Peer-to-Peer VoIP Communications Using Anonymisation Overlay Networks

Ge Zhang, Simone Fischer-Hübner

Peer-to-Peer VoIP Communications Using Anonymisation Overlay Networks*

Ge Zhang and Simone Fischer-Hübner

Computer Science Department,
Karlstad University, Universitetsgatan 2, 65188, Karlstad, Sweden
{ge.zhang, simone.fischer-huebner}@kau.se

Abstract. Nowadays, Voice over Internet Protocol (VoIP) which enables voice conversation remotely over packet switched networks gains much attentions for its low costs and flexible services. However, VoIP calling anonymity, particularly to withhold "who called whom", is difficult to achieve since VoIP infrastructures are usually deployed in an open networking environment (e.g., the Internet). Our work studies an anonymisation overlay network (AON) based solution to prevent surveillance from external attackers, who are able to wiretap the communication channels as well as to manipulate voice packets in the channels. However, it has been demonstrated that the VoIP combined with traditional AONs are vulnerable to two attacks, namely watermark attack and complementary matching attack. Taking these two attacks into account, we investigate the "defensive dropping" method in VoIP: A VoIP user-agent sends packets to an AON in a constant rate, but packets during periods of silence are marked. Then, the AON drops some silence packets and forwards the remaining ones to their destinations. The result of our experiments shows that the dropping rate must be carefully selected to counteract both of the two attacks. Finally, we discuss further threats in terms of this solution.

1 Introduction

VoIP is becoming increasingly popular due to its benefit on low-cost, flexibility and scalability compared to traditional telephony systems. However, VoIP infrastructures are usually deployed in large-scale packet-switched networks, such as the Internet, which is open in contrast to traditional Public Switched Telephone Network (PSTN). In this way, it is relatively easy for attackers to break into the VoIP networking environment and access the service infrastructures. On the other hand, generated voice packets sometimes have to be routed over some untrustful networking environment to reach their destinations. Therefore, VoIP services face much security and privacy challenges which have not occurred on traditional telephony systems.

* The authors would like to thank Stefan Köpsell and Stefan Berthold for their valuable comments and suggestions.

Like there are privacy requirements by users on other online applications, VoIP users may also request privacy protection. For instance, users may want to keep their conversation content secret, so that eavesdroppers on the communication channels are unable to intercept the content. This requirement can be realized by encrypting voice packets. Moreover, to prevent surveillance, users may sometimes want to make conversations anonymously so that potential attackers over the communication channels cannot find out "who called whom". To achieve this requirement, one potential solution is to employ anonymisation overlay networks (AON), which contain scalable relay proxies. An AON accepts k flows in and produces k flows out. The flows out are differently encoded than the flows in. Thus, it is difficult for attackers to relate between the flows and users. In practice, AONs are mostly used for email and web surfing, rather than VoIP. It is due to several specific features of VoIP, which make VoIP conversation on AONs vulnerable to several attacking methods: Attackers can embed a watermark into flows before they enter an AON by slightly delaying some randomly selected packets and then decode the watermark from the flows leaving the AON [18]. In this way, the attackers can correlate the flows separated by the AON to deduce "who called whom". Furthermore, VoIP users usually follow human conversation pattern, which means that one user speaks while another listens. Attackers also can exploit this pattern to pair flows for tracing [16]. This paper proposes a scheme based on the concept of defensive dropping [6] to prevent these two attacks: A VoIP user-agent sends voice packets to an AON for both speech and silence periods. However the packets during silence periods are marked and indicated by encrypted flag bits. When the AON receives the packets, it drops some silence packets and forwards the remaining ones to their destination. The dropping method makes the relationship between flows more difficult to be observed. The results of our simulation shows that this solution makes the two attacks more difficult, however the dropping rate must be carefully selected.

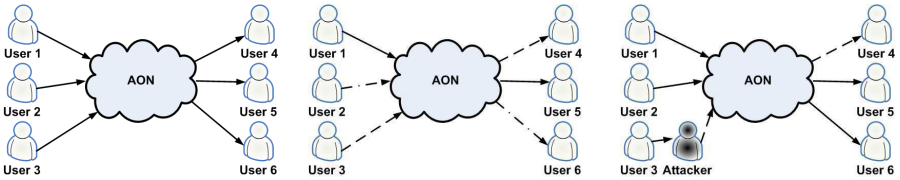
The rest of this paper is organized as follows. Section 2 introduces the background of VoIP and anonymisation overlay networks. Section 3 shows related work in VoIP anonymity. Section 4 proposes the solution and its evaluation. In Section 5, we discuss further security and anonymity issues regarding this solution. Finally, we summarize this paper and present future work in Section 6.

