fact that its quality and reliability are constantly improving. Furthermore, the cost for the users is negligible compared to traditional fixed or mobile telephony [1]. VoIP can be used to make phone (or video) calls between any two terminals, which can be personal computers, VoIP phones, mobile phones or even two traditional phones (provided that there is an underlying IP network). Therefore, VoIP is being used not only for domestic purposes but also in businesses.

The security of VoIP has been extensively studied over the last years, whereas well-determined security protocols and mechanisms are being used [1]. In general, VoIP security issues are very similar to other cyber security issues, like call hijacking, ID spoofing, Denial of Service, as well as phishing and malware threats (see, e.g., [2]). To address these risks, security mechanisms such as encryption, entity authentication and deep packet inspection are widely used (see, e.g., [3–5]), whilst building a network with VoIP traffic in a Virtual Private Network has been also considered [6]. Several attacks on VoIP technologies have been mounted, based on the absence of security protocols, or weaknesses of the protocols, or of their implementations (see, e.g., [7–15]).

In this paper, we focus on a free open-source VoIP platform, namely Linphone [16], which utilizes SIP, TLS, SRTP and ZRTP. An experimental environment has been set up, consisting of 25 users with several types of devices, in order to study possible security issues or misconfigurations that arise during communication. The main finding from our preliminary analysis rests with the fact that the TLS is not being enabled by default to protect signalling data; this in turn gave rise to the following observations: (i) even if a user has enabled TLS, he will receive a direct call—without a proxy—from any other user that he has not enabled TLS, without receiving any relative warning for this degradation of the security service; (ii) the absence of TLS facilitates an external scan of our system from an attacker, which will result in annoying "ghost" calls; (iii) although the transmitted data are encrypted and, thus, unintelligible, the absence of TLS enables the easy differentiation of the transmitted data in terms of the underlying protocol. Therefore, it becomes even more evident that TLS should be a default option in VoIP platforms. In addition, we show that deploying a VoIP service over a Virtual Private Network may be an option to enhance security, since it seems that it does not significantly affect the

overall performance. Finally, we discuss a simple approach to address a specific type of attack on the ZRTP—which is the underlying protocol for exchanging the secret encryption key; all the proposed countermeasures apply not only to Linphone but to any other VoIP application or platform.

It should be explicitly pointed out that the aim of the paper is not to make a comparative study between VoIP services neither to qualify our test case—that is Linphone—in terms of security. Our ultimate goal is to study whether a VoIP service with increased level of security due to incorporation of several cryptographic mechanisms may also yield some risks owing to misconfigurations of the security parameters, as well as to discuss additional safeguards that apply to all cases.

The paper is organized as follows; the main protocols found in VoIP applications—namely SIP, TLS, (S)RTP and ZRTP—are presented in Section 2, whereas a brief description of the architecture of Linphone is given in Section 3. Section 4 constitutes the main part of the paper, presenting the analysis to identify possible security concerns and the main findings of this analysis; we also discuss possible approaches to alleviate raised security issues. Finally, concluding remarks are given in Section 5.

2. VoIP Protocols

To establish a VoIP transmission, there is need to setup the call, as well as to encode and to transfer data between different networks. Call signalling (which refers to the setup, configuration and termination of calls) is performed by a call processing manager (IP PBX). The most commonly used call signalling protocols are H.323, specified by the International Telecommunications Union (ITU) in 1996, and the Session Initiation Protocol (SIP), specified by the Internet Engineering Task Force (IETF) in 1999. A comparison between these two protocols is provided in [17]. Linphone is based on SIP for call signalling.

Once a connection is established between two terminals, the transmission must be initiated. To this end, the Real Time Protocol (RTP) (RFC 3550 [18]) is being used, in order to digitize, compress and packetize the traffic which is to be sent over the network. RTP also provides time stamping, a sequence numbering, payload type identification and delivery monitoring; the sequence numbers allow endpoints to check for lost or out of order packets; this is important, due to the connectionless nature of UDP over which RTP typically runs.

RTP Control Protocol (RTPC) is part of RTP and may be used for quality control. Moreover, when transmitted data are encrypted, we refer to Secure RTP (SRTP), which is discussed in the sequel. Linphone supports SRTP for encrypting the data that are being transmitted.

2.1. The Session Initiation Protocol (SIP)

SIP is an application-layer control protocol that can establish, modify, and terminate multimedia sessions such as Internet telephony calls (RFC 3261 [19]). SIP is a text-based client-server protocol, inspired by both HTTP and SMTP, being able to handle UDP, TCP and SCTP transport layer protocols. In practice, SIP provides only signalling and should be used in conjunction with other protocols in order to provide complete services to the users; however, the basic functionality of SIP does not depend on any of these protocols.

