iVisher: Real-Time Detection of Caller ID Spoofing

Jaeseung Song, Hyoungshick Kim, and Athanasios Gkelias

Voice phishing (vishing) uses social engineering, based on people's trust in telephone services, to trick people into divulging financial data or transferring money to a scammer. In a vishing attack, a scammer often modifies the telephone number that appears on the victim's phone to mislead the victim into believing that the phone call is coming from a trusted source, since people typically judge a caller's legitimacy by the displayed phone number. We propose a system named iVisher for detecting a concealed incoming number (that is, caller ID) in Session Initiation Protocol–based Voice-over-Internet Protocol initiated phone calls. Our results demonstrate that iVisher is capable of detecting a concealed caller ID without significantly impacting upon the overall call setup time.

Keywords: Voice over IP, security, caller ID concealment, SIP.

I. Introduction

Voice phishing (vishing) is a variant of phishing. Scammers, called vishers, use phone calls to deceive victims into disclosing confidential information or transferring money, by masquerading as a trusted authority (for example, a government agency, bank, etc). As legitimate callers also often ask for such confidential information over the phone, it's not easy for people to distinguish between a visher and a legitimate caller. Moreover, the use of the telephone itself means that certain population groups, such as the elderly, are more vulnerable to vishing. Such factors have led to an increase in vishing attacks [1]. In 2011, for example, the damage due to vishing in Korea was estimated to be about USD 90 million, which was roughly double that of 2010 [2].

To avoid vishing attacks, the call recipient needs to check whether the caller is a trusted entity. However, the incoming number (that is, caller ID) displayed on the phone screen is not sufficient to detect vishing attacks since vishers can modify the displayed number on the phone by using a technique called "caller ID spoofing"; therefore, the recipient cannot be certain, from the displayed number alone, that the phone call is coming from a trusted sender. Rather than the displayed number, the recipient can use the phone caller's voice characteristics, such as pitch, accent, and pronunciation, to effectively detect vishers [3]. For the time being, the best option is to try to educate users about these attacks and the associated risks — however, many security researchers have warned that the effectiveness of such education is inherently limited [4]–[5].

Motivated by the lack of automated solutions to detect vishing [6], we propose iVisher, a system to mitigate vishing attacks by detecting whether a given number displayed on a phone screen has been modified by means of spoofing. iVisher authenticates the caller ID of an incoming call and blocks previously reported caller IDs by performing reachability

Manuscript received Aug. 18, 2013; revised Apr. 15, 2014; accepted Apr. 21, 2014.

This research was supported by MSIP (Ministry of Science, ICT & Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2014-H0301-14-1010) supervised by the NIPA (National IT Industry Promotion Agency).

Jaeseung Song (jssong@sejong.ac.kr) is with the Network Research Division, Sejong University, Seoul, Rep. of Korea.

Hyoungshick Kim (corresponding author, hyoung@skku.edu) is with the Department of Computer Science and Engineering, Sungkyunkwan University, Suwon, Rep. of Korea.

Athanasios Gkelias (a.gkelias@imperial.ac.uk) is with the Department of Electrical and Electronic Engineering, Imperial College London, UK.

© 2014 ETRI

analysis to the display name of a suspicious incoming call (that is, a display name suspected of caller ID spoofing). This analysis uses a gateway (that knows the actual caller ID of the call) in the handling of reachability analysis messages that are attempting to corroborate the actual caller ID and the display name. To evaluate the performance of iVisher, we analyze the signaling message overhead incurred in the iVisher system. The analysis shows that the proposed method is fast enough to be used at runtime. Moreover we simulate the proposed mechanism in a real-world environment and provide effective simulation results showing that iVisher does not introduce any significant impact upon the overall call setup time while detecting caller ID spoofing. In summary, we make the following contributions:

- We introduce a framework that is able to detect a possible vishing attack through checking the verification of the display name of an incoming call during runtime.
- We present the methods used for the initialization of a request for caller ID authentication and the delivering of its result. Such methods are applicable to various terminal types that include not only the latest smartphones but also legacy phones.
- We demonstrate the feasibility of the proposed method by evaluating its performance through numerical analyses and simulation, as well as discussing the potential implementation issues of iVisher and incentives for various stakeholders.

The next section gives an overview and analysis of the current state of vishing attacks to illustrate just how attackers hide their real caller ID. This is followed by a discussion of related works in Section III. We then propose a system named iVisher capable of detecting vishing attacks in real-time in Section IV. A way of implementing the proposed system is described in Section V. Section VI shows that iVisher is capable of detecting vishing attacks without a significant overhead through a simulation and a mathematical performance analysis. The paper finishes with conclusions in Section VII.

II. Background of Caller ID Spoofing

Caller ID spoofing is a technique that modifies the displayed number of an incoming call. This is crucial in vishing attacks since most people rely on the displayed number to authenticate the caller. Unfortunately, caller ID spoofing can be easily implemented in Voice-over-Internet Protocol (VoIP) networks. Here, we focus on caller ID spoofing in VoIP services based on Session Initiation Protocol (SIP) architectures, since most vishing attacks are initiated through SIP architecture [7]–[8].

1. What is Caller ID?

According to RFC 3261 [7], a caller ID is provided by the "From" header of an SIP message. The "From" header contains two pieces of information: the display name and the uniform resource identifier (URI), which is a string of characters similar in form to an e-mail address and typically containing a username and host name. The format of a typical SIP message is as follows:

```
[Format]
From : "display name" sip:URI
[Example]
From : "+1-666-666-6666" <sip:victor@hack.com>
```

2. Ways of VoIP Spoofing

Caller ID spoofing techniques modify the display name rather than the URI. This is because the URI is needed in the actual communication process to identify the caller's phone. If an attacker can change the display name of a caller ID into the number of a trusted institution, such as the phone number of the bank that the recipient commonly uses, the phone recipient might think that this call is from a trusted institution. Thus, the attacker can attempt to deceive the recipient by making use of such relationships that rely on trust.

There are many ways to falsify a caller ID depending on how the caller ID is modified, such as using a softphone [9], controlling a telephone private branch exchange (PBX) [10]–[11], or using an online service (for example, http://www.spoofcard.com).