2 Background

2.1 VoIP, codec and silence suppression

VoIP applications typically use two kinds of protocols: a signaling protocol for call setup and termination (e.g., the Session Initiation Protocol (SIP) [11]) and a media delivery protocol for voice packets transmission (e.g., the Realtime Transport Protocol (RTP) [12]). To realize a VoIP conversation, both caller and callee should send encoded voice packets to each other, so called *bidirectional voice flows*. Transmitted voice packets are encoded and decoded using a speech codec algorithm (e.g., G.711 [1] and Speex [4]) negotiated in the signaling level. The

codec takes the voice from users as input, which is typically sampled at either 8k samples or 16k samples per second (Hz). As a performance requirement, the packet inter-arrival time of voice flow is usually fixedly selected between 10 and 50 ms, with 20 ms being the common case. Thus, given a 8kHZ voice source, we have 160 samples per packet with 20 ms packets interval. Moreover, the size of each voice packet depends on the encoding bit rate of adapted codec. Two types of encoding bit rates can be distinguished: **Fixed Bit Rate (FBR)** and **Variable Bit Rate (VBR)**. With FBR (e.g., G.711), end points produce voice packets always with the same size. On the other hand, VBR (e.g., Speex) means that the encoding bit rate varies according to the type of voice. In this way, user-agents produce voice packets with different size. Moreover, there are two kinds of voice packets: the packets generated during speech of a user, namely *speech packets* and the packets generated during silence of a user, namely *silence packets*. Some VoIP user-agents allow discontinuous voice packets transmission (**silence suppression**) [21], which is a capability of user-agents to stop sending silence packets during silent periods of its owner. In this circumstance, networking resources (e.g., bandwidth) can be significantly saved.



(a) No characteristic can be distinguished for flows

(b) Each traffic flow has its own characteristic

(c) An attacker injects characteristics into a flow

Fig. 1. Traffic analysis attacks on a VAS

2.2 Anonymisation Overlay Networks (AON)

AONs enable anonymous communications in untrustful networks. D. Chaum [8] firstly proposed a mix-based concept: Mix-networks contain scalable relay proxies that accept k incoming messages and forward them. The proxies change the order and appearance of messages so that it is difficult for an eavesdropper to distinguish which outgoing message corresponds to which incoming message. As a result, given a message, the attacker is unable to pair its communication partners. Generally, two kinds of AONs exist in reality: high-latency AONs for time insensitive applications (e.g., email) and low-latency AONs for time sensitive applications (e.g., web surfing). As VoIP applications have restrict requirements on real time, it is fairly clear that high-latency AONs are not suitable to protect VoIP anonymity. Thus, the AONs in the rest of this paper only refer low-latency AONs.

Traffic analysis attack aims to correlate the flows on both side of AONs by exploiting a certain characteristic (e.g., packet size or packet inter-arrival time) of the flows. A flow entering an AON and a flow leaving it can be related if they share similar characteristics. For example, an attacker cannot relate the flows in Figure 1(a) as all flows look similar. It is hard to say with whom user1 communicates (could be user4, user5, or user6). However, each flow has its own characteristics in Figure 1(b). In this case, attackers can easily relate the flows by their characteristics (user1 $\leftrightarrow$ user5, user2 $\leftrightarrow$ user4, user3 $\leftrightarrow$ user6). Instead of passively observing, Attackers can also modify a flow to insert more characteristics before it enters the AON (active attack), as illustrated in Figure 1(c).

Many proposed defending methods (e.g., reorder, packets-padding, dummy traffic and packets-delaying, etc) aim to eliminate (or generalize) the characteristics of flows to mitigate traffic analysis attacks. However, these methods usually introduce much overhead on networks and impact the service performance.

