Abstract
This paper reviews some security challenges currently faced by VoIP systems as well as their potential solutions. Particularly, it focuses on Zfone, a vendor-neutral security solution developed by PGP’s creator, Phil Zimmermann. Zfone is based on the Z Real-time Transport Protocol (ZRTP), which is an extension of the Real-time Transport Protocol (RTP). ZRTP offers a very simple and robust approach to providing protection against the most common type of VoIP threats. Basically, the protocol offers a mechanism to guarantee high entropy in a Diffie-Hellman key exchange by using a session key that is computed through the hashing several secrets, including a short authentication string that is read aloud by callers. The common shared secret is calculated and used only for one session at a time. However, the protocol allows for a part of the shared secret to be cached for future sessions. The mechanism provides for protection for man-in-the-middle, call hijack, spoofing, and other common types of attacks. Also, this paper explores the fact that VoIP security is a very complicated issue and that the technology is far from being inherently insecure as many people usually claim.

Introduction
Voice over IP (VoIP) is transforming the telecommunication industry. It offers multiple opportunities such as lower call fees, convergence of voice and data networks, simplification of deployment, and greater integration with multiple applications that offer enhanced multimedia functionality [1]. However, notwithstanding all these technological and economic opportunities, VoIP also brings up new challenges. Among them, security is perhaps the most compelling. This paper is about VoIP security, in general, and specifically about a new technology, Zfone, which pretends to overcome some of this technology’s most serious security challenges. Zfone was developed by veteran security expert Phil Zimmermann and is intended to provide vendor-neutral encryption services for VoIP communication.

In order to better understand how this new technology works, this paper briefly introduces several important concepts on traditional telephony as well as their counterpart on VoIP systems. Also, this paper reviews the most common security threats that prey on VoIP systems such as denial-of-service attacks, man-in-the-middle attacks, call hijack, and a few others. Finally, the paper explains how Zfone operates and how this technology is used to prevent aforesaid common VoIP security threats.

The old PSTN
To understand why we need Zfone, first we need to look at the differences between the old public switched telephone networks (PSTN) and VoIP. Let’s start with the old PSTN system.

As of 2002, there were 1.1 billion telephone lines in the World [5]. These 1.1 billion telephone lines operate within a large and complex worldwide circuit-switched network. Figure 1 provides a quick glimpse on how this huge network operates. Let’s say that Customer A picks up his/her telephone. This action closes an electrical loop at the phone company’s switch (Switch 1), which is located at the central office. When Customer A dials a number on the phone, the switch receives and analyzes every digit to determine whether the call is served out of the same switch or not. If the call is served locally, the switch creates a physical link or circuit between both
phones and voice traffic is routed all the way through. If the call is served out of a different remote switch (say, Switch 3), the local switch sends out a query to the signaling network requesting routing information as well as other data. The signaling network—also known as signaling system 7 (SS7)—is a highly complex intelligent system that includes advanced switches and data storage engines. The SS7 system analyzes the information contained in the local switch’s query and determines whether the other party’s line (Customer B’s line) is available or not. Also, the SS7 system uses the same information to decide the best path to route the call through. If Customer B’s line is not available, the remote switch notifies the SS7 system. Consequently, the SS7 system instructs the originator (Switch 1) to put a busy tone to alert the first caller that the line is not available at the moment. If Customer B’s line is available, the SS7 system provides all involved switches with the information necessary to create a physical link between Customer A’s and Customer B’s phones. From this point on, both ends are physically interconnected and the SS7 system* continues monitoring the call until new information is provided by the two parties’ switches indicating whether the call ended or not [8].