3. VoIP Spoofing Mechanism

Signaling System No. 7 (SS7) [12] has been standardized for signaling in a telecommunication environment and became the backbone of all worldwide telephony networks. For the seamless integration of the Internet Protocol (IP) network with the public switched telephone network (PSTN), it is important to retain the SS7 ISDN User Part (ISUP) information at the points of interconnection and to use this information for the purpose of call establishment [12].

To understand caller ID spoofing techniques at the system level, we explain how a VoIP phone call takes place in a telecommunication network comprising a PSTN circuit switching (CS) network and an IP network. The architecture of such a network is depicted in Fig. 1. More specifically, a VoIP phone initiates a phone call based on SIP signaling and that is destined to a PSTN/CS number. The SIP messages contain the calling party's identity; that is, its caller ID (see Section II-1).

In the example in Fig. 1, the URI and the display name of the

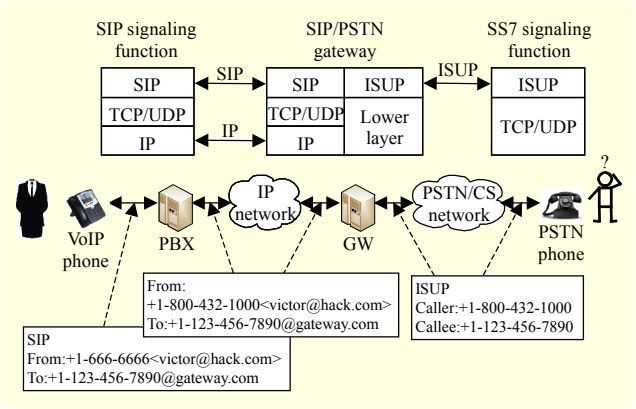

Fig. 1. VoIP call flow with "caller ID spoofing."

caller ID are "victor@hack.com" and "+1-666-666-6666," respectively. The SIP messages access the IP network through a PBX. An SIP/ISUP interworking gateway (GW) is then used to bridge SIP and ISUP networks so that calls originating in the IP can reach ISUP [13] telephone endpoints, and vice versa [14]. In other words, the GW translates SIP messages into ISUP messages. After the translation, the URI (victor@hack.com) is kept behind the GW and only the display name (+1-666-666-6666) is shown to the recipient.

Unfortunately, an attacker (the caller) can easily modify the display name of their own caller ID at the PBX. In the example in Fig. 1, the original number +1-666-666-6666 is spoofed to +1-800-432-1000, which could be the number of a recipient's bank. Since any modification that took place before the GW translation remains behind the GW, only the spoofed display name +1-800-432-1000 is now shown to the recipient. Therefore, it is almost certain that the recipient will think that this call is from their bank rather than from a stranger.

As we already mentioned in Section II-2, in vishing attacks, attackers can only change the display name that appears on the screen of the recipient's phone since the URI part should be used for the SIP/ISUP endpoint communication. We should also emphasize that caller ID spoofing is usually performed at the PBX, before the mapping between SIP and ISUP takes place via the GW, since it's hard to compromise a GW in practice. Thus, in this paper, we assume that the GW has not been compromised.

III. Related Work

Griffin and Rackley [15] introduced general issues relating to vishing attacks. Maggi [16] analyzed typical characteristics of vishing attacks with a collection of detailed reports submitted by victims. Despite the enormous financial loss incurred by vishing attacks, there is little research examining countermeasures to mitigate these attacks. There are some websites that maintain spoofed caller IDs based on reports from victims. However, the numbers in these websites are the display name, thereby the attackers can still perform VoIP spoofing without being affected by systems that filter reported fake phone numbers. In contrast, the proposed system provides an on-demand runtime verification based on a vishers' URI as opposed to their display name.

In the case of spam e-mails, which cause the bulk of e-mail traffic, the Spam over Internet Telephony (SPIT) [17] system uses blacklists and whitelists to filter undesired e-mails — this is considered an adequate solution to the problem. Since similar threats are expected to occur in large-scale networks that are based on the IP Multimedia Subsystem (IMS), the 3rd Generation Partnership Project (3GPP) has been working on the standardization of SPIT-similar solutions, so-called protection against unsolicited communication over IMS (PUCI) [18]. However, for time-critical communication (that is, VoIP), systems akin to SPIT limit the use of its filtering mechanism because of the processing time needed for matching each incoming call to the stored lists and the time a user needs to react to the call.

Other privacy-preserving solutions for SIP, such as [18]–[20], hide the caller and callee IDs. These solutions have been introduced to address vulnerabilities in protocols used for accounting and charging, as well as to provide a way to protect the service provider and the callee against accounting frauds. These approaches can prevent attackers from using spoofed caller IDs because they hide the caller and callee IDs. Compared to these solutions, the proposed system performs actual ID verification (in collaboration with stakeholders such as internet service providers and network operators); therefore, it provides an additional level of security to the callees.

A biologically-inspired vishing attacks detection scheme is introduced in [6]. This scheme analyses the codec parameters from mobile communications and discriminates suspicious calls from others. However, the algorithm proposed in [6] causes performance degradation since all analysis and decisions should be made on the side of the callee. In the iVisher system, a callee initiates the verification process and decides whether the call is suspicious. In addition, in iVisher, most of the verification workload is placed on the core network.

Finally, Wang and others [21] analyzed trust issues in VoIP systems and provided evidence that real-world VoIP services are vulnerable to unauthorized call diversion, which could lead to other attacks, such as the redirection of incoming calls to bogus entities. They suggested the use of Secure Sockets Layer (SSL) or Transport Layer Security (TLS) between a callee and the callee's corresponding VoIP server, so as to mitigate VoIP attacks. However, their proposed methods are unable to detect caller ID spoofing attacks. As [22] stated, there is no effective

solution to mitigate caller ID spoofing attacks compared to other VoIP security issues.

IV. iVisher System

Having explained the ID spoofing procedure, in this section we introduce a scheme called iVisher, which performs caller ID verification (CIV) to detect if ID spoofing has taken place.

1. Overview

iVisher traces back a call through the interworking GW to the PBX that manages the actual caller ID associated with the display name of a given incoming call. In other words, iVisher checks if the user associated with the display name on a recipient's phone is really the one making the phone call. The CIV result is delivered to the recipient to warn them of any suspicious incoming phone calls. In e-mail phishing, it is not easy to interact with the e-mail sender, whereas in vishing, it is possible to communicate with the phone caller since the communication happens in real-time. We take advantage of this real-time interaction to detect whether the caller is a visher.

