A model for user based IP traffic accounting

vom Fachbereich Informatik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

genehmigte Dissertation
von

Ge Zhang

Berichterstatter: **Prof. Dr. Paul Müller**
Prof. Dr. Jens Schmitt

Dekan: **Prof. Dr. Reinhard Gotzhein**

wissenschaftliche Aussprache am 12. Februar 2007

D 386

To my father

Abstract

Nowadays, accounting, charging and billing users' network resource consumption are commonly used for the purpose of facilitating reasonable network usage, controlling congestion, allocating cost, gaining revenue, etc. In traditional IP traffic accounting systems, IP addresses are used to identify the corresponding consumers of the network resources. However, there are some situations in which IP addresses cannot be used to identify users uniquely, for example, in multi-user systems. In these cases, network resource consumption can only be ascribed to the owners of these hosts instead of corresponding real users who have consumed the network resources. Therefore, accurate accountability in these systems is practically impossible. This is a flaw of the traditional IP address based IP traffic accounting technique.

This dissertation proposes a user based IP traffic accounting model which can facilitate collecting network resource usage information on the basis of users. With user based IP traffic accounting, IP traffic can be distinguished not only by IP addresses but also by users. In this dissertation, three different schemes, which can achieve the user based IP traffic accounting mechanism, are discussed in detail. The in-band scheme utilizes the IP header to convey the user information of the corresponding IP packet. The Accounting Agent residing in the measured host intercepts IP packets passing through it. Then it identifies the users of these IP packets and inserts user inforation into the IP packets. With this mechanism, a meter located in a key position of the network can intercept the IP packets tagged with user information, extract not only statistic information, but also IP addresses and user information from the IP packets to generate accounting records with user information. The out-of-band scheme is a contrast scheme to the in-band scheme. It also uses an Accounting Agent to intercept IP packets and identify the users of IP traffic. However, the user information is transferred through a separated channel, which is different from the corresponding IP packets' transmission. The Multi-IP scheme provides a different solution for identifying users of IP traffic. It assigns each user in a measured host a unique IP address. Through that, an IP address can be used to identify a user uniquely without ambiguity. This way, traditional IP address based accounting techniques can be applied to achieve the goal of user based IP traffic accounting. In this dissertation, a user based IP traffic accounting prototype system developed according to the out-of-band scheme is also introduced. The application of user based IP traffic accounting model in the distributed computing environment is also discussed.

Acknowledgements

I am grateful to have chance to thank all those people who have helped and supported me working on this dissertation.

First of all, I would like to express my sincere thanks to Professor Paul Müller for his advice and support during my doctoral research endeavor. As my doctoral supervisor, he provided me with valuable suggestions and comments to help me to establish the overall direction of the research and to move forward with investigation in depth. I also thank him for providing me with the opportunity to work with a talented team of researchers.

I would also like to thank my co-supervisor Professor Jens Schmitt for his helpful proposals and sharing his experience and knowledge in Internet accounting, charging and pricing.

I greatly appreciate Dirk Henrici for his generous time in proof reading and his comments. He is always ready to help.

I thank Bernd Reuther for his advice and cooperation during the project "NIPON".

Furthermore, I would like to thank all my colleagues in AG ICSY – especially Markus Hillenbrand, Jochen Mueller and Joachim Götze.

Last but not least, I would like to dedicate this dissertation to my parents for their invaluable constant support and encouragement, to my wife Ling Xue for her love, understanding and supporting, and to my lovely daughter Zhi Heng and my son Xuan Miao for giving me motivation to finish this dissertation.

Content

Chapter 1 Introduction

1.1 Background and Motivation

The boom of the Internet seems to be returning after the Internet bubble burst. The Internet is seeing exponential growth in the number of Internet users, connected hosts [GDS1], Internet service providers and traffic volume. The Internet is now becoming a convergent platform providing diverse services to users all over the world. The Internet services cover wide areas such as business, communication, entertainment, news, education, etc. The Internet is gradually playing a more and more important role in people's daily life and world economics.

As a consequence of the growth of the Internet, more and more IP traffic is being poured into the Internet. According to [Odly03], Internet traffic was increasing very rapidly, close to doubling each year since 1997 in the United States. Moreover, it continues growing close to this rate. The increasing number of Internet users, growing Internet economics, emerging Internet services and technologies such as peer to peer and Grid computing will contribute more IP traffic to the Internet.

The increasing traffic volume causes pressure on the networks, which may suffer performance decline due to congestion. Packet loss, increased delay, throughput degradation, unsatisfied QoS, etc. are possible consequences of congestion. In order to meet the increasing requirements of Internet users on high speed and huge capacity, many efforts have been made in enhancing the network equipment by renewing network devices, adopting new techniques [TIA05, Pion06]. Besides improving the capability of the network hardware, measures should also be taken to facilitate reasonable Internet usage and avoid unnecessary IP traffic generation to control network congestion and prevent Internet performance decline. IP traffic accounting is a solution which can provide information reflecting users' network resource consumption for the purpose of charging and billing. Charging users for their network resource consumption can stimulate their reasonable network resource usage on the one hand, yet allocate cost to users according to their network usage on the other hand. Accounting for users' network usage can also provide auditable information, which can help building a more secure Internet. A secure Internet is now considered to be one of the characters of future Ineternet by the NSF's GENI project [KLAD05].

Nowadays, charging and billing users for network resource consumption is commonly used by Internet service providers (ISP). Although the flat rate charging policy still plays a dominant role, it is gradually being re-

placed by the IP traffic volume based charging policy. The flat rate charging policy is inefficient, wastes network resources, allocates cost unfairly and hinders deployment of broadband services [EdVa99]. The IP traffic volume based charging policy, on the contrary, can allocate cost according to users related traffic volume. This can facilitate reasonable network usage, allocate cost to users fairly and stimulate ISPs provide services with better quality. According to the analysis by the "Analysys Research" [Anal06], traffic volume based charging will become dominant in Europe by 2008 [Anal03]. Figure 1.1 depicts the development of pricing structures for broadband in Western Europe.

Although many researches have been made concerning IP traffic accounting and charging [FaSP98, SFPW98, RFC2123, M3I00, KSWS98, KSWS99] and many commercial IP traffic accounting products [Apog, BeSy, Exte, NARUS, NeEy, Netflow, OpSy, XACCT] have been developed, almost all of them utilize the IP address based accounting technique. With the IP address based accounting technique, IP addresses are used to identify the corresponding consumers of the network resources. This is based on the assumption that each IP address is owned by one person or one institute, who or which will be responsible for this IP address. This is correct for IP traffic accounting in single user systems.

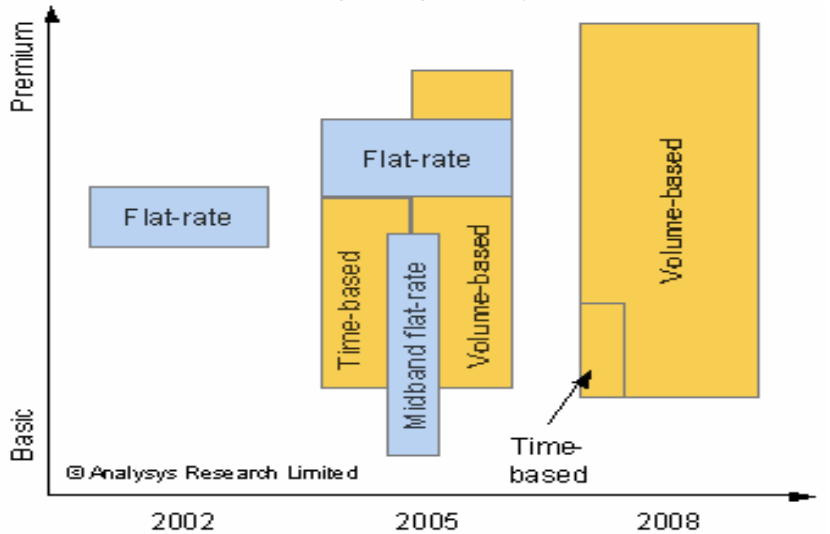


Figure 1.1 Development of pricing structures for broadband, Western Europe [Anal03]

However, there are some situations in which IP addresses cannot be used to identify users uniquely:

- In multi-user systems such as UNIX server, Windows 2000/2003 Terminal Server, etc., an IP address may be shared by several users at the same time. In this case, an IP address cannot be uniquely mapped to one user.
- In some hosts, a user account with special privilege may be shared by different users for the purpose of running applications that require special authorization. The IP address based IP traffic accounting technique cannot distinguish users of the IP traffic generated by different applications in this case.
- Even in a single user system, different user accounts may exist. Although these users cannot use the single user system at the same time, for the IP traffic accounting system the log information about users' login and logout events must be collected to help identify the actual user of IP traffic. Usually this process cannot provide accounting information in real time.
- With the development of distributed computing technology such as Grid computing [FoKT01, FKNT02], Web Services [BHMN04], etc., a single user system may virtually become a multi-user system, since different remote users can run their applications simultaneously in a distributed computing node which is a single user system. In the distributed computing environment, one or more user accounts may be allocated to the remote users for the purpose of getting authorization to run their applications in the distributed computing node. When these applications access the network, the IP address of this distributed computing node cannot be used to identify the originators, i.e. the remote users, of IP traffic uniquely.

Therefore, network resource consumption described above can only be ascribed to the owners of these hosts instead of corresponding users who have consumed the network resources. Consequently, the problem of how to allocate costs of the network resource usage to different users will be left to the owners of the hosts. In fact, however, it is usually difficult for the owners of these hosts to distinguish different users' network resource consumptions. Therefore, fair cost allocation in these systems is practically impossible. This is a flaw of the traditional IP address based IP traffic accounting technique. This is a problem with which institutes, organizations, and especially the computer centers of universities are confronted. With the emergence of Grid computing, Web Services, this problem may even have to be faced by the owners of single user systems.

The IP address based IP traffic accounting mechanism provides coarse granular accounting information, i.e. host level accounting information. In order to amend the insufficiency of the traditional IP traffic accounting

technique described above, the NIPON project [Muel00, NIPO00] was suggested to do research on user based IP traffic accounting techniques for the purpose of providing fine granular IP traffic accounting mechanisms to distinguish the network resource usage among users in the same host. This is also the motivation of this dissertation.

In this dissertation, the user based IP traffic accounting technique is proposed to provide more accurate and finer granular accounting information corresponding to the real network resource usage of different users.

1.2 Contributions

The user based IP traffic accounting technique proposed in this dissertation can provide more accurate and finer granular IP traffic accounting information than the traditional IP address based IP traffic accounting. With the help of the user based IP traffic accounting charging and billing on IP traffic can be more fair and reasonable. Compared with the traditional IP address based IP traffic accounting mechanism, several novel research contributions have been made in this dissertation. They are as follows:

1. The dissertation distinguishes the user concept and proposes the user model for IP traffic accounting. User was an ambiguous and implicit concept in traditional IP traffic accounting. According to the user model proposed in this dissertation, the user based IP traffic accounting concept is clearly defined and explained. The flaw of the traditional IP address based IP traffic accounting can be explained with this user model. This user model can be further extended to provide more accurate accounting information in distributed computing environments, when process ID attributes are applied to identify users.
2. An Accounting Agent mechanism is suggested for the purpose of identifying users of IP traffic.
3. This dissertation proposes three different schemes to make the user based IP traffic accounting technique become reality. These three schemes are:
   a) The in-band scheme utilizes the IP header to convey corresponding user information of IP traffic. A complete protocol mechanism of the in-band scheme is illustrated. In this dissertation, the process of user identification, message format, user information transmission, security communication mechanism, and other issues such as implementation, interoperability with other protocols are carefully designed and defined.
   b) The out-of-band scheme does not utilize the IP header to convey user information. Instead, a packet with user information message is transferred in a special channel between Accounting

Agent and meter separately and asynchronously from normal IP traffic transmission. The Accounting Agent can also function as a standalone meter to generate user based accounting information directly in the measured host.

c) The Multi-IP scheme simplifies a multi-user system to single user systems by assigning different users in a multi-user system unique IP addresses. Through this mechanism, an IP address can be regarded as a user without ambiguity. Therewith the traditional IP address based accounting mechanism can be applied to achieve finer granular user based IP traffic accounting.

4. The dissertation presents the design and development of a user based IP traffic accounting prototype system that realizes the user based IP traffic accounting architecture on the basis of the out-of-band scheme. The prototype system is implemented in two different non-open source operating systems, i.e. in Solaris and Windows 2000 Terminal Server. In this prototype system, the Accounting Agent is designed as a standalone meter to generate user based IP traffic accounting information in a MIB database. The SNMP protocol can be used to collect accounting records in MIB. The test results of the prototype system show that this user based IP traffic accounting prototype system does not affect too much the performance of the measured host in which the Accounting Agent resides.
5. This dissertation also analyzes the necessity and requirements of applying the user based IP traffic accounting technique in distributed computing environments. In distributed computing situations such as Grid computing, Web Services the user model should be extended to use process ID combining with User ID and IP address to identify the users of IP traffic. This dissertation provides the user based IP traffic accounting solution in distributed computing environments.

1.3 Structure of the dissertation

The dissertation is organized as follows:

Chapter 2 Internet accounting technologies survey

This chapter introduces the basic concepts and mechanisms of IP traffic accounting technology. It illustrates the Internet accounting architecture, investigates issues and mechanisms in Meter Layer, Mediation Layer, Billing and OSS/BSS Layer, Management Layer. In Meter Layer, meter location strategies, traffic measurement methods, metrics and format of meter records are analyzed. In Mediation Layer, the methods of collecting meter records, accounting protocols, meter records processing, usage records generation, formatting, storage, and transmission are addressed. In IP Bill-

ing and OSS/BSS layer, the billing models are briefly reviewed. In Management Layer, issues concerning configuration, policies and rules, management interface, etc. are discussed. Criteria for designing and evaluating different layers of the Internet accounting architecture are also summarized. This chapter also discusses the IPv6 and its impact on IP traffic accounting. An overview of the IPv6 is introduced. The IPv6's impact on Meter Layer and Mediation Layer is analyzed.

Chapter 3 User based IP traffic accounting

This chapter gives an overview of the user based IP traffic accounting technique. It analyzes the flaw of the traditional IP address based IP traffic accounting and explains the motivation for user based IP traffic accounting. The related works on user based IP traffic accounting are summarized. This chapter introduces the basic concept of user based IP traffic accounting, proposes a user model and discusses some basic principles required for realizing the user based IP traffic accounting mechanism. In this chapter, the user based network access control mechanism is also addressed.

Chapter 4 In-band scheme

This chapter illustrates the in-band scheme of user based IP traffic accounting. It explains the principle of the in-band scheme, describes the user identification mechanism of the in-band scheme and the process of transferring corresponding user information of IP traffic by integrating user information into IP headers. The format of the User Information option is defined, the principle of choosing the location for integrating the User Information option in IP packet is discussed, and the security mechanism for transferring user information is carefully designed. This chapter also makes considerations on issues such as: where is it suitable for implementing Accounting Agent; how does IP packet fragmentation influence the in-band scheme; how can the in-band scheme coexist with IPSec; performance and dependability issues.