In SIP, every element is identified by a SIP URI (Uniform Resource Identifier). The elements in a SIP network are:

- The User Agent, which is the end-point of the network, being able to initiate, modify, or terminate a session (e.g., a computer). User Agents may have two roles, i.e., the User Agent Client (UAC) and the User Agent Server (UAS); the caller's phone acts as a client and the callee's phone acts as a server.
- The Proxy Server, which gets a request from a user agent (i.e., an INVITE message) and forwards it to another user;
- The Registrar Server, which is responsible for registering users to the network. The Registrar Server accepts registration requests from user agents and helps users to authenticate themselves

within the network. It stores the URI and the location of users in a database to help other SIP servers within the same domain.

- The Redirect Server receives requests and looks up the intended recipient of the request in the location database created by the registrar.
- The Location Server provides information regarding the caller's possible locations to the Redirect and Proxy Servers.

To secure SIP signalling messages from tampering and eavesdropping, the Transport Layer Security (TLS) can be used; this is described next.

2.2. The Transport Layer Security Protocol (TLS)

The Transport Layer Security (TLS) protocol, as a successor of the Secure Sockets Layer (SSL) protocol, is being considered as a somehow de-facto security standard. Three versions of TLS have been standardized, namely 1.0, 1.1 and 1.2, while the new forthcoming standard 1.3 is currently under development. TLS is based on symmetric encryption for ensuring confidentiality of data transmitted, whereas the symmetric key is being interchanged via public key cryptographic algorithms. The TLS is a client-server model, where the client is able to authenticate the server and, optionally, the server is also able to authenticate the client; the authentication is based on digital certificates of the involved entities. Apart from confidentiality and entity authentication, the TLS also provides data integrity via appropriate Message Authentication Code (MAC).

When using SIP over TLS, the whole SIP signalling is encrypted. It should be pointed out though that this holds only on the segments of the communication which actually use TLS. For example, if a client (caller) sends the SIP message with UDP to the proxy and the proxy forwards the SIP message to another client (callee) over TLS, then the UDP communication remains unencrypted.

2.3. The Secure Real Time Protocol (SRTP) and the ZRTP Protocol

The Secure Real-time Transport Protocol (SRTP) defines a profile of RTP (Real-time Transport Protocol), intended to provide encryption and message authentication to the RTP data—namely, SRTP secures conversations by encrypting audio and video media traffic [20]. When using SRTP only the media payload is encrypted; the RTP headers are still sent in plaintext. The encryption of traffic (media) is independent from the encryption of signalling; therefore, all possible combinations (SIP over TLS and RTP, SIP over UDP and SRTP, SIP over TLS and SRTP) can be adopted.

For SRTP, both parties need to securely exchange the secret encryption key. There are several methods for SRTP key exchange. The most used such techniques are SDES (Session Description Protocol Security Descriptions) (RFC 4568), DTLS (Datagram Transport Layer Security) and ZRTP. With SDES, the encryption key is exchanged in the session description and its security rests with the security of the SIP protocol since any SIP proxy has access to the secret key—and, thus, all SIP proxies should be trusted. DTLS [21] resembles the TLS protocol, employing digital certificates for entity authentication in secure key exchange – and thus a Public Key Infrastructure (PKI) is needed. On the other side, ZRTP [22] (where "Z" stands for Phil Zimmerman, its inventor) utilizes a somehow modified Diffie-Hellman key exchange approach without necessitating a Public Key Infrastructure or digital certificates. ZRTP incorporates a mechanism that is based on the so-called Short Authentication String (SAS) to resist against Man-In-The-Middle (MITM) attacks. By these means, when the key has been exchanged, the two communicating parties have also securely interchanged SAS which is obtained from the hash value of the Diffie-Hellman key and, then, they verbally cross-check this value; if the values do not match, a MITM attack is indicated owing to the fact that an attacker would have exchanged different keys with each party and, therefore, the two parties would have computed different Short Authentication Strings. The attacker is not able to compute which would be the common SAS for these two parties in case of his/her absence.

3. The Linphone Application

Linphone [16] is an open-source example VoIP application that utilizes SIP. Linphone, which is provided by Belledonne Communications, is applicable to any mobile and desktop platform (iOS, Android, Windows Phone, BlackBerry, GNU/Linux, Windows Desktop, MAC OSX). The free Linphone SIP service is released with an open-source license; the SIP server software (written in C++) powering this service is called Flexisip.

As described in [23], the combination of Linphone and Flexisip SIP proxy provides secure end-user registration and call setup. More precisely, Linphone client establishes and maintains a SIP TLS connection to the Flexisip server. The Linphone client verifies the SIP server's identity based on the X.509 digital certificate of the server (a list of trusted root authorities is provided at compilation time). In this way, message and entity authentication, as well as confidentiality, of the information exchanged between the Linphone client and the Flexisip server is ensured.

The Flexisip server is also responsible for performing the authentication of the SIP messages coming from clients, using either digest authentication from a password database or TLS client-based authentication: the choice between the two methods is a matter of configuration in Flexisip and Linphone client.