---

* Actually, the SS7 system performs other services such as call forwarding, automatic ring-back, and local number portability among others.
† We only need to remember the infamous phone phreakers of the 1970s. The word Phreaker (compound of phone + freak) refers to people who exploit PSTN vulnerabilities to make free phone calls.
‡ The last mile refers to “the final leg of delivering connectivity from a communications provider to a customer” (see http://en.wikipedia.org/wiki/Last_mile).
§ The local loop refers to “the physical link or circuit” that connects “from the demarcation point of the customer premises to the edge” or Central Office of the communication provider’s network (see http://en.wikipedia.org/wiki/Local_loop).
their old telephone service to carry their voice-based and sometimes data communications, while considering Internet-based telephony as inherently insecure—something we will later demonstrate is not necessarily true. But first, let’s discuss how VoIP operates.

**VoIP in a nutshell**

VoIP is a family of technologies that allows voice communication over standard digital packet-switched networks. Specifically, VoIP works on top of the Internet, largely based on the TCP/IP protocol suite. Figure 2 shows the whole VoIP protocol stack.

![Figure 2: Voice over IP protocol stack.](image)

Previously, we described how the PSTN operates on top of two different interdependent systems. The first deals with voice traffic, creating physical links that interconnect end users terminals or phones. The second is an intelligent signaling system (SS7) that provides routing information as well as other advanced functionality. Each one provides some specific services that when combined constitutes the whole telephone system as we know it today. VoIP operates in a similar fashion. However, contrary to the PSTN system that requires two independently built and maintained systems—one for voice traffic and the other one for call signaling—VoIP uses a common infrastructure, the Internet, and two different suites of protocols to provide the required differentiated functionalities.

![Figure 3: Typical VoIP network.](image)
Figure 3 shows a simplified version of a typical VoIP network. In this case, contrary to the PSTN system, both signaling information and voice data travel on the same data network. For signaling, VoIP uses two families of protocols: H.323 and SIP*. The first protocol was developed by the ITU-T† to provide call handling and control, codec specification, and data conferencing on ISDN. H.323 is actually an umbrella for several standards such as: H.225 for call handling, H.245 for call control, G.711/721 for codec specification, and T.120 for data conferencing [8]. The second protocol, SIP, was specifically created by the IETF‡ for multimedia communication over TCP/IP networks. As a consequence of that, SIP has become the de facto standard for VoIP communication today. For data/voice transmission, VoIP uses mostly RTP§, another IETF standard. RTP provides for the reliable delivery of data with real-time constraints such as audio and video. Also, RTP relies on another protocol, RTCP**, to provide quality-of-service features such as congestion control and to collect network performance information. Both H.323/SIP and RTP/RTCP provide the signaling and data transmission functionality that VoIP requires. However, in this paper we will focus on the couple SIP/RTP which today is considered as the de facto standard for VoIP communication. Thus, the following two sections will describe how these two protocols work in more detail.

SIP

Figure 3 actually shows an example of a SIP/RTP-based VoIP network. SIP is a very flexible protocol that can establish and take down any type of session [4]. It is text-based and similar to HTTP in syntax. Also, it uses another protocol, SDP††, to specify additional call details such as codec type, size of packets, and so on. Additionally, just like HTTP, SIP uses Uniform Resource Indicators (URI) to identify logical addresses instead of physical, hardwired addresses. For instance, SIP can use nicknames, email addresses, or telephone numbers to identify call addresses. SIP is so flexible that it allows for creating new services in addition to duplicating traditional telephone functionality.