Chapter 5 Out-of-band scheme

This chapter illustrates the out-of-band scheme of user based IP traffic accounting. It explains the principle of the out-of-band scheme, describes the process of transferring user information messages, and defines the formats of user information message. This chapter also depicts the principle of utilizing the Accounting Agent as standalone meter. Security, dependability and implementation related issues are discussed.

Chapter 6 Multi-IP scheme

This chapter illustrates the Multi-IP scheme of user based IP traffic accounting. It explains the principle of the Multi-IP scheme, discusses the IP addresses assigning mechanism, and analyzes issues concerning Multi-IP scheme. In this chapter, a comparison of three different user based IP traffic accounting schemes is made.

Chapter 7 User based IP traffic accounting prototype implementation

This chapter introduces the implementation of a user based IP traffic accounting prototype system. It presents the architecture of the prototype system, describes the details of implementing the Accounting Agent in Solaris and Windows 2000 Terminal Server. The implementation of meter, Reader and Manager are explained. The performance test of this prototype system is analyzed.

Chapter 8 User based IP traffic accounting in distributed computing environments

This chapter introduces the application of the user based IP traffic accounting mechanism in distributed computing environments. The requirements on user based IP traffic accounting in distributed computing environments are analyzed, solutions in meeting these requirements and amending the insufficiency of the traditional IP traffic accounting mechanism are proposed.

Chapter 9 Conclusion and future work

This chapter summarizes this dissertation, and gives a future vision of the user based IP traffic accounting technique.

1.4 Guidelines

To help reading this dissertation we give the following guidelines (Figure 1.2). Readers can choose different flows according to their preferences.

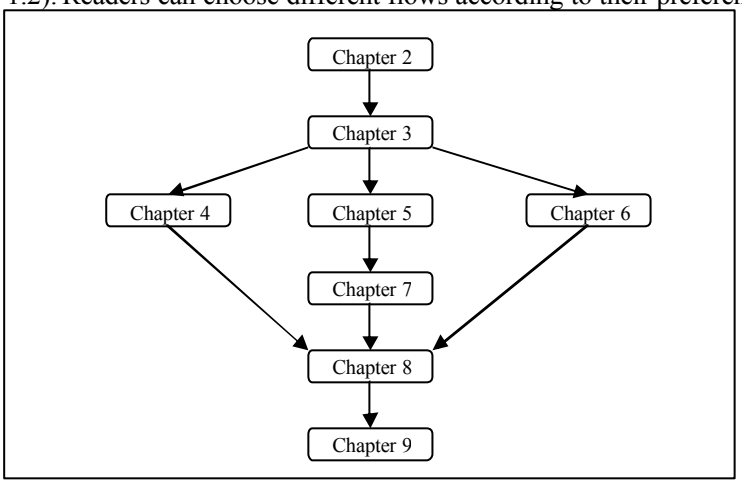


Figure 1.2 Guidelines for reading this dissertation

Chapter 2 Internet accounting technologies survey

This chapter introduces the technologies of the Internet accounting. The common Internet accounting system architecture consists of three layers: Meter Layer, Mediation Layer and Application Layer. The issues in the Meter Layer concern how, where and which traffic should be measured, and how the measured metrics data should be organized, stored and transferred. The technologies applied in the Mediation Layer regard data collection and processing. The Application Layer consists of applications for different purposes such as billing, trend analysis etc. In this chapter the methods, protocols, standards and corresponding comparisons and evaluations concerning Internet accounting will be surveyed.

2.1 Internet Accounting Architecture

2.1.1 Terminology

Internet accounting related terminology has different definitions in different standard organizations such as IETF (Internet Engineering Task Force) [IETF], ISO (International Organization for Standardization) [ISO], ITU (International Telecommunications Union - Telecommunication Standardization Sector) [ITU-T], ETSI (European Telecommunications Standards Institute) [ETSI]. In this dissertation, the definitions by the IETF are adopted.

Accounting

Accounting is the process of collecting and analyzing network service and resource usage metrics for the purpose of capacity and trend analysis, cost allocation, auditing, billing, etc. Accounting management requires that resource consumption is measured, rated, assigned, and communicated between appropriate business entities.

Billing

Billing is the process of consolidating the charging records on a per customer basis and delivering a certain aggregate of these records to a customer.

Charging

Charging means by help of needed functions to determine the tariffs to be assigned to the service utilization (i.e. the usage related elements applicable for charging).

Intra-domain accounting

Intra-domain accounting involves the collection of information on resource usage within an administrative domain, for use within that domain.

In intra-domain accounting, accounting packets and session records typically do not cross administrative boundaries.

Inter-domain accounting

Inter-domain accounting involves the collection of information on resource usage within an administrative domain, for use within another administrative domain. In inter-domain accounting, accounting packets and session records cross administrative boundaries.

Mediation

Mediation is the process of collecting network service and resource usage information from meters, processing the collected data to generate usage records, storing and distributing the usage records.

Meter

A meter is an application or device with metering functions.

Metering

Metering is the process of monitoring, measuring and recording resource consumption.

Multi-user system

A multi-user system is an operating system environment in which IP addresses of the environment are shared by different users at the same time.

Network Element

The network elements are the network devices or application servers that are used to provide communication services.

Rating

Rating is the process of determining the price of the unit service according to the price schemes.

Raw Data Record (RDR) or Metering Record

Raw Data Records or Metering Records are the records containing the measurement results which are generated by meters through monitoring measured objects.

Real time accounting

Real-time accounting involves processing the information on resource usage within a defined time window. Time constraints are typically imposed in order to limit financial risk.

Resource

A quantifiable asset employed by a Service Provider, or on behalf of a Service Provider by another Service Provider, to fulfill a request of a Service Consumer. (Examples include: files, communications, goods, etc).

Single user system

A single user system is an operating system environment in which the IP addresses of the environment are exclusively owned and used by only one user during a period of time.

Usage

Usage is the consumption of resources and services by a user.

Usage Attribute

A Usage Attribute is a parameter whose value indicates some aspect of usage of a given service and/or resource.

Usage Record (UR)

A Usage Record is a data item for a specific user containing information of resource or service usage.

2.1.2 Internet accounting models

The OSI proposed an accounting framework and terminology in [ISO74984]. In "Internet Accounting: Background" [RFC1272] C. Mills et al adopted the OSI accounting framework for the purpose of reporting network usage. The OSI accounting model consists of three basic entities: Meter, Collector, and Application (see Figure 2.1):

- The Meter entity is responsible for network resource consumption measurement and aggregating the results of measurement.
- The Collector entity gathers data from Meter and stores them.
- The Application entity is responsible for processing, formatting and storing Meter data. In addition, the management on Meter is implicitly performed by Application.

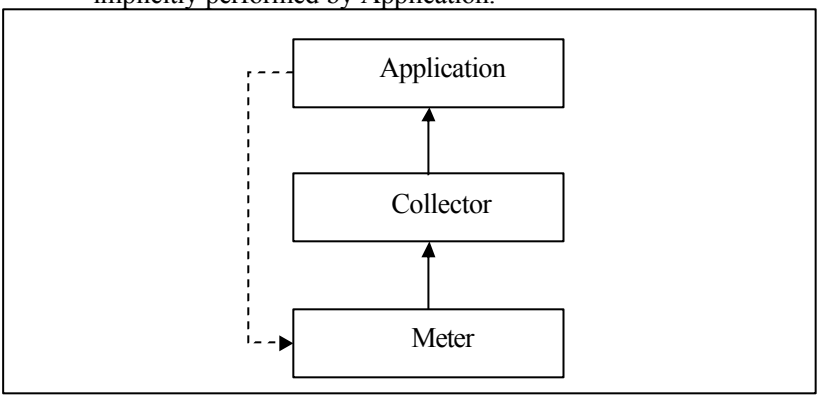


Figure 2.1 OSI Accounting Model [RFC1272]

The Meter collects resource consumption information in the form of accounting metrics. This information is then gathered by a Collector via an accounting protocol. After that, the Collector processes the Meter data to generate the Application required usage records. This process may include eliminating duplicate data, aggregating and correlating related records, calculating interim accounting information, and generating usage records. The generated usage records are then submitted to different Applications,

which may carry out billing, auditing, cost allocation, trend analysis or capacity planning functions.

[RFC1272] proposed the accounting administrative domain concept. An accounting management system deals with and processes accounting related activities in a subset of the Internet, which is the accounting administrative domain. The accounting system cares only about the traffic in its administrative domain or the traffic crossing the boundaries of its administrative domain. On the basis of the administrative domain concept, [RFC2975] proposed the accounting management architecture as Figure 2.2.

In this architecture, the accounting server should have the ability of distinguishing between inter-domain and intra-domain accounting events and routing them appropriately. A Network Access Identifier (NAI) [RFC2486, RFC4282] may be required for the purpose of identifying the accounting administrative domain of the accounting events. Intra-domain accounting events are typically routed to the local billing server, while inter-domain accounting events will be routed to the corresponding accounting servers located in other administrative domains. Proxies can be used for the purpose of forwarding accounting events between different administrative domains.

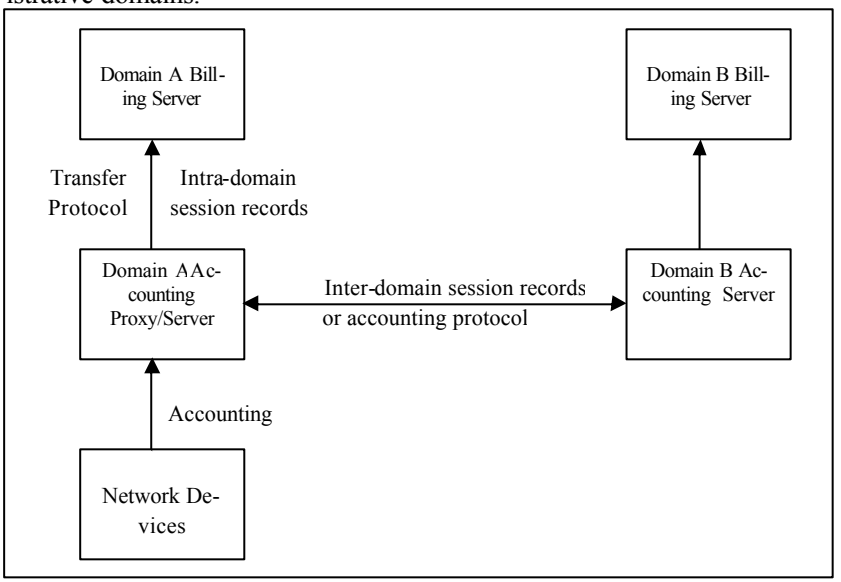


Figure 2.2 Accounting management architecture [RFC2975]

N. Brownlee et al extended the OSI accounting model in "Traffic Flow Measurement: Architecture" [RFC2722]. Compared with the OSI account-

ing model, the traffic flow measurement model in Figure 2.3 adds a Manager component to explicitly configure and control the Meter and Meter Reader entities. With the new added Manager, the accounting system can be managed with a unified pattern. This is better than the implicit management by applications in the OSI Accounting Model which is implementation dependant. The Meter Reader is the same as the Collector in the OSI Accounting Model.

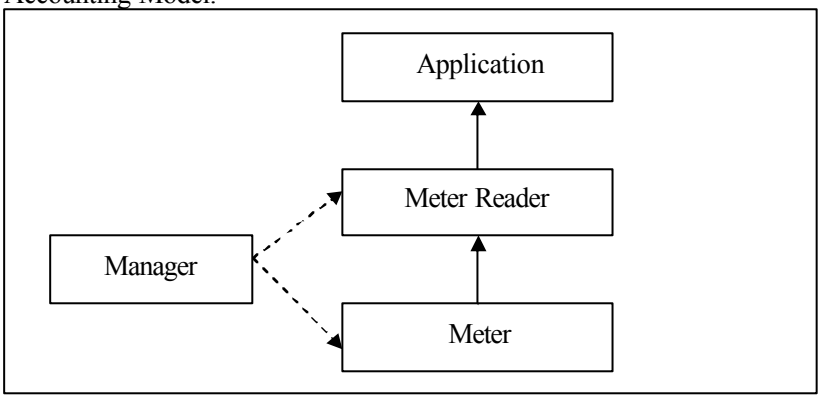


Figure 2.3 Traffic Flow Measurement Model [RFC2722]

After surveying commercial accounting system products such as NARUS, XACCT, etc. [Zhan01], the Internet accounting architecture can be concluded as Figure 2.4.

An Internet Accounting system consists of three layers: Meter Layer, Mediation Layer and OSS / BSS (Operating / Business Support System) Layer. The Meter Layer monitors the network activities and records the measurement results in Raw Data Records (RDR) like an electricity meter. The Mediation Layer collects the Raw Data Records from various Network Elements, and processes the RDRs to produce the Usage Records (UR), stores the Usage Records in a database, and distributes the Usage Records to different applications in the Layer 3. The applications (e.g. Billing, Fraud Detection, Traffic Analysis, etc.) in the Layer 3 process the Usage Records for different application purposes and generate various reports. According to the definition, accounting functions mainly concern Meter Layer and Mediation Layer in the architecture depicted above.

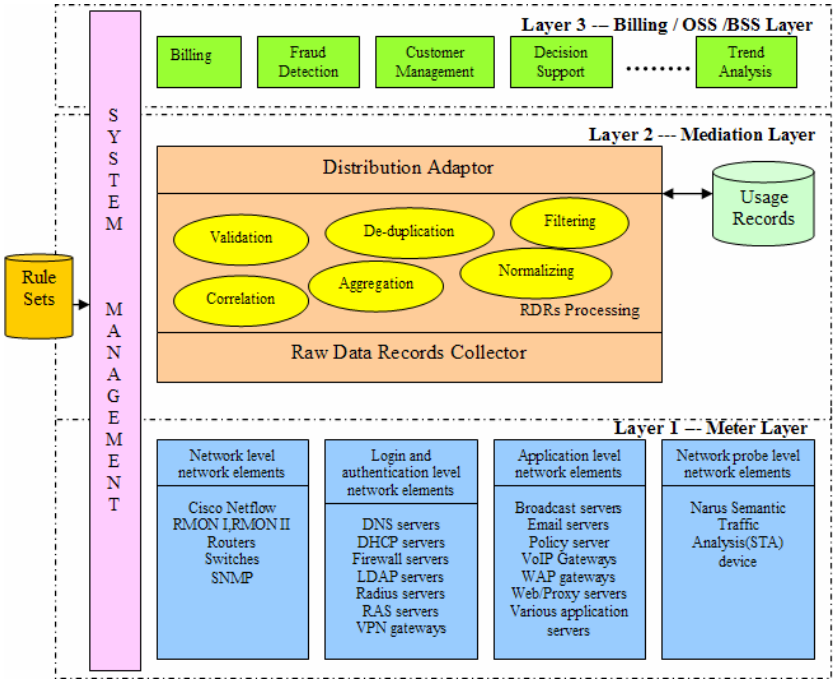


Figure 2.4 Internet Traffic Accounting System Architecture

2.2 Meter Layer

2.2.1 Function requirements for the Meter Layer

The Meter Layer is composed of Network Elements which have the capability of monitoring, measuring and logging information about traffic passing through them. The Network Elements record network activities in RDRs and store them in certain formats. These RDRs can be sent to or collected by the Mediation Layer.