Voice and video over RTP are encrypted using AES with either a 128 or 256 bits key length, according to the specifications of the SRTP. For key exchange, Linphone supports either SDES, DTLS and ZRTP.

4. Experimental Analysis on Security Features of Linphone

To perform our analysis, an experimental environment was set up in a school laboratory in Athens, Greece, consisting of 25 Linphone users, using mainly the version 3.10.2. The corresponding devices were: (i) fifteen (15) Pentium 4 desktops, with Windows XP; (ii) five (5) Dual Core desktops, with Windows 7 home edition; (iii) one (1) Dual Core desktop, with OpenSuse Leap and (iv) four (4) Android smartphones, with Quad-core processor and 1 Gb RAM. The desktops were connected to the same LAN, whose connection to the Internet was at 10 Mbps (download) and 500 kbps (upload) through the Greek School Network, which is the official and exclusive network that securely interconnects all schools for Primary and Secondary Education in Greece; the internal LAN speed was at 100 Mbps. The experiments took place on the first half of December 2016. As it is also described in the subsequent Section 4.4, some VoIP calls initiated from home networks.

4.1. A TLS Issue

During the experiments, any user could directly call any other user of the network without a proxy server, by directly using the destination IP address; this is shown in Figure 1, where the user root@192.168.3.150 calls the user 1000@192.168.3.153. In this way, outgoing calls are permitted without registration.

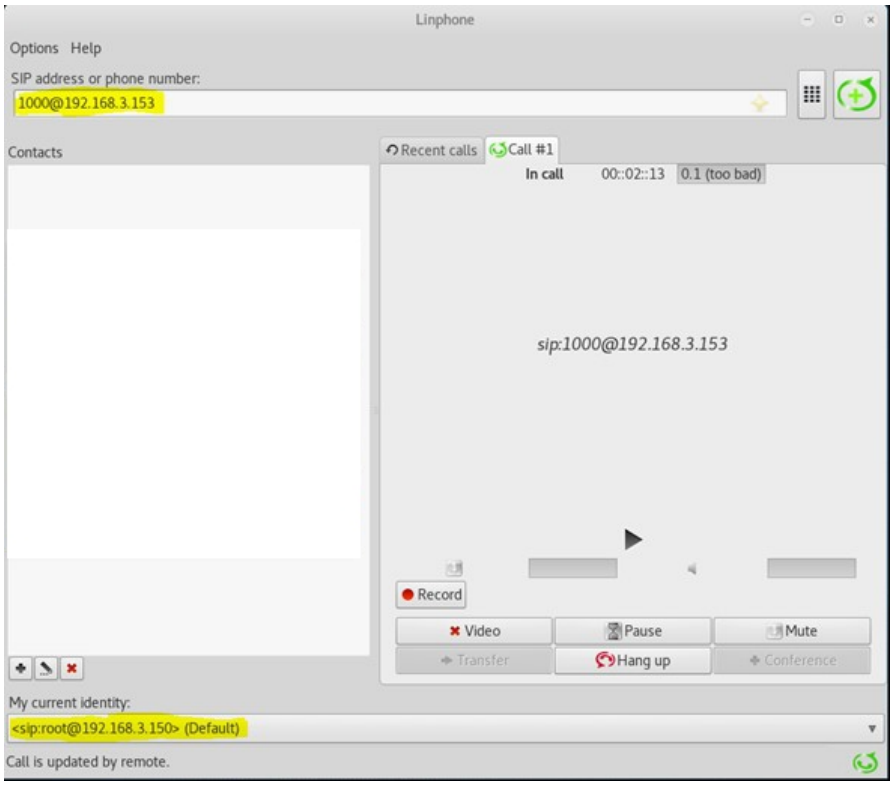

Figure 1. A direct Linphone call.

The experiments illustrated that in the case of Linphone, if TLS is not activated explicitely, i.e., if a user does not explicitly specify the use of TLS (see Figure 2a), then the signalling data will remain unencrypted. In other words, TLS is not enabled by default.

We also examined the case that one TLS-enabled user A calls a non TLS-enabled user B and vice versa (Figure 2). This is a realistic scenario, due to the fact that, as stated above, TLS is not the default option in Linphone. As expected, we noticed that if user A tries to call user B, this call cannot be established. However, interestingly enough, if user B tries to call user A, then the call is being established in UDP (i.e., non-TLS) mode. In other words, the TLS-enabled user accepts calls from a non TLS-enabled user, which means that the security requirements that A has set (i.e., the use of TLS) do not apply for any call that he/she receives; an analysis of the data transmitted through Wireshark indicates that the overall communication is being "degraded" from TLS to UDP, whilst user A does not necessarily realizes it. This can be a serious security issue, if no warning is provided, since any user that chooses to enable TLS assumes that TLS will always protect the signalling data.

