Security Issues for P2P-based Voice- and Video-Streaming Applications

Jan Seedorf

NEC Laboratories Europe
Kurfuerstenanlage 36
69115 Heidelberg, Germany
jan.seedorf@nw.neclab.eu

Abstract. P2P computing offers a new interesting field for security researchers. Being highly distributed and lacking centralised, trusted entities for bootstrapping security mechanisms, these systems demand novel approaches for *decentralised* security solutions.
Recently, a new class of P2P-applications has arisen: P2P-based voice and video streaming systems. The properties of these novel applications impose new, interesting security challenges which have only been started to be addressed by researchers. This paper presents a summary of existing work in the area, derives and discusses open research problems, and finally outlines approaches towards potential solutions for securing P2P-based voice and video streaming applications.

1 Motivation

For many years, distributed computer systems have been dominated by the *client-server paradigm*. In recent years, however, a new paradigm appeared for distributed systems: *Peer-to-Peer (P2P) computing*. In networks based on this new paradigm, all entities are considered equal and provide equivalent services to other entities. At the same time, all entities can use services from all other participants of the network.

P2P computing offers a new interesting field for security researchers. Lacking centralised, trusted entities for bootstrapping security mechanisms, these systems demand novel approaches for *decentralised* security solutions. Lately, a new class of P2P-applications has arisen: P2P-based voice and video streaming systems. Examples for such systems are P2P-Voice-over-IP applications like Skype [32] or P2PSIP [30] as well as P2P-video-streaming applications like PPlive [22] or Zattoo [37]. We subsume these applications as *Real-Time Communication Applications* (*RTC-applications* for short).

RTC-applications have some important differences to other P2P-applications, e.g., file-sharing. These differences result in specific security requirements. For instance, users expect to reach a telephone callee within seconds, or to switch a TV-channel within milliseconds. In P2P-networks infiltrated by attackers (which can drop or misroute messages), it is challenging to meet these real-time requirements. In contrary, for filesharing it is perfectly acceptable for the user if it takes in the order of tens of seconds to start a download. Thus, the real-time nature of RTC-applications puts constraints on the maximum time a P2P-lookup and P2P data transmission may take. This enables attacks on the availability of P2P real-time communication applications by simply *delaying* messages. Hence, there is a need for security mechanisms which can not only guarantee P2P routing and lookups in the presence of attacker nodes but also within reasonable time.

Another key difference with respect to security is the kind of data stored in the P2P network. In P2P-VoIP applications, the binding of a user's identity and the current location of the user (e.g., his IP-address) is stored as a data item in the P2P network. Attackers can redirect telephony calls to themselves simply by forging this binding, e.g., by updating the data item for a particular target identity. Thus, impersonation attacks are a very serious threat for P2P VoIP applications. It is therefore necessary to develop decentralised solutions for cryptographic content protection for these applications. In contrary, for filesharing applications user impersonation attacks are not a threat.

Other application-specific challenges are for example the risk of content pollution in P2P video streaming systems [15] or the regulatory requirement for lawful interception in large-scale communication systems [28]. Thus, P2P-based voice and video streaming applications impose some novel security research challenges. While there exist quite a few works on securing P2P-networks in general, only very few papers consider the unique security challenges for running real-time communication applications over a P2P-substrate [12] [27].

The goals of this paper are the following:

- highlighting the unique security requirements for P2P VoIP and P2P Video streaming applications and show why existing, generic solutions for P2P security do not address these problems or are not applicable
- presenting the resulting research challenges, taking into account existing work in this area
- sketching approaches towards potential solutions, focusing on the specific characteristics of P2P-RTC-applications which can be used to develop decentralised security mechanisms

The rest of this paper is organised as follows. Section 2 provides a short overview on existing P2P VoIP and P2P video streaming systems. Section 3 discusses the corresponding security challenges for these types of P2P-based real-time communication applications. A survey on previous work to secure P2P VoIP and video streaming applications is presented in section 4. Section 5 highlights open research problems with respect to security and outlines potential approaches for addressing these problems. Finally, section 6 concludes the paper with a summary.

2 Brief Overview of P2P-based Voice and Video Applications

In general, P2P networks can be classified into *unstructured* and *structured* systems [19]. Early systems (e.g., Napster, Gnutella) used flooding for message routing in the network. Such *unstructured* systems cannot give any formal guarantees that a message in the network will reach its destination. Furthermore, broadcast messages impose an unnecessary traffic burden on the network.

To improve lookup time for a search request, so-called *Structured Overlay Networks*[1] have been proposed by researchers. These networks provide load balancing and efficient routing of messages. Structured P2P networks are based on *Distributed Hash Tables (DHTs)* [25] [34] and provide *key-based routing*: given a key (as a search request inserted by a participating node into the network), the network returns the node responsible for storing data belonging to that key. Routing as well as node responsibility for keys is based on a hash function: Node-IDs and key-IDs are computed using the same, system-wide hash function. The node which has the ID closest to a certain key-ID is responsible for storing data for this key-ID[2].