3 Related work

Many researchers have recently paid their attention on VoIP privacy. RFC3323 [10] and C. Shen, et al. [13] discussed the requirement of VoIP anonymity and proposed Trusted Third Party (TTP) based solutions for protecting sensitive information on signaling layer. M. Srivatsa, et al. [14] [15] proposed and compared several privacy-aware routers setup protocols for VoIP applications in P2P networking environments. C. V. Wright, et al. [20] [19] demonstrated that attackers can observe the used language or even the content of a conversation from the varying size of voice packets even if the packets are end-to-end encrypted. However, the scope of our paper is different to theirs: *our paper focuses on VoIP anonymity on media layer to hide who called whom from external attackers.*

C. A. Melchor et al [9] discussed three techniques providing strong traffic analysis resistant for VoIP media flow communications. The three techniques are global dummy traffic, broadcasting and private information retrieval respectively. They further evaluated the performance of these techniques based on a theoretical model. Nevertheless, these techniques are too costly to be deployed in reality.

X. Wang, et al. [18] investigated a practical solution in which users can make a skype [3] call over a commercial AON. Their test proved that VoIP over AONs can be practical. However, they also demonstrate that such a solution is still vulnerable to active timing attack: An attacker can embed unique watermark into the encrypted VoIP flow by slightly delaying of random selected packets. In this way, an attacker can find out who called whom by encoding and decoding watermarks on both side of an AON. O. Verscheure et al. [16] [17] proposed a method to reveal who called whom by exploiting the human conversation pattern: When one speaks, the other listens. This "alternate in speaking and silence" represents a basic rule of VoIP communication. Therefore, an attacker can easily pair the caller and callee by matching the bidirectional flows on signaling layer as long as silence suppression has been applied in the VoIP system. Nevertheless, neither

of them suggested corresponding countermeasures. In this paper, we address a potential defending solution to mitigate these two attacks.

4 VoIP anonymization using AONs

4.1 System model

User side: We limit user-agents so that they only support FBR codec instead of VBR codec. Thus, each generated voice packets should be of equal size. Otherwise, with VBR traffic analysis is easily possible. We further assume that user-agents provide an option for silence suppression. In this way, user-agents constantly produce voice packets (both speech and silence packets) if silence suppression is disabled. Otherwise, user-agents will stop generating speech packets. Finally, we assume that each voice packet is end-to-end encrypted (e.g., using SRTP).

AON: We assume that the anonymity service is provided by a fixed sequence of n proxies, which is similar to some existing implementations (e.g., An.on [7]). The AON supports bidirectional communications from the first one to the last one and vice versa. We treat the entire AON as a black box and will not focus on the system-specific details of the path establishment protocol between them. We assume that the signaling service can coordinate at least k pairs of users to establish (and terminate) voice flows over the AON simultaneously. We split the users into two groups: namely $\mathbb{L}$ and $\mathbb{R}$, where callers in $\mathbb{L}$ call callees in $\mathbb{R}$ over the AON, as illustrated in Figure 2. The k callers in $\mathbb{L}$ are indicated as $L_1, L_2, ..., L_k$ and the callees in $\mathbb{R}$ are referred as $R_1, R_2, ..., R_k$. However, readers should notice that the relationships between the users in the two groups are not deterministic: It does not mean that L_i certainly called R_i $(1 \leq i \leq k)$.

Voice flows: As introduced above, a VoIP conversation consists of 2 flows with opposite directions from two users. Considering the AON deployed in the middle, the 2 flows are splited into 4 flows. We use $\overrightarrow{L_iM}$ denotes the flow from L_i to the AON and $\overrightarrow{ML_i}$ indicates the flow from the AON to L_i $(1 \leq i \leq k)$. The same notation is applied for the flows between the AON and R_i.

