A Collaborative Forensics Framework for VoIP Services in Multi-network Environments

Hsien-Ming Hsu[1], Yeali S. Sun[1], and Meng Chang Chen[2]

[1] Dept. of Information Management, National Taiwan University, Taipei, Taiwan
{d94002,sunny}@im.ntu.edu.tw
[2] Institute of Information Science, Academia Sinica, Taipei, Taiwan
mcc@iis.sinica.edu.tw

Abstract. We propose a collaborative forensics framework to trace back callers of VoIP services in a multi-network environment. The paper is divided into two parts. The first part discusses the critical components of SIP-based telephony and determines the information needed for traceback in single and multiple Autonomous Systems (ASs). The second part proposes the framework and the entities of collaborative forensics. We also propose an algorithm for merging collected data. The mechanism used to execute collaborative forensics with co-operating units is presented and the procedures used in the collaborative architecture are described. For every entity, we suggest some interesting topics for research.

Keywords: collaborative forensics, VoIP services, traceback, SIP.

1 Introduction

The Public Switched Telephone Network (PSTN) has dominated voice communications over a long period. With the growth of the Internet, however, VoIP (Voice over IP) services based on packet-switched technology have become widely accepted and could eventually replace PSTN. Currently, a major drawback of VoIP services is that they are vulnerable to many potential security threats inherited from the Internet Protocol (IP). A taxonomy for mitigating potential VoIP security and privacy problems is defined in [1].

While VoIP services have many desirable communication features, they have also become a tool for illegal activities, as criminals can communicate via VoIP services and avoid being intercepted by law enforcement agencies (LEAs). There are a number of reasons why LEAs have difficulty intercepting and tracing back VoIP calls. Two major reasons are that 1) diverse techniques are used to access the Internet, e.g., campus networks, General Packet Radio Service (GPRS), Public 802.11 wireless network, and 3G; and 2) the dynamic addresses assigned to the caller/callee, are frequently located behind a Network Address Translation (NAT) router. Therefore, how to help LEAs identify IP packets lawfully is a major problem in various networks [2].

The goal of the VoIP traceback task is to trace the identities and geo-locations of the caller and callee of a VoIP service. To achieve this goal, Network Operators, Access Providers and Service Providers (NWO/AP/SvP) have to cooperate to record the identities of the parties and other necessary information. In this paper, we argue that

C.C. Yang et al. (Eds.): ISI 2008 Workshops, LNCS 5075, pp. 260–271, 2008.

the information needs to be recorded by operators of SIP- (Session Initiation Protocol) based networks. We propose a collaborative forensics framework, protocol and mechanism that automatically collects, associates, manages, and links information in order to reconstruct criminal acts. As a result, different parts of an event can be linked to build a complete picture of an incident that could be used as evidence in a court of law. By correlating related events, we can determine how a network incident (i.e., crime/attack) occurred, including the origin, the method used, and the people responsible. Ultimately, we hope to apply our findings to help prevent criminal activities on the Internet.

The proposed collaborative forensics framework is based on two assumptions: a) each NWO/AP/SvP has the administrative capability to handle event interception and to provide correct information to LEAs; and b) the collaborative forensic operating environment is secure.

The primary contributions of this paper are the follows:

- We discuss the protocol and the critical components of VoIP services and determine the information that needs to be recorded for possible forensic investigations.
- We explain how to perform traceback by using the recorded information in two scenarios, single and multiple AS networks.
- We propose a cooperative architecture, protocol and mechanism for collaborative forensics. In addition, we propose an algorithm for merging the collected data.
- We suggest several interesting research avenues related to the development of the collaborative framework.

The remainder of this paper is organized as follows. Section 2 contains a review of related works. In Section 3, we describe SIP-based VoIP services and traceback. In Section 4, we discuss the proposed collaborative forensics framework. Then, in Section 5, we summarize our conclusions and indicate future research avenues.