Figure 2 depicts the key elements of the CIV procedure. For a caller ID X, we use X_{uri} and X_{name} to denote the URI and the display name of X, respectively. When a visher V initiates a call to a user U with V_{uri} and $Bank_{\text{name}}$ (label ①), the interworking GW stores the V_{uri} and $Bank_{\text{name}}$ and uses this information to translate the SIP messages from V into ISUP messages toward U, and vice versa. The information is also used to derive V_{uri} and to forward a CIV request (CIVR) to PBX2 (which uses $Bank_{\text{name}}$ as display name). Since the GW maintains a list of actual URIs and their corresponding display names for calls having a different display name, deriving the URI from the list is not a trivial task. The translation from a tel URI, which describes resources identified by telephone numbers, in an SIP message to an ISUP format is simple and is described in RFC3398 [13]. The GW fills in the Nature of Address (NOA) for the international telephone number format and the numbering plan indicator (NPI) for the actual value of a telephone number (recommendation E.164) based on the given SIP message header.

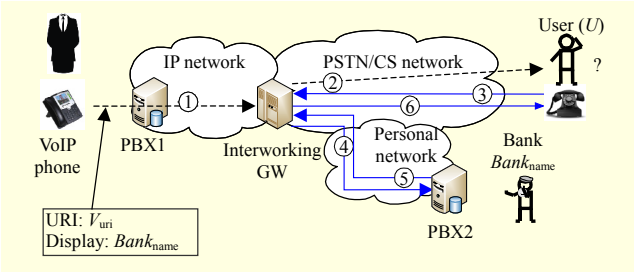

Fig. 2. Procedure for Caller ID Reachability Analysis (CIV).

Depending on user U's network, the ISUP messages are delivered in two different ways, as follows (In the PSTN/CS, only the $Bank_{\text{name}}$ is delivered to U (label ②):

- In the CS network of Global System for Mobile Communications (GSM) and Universal Mobile Telecommunications System (UMTS), a GW mobile switching center (GMSC) identifies the location of U via an internal database that stores subscribers' information (that is, the visitor location register (VLR)/home location register (HLR) and home subscriber server (HSS)), and delivers the call to U through the serving mobile switching center of U (SMSC-U).
- If the user U is located in PSTN, the call is routed through PSTN until it reaches the correct switch that can deliver the call to U.

The purpose of the CIV is to help U to decide whether to trust the caller of an incoming call. Since there exist many legitimate reasons to use caller ID spoofing, as described in Section II-2, a simple caller ID deviation check on a GW might not be an effective solution. Therefore, if U is suspicious of the connected call, he/she can initiate a CIVR via GW to PBX2 that manages the $Bank_{\text{name}}$ (labels ③–④). Since PBX2 controls the entities using the $Bank_{\text{name}}$, it checks that U is now connected to one of its entities and returns the reports back to U (labels ⑤–⑥). In the subsequent sections, we will describe this procedure in more detail.

2. CIV Procedure

Once a VoIP call is converted into a PSTN/CS call at the GW, all intermediate nodes between the GW and the recipient only hold the caller's display name. However, the caller's URI is known at the corresponding interworking GW alone, since this information is needed to perform the SIP-ISUP signal mapping. The iVisher scheme uses the interworking GW to perform CIV, which checks if the caller is using a spoofed caller ID. The CRV process consists of the following three steps:

1) *Initiation*. Typically, a verification process can be performed either as a part of the call setup procedure or after the call has been established. In the first case, the verification process can be initiated automatically based on a user's (that is, a callee's) service subscription. However, integrating CIV into the call setup process incurs call setup delays. In the second case, only selected calls are subject to the CIV process, which results in lower network loads. iVisher supports both options for initiating the CIV procedure, and they are called network-initiated CIV and user-initiated CIV, respectively.

- *Network-initiated CIV.* This step can also be performed as a supplementary service for every incoming call without the

ETRI Journal, Volume 36, Number 5, October 2014
http://dx.doi.org/10.4218/etrij.14.0113.0798

need for U to initiate the CIV process. In this method, the CIV service is only provided to users who subscribe to the service. Therefore, when there is a call destined to a subscriber of the CIV service, the GW automatically initiates the service. For this, the GW first checks that V_{uri} is equal to $Bank_{name}$. If the GW detects any deviation and U subscribes to the CIV supplementary service, then it invokes CIV procedures without human intervention.

- *User-initiated CIV.* If U suspects that the incoming call, with display name $Bank_{name}$, is a fake one, U will initiate the CIV process by asking the corresponding GW to test whether the actual subscriber corresponding to the display name $Bank_{name}$ is calling U (label ③). Dual-tone multi-frequency signaling (DTMF) [23] or user-to-user signaling (UUS) [24] can be used for the initiation of CIV. Note that PSTN terminals can only use DTMF for the initiation.

2) *Verification Request.* Upon receiving a verification request for $Bank_{name}$ (label ③), the corresponding GW will take the following actions:

a. It finds the stored URI (V_{uri}) that corresponds to the $Bank_{name}$ used to initiate the call between V and U. Since this request is performed as part of the call establishment (for network-initiated CIV) or additional signaling over existing telephone lines (for user-initiated CIV), the GW can easily derive URI V_{uri} from the request.
b. It generates a CIVR message consisting of V_{uri} and $Bank_{name}$ (detailed examples and behaviors of the GW are described in Sections V-1 and V-2).
c. It forwards the CIV message to the PBX (PBX2 in Fig. 2) that corresponds to the actual subscriber (they will have $Bank_{name}$ as their display name) to check whether this subscriber is now calling to U (label ④).

It is safe here to assume that major enterprises, such as banks and government agencies, use PBX systems to allow internal users to switch between calls placed on local lines and calls placed on external lines. The PBX is owned and operated by the enterprise rather than a telephone company or service provider. Such enterprises can protect their customers from vishing attacks by deploying the iVisher system at their PBX. The existence of an iVisher-compatible PBX is a fundamental requirement of the iVisher system.