To establish the importance of protecting signalling data, we should refer to the approach described in [24], in which it is shown how encrypted Linphone SRTP data can be decoded by a malicious individual and, thus, the original RTP voice data can be decoded—by identifying the secret keys which are transmitted in plaintext within SIP. In such a case the keys for the calling party can be found in the SIP INVITE message, and the keys for the called party can be found in the SIP 200 OK message. This is the case when neither TLS nor ZRTP is used, whilst the key management is being controlled by the signalling protocol entirely. Hence, such a setting should be avoided.

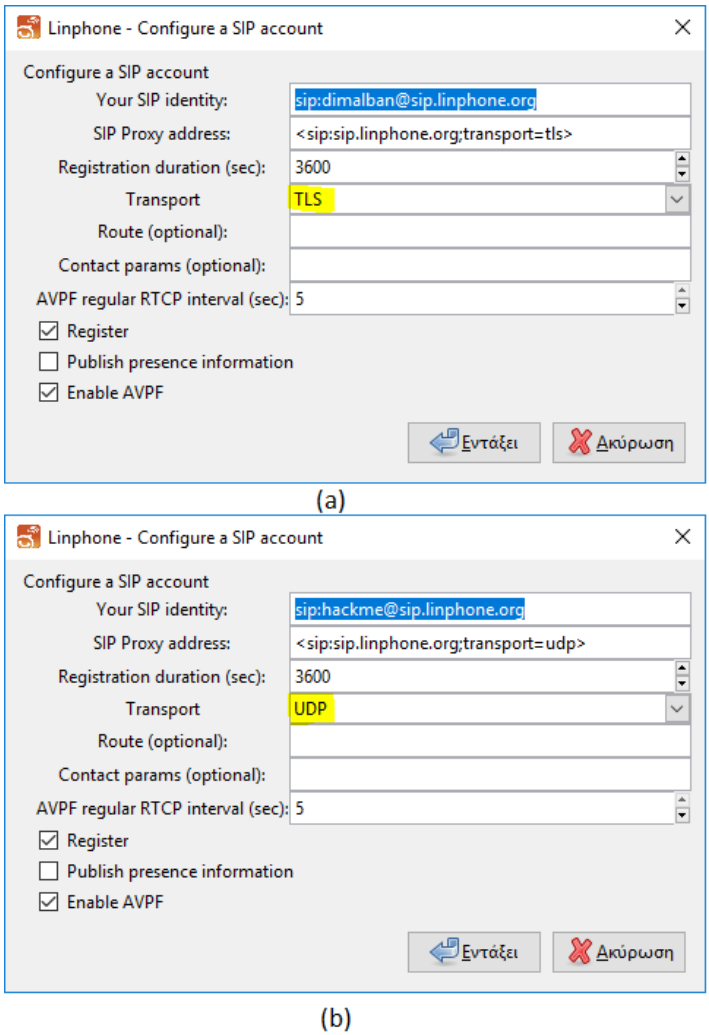

Figure 2. The two Linphone Session Initiation Protocol (SIP) accounts, (**a**) with Transport Layer Security (TLS) and (**b**) without TLS respectively.

4.2. "Ghost" Calls

During the usage of the Linphone application in our experimental network environment, without activating TLS, we came across some strange incoming calls. More precisely, our application received some insistent incoming calls, with a repetition period of about one minute and, thus, they were quite annoying. The SIP-URI address of the caller was of the form 9999@ip, where "ip" was the IP address of our network router. Therefore, we appropriately used Wireshark [25] to identify the actual IP address of the "uninvited visitor"; the result obtained shows that the IP address belongs to a company providing cloud services (virtual servers), which means that, with high probability, these calls are being initiated by someone who uses such virtual servers. It should be also pointed out that, through performing a network scan to SIP devices, we were able to find out that the caller was an Asterisk PBX.

On the other day, we again encountered such strange calls, from a different caller; the SIP-URI address of the caller was 1000@ip, where again "ip" was the IP address of our network router (Figure 3). By working as above, we again managed to find out the actual source of the call, which corresponds to another company which also provides virtual servers as a cloud service. In both cases, we immediately informed those companies about our experimental observations.

When enabling TLS, such "ghost" calls did not seem to take place; however, according to the analysis obtained through Wireshark, there were still attempts for ghost calls with a repetition period

of about one minute, as illustrated in Figure 4. Therefore, TLS seems to provide protection from such ghost calls.

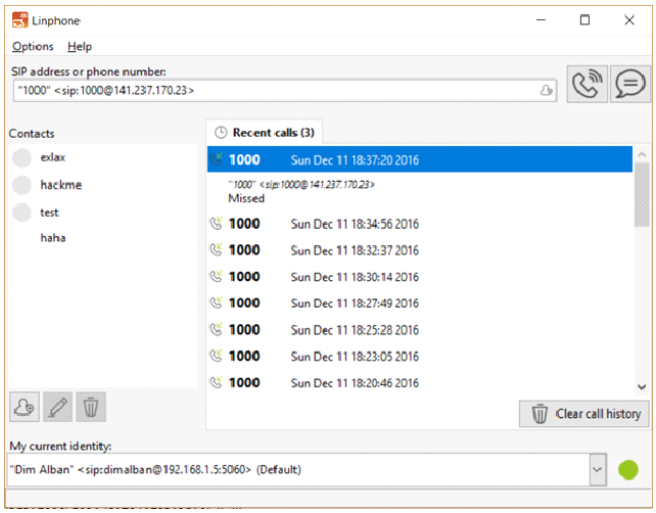

Figure 3. Repetitive ghost calls.