2 Related Work

Although VoIP provides many desirable services, such as convenient voice calls, the services are vulnerable to a number of potential security threats inherited from the root Internet Protocol (IP). In recent years, VoIP has become a tool for illegal activities as criminals have exploited the security loopholes in IP. In [1], the authors investigate the risks of VoIP technology and define a taxonomy to enhance VoIP security and mitigate threats to privacy. Because VoIP services rely on the Internet, they are vulnerable to threats from different protocol layers. In [3], attacks are categorized by the vulnerabilities of VoIP devices, configurations, infrastructures, protocols and applications. Meanwhile, some works have focused on developing a VoIP intrusion detection system [4, 5]. There has also been a substantial amount of research on how to establish an LEA architecture for VoIP services [6, 7], and how to enhance VoIP for use by emergency services [8]. Newly-developed anonymous VoIP telephone services (e.g., Skype [9]) make the traceback task even more difficult for LEAs. To resolve this problem, Wang et al. [10] proposed a method that effectively traces anonymous calls by embedding a unique watermark on the inter-packet timing of the VoIP flow in real-time.

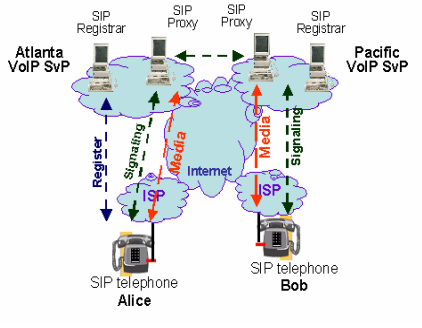

Fig. 1. SIP-based IP telephony

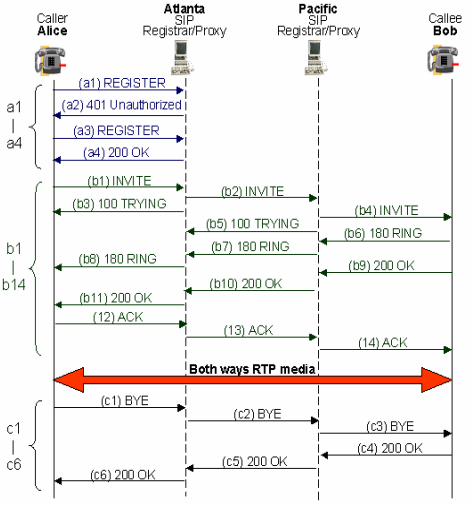

Fig. 2. SIP Signaling

In this research, we follow the approaches in [11, 12, 13] and propose a framework for collaborative forensics. We also propose sending XML (eXtensible Markup Language) [14] formatted messages via web services. Our objective here is twofold: 1) to help LEAs obtain the information necessary to trace back VoIP phone calls; and 2) to help domain experts construct domain knowledge to enhance existing systems collaboratively.

3 VoIP Traceback

In this paper, we only consider SIP-based IP telephony. To provide VoIP services, an SIP-based telephone system utilizes multiple protocols, including the Session Initiation Protocol (SIP) [15] and the Real-time Transport Protocol (RTP) [16]. As mentioned earlier, there are a number of reasons why LEAs have difficulty intercepting and tracing back VoIP calls. Two major reasons are that 1) diverse techniques are used to access the Internet (e.g., campus networks, General Packet Radio Service (GPRS), Public 802.11 wireless network, and 3G); and 2) dynamic IP addresses assigned to a caller/callee are frequently located behind the Network Address Translation (NAT) router. In this section, we describe the communications and the critical points for VoIP services, and determine the information that needs to be recorded. Then, based on the recorded information, we discuss how to perform traceback with single and multiple ASs.

3.1 The Communications and Critical Points of VoIP Services

The architecture of SIP-based IP telephony is shown in Fig. 1. The Registrars and Proxies are the SIP servers. A Registrar is responsible for registration, after which the Proxy servers relay the signaling to the callee's address and offer the service. The

Table 1. Information recorded by the SIP Registrar Server

Attributes	Description
Account	User's network-based phone account
Source IP	Obtained from the user's registered message
Timestamp	The time the call was registered

Table 2. Information recorded by the SIP Proxy server

Attributes	Description
Caller's Account	Caller's account or telephone number; obtained from the caller's INVITE message.
Callee's Account	Callee's account or telephone number; obtained from the caller's INVITE message.
Caller's IP/Port (Signaling)	Caller's IP and Port number; obtained from the caller's INVITE message.
Callee's IP/Port (Signaling)	Callee's IP and Port number; obtained from the callee's OK message.
Caller's IP/Port (media)	Caller's IP and Port number; obtained from the caller's SDP on INVITE message.
Callee's IP/Port (media)	Callee's IP and Port number; obtained from the OK message of the callee's SDP
Time: From	The time Proxy received the INVITE
Time: To	The time Proxy received the BYE
Answering time	The time Proxy received the OK