The main functions of the Meter Layer are:

- Monitor and measure the traffic
- Record the measurement information about network activities in RDRs
- Normalize the RDRs into a certain format and store them temporarily
- Provide mechanism to transport RDRs to the Mediation Layer

The meter process can be described as Figure 2.5 below.

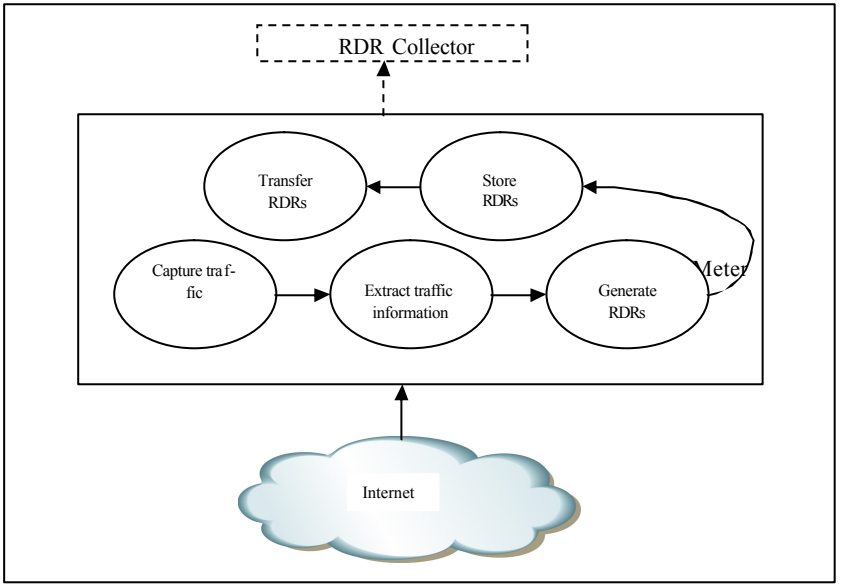


Figure 2.5 Accounting processes in the Meter Layer

2.2.2 Meter location

According to [RFC1272], metering is the process that examines the traffic that takes place in a communication medium or between a pair of communication parties. In order to fulfill the meter functions, the placement choice of the meter plays a crucial role. When choosing meter placement, factors such as availability, accuracy, overhead and efficiency need to be taken into consideration.

2.2.2.1 Availability of network activity information

The availability is the most important factor in choosing meter placement. Meters must be located in the place where corresponding network activities can be watched and all needed traffic information can be collected. Otherwise, not all or even no network resource usage information can be gathered. For example, in order to achieve user based traffic accounting[1] in multi-user systems, not only IP addresses but also user information needs to be gathered. Therefore, in order to collect user information, the meter must be placed into the multi-user host, rather than outside the multi-user host as in traditional IP address based accounting. Outside the multi-user system, no mechanism can obtain user information directly.

[1] For details about user based IP traffic accounting please refer to chapter 4

2.2.2.2 Accuracy and granularity

Accuracy and granularity of the meters are decided by the accounting policies. They can also affect the decision of the meter placement. A Network Address Translator (NAT) [RFC1631, RFC3022] server, for example, can translate private IP addresses to a public IP address. If a NAT server is used by a company to translate IP addresses of its intranet to a public IP address for Internet access, and if a coarser granularity policy is applied for the meter to measure only the Internet traffic generated by this company, then the meter can be placed outside the NAT server (e.g. in the routers with which the NAT server is linked). If a finer granularity policy is applied, e.g. to measure the Internet usage of every Intranet host, placing the meter in the NAT server is the best choice. Figure 2.6 depicts the meter location choices for this NAT server example in the case of coarser and finer granularity policy.

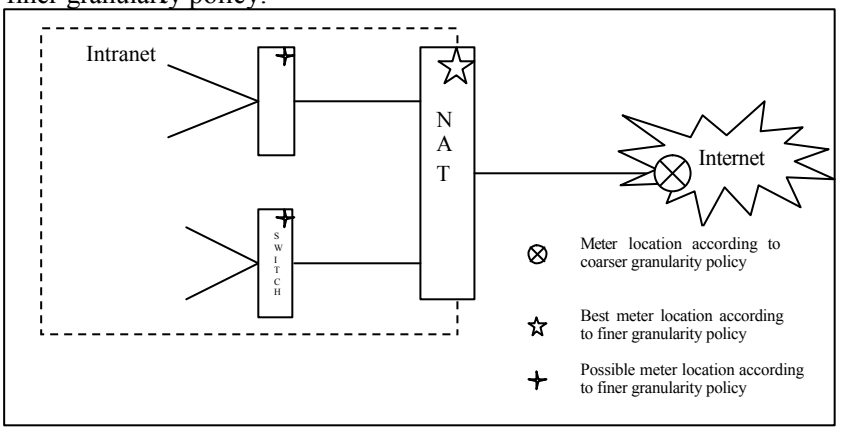


Figure 2.6 Meter location decided by granularity requirement

2.2.2.3 Efficiency and overhead

Theoretically, there are two extreme choices for meter placement: one choice is to distribute meters in all network elements; another choice is to place meters in the locations where all traffic must pass through, e.g. in routers.

The first choice is not efficient since every meter can only measure its corresponding host. Overhead is another problem for the first location choice. Integrating meter function into a host will cause system performance decline, although more or less of the resulted effect on performance depends on the complexity of the meter and the hardware ability of the host [ZhRM05]. Distributing meters in all hosts will also generate extra

traffic when RDRs are collected. Despite of these disadvantages, accurate and finer granularity are advantages of this choice.

Contrary to the distributed meter placement, the second choice of placing the meter in the traffic aggregating center is more efficient. Only one meter is needed to measure all the traffic that goes into and out of a network. This meter location strategy can also decrease the unnecessary traffic related to RDRs collection. Routers are usually located in the position where traffic is aggregated and are suitable places for integrating meter functions. Routers located in the network boundaries are suitable for metering traffic between networks. Therefore, meters are usually placed in or near routers. In this case, the overhead accompanied with the meter functions has to be suffered alone by the network element with meter function. Other two advantages of choosing routers as the placements for integrating meter functions are the traffic control ability and their ability of facilitating intermediate system accounting [RFC1272]. However, the precondition for this choice is that the availability and accuracy requirements can be met.

2.2.2.4 Network topology

Network topology also plays an important role in choosing meter location. Considering to measure traffic generated by hosts in a network segment, if this network segment is designed with bus topology, all traffic can be seen by all nodes connected to this network segment. Therefore, a meter can be placed in any location to watch traffic in this network segment (Figure 2.7a). If a network segment with ring topology needs to be measured, then every node in the ring network segment must be equipped with a meter in order to collect all traffic information of this network segment (Figure 2.7b). Physically, a bus or a ring network can be wired with a star topology by a hub. Therefore, the center position of the star topology is the best location for placing the meter (Figure 2.7c). In addition, for the mesh topology, meters must be distributed into every node to gather all traffic information of the network (Figure 2.7d).

2.2.2.5 Accounting policy

Accounting policies can affect the accuracy and granularity of meters [RFC3334]. Consequently, these policies can affect the choice of meter location indirectly. If the charging policy decides that traffic inside intranet is free whereas the traffic related to the Internet needs to be charged, then meters only need to be located in the boundary routers of the intranet to collect required accounting information of Internet traffic. If intranet traffic must be charged also, then meters must be placed in the corresponding locations in the intranet to gather accounting information.

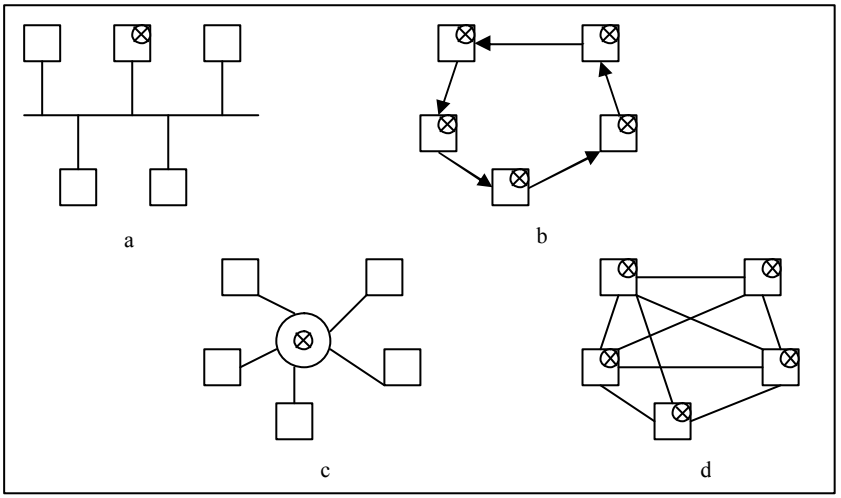


Figure 2.7 Meter location and network topology

2.2.3 Traffic measurement technology

The main task of a meter is to monitor and measure the network activities for the purpose of collecting traffic information. This can be done mainly through two different kinds of monitoring and measurement means, i.e. active and passive methods.

2.2.3.1 Active method

The active measurement method is to watch the response to the simulated network usage activities of actual users for the purpose of obtaining the corresponding statistics information. In Figure 2.8, a Traffic Generator and Traffic Monitor are located outside the measured network. The Traffic Generator sends the simulated traffic into the network. The Traffic Monitor receives the simulated traffic which crosses the measured network and then records the one-way traffic measurement information. The traffic round trip measurement can also be made by configuring a Response Meter to monitor the response traffic.

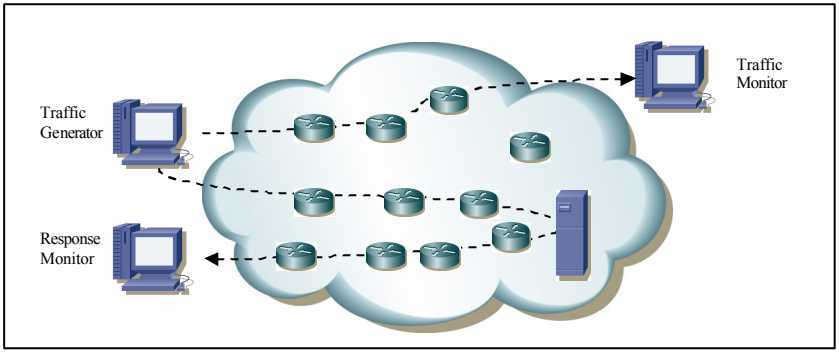


Figure 2.8 Active measurement method

As illustrated in Figure 2.8, the active measurement method utilizes a black-box methodology. The measurement mechanisms are not required to be integrated into the intermediate network elements such as routers. For example, the active method can be used to measure an ISP system without any measurement mechanism being integrated in the ISP system. The active method is usually suitable for measuring network performance such as availability, packet loss, delay, throughput, etc. For example, in order to test the availability of a host, the tool application "ping" can be used to send ICMP Echo request packets to the host. Through checking the response packets, performance information such as the availability of the host, the round trip time, packet loss, etc. can be gathered.

The ICMP protocol [RFC792, RFC2463] provides a feedback mechanism about network problems. Hence, it is often used for reporting network errors and congestions to assist network troubleshooting. "*Ping*" and "*Traceroute*" are two frequently used ICMP tools for testing hosts' connectivity and detecting the packets transmission path, respectively. Even though ICMP suffers from bad reputation from denial of service attacks, ICMP is still the most widely applied active measurement mechanism.

TCP and UDP protocols are also used for active measurement purposes to assess network bandwidth, packet loss, etc. NetPerf [Netp], for example, is an active measurement tool implemented with TCP and UDP protocols.

The active method is simple and easy to be deployed. Its measurement ability does not rely on any measurement capability built in the network devices such as routers. The traffic generation is controllable and predictable on a continuous basis.

The active method has, though, several disadvantages:

1. The first disadvantage is its effect to the performance of the measured network due to injecting extra traffic into the network. In addition, the behavior of the network may be distorted by the simulated traffic.

2. Test packets may be transferred along a totally different path as the real transmission path. This may generate incorrect measurement results.
3. Some types of test packets may be rejected by firewalls or may be processed at low priority by routers. This may also influence the feasibility of this method and also the correctness of the measurement results.
4. The accuracy of this method may be low because the simulation mechanism cannot reflect the statistics of the real world. Consequently, this method is limited to be applied only for network QoS assessment and problem diagnosis.
5. Some periodic events cannot be reflected by this simulation method.

The active method may be not suitable for accurate accounting measurement, since this method can only reflect network performance, but not the network resource usage. However, the information collected through this method can be used as complementary information for Internet accounting.

2.2.3.2 Passive method

The passive measurement method observes and analyzes real network traffic to collect the statistic information [PHGG04]. This method is usually used to measure network performance, real network traffic, resource, and service usage. The passive measurement methods can be classified into three categories: packet based method, flow based method, and sampling method.

1. Packet capturing

In traffic measurement, packet capturing is a key technique to implement the passive measurement method. Packet capturing is a mechanism used to capture packets along their way from source to the destination. Therefore packet capturing can be performed either in two endpoints of the communication pair or in the intermediate points such as routers. Which one is chosen for packet capturing depends on the strategy of the meter location selection described in 2.2.2. Packet capturing in end-systems can be realized with software mechanisms such as libpcap [Libp] in UNIX and LINUX systems and Winpcap [Winp] in Windows systems. In routers and switches, packets can be captured with either Port Mirroring or Network Splitter mechanism. The Port Mirroring mechanism copies all packets passing through a port to a destination port. Through that, the mirrored packets can be collected and processed. This mechanism requires no additional hardware, but the mirror operations will cause processing overhead on the routers or switches. The Network Splitter is a hardware which can

split the signal and then send a signal to the original path and another to the packet capturing device. The advantage of this mechanism is obviously that no processing overhead will be applied to routers or switches, but the additional hardware requirement is its disadvantage.

The challenge for packet capturing is the problems caused by the massive amount of data, especially when capturing packets in backbone networks. Below a data volume calculation example is given.

Assuming packet capturing in a 100Mbps network, the link utilization is 50% and the average packet length is 1000 bytes. The amount of packets that should be captured is:

Packet amount = Link Speed x In & Out x Link Utilization / average packet length

= 100 Mbps x 2 x 0.5 / (1000 x 8)

= 12500 packets/s

If every packet must have an entry to record the measurement result, in each second 12500 entries will be generated. If attributes such as source IP address (4 bytes), destination IP address (4 bytes), source port (2 bytes), destination port (2 bytes) and packet size (4 bytes) are recorded in each entry as measurement metrics, the required total storage space for one second packet capturing is:

Size of records = (4 + 4 + 2 + 2 + 4) x 12500 packets/s = 200000 bytes/s

The packet amount will further explode accompanied by the increment of the port speed or the decrement of average packet size. The experience of capturing all packets in an Internet backbone described in [FDLM01] has shown that the amount of the collected data in 24 hours exceeded 3.3 TB.

