International Journal of Computer Network and Security(IJCNS)
Vol 4. No 1. Jan-Mar 2012 ISSN: 0975-8283
www.ijcns.com

Privacy Preserving of VoIP against Peer-to-Peer Network Attacks And Defense

K. Bharathkumar [1], R. Premalatha Kanikannan [2], Dr.Rajeswari Mukesh[3],M. Kasiselvi [4],T. Kumanan[5].

[1]PG Student, Department of Computer Science and Engineering, Meenakshi College of Engineering, Chennai
visitkbh@yahoo.com
[2]Director, Meenakshi College of Engineering, Chennai
principal@thirusevenhillspolytechnic.com
[3]HOD, Department of Computer Science and Engineering, Meenakshi College of Engineering, Chennai
rajimukesh95@yahoo.co.in
[4]Assistant Professor, Department of Computer Science and Engineering, Meenakshi College of Engineering, Chennai
m.kasiselvi@gmail.com
[5]Research Scholar, Department of Computer Science and Engineering, Anna University of Technology, Coimbatore
kumananvetri@yahoo.com

***Abstract*–Peer-to-Peer VoIP (voice over IP) networks, exemplified by Skype, are becoming increasingly popular due to their significant cost advantage and richer call forwarding features than traditional public switched telephone networks. One of the most important features of a VoIP network is privacy (for VoIP clients). Unfortunately, most peer-to-peer VoIP networks neither provide personalization nor guarantee a quantifiable privacy level. In this project a novel flow analysis attacks to demonstrate the vulnerabilities of peer-to-peer VoIP networks and a method to overcome these attacks have been proposed. There are two important challenges in the design privacy-aware VoIP networks: Can provide personalized privacy guarantees for VoIP clients that allow them to select privacy requirements on a per-call basis and how to design VoIP protocols to support customizable privacy guaranteed This project can propose a practical solutions to address these challenges using a quantifiable k-anonymity metric and a privacy-aware of VoIP route setup and to route maintenance protocols. A detailed experimental evaluation that demonstrates the performance and scalability of the proposed has been implemented in this paper.**

***Index Terms*–VoIP networks, privacy, k-anonymity, mix networks, flow analysis attacks.**

I. INTRODUCTION

In VoIP networks several authors have used mix as a network routing element to construct anonymizing networks such as Onion Routing [2], Tor [3], Tarzan [4], or Freedom [5]. The mix network provides good anonymity for high-latency communications by routing network traffic through a number of nodes with random delay and random routes. However, emerging applications, such as VoIP, SSH, online gaming, etc., have additional quality of service (QoS) requirements; for instance, International Telecommunication Union (ITU) recommends up to 250-ms one-way latency for interactive voice communication; the recent case study indicates that latencies up to 250 ms are unperceivable to human users, while latencies over 400 ms significantly deteriorate the quality of voice conversations. This project examines anonymity for QoS sensitive applications on mix networks using peer-to peer VoIP service as a sample application. A peer-to-peer VoIP network typically consists of a core proxy network and a set of clients that connect to the edge of this proxy network. This network allows a client to dynamically connect to any proxy in the network and to place voice calls to other clients on the network. VoIP uses the two main protocols: route setup protocol (RSP) for call setup and termination, and real-time transport protocol (RTP) for media delivery. In order to satisfy QoS requirements, a common solution used in peer-to-peer VoIP networks is to use a route setup protocol that sets up the shortest route on the VoIP network from a caller src to a receiver dst.1 RTP is used to carry voice traffic between the caller and the receiver along an established bidirectional voice circuit. In such VoIP networks, preserving the anonymity of caller receiver pairs becomes a challenging problem. In this project, we focus on attacks that attempt to infer the receiver for a given VoIP call using traffic analysis on the media delivery phase. We make two important contributions. First, we show that using the shortest route (as against a random route) for routing voice flows makes the anonymizing network vulnerable to flow analysis attacks. Second, we develop practical techniques to achieve quantifiable and customizable k-anonymity on VoIP networks. Our proposal exploits the fact that audio codecs (such as G.729A without silence suppression2) generate statistically identical packet streams that can be mixed without leaking much information to an external observer.