Table 3. The NAT/DHCP

Attributes	Description
Account	User's network-based phone account
Private IP/Port	The private IP/Port with NAT
Public IP/Port	The public IP/Port assigned to NAT/DHCH
Time: From	The time the private IP made the call
Time: To	The time the private IP was interrupted

signaling of the SIP protocol is shown in Fig. 2 [15]. Tables 1, 2, and 3 list the respective information that the SIP Registrar Server, SIP Proxy Server and NAT/DHCP (Dynamic Host Configuration Protocol) need to record for traceback.

3.2 Traceback within a Single AS Network

We use a scenario of a caller (Alice) and a callee (Bob) in a single AS network to explain how we perform traceback from Bob to Alice, as shown in Fig. 3.

Case 1: Both Alice and Bob are with public IPs in a single AS. In this scenario, all the required information about the caller and callee is recorded by the Registrar and Proxy service providers when the connection for the session is set up. The information, which is distributed over a number of components (as shown in Fig. 3), can be easily collected,

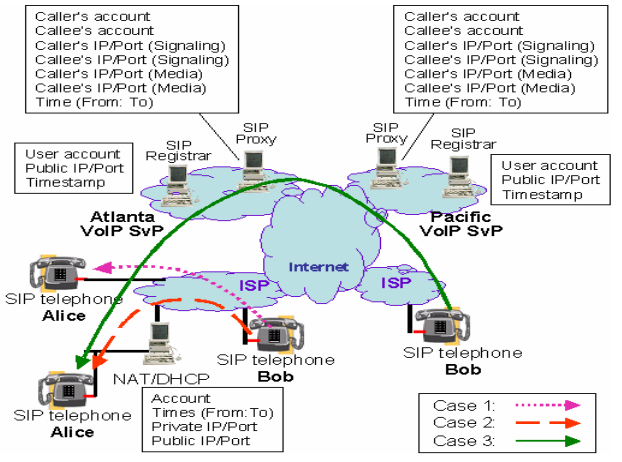

Fig. 3. The necessary information recorded by the components of SIP-based telephony to trace back calls in Single- and Multi-operators

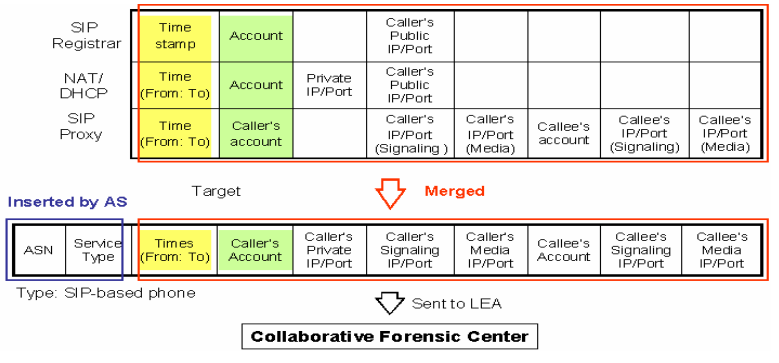

Fig. 4. The data recorded by the NAT/SIP Registrar/SIP Proxy is merged as a Local Event

extracted and merged into a Local Event (LE). The LE can be represented as an XML formatted message and reported by the administrator using the SEAL Protocol (introduced in the next section). Before the result can be sent to the LEA, the autonomous system number (ASN) and service type are inserted for classification and storage purposes, as shown in Fig. 4.

Case 2: Alice is with a private IP and Bob is with public IP in a single AS.

Because Alice is behind the NAT with the private IP, we need the NAT router to record the mappings of private IPs and public IPs during the connection period. The information gathered by the NAT router, SIP Registrar and SIP Proxy is collected, extracted and merged as a Local Event. This is same as case 1, except for the private IP, as shown in Fig. 4. Then, based on the reported events, we can perform a trace back to the caller across the NAT from the callee. For collaborative forensics, we have to decide how long information about Local Events should be kept. A tradeoff between storage requirements and the need for accuracy (mainly, the false negative rate) has to be made carefully.