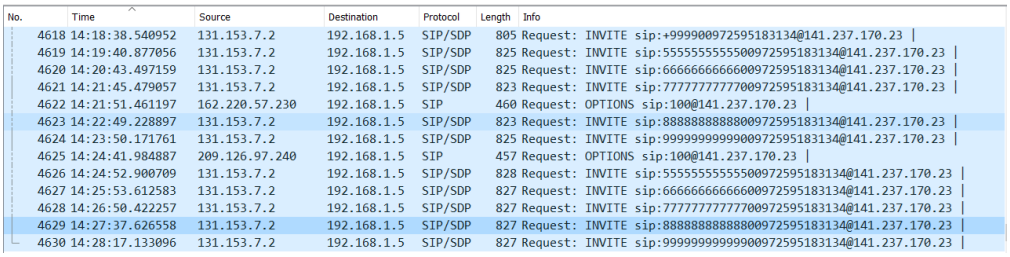

Figure 4. Ghost calls that do not force a call ring due to the TLS.

It appears that these "ghost" calls is a common issue whenever an attacker attempts to scan a SIP network with a tool called SipVicious [26]. This tool is typically used to perform security tests, but it can be also used by an attacker to collect information from a SIP network (e.g., targeting PBX systems so as to find phone lines on that PBX system which have a weak passwords); launching an INVITE scan though on a VoIP Phone may force it to ring [27]. Therefore, the use of TLS not only protects any signalling data from eavesdropping but it can also stop ghost calls.

4.3. Capturing the Transmitted Data

With the use of Wireshark [25] we examined the communication (voice) data transmitted with Linphone, which is encrypted due to SRTP. For the secret key exchange, we allowed the use of ZRTP, whereas we also set up TLS for protecting signalling data—i.e., the maximum possible overall level of protection was applied. As expected, the capture stream is meaningless due to the encryption, whereas we were not even able to isolate the SRTP data from other data (such as ZRTP data).

In case that TLS is disabled, although we were still not able to recover the plaintext, the ZRTP communication could be isolated from other data, as shown in Figure 5 (appropriate filters are being used); the same also holds for the SRTP communication.

Although it seems that we are not able to reveal the transmitted voice message, the fact that ZRTP and SRTP communication can be discriminated exhibits that the absence of TLS suffices to provide some information (possibly useful) to an attacker.

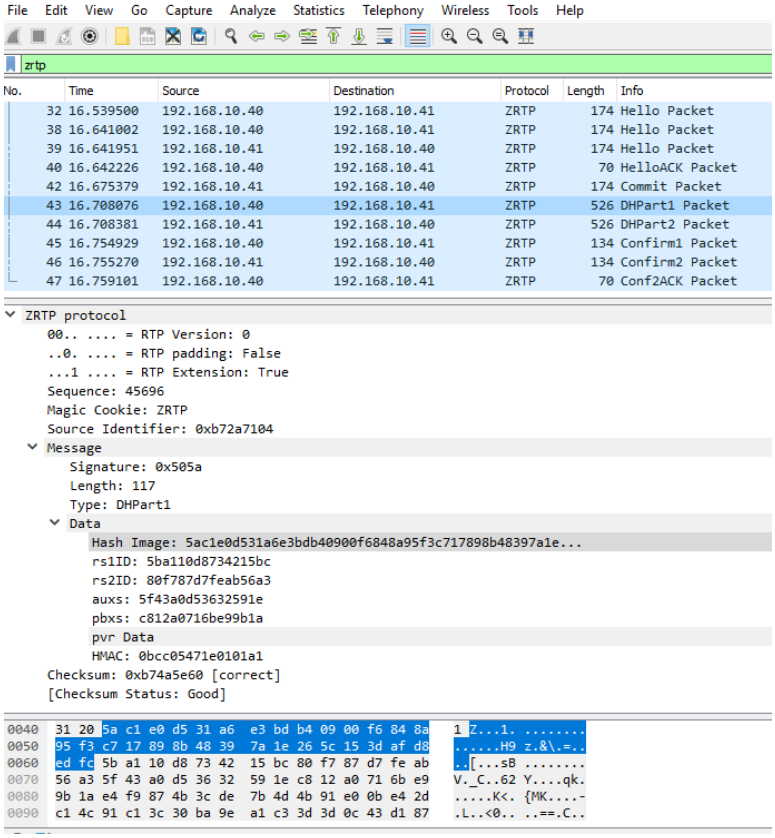

Figure 5. Identifying ZRTP communication when TLS is not used.