Today, unstructured P2P networks prevail on the Internet due to their simple routing algorithms and low overhead [19]. In addition, many types of applications (e.g., file-sharing) do not need the formal guarantees offered by structured P2P networks. DHT-based (i.e., structured) P2P systems are heavily studied by researchers and are slowly starting to appear as real applications on the Internet.

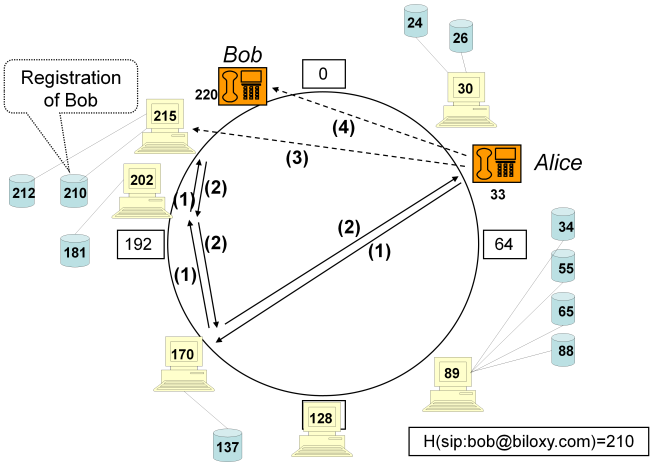

Fig. 1. P2PSIP: P2P-based VoIP Signalling

[1] In the literature, the terms *P2P network* and *overlay network* are used interchangeably. We follow this notation.

[2] For a detailed illustration on how DHTs work and a survey on the various DHT-algorithms which have been proposed the reader is referred to [19].

2.1 P2P-based IP Telephony and Video Telephony

A very famous P2P-based *VoIP* (Voice-over-IP) application is Skpye [32]. Skype's security model is based on two principles: a) a central login server and b) a *security-by-obscurity* approach regarding the communication protocol and peer software [6] (i.e., communication between peers is encrypted and the peer software is only available in binary code). Thus, researchers have analysed Skpye through (mostly passive) measurements [6] [10] in order to derive a view into Skype's interior functioning. In addition, reverse engineering of Skype's peer software has been performed in detail [8]. From a security research perspective, Skype is not suitable for investigating secure distributed algorithms for communication and data storage in P2P-VoIP systems due to its closed source code and security-by-obscurity paradigm.

As an alternative to Skype, researchers have proposed DHT-based P2P VoIP signalling with the Session Initiation Protocol (SIP [24]) [30]. This approach resembles an *open-specification* paradigm and is commonly referred to as *P2PSIP*. With P2PSIP, the binding of a user's identity (i.e., his SIP-URI) and his current location (i.e., IP-address and port) are stored in a DHT. The SIP-URI is the key and the current location of the callee is the corresponding data item stored at the responsible node in the DHT. Thus, the DHT can be used to obtain the current location of a desired callee. Technically, P2PSIP enables the establishment of any kind of real-time user-to-user multimedia session (e.g., voice, video, instant messaging).

Figure 1 [28] shows an example[3] where two users, Alice and Bob, have joined a Chord DHT [34]. Bob has registered his SIP-URI `bob@biloxy.com` with the P2P network by hashing his SIP-URI and storing his current location at the node responsible for `hash(SIP-URI)`. In the example, Bob's SIP-URI hashes to 210 and is stored at node 215 in the network because this node has the closest node-ID to the key-ID 210. If Alice wants to call Bob, she computes the key-ID for Bob's URI by hashing his SIP-URI and inserts a `key-lookup(210)` query into the DHT. The lookup request gets routed through the P2P network (1) and finally returns the IP-address and port of the node responsible for the requested content (2). Alice can then contact that node (i.e., 215) directly, without using the DHT, to receive Bob's location (3). Finally, Alice can call Bob (4) using regular SIP signalling.

Notably, the P2P network is only used for signalling. The actual media transfer of (normally) RTP[4] packets is taking place directly between the users and is not routed over the P2P network. At the time of this writing, several independent prototype implementations of P2PSIP exist [31] [4]. Furthermore, there is a P2PSIP working group in the IETF which has the goal of standardising a protocol for serverless VoIP signalling with SIP [21].

[3] Throughout this paper, we exemplify structured P2P networks and attacks on these networks with Chord [34]. Similar attacks are possible if other DHTs are used as the P2P structure, though technically slightly different due to the algorithmic details of the varying DHT-routing schemes proposed in the literature.

[4] Real-Time Transport Protocol

2.2 P2P IPTV and Live Video Streaming

In contrast to P2P VoIP applications, P2P IPTV and live video streaming applications are characterised by the fact that the actual media stream is transmitted over the P2P-overlay. This is the main reason why these systems are usually based on unstructured P2P networks: The requirements of video streaming for low delay and high, constant bandwidth render the overhead of structured P2P networks infeasible. Instead, participating nodes form a so-called swarm in an unstructured network, and exchange only a few, single-hop signalling messages. The video stream is split into so-called *chunks* which are exchanged among peers. Depending on the system, peers can either push chunks to their neighboring peers (i.e., the ones to which they have a direct link in the P2P topology) or pull chunks from their neighboring peers. Researchers are currently investigating the effect of different such chunk scheduling strategies on overlay topology and overall system performance (e.g., [1] [9]).