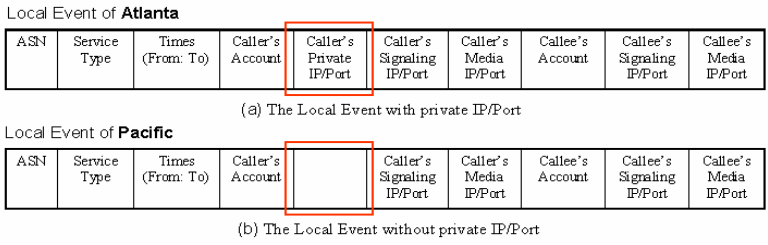

Fig. 5. The Local Events Protocol with/without private IP/Port

3.3 Traceback with Multi-AS Networks

Case 3: Traceback under multi-AS is similar that for the single AS cases described in the previous section, except that Alice and Bob's IPs are located at different service providers. In Fig. 3, the caller (Alice) and the callee (Bob) belong to different VoIP service providers, each of which has its own SIP register and SIP proxy servers. Assume that caller Alice is either behind the NAT router with a private IP or she uses the dynamic IP. After Alice completes the registration, the signaling will be relayed from the Atlanta SIP proxy to the Pacific SIP proxy, which will then relay it to Bob. Each Atlanta and Pacific SIP proxy can obtain the complete information within its AS and produce its Local Event independently, as shown in Fig. 5.

When tracing back from the callee Bob, we can get the caller's public IP from the Local Event of Pacific, but not the private IP. However, if we can match the Local Event of Pacific with the Local Event of Atlanta, we can obtain the caller's private IP, as shown in Fig. 5 (b).

4 The Framework of Collaborative Forensics for VoIP Services

Operators may not want to share their information with others for a variety of reasons (e.g., privacy concerns, commercial competition, policy, cultures, and implementation differences). One way to solve the problem is to design a mechanism that can be supervised by an independent authority. The mechanism would aggregate, integrate and correlate local information (i.e., Local Events) from operators to carry out the traceback task without violating privacy laws. Since the LE only contains information needed for traceback and the collaborative framework is under the supervision of an independent authority, network operators should not be reluctant to collaborate.

Network operators already have systems and databases for the distribution of information needed for traceback. Therefore, we only need a cooperative architecture, protocol and mechanism to automatically collect, associate, manage, link and reconstruct information about criminal activity in real-time for a fast response. The proposed collaborative framework, called SKYEYE, is designed to meet this need.

4.1 SKYEYE Entities and Their Functions

In this section, we introduce the entities of SKYEYE and their functions. Fig. 6 illustrates the SKYEYE entities and procedures, as well as the cooperating units, i.e., LEA, existing systems, NWO/AP/SvP and FRTs (Fast Response Teams).