4.4. VoIP over a Virtual Private Network

As an additional security safeguard, we also consider the option of building the Linphone VoIP service over a Virtual Private Network (VPN). In general, such an approach allows for strengthening both confidentiality via the encryption of the traffic (i.e., a second–layer encryption that the VPN performs upon SRTP) as well as the authentication of the users. A possible drawback of adopting the VPN is that the quality of the VoIP service may be degraded, whilst it should also pointed out that the users must be educated in how to engage and disengage the VPN when making calls.

For our research purposes, we built a VPN server on a virtual Windows 2012 server provided by a cloud services company. Next, we proceeded by registering our users in this VPN server; in this way, each user has a unique static IP address in the VPN. For each VoIP call though, it is up to the users to set up the underlying VPN connection, which is a non–trivial task for typical users (as those employed in our experimental environment). However, we efficiently addressed this issue via developing an appropriate software in C# to automatically handle VPN connections in a Windows platform. Through this software, a user simply inputs his/her credentials for accessing the VPN. By these means, we initiated a VPN VoIP call from a user outside our experimental LAN to a user that resides within our LAN since they both reside in the same VPN (Figure 6).

To study the overall performance, several VoIP calls over the aforementioned VPN have been carried out for a time period of about 15 days. The calls initiated from our LAN, as well as from 3G and 4G mobile networks, whilst a few calls have been initiated from home networks. The underlying VPN protocols were either PPTP or L2TP/IPSec. For all this time period, the users did not raise any complaint on the quality of the service, even in cases of video calls (note that, according to measurements obtained through stream testers, the VPN server operated on about 60 Mbps in both upload and download streams). In rare cases, the quality of the communication—as this information is

provided by Linphone in real time - has been noticed to be below "good". Usage of several typical download and upload stream testers further confirmed this observation.

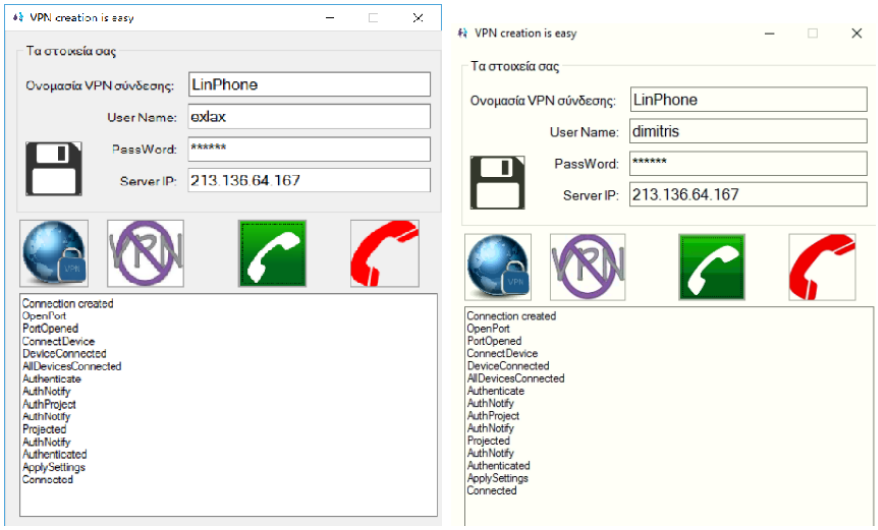

Figure 6. The two VPN connections (the user *exlax* resides outside our LAN, whereas the user *dimitris* resides in our LAN).

4.5. A Further Enhancement

As stated above, the security of the ZRTP rests with the assumption that the voice channel, over which the Short Authentication String (SAS) is validated by the users, suffices to provide the properties of integrity and source authentication. It is up to the users to speak loudly the SAS that they obtain and to proceed with the communication only if the two values of SAS coincide, whereas the mutual authentication is based on recognizing each other's voice. However, as it is shown in [10], an automated SAS voice imitation MITM attack can be successfully mounted. Such an attack can be based on building arbitrary SAS strings from a victim's voice by either reordering previously eavesdropped SAS strings spoken by the victim or from a few previously eavesdropped sentences (less than 3 min) spoken by the victim.

To overcome this issue—which is present to any VoIP SRTP communication relying on ZRTP for secure key exchange—a simple enhancement to further strengthen the authentication mechanism is next proposed; instead of depending only on voice recognition, the SAS comparison may be performed by writing down the SAS value—which in general consists of a few characters—and sending a photo of this written paper to the other party (Figure 7). If this photo is also being sent in conjunction with a photo of the sender keeping this paper in her/his hands (e.g., through a small duration video call), then any doubt of the authentication of the other party is dismissed. Of course, the proposed approach is not an amendment to the protocol itself but to the procedure that should be followed by the users in the process of comparing their SAS values. It should be pointed out that we communicated this simple idea to the Belledonne Communications and, according to the response received, this approach will be included as a suggestion for users in future manuals.