II. RELATED WORK

A. Tarzan

Tarzan is a peer-to-peer anonymous IP network overlay. Because it provides IP service, Tarzan is general-purpose and transparent to applications. Organized as a decentralized peer-to-peer overlay, Tarzan is fault-tolerant, highly scalable, and easy to manage. Tarzan achieves its anonymity with layered encryption and multihop routing, much like a Chaumian mix. A message initiator chooses a path of peers pseudo-randomly through a restricted topology in a way that

adversaries cannot easily influence. Cover traffic prevents a global observer from using traffic analysis to identify an initiator. Protocols toward unbiased peer-selection offer new directions for distributing trust among untrusted entities. Tarzan provides anonymity to either clients or servers, without requiring that both participate. In both cases, Tarzan uses a network address translator (NAT) to bridge between Tarzan hosts and oblivious Internet hosts. Measurements show that Tarzan imposes minimal overhead over a corresponding non-anonymous overlay route.

B. Onion Routing

Onion Routing is a general purpose infrastructure for private communication over a public network [2, 3, 4]. It provides anonymous connections that are strongly resistant to both eavesdropping and traffic analysis. The connections are bidirectional, near real-time, and can be used for both connection-based and connectionless traffic. Onion Routing interfaces with the shelf software and systems through specialized proxies, making it easy to integrate into existing systems.

C. Attacks and Defenses- Passive attacks – Observing user traffic patterns.

Observing a user's connection will not reveal her destination or data, but it will reveal traffic patterns (both sent and received). Profiling via user connection patterns requires further processing, because multiple application streams may be operating simultaneously or in series over a single circuit.

D. Observing user content

While content at the user end is encrypted, connections to responders may not be (indeed, the responding website itself may be hostile). While filtering content is not a primary goal of Onion Routing, Tor can directly use Proxy and related filtering services to anonymize application data streams.

E. Option distinguishability

We allow clients to choose configuration options. For example, clients concerned about request linkability should rotate circuits more often than those concerned about traceability. Allowing choice may attract users with different needs; but clients who are in the minority may lose more anonymity by appearing distinct than they gain by optimizing their behavior.

F. End-to-end timing correlation

Tor only minimally hides such correlations. An attacker watching patterns of traffic at the initiator and the responder will be able to confirm the correspondence with high probability. The greatest protection currently available against such confirmation is to hide the connection between the onion proxy and the first Tor node, by running the OP on the Tor node or behind a firewall. This approach requires an observer to separate traffic originating at the onion router from traffic passing through it: a global observer can do this, but it might be beyond a limited observer's capabilities.

G. End-to-end size correlation

Simple packet counting will also be effective in confirming endpoints of a stream. However, even without padding, we may have some limited protection: the leaky pipe topology means different numbers of packets may enter one end of a circuit than exit at the other.

H. Active attacks – Compromise keys

An attacker who learns the TLS session key can see control cells and encrypted relay cells on every circuit on that connection; learning a circuit session key lets him unwrap one layer of the encryption. An attacker who learns an OR's TLS private key can impersonate that OR for the TLS key's lifetime, but he must also learn the onion key to decrypt *create* cells (and because of perfect forward secrecy, he cannot hijack already established circuits without also compromising their session keys). Periodic key rotation limits the window of opportunity for these attacks. On the other hand, an attacker who learns a node's identity key can replace that node indefinitely by sending new forged descriptors to the directory servers.

I. Iterated compromise

A roving adversary who can compromise ORs (by system intrusion, legal coercion, or extralegal coercion) could march down the circuit compromising the nodes until he reaches the end. Unless the adversary can complete this attack within the lifetime of the circuit, however, the ORs will have discarded the necessary information before the attack can be completed. (Thanks to the perfect forward secrecy of session keys, the attacker cannot force nodes to decrypt recorded traffic once the circuits have been closed.) Additionally, building circuits that cross jurisdictions can make legl coercion harder—this phenomenon is commonly called "jurisdictional arbitrage." The Java Anon Proxy project recently experienced the need for this approach, when a German court forced them to add a backdoor to their nodes

J. Run an onion proxy

It is expected that end users will nearly always run their own local onion proxy. However, in some settings, it may be necessary for the proxy to run remotely— typically, in institutions that want to monitor the activity of those connecting to the proxy. Compromising an onion proxy compromises all future connections through it.

K. Run a hostile OR

In addition to being a local observer, an isolated hostile node can create circuits through itself, or alter traffic patterns to affect traffic at other nodes. Nonetheless, a hostile node must be immediately adjacent to both endpoints to compromise the anonymity of a circuit. If an adversary can run multiple ORs, and can persuade the directory servers that those ORs are trustworthy and result in a different negotiated session key, and so the rest of the recorded session can't be used.

M. Smear attacks

An attacker could use the Tor network for socially disapproved acts, to bring the network into disrepute and get its operators to shut it down. Exit policies reduce the possibilities for abuse, but ultimately the network requires volunteers who can tolerate some political heat.