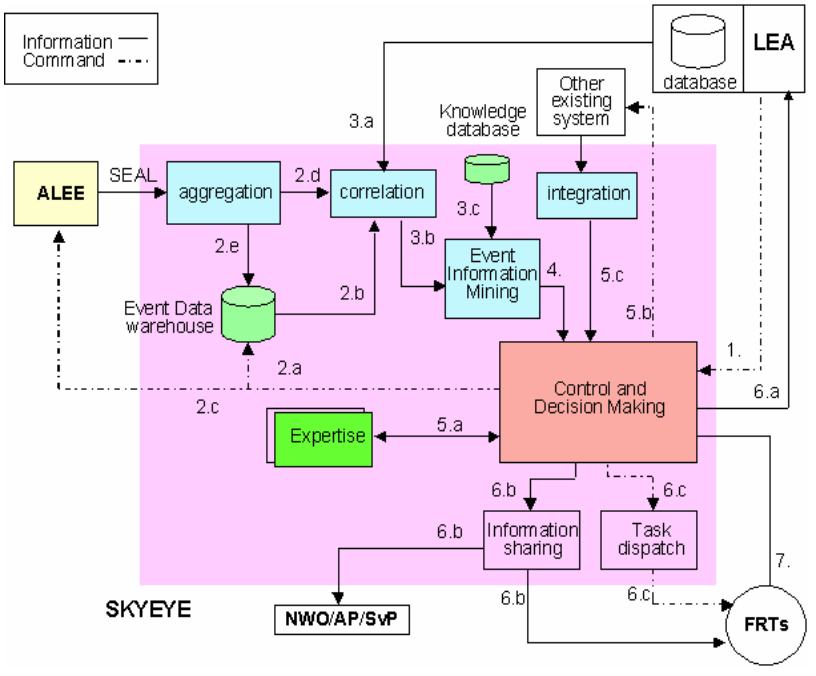

Fig. 6. The architecture and procedures of SKYEYE

4.1.1 Local Event (LE)

Local Event data is collected from the NAT/DHCP, SIP Registrar and SIP Proxy in an AS by ALEE using the specification of SkyEye-ALee (SEAL) Protocol. For one VoIP call, the ALEE will produce two copies of the Local Event; one will be stored in the local operator's database for backup, and the other will be sent to SKYEYE for forensic analysis. The service type of the Local Event (e.g., SIP, GSM, 3G) will be labeled accordingly by ALEE. No matter whether the caller and callee are located in the same AS or not, their IPs are encapsulated in the LE, as shown in Figs. 4 and 5. For traceback to the caller, we only need the LE's from the callee and caller. In other words, the LEs of the intermediate ASs are not needed.

4.1.2 Access Local Event Entity (ALEE)

ALEE is the interface of AS that connects and communicates with SKYEYE, which is independent of the AS realm. For each SIP-based phone call, the ALEE will automatically collect the required information about the caller and callee from the NAT/DHCP, SIP Registrar and SIP Proxy in the operating network (AS) and merge the pieces of information to produce the Local Event. When ALEE receives the command from SKYEYE, the queried LEs will be sent to SKYEYE again for confirmation.

4.1.3 The Flexible SKYEYE ALEE (SEAL) Protocol

The SEAL protocol is designed to transport Local Events presented in XML-formatted messages; therefore, it can easily be extended to accommodate different access network technologies and different services.

4.1.4 The SKYEYE

The SKYEYE is the kernel of the collaborative forensic mechanism. All the collaborative investigating activities are executed in this entity. It is comprised of the following modules and functions: aggregation, event correlation, event information mining, integration, and expertise repository.

Below, we describe each module and its function in the collaborative forensics framework. We also indicate issues that require further research.

- *The Aggregation Module (AGG)*: The AGG is the interface of ALEE that collects all LEs from ALEE and stores them in the event data warehouse (EDW). To improve the querying and search functions, further research is required on: a) LE classification and storage; and b) efficient aggregation methods.
- *The Correlation Module (CORR)*: The CORR's function is to correlate related LEs in order to build a complete picture and determine *how* a network incident (crime, attack) occurred. CORR tries to find out the origin, the method, the people responsible, and the identity of potential victims. Further research is required on a) an efficient algorithm for correlating related events; and b) how to identify the key attributes of a crime for correlative forensics.
- *The Event Information Mining Module (EIM)*: By exploiting data mining techniques, the EIM tries to discover useful knowledge from LEs in order to predict criminals' intentions and thereby prevent crimes occurring. Further research is required on: a) techniques for Event information mining; and b) a model of criminal behavior that can be used to predict possible criminal activity.
- *The Integration Module (INTE)*: The INTE is an interface for integrating information from existing information systems for forensic analysis.
- *Control and Decision Making Module (CDM)*: This is the core module of SKYEYE. Its main functions are to control the processes of collaborative forensics and make necessary decisions for other tasks. The CDM can look up an expertise repository or consult domain experts via virtual panels for decisions. The CDM issues orders for Fast Response Teams (FRTs) to execute tasks, and provides updates about the latest situations and information about incidents. Further research is required to a) develop standard operating procedures (SOPs) for the CDM; b) establish a decision-making procedure; and c) improve methods the used to identify potential victims and criminal companies.
- *Fast Response Teams (FRTs)*: FRTs are specialized units that perform diverse tasks. They receive and execute orders sent by the CDM. They also report the latest situations and incidents to the CDM.
- *Information Sharing*: Event information is shared with partners for event detection and prevention, defense in depth and fast responses to events, and to alert potential victims.

4.2 Collaborative Forensics Work of SKYEYE

Next, we describe the execution of collaborative forensics with cooperating units. The steps are as follows.

Step (1). LEA sends commands to the Control and Decision-Making module (CDM) of SKYEYE. Each command includes two essential elements, the Callee's

Account and the Calling time-parameter. The former is the starting point for the traceback, and the latter is used to identify the right call. Both of them are key attributes of LE searches of the Event Data Warehouse (EDW) and the operator's database.