3) *CIV processing and response.* Upon receiving the CIV message (label ④), the PBX2 acts as follows:

a. It creates a $Bank_{uri}$ list (note that more than one X_{uri} may correspond to the same X_{name}) of its registered subscribers that share the $Bank_{name}$ and that are currently online.
b. It checks whether any $Bank_{uri}$ in the list is the same as V_{uri} in the CIVR.

In many typical telecom systems, there is a PBX in a company that manages all outgoing calls for the purpose of the calling line identification presentation (CLIP) service, which provides the information of the calling party. Such PBXs authenticate the calling party and insert a common display name into all outgoing calls made by their registered subscribers. This allows the PBXs to track all active call-related information (such as the caller ID and callee ID of a call). In the CIV procedure, such information is used to generate verification results.

PBX2 reports the verification results to the corresponding GW (label ⑤), and then GW forwards the results to U (label ⑥). As addressed previously, the CIV results for a call can be delivered to the recipient using various implementation technologies. For example, the SMS message can contain relevant information about the spoofed call, such as the caller ID, display name, and the caller's location, so that if the caller ID is not identical to the display name or the call has originated from a foreign country, the recipient can be in doubt about the validity of the call.

3. Implicit Spoofing

As explained earlier, one of the fundamental requirements of the proposed methodology is the availability and use of an iVisher-compatible PBX system by the involved enterprise. This is because the PBX is responsible for acknowledging receipt of a CIVR and reporting the ID verification result. If the displayed name is an invalid number or corresponds to any random PSTN/CS number, the request will not be routed to any iVisher-compatible PBX and the GW will not receive any response for the request. Such a lack of response can be used as an implicit indication of spoofing. In this case, a timeout can be applied to identify these calls. The duration for the timeout should be long enough so that the GW has to receive all responses (should they exist). However, since the CIV process is performed independently from the call setup procedure, it does not incur any call setup delay.

V. Suggested Implementation

It is clear from the above that a series of modifications are required in existing systems and protocols to accommodate the iVisher system and its functions. Fortunately, all these modifications and the required signaling exchange can be easily implemented at the application layer. Now, we outline the entities involved in the proposed CIV process and the expected changes required of them.

1. User Terminal

First of all, if the CIV process is triggered by the network as a supplementary service, no modification is required on the end

```
1 0000 0001 CIV discriminator
2 0000 0000 Message type - request (0) or result (1)
3 0000 1100 Content type - display name, result, etc.
4 1111 1100 Length of content
5 1011 1001 Content (for example, Bank_name)
```

Fig. 3. Example of CIVR message using UUS.

user's terminal. The terminal only needs to receive the result of the CIV process, which can be provided easily at the application layer by existing services. Various technologies, such as Short Messaging Service (SMS) or background audible tones, can be used to deliver the CIV result to the recipient *U*, according to the capability of terminals.

For the scenario where the end user initiates the CIV process, the application that provides the voice-call functionality at the end user's terminal should be modified to allow the exchange of a CIVR during a phone call. In this case, UUS or DTFM can be used to initiate the CIVR.

Figure 3 illustrates an example of a CIV initiation message using the UUS protocol — UUS is a service that allows the end users' terminals to exchange information for up to 128 octets during a call. In this figure, the "CIV discriminator" and "Message type" fields indicate that this is indeed a CIVR message (lines 1–2). The following fields (lines 3–5) illustrate the information (that is, type, length, and content) about the display name $Bank_{name}$.

2. GW

A GW converting SIP to ISUP plays a key role in the CIV check. Since the GW behaves as a proxy on behalf of the recipient terminal, it requires functions to handle CIV messages, such as being able to modify and forward a request. In other words, when the GW receives a CIVR, it should check the actual caller ID of the call, add the caller ID into the request, and then forward the request to the number addressed in the display name of the call. A function must also be implemented in the GW to deliver the CIV results of the request to the recipient terminal using, for example, SMS or audible tones.

While this is a significant and challenging change, operator-sponsored forums like Open Mobile Terminal Platform (OMTP) are working with key mobile operators to discuss and recommend mobile terminal requirements to help protect against threats such as malware and fraud attacks. As incidents of vishing continue to rise, it seems likely that having the availability to selectively filter such a fraudulent offence will, in time, persuade operators to upgrade their systems. Note that the CIV message processing can be implemented only in software on the GW without incurring unacceptable costs; therefore, no extra hardware is introduced.

An attacker can use the proposed iVisher countermeasure scheme to launch a denial-of-service (DoS) attack on a GW. To prevent such attacks, an intrusion detection system (IDS) [25] recognizing deviations from expected verification procedures can be integrated into the iVisher system.

3. PBX of Concerned Enterprises

Furthermore, the PBXs of involved enterprises need to be modified to process the CIVR messages and forward the CIV check results to the corresponding GW. These PBXs have to maintain a list of caller IDs of registered subscribers who use the same display name. When these PBXs receive a CIVR, they check whether the caller ID in the request is on the list of registered caller IDs. If the caller ID is not on the list, then they reply to the GW with detailed information. This checking process on the PBXs could allow an imposter to retrieve sensitive information (for example, the presence of an employee at a company); therefore, it poses concerns over privacy. For such a concern, we could add a network behavior analysis function on the GW to detect suspicious activities among network traffic. Intrusion detection techniques [26]–[27] can be effectively used to implement this functional component. We note that a lightweight IDS [25] (for example, using rule-based misuse detection) might perform well enough since there are only a few attack patterns for the CIVR protocol (for example, a high rate of CIVRs from the same source), unlike complex rules for a typical host machine. Since the PBX systems belong to enterprises that are willing to protect their customers from vishing attacks by deploying the iVisher system, such modifications can be easily implemented.

VI. Performance Evaluation

There are two important criterions for evaluating the signaling performance of protocols for SIP and PSTN/CS interworking: signaling load and call setup delay [28]. Here, we discuss the signaling overhead incurred by the CIV process through numerical analysis (Section VI-1) and simulation on a real VoIP service (Section VI-2).

More specifically, our numerical analysis focuses on the SIP-ISUP call setup signaling process for UMTS networks as a reference. The reason we have chosen to use the cellular end-user case as an example is because it involves two different networks (that is, PSTN and UMTS); therefore, it requires additional signaling as compared to the PSTN end-user case (however, our analysis can be easily extended to the latter case). On the other hand, in our simulation setup, both end users are located in the real IP network so that realistic network characteristics, such as number of hops between network entities, can be used in our call setup delay analysis.