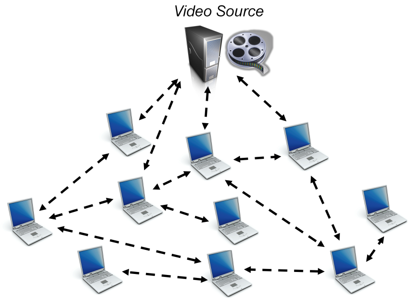

Fig. 2. P2P Live Video Streaming

Figure 2 shows a schematic view of a generic P2P-based video streaming system. A source node splits the video stream into chunks and distributes these chunks to initial peers in the overlay. Then chunks are exchanged among peers according to a chunk scheduling strategy (push/pull). Depending on the concrete strategy, peers may query their neighbors for certain chunks (in the case of a pulling system) or offer their neighbors currently buffered chunks (in the case of a push system)[5].

Examples for popular systems are PPLive [22], SOPCast [33], TVants [35], Zattoo [37], or TVUPlayer [36]. In addition, high-definition (HD) P2P streaming (i.e., up to 10 Mbit/s) is expected to be a reality in the near future. Already, several commercial P2P video streaming applications have launched tests for HD P2P-TV, e.g., Babelgum [2] or Zattoo [37].

[5] For an in-depth tutorial on P2P live streaming systems the reader is referred to [17].

3 Security Challenges for P2P-based Voice- and Video-Streaming Applications

Due to the dynamic nature of P2P systems (e.g., nodes can join and leave the network frequently) and the lack of central entities on routing paths, many existing security solutions are not (or at least not directly) applicable to P2P networks. In principle, nodes in a P2P network must be regarded as not trustworthy and attacker nodes may drop, modify, or misroute messages.

The security of P2P systems has been studied by researchers, mostly considering file-sharing as the prototypical P2P application. Real-time communication applications, however, impose additional challenges which have not received a lot of attention in the literature. These specific challenges are due to the real-time requirements, the type of data stored in the network, privacy considerations, and the risk of unsolicited communication. In addition, VoIP applications in particular have the regulatory requirements of lawful interception and emergency calls. In this section we discuss these challenges in detail.

3.1 Authentication

Douceur has shown that in any distributed system without some form of a central entity for identity assertion, attackers are able to create virtual, fake identities [16]. Thus, decentralised authentication of participating nodes is a challenge for any P2P application. For P2P-VoIP, the authentication of user-identities (i.e., SIP-URIs) without a central authority which is trusted by all nodes in the system (e.g., a root certificate authority) is an additional problem. Authentication and encryption of real-time streams, on the other hand, is technically straightforward with SRTP [5]. Thus, confidentiality of media streams is not a challenge but rather that proper mechanisms for authentication and key exchange exist.

3.2 Real-time Communication Availability Requirements

Real-time communication applications demand low delay and in the case of video additionally high, constant bandwidth. These characteristics make RTC applications more susceptible to attacks on availability than other P2P applications. Attackers can severely degrade services simply by dropping or delaying messages transmitted over the P2P network, even if this gets eventually detected by the sender and messages are re-sent over a different path. The simplest case are so-called *free-riders*: nodes that consume services offered by other nodes but which do not themselves contribute services to the P2P network. In addition to such *passive* attacks, *active* attackers can absorb even more resources from the P2P routing layer in vain by forwarding messages wrongly or with false content, potentially delaying the detection and retransmission of messages even further.

For VoIP, such behaviour attacks the availability to reach desired callees. For instance, it has been shown that even a small amount of free-riders in a P2PSIP network can significantly decrease the ability for calls to reach the callee [29].

For live video streaming, not enough chunks might arrive at users' clients so that either the video cannot be played at all or with less quality (in the case of codecs which can compensate lost chunks by playing the video stream at a lower bitrate), degrading the overall quality of the application for users [17].

3.3 Spam and Content Pollution Attacks

Unsolicited communications (e.g., *Spam over IP Telephony*, *SPIT*) is a bigger threat for real-time communications than for other P2P-applications because it immediately disturbs the user with a ringing phone or potentially offending video content. SPIT is expected to become a threat in VoIP networks in the near future [23] due to lacking authentication mechanisms, low costs for call initiations, and vulnerable devices. Following this reasoning, SPIT can be expected to be an even bigger threat for P2P-based VoIP because similar assumptions hold but authentication is likely to be even more difficult in a P2P scenario. More importantly, many mechanisms proposed by researchers against SPIT (e.g., [3] [23]) rely on central entities (i.e., signalling servers) on the routing path. This property renders such mechanisms not very useful in a P2P setting: Due to the dynamic membership in P2P networks routing paths change frequently and messages for a certain callee do not necessarily traverse a specific path on which protecting services can be deployed. Additionally, spammers can harvest target locations in a straightforward manner by simply trying lookup requests in the DHT consecutively over the co-domain of the hashfunction used.