Step (2.a). The CDM module sends a request to the EDW and ALEE for a Local Event (LE) search with Callee's Account and Calling time-parameter. If an LE exists it should be stored in the EDW, because it would have been produced and sent to the EDW by ALEE when a call was terminated. The only reasons for the non-existence of an LE are that it was deleted when the stored data expired or some transmission errors occurred.

Step (2.b). The responses of EDW depend on the number of LEs found in its database. For example, given a Calling-parameter, $19^{h}30^{m}$:$19^{h}52^{m}$, for each call, there should be two LEs, produced by the ASs of the caller and the callee respectively and stored in the EDW. The LEs will be accessed and sent to correlation module (CORR) to be double-checked with the LE information from the operator. For these LEs, the Caller's Related Information (e.g., Account, Public IP/Port) and ASN can be obtained and passed to the LEA (Step 6.a). For SIP-based phone calls, the data collection task has been completed in principle.

Step (2.c). No matter whether the LEs can be found in the EDW or not, the CDM module sends the request to the operator for Local Events (LEs) for confirmation. The LEs will be sent to Aggregation module (AGG) via SEAL. First, the AGG will check whether the LEs are still stored in the EDW. If they are, they will be sent to the CORR module (Step 2.d) for double-checking and correlating; otherwise, they will be stored in the EDW first (Step 2.e), and then forwarded to the CORRR.

Step (3.a). The CORR module double-checks the LEs sent by the operators and the EDW, and then correlates them with the data in the LEA database. For the forensic investigation, it is necessary to confirm the true identity of the caller by his/her Account during the Calling time. Based on the key attributes of Local Events, the CORR tries to fit the parts of events together correctly and build a *complete* picture of the incident, which can be used as evidence in a court of law.

Step (3.b). The related Local Events are sent to the Event Information Mining module (EIM).

Step (3.c). Based on the related Local Events, the EIM will consult the forensic domain knowledge in the Knowledge DataBase (KDB) to guide the search or evaluate the behavior models or predict criminal activity.

Step (4). The EIM mines the LEs for further useful information that could be used to predict criminal activity. All of the predicted results are sent to CDM module.

Step (5.a). The CDM may need to consult the domain experts via the virtual panel.

Step (5.b). The CDM tries to determine if any information is missing or confusing, and requests data or evidence from other systems.

Step (5.c). The other systems send their responses to the Integration module (INTE), which can integrate information in different formats. The result is sent to CDM to support decisions about subsequent action.

Step (6.a). All the decisions will be passed to the LEA. Step (6.b). The information is shared with the cooperating units to alert them in order to prevent possible criminal activity, and with FRTs (Fast Response Teams) to support their tasks. Step (6.c).

Step (7). Any changes and updates should be sent back to the CDM module.

4.3 The Algorithm for Data Merging

The data merging algorithm listed in Fig. 7 is used for VoIP services only. For the other types of service, the corresponding algorithms need to be defined according to their individual needs and the information needed for traceback and forensics. When an SIP-based phone call is terminated, all information recorded by the SIP Proxy, SIP Registrar and NAT router is collected by the AS administrator and sent to ALEE. Then, ALEE processes the data merged by the algorithm in Fig. 7. The output is the Local Event presented as an XML-formatted SEAL message.

```
Algorithm for Data_Merging
Input: SIP_Proxy, SIP_Registrar, NAT_router
Output: Local_Event

begin
   NAT_R:=NAT Record;
   SP_t:=SIP_Proxy(Time(From:To));
   SR_t:=SIP_Registrar(Timestamp);
   NAT_t:=NAT(Time(From:To));
   CRA_SP:=Caller's Account of SIP Proxy;
   CRA_SR:=Caller's Account of SIP Registrar;
   CRA_NAT:=Caller's Account on NAT router;
   CRA_U:=Caller's Account of User;
   CEA_SP:=Callee's Account of SIP Proxy;
   Pu_IP/Pt:=Public IP and Port;
   Pt_IP/Pt:=Caller's Private IP and Port;

   create a Local_Event
   LE.ASN:= ASN of operator
   LE.Service_Type:= Service_Type of calling;
   if (NAT_R is not empty)
      do (LE.time(From:To):= NAT_t;
          && LE.Caller_Account:= CRA_NAT;
          && LE.Private_IP_Port:= Pt_IP/Pt);
      else
      do (LE.time(From:To):= SP_t;
          && LE.Caller_Account:= CRA_SP);

   do (LE.caller_Public_IP_Port(signaling):= Pu_IP/Port(signaling);
       && LE.Caller_Public_IP_Port(Media):= Pu_IP/Port(Media));
   do (LE.Callee_Account:= CEA_SP;
       && LE.Callee_Public_IP_Port(Signaling):= CEA_SP(signaling);
       && LE.Callee_Public_IP_Port(Media):= Pu_IP/Port(Media));
end
```