From above we can draw the following conclusion: the great amount of captured data requires huge memory or storage space for storing the data temporarily. Consequently, the traffic information entries will grow very fast coordinately to the packet amount and powerful CPU processing ability will be required to analyze and process the captured data. The solutions for these problems include:

- Distributed packet capturing. This can alleviate the pressure from one central captor through distributing the packet capturing operation to more capturing points. Consequently, the requirements on huge storage space and powerful CPU capability can be easily met with the distribution mechanism.
- Design filter policy to avoid capturing unnecessary traffic. For example, filtering all traffic inside an intranet by traffic captors may reduce the data volume significantly.

- Apply flow based measurement. This can greatly reduce the generated table size of the collected information through compressing the entries.
- Utilize sampling method if accuracy requirement can be met.

2. Flow based measurement

A traffic flow is defined as a sequence of packets that share the same protocol and transport layer information between a given source and destination endpoints pair during a period of time [CISCO, RFC2724, RFC3917].

Flows are usually identified by protocol related properties of IP packets between endpoints. For example, an 8-tuple <Source IP Address, Destination IP Address, Source Port Number, Destination Port Number, Protocol number, Type of Service, Start Time, End Time> may be used to identify a flow. Flows may also be distinguished by the protocols. For example, TCP packets are different to UDP packets, therefore TCP and UDP flows can be distinguished even if their source and destination hosts are the same. Flows may have different granularities. 4-tulpe <Source IP Address, Destination IP Address, Start Time, End Time> identifies an end-to-end flow, whereas the above 8-tulpe identifies an application-to-application flow.

A flow may be uni-directional or bidirectional. For uni-directional flows, IP packets from endpoint A to endpoint B and IP packets from the endpoint B to the endpoint A belong to two different flows. Whereas for bi-directional flows, IP packets from endpoint A to endpoint B and IP packets from the endpoint B to the endpoint A belong to the same flow.

The process of the flow-based measurement is: when the first packet of a flow is captured, an entry for this flow is created to record the flow information that is extracted from the packet header; the subsequent packets of this flow are recorded and identified within the same flow entry. When the flow expires, all RDRs of this flow can be exported to the Mediation Layer. Generation of the RDRs is based on decoding and analyzing the packet headers in flows. Cisco NetFlow uses this approach. Figure 2.9 below illustrates the principle of flow based measurement.

Many efforts have been made by standard organizations and companies to provide standards and products for flow based measurement.

The IETF Real-time Traffic Flow Measurement (RTFM) WG [RTFM] has proposed a general flow based measurement architecture [RFC2722] (see Figure 2.3 in 2.1.2). Under this architecture, the RTFM meter MIB, required attributes, etc. are proposed [RFC2720, RFC2723, RFC2724]. NeTraMet, an open source implementation of the RTFM architecture, is also developed by N. Brownlee et al [RFC2123].

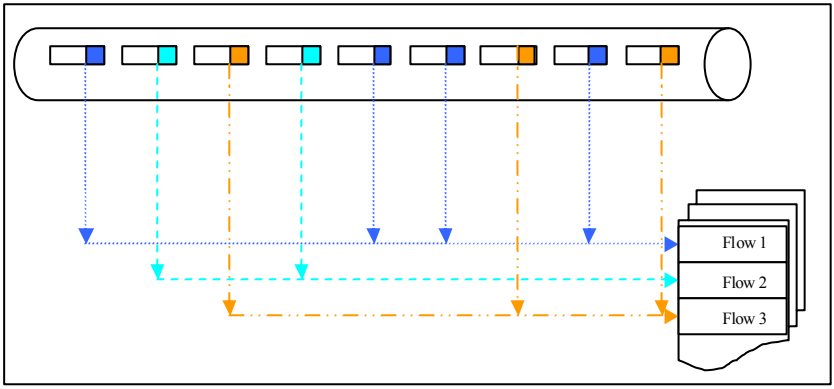


Figure 2.9 Principle of flow based measurement

NetFlow is a flow based measurement software application, which is part of the IOS of CISCO routers and switches. While the NetFlow enabled routers and switches capture flow information from the first packet of a flow to build an entry in the NetFlow cache, information of subsequent packets with the same flow properties is recorded in the same flow entry. The information of flows is temporarily stored in the NetFlow cache, which is managed by the NetFlow cache management software. Expired flows are grouped together into a "NetFlow Export" UDP datagram (see Figure 2.10) for exporting from the NetFlow enabled device. For example the Flow datagram can be exported from NetFlow enabled devices at least once per second, or, as soon as a full UDP datagram of expired flows is available. The NetFlow Export capability can be configured to meet different performance requirements.

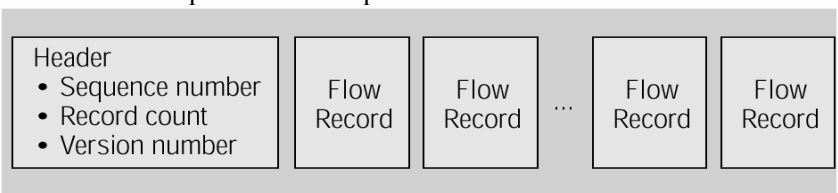


Figure 2.10 NetFlow Export Datagram Format of Version 1, 5, 7, 8 [Netf]

The flow based measurement approach has several advantages:

- With flow based measurement, even if every packet is still captured, entries for every flow instead of for every packet will be created. Through this, the size of generated RDRs will be significantly reduced. Therefore, less memory or storage space will be required and consequently the transmission of these RDRs will result in less overhead on network when RDRs are collected by Mediation Layer.

- Pre-processing the captured packets to form flows in Meter Layer will lessen the processing pressure in Mediation Layer.
- The Raw Data Records are organized on the basis of flows. They can be easily collected and correlated by the Mediation Layer.

Despite of the above described advantages, the flow based measurement also has disadvantages. Flow classification, aggregation, etc. processing operations will affect the performance of the network elements in which the meter functions are integrated. Especially when these processing operations are performed in network devices such as routers, the effect on performance may become more noticeable. For example, according to [LuCo99], enabling NetFlow on a Cisco router can drain CPU usage as much as 30 percent.

3. Sampling

Sampling is the process of systematic or random selection of a subset of elements (the sample) out of a set of elements (the parent population) [RFC3917]. Sampling can be regarded as a discontinuous packet capturing method. Sometimes, it is an alternative method to the continuous packet capturing method if measurement accuracy can be achieved. Sometimes, when traffic rate is too high to capture all packets reliably, sampling is the only choice to provide approximate measurement results. Sampling methods for network traffic measurement have been presented for instance in [ClPB93, CoGi98, DuLT01, Zseb03].

Time based sampling, count based sampling or content based sampling strategies can be utilized for selecting the packets. The count based sampling strategy selects one packet from a fixed number of packets for measuring purpose, whereas the time based sampling strategy sets a fixed time interval to perform the packet capturing operation. The content based sampling strategy selects the packets according to some characteristic parameters in the packets. Sampling packets in which the protocol field of IPv4 header is set to TCP can be regarded as an example of applying the content based sampling strategy. The concrete sampling strategy is decided by the measurement policy and can be configured during the measurement process.

The sampling method provides an average or statistic level measurement result. This method is suitable for measuring video, audio, etc. data transmissions in which the transmission rates are stable. In these cases, sampling results can provide near real results. However, sampling cannot provide accurate measurement result, and sometimes it may even miss valuable data. For example, if the time interval is not configured suitably (e.g., the time interval is too large), some burst packets cannot be captured. Therefore, sampling is not suitable for accurate accounting, charging and billing purposes.

2.2.4 Measurement metrics

The measurement metrics are the properties and quantities which can be measured and reported to reflect the status of the measured objects. They designate what kind of attribute information of the measured object should be collected. What should be measured by meters is decided by accounting policies. Different application areas, different charging and billing policies require different measurement metrics. The measurement metrics is also decided by the measurement accuracy and granularity. Time based flat rate charging policy for dial-in Internet service, for instance, requires only recording the login and log-out time of users whereas volume based charging policy for the same dial-in Internet service requires recording how many bytes are consumed by users.

Many accounting attributes have already been described in IETF and ITU-T documents [RFC2866, RFC3588, RFC2720, RFC2512, RFC2212, ITU-T Q.825]. Generally, the traffic measurement metrics fall into three categories:

1. Identification attribute

Identification attributes indicate the classification property or the owner of a meter record. Usually they are used as index for meter records aggregation. The Source IP address, for example, is generally used as an index for aggregating records to accumulate calculable information of the same host. In this case, the Source IP address identifies the host uniquely. Several attributes can be grouped together as identification attributes to identify a meter record. An example is the identification attributes for a flow. 8-tuple flow identifier <Source IP Address, Destination IP Address, Source Port Number, Destination Port Number, Protocol number, Type of Service, Start Time, End Time> can be regarded as an identification attributes group.

2. Statistic attribute

Statistic attributes are the quantifiable characteristics reflecting the statistic of the measured objects. The value of the statistic attributes can be generated and updated through calculation operations. These attributes may be updated continuously reflecting the changing status of the measured objects until the meter records are collected by the Mediation Layer. For traffic measurement, bytes sent, packets received, time duration, etc. are normal statistic attributes.

3. Status attribute

Status attributes are the attributes that describe the status, types, etc. unquantifiable characteristics of the measured objects. These attributes can be used by applications in OSS/BSS Layer for qualitative analysis or processing. For example, the “ Acct-Terminate-Cause” attribute defined in

RADIUS Accounting attributes [RFC2866] can be considered as a status attribute, since it records the reason of the session termination and may be a valuable parameter for charging and billing. Identification attributes and status attributes can be overlapped. Still using the Source IP address as an example, it can be used as an identification attribute for the purpose of uniquely identifying a flow and it can also be regarded as a status attribute.

An example of classifying accounting attributes defined in several accounting documents is given below:

Document	Identification	Statistic	Status
RADIUS Accounting [RFC2866]	Acct-Session-Id Acct-Authentic Acct-Multi-Session-Id	Acct-Input-Octets Acct-Output-Octets Acct-Session-Time Acct-Input-Packets Acct-Output-Packets Acct-Link-Count	Acct-Status-Type Acct-Delay-Time Acct-Terminate-Cause
RTFM [RFC2720] [RFC2724]	(Source Address) Source Peer Address Source Trans Address … (Destination Address) Destination Peer Address Destination Trans Address … Rule Set Number Source SubscriberID Destination Subscriber ID Session ID	Forward Bytes Forward Packets Reverse Bytes Reverse Packets QoSRate QoSSlackTerm QoSTokenBucketRate QoSTokenBucketSize QoSPeakDataRate QoSMinPolicedUnit QoSMaxPolicedUnit	First Time Last Active Time Source Class Destination Class Flow Class Source Kind Destination Kind Flow Kind QoSService QoSStyle

The measurement metrics usually depend on the application area of the accounting, the charging policy, Service Level Agreement (SLA).

Through analyzing accounting standards, we can find that different application areas of accounting may have different measurement attributes. However, the same attributes may be defined with different names. Standards need to be defined to facilitate the compatibility among different standards. That is why IPFIX WG [IPFIX] was built to unify the report. Otherwise, the adaptation mechanism must be built in the Mediation Layer for the purpose of merging these records together following different standards.

The charging policy also plays an important role in deciding the measurement metrics to be used. Flat-rate charging policy does not require any traffic measurement. Different usage based charging policies require different measurement metrics. Duration based charging requires network usage time to be recorded, volume based charging policy requires the sent and received bytes to be recorded and calculated, whereas the content based charging policy requires what kind of information is consumed to be recorded.

As the Internet is becoming a convergent service platform, Service Level Agreements (SLA) are more and more used by service providers to define contracts for providing services to customers. The SLA signed between a service provider and a service subscriber prescribes not only legal obligations for both sides but also the agreed QoS parameters. Monitoring and controlling the fulfillment of the agreed QoS are interested by both service provider and consumer. QoS parameters of different services defined in SLAs construct the measurement metrics. With the emergence of more and more new services, each service may have its own QoS parameters. Different attributes require different measurement techniques. As [HaRa01] described, the measurement activities can be classified into four levels: traffic level, network traffic, system level, client side application measurement and application wide measurement.

2.2.5 Raw Data Record format

Raw Data Record (RDR) format specifies how meter information is organized and encoded into a record. Usually the decision of choosing what kind of RDR format depends on the accounting protocols. If an accounting protocol is chosen for conveying the RDRs, the RDRs' format must be compatible with the requirement of the accounting protocol. Otherwise, the RDRs must be converted before transmission. Certainly, proprietary formats may be used or defined by different vendors, but this may cause compatibility problems in exchanging accounting information.

A survey of different accounting protocols showed that the accounting record formats can be classified into several categories:

- ASN.1 format.

  It uses ASN.1 Basic Encoding Rules (BER) to encode lists of attributes into a record. For instance, SNMP MIB, RTFM and AToM-MIB use this format. This scheme can fit well with SNMP based network management systems and provides a good way to record the network activities. However, the structure of ASN.1 based scheme is very complex. Purpose build tools are needed to deal with these structures.

- Binary format.

  It uses octets to encode the attributes into records. RADIUS and DIAMETER use this type of record format. The structure of this scheme is simple, but it also needs to be processed with purpose build tools. Furthermore, its extensibility is not good.

  RADIUS utilizes <Type, Length, Value> triple format depicted in Figure 2.11 to construct accounting records.

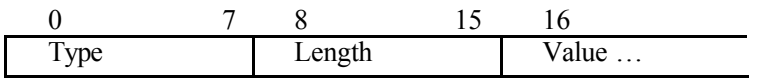


Figure 2.11 Accounting record format used by RADIUS

DIAMETER uses a similar Attribute Value Pair (AVP) format to assemble accounting records. Figure 2.12 below depicts the AVP format of DIAMETER.

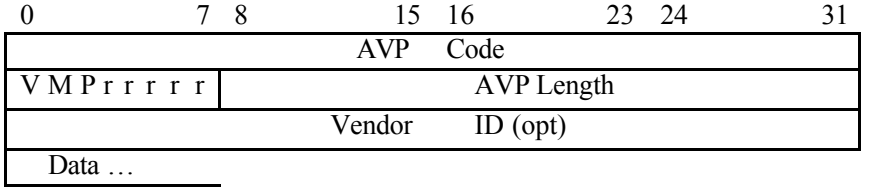


Figure 2.12 AVP format of DIAMETER

- Plain Text format.

  It uses ASCII text to encode attributes into a record. Accounting Data Interchange Format (ADIF) is an example of this kind of record format. It was proposed in [AbLi01] and aimed at becoming a standard accounting record format. ADIF is a text based format described in BNF grammar and it is designed to compactly represent accounting data in a protocol-independent manner. It is easy to read and understand the records in this format.

- Tagged Text.

  Tagged Text format has not only the advantage of good readability but also enhances the flexibility and extensibility of record structure. Compared with the record structure in which the data type of a record is implicitly defined by its position, tags can explicitly identify the data type of a record. Therefore, the record format alteration such as removing or adding data elements can be easily performed without affecting the record structure. IPDR [IPDR06], TIPHON [TIPHON] are examples using XML tag based format. The size of files or records with this format might be bigger, but compression can be used to reduce the file size before storage or transport.