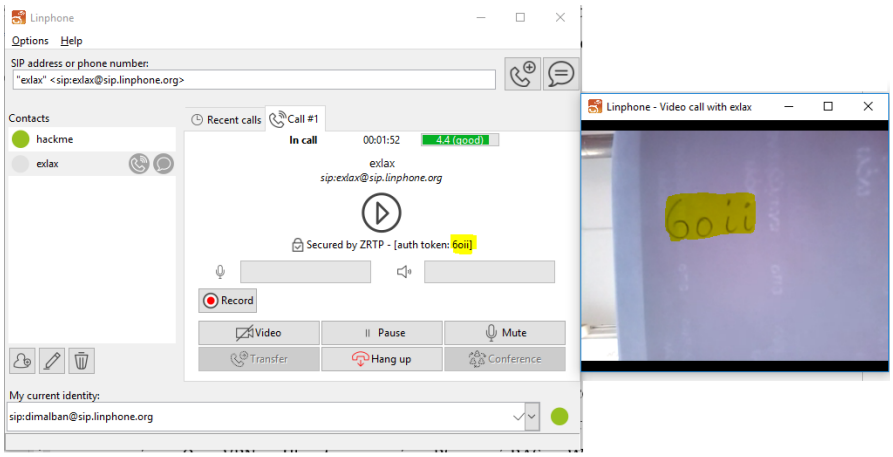

Figure 7. An example of writing down SAS instead of speaking it loudly.

5. Conclusions

In this paper, a preliminary study on the Linphone VoIP service—as a case study—is provided, focusing on security features. It was found that it is essential to ensure that protection of signalling data should be a default option to avoid security misconfigurations. More precisely, it is shown that TLS for protecting signalling data may not be enabled by default and, as a result, even if a user has spontaneously enabled TLS, a direct call may be established with another user who has not enabled TLS, thus degrading the security of the signalling data. Additionally, experiments indicated that TLS also protects from the so-called ghost calls, which are related to malicious scanning of the SIP network. In the process, we also proposed a simple amendment to ZRTP secure key exchanges that could be followed by users when using the ZRTP protocol. Furthermore, according to the experiments performed, we concluded that a Virtual Private Network may be used in conjunction with the VoIP service without significantly reducing the overall performance.

As a conclusion, it becomes evident that TLS should be a default option in any VoIP application. Moreover, VoIP service providers may provide additional useful guidance to their users for further strengthening the overall security, such as to consider employing a Virtual Private Network in a VoIP communication, as well as in case that ZRTP is used to consider writing down the SAS value in a way described in Section 4.5, as an additional safeguard against MITM attacks that rely on automated voice imitation.

It has to be noted that although Linphone was chosen as case study, due to its open source nature and functionality, it becomes evident that the aforementioned conclusions may also apply to other VoIP solutions.

Acknowledgments: The authors would like to thank the anonymous reviewers for their important suggestions, which helped to greatly improve the manuscript.

Author Contributions: The work in this paper formed part of the M.Sc. Thesis of D.A., working under the supervision of K.L. The main ideas that motivated this work are developed by S.S. All the experiments have been performed by D.A. The paper is mainly written by K.L. and S.S., based on the above Thesis.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DTLS	Datagram Transport Layer Security
IP	Internet Protocol
IPSec	Internet Protocol Security
L2TP	Layer 2 Tunneling Protocol
MITM	Man-In-The-Middle
PBX	Private Branch eXchange
PPTP	Point-to-Point Tunneling Protocol
RTP	Real Time Protocol
SAS	Short Authentication String
SCTP	Stream Control Transmission Protocol
SDES	Session Description Protocol (SDP) Security DEScription
SIP	Session Initiation Protocol
SRTP	Secure Real Time Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
ZRTP	(Zimermann) Real-time Transport Protocol