---

* SIP is an acronym for Session Initiation Protocol. IETF RFC 3261 describes version 2.0 of SIP.
† The ITU-T or International Telecommunication Union’s Telecommunication Standardization Sector is responsible for developing international telecommunication standards.
‡ The Internet Engineering Task Force (IETF) is part of the Internet Society and is responsible for developing and promoting Internet standards.
§ RTP stands for Rea-Time Transport Protocol as specified by IETF RFC 3550 and RFC 3551.
** RTCP is an acronym for Rea-Time Transport Control Protocol and is part of the RTP specification.
†† SDP is an acronym for Session Definition Protocol as described by IETF RFC 2327.
To illustrate how SIP operates, Figure 4 shows the sequence of events that characterize a typical SIP call. The first step is to register the two SIP nodes into a common SIP Registrar database—see lines 1 and 2 in Figure 4. The SIP Registrar collects information about the whereabouts of SIP users. Specifically, the Registrar keeps information about the IP addresses and ports for every SIP registered user—in fact, it matches SIP’s URIs with physical addresses in a similar fashion that DNS servers do with IP addresses and DNS domain names. In this case we assume that both LANs belong to a single corporative domain but are located at different and remote premises. Now, if Joe in LAN 1 wants to make a call to Jane in LAN 2, Joe’s SIP phone will send its INVITE message to the SIP Proxy. The SIP Proxy is responsible for redirecting SIP messages on behalf of local nodes. SIP Proxies get address information from available SIP Registrars—see line 4 in Figure 4. Once the SIP proxy receives a request from a local node, it immediately starts sending out signaling information to the remote node on behalf of the local one—lines 5/6 in Figure 4. Then, if the other party is available, the SIP proxy helps both ends to setup a link to exchange data/voice traffic through RTP. Of course, it is a very raw description of how SIP operates. For instance, we do not discuss the role that Firewalls and other network elements may play in VoIP communication. Later, when we discuss some security issues associated to VoIP, we will come back to these issues. For now, let’s continue studying RTP.

RTP

Although we have not said it yet, SIP can use either TCP or UDP to provide the network transport functionality. Typically, it uses TCP to guarantee a more reliable connection. Unfortunately, TCP is not suited for real time communication applications such as Internet telephony. Therefore, RTP operates on top of UDP, which is a connectionless, best-effort protocol [4]. RTP was primarily designed to provide multi-participant multimedia conferences and consists of two closely-associated components: the real-time transport protocol, which provides real-time data transmission capabilities, and the RTP control protocol (RTCP), that provides Q-o-S monitoring and events information capabilities [6]. Being UDP-based, RTP does not guarantee delivery or prevent out-of-order transmission. Also, RTP does not assume that the

---

* Also, the INVITE message has an SDP capability construct as payload. The SDP capability construct may contain information about version, content type, call-id, contact information, and so on. See Goode in the References for more details.
underlying network is reliable or that it guarantees correct packet delivery. Its main goal is to provide end-to-end delivery of real-time data by specifying additional services such as payload type identification (video, audio, or both), sequence numbering, time-stamping, and basic Q-o-S monitoring.

Figure 5 shows an example of an RTP-based videoconferencing application. In this example, RTP mixes two data stream, one for video (blue) and the other one for audio (red), into a single stream of data. Each data stream constitutes a synchronization source (SSRC) and every SSRC maintains a separate space with its unique SSRC identifier. (Of course, we are disregarding all signaling traffic and components.) RTP can handle multiple SSRC as well as contributing sources (CSRC). The latter corresponds to those cases where multiple SRCs are combined by an RTP mixer that allows the receiver to identify what source is currently active—i.e., what user is currently talking [6].

At this point, it is good to notice that RTP traffic and SIP traffic can travel through different routes. As might be expected, it is a consequence of the way the Internet works—in packet-switched networks, individual packets travel through the best available route regardless of the route other packets in the same stream may have used previously. Of course, this issue may have serious consequences when security considerations are involved. For instance, many countries require that law enforcement agencies be allowed to eavesdrop on targeted communication between suspected criminals [9]. Previously, with the PSTN system, things were a little easier. Service providers usually have control on all traffic, both signaling and data voice, what allowed law enforcement agencies to wiretap telephone lines at, for instance, the central office. Now, because data voice can travel through a different path than signaling—and it may happen that such path is not under the service provider’s control—wiretapping is a lot more complicated to achieve. In fact, VoIP’s eavesdropping may require wiretapping “at every possible entry/exit point” [9].