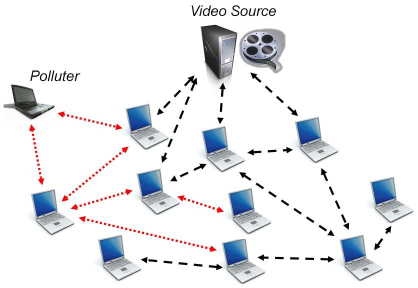

Fig. 3. Pollution Attack in a P2P Live Video Streaming System

A related threat for video streaming systems are so-called content pollution attacks, where attackers forward unsolicited chunks to other peers. Figure 3 [15] shows a P2P network where a malicious node sends bogus chunks to other peers, falsely marking these chunks as legitimate. These corrupted chunks get propagated through the P2P network, thereby not only degrading performance of the polluter's direct neighbors but also of other peers. One of the few works where such attacks have been studied is [15]. The results show that content pollution attacks can severely impact the quality-of-service in P2P live video streaming systems.

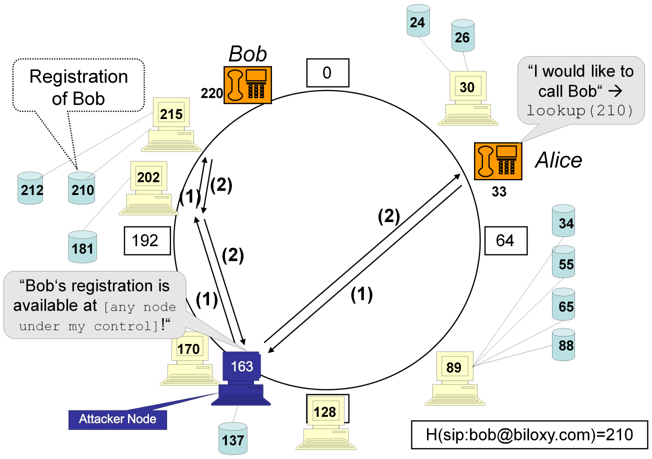
Registration of Bob
Bob
220
0
24
26
30
“I would like to call Bob“ → lookup(210)
Alice
33
215
202
212
210
(1)
(2)
(2)
192
64
34
55
65
88
181
(2)
(1)
“Bob‘s registration is available at [any node under my control]!“
(1)
170
163
89
128
Attacker Node
137
H(sip:bob@biloxy.com)=210

Fig. 4. Man-in-the-Middle Attack in a P2PSIP System

3.4 Integrity of Data Items and P2P Messages

With P2PSIP, SIP registrations, i.e., the binding of an identity with a current user location, are stored in the P2P network. Attackers can impersonate a target identity by redirecting calls through forged P2P messages. Figure 4 exemplifies such an attack. In this example, an attacker has joined the DHT with node-ID 163 which replaces node 170 on the routing path between Alice and the node responsible for storing Bob’s SIP registration (compare with figure 1). The attacker can modify Bob’s SIP registration before forwarding the result of the DHT-lookup to Alice, thereby redirecting Alice to any node under his control. Similar attacks are possible during storing of data items in the DHT or in the case that the attacker node is actually responsible for storing SIP registrations itself[6]. To prevent such man-in-the-middle attacks on P2P messages, the integrity of data items which are stored in the P2P network must be cryptographically protected. It is, however, unrealistic to assume that all P2P nodes can agree on a centralised PKI-like certification infrastructure.

Likewise, without integrity protection attackers can modify chunks in P2P live video streaming systems. This can result in content pollution attacks with advertisements or offending content. However, in systems with a single or few sources it may be reasonable to assume that peers can obtain a certificate of the source (e.g., through an out-of-band channel).

3.5 Privacy Concerns

In any P2P-based real-time application, participating nodes forward signalling messages. For P2P-based VoIP systems, any node on the routing path can log who tried to call whom. Since P2P nodes cannot be considered trustworthy and

[6] For instance, note that in figure 4 node 163 has also taken over the responsibility for key-ID 137 in the DHT routing structure (compare with figure 1).

routing paths change frequently (due to the joining and leaving of nodes, e.g., compare figure 1 with figure 4) important privacy concerns arise for callers as well as for callees. In contrary to centralised VoIP where there is a trust relationship between the user and its provider (and in many countries also legal obligations for the provider to observe user privacy), users may not feel comfortable that unknown nodes route and log their call establishments. Analogically, neighboring nodes in a P2P live video streaming system can detect which channel a certain node is watching by observing chunk exchanges with this node.