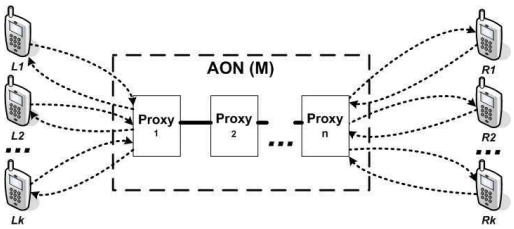


Fig. 2. System model: k pairs of users build VoIP conversations over an AON

4.2 Threat model

We assume that attackers are unable to control all the proxies in the AON. Thus, attackers treat the entire AON as a blackbox. We assume that attackers have the ability to observe and manipulate the flows entering and leaving the AON (e.g., the attacker is an Internet Service Provider (ISP)). However, we further assume that it is computational infeasible for the attackers to decrypt the voice packets in any flow. The attackers need to find out who called whom by relating the flows on both side of the AON. *For example, let us assume that* L_1 *called* R_2. To confirm this, the attackers should relate $\overrightarrow{L_1M}$ with $\overrightarrow{MR_2}$, $\overrightarrow{ML_1}$ with $\overrightarrow{MR_2}$, $\overrightarrow{ML_1}$ with $\overrightarrow{R_2M}$ or $\overrightarrow{L_1M}$ with $\overrightarrow{R_2M}$ by a certain kind of pattern. Two known threats on VoIP anonymity are the following ones:

- Watermark attack [18]: Suppose a flow contains continuous voice packets P_1, P_2, ..., P_n with time stamps t_1, t_2, ..., t_n. An attacker intercepts the flow before it enters the AON. She randomly selects 2 packets: P_i and P_j and extend to 4 packets in 2 groups (P_i, P_{i+d}) and (P_j, P_{j+d}). Then the attacker can obtain the inter-packet delay (IPD) for this two groups: $IPD_1 = t_{i+d} - t_i$, $IPD_2 = t_{j+d} - t_j$. Finally, the attacker can calculate the normalized difference $IPDD = (IPD_2 - IPD_1)/2$. If the VoIP user-agents send voice packets in a constant rate, the IPDD should be symmetric centered around 0. To insert watermark into the flow, the attacker can delay P_i and P_{j+d} for a small time interval α to embed bit '1' or delay P_{i+d} and P_j to embed bit '0'. The marked flow should be forwarded to the AON and the attacker can observe the marked flow on the other side: the IPDD distribution of selected packets should be shifted from 0 by α. [18] shows that only α=3 ms delay is enough to successfully embed watermark in Skype flows. Certainly, networking delay jitter can result in error bits. Nevertheless, the attacker can select more packets in a flow under attack to increase redundancy. With introducing only 3 ms delay, it is rather difficult for proxies of the AON to detect and synchronize the delayed packets. Considering the example shown above, the watermark attack can help the attacker to correlate either $\overrightarrow{L_1M}$ with $\overrightarrow{MR_2}$ or $\overrightarrow{ML_1}$ with $\overrightarrow{R_2M}$.
- Complementary matching attack [16]: Some VoIP user-agents apply codec with silence suppression in which the user-agents either generate small size packets or does not generate packets at all during the silence period. Moreover, human conversations usually follow the "alternate in speaking and silence" rule: When one speaks, the other listens. In this way, an attacker can detect silence or speech by observing a flow. For example, if the attacker has recorded all bidirectional flows during a given time T as set $\mathbb{F}$, she can select F_i and F_j as query flows. The attacker can calculate *the pairing index value* C:

$$C(i,j,T) = \sum_{t=1}^{T} \frac{XOR(F_i[t], F_j[t])}{T} \tag{1}$$

$F_x[t] \in 0, 1$, where 1 indicates that the flow F_x represents speech at time t and 0 indicates silence at time t. Thus, according to the "alternate in speaking and silence" rule, the higher of C(i,j,T), the higher probability that F_i and F_j belong to one conversation. [16] demonstrates the high accuracy provided by this method. Regarding the previous example, the complementary matching attack enables attackers to correlate either $\overrightarrow{ML_1}$ with $\overrightarrow{MR_2}$ or $\overrightarrow{L_1M}$ with $\overrightarrow{R_2M}$.