III. PROPOSED SYSTEM OF VoIP

In this approach we are not including a server as we are creating a personal network without the involvement of the third party. Next we are using a peer-to-peer network setup; the peer-to-peer VoIP network consists of a core proxy network and a set of clients that connect the edge of those proxy networks. Finally we are setting up a VoIP route setup protocol.

Advantages

- Without Third party server, Peer-to-Peer VoIP network has been used.
- Naïve Tracing Algorithm, Shortest path Algorithm, Flow analysis Algorithm has used.
- Sending and Receiving voice can be more secure.

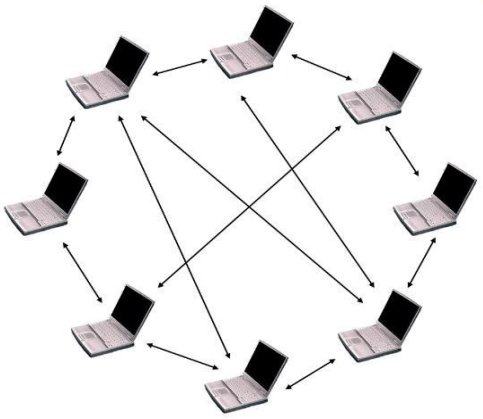


Fig.1 Without Third party server

IV. OVER ALL ARCHITECTURE

independent, then occasionally some user will choose one of those ORs for the start and another as the end of a circuit.

L. Replay attacks

Some anonymity protocols are vulnerable to replay attacks. Tor is not; replaying one side of a handshake will

Utilizing the latest technologies in voice such as VoIP (Voice-Over-Internet-Protocol), and other networking technologies, we can provide your company incredibly reliable and robust IP based solutions to get the job done. Big or small, we offer a broad range of services for every type of company. IP based voice solutions offer compelling advantages over traditional technologies. You can utilize existing data infrastructure by combining both data and voice in one line. With bundled IP solutions by VoIP Networks you can have all calls routed to multiple devices and remote locations, even over the Internet to your home. Regardless of where you are, you and your employees can always be available.

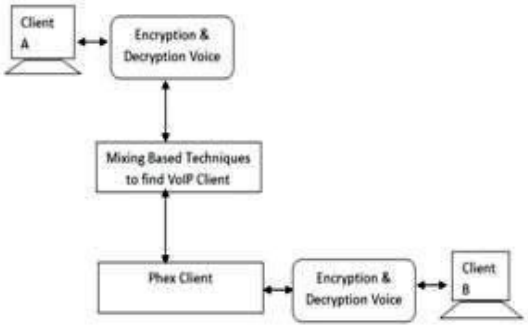


Fig 2. Architecture of VoIP Peer to Peer Network

Privacy is the ability of an individual or group to seclude them or information about themselves and thereby reveal them selectively. The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Privacy is sometimes related to anonymity, the wish to remain unnoticed or unidentified in the public realm. When something is private to a person, it usually means there is something within them that is considered inherently special or personally sensitive. The degree to which private information is exposed therefore depends on how the public will receive this information, which differs between places and over time. Privacy partially intersects security, including for instance the concepts of appropriate use, as well as protection, of information.

K-anonymity provides privacy protection by guaranteeing that each released record will relate to at least k individuals even if the records are directly linked to external information. This paper provides a formal presentation of combining generalization and suppression to achieve k-anonymity. Generalization involves

replacing a value with a less specific but semantically consistent value. Suppression involves not releasing a value at all. The Preferred Minimal Generalization Algorithm, which is a theoretical algorithm presented herein, combines these techniques to provide k-anonymity protection with minimal distortion. The real-world algorithms Data fly and m-Argus are compared to MinGen. Both Data fly and m-Argus use heuristics to make approximations, and so, they do not always yield optimal results. It is shown that Data fly can over distort data and m-Argus can additionally fail to provide adequate protection.

Mix networks were invented by David Chaum in 1981. Digital mixes create hard-to-trace communications by using a chain of proxy servers. Each message is encrypted to each proxy using public key cryptography; the resulting encryption is layered like a Russian doll (except that each "doll" is of the same size) with the message as the innermost layer. Each proxy server strips off its own layer of encryption to reveal where to send the message next. If all but one of the proxy servers is compromised by the tracer, untraceability can still be achieved against some weaker adversaries.