3.6 Lawful Interception and Emergency Calls

In most countries laws and regulation require the technical possibility for *Lawful Interception* of voice (or video) communications between two (or more) users[7], i.e., the process of legally authorised wiretapping of communications carried out by law enforcement organisations [28]. Lawful Interception of real-time communications over IP networks is technically challenging compared to other, traditional telephone networks [27]. Using a P2P network for VoIP signalling complicates Lawful Interception even more, mostly due to the lack of a central entity which is on every signalling path for a target identity. The implications of applying the P2P paradigm to VoIP signalling for Lawful Interception have been analysed in [28]. In summary, the lack of a central entity, changing participants and varying data responsibility, P2P-routing characteristics, and the non-trustworthiness of nodes make Lawful Interception in P2P-VoIP networks technically very challenging if not impossible at all. Figure 5 [28] illustrates in an example how the routing characteristics in P2P networks complicate Lawful Interception: In this figure, Bob is trying to call Alice (assuming that her SIP-URI hashes to 55). Note that the routing path is completely disjoint from the case where Alice calls Bob (compare with figure 1). For Lawful Interception this means that incoming and outgoing phone calls for a certain target identity cannot be intercepted at a single point in the network.

Similar to Lawful Interception, the possibility to make emergency calls may be a regulatory requirement for P2P-based VoIP services. However, it may not be possible to enforce prioritisation of signalling messages for emergency calls in a P2P scenario because it cannot be assumed that all participating nodes are trustworthy and thus handle specially marked messages accordingly. In addition, securely determining the physical location of VoIP users in real-time may be difficult in an infiltrated network since attackers can drop, delay, modify, or misroute messages [27].

4 Existing Security Approaches

A considerable amount of research has been done regarding P2P security in general. Specifically for structured, DHT-based overlays, several mechanisms for

[7] The precise requirements for Lawful Interception vary from country to country and may depend on the size and business model of an operator that provides voice/video services.

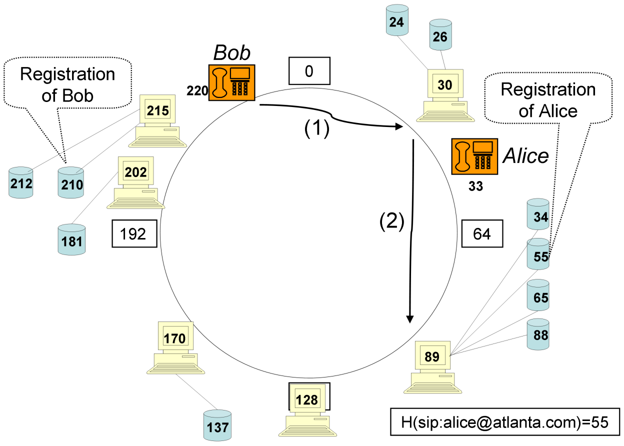

Fig. 5. Lawful Interception in P2PSIP Systems: Different Routing Path for Incoming and Outgoing Calls

secure node-ID assignment and for securing DHT routing in the presence of attackers have been proposed in the literature (e.g., [11] [13] [14]). While all this work is applicable to P2P-networks for real-time communications, it does not consider the specific challenges for RTC-applications in particular (compare 3).

In [27], the specific security challenges for P2PSIP as a DHT-application are analysed. Potential solutions sketched on a high level are non-scalable security add-ons (i.e., centralised solutions), distributed trust and reputation systems, and approaches which use self-certifying data items ([26], see below) [27]. In addition, secure routing techniques [14] are suggested to increase lookup availability in the presence of free-riders and active attackers in the P2P network. Furthermore, a (MIX-like) pseudonimity service is envisioned to protect the privacy of callers. Similarly, Chopra et al. provide a survey on security aspects for P2P-based VoIP applications and briefly discuss potential solutions on a general level [12]. Besides the approaches already mentioned in [27], they regard a PGP-like web-of-trust among users and SIP end-to-end encryption as promising approaches but do not present detailed algorithms. Also, the IETF P2PSIP working group [21] is considering security and has some initial, but immature proposals for end-to-end encryption and secure node-ID assignment [18].

As a concrete solution specifically targeted at P2P-based VoIP systems, self-certifying SIP-URIs have been proposed to protect the integrity of data items (i.e., SIP-URI/location-bindings) stored in P2PSIP networks [26]. Self-certifying identities are identities where the ownership of an identity can be verified without relying on a trusted third party (such as a certification authority). Technically, self-certifying identities are created (at least in part) as the hash of a public key. The owner of the identity can use the corresponding private key for signing messages or in general for proving the ownership of the identity. For P2PSIP, self-certifying SIP-URIs have the advantage that users can cryptographically protect their location bindings which are stored in the DHT while relying on a completely decentralised solution for verification of such signatures [26].

A related approach for secure identity assertions in P2PSIP networks called *P2PNS* has been presented by Baumgart [7]. P2PNS proposes a two-stage name resolution which uses a static cryptographic node-ID per user. The mappings of SIP-URI to node-ID and from node-ID to current location are stored in a DHT (since the first mapping is static, it can be cached). Similar to self-certifying SIP-URIs [26], this approach associates a public/private key pair statically with a SIP-URI.