4.3 Defensive method

In this paper, our proposed defensive solution is fundamentally probabilistic. We aim to counteract against watermark attack [18] and complementary matching attack [16]. However, we do not aim to provide a solution to prevent long-term statistical attacks. We notice that the watermark attack correlates flows by exploiting normal distribution of IPDD while the complementary matching attack takes advantage of on-off flow pattern caused by silence suppression. As a result, we have a dilemma:

Dilemma 1 *If users apply silence suppression, they are vulnerable to the complementary matching attack; If users do not apply silence suppression, they are vulnerable to the watermark attack.*

If silence suppression is applied, the packet inter-arrival time for a given flow is not constant, which means the distribution of IPDD is indeterministic. This makes the watermark attack more difficult to success. However, in return, the on-off behavior on flows exposes the silence and speech period of users so that the complementary matching attack is easy to mount. On the other hand, if silence suppression is not applied, then the complementary matching attack does not work at all. Nevertheless, with constant packet inter-arrival time, the watermark attack is rather easy. Taking the dilemma into account, we propose a solution based on the "defensive dropping" concept [6]. The user-agents do not apply silence suppression but Voice Activity Detection (VAD): The user-agents generate voice packets to the AON in a constant rate whatever they detect silence or speech. However, the originator user-agent can instruct the proxies of the AON to drop some of the silence packets according to a dropping rate dr. This can be easily achieved by putting one bit ('0' for keeping and '1' for dropping) inside the encryption layer for each proxy. The proxy for dropping a silence packet is randomly selected. Dropping these silence packets introduces less negative impact on the performance of a VoIP conversation.

Theoretically, this solution can decrease the success ratio of both two attacks. Firstly, it weakens the timing relationship between flows entering the AON and flows leaving it. All flows entering the AON are with constant packet inter-arrival time, but all flows leaving the AON are with varying time characteristics. Furthermore, not all silence packets will be dropped so that the on-off behavior on flows are still unclear, which means the complementary matching attack is more difficult. Nevertheless, the amount of dropped packets depends on the

dropping rate dr. In extreme cases, either $dr = 0\%$ (no silence packets should be dropped) or $dr = 100\%$ (all silence packets should be dropped) violate our design. We demonstrate our simulation to show the impact of the value of dr to the two attacks in the next section.

4.4 Simulation

To find out whether the proposed solution can effectively prevent the attacks mentioned above and how the dr affects the result, we did a series of simulations.

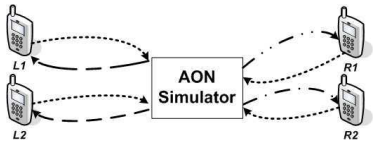


Fig. 3. The AON simulator accepts padded flows and produces simulated outgoing flows

Raw data preparation: We collected trace files recorded from 2 real VoIP conversations by using X-lite[5] with silence suppression enabled. There are 4 users (labeled as L_1, L_2, R_1 and R_2) participating in these 2 conversations, each of which elapsed for 10 minutes. We let L_1 *called* R_1 *and* L_2 *called* R_2. We collected the 4 voice flow traces originated by the 4 users respectively.

User-agents simulator: We then deliberately padded silence packets into each flow to normalize the packet inter-arrival time. In this way, the time intervals between any two adjacent packets are nearly equal. The padded flows simulate the 4 flows from user-agents to AON: $\overrightarrow{L_1M}$, $\overrightarrow{L_2M}$, $\overrightarrow{R_1M}$ and $\overrightarrow{R_2M}$. Furthermore, regarding n proxies of AON, each packet in these flows contains n 1-bit flags indicating whether such a packet should be dropped and which proxy should drop it. Only padded silence packets can be randomly selected for being dropped according to a dropping rate dr. Speech packets should not be selected.

AON simulator: We implemented a AON simulator using Perl language. The AON simulator accepts the 4 padded flows ($\overrightarrow{L_1M}$, $\overrightarrow{L_2M}$, $\overrightarrow{R_1M}$ and $\overrightarrow{R_2M}$), drops the packets marked by user-agents and generates the corresponding flows leaving the AON ($\overrightarrow{ML_1}$, $\overrightarrow{ML_2}$, $\overrightarrow{MR_1}$ and $\overrightarrow{MR_2}$). The AON simulator is based on an ideal environment that no delay, jitter occurs during the transmission.