1. Numerical Analysis of Signaling Load

In this section, we compare the signaling load required by iVisher with the load required to setup an SIP-UMTS phone call. For performance analysis, we need to consider the relative weights of the exchanged signaling messages. We assign reasonable weight values using a method similar to the one introduced in [28]. These weight values are listed in Table 1.

In GSM, UMTS, and GPRS core networks, the Mobile Application Part (MAP) [29] is the application layer protocol used to access the HLR, VLR, MSC, and so on. Among other functions, MAP messages are used to provide mobility services, such as location management (roaming), authentication, and so on. ISUP [12] messages are a part of SS7 [12] and are used to set up telephone calls as well as to support supplementary services in PSTN. ISUP messages will be used for CIV signaling, hereafter referred to as ISUP CIV messages. The Domain Name System (DNS) [30] messages provide user-friendly distributed naming services for devices or services on the Internet. The weights of these signaling messages were chosen to reflect the protocol complexity as well as the number of nodes and geographical distance each message must cross. Finally, as described in Section IV, the CIVR messages are used to initiate a CIVR or to deliver the CIV result to the end user. Since CIVR messages contain simple data (of a complexity similar to that of DNS messages), a low overload weight value of 0.5 is assigned to them.

SIP-UMTS call setup (C1). Figure 4 shows the message exchange process to initiate an SIP-UMTS phone call. Table 2 shows the signaling overload of the exchanged messages for each protocol used in the SIP-UMTS call setup procedure.

Table 1. Message processing overloads.

Symbol	Description	Value
o_{sip}	Overload of an SIP message	1.0
o_{isup}	Overload of an ISUP message	1.0
o_{map}	Overload of an MAP message	1.5
o_{dns}	Overload of a DNS message	0.5

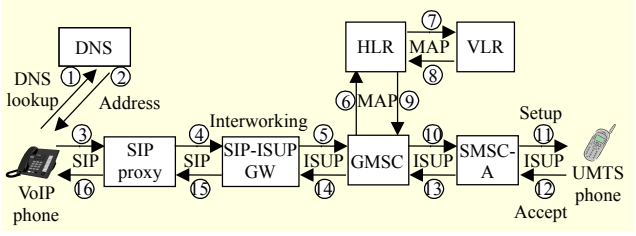

Fig. 4. SIP-ISUP call setup procedure for a UMTS end user.

Table 2. Signaling overload of exchanged messages for each protocol used in SIP-UMTS call setup procedure (see Fig. 4).

Protocol	Messages	Overload
DNS	{1, 2}	o_{dns} (0.5) × 2 = 1
SIP	{3, 4, 15, 16}	o_{sip} (1.0) × 4 = 4
ISUP	{5, 10, 11, 12, 13, 14}	o_{isup} (1.0) × 6 = 6
MAP	{6, 7, 8, 9}	o_{map} (1.5) × 4 = 6

Table 3. Signaling overload of exchanged messages for each protocol used in user-initiated CIV procedure (see Fig. 5).

Protocol	Messages	Overload
CIV	{1, 10}	o_{dns} (0.5) × 2 = 1
ISUP	{2, 3, 4, 5, 6, 7, 8, 9}	o_{sip} (1.0) × 8 = 8

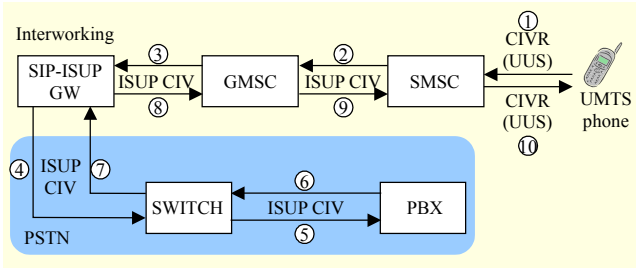

Fig. 5. User-initiated CIV procedure.

From Table 2, we can calculate the overall normalized signaling load required for SIP-UMTS call setup by summing the individual overloads as follows:

$$O_{SIP\text{-}UMTS} = (o_{dns} \times 2) + (o_{sip} \times 4) + (o_{isup} \times 6) + (o_{map} \times 4) = 1.$$

Therefore, seventeen normalized load units of signaling are required to setup an SIP-UMTS phone call. This value will be used as a reference to evaluate the performance of iVisher in terms of signaling load.

User-initiated CIV (C2). As explained in Section IV-2, we assume that major enterprises, such as banks and government agencies, will be connected to the PSTN through PBX systems. Therefore, the CIV-required signaling is exchanged between the end-user and the enterprise's PBX through the corresponding interworking gateway (SIP-ISUP GW). Figure 5 depicts the message exchange process required for CIV initiated by a UMTS end user. Table 3 shows the signaling overload of the exchanged messages for each protocol used in the user-initiated CIV procedure. From Table 3, we can calculate the overall normalized signaling load for the user-initiated CIV procedure as follows:

$$O_{USER\text{-}CIV} = (o_{isup\text{-}civ} \times 8) + (o_{civ} \times 2) = 9.$$

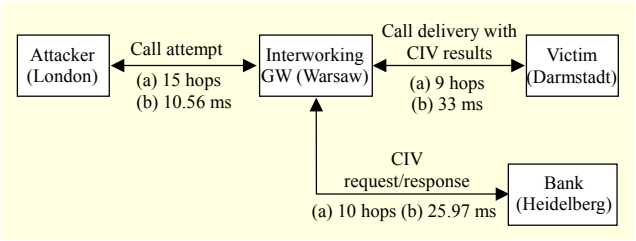

Fig. 6. Simulation scenario and configuration: (a) total route length in hops and (b) mean of the round-trip time (RTT) values of all hops.

Table 4. Signaling overload of exchanged messages for each protocol used in the network-initiated CIV procedure (see Fig. 6).