Different accounting record formats cause problems for sharing accounting information among different accounting systems. With the achievement of new techniques in areas such as wireless communication, Web Services, Grid computing, etc., inter-domain accounting tends to be increasingly required. Standard accounting formats will facilitate the accounting information exchange among different administrative domains. It is generally agreed that a standard record format is needed for communication between service elements and accounting servers. In order to develop this kind of standard record format, issues like separating accounting re-

cord format from accounting protocols, standardizing data types, compactness, extensibility, etc. must be taken into consideration. The ADIF is an effort by the IETF for this purpose, although the ADIF still needs to be further developed. It is designed not only for meter devices storing and transferring accounting information with the ADIF format (in the Meter Layer), but also for accounting servers generating ADIF format data after processing meter records gathered by accounting protocols (in the Mediation Layer). The IPDR is another effort on standardizing accounting information formats made by the IPDR organization. It uses XML based format for encapsulating accounting attributes into records. Due to lack of compactness of the XML based format, the IPDR format is more suitable for representing Usage Records in Mediation Layer. For the introduction about the IPDR format, please refer to chapter 2.3.4.1

2.2.6 Category of meters

The Meter Layer includes different kinds of network elements that can be used for metering purposes. These network elements are provided by different vendors, and most of them are not designed only for metering purposes. Therefore, the approaches of measuring traffic and the formats of Raw Data Records are different. According to [Schw99], till September 1999 there were about at least 160 types of network elements. These network elements can be classified into three categories: traffic based meter, application level meter and probe meter.

2.2.6.1 Traffic based meter

Traffic based meters are network devices that are integrated with meter functions. Routers, although they are applied for traffic forwarding purposes, can be configured to provide traffic meter functions. They can monitor every packet passing through them and extract accounting information from packet headers to generate RDRs. Traffic based meters provide only network level accounting information.

Below are some network elements that can be used as traffic based meters:

- Cisco NetFlow capable devices
- RMON I, RMON II devices
- Routers
- Switches

The traffic based meters have several advantages:

- They provide finer granularity of the resource utilization information. The RDRs include attributes such as source and destination IP addresses, source and destination port numbers, packets

and bytes count, timestamps, type of service, etc. These attributes can be employed for usage based accounting.

- The RDRs may be organized on flow basis. They can be easily collected and correlated by the Mediation Layer.
- Since meter functions are integrated into the network devices, event driven RDR collection mode can be supported. This can improve the real time capability of IP accounting.

The traffic based meters also have some disadvantages:

- Since traffic based meters usually work in the network layer, they cannot provide detailed information about application level usage. For example, although NetFlow can record the Type Of Service (TOS) attribute, it cannot provide more detailed attributes of the services it records.
- The traffic based meter function is usually embedded in network elements such as routers. Since the network elements are in general not designed for dataflow metering purposes, such a function may affect the performance of the network elements [LuCo99].

2.2.6.2 Application level meter

The Internet provides not only communication services but also many value added services. Traffic based meters can only provide meter function in the network level. Application level meters can monitor application service usage and generate corresponding RDRs.

Application level RDRs can be generated with either passive or active methods. The passive method utilizes log files generated by application servers while providing services. Usually log files are created for the purpose of debugging or archiving, but they contain application service consumption related information which can then be used partially or totally for accounting purposes.

The main advantage of the log file based meters is that application servers have the capability of logging the service usage events when they provide various services. No special meter functions need to be integrated. This is suitable for measuring legacy systems without modifying them. However, this type of meters has several limitations:

- Completeness: Log files are not designed for the purpose of accounting, so the log files from one network element cannot capture and record enough information about the customer activities. Information about a single session may be composed of several attributes that must be extracted from several log files in different network elements.

- Real time: There is no event driver mechanism in the log file based meter. The network elements only record the network activities in the log files. They cannot inform other systems about their new log records. So the log files should be scanned periodically, this cannot meet the requirements for realtime capability.
- Different formats of log files: Different types of network elements or even the same type of network elements produced by different vendors have different formats of their log files. A custom interface must be added to process these log files of different formats. The Mediation Layer has to be flexible enough to process these formats.
- Data redundancy: Since most of the log files are not designed for accounting, the log files may include many useless data for IP Billing; these redundant data must be filtered before or after they are sent to the Mediation Layer.
- Coverage: Many application sessions can take place from client to client. In this situation, no server or middle node may exist for logging the network activities. For example, Microsoft NetMeeting [Netm] can provide the communication between two PCs without using a server. In this case, there is no network element to log the information about the communication between the two PCs.
- Maintenance: Log files are stored in disks. With increasing size of the log files, it may be necessary to backup or delete log files in order to avoid running out of disk space. These processes must be synchronized with the Mediation Layer, but there is no mechanism to do this. For example, if an application server wants to delete a log file after backing it up, it must be known if all Raw Data Records in this log file have been collected by the Mediation Layer, before the application server finally deletes the file.

Another method is integrating meter functions into application servers. The application servers can generate meter RDRs, and then accounting protocols can be used to gather these RDRs. Advantages for this method are:

- More accurate service oriented RDRs can be generated. This can facilitate usage based accounting.
- With integrated meter functions, application level meters can support event driven RDRs collection; therefore, real time accounting can be achieved.
- With integrated meter functions, the RDRs can be managed efficiently.

The disadvantage of this active method is that the integrated meter function in application servers may affect their performance.

Some application servers that can provide application level meter functions are listed below:

- DNS servers – associate host names with IP addresses.
- DHCP servers – assign IP addresses to users dynamically while they log on.
- Firewall servers – ensure security of private networks.
- LDAP servers – directory service to manage user accounts.
- RADIUS servers – provide information on individuals logged in via modems or ISDN connections.
- RAS servers.
- VPN gateways
- Broadcast servers – support music, video, games or education on demand.
- Email servers – log information about each email sent or received across the corporate network or via the Internet.
- Policy server – implement routing policy on request from subscribers, requiring quality of service through the network.
- VoIP Gateways – support video conferencing or VoIP over LANs, intranets and the Internet.
- WAP gateways – log information about the activity of each WAP session handled by the WAP gateway.
- Web/Proxy servers – log access activities.

2.2.6.3 Dedicated meter

Traffic meters and application meters are adhered to network elements which are originally not designed for accounting purpose. Therefore, they can only provide limited meter functions and may cause performance decline problems to corresponding network elements. Dedicated meters are network elements designed specifically for accounting purposes. They are usually probe like devices placed at a key position in a network. They can capture IP packets through the network, extract the headers and specific data fields, and analyze the protocols of all layers to generate RDRs. Therefore, dedicated meters can provide comprehensive network resources, and services consumption information.

For example, the STA device of the Narus uses this approach [NARUS]. The STA device is a typical kind of network probe device; it uses a session based semantic analysis technique. The STA device can detect, analyze and characterize contextual information transacted by an application during an identifiable period. Based on industry standard and proprietary protocol semantics, this device builds a statistical repository of all sessions per user over time.

The dedicated meters have several advantages:

- They can capture all packets passing through the meter, they can analyze all seven layer protocols and they can record all information about the network activities and services.
- They do not rely on other network elements to record information. They do not disturb the operation of the network elements.
- The usage information is collected in real time. Therefore, they can meet the real time requirements of the Mediation and OSS Layer.
- The RDRs can be recorded on the basis of sessions. It is easy for the Mediation Layer to generate the Usage Records for users.

The dedicated meters also have some disadvantages:

- They are additional network elements that must be invested, installed and maintained.
- A large scale network requires more probes to be deployed to record all traffic flows. These probes must be synchronized to remove duplicate events. This is a tough problem requiring a well considered solution.
- Only existing standard applications can be correctly analyzed and measured. New protocols of custom designed applications may be hard for dedicated meters to be measured correctly.
- Application semantics may be difficult for external meters to understand. Especially when network traffic is encrypted, meters outside the application servers cannot get any usage information.
- If the dedicated meters are located in high speed networks, massive traffic volumes will make real time measurement very difficult.

2.2.7 Criteria for meter design

Most of the network elements that are used as meters are not only designed for generating meter RDRs. Therefore, there are no uniform criteria for evaluation of meter products. If a network element meets some basic requirements for a meter, it can be used as a meter.

The basic criteria for a dataflow meter are:

- Capability of measuring and recording one or more types of network activity or service.
- Capability of generating Raw Data Records and storing them for a period of time in some format.
- Provision of a communication mechanism to support collecting Raw Data Records by the Mediation Layer devices.

For measuring and recording the network activities better, meter products have to meet more advance requirements:

- Real time capability: recording network activities in real time, and reporting RDRs to the Mediation Layer in real time or supporting RDRs collection by the Mediation Layer in real time.
- Minimal impact on the performance of the network elements: Since most of the network elements with meter functions are not designed for the purpose of measuring and recording network activities, the meter functions should not cause too much performance decline to the network elements.
- Reliability: If failures of network or network elements occur, mechanisms, which can prevent from loss of recording network activities, have to be provided. For example, the RDRs have to be stored before they can be collected by Mediation Layer devices. Therefore, the non-volatile storage for undelivered Raw Data Records is a good way to achieve reliability. The reliable mechanisms to transfer RDRs should be applied in order to avoid data loss. During the process of transporting RDRs to the Mediation Layer, a retransmission mechanism needs to be introduced. The Raw Data Records ought not to be deleted before a confirmation is received that they have been collected by the Mediation Layer.
- Configurable rule sets for the generation of RDRs: the configurable rule sets define how to record network activities, which attributes should be recorded, and how to generate the Raw Data Records. This can help the dataflow meter to be more flexible in recording network activities.
- Capable of measuring and recording usage sensitive information: since usage based billing will be more popular in the future, meters should provide enough parameters about the measured network activities to meet more accurate requirements.

2.3 Mediation Layer

Since Internet accounting systems were first built on the basis of traditional telecommunication accounting technology, some of the concepts from telecommunication accounting technology are still used today for Internet accounting. The term “Mediation” in the traditional telecommunication field means taking data off the switch and capturing it. The data is then passed to a rating system to calculate the actual charges. The Internet mediation process is similar to the traditional telecommunication mediation, but the Internet mediation is more complex than the one that is used in traditional telecommunication. The Internet mediation layer first collects the RDRs from various network elements, and then, according to mediation rules, filters, de-duplicates, merges, correlates, aggregates, and nor-

malizes the collected RDRs to generate the Usage Records (UR) in some format, stores the URs, and finally distributes these formatted URs to different applications in the OSS Layer for different application purposes.

2.3.1 Functional requirements for the Mediation Layer

The main functions of the Mediation Layer are:

- Collect RDRs from different meters in the Meter Layer
- Process collected RDRs
- Generate URs in different formats
- Store URs
- Distribute URs adaptively to different applications in the OSS Layer

2.3.2 Meter data collection

The meter data collector is responsible for collecting RDRs from various types of meter devices. The following factors must be taken into consideration when using the meter data collector: interfaces with different types of meters, location of meter data collector, RDR collection modes, and accounting protocols choice.

2.3.2.1 Interfaces with different types of meters

Due to the diversity of meters, collectors should have the capability of interacting with different types of meters, adapting to different formats of RDRs, gathering RDRs according to different accounting policies. A collector device can have one or more interfaces with corresponding meters. In order to collect RDRs from different meters, more than one collector device are usually required to be deployed in large networks.

2.3.2.2 The placement of the collector

Three principles should be taken into consideration in choosing the placement for collector deployment:

- Completeness: Enough RDRs can be collected according to the mediation rule sets. This means that the collected RDRs can be used sufficiently to generate URs to meet the requirements of the applications in the OSS Layer. Therefore, usually the collector devices are located in key places, where enough RDRs can be collected, but less redundant or useless RDRs will be collected.
- Low overhead: Minimal traffic on the network. This means that the RDRs collection should not cause too much impact on the performance of the network. Therefore, the collector devices are usually located close to the information sources.

- Efficiency: The collector should also be located in the place where RDRs can be gathered in time, so that the requirement for real-time accounting can be met.

2.3.2.3 Approaches of Raw Data Records collection

From the collector's point of view, the RDR collection approaches can be classified into three types of modes: push, pull and hybrid mode. With the push mode, the meters can send RDRs to collectors actively, whereas with the pull mode the collectors inquire meters actively for RDRs. The hybrid mode is a combination of push and pull mode. With the hybrid mode, both meters and collectors can start the data collection process actively according to configured rules.

There are four data collection models which are used today by meter data collectors: polling model, event- driven model without batching, event-driven model with batching, event-driven polling model. These four models implement the push, pull or hybrid approaches, respectively [RFC2975].

1. Polling model

With the polling model, a collector polls meters for RDRs at regular intervals. The polling interval should be properly configured against loss of data. The maximum polling interval is determined by the available memory for meters without non-volatile storage and by the size of non-volatile storage for meters with non-volatile storage.

With this model, the collector needs to poll all managed meters periodically. In some cases, many meters may contain no relevant data during a period of time. Another problem of this model is the latency. The usage of the interval implies an average latency for each meter, which might be too high for RDRs that require low processing delay. Therefore, this model is not suitable for real time RDRs collection.

2. Event-driven model without batching

In the event-driven model without batching, a meter will generate an event to inform the collector when the meter is ready to transfer RDRs. This model offers the lowest latency since events are processed immediately. This model can be used for real time RDRs collection.

Event-driven without batching usually transfers one event per packet, so this model is inefficient.

3. Event-driven model with batching

In the event-driven model with batching, a meter will inform the collector when a batch of a given size RDRs has been gathered, or when RDRs of a certain type are available or after a minimum time has elapsed. With

this model, more than one RDR can be transferred in an event packet and consequently this model is more efficient.

Since the event-driven model with batching usually triggers RDRs to be sent to the collector after a batch of RDRs is prepared, the latency of this model is lower than the polling model but higher than event-driven model without batching. With the help of implementing a schedule algorithm, event-driven model with batching is able to deliver urgent events to the collector immediately. For example, high-value RDRs can be sent at once without batching, while all other RDRs will be batched. With this approach, this model can be used for the real time RDRs collection.

4. Event-driven polling model

According to the event-driven polling model, a collector will poll the meter for RDRs only when it receives an event. The meter can generate an event when a batch transmission condition is triggered.

Compared to the non event driven polling model, whenever the collector polls the meter, the meter already has RDRs to be sent. The blind polling without any result can be avoided and therefore the efficiency can be improved.

Since this model needs at least two round-trips to deliver RDRs, i.e. one for the event notification and another for the resulting poll, the latency in this approach is higher than that in the event-driven model with batching.

2.3.2.4 Real time Collection Consideration

Real time RDRs collection is the precondition of real time IP Billing. Different RDRs collection methods have different realtime characteristics.

- The pure batch approaches have poor real time capability. The RDRs will be collected only when a certain volume or time limitation is reached. Therefore, the RDRs collection is hard to be synchronized with the network activities.
- The polling approaches can have near real time capability. This approach collects RDRs at regular intervals. If the interval is set short enough, the nearest synchronization may be reached with the network activities.
- The event driven approach can collect RDRs in real time. An event will be sent to the collector at once when RDRs are generated by the network element. Then the collector is able to collect the RDRs. This approach has good real time capability.