Figure 3 shows an overview of our test. In this way, we use our simulators to simulate the flows entering the AON as well as the flows leaving it. Thus, there are 8 flows in total. We further simulate the attacks with varying dropping rates

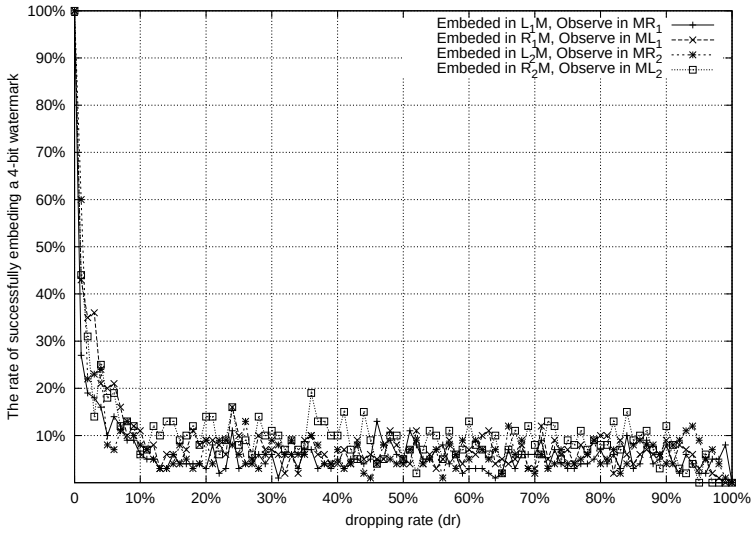


Fig. 4. The relationship between the dropping rate dr and the success ratio of watermark attack

(dr) on the 8 flows to investigate whether the attacks can still be successfully mounted.

In the first simulation, we investigate the watermark attack. We embed a 4-bit unique watermark with 50 redundancy for each flow entering the AON ($\overrightarrow{L_1M}$, $\overrightarrow{L_2M}$, $\overrightarrow{R_1M}$ and $\overrightarrow{R_2M}$). The packets for watermarking are randomly selected from the first 500 packets in each flow. We embed watermarks with a delay of only 10 ms ($\alpha = 10ms$). Then we use our AON simulator to simulate the corresponding flows leaving the AON ($\overrightarrow{ML_1}$, $\overrightarrow{ML_2}$, $\overrightarrow{MR_1}$ and $\overrightarrow{MR_2}$). We verify whether the same watermark can be recovered from these flows again.

We scale the dropping rate (dr) from 0% to 100%. The simulation for each dr is repeated for 100 times, and the result is shown in Figure 4. We find that with $dr = 0\%$, all the watermark can be successfully recovered. Thus, the success ratio of embedding a watermark is 100%. However, as long as $1\% \leq dr \leq 10\%$, the success ratio decreases significantly. Moreover, when $10\% \leq dr \leq 100\%$, the success ratio repeatedly varies between 0% and 20%.

In the second simulation, we focus on the complementary matching attack. As introduced above, the flows entering the AON ($\overrightarrow{L_1M}$, $\overrightarrow{L_2M}$, $\overrightarrow{R_1M}$ and $\overrightarrow{R_2M}$) are with a constant packet inter-arrival time. Thus, the on-off pattern of these flows are not distinguishable for attackers. As an alternative, attackers can still try to correlate the flows leaving the AON ($\overrightarrow{ML_1}$, $\overrightarrow{ML_2}$, $\overrightarrow{MR_1}$ and $\overrightarrow{MR_2}$), since some silence packets will be dropped for these flows. In this way, we use the Equation 1 to calculate the pairing index C for 4 pairs of flows: $\overrightarrow{ML_1}$ with $\overrightarrow{MR_1}$, $\overrightarrow{ML_1}$ with $\overrightarrow{MR_2}$, $\overrightarrow{ML_2}$ with $\overrightarrow{MR_1}$ and $\overrightarrow{ML_2}$ with $\overrightarrow{MR_2}$.