**VoIP most critical vulnerability**

Although VoIP may be considered another type of data network service, it is important to understand that this technology is not just another type of service. VoIP has a very special characteristic: it is “time critical” [11]. Time has a tremendous impact on this technology’s ability to provide Quality of Service, and to transmit meaningful information as well. Consequently, security considerations for VoIP must take additional steps to fulfill specific

© 2006 Samuel Sotillo
quality demands. First, the technology requires a very low latency—less than 150ms. Second, packet loss cannot exceed the mark of 3%. Third, the technology is highly sensitive to “unquantifiable disrupting factors such as jitter” [11]. Thence, at the end, all these factors converge and constitute the most critical of all VoIP security vulnerabilities: this technology’s inherent sensitivity to disruptions.

Now that we understand how VoIP operates and why it is time critical, the following section will briefly describe some of the most common VoIP threats.

**Common threats**

Thus far, it is safe to claim that VoIP is just another Internet service. Therefore, VoIP systems are exposed to many of the same attacks that predate other Internet services—for instance, operating systems vulnerabilities, denial of service attacks, spoofing, and so on. In any case, this section discusses some of the most common threats that prey on VoIP systems.

**Denial of Service attacks**

In a denial-of-service (DOS) attack, the attacker usually creates a large number of connections or service requests that ultimately overwhelm the target system’s resources [12]. In VoIP networks, DOS attacks come in several forms. Figure 6 illustrates an example of a DOS attack.

In Figure 6, an attacker gains access to a host in the same segment where the VoIP application is located. Because VoIP communication depends on keeping the amount of delay below the 150ms mark, all the attacker needs to do is to create a large quantity of traffic, either RTP or any other high priority traffic, large enough to overwhelm the VoIP system’s resources. Still, the traffic does not need to be aimed at the VoIP system alone and does not need to be VoIP specific. In fact, any kind of traffic that increases the overall network utilization† above 60%-80% may bog down the network and increase the overall delay beyond the QoS threshold. In such conditions, all VoIP services may be down for this particular segment.

---

* In VoIP jargon, *jitter* refers to the variation in time or delay between arriving packets. Possible causes of jitter in VoIP networks are: network congestion, timing drift, and/or changes of routing information (see http://searchnetworking.techtarget.com/sDefinition/0,,sid7_gci213534,00.html).

† We assume that Ethernet is the technology in place.

© 2006 Samuel Sotillo
Unfortunately, DOS attacks are very complex and cannot be addressed through a single do-it-all solution.

**Eavesdropping**

Eavesdropping is the interception, listening, and/or recording of private conversations between parties*. Unfortunately, RTP does not include any mechanism to prevent eavesdropping (such as encryption), which allows an attacker listening the network—for instance, with a packet sniffer—to intercept, listen, and record VoIP communications. Eavesdropping attack are a consequence of failing to use appropriate encryption.

**Man in the Middle attacks**

A man-in-the-middle attack occurs when a third party (the attacker) poses as the other party in a communication which allows an attacker to monitor, record, obstruct, or modify passing information. Man-in-the-middle attacks may be as simple as using a packet sniffer to collect, analyze, and alter protocol payloads. Also, it can be as complex as using ARP spoofing to overcome broadcast segmentation in Ethernet networks and therefore to force all IP packets from the calling parties to pass through the attacker’s host first [9]. Man-in-the-middle attacks are a consequence of a lack of strong encryption and appropriate authentication in raw SIP-based communication.

**Call hijack**

A call hijack occurs when an attacker effectively controls one end of a VoIP call. Call hijack usually occurs after the call has been set up. The most common type of call hijack attack occurs when the attacker first disable the legitimate party (for instance, using a DOS attack) and then proceeds to alter the registered information about this party in the VoIP Registrar’s database. As explained above, SIP-based VoIP communication starts when callers register their information with the corresponding VoIP Registrar (see Figure 4). If an attacker is able to modify this information—for instance, replacing the original IP address with its own address—then the VoIP Proxy will direct all incoming calls to the attacker’s IP address instead of the original caller’s IP address [10]. As previously mentioned, man-in-the-middle and call hijack attacks are possible because SIP is a text-based protocol that does not implement any type of encryption—SIP messages travel in the clear. SIP also lacks authentication and appropriate integrity check of signaling data.