2.3.2.5 Accounting protocols

Accounting protocols are used to convey meter data for accounting purposes [RFC2975]. Accounting protocols define the specifications of trans-

ferring meter records from Meter Layer to Mediation Layer. The rules for transferring meter data, exchanging message format and accounting attributes are prescribed in the accounting protocols. Usually accounting protocols utilize the client server model. An accounting client is generally a meter, which performs monitoring and measurement operations to generate meter records and reports them to the accounting server. An accounting server works in the Mediation Layer and is responsible for collecting and storing RDRs.

Several international standard organizations such as the Authentication Authorization Accounting (AAA) Working Group of the IETF [AAAWG] and the Authentication Authorization Accounting ARCHtecture (AAAARCH) Research Group of the IRTF [A4RCH] have made efforts in constituting Internet accounting related protocols. The following protocols are of major importance in the Internet accounting area: RADIUS, DIAMETER, and SNMP.

1. RADIUS

The Remote Authentication Dial In User Service (RADIUS) protocol [RFC2058, RFC2138, RFC2865] was developed by Livingston Enterprises for exchanging authentication, authorization, and configuration information between a network access server (NAS), which operates as a RADIUS client, and a RADIUS server. [RFC2059, RFC2139, RFC2866] extended the RADIUS basis protocol to support accounting. This accounting protocol prescribes accounting packet format, accounting attributes, and how the accounting information is transferred. RADIUS operates in a client server mode for authentication, authorization and accounting. Authentication, authorization and accounting information details are carried by RADIUS attributes which are encoded in a type-length-value format. The flexible authentication mechanism of RADIUS can support multiple authentication methods such as CHAP and PAP. The communication security is guaranteed by encryption and shared secret key mechanisms. Figure 2.13 below depicts an example of the RADIUS accounting process.

The accounting function of the RADIUS protocol can work independently from RADIUS authentication and authorization. At the beginning of a session, an Accounting Start packet with type of service information is sent from a RADIUS client to the RADIUS server. Then, the server sends back an acknowledgement packet to the client. After that, the NAS can deliver service to the end-user. At the end of the session an Accounting Stop packet with information such as type of service, input and output octets, input and output packets, elapsed time, etc. is sent from the RADIUS client, i.e. NAS, to the server, which in turn sends back an acknowledgement packet. Through this process, the accounting information about the resource consumption of this session can be recorded.

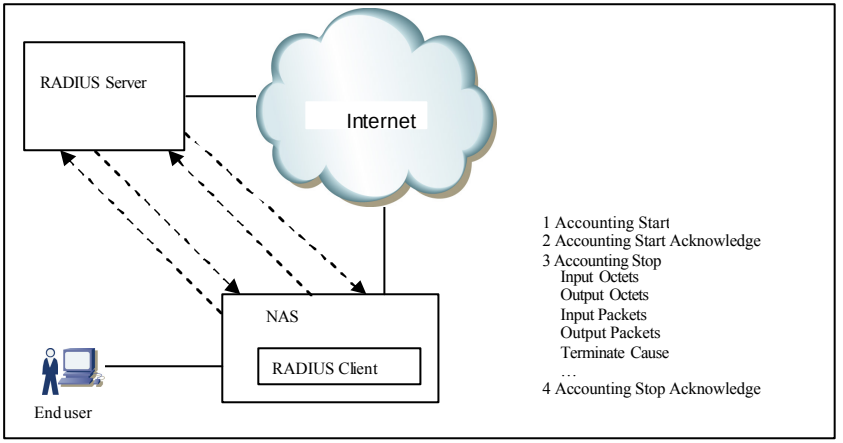


Figure 2.13 Accounting process of RADIUS protocol

After it was published by the IETF, the RADIUS protocol has attracted a wide range of customers, especially ISPs. It is a widely deployed Internet accounting protocol [Hass03]. Despite of the wide application of RADIUS, it has several drawbacks:

- The extensibility of RADIUS is restricted by its limited command and attributes space [RFC3423].
- UDP chosen by RADIUS for accounting information transmission cannot provide reliable transfer service.
- RADIUS encrypts only the password in Access-Request packet. Other information in accounting packets is not encrypted. Security is thus a problem for RADIUS.

2. DIAMETER

Although RADIUS has been widely deployed and supported by ISPs and enterprise network managers, it cannot meet the further requirements of providing AAA services for hundreds and thousands of concurrent end users accessing network services over a variety of technologies. In order to eliminate the inherent deficiencies of RADIUS, the DIAMETER protocol [RFC3588] was defined as a successor to RADIUS. The DIAMETER protocol was developed to provide a framework for Mobile-IP [RFC3344, RFC3775] and NASREQ [RFC2881, RFC2882]. It is a lightweight peer based AAA protocol which can be used for AAA, policy and resource control [Metz99]. Moreover, DIAMETER has been chosen as the next generation AAA protocol by the AAA Working Group of the IETF [PBSP01].

DIAMETER was designed to be compatible with RADIUS. Many mechanisms in RADIUS, such as encoded attribute value pairs (AVP), proxy server support, etc., are adopted by DIAMETER [EkSP00]. The

DIAMETER base protocol provides facilities such as delivery of AVPs (attribute value pairs), capabilities negotiation, error notification, extensibility, user sessions, accounting, etc. DIAMETER accounting is based on the server directed model with the capabilities of transferring accounting information in real time. Batch accounting is not supported by DIAMETER. The DIAMETER base protocol did not define application service related accounting attributes except some mandatory AVPs such as Session-ID that must be included in the accounting records. Service specific AVPs need to be defined in separate DIAMETER application documents. In order to provide reliable transport, the Stream Control Transmission Protocol (SCTP) is chosen by DIAMETER as a transport protocol. TCP can also be used. Improved retransmission and a fail-over scheme have also been employed in [RFC3588].

Figure 2.14 below illustrates an example of the DIAMETER accounting process.

1. Authentication: The DIAMETER client (e.g. a NAS) sends an AA-Request with username/password pair of the remote user to the DIAMETER Server.
2. Authorization: The DIAMETER server checks the password of the user. If it is valid, then it sends an AA-Response with authorization information (e.g. IP-address, network mask, allowed session time, etc.) to the DIAMETER client.
3. The DIAMETER client sends an Accounting-Request (ACR) command including AVPs (e.g. a unique Session-ID, Accounting-Record-Type set to START_RECORD) to the DIAMETER server.
4. The DIAMETER server replies with an Accounting-Answer (ACA) command including the same Session-ID and corresponding AVPs to acknowledge the accounting request.
5. When the user logs out, the DIAMETER client sends an ACR command with Accounting-Record-Type set to STOP_RECORD to the DIAMETER server. The ACR command with STOP_RECORD type is sent to terminate an accounting session and contains cumulative accounting information relevant to the existing session.
6. The DIAMETER server replies with an ACA to acknowledge the accounting request.

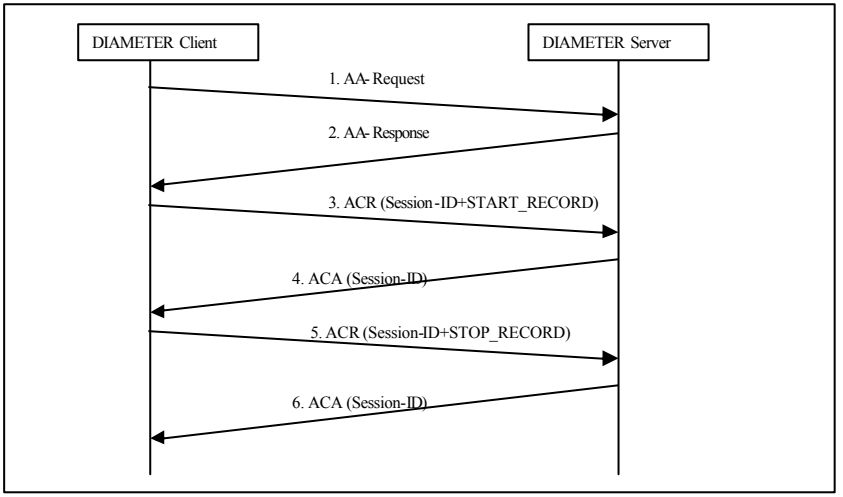


Figure 2.14 Accounting process of DIAMETER protocol

3. SNMP

The Simple Network Management Protocol (SNMP) proposes a management model to facilitate the exchange of management information between network devices. It is usually used by administrators for the purpose of managing network performance, finding and solving network problems. There are three versions of SNMP:

- SNMP version 1 is defined by [RFC1155, RFC1157, RFC1212, RFC1215]
- SNMP version 2 is defined by [RFC2578, RFC2579, RFC2580, RFC3416, RFC3417, RFC3418]
- SNMP version 3 is defined by [RFC3411, RFC3412, RFC3413, RFC3414, RFC3415]

These standards define the SNMP protocol, the architecture of SNMP management frameworks, the Structure of Management Information (SMI), protocol operation, security model, etc., respectively. Each new SNMP version is enhanced and improved on the basis of the earlier version. The description of interoperability and compatibility among SNMPv1, SNMPv2 and SNMPv3 can be found in [RFC3584].

In SNMPv1 and SNMPv2, an SNMP system is regarded consisting of three components:

- Managed devices. A managed device is a network element in which the SNMP agent resides. A managed device collects management information and stores it in MIB, which can then be delivered to Network Management Stations (NMS) through SNMP protocol by agent.

Managed devices can be network devices such as routers, hubs, switches, access servers, hosts, printers, etc.

- Agent. An agent is a software component that resides in the managed device for the purpose of SNMP network management. It is responsible for communicating with NMS to fulfill SNMP management functions.
- Network Management Stations (NMS). NMS is responsible for monitoring and controlling managed devices.

Figure 2.15 below depicts the relationship between the above three components.

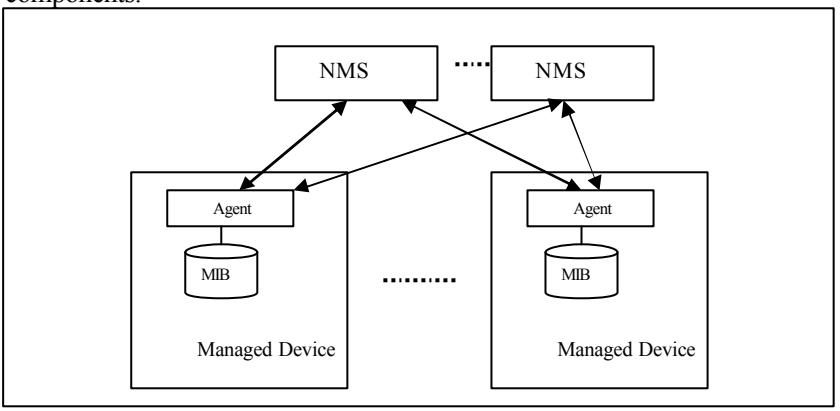


Figure 2.15 components in SNMP protocol

The managed information is stored in MIBs. The NMSs interact with agents residing in managed devices to retrieve the information using the "get" operation or to alter parameters using the "set" operation. In this operation scheme, the NMS plays an active role whereas the agent plays a passive role. In order to be consistent with the simplicity strategy of the SNMP protocol, a limited number of unsolicited messages ("trap") were proposed in SNMP. These traps are usually served for error reports. Different transport protocols can be served for exchanging SNMP messages. In the IPv4 environment, UDP is the preferred transport protocol for SNMP and must be implemented [RFC3417].

In SNMPv3, a new view of the SNMP entities is suggested. An SNMP management system can be regarded as any combination of the following types of application layer functionalities:

- Command generators: Initiate read- and/or write-class messages.
- Command responders: Respond to read- and/or write-class messages.
- Notification originators: Originate notification-class messages.
- Notification receivers: Receive notification-class messages.
- Proxy forwarders: Forward SNMP messages.

In this new view, an entity of an SNMP system can play several different roles through combining different functionalities.

- SNMP entities with command responder and/or notification originator applications function as traditional SNMP agents.
- SNMP entities with command generator and/or notification receiver applications function as traditional SNMP managers.
- SNMP entities with command generator and/or notification receiver, plus command responder and/or notification originator applications function as SNMP dual-role entities.
- SNMP entities with command generator and/or notification receiver and possibly other types of applications for managing a potentially very large number of managed nodes function as SNMP management stations.
- SNMP entities with proxy forwarder applications function as traditional SNMP proxy agents.

This new view on the SNMP entities can facilitate the design, development and deployment of more focused and more complex SNMP management systems.

SNMP utilizes Management Information Base (MIB) for storing information of the managed objects. An MIB is a collection of information that is organized hierarchically. Objects in the MIB are defined using Abstract Syntax Notation One (ASN.1) [ISO8824]. The type of object is defined by name, syntax and encoding. The name is used to identify managed objects uniquely with an Object Identifier. The syntax defines the data type such as integer, string, etc. of the managed object. The encoding describes the format of the management information.

[RFC1155] specifies the hierarchical tree structure in representing the Object Identifier. Each node except the root in the MIB tree is labeled with a number. Through traversing the tree from root to a node, a sequence of integers, which can then be used as the Object Identifier, can be constructed. Figure 2.16 illustrates how the flowDataSourcePeerAddress variable in traffic flow meter MIB [RFC2720] is identified by the MIB tree. The managed object flowDataSourcePeerAddress can be uniquely identified either by the object name:

iso.org.dod.internet.mgmt.flowMIB.flowData.flowDataTable.flowDataEntry.flowDataSourcePeerAddress or by the equivalent object descriptor: 1.3.6.1.2.40.2.1.1.9

SNMP protocol operates in a request-response mode. The NMS sends a request and the managed device feeds back corresponding responses. SNMPv1 defines four protocol operations: Get, GetNext, Set, Trap. In SNMPv2, two new operations are added: GetBulk, Inform. Through the GetBulk operation the efficiency of data retrieving ability is enhanced.