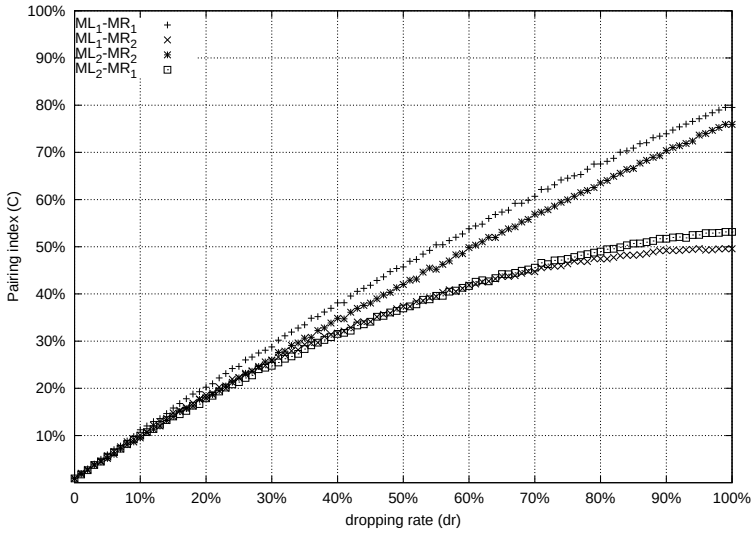


Fig. 5. The relationship between the dropping rate dr and the success ratio of complementary matching attack

The value of the pairing index C for each correlating attempt is plotted in Figure 5. We find that the pairing indices are low and nearly equal when the dropping rate dr is between 0% and 10%. The pairing indexes increase with the dr. However, the pairing index (C) of $\overrightarrow{ML_1}$ with $\overrightarrow{MR_1}$ and $\overrightarrow{ML_2}$ with $\overrightarrow{MR_2}$ increase significantly faster than the pairing index of $\overrightarrow{ML_1}$ with $\overrightarrow{MR_2}$ and $\overrightarrow{ML_2}$ with $\overrightarrow{MR_1}$. Especially when the $dr = 100\%$, the pairing index for $\overrightarrow{ML_1}$ with $\overrightarrow{MR_1}$ and $\overrightarrow{ML_2}$ with $\overrightarrow{MR_2}$ are around 80% and 75% respectively, which are distinguishable higher than the value of $\overrightarrow{ML_1}$ with $\overrightarrow{MR_2}$ and $\overrightarrow{ML_2}$ with $\overrightarrow{MR_1}$ (only 53% and 50% or so). In this case, it is easy to find out that L_1 called R_1 and L_2 called L_2.

The result of our experiments shows that the value of dropping rate (dr) should be carefully selected. If it is set too low, the solution will be still vulnerable to the watermark attacks. However, if it is set too high, the solution is vulnerable to the complementary matching attack. Fortunately, from the result shown in Figure 4 and Figure 5, we can find an optimal range for dr (e.g., from 5% to 10%), which is a trade-off point.

5 Further threats

This section discusses further classical attacks on this solution.

– Passive attack: The attackers observe the unique features between the flows entering and the flows leaving the AON to pair the flows. Our solution enables that all the flows entering the AON share the same features (e.g., the

equal packet size and packet inter-arrival time) and all the flows leaving the AON have different features (e.g., packet inter-arrival time) in a highly indeterministic manner. In this way, passive attack is not easy to launch.
- Replay attack: The attackers can record several packets from a flow entering the AON and replay these packets later to the AON. Our solution is unable to prevent this attack. Nevertheless, in future work, we consider to assign a time stamp for each generated voice packet. Voice conversation has a high requirement on the real time. Usually, a voice packet arriving too late, saying after 400 ms, is considered less useful [2]. In this way, we can arbitrarily set a timeout threshold to drop late-arrival packets to prevent replay attack.
- Short-term intersection attack: Given k pair of users are making conversations over an AON. Meanwhile, a new conversation is built over the AON between A and B, who are not belong to the k pair of users. Thus, it is easy to find out that A called B. We leave this problem to be solved by signaling services: the signaling service should coordinate a number of users to establish (and terminate) conversations over the AON simultaneously as a batch to achieve required anonymity set.

6 Conclusion and future work

Considering the unique properties of VoIP voice flows, we focus on an AON-based VoIP anonymity solution to defend against the attackers who are interested in "who called whom". In this paper, we especially consider two known attacking methods on VoIP anonymity: watermark attack, which encodes and decodes watermark on the flows of both sides of the AON by slightly packets delay; and complementary matching attack, which pairs flows by the on-off traffic mode and human conversation patterns. We propose a solution based on defensive dropping concept: The user-agent randomly marks a certain amount silence packets and these silence packets will be dropped by the AON. In this way, the relationship between corresponding flows can be obscured. The result of our simulation shows that the effectiveness of this solution highly depends on the dropping rate (dr). We finally discuss more potential attacks regarding this solution.