**Spoofing attacks:**

Spoofing attacks are very similar to Call hijack attacks. However, in this case, the attacker assumes total control of the other party’s identity, even before a call has been initiated [10].

**Call fraud**

Call fraud attacks are intended to facilitate the illegal use of the VoIP infrastructure to place free phone calls. In the 1970, phone phreakers exploited certain PSTN vulnerabilities to perpetrate call fraud attacks (see note in page 2) [10].

**Strengthening VoIP security**

Since its inception, many attempts have been done to make VoIP communication more reliable and robust. For instance, to prevent eavesdropping, man-in-the-middle, and spoofing attacks

---

* There is a distinction between unauthorized and authorized eavesdropping. In the US and many other countries, law enforcement agencies may be allowed by law to listen and/or record conversations between suspected criminals. Particularly, wiretapping activities in the US are regulated by the Communication Assistance for Law Enforcement ACT (CALEA).

© 2006 Samuel Sotillo
security experts recommend the use of SIP over TLS [10, 11] as well as IPSec in the context of virtual private networks (VPN). Additionally, a new breed of protocols is emerging that may prevent eavesdropping attack as well. For instance, to guarantee a more reliable transmission of multimedia data, the IETF developed the Secure Real-time Transport Protocol (SRTP)\(^*\). SRTP provides encryption, message authentication, and integrity checking. The protocol uses AES with a 128 bits encryption master key length and a default 112 bits salt key length. The master key is exchanged only once because all necessary session keys are derived from the same master key through a key derivation function. However, SRTP does not define any specific key management protocol—it uses an external one such as TLS or Mickey [9]. SRTP uses HMAC-SHA1 for authentication. To provide protection against replay attacks, the protocol provides for storing previous used messages digests that are compared with newly generated ones in order to guarantee that old messages are not play again.

Although most of these solutions are highly effective, still many of them are far from perfect—perhaps too far. For instance, to prevent eavesdropping, man-in-the-middle, and other spoofing attacks, VoIP communication must rely on TLS or IPSec. However, the use of these technologies is possible only under specific circumstances—mainly, in applications that require end-to-end communication among members of the same organization within a private or corporate network. Neither VPN nor TLS allows for truly free and secure communication among peers, independently of their affiliation which is the way the PSTN operates. TLS [2] could be close to it but it still relies on a third party to provide part of its session key management mechanism—TLS uses PKI encryption which relies on centrally-managed certificates. These certificates are used to create the master key needed by the Diffie-Hellman encryption mechanism used to encipher SIP session information. Finally, many security solutions for VoIP are vendor-centered, which means that secure communication is only guaranteed among elements developed and manufactured by the same vendor. Of course, vendor-centered technologies translate into less portability and freedom, two of the most appreciated features of today’s PSTN communication.

**Zfone**

Zfone is a vendor-neutral security solution for VoIP communication. The technology was developed by Phil Zimmermann, creator of the famous PGP encryption technology. Zfone relies on a new protocol, the Z Real-Time Transport Protocol (ZRTP).

Zfone is not a VoIP client. Instead, it is a new security service that sits on top of the TCP/IP stack. Whenever a new VoIP call is negotiated, Zfone reacts and gains control of the VoIP stream, negotiating a new encryption key between the parties and then encrypting the VoIP packets on the fly [7].

**Zfone architecture: ZRTP**

According to the protocol specification, ZRTP is a key agreement protocol which uses Diffie-Hellman (D-H) key exchange during call setup “in-band” in the RTP stream [13]. The initial D-H exchange creates a “shared secret” which is later used to generate a master key and salt for SRTP sessions. The protocol assumes that the call is established using a signaling mechanism such as SIP.