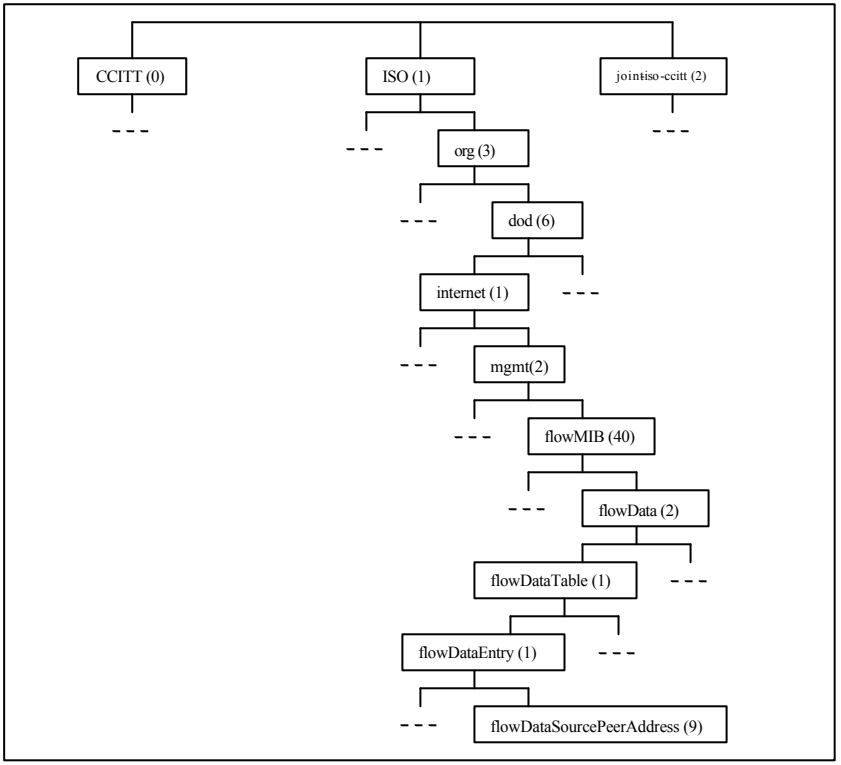


Figure 2.16 Tree structure in representing Object Identifier in MIB

SNMP is restricted to a low-frequency access scheme for MIB information. Whether it can satisfy all requirements for authentication and authorization is a controversial issue. Generally, it is not considered as an AAA Protocol [RFC3127, PBSP01]. However, it is commonly agreed that SNMP is suitable for accounting. Its support on accounting is mainly in its ability of transferring accounting records and storing them in an SNMP Management Information Base (MIB).

4. Accounting protocols comparison

This section provides a comparison among RADIUS, DIAMETER and SNMP protocols, especially on their accounting functionalities. The comparison is based on the RADIUS, DIAMETER and SNMP related protocol standard documents of IETF and [RFC3127, RFC2989, EkSP00].

- ***Operation model***

RADIUS operates in a client server model. The RADIUS server does not initiate any message, but only replies to the requests from RADIUS client. Therefore RADIUS can be regarded as operating in a pure client server

model. From the perspective of data collection, RADIUS works in an event based push mode.

DIAMETER operates in a similar way like the RADIUS except that not only the DIAMETER client but also the DIAMETER server can initiate a request. Therefore DIAMETER can be regarded as operating in a peer to peer model. DIAMETER can be also regarded as working in an event based push mode.

SNMP operates in the polling model. The NMSs initiate requests for retrieving data from management agents regularly. Through receiving responses from the management agents management information can be gathered.

- ***Transport protocol***

RADIUS utilizes UDP as transport protocol for information delivery. Since UDP provides no reliable transmission mechanism, the RADIUS implementations have to handle acknowledgement, retransmission, etc. reliable transport related issues.

DIAMETER, as it is developed to eliminate the inherent drawbacks of RADIUS protocol, emphasizes reliable transport through selecting SCTP (Stream Control Transmission Protocol) [RFC2960] or TCP for information transmission. SCTP is a transport level protocol like TCP and UDP, and it has been approved by IETF as a standard. It can detect errors such as discarded, duplicated or corrupted data, and can retransmit damaged data or reorder data. TCP is a well known and most frequently applied reliable transport protocol. It can provide reliable transport services through mechanisms such as acknowledgement, sequence number, flow and congestion control, etc.

Principally, *SNMP* can utilize any transport protocol conveying SNMP information. Yet it prefers UDP as the transport protocol in IPv4 environment because of its simplicity and efficiency [RFC3417]. SNMP over TCP was proposed as an option and was suggested to be applied for bulk data transmission [RFC3430].

- ***Proxy mechanism***

Proxy is an intermediate node located on the path between two communication endpoints. It can usually provide additional functionalities or services for specific purposes.

RADIUS supports the proxy mechanism mainly for the forwarding purpose. For example, a proxy server can adapt different types of links between client and server, or a proxy can be used for the purpose of retransmission so that the retransmission policy is robust and scalable. Since the behavior of the proxy is not defined in RADIUS explicitly, it may vary between implementations [RFC3588].

In DIAMETER, the term agent instead of proxy is used. DIAMETER classifies the agents into four categories in DIAMETER Base Protocol [RFC3588] according to the role an agent plays. Relay agents are responsible for forwarding requests and responses between clients and servers. They generate no messages and modify no messages. Proxy agents can not only forward requests and responses but also make policy decisions related to resource usage and provisioning. They may generate Reject messages in case of policies being violated. Redirect agents offer central DIAMETER routing configuration services for DIAMETER agents with relay function. Redirect agents do not provide relay service. The translation agent is responsible for translating protocols between DIAMETER and other AAA protocols such as RADIUS.

In SNMP, a proxy is defined mainly for converting protocols, enhancing the accessing flexibility of the managed devices without increasing their complexity and providing the secure communication ability over insecure links. Proxy mechanism can also facilitate the coexistence between different SNMP versions [RFC3584].

- ***Real time accounting***

Real time accounting requires that the accounting information processing to be completed within a time limitation. Since both RADIUS and DIAMETER support event based push model data collection, clients can send accounting data to the accounting server synchronously with the event notifications within a time window. SNMP agents can send Trap or Notification PDU to SNMP manager. Through that, real time accounting can be achieved.

- ***Accounting Record Extensibility***

RADIUS can support new attributes or vendor specific extensions. Due to the limited attribute number space of RADIUS, it is commonly considered as an accounting protocol with restricted extensibility.

DIAMETER was designed to improve the extensibility of RADIUS. Its attribute code is expanded to 32 bits comparing 8 bits with RADIUS. This can enhance the accounting record extensibility greatly. AVP numbers 1 to 255 are reserved for compatibility with RADIUS, and AVP numbers 256 and above are used for DIAMETER. However, these AVP numbers must be allocated by the Internet Assigned Numbers Authority (IANA) [IANA].

The MIB structure provides strong extensibility inherently. Therefore, vendor specific extensions can be easily integrated into SNMP.

- ***Batch Accounting***

Batch accounting is the ability of grouping accounting records for one time transmission rather than single record transmission.

RADIUS accounting supports only the pure event driven data collection model without the batch function.

Although initially batch accounting for small batches has been taken into account in the DIAMETER accounting extension draft [ACPZ99], the DIAMETER standard [RFC3588] did not adopt the batch accounting ability.

SNMP supports both polling and event driven data collection model. It can support methods for batch accounting. The GetBulk operation defined in SNMP offers a mechanism for retrieving a large amount of management information through exchanging a minimal number of request messages.

- ***Guaranteed Delivery***

Both RADIUS and DIAMETER employ application level acknowledgement and retransmission mechanism to guarantee data delivery. Since the retransmission behavior and fail-over mechanism are not exactly defined in RADIUS, the retransmission algorithm and fail-over processing of RADIUS is implementation dependent. This makes protocol reliability vary between implementations.

DIAMETER follows recommendations on the use of transport by AAA protocols in standard [RFC3539]. Fail-over algorithms and the associated state machine are well defined and supported by DAIMETER. Therefore DIAMETER can provide better support for guaranteed delivery.

SNMP can also provide guaranteed data delivery. Especially when the pull model is applied for data collection, the collectors know whether all data has been transferred.

- ***IPv6 support***

RADIUS was designed for running over IPv4. With the IPv6 slowly approaching, support for IPv6 has also been considered by IETF in [RFC3162]. Through introducing several new attributes, RADIUS can be integrated into the IPv6 environment.

DIAMETER has no problem in being adopted into the IPv6 environment, since the IPv6 issues have been taken into consideration during the protocol development. The AddressType defined in Address AVP can identify the content and format in the corresponding AVP.

- ***Security***

AAA nodes, due to their objectives for providing Authentication, Authorization and Accounting services, are always interested by attackers. Therefore security is one of the most important issues that must be taken into consideration seriously when designing AAA protocols.

Transport layer security is not explicitly defined in RADIUS documents. An exception is that IPSec [RFC1825, RFC2401, RFC4301] is implicitly expected to support RADIUS, due to the fact that IPSec is mandatory to be implemented for IPv6. The data confidentiality support by RADIUS is insufficient. RADIUS encrypts only user passwords in the exchanged packets between client and server and leaves other information exposed to po-

tential attackers. The proxies in RADIUS also play a role in affecting the data integrity and confidentiality. Since proxies may need to modify messages through them, this makes the entire encryption difficult and the RADIUS systems may be vulnerable against attacks from untrusty proxies.

DIAMETER adopts IPSec and TLS [RFC2246, RFC3546] for providing transport layer security. DIAMETER clients must support IPSec and may support TLS, whereas DIAMETER servers must support both. IPSec and TLS can provide only hop to hop security. The security is not sufficient if relays or proxies are involved, since hop to hop security cannot protect the entire user session. DIAMETER also provides end to end security, which can protect the entire communication path from the originating node to the terminating node.

SNMPv3 provides several security services to protect SNMP systems against potential attacks. [RFC3414] defines three different security function modules. The authentication module provides data integrity and data origin authentication to prevent SNMP systems from possible information modification or masquerade attacks. The timeliness module can provide protection against message delay or replay. The privacy module can prevent the message payload from being disclosed.

2.3.3 Raw Data Record (RDR) processing

After RDRs are gathered, they will then be further processed to generate Usage Records (UR) for different application purposes. Policies or rules will be applied in the course of RDRs processing to control the generation of the URs. The main processing functions include: validation, deduplicating, filtering, correlation, aggregation, and normalization.

2.3.3.1 Validation & Correction

After the RDRs are collected, some of them may be invalid or may contain errors. The Validation & Correlation module checks, if the RDRs are valid. If a RDR is verified to be valid, it can be handed down to other modules for further processing. If a RDR is not valid, then the validation module will try to correct the errors according to the predefined rules. If these errors can be corrected and the RDRs become valid, then they will be handed down to other processing modules. Otherwise, the RDRs with uncorrectable errors will be discarded, and this discarding information will be written down in log files.

2.3.3.2 Filtering

The filtering module discards the RDRs which are worthless for generating URs. The filtering mechanisms can be integrated in both Meter Layer

and Mediation Layer. In meters, filter policies control what should be measured and transferred to collectors. Through that, worthless data will not be generated and transferred, and consequently less traffic will be produced. Although some meters can provide tailored information to collectors according to filter policies, some meters without filter function, like log file based meters, may still generate worthless RDRs. After the RDRs are collected, the filtering module in the Mediation Layer can exclude useless RDRs according to filter policies. Through filter processing, the data volume that needs to be processed will be decreased.

2.3.3.3 De-duplication

The de-duplication module discards duplicated RDRs collected from different meters. Duplicated RDRs are usually generated due to overlapped meter policies. For example, router A is an edge router in network A and router B is an edge router in network B. These two routers are configured to perform meter operations. Both routers are configured to measure only traffic in and out their own networks. Therefore the traffic between router A and router B will be measured by both meter A and meter B, and duplicated RDRs will be generated. When RDRs from these two routers are gathered by the same mediation system, many duplicated RDRs are included. Discarding these duplicated RDRs will also reduce the data volume that needs to be processed to generate the URs.

This module can be implemented by comparing the key fields of RDRs to decide if these RDRs are duplicated or not. For example, attributes in RDRs like source IP address, destination IP address, source port number, destination port number, timestamp, etc. can be used to distinguish the redundant RDRs. If several RDRs from different meters have the same source and destination address, the same source and destination port number, and the same identification, only one of them needs to be kept, others can be discarded.

2.3.3.4 Correlation & enhancement

This module merges several RDRs, which are related to each other, to create a single record. Through that, the complete attributes of an event can be assembled. This can provide a single, complete view of information about an event. Figure 2.17 gives an example about how NetFlow records, RADIUS records, and LDAP records are correlated to generate a single attribute rich record.

In this example a NetFlow record needs to be mapped to the actual user who generates it. However, the record contains only source IP address information. A RADIUS record logs the information about this IP address

and to whom it was allocated during a period of time. Using the User Name found in the RADIUS record as an index, the corresponding registration information such as Real Name, Contact Address, etc. about this user can be discovered in a LDAP record. Through the above described correlation process, a more detailed single record can be generated. This process can also be called enhancement process. Enhancement is used when a RDR does not include enough information to generate a usage record, other RDRs will be referenced to enrich the information of this RDR.

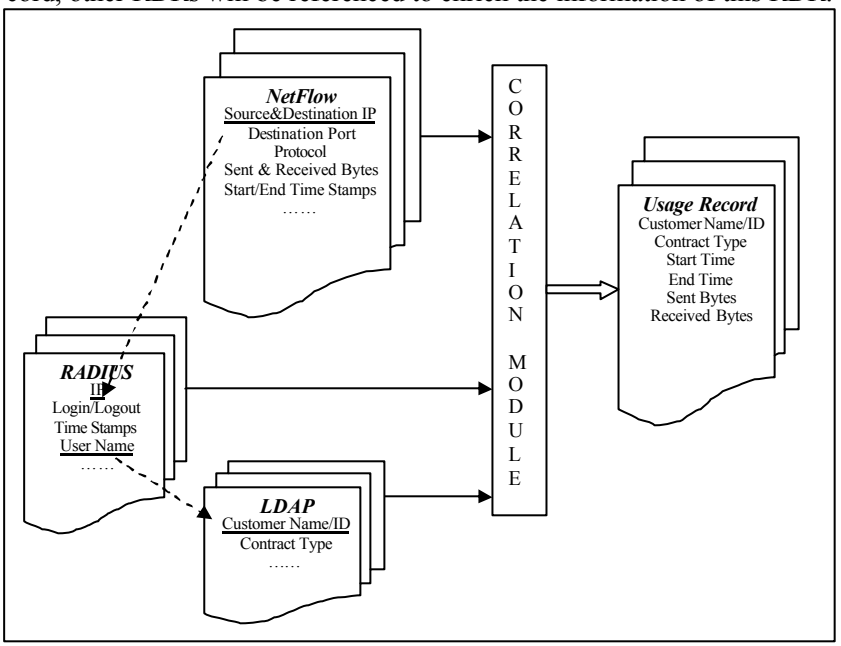


Figure 2.17 An example of the correlation process

2.3.3.5 Aggregation

An aggregation module accumulates a set of RDRs to produce a UR through statistical calculation according to corresponding rules. The UR can then be used by the OSS Layer applications. In the aggregation module, different rules will be used to control the aggregation of the RDRs. The rules can define which fields are the key for the aggregation process, which operations should be performed on the aggregated fields, and under which circumstances the aggregation should be timeout. For example, during a VoIP call interim RDRs are generated before the end of the call is reached. These interim RDRs of this VoIP call should be merged with the Start RDR and the End RDR to generate a single record to represent this

VoIP call event. Although both correlation and aggregation modules utilize several RDRs to generate a single record, they are applied for different purposes. The correlation process assembles RDRs in a "horizontal" direction to generate a complete and more accurate record, whereas the aggregation process consolidates RDRs in a "vertical" direction to generate a summary record.

2.3.3.6 Normalization

After the above described processes, Usage Records are generated. These URs can first be stored in an intermediate format, in a standard format, or even in the raw format as they are collected. The normalization module will then transform the URs to different formats according to the rules in order to meet the requirements of different applications in OSS Layer. Since different applications in OSS Layer may need different formats of URs to meet different requirements, the normalization module should support generating specified formats such as IPDR, XML, Text, etc. It should also have the extensible capability of supporting formats which might be developed in the future.