The first study of an actual implementation of secure routing algorithms in a P2PSIP prototype has been presented in [29]. The results of this study show that even a small amount of attackers can significantly delay call setup times for users in a P2PSIP network. In addition, the conducted experiments demonstrate the principle effectiveness of the proposed security algorithms. Overall, this work points out the need for further research regarding secure DHT routing algorithms suitable to fulfill real-time communication requirements.

There is only very few research on malicious behaviour in P2P live streaming systems. Presumably, this is due to the fact that these systems are fairly new and have only been started to be analysed in the literature. In [17], security issues for such systems are highlighted and briefly discussed. Dhungel et al. investigate content pollution attacks in a concrete P2P live streaming system [15]. Their results show that such attacks can significantly degrade the availability in P2P live streaming systems. As potential countermeasures, the authors investigate blacklisting of polluters, encryption of the stream, hash verification, and signing of chunks by the source. In conclusion, chunk signing seems to be the most effective countermeasure against content pollution attacks.

In summary, only a few proposals exist for securing P2P-based VoIP systems. Further, existing work in this area is lacking concrete experiments and measurements which show the applicability and scalability to real P2P environments. More importantly, existing work does not address some of the crucial issues described in the previous section (section 3): To the best of our knowledge, there are no concrete research proposals for Lawful Interception, emergency calls, SPIT prevention, and privacy concerns specifically targeted at P2P-based VoIP systems. Even less work exists in the area of securing P2P live video streaming systems. This is worrying given the fact that these systems are highly popular and user communities for P2P video streaming are increasing at fast pace.

5 Open Research Problems and Outline for Potential Solutions

We have shown that P2P-based voice and video applications impose some interesting, new security challenges which have only been sparsely addressed by the security research community so far. Specifically, with respect to existing work, we regard the following as open research problems and interesting future work for security researchers:

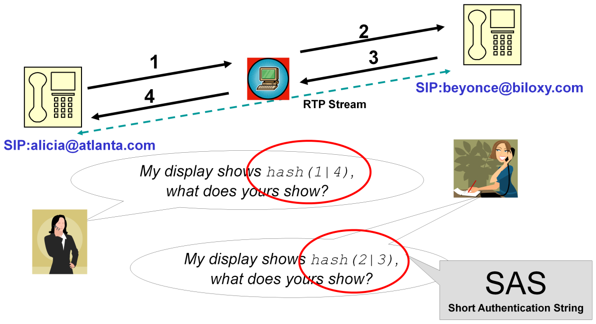
1
2
3
4
SIP:beyonce@biloxy.com
RTP Stream
SIP:alicia@atlanta.com
My display shows hash(1|4), what does yours show?
My display shows hash(2|3), what does yours show?
SAS
Short Authentication String

Fig. 6. Application-intrinsic End-to-End Authentication with ZRTP

- Developing P2P routing algorithms which can handle malicious nodes on the routing path and still guarantee RTC application requirements (i.e., low delay, high bandwidth, constant bandwidth)
- Investigating decentralised solutions (i.e., mechanisms which do not depend on a signalling entity which is on each routing path) for detecting and preventing unsolicited communications such as SPIT or content pollution attacks
- Investigating architectures for distributed user authentication in P2P networks with user-to-user real-time communication (e.g., VoIP or Video Telephony)
- Finding solutions for Lawful Interception in serverless VoIP networks like P2PSIP
- Enabling reliable, secure, and prioritised emergency calls in P2P-based VoIP systems with malicious nodes
- Designing distributed algorithms for preserving privacy on the P2P routing layer for P2P-based VoIP and live video streaming systems

Besides their unique threats and security considerations, real time communication applications also offer some special characteristics that could be exploited to bootstrap innovative and decentralised security mechanisms. For instance, in the case of direct real-time communication between two or more users, these users are able to recognize their communication partner by using the application itself (this property is inherent with any voice or video application where users *hear* or *see* a communication partner). This fact can potentially be beneficial for decentralised authentication or reputation systems in applications with user-to-user real-time communication (e.g., VoIP or direct video calls between users).

ZRTP [38] is a good example how such *application-intrinsic* authentication can be exploited to secure a communication channel. ZRTP enables a Diffie-Hellman key exchange over an RTP media stream. However, the key exchange is protected against man-in-the-middle attacks through a short authentication string (SAS)[8]. This short authentication string is displayed to the users which can compare it (thereby detecting attacks) by reading it over the RTP stream. Thus, an attacker would need to forge the voice of users in real-time in order to still launch an undetected man-in-the-middle attack on the key exchange. Figure 6 shows an example of a key exchange with ZRTP where two users compare the SAS (which is displayed in their VoIP client) over a voice media stream. If the SAS is the same on both sides the key exchange was successful and secure. In this way, ZRTP offers authentication of a known communication partner (and key exchange) without using any certificate infrastructure.