To illustrate how different is ZRTP from other PKI-based solutions, Figure 7 shows an example of a typical PKI-based VoIP solution.

\(^*\) See RFC 3711.
In Figure 7, the first step consists of the two calling parties negotiating with a certification authority the corresponding certificate or digital signature. Also, the VoIP proxy needs its own certificate in order to establish secure connection with the respective VoIP clients. Line 1 (black) shows this process as performed by all parties. Later, before starting the call, one party (the caller) starts negotiating a new master key/salt with the VoIP proxy—line 2 (blue) in figure 7. Both the VoIP proxy and the caller use their certificates to negotiate the key combination that is used to encrypt signaling information between the VoIP Proxy and the VoIP clients—line 3 (red). Something similar happens between the VoIP proxy and the other party (the receiver). Once all parties have agreed on their respective session keys a new negotiation starts between the two calling parties to negotiate the corresponding master key/salt combination to be used for the SRTP stream--line 4 (green). Lines 3 and 4 represent encrypted sessions that use D-H based on the common shared master key/salt combinations previously obtained.

From Figure 7, it is obvious that PKI-based systems are complex and expensive to implement. First, the system requires a centralized certification authority that provides the initial certificate or digital signatures. Centralize CA may be appropriate in some cases—the military for instance—but they can also be painful to implement when considering the diversity of providers that characterizes traditional telephone systems. Second, the whole negotiating process is long and costly and may impose a big overhead that could affect the system QoS parameters.

Although there are other alternative technologies that stay away from using PKI, most of them are either vendor-centered or lack of a truly secure authentication mechanism—for instance, AT&T 3600, PGPfone, and CryptoPhone among others use a short authentication string mechanism similar to Zfone [13]. On the other hand, Zfone uses a mechanism similar to SSH. Figure 8 illustrates an example of Zfone.
Zfone starts as soon as the signaling phase of the VoIP call setup process is done already—blue lines in Figure 8. At this point, both parties are ready to start sending out RTP packets—red line in the same Figure. In our case, ZRTP is an extension of RTP. Therefore, ZRTP packets are embedded in RTP packets and because RTP endpoints ignore unknown extensions, the protocol is backward compatible. During this phase (discovery), both parties exchange information about the whether they support ZRTP or not as well about the supported version. Also, the parties exchange additional information about the hash, cipher, public key type, and short authentication string (SAS) algorithm they support. Also, they exchanged a unique ZID that allows determining if previously shared secrets exist. Next, a new phase is initiated (hash commitment) where the initiator of the call chooses the hash, cipher, public key type, and SAS algorithm. Then, the same initiator chooses a new fresh random D-H secret value which is based on the public key type previously selected. Then, the initiator computes a new public value. Something similar happens with the other party who must also compute its own public value. The next phase is the D-H exchange, where both parties agree on a new common shared secret. This shared secret is computed by hashing and concatenating the D-H shared secret and any possible set of shared secrets that exist between the parties, including the SAS. Both parties follow a similar path until a new SRTP master key/salt combination is generated—green line in Figure 8. Finally, both parties switch to SRTP and start sending voice data [13]—black line in Figure 8.

In Figure 8 we show another important component of the Zfone architecture—its graphic user interface (GUI). As we saw previously in Figure 7, PKI-based VoIP systems rely on a third-party to provide the digital certificate or signature used to authenticate the initial phase of negotiating. In Zfone-based systems, during the D-H key agreement [3] phase, both parties share a small piece of information, the SAS, which is computed based on the parties shared secrets. The information is used to compute another value that is then shown on the other party’s GUI. In order to authenticate the call, both parties must read their respective SAS value aloud to verify they match.