Protocol	Messages	Overload
DNS	{1, 2}	$o_{dns}(0.5) \times 2 = 1$
SIP	{3, 4, 19, 20}	$o_{sip}(1.0) \times 4 = 4$
ISUP	{5, 6, 7, 8, 9, 14, 15, 16, 17, 18}	$o_{isup}(1.0) \times 10 = 10$
MAP	{10, 11, 12, 13}	$o_{map}(1.5) \times 4 = 6$

We can see that nine normalized load units of signaling are required for the user-initiated CIV process, which is roughly half (about 52.9%) of the signaling required for the SIP-UMTS call setup. In other words, the total signaling load required during the call setup and user-initiated CIV process is 26 units. From this, we concluded that the amount of traffic used in iVisher, with a user-initiated CIV procedure, is roughly 1.5 times greater than that of the conventional call setup (C1) without any protection at all. This additional signaling traffic load does not affect the call setup procedure because the CIV process is performed after the call has been established.

Network-initiated CIV (C3). As explained in Section IV-2, for a network-initiated CIV, the CIV process can also be integrated as part of the call setup procedure. Figure 6 shows the message exchange process required for this case. In this method, the victim's SIP-ISUP GW first checks the display name of the call before call setup is complete. If the display name is not same as the caller ID of the call, the GW queries the PBX for a CIVR. Table 4 shows the signaling overload of the exchanged messages for each protocol used in the network-initiated CIV procedure.

From Table 4, we can calculate the overall normalized signaling load for the network-initiated CIV procedure as follows:

$$O_{NETWORK\text{-}CIV} = (o_{dns} \times 2) + (o_{sip} \times 4) + (o_{isup} \times 10) + (o_{map} \times 4) = 21.$$

Compared with the conventional call setup procedure (C1), the network-initiated CIV call setup uses four more ISUP-CIV messages. Therefore, in total, twenty-one normalized load units of signaling are required for this approach. From this, we concluded that the amount of traffic used in iVisher, with a network-initiated CIV procedure, is roughly 1.2 times greater than the traffic required for a conventional call setup without any protection at all. This additional amount may incur an additional call setup delay because the CIV process is performed as part of a call setup procedure.

2. Call Setup Delay Analysis Using Simulation

To show that the latency overhead incurred by the CIV process is, in practice, acceptable for voice calls, we tested our CIV design on a real VoIP service and analyzed the results. For testing, we chose a scenario whereby we used a network-initiated CIV and an interworking GW that automatically initiates a CIVR when it receives a spoofed call before delivering the call to the victim. This is because the call setup delay of a user-initiated CIV is no different to that of the conventional call setup, and we wanted to remove human intervention in the process of responding to incoming call events. Figure 7 shows the configuration of our simulation. The GW was located in Warsaw (Poland), and the attacker, victim, and bank were located in London (UK), Darmstadt (Germany), and Heidelberg (Germany), respectively. In our testing, a system with a 2.4 GHz Intel Core2 Duo machine with 2 GB of RAM was used as the interworking GW. Routing lengths and packet delay are measured by using the traceroute tool. A modified version of Linphone[1] and Asterisk[2] are used for VoIP

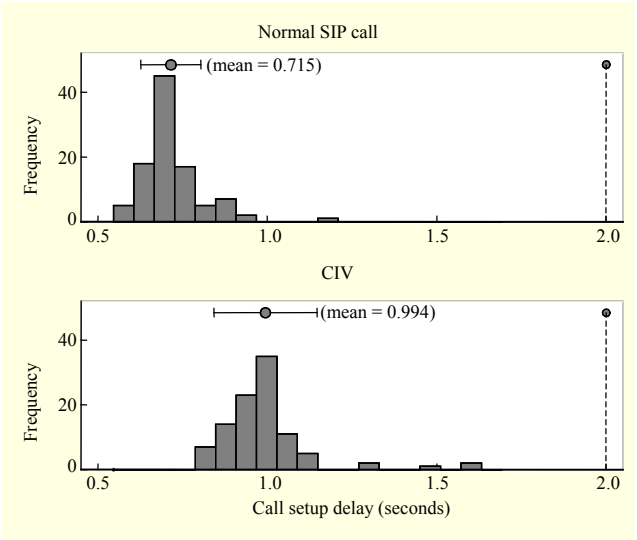

Fig. 7. Histograms of the call setup delay values for a normal SIP call (above) and CIV (below). The y-axis represents the number of values of each bin.

1) http://www.linphone.org/
2) http://www.asterisk.org/

ETRI Journal, Volume 36, Number 5, October 2014
http://dx.doi.org/10.4218/etrij.14.0113.0798

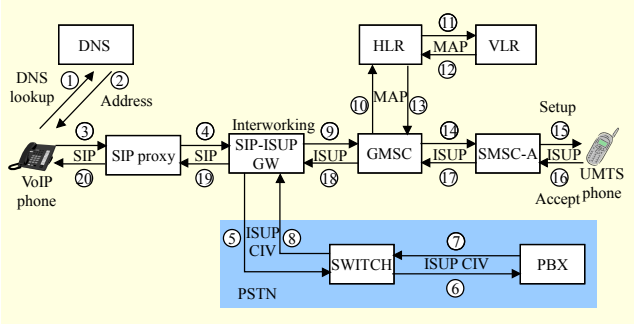

Fig. 8. Network-initiated CIV procedure.

clients and PBX, respectively.

Because of technical difficulties experienced in our testing using PSTN, we assume that both the attacker and the victim are located in IP networks. Under this setting, we test the network-initiated CIV scenario and measure the call setup time (CIV). We repeat the same procedure 100 times to analyze the average performance. For comparison, we also measure a normal SIP call setup establishment time (normal); that is, the time difference between the first invite and the following 180 ringing responses that indicate that the callee is being alerted. Figure 8 demonstrates the distribution of the call setup delay values in our experiments.

In Fig. 8, normal SIP calls and CIV delay values have similar distributions, but the distribution of CIV values are shifted toward the right; the mean of the CIV values (about 0.994 s) was slightly greater than that of the normal SIP calls (about 0.715 s). This is due to the delay gap (with a mean value of 0.281 s) needed to handle the CIV related messages between the interworking GW (Warsaw) and bank (Heidelberg). Note that this additional delay is not significant in practice since it still guarantees the recommended or required average delay values (3.0 s, 5.0 s, or 8.0 s for local, toll, and international calls, respectively [31]) and is significantly less than the average setup delay of existing systems (2.0 s [32]). These results indicate that iVisher, operating under a network-initiated CIV, can be implemented to meet real-time constraints on voice calls without having a significant impact on the overall call setup time while providing an effective caller ID validation service.