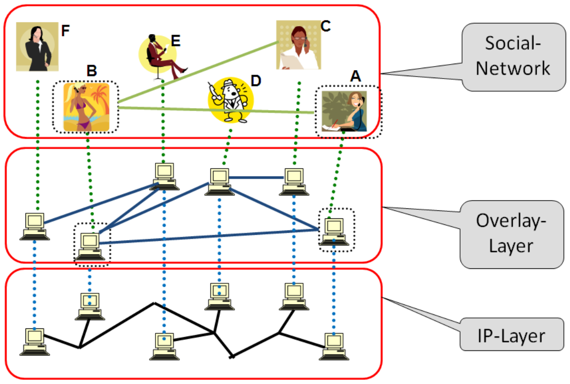

Fig. 7. Application-intrinsic Social Network in a P2P-based VoIP System

Further, often times social relationships exist between real-time communication partners. For instance, some VoIP users may frequently call known friends or family members. Researchers have started to consider the use of such social networks for security. In the field of P2P computing, Marti et al. have proposed to modify DHT routing such that messages are routed with higher probability to users with whom the user of the P2P-application has a link in a social network [20]. However, their approach maps the links in an external, web-based social network to P2P-nodes. For real-time communications among users, it may be possible to create such social links based on the interaction among users within the P2P application itself (e.g., VoIP calls). In addition, cryptographic key exchange (using ZRTP or similar solutions) with communication partners based on the social relationships among users could constitute an anchor for bootstrapping more complex security architectures. Furthermore, such *application-intrinsic* social networks could also be used to mitigate content pollution or spamming attacks by forming a distributed reputation system among socially connected users.

Figure 7 exemplarily shows the social relationships among users in a P2P-VoIP application. While some users have a direct relationship on the overlay

[8] Technically, the SAS is part of the hash of the parameters exchanged in the Diffie-Hellman key exchange.

layer and in the social network (e.g., users A and B), other users are directly connected in the social network but do not share a direct link in the P2P network (e.g., users B and C). In particular, users B and C might be able to exchange cryptographic keys securely and establish an encrypted connection over the P2P layer even though every message between them gets sent over the untrusted (and potentially malicious) user D.

It is, however, an open research problem how these properties, i.e., application-intrinsic authentication and social relationships among users, can precisely be exploited and combined to bootstrap more sophisticated decentralised security mechanisms.

6 Conclusion

In this paper, we looked at the security issues in P2P-based voice and video applications. In particular, we discussed threats that either impose a significant challenge due to the real-time requirements of such applications or are in other ways specific to these kinds of applications. We surveyed existing work, showing a discrepancy between the popularity regarding these applications on the one hand and investigations of corresponding security solutions by the research community on the other hand. Based on this analysis, we highlighted open research problems in this area, which include secure routing, prevention of unsolicited communication, emergency calls, distributed user authentication, and privacy considerations. As an outline for future work, we identified certain properties of P2P real-time communications which in principle enable approaches towards decentralised authentication and bootstrapping of distributed security mechanisms.

In summary, our study shows that security in P2P-based voice and video applications has not received enough attention in the research community. This is worrying, as more and more real-world applications appear on the Internet. P2PSIP is currently being standardised in the IETF and P2P live video streaming applications are offering services to an increasingly large audience. Thus, there is a need for further investigations, addressing the open problems we identified in this study and striving to develop innovative, decentralised solutions for real-time communication applications over P2P networks.

Acknowledgements

This work was partially supported by NAPA-WINE, a research project supported by the European Commission under its 7th Framework Program (contract no. 214412). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the NAPA-WINE project or the European Commission.