Fig. 7. The Algorithm for Data Merging

5 Conclusion and Future Works

In this paper, we have discussed the critical components of VoIP services, and defined the information that needs to be recorded for forensic investigations. Based on the recorded information, we explain how to perform traceback in single and multiple AS networks. We propose an architecture, protocol and mechanism for collaborative

forensic tasks. In addition, we describe the entities of the architecture and their functions, and define the SEAL protocol with XML formatted messages.

The proposed SKYEYE model is the kernel of the collaborative forensic mechanism. The aggregation, event correlation, event information mining, integration and expertise modules are still under development.

Our ultimate goal is to establish a collaborative forensic center that can automatically collect, associate, manage, link and reconstruct information about possible criminal activities, as well as share the real-time information from different autonomous systems with all cooperating units.

Acknowledgements. This work is partly supported by the National Science Council of Taiwan under Grant No: NSC 96-3114-P-001-002-Y. The study would not have been completed without the help of the ANTS lab members, especially Liang-Ming Wu and Hao-Wen Ke.

References

1. Endler, D., Ghosal, D., Jafari, R., Karlcut, A., Kolenko, M., Nguyen, N., Walkoe, W., Zar, J.: VoIP Security and Privacy Threat Taxonomy, Public Release 1.0 (2005)
2. ETSI TR 101 944: Telecommunications security; Lawful interception (LI); Issues on IP Interception (2001)
3. Dhamankar, R.: Intrusion Prevention: The Future of VoIP Security. White paper. Tipping Point (2005), http://www.tippingpoint.com/pdf/resources/whitepapers/503160-001_TheFutureofVoIPSecurity.pdf
4. Sengar, H., Wijesekera, D., Wang, H., Jajodia, S.: VoIP Intrusion Detection Through Interacting Protocol State Machines. In: IEEE Dependable Systems and Networks Conference (2006)
5. Wu, Y., Bagchi, S., Garg, S., Singh, N., Tsai, T.: SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments. In: IEEE Dependable Systems and Networks Conference (2004)
6. Milaovic, A., Srbljic, S., Razjkevic, I., Sladden, D., Skrobr, D., Matosevic, I.: Distributed System for Lawful Interception in VoIP Networks. In: EUROCON (2003)
7. Karpagavinayagam, B., State, R., Festor, O.: Monitoring Architecture for Lawful Interception in VoIP Networks. In: Second International Conference on Internet Monitoring and Protection (2007)
8. Mintz-Habib, M., Rawat, A., Schulzrinne, H., Wu, X.: A VoIP Emergency Services Architecture and Prototype. Computer Communications and Networks (2005)
9. Skype-the Global Internet Telephony Company
10. Wang, X., Chen, S., Jajodia, S.: Tracking Anonymous Peer-to-Peer VoIP Call on the Internet. In: Proceedings of the 12th ACM Conference on Computer and Communications Security (2005)
11. Goodell, G., Aiello, W., Griffin, T., Ioannidis, J., McDaniel, P., Rubin, A.: Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing. In: The 10th Annual Network and Distributed System Security Symposium (2003)
12. Dawson, M., Winterbottom, J., Thomson, M.: IP Location- IP Location in Wireline Public Carrier Networks. McGraw-Hill Companies, New York (2007)

13. Nena, J.: Homeland Security Techniques and Technologies. Charles River Media, INC. (2004)
14. Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E.: Extensible Markup Language (XML) 1.0., 2nd edn. W3C Working Draft (2000)
15. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: SIP: Session Initiation Protocol (SIP). RFC 3261, IETF Network Working Group (2002)
16. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: A Transport Protocol for Real-time Applications. RFC 3550, IETF Network Working Group (2003), http://www.ietf.org/rfc/rfc3550.txt?number=3550