3. Discussion

On-Demand runtime verification. It was shown that the iVisher system incurs several costs, such as the processing of CIV messages. However, these costs do not affect a normal call establishment procedure since the CIV process is performed independently in runtime. Only suspicious calls, where there is a need for verification, will utilize the verification data exchange, and this would decrease the network and processing loads on the switch. For this on-demand verification, we propose the user-initiated CIV procedure (see Section IV-2). This procedure introduces a function that enables verification of the caller ID upon request. During the call, whenever the end user wants to check the caller, they trigger a function on their telephony device.

Incentive analysis. In addition, the proposed system increases incentives for stakeholders to participate. Usually incentives are needed to induce participation. In this proposal, we believe that each stakeholder does or does not have reasonable enough incentives and outline these as follows:

- Frequent targets (for example, major banks, hospitals, and government agencies) of vishing attacks may wish to mitigate vishers, masquerading as them, so as to protect their clients.
- Internet service providers (ISPs) have little incentive to stop vishing attacks since they do not suffer from any high costs resulting from caller ID spoofing. So, the role of the government is important here; it can play a useful supporting role in leading ISPs to invest in the fight against vishers.
- End-users who are subscribed to a service provided by the iVisher system can check a suspicious call so that they may avoid vishing attacks. Vishing attacks are targeting end users in an attempt to make them make a wrong decision. However, in telecommunication systems, end users do not receive enough information to make a correct decision. Thus, providing end users with a right to initiate a caller ID checking procedure and delivering the checking results can meet end-users' expectations.

Comparison with existing techniques. Table 5 shows how the proposed method is different from the other existing technologies previously addressed in Section III. For this purpose, we use the following criterion:

- Caller ID spoofing: Does the solution mitigate caller ID spoofing attacks?
- Runtime checking: Does the solution provide a mechanism capable of detecting a vishing attack during runtime?
- Latency: Does the solution add significant network latency to detect vishing attacks?
- Capital expenditure (CAPEX) & operational expenditure (OPEX): Expected costs when implementing CAPEX and using OPEX (the proposed solution).

As Table 5 shows, the proposed iVisher system provides a suitable mechanism for caller ID spoofing attacks in terms of the given criteria. The iVisher system, however, does not support other types of VoIP attacks, in particular VoIP farming attacks that can be protected by [21]; therefore, other mechanisms need to be integrated into the iVisher system to protect network systems from such attacks.

Table 5. Comparison with the existing techniques described in Section III.

Method	Detect Caller-ID spoofing	Runtime checking	Call setup latency	CAPEX & OPEX
CIV check	Detect caller ID spoofing attack properly	Support user and network initiated runtime checking	User-initiated check does not incur any latency. Network-initiated check incurs a little latency.	Can reuse existing supplementary services. PBX can easily be modified through software updates. (Section V)
Bio-inspired [6]	Provide a mechanism detecting caller ID spoofing	Support user initiated runtime checking	Signal processing on a callee's terminal adds a lot of latency.	High. All terminal has to implement signal processing function.
SPIT-alike [17]–[18]	Support based on black and white list.	No runtime checking	The numbers of URIs in blacklists and whitelist affect latency.	Infrastructure entities need to be modified to support black and white list.
Trust system [21]	Do not support caller ID spoofing attacks but provide a mechanism for farming attacks.	No runtime checking	Incur very little latency.	All nodes have to support SSL or TLS.

VII. Conclusion

We proposed a novel system named iVisher to detect ID spoofing–based vishing attacks in SIP-PSTN/CS networks. iVisher performs CIV by tracing back an incoming call to its corresponding SIP-ISUP interworking GW to identify whether the display name has been spoofed. We also discussed important implementation issues and analyzed the incurring signaling overhead of the proposed authentication method. In addition, we simulated our CIV process in a real VoIP environment, and the results reveal that the proposed verification process does not introduce any significant call setup delay on the overall call setup time. It was shown that the iVisher system can be implemented without significant signaling overhead or modifications of existing network infrastructures and protocols.

Two important caveats need to be noted regarding the present system. Firstly, the performance of the iVisher system can be changed depending on the end-users' responses. Unfortunately, when many users encounter security warning messages, they often disregard the messages without caution. Therefore, it seems a challenging issue to attract users' attentions to the ID spoofing verification results. Next, the support and participation of concerned entities that legally use the same originating caller ID for business purposes are keys to the success of the proposed system. It is our hope that such entities will be motivated to use iVisher as a means of enhancing their protection against fraud, which in turn will enhance their respective security reputations.

Future work is envisioned to extend the system to detect other attacks (for example, spam messages) in VoIP networks. We will also consider how to integrate other existing systems with iVisher to reduce maintenance costs.

References

[1] S.T. Chow, C. Gustave, and D. Vinokurov, "Authenticating Displayed Names in Telephony," *Bell Lab Techn. J.*, vol. 14, no. 1, 2009, pp. 267–282.

[2] "Voice Phishing Victims Retrieve Stolen Money," *Yonhap News*, Mar. 2012.

[3] J.-H. Chang, "Statistical Model-Based Voice Activity Detection Based on Second-Order Conditional MAP with Soft Decision," *ETRI J.*, vol. 34, no. 2, Apr. 2012, pp. 184–189.

[4] J. Evers, "Security Expert: User Education is Pointless," Oct. 2006. http://news.cnet.com/2100-7350_3-6125213.html

[5] S. Gorling, "The Myth of User Education," *Proc. Virus Bulletin Int. Conf.*, Oct. 19, 2006.

[6] J.H. Chang and K.H. Lee, "Voice Phishing Detection Technique Based on Minimum Classification Error Method Incorporating Codec Parameters," *IET Signal Process J.*, vol. 4, no. 5, Oct. 2010, pp. 502–509.

[7] IETF RFC 3261, "*SIP: Session Initiation Protocol*," June 2002.

[8] E.-J. Yoon et al., "A Secure and Efficient SIP Authentication Scheme for Converged VoIP Networks," *Comput. Commun.*, vol. 33, no. 14, Sept. 2010, pp. 1674–1681.