The scope of this paper is limited in the VoIP media layer. Nevertheless, media layer and signaling layer are usually viewed as a whole for VoIP applications. Thus, our future work will extend the scope to the anonymity protection on signaling layer. For example, how the signaling service can coordinate at least k pairs of users to establish (and terminate) voice flows over the AON simultaneously to achieve required anonymity set. Moreover, we are going to investigate the performance, usability aspects of this solution in the future.

References

1. G.711. http://www.itu.int/rec/T-REC-G.711/e, visited at 21th-Oct-2009.
2. ITU-T. Recommendation G.114 - One-way Transmission Time, 2003.
3. Skype. www.Skype.com, visited at 21th-Oct-2009.

4. Speex. http://www.speex.org/, visited at 21th-Oct-2009.
5. X-lite. http://www.counterpath.com/x-lite.html, visited at 15th-Nov-2009.
6. C. Wang B. N. Levine, M. K. Reiter and M. Wright. Timing attacks in low-latency MIX systems (extended abstract). In *Proceedings of the* 8^{th} *international conference on Financial Cryptography (FC '04)*, pages 251–265, Berlin, Heidelberg, 2004. Springer-Verlag.
7. O. Berthold, H. Federrath, and S. Köpsell. Web MIXes: a system for anonymous and unobservable Internet access. In *Proceedings of the 2001 International workshop on Designing Privacy Enhancing Technologies*, pages 115–129, New York, NY, USA, 2001. Springer-Verlag New York, Inc.
8. D. L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, 1981.
9. C. A. Melchor, Y. D., and J. Iguchi-Cartigny. Closed-circuit unobservable Voice over IP. In *Proceedings of the* 23^{rd} *Annual Computer Security Applications Conference (ACSAC '07)*, pages 119–128, Los Alamitos, CA, USA, 2007. IEEE Computer Society.
10. J. Peterson. A privacy mechanism for the Session Initiation Protocol (SIP), 2002. RFC 3323.
11. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol, 2002. RFC 3261.
12. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications, 2003. RFC 3550.
13. C. Shen and H. Schulzrinne. A VoIP privacy mechanism and its application in VoIP peering for voice service provider topology and identity hiding, 2008. Technical report.
14. M. Srivatsa, L. Liu, and A. Iyengar. Preserving Caller Anonymity in Voice-over-IP Networks. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP '08)*, pages 50–63, Washington, DC, USA, 2008. IEEE Computer Society.
15. M. Srivatsa, L. Liu, and A. Iyengar. Privacy in VoIP networks: A k-anonymity approach. In *Proceedings of the* 28^{th} *IEEE Conference on Computer Communication (INFOCOM '09)*, Washington, DC, USA, 2009. IEEE Computer Society.
16. O. Verscheure, M. Vlachos, A. Anagnostopoulos, P. Frossard, E. Bouillet, and P. S. Yu. Finding "Who is talking to whom" in VoIP networks via progressive stream clustering. In *Proceedings of the* 6^{th} *International Conference on Data Mining (ICDM '06)*, pages 667–677, Washington, DC, USA, 2006. IEEE Computer Society.
17. M. Vlachos, A. Anagnostopoulos, O. Verscheure, and P. S. Yu. Online pairing of VoIP conversations. *The VLDB Journal*, 18(1):77–98, 2009.
18. X. Wang, S. Chen, and S. Jajodia. Tracking anonymous peer-to-peer VoIP calls on the Internet. In *Proceedings of the* 12^{nd} *ACM conference on Computer and communications security (CCS '05)*, pages 81–91, New York, NY, USA, 2005. ACM.
19. C. V. Wright, L. Ballard, S. E. Coull, F. Monrose, and G. M. Masson. Spot me if you can: Uncovering spoken phrases in encrypted VoIP conversations. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP '08)*, pages 35–49, Washington, DC, USA, 2008. IEEE Computer Society.
20. C. V. Wright, L. Ballard, F. Monrose, and G. M. Masson. Language identification of encrypted VoIP traffic: Alejandra y Roberto or Alice and Bob? In *Proceedings of* 16^{th} *USENIX Security Symposium on USENIX Security Symposium (SS '07)*, pages 1–12, Berkeley, CA, USA, 2007. USENIX Association.
21. R. Zopf. Real-time Transport Protocol (RTP) payload for Comfort Noise (CN), 2002. RFC 3389.