Flow analysis is the process of intercepting and examining messages in order to deduce information from patterns in communication. It can be performed even when the messages are encrypted and cannot be decrypted. In general, the greater the number of messages observed, or even intercepted and stored, the more can be inferred from the flow. Flow analysis can be performed in the context of military intelligence or counter-intelligence, and is a concern in computer security.

Privacy has long been a hot button issue for both the VoIP clients and the law enforcement bodies. On one hand, users want their phone conversations to be anonymous; anonymity offers them possible deniability, thereby shielding them from law enforcement bodies. On the other hand, Federal Communications Commission considers the capability of tracking VoIP calls of paramount importance to the law enforcement and the national security interests of the United States.

Similar to other VoIP privacy papers, we leave aside the controversy between anonymity and security. Instead, we focus on technical feasibility of privacy attacks and defenses on VoIP networks. Mix is a routing element that attempts to hide correspondences between its input and output messages. A large number of low-latency anonymizing networks have been built using the concept of a mix network.

Onion routing and its second generation Tor aim at providing anonymous transport of TCP flows over the Internet. ISDN mixes propose solutions to anonymize phone calls over traditional Public Switched Telephone Networks. In this paper, we have focused on VoIP networks given its recent widespread adoption.8 It is widely acknowledged that low-latency anonymizing networks are vulnerable to timing analysis attacks, especially from well-placed malicious attackers. Several papers have addressed the problem of tracing encrypted traffic using timing analysis.

All these papers use interpacket timing characteristics for tracing traffic. Complementary to all these approaches, we have introduced flow analysis attacks that target the shortest path property of voice routes and presented techniques to provide customizable anonymity guarantees in a VoIP network. Unlike the timing analysis attacks, our approach does not rely upon interpacket times to detect caller-receiver pairs; instead, we analyze the volume of flow in the VoIP network and deduce possible caller-receiver pairs using the flow information and the underlying VoIP network topology.

Tarzan presents an anonymizing network layer using a gossip-based peer-to-peer protocol. We note that flow analysis attacks target the shortest path property and not the protocol used for constructing the route itself; hence, a gossip-based shortest path setup protocol is equally vulnerable to flow analysis attacks. Traditionally, multicast and broadcast protocols have been used to protect receiver anonymity. However, in a multicast-based approach achieving k-anonymity may increase the network traffic by k-fold. In contrast, our paper attempts to reroute and mix existing voice flows and thus incurs significantly smaller overhead on the VoIP network.

We have addressed the difficulty of providing privacy guarantees in peer-to-peer VoIP networks. We have developed flow analysis attacks that allow an adversary to identify a small and accurate set of candidate receivers even when all the nodes in the network are honest. We have used network flow analysis and statistical inference to study the efficacy of such an attack. Second, we have developed mixing-based techniques to provide a guaranteed level of anonymity for VoIP clients. We have developed an anonymity-aware route setup protocol that allows clients to specify personalized privacy requirements for their voice calls using a quantifiable k-anonymity metric. We have implemented our proposal on the Phex client and presented detailed experimental evaluation that demonstrates the performance and scalability.

V. RESULTS

A. Running the Tcl code:

NS2 executes .tcl file format. You can execute a .tcl file by typing the following syntax in the terminal: “ns [file name].tcl”,but make sure you are the directory where the .tcl file is present. Now type the following command in the terminal to view simulation of VOIP “ns voip.tcl” The output of the tcl file is:

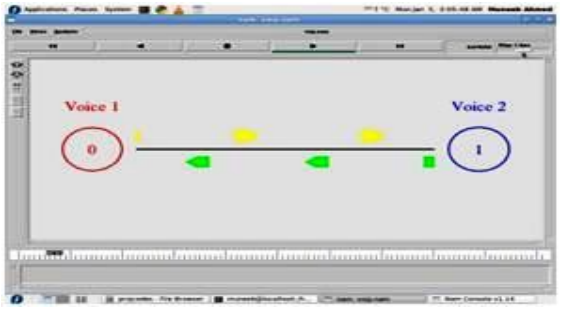


Fig 3. Simulation of VoIP Handling the trace file:

B. Handling the Trace File

On execution of .tcl code, two output files are generated. One is the .NAM file with which we see the graphical simulation of our code. The other one is the .tr trace file, with which we can analyze the output of our simulation. The looks like:

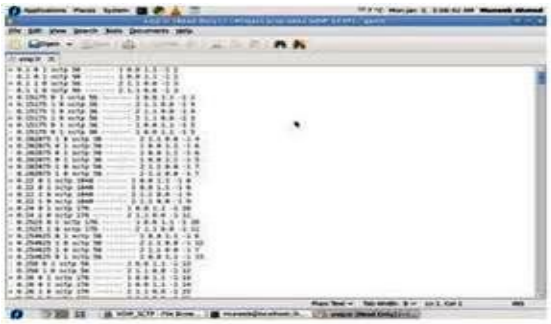
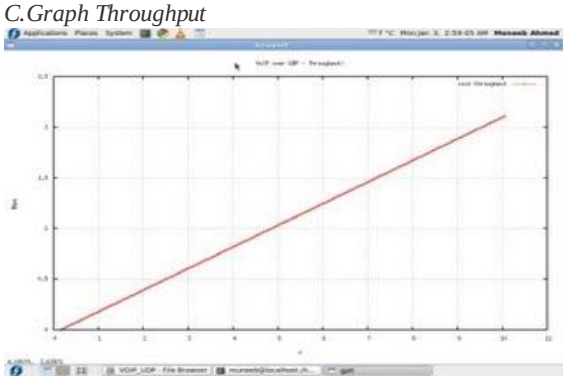


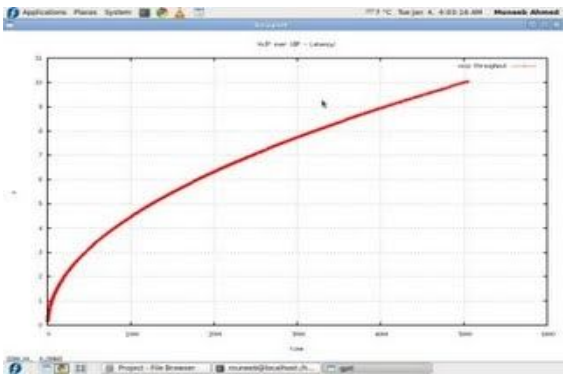
Fig 4. VoIP Trace files

This file contains various parameters such as arrival time of packets, packet size transport agent etc. Using the trace file, we can get the graphical outputs to analyze the behavior of our simulation.

C.Graph Throughput



D.Graph Latency



VI. CONCLUSION

The proposed scheme is we have addressed the problem of providing privacy guarantees in peer-to-peer VoIP networks. (i) We have developed flow analysis attacks that allow an adversary (external observer) to identify a small and accurate set of candidate receivers even when all the nodes in the network are honest. We have used network flow analysis and statistical inference to study the efficacy of such an attack, (ii) we have developed mixing-based techniques to provide a guaranteed level of anonymity for VoIP clients. We have developed an anonymity-aware route setup protocol that allows clients to specify personalized privacy requirements for their voice calls (on a per-client per-call basis) using a quantifiable k anonymity metric. We have implemented our proposal on the Phex client and presented detailed experimental evaluation that demonstrates the performance and scalability of our protocol, while meeting customizable privacy guarantees.

In the second phase it has planned to build experiment environment based on the proposed system and to extend that to implement one more algorithm in Application Layer.

REFERENCES

[1] Mudhakar Srivatsa, Arun Iyengar, Ling Liu, and Hongbo Jiang, “Privacy in VoIP Networks: Flow Analysis attacks and defense”, IEEE Transactions on parallel and distributed Systems, Vol.22, NO.4, April 2011.

[2] D. Goldschlag, M. Reed, and P. Syverson, “Onion Routing for Anonymous and Private Internet Connections,” Comm. ACM, vol. 42, no. 2, 1999.

[3] B. Fortz and M. Thorup, “Optimizing OSPF/IS-IS Weights in a Changing World”, IEEE J. Selected Areas in Comm., vol. 20, no. 4, pp. 756-767, May 2002.

[4] M.J. Freedman and R. Morris, “Tarzan: A Peer-to-Peer Anonymizing Network Layer,” Proc. Ninth ACM Conf. Computer and Comm. Security (CCS), 2002.

[5] L. Qiu, V.N. Padmanabhan, and G.M. Voelker, "On the Placement of Web Server Replicas," Proc. IEEE INFOCOM, 2001.

[6] "The Network Simulator NS-2," http://www.isi.edu/nsnam/ns/,2010.

[7] "The Network Simulator NS-2: Topology Generation,"http://www.isi.edu/nsnam/ns/ns-topogen.html, 2010.

[8] "TelegeographyResearch," http://www.telegeography.com,

[9] "Phex Client," http://www.phex.org, 2010.

[10] G. Perng, M.K. Reiter, and C. Wang, "M2: Multicasting Mixes for Efficient and Anonymous Communication," Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS), 2006.