[9] H. Dwivedi, "Unconventional VoIP Security Threats," *in Hacking VoIP: Protocols, Attacks, and Countermeasures*, No Starch Press, 2008.

[10] A. Majumder and J. Caffery Jr. "Power Line Communications," *IEEE Potentials*, vol. 23, no. 4, Oct. 2004, pp. 4–8.

[11] M. Qadeer and A. Imran, "Asterisk Voice Exchange: An Alternative to Conventional EPBX," *Int. Conf. Comput. Electr. Eng.*, Phuket, Thailand, Dec. 20–22, 2008, pp. 652–656.

[12] R.P. Ejzak, C.K. Florkey, and R.W. Hemmeter, "Network Overload and Congestion: A Comparison of ISUP and SIP," *Bell Labs Techn. J.*, vol. 9, no. 3, 2004, pp. 173–182.

[13] ETSI EN 300 356-1, *Integrated Services Digital Network*

(ISDN); Signaling System No. 7; ISDN User Part (ISUP) Version 2 for the International Interface; Part 1: Basic Services, 1999.

[14] IETF RFC 3398, *Integrated Services Digital Network (ISDN) User Part (ISUP) to Session Initiation Protocol (SIP) Mapping*, Dec. 2002.

[15] S.E. Griffn and C.C. Rackley, "Vishing," *Proc. Annual Conf. Inf. Security Curriculum Develop*., 2008, pp. 33–35.

[16] F. Maggi, "Are the Con Artists Back? A Preliminary Analysis of Modern Phone Frauds," *IEEE Int. Conf. Comput. Inf. Technol*., Bradford, UK, June 29–July 1, 2010, pp. 824–831.

[17] J. Quittek et al., "On Spam over Internet Telephony (SPIT) Prevention," *IEEE Commun. Mag*., vol. 46, no. 8, Aug. 2008, pp. 80–86.

[18] 3GPP TR 33.937, *Study of Mechanisms for Protection against Unsolicited Communication for IMS (PUCI)*, 2012.

[19] A. Tsakountakis, G. Kambourakis, and S. Gritzalis, "SIPA: Generic and Secure Accounting for SIP," *Security Commun. Netw*., vol. 5, no. 9, Sept. 2012, pp. 1006–1027.

[20] G. Karopoulos, G. Kambourakis, and S. Gritzalis, "PrivaSIP: Ad-hoc Identity Privacy in SIP," *Comput. Standard Interfaces J*., vol. 33, no. 3, Mar. 2011, pp. 301–314.

[21] X. Wang et al., "Voice Pharming Attack and the Trust of VoIP," *Proc. Int. Conf. Security Privacy Commun. Netw*., no. 24, 2008, pp. 1–11.

[22] M. Nassar et al., "Holistic VoIP Intrusion Detection and Prevention System," *Proc. Int. Conf. Principles, Syst. Appl. IP Telecommun*., 2007, pp. 1–9.

[23] 3GPP TS 23.014, *Support of Dual Tone Multi-frequency (DTMF) Signaling*, 2012.

[24] 3GPP TS 23.087, *User-to-User Signaling (UUS) Supplementary Service; Stage 2*, 2012.

[25] M. Roesch, "Snort - Lightweight Intrusion Detection for Networks," *Proc. USENIX Conf. Syst. Administration*, 1999, pp. 229–238.

[26] W. Lee and S.J. Stolfo, "Data Mining Approaches for Intrusion Detection," *Proc. USENIX Security Symp*., vol. 7, 1998, p. 6.

[27] R.A. Kemmerer and G. Vigna "Intrusion Detection: A Brief History and Overview," *Comput. Mag*., vol. 35, no. 4, Apr. 2002, pp. 27–30.

[28] J. Lennox et al., "Interworking Internet Telephony and Wireless Telecommunications Networks," *ACM SIGCOMM Comput. Commun. Rev*., vol. 31, no. 5, Oct. 2001, pp. 25–36.

[29] 3GPP TS 29.002, *Mobile Application Part (MAP) Specification*, 2011.

[30] IETF RFC 1034, *DOMAIN NAMES - Concepts and Facilities*, Nov. 1987.

[31] ITU-T E.721, *Network Grade of Service Parameters and Target Values for Circuit Switched Services in the Evolving ISDN*, 1999.

[32] S.A. Ahson and M. Ilyas, "Measurement and Analysis on Quality of Skype VoIP," *in VoIP Handbook: Applications, Technologies, Reliability, and Security*, *CRC Press, Inc*., Dec. 2008.

Jaeseung Song is an assistant professor in the Computer and Information Security Department at Sejong University, Seoul, Rep. of Korea. His research areas include IoT/M2M platforms, big data analytics, and the reliability and security of networked software systems. Prior to his current position, he worked for NEC Europe Ltd., Heidelberg, Germany, from 2012 to 2013, as a leading standard senior researcher. He also worked for LG Electronics, Seoul, Rep. of Korea, from 2002 to 2008, as a senior researcher. He received his PhD from Imperial College London, UK, in 2012. He received his BS and MS degrees in Computer Science from Sogang University, Seoul, Rep. of Korea, in 2000 and 2002, respectively.

Hyoungshick Kim is an assistant professor at Sungkyunkwan University, Suwon, Rep. of Korea. His research interests include security engineering, usable security, and social computing. He received his PhD in computer security from the University of Cambridge, UK, in 2012. After completing his PhD, he worked as a post-doctoral fellow at the University of British Columbia, Canada. He also worked at Samsung Electronics Co., Ltd., Suwon, Rep. of Korea, researching and developing technologies for trustworthy home networks.

Athanasios Gkelias received his PhD and MSc degrees from King's College London, UK, in 2005 and 2001, respectively, and his diploma in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2000. Currently, he is a post-doctoral researcher at Imperial College London, UK. In the past he served as the project manager of the University Defense Research Centre in Signal Processing at Imperial College, sponsored by the UK Ministry of Defence. He also participated in several funded ICT projects, such as Mobile-VCE, IBM-ITA, MEMBRANE, e-SENSE, and MIND. In the summer of 2008, he was at the Bell-Labs Research Centre, Alcatel-Lucent, UK, working as a visiting researcher on wireless mesh networks. He has published more than 40 peer-reviewed journal and conference papers, has been a TPC member, and is in the organizing committee of various international conferences.