**How good is Zfone?**

At first sight, Zfone seems to be a very unorthodox approach. Although it is evidently simpler than PKI-based systems, the whole issue of having to read a secret value aloud sounds really bizarre. However, Zfone offers many advantages over its PKI-based counterparts.
First, in a PKI-based system, the initial shared secret or signature is usually in use for more than one call. In fact, most digital certificates and signatures may have a relatively long life expectancy. On the contrary, Zfone’s shared secret is different for every call. Although part of the shared secret is cached between calls, a new one is computed for every new call based on that same cached value. If a man-in-the-middle attack is attempted on this cached shared secret, the fact that the SAS is read aloud suffices to prevent it because the read aloud values will be totally different from the first call on. Even in those cases where the man-in-the-middle attack occurs at the first session and the callers did not check their SAS, the moment they do check their SAS the moment they will find out that the man-in-the-middle attack actually happened at the first session.

Second, because the PKI-based system recycles the initial shared secret over and over again, the entropy of the encryption mechanism remains the same. In Zfone-based systems, because the cached shared secret of the previous call is mixed in with the new random-generated session keys, the entropy of the encryption system augments from one call to the next one.

Third, likewise other protocols that use encryption, Zfone prevents eavesdropping attacks. However, eavesdropping is not only impossible during the first call session but also at any other subsequent session. Concerning Dos attacks, Zfone could be an alternative in cases where the targeted system is the same VoIP system. In that case, authentication and encryption may guarantee that the VoIP client does not pay attention to unauthorized RTP traffic. However, DOS attacks are very complicated and there are multiple scenarios where the attacks might be successful despite of authentication and encryption.

Four, Zfone is simpler and less expensive to implement than most PKI-based systems. Clients have no need to be aware of Zfone functionality, which makes the system portable and more flexible. Zfone does not require special equipments and it could be used on most OS platforms available.

Finally, it is good to mention that there are other approaches to session key agreement. For instance, the Multimedia Internet Keying Protocol (Mickey) computes its shared secret using information available in the signaling. Unfortunately, this approach could compromise the session shared secret if the signaling contribution is not strong enough. However, ZRTP may use this signaling component when computing the common shared secret. Even if the signaling component is weak, the fact that ZRTP hashes together all available secrets reduces the potential impact that a weak link like this may have on the final resulting session key [13].

Conclusions

In this paper we learned that VoIP is a new family of technologies that allows voice communication over standard digital packet-switched networks. In particular, VoIP operates on top of the Internet’s TCP/IP protocol suite. VoIP has some commonalities with the old PSTN. Basically, both systems have two differentiated functionalities. First, they have a signaling component that monitors, route, and tear down calls. Second, a media transport component that is responsible for moving data from one point to another. However, despite their commonalities, VoIP is a totally different approach to voice communication. VoIP does not require separated infrastructures to provide the aforementioned functionality. Because it operates in the Internet, the technology uses the Internet infrastructure to accomplish its goal. Thus, differentiated functionalities are provided not by independently built and maintained systems, but by a common set of protocols.

Also, due in part to the fact that VoIP operates on top of the Internet, we learned that this technology faces more security challenges than traditional POTS. However, we also learned that the technology is far from being inherently insecure as many people are quick to claim. Because
of the same flexibility intrinsic to the Internet, VoIP offers many more options for providing security than traditional POTS.

For instance, we studied that one of these new options for security is Zfone, a new technology that provides encrypted communication between VoIP callers. Zfone is vendor-neutral and offers a lot of flexibility and simplicity while keeping a high standard of security against the most common VoIP security threats. Although the technology’s approach is unorthodox, we think that both Zfone and its partner ZRTP protocol are good seeds for future advances in VoIP security.

To conclude, we acknowledge that providing security solutions to VoIP communication is not an easy task. However, considering the potentially disruptive force of this technology—both in technological and economic terms—we believe that solutions such as Zfone are a good step in the right direction to guarantee that VoIP services will provide the same level of trust and reliability that for more than a century now have characterized the current PSTN system.

References