References

1. L. Abeni, C. Kiraly, R. Lo Cigno: *On the Optimal Scheduling of Streaming Applications in Unstructured Meshes*, ifip Networking 2009, May 2009
2. *Babelgum*, `http://www.babelgum.com`
3. V. A. Balasubramaniyan, M. Ahamad, and H. Park: *CallRank: Combating SPIT Using Call Duration, Social Networks and Global Reputation*, in CEAS 2007 Fourth Conference on Email and AntiSpam, 2007
4. S. Baset: *P2PP prototype implementation*, `http://www1.cs.columbia.edu/~salman/peer/`
5. M. Baugher, D. McGrew, M. Naslund, E. Carrara, K. Norrman: *The Secure Real-time Transport Protocol (SRTP)*, RFC 3711 (Draft Standard), March 2004, `http://www.ietf.org/rfc/rfc3711.txt`
6. S. Baset, H. Schulzrinne: *An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol*, Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM 2006), April 2006
7. I. Baumgart: *P2PNS: A Secure Distributed Name Service for P2PSIP*, Proceedings of the 5th IEEE International Workshop on Mobile Peer-to-Peer Computing (MP2P'08) in conjunction with IEEE PerCom'08, Hong Kong, China, p. 480-485, March 2008
8. P. Biondi, F. Desclaux: *Silver Needle in the Skype*, BlackHat Europe 2006, March 2006, available at `http://www.blackhat.com/presentations/bh-europe-06/bh-eu-06-biondi/bh-eu-06-biondi-up.pdf`
9. T. Bonald, L. Massoulie, F. Mathieu, D. Perino, A. Twigg: *Epidemic live streaming: optimal performance trade-offs*, International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS), 2008
10. D. Bonfiglio, M. Mellia, M. Meo, D. Rossi, P. Tofanelli: *Revealing skype traffic: when randomness plays with you*, Proceedings of SIGCOMM 2007, 2007
11. M. Castro. P. Druschel, A. Ganesh, A. Rowstron, D. S. Wallach: *Secure routing for structured peer-to-peer overlay networks*, Proc. of the 5th Symposium on Operating Systems Design and Implementation, Boston, MA, December 2002, ACM Press
12. D. Chopra, H. Schulzrinne, E. Marocco, E. Ivov: *Peer-to-Peer Overlays for Real-Time Communication: Security Issues and Solutions*, IEEE Communications Surveys & Tutorials, Vol.11, No.1, January 2009
13. T. Condie, V. Kacholia, S. Sankararaman, P. Maniatis, J. M. Hellerstein: *Maelstrom: Churn as Shelter*, University of California at Berkeley Technical Report No. UCB/EECS-2005-11, November 2005
14. G. Danezis, C. Lesniewski-Laas, M. F. Kaashoek, R. Anderson: *Sybil resistant DHT routing*, 10th European Symposium on Research in Computer Security, Milan, Italy, September 12-14, 2005, LNCS, Vol. 3679, Springer
15. P. Dhungel, X. Hei, K. W. Ross, N. Saxena: *The pollution attack in P2P live video streaming: measurement results and defenses*, Proceedings of the 2007 Workshop on Peer-to-peer Streaming and IPTV (2007), pp. 323-328
16. J. R. Douceur: *The sybil attack*, Revised Papers from the First International Workshop on Peer-to-Peer Systems, Cambridge, MA (USA), March 2002, LNCS, Vol. 2429, Springer
17. X. Hei, Y. Liu, K. Ross: *IPTV over P2P streaming networks: the mesh-pull approach*, IEEE JCommunications Magazine, Vol. 46, No. 2, pp. 86-92, February 2008

18. C. Jennings, B. Lowekamp, E. Rescorla, S. Baset, H. Schulzrinne: *REsource LOcation And Discovery (RELOAD) Base Protocol*, draft-ietf-p2psip-base-02, internet draft, (work in progress), http://tools.ietf.org/html/draft-ietf-p2psip-base
19. E. K. Lua, J. Crowcroft, M. Pias, R. Sharma and S. Lim: *A Survey and Comparison of Peer-to-Peer Overlay Network Schemes*, IEEE Communications Surveys and Tutorials, Vol. 7, No. 2, 2005, pp. 72-93
20. S. Marti, P. Ganesan, H. Garcia-Molina: *DHT Routing Using Social Links*, 3rd International Workshop on Peer-to-Peer Systems (IPTPS), 2004
21. P2PSIP Status Pages: *Peer-to-Peer Session Initiation Protocol (Active WG)*, http://tools.ietf.org/wg/p2psip/
22. *PPLive*, http://www.pplive.com
23. J. Quittek, S. Niccolini, S. Tartarelli, and R. Schlegel: *On Spam over Internet Telephony (SPIT) Prevention*, IEEE Communications Magazine, Vol. 22, No. 5, 2008
24. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler: *SIP: Session Initiation Protocol*, RFC 3261, 2002
25. A. Rowstron, P. Druschel: *Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems*, Proc. of the 18th IFIP/ACM International Conference on Distributed Systems Platforms, Heidelberg, Germany, November 2001
26. J. Seedorf: *Using Cryptographically Generated SIP-URIs to Protect the Integrity of Content in P2P-SIP*, 3rd Annual VoIP Security Wksp., Berlin, Germany, June 2006
27. J. Seedorf: *Security Challenges for P2P-SIP*, IEEE Network Special Issue on Securing Voice over IP, Vol. 20, No. 5, pp. 38-45, September 2006
28. J. Seedorf: *Lawful Interception in P2P-Based VoIP Systems*, IPTComm 2008, LNCS 5310, pp. 217-235, July 2008
29. J. Seedorf, F. Ruwolt, M. Stiemerling, S. Niccolini: *Evaluating P2PSIP under Attack: An Emulative Study*, IEEE Globecom 2008, November 2008
30. K. Singh, H. Schulzrinne: *Peer-to-Peer Internet Telephony using SIP*, Int. Wksp. on Network and Operating Systems Support for Digital Audio and Video, pp. 63-68, 2005
31. SIPDHT: http://sipdht.sourceforge.net
32. *Skype: Make the most of Skype - free internet calls and great value calls*, http://www.skype.com
33. *SopCast - Free P2P internet TV — live football, NBA, cricket*, http://www.sopcast.com
34. I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, H. Balakrishnan: *Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications*, IEEE/ACM Transactions on Networking, Vol. 11, No. 1, February 2003, IEEE Press
35. *TVANTS p2p*, http://www.tvants.com
36. *TVU Networks*, http://www.tvunetworks.com
37. *Zattoo: TV meets PC*, http://www.zattoo.com
38. P. Zimmermann, A. Johnston, J. Callas: *ZRTP: Media Path Key Agreement for Secure RTP*, Internet-Draft (work in progress), February 2009