References

1. Kuhn, D.R.; Walsh, T.J.; Fries, S. *Special Publication 800-58: Security Considerations for Voice over IP Systems*; NIST: Gaithersburg, MD, USA, 2005.
2. VoIP Security Alliance. VoIP Security and Privacy Threat Taxonomy; Technical Report; 2005. Available online: https://www.voipsa.org/Activities/VOIPSA_Threat_Taxonomy_0.1.pdf (accessed on 26 November 2017).
3. Wahab, A.; Bahaweres, R.B.; Alaydrus, M.; Muhaemin; Sarno, R. Performance analysis of VoIP client with integrated encryption module. In Proceedings of the 1st International Conference on Communications, Signal Processing, and Their Applications (ICCSPA), Sharjah, UAE, 12–14 Febuary 2013; pp. 1–6.
4. Azfar, A.; Choo, K.K.R.; Liu, L. A Study of Ten Popular Android Mobile VoIP Applications: Are the Communications Encrypted? In Proceedings of the 47th Hawaii International Conference on System Sciences (HICSS), Waikoloa, HI, USA, 6–9 January 2014; pp. 4858–4867.
5. Cattaneo, G.; Catuogno, L.; Petagna, F.; Roscigno, G. Ensuring non-repudiation in human conversations over VoIP communications. *IJCNDS* **2016**, *16*, 315–334.
6. Ashraf, A.; Muhammad, W.; Ahsan, R.S. Efficient Implementation of VoIP over VPN wrt Packet Delay and Data Security. *Int. J. Multidiscip. Approach Stud.* **2016**, *3*, 89–95.
7. Keromytis, A.D. A Look at VoIP Vulnerabilities. *USENIX Mag.* **2010**, *35*, 41–50.
8. Khan, L.A.; Baig, M.S.; Youssef, A.M. Speaker recognition from encrypted VoIP communication. *Digit. Investig.* **2010**, *7*, 65–73.
9. Schürmann, D.; Kabus, F.; Hildermeier, G.; Wolf, L. Wiretapping, End-to-End Encrypted VoIP Calls: Real-World Attacks on ZRTP. *PETS* **2017**, *3*, 4–21.
10. Shirvanian, M.; Saxena, N. Wiretapping via Mimicry: Short Voice Imitation Man-in-the-Middle Attacks on Crypto Phones. In Proceedings of the ACM Conference on Computer and Communications Security, Scottsdale, AZ, USA, 3–7 November 2014; pp. 868–879.
11. Shirvanian, M.; Saxena, N. On the security and usability of crypto phones. In Proceedings of the 31st Annual Computer Security Applications Conference, Los Angeles, CA, USA, 7–11 December 2015; pp. 21–30.
12. Sisalem, D.; Kuthan, J.; Ehlert, S. Denial of service attacks targeting a SIP VoIP infrastructure: Attack scenarios and prevention mechanisms. *IEEE Netw.* **2006**, *20*, 26–31.
13. Wright, C.V.; Coull, S.E.; Monrose, F.; Masson, G.M. Uncovering spoken phrases in encrypted voice over IP conversation. *ACM Trans. Inf. Syst. Secur.* **2010**, *13*, 1–30.

14. Zhang, G.; Fischer-Hübner, S.; Martucci, L.A.; Ehlert, S. Revealing the Calling History of SIP VoIP Systems by Timing Attacks. In Proceedings of the International Conference on Availability, Reliability and Security, Fukuoka, Japan, 16–19 March 2009; pp. 135–142.
15. Zhang, R.; Wang, X.; Farley, R.; Yang, X.; Jiang, X. On the feasibility of launching the man-in-the-middle attacks on VoIP from remote Attackers. In Proceedings of the ACM Symposium on Information, Computer and Communication Security, Sydney, Australia, 10–12 March 2009; pp. 61–69.
16. Linphone. Open Source VOIP Project. Available online: http://www.linphone.org/ (accessed on 26 November 2017).
17. Packetizer. H.323 Versus SIP: A Comparison. Available online: http://www.packetizer.com/voip/h323_vs_sip/ (accessed on 26 November 2017).
18. Network Working Group. *RFC 3550-RTP: A Transport Protocol for Real-Time Applications*; Technical Report; The Internet Engineering Task Force: Fremont, CA, USA, 2003.
19. Network Working Group. *RFC 3261-SIP: Session Initiation Protocol*; Technical Report; The Internet Engineering Task Force: Fremont, CA, USA, 2002.
20. Network Working Group. *RFC 3711-SRTP: The Secure Real-time Transport Protocol (SRTP)*; Technical Report; The Internet Engineering Task Force: Fremont, CA, USA, 2004.
21. McGrew, D.; Rescorla, E. *Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-Time Transport Protocol (SRTP)*; RFC 5764 Technical Report; The Internet Engineering Task Force: Fremont, CA, USA, 2010.
22. Zimmermann, P.; Johnston, A.; Callas, J. *RFC 6189-ZRTP: Media Path Key Agreement for Unicast Secure RTP*; Technical Report; The Internet Engineering Task Force: Fremont, CA, USA, 2011.
23. Belledonne Communications. Secured Communications Using Linphone & Flexisip Solution Description. Available online: http://www.belledonne-communications.com/uploads/images/Solutions-SecuredCommunications.pdf (accessed on 26 November 2017).
24. Critelli, A. Hacking VoIP—Decrypting SDES Protected SRTP Phone Calls, 2014. Available online: https://www.acritelli.com/hacking-voip-decrypting-sdes-protected-srtp-phone-calls (accessed on 20 May 2017).
25. Wireshark (Network Protocol Analyzer). Available online: https://www.wireshark.org/ (accessed on 26 November 2017).
26. Gauci, S. SipVicious: SIP Penetration Testing and Exploitation Kit. Available online: http://blog.sipvicious.org (accessed on 26 November 2017).
27. Gauci, S. If SipVicious Gives You a Ring. SipVicious Blog; 2012. Available online: http://blog.sipvicious.org/2012/12/if-sipvicious-gives-you-ring.html (accessed on 26 November 2017).