The normalization process can be executed in static or dynamic ways. The static way is to generate URs in different formats for different application purposes, and then store them for future collections by OSS Layer applications. The dynamic way is to generate and store URs in a standard or intermediate format. If an application in OSS Layer requires the URs to be collected with some format, the normalization module converts the URs to the required format at first and then the URs are sent to the OSS Layer. If applications in the OSS Layer have the ability of adapting to different formats or supporting the standard format, no normalizing process will be needed. This can certainly reduce the overhead caused by the normalization process and speed up the whole accounting and billing process. Hence, standard format is very important and necessary.

2.3.4 Usage Record format

In order to interface with the OSS Layer applications easily, the Mediation Layer should have the ability of generating all formats which may be needed by different applications in the OSS Layer.

Till now, there is no standard format recording Internet usages. Usually, even different accounting system providers define and use their proprietary UR formats. Different UR formats will certainly cause interoperation problems. Therefore it is necessary to develop a standard format to make it easy to exchange URs between Mediation Layer and OSS Layer. It is also necessary for future IP Billing applications.

IPDR (Internet Protocol Data Record) [IPDR02] is a record format, which is intended to be used as the standard. The IPDR format is defined by the IPDR working group. The IPDR organization is a non-profit open consortium consisting of equipment vendors and service providers. It aims to facilitate the exchange of IP usage data and control data information among different systems such as network elements, operation support systems, business support systems, etc.

The IPDR standardizes a usage record format and delivery protocol. It represents the usage in encapsulation techniques such as XML. It defines an open, extensible, flexible record that encapsulates the essential parameters for different IP service transactions.

2.3.4.1 IPDR record structure

The IPDR record structure is designed to have the ability of characterizing any type of usage. There are five components common to all IPDR records. These components are the 'who, what, where, when, and why' values that describe a particular usage event [IPDR02].

- Who
  User ID (in some form, if available)
- When
  Start and Stop Time or Event Time of service usage
- What
  Service type
  Usage measures / quantities (e.g. bytes, packets, flows, hits, transactions, time duration)
  QoS measures
  State information
  Event code (logon, logoff, threshold exceeded)
  Other information about state transition or current state
- Where
  Traceability / Context
  Source Identifier
  Destination Identifier
  Service Element Identifier (originator)
- Why
  Event trigger type – (i.e., why is the network and service element reporting this data)

In addition to the '5Ws' defined, each record may include reference pointers to other IPDR records which either capture related usage information, or contain usage information that was used to create the given record.

In addition to the IPDR record structure, the IPDR specification defines a set of interfaces for exchanging IPDRs between IPDR-enabled devices or systems.

2.3.4.2 Services considered in IPDR

The IPDR considers that many services are structured in a hierarchy. For each service the IPDR specification describes the definition of the service, service requirements, attribute list for service usage and, for some services, the basic flow. The IPDR also defines the formal specification of the records for each service considered.

IPDR is an open standard; it can be extended to support new services in the future. Now the IPDR contains the following services [IPDR03]:

- Application Services Provision
- Voice over IP (VoIP)
- E-mail Services
- Access and Authorization Services(AA)
- Internet Access
- Content Service
- Push Delivery
- Wholesale Requirements
- Streaming Media
- Video on Demand (VoD)

2.3.5 Adaptor for distribution of Usage Records

The adaptor for distribution of URs interfaces with all applications in the OSS Layer. It communicates with the applications of the OSS Layer, and uses push or poll approaches to send the URs to the downstream applications.

2.3.5.1 Transfer mechanism

To interface with different OSS Layer applications, the adaptor for distribution of URs should support different kinds of transfer mechanisms such as TCP, UDP, FTP, SMTP, Network File Sharing, CORBA, COM, SQL, etc. This allows quick and easy integration of the mediation system with OSS layer applications.

2.3.5.2 Reliability

In order to achieve reliability, the re-transmission mechanism should be used to transfer the URs again, once the URs are not delivered correctly to the OSS Layer applications due to network failure or other reasons.

2.3.5.3 Security

The URs contain private information. These records are concerned with money in the billing system, so keeping security during the transmission of URs is very important. It is more necessary to add the security mechanism when the URs are being transmitted to remote applications.

2.3.6 Criteria for evaluation of the Mediation Layer

The Mediation Layer acts as a bridge, enabling the RDRs to be collected from different kinds of network elements, and transforming it into high value information, and then distributing the processed information to various downstream applications.

Some criteria for the evaluation of the functionality of products used in the Mediation Layer are given below:

- Can collect RDRs from multiple types of network elements
- Support push and poll approaches for RDR collection
- Support multiple transport protocols for RDR collection
- Support multiple data formats in input streams
- Support rule based RDR collection
- Support rule based RDR processing, such as filtering, validation, correction, de-duplication, correlation, enhancement, normalizing, etc.
- Can generate different formats of URs
- Can manage the storage of URs
- Support the distribution of URs to multiple applications
- Support multiple transport protocols for the distribution of URs
- Provide the management mechanism for the mediation system

Some criteria for the evaluation of the performance of products used in the Mediation Layer are given below:

- Scalability. The Mediation Layer can be dynamically extended to support adding new network elements and new OSS Layer applications easily without affecting the system's operations. The Mediation Layer should be able to accommodate to the changes in the Meter Layer or the OSS Layer.
- Flexibility. The mediation system can easily support different network infrastructure and application logic with the configuration of the rule sets.
- Reliability. The mediation system should be designed to be fault tolerant. During the collection of RDRs and the distribution of usage records, the data retransmission mechanism should be used if faults occur. The validation and correction mechanism should also be used to guarantee the availability of the RDRs. Non-volatile storage should be

used to facilitate disaster recovery and minimize the threat of losing valuable accounting data.

- Security. Since data throughout the mediation system includes private information and concern money in IP Billing cases, the importance of keying the security of data is self-evident. During the transport and the processing of the data, security mechanisms should be considered.
- Real-Time capability. Different applications require different real time capabilities. In some business cases real time means better service and more customers, and that also means more economic profit. In the Mediation Layer, the real time capability can be reached and enhanced in three stages: the RDRs collection, processing and usage records distribution.

2.4 Management of accounting systems

The accounting management system concerns Meter Layer and Mediation Layer. Its main functions include: configuration, policy or rule distribution, rule explanation and execution, management interface offering, status monitoring, operations control.

2.4.1 Configuration

Configuration is the process of adjusting an accounting system to regulate its running behaviors according to the predefined rules.

In Meter Layer, issues like what should be measured, which measurement granularity should be achieved, how often the RDRs should be reported, which kind of accounting protocol should be supported, which kind of formats should be used for RDRs, how much storage space should be allocated for RDRs, etc. need to be configured before a meter measurement start.

Due to the diversity of meter devices, their configuration may be implicit or explicit. Some network elements may be not designed and employed originally for meter purposes, despite the fact that they can generate meter related information. In this case, these network elements are configured according to their original purposes instead of according to meter rules, since configuration for meter purposes by accounting management system may be impossible. Therefore the configuration for these network elements is implicit, and the Mediation Layer should be configured correspondingly to accommodate these kinds of meters. Network elements that integrate meter functions can be configured according to meter rules. This configuration is explicit. Their behaviors can be adjusted and controlled by the accounting management system.

The Mediation Layer is the bridge between Meter Layer and OSS Layer. On the one hand, there are many different types of network elements which can be used as meters. With the appearance of new network elements, in the future, these new network elements may also be used as meters. Therefore the Mediation Layer should be flexible and extensible to accept these dynamic changes of network elements. On the other hand, the Mediation Layer has to interface with different applications in the OSS Layer. The Mediation Layer should also be flexible and extensible to meet the requirements of the changes of the OSS Layer applications. Furthermore, during the processing of the RDRs, the behaviors of different processing modules should also be controlled.

All these requirements can be met through the dynamic configuration ability with the help of rules. The accounting management system can dynamically configure the mediation system by modifying the rule sets, actively adding or deleting its relationship with network elements or downstream applications.

2.4.2 Policy and rule management

Policies and rules play a very important role in regulating the behaviors of accounting systems. Policy and rule management concerns policy explanation, rule distribution, and rule execution.

Policy explanation is responsible for converting policies into executable rules. On the one hand, different policies may be derived from different application purposes. They must be translated into understandable accounting system rules. On the other hand, since there is no standard rule suitable for all meter devices, the same policies may be required to be converted into different executable rules for different meter devices.

After rules are generated from policies, they should be distributed to different components in the accounting system to control their behaviors. The rules setting can be static or dynamic. The static rules cannot be changed during a period of time. If an accounting component hard codes its rules, the rules cannot be altered except by modifying the source code of this component. Another situation for static rule setting is to mount rules when an accounting component is started. This setting will not be changed until the accounting component is stopped. The static setting has the advantage of simplifying the rule distribution process. The disadvantage of it is that the accounting components cannot meet the requirement of changing the environment dynamically. The dynamic rule setting can change the accounting related rules on the fly during the accounting components running. The rules can be downloaded to the accounting components actively or passively from the accounting management system. Through that, new

requirements to the accounting components can be met effectively, and flexibility and extensibility can be easily achieved.

Rules require to be enacted by execution engines. Below an abstract rule set example is given:

Rule Set ::= Rule | {Rule ;}

Rule ::= Test ; Action

Test ::= value=attribute & mask

Action ::= operation_code | {parameter ;}

The "Test" part calculates the condition value; the " Action" part operates according to the condition value. Different types of rule sets may need different rule execution engines. Usually these execution engines are integrated into the accounting components as a part of the accounting functions.

In the Meter Layer, issues like what should be measured, which measurement granularity should be achieved, how often the RDRs should be reported, which kind of accounting protocol should be supported, which kind of format should be used for RDRs, how many storage space should be allocated for RDRs, etc. can be controlled by meter rules.

In the Mediation Layer, during the process of collecting RDRs, processing RDRs and distributing URs, the rule sets control all these activities.

During the collection of RDRs, the rule sets decide from which network elements RDRs should be collected, which kind of RDRs should be collected, which RDRs should be filtered, the poll intervals of the collection of RDRs, and which transfer protocols should be used, etc.

During RDRs processing, rule sets can control the validation, correction, filtering, de-duplication, correlation, enhancement, aggregation and normalization of the RDRs. Rule sets also control the storage of URs.

Rule sets also play an important role in the adaptor for distributing URs. They specify which applications the URs should be sent to, which transfer protocols should be used, which format of URs should be chosen to meet the requirements of the downstream applications, etc.

2.4.3 Management interface

The accounting management system should provide an administration interface to the accounting system administrator. Through that, an accounting system administrator can manage and control the accounting system. The management interfaces may concern operation interface, monitor interface, report interface, policy and rule set interface, user management interface, etc.

2.4.4 Status monitoring

The accounting management system should monitor the status of the accounting system and generate the monitoring report. The statistic results can be reported to the accounting system administrator through the management interface. By monitoring the accounting system, the accounting system administrator can verify the health status of the entire system and take corresponding management actions.

2.4.5 Operations control

The accounting management system can provide the control interface for the administrator to control the accounting system operations. The accounting system administrator can start up, shut down, suspend or change the behaviors of operations in the accounting system through operation control mechanisms.

2.5 IP Billing and OSS/BSS Layer

The OSS/BSS Layer is the highest layer of the Internet accounting system architecture. This layer consists of different types of applications such as Billing, decision support, fraud detection, trend analysis, etc. These applications use different communication mechanisms to gather the Usage Records from the mediation systems. These applications concern different application areas. Below only the technology of Billing will be discussed.

Billing is the process of consolidating charge records on a per customer basis and delivering a certain aggregate of these records to a customer. Billing process consists of collecting URs from mediation devices, calculating the charge according to the price schemes, reporting their expenditure to the customers or delivering the invoice. Figure 2.18 illustrates this process.

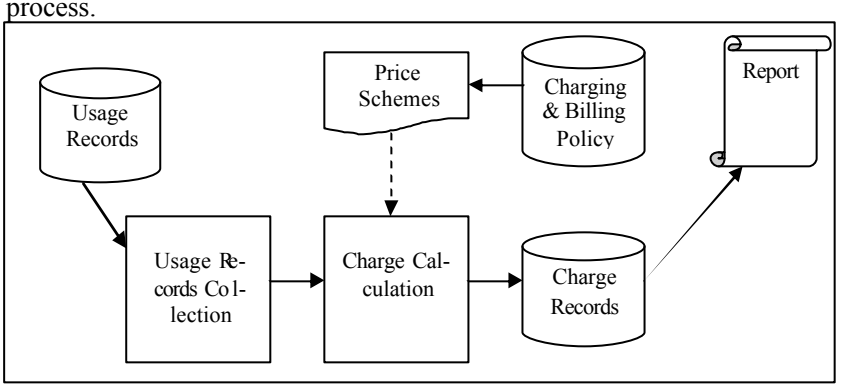


Figure 2.18 IP Billing process

2.5.1 Billing modules

The Billing system consists of several modules.

2.5.1.1 Usage Records collection module

The Usage Records collection module interfaces with different mediation systems to gather the URs generated by the mediation systems. The push or pull approaches can be used in the collection of URs. If the mediation systems are designed flexible enough to transfer the URs of different formats with different kinds of transport protocols, the Billing systems need not pay more attention to interface with these mediation systems. Otherwise, the adaptation mechanisms must be applied in the Billing systems. During the collection of URs, rule sets can also be used to meet the dynamic change requirements, and this can make the system more flexible. After the URs are collected, they will be sent to the Charging module.

2.5.1.2 Charging module

The charging module accepts the URs collected by the URs collection module, and calculates customers' actual charge according to the price schemes. The results of the calculation will be generated and stored as charge records. The charge records will be organized to be presented to the customers eventually. After the charge calculation the generated charge records are stored in a database, they can then be used to generate the detailed reports or invoices to customers.

2.5.1.4 Price Scheme module

The price schemes define how the URs should be calculated, and they also define the price per unit of usage. The price schemes can be configured dynamically to meet the requirements of a frequently changed charging and billing policy. This module should provide the interface for price rule sets definition and modification.

In the course of charge calculation, the price schemes play a very important role, since all calculating rules are defined in them. Through the adjustment of the price schemes, the billing system can provide flexible billing capabilities to meet the requirements of complex billing plans.

2.5.1.3 Report module

The report module uses the charge records to generate bills for every customer. A bill may be a printed invoice, or additionally the customers can inquire and browse the billing information with a web browser. Therefore, the report module provides these functions:

- Organize and generate bills. The detailed usage information should be provided to the customers when needed. Customers can also design their own style of reports.
- Inquire bill
- Display bill
- Print bill

2.5.2 Criteria for evaluation of billing systems

Here the criteria for evaluating IP Billing systems are listed:

- Can gather IP usage records from different mediation systems
- Can calculate the charge of the customers' usage according to the price schemes
- Can generate the charge records and store them in a database
- Can organize the reports and generate the bills for customers
- Can provide multiple forms of bills, such as web forms, printed invoices, etc., to customers
- Can support flexible price schemes
- Can provide system management interfaces
- Can provide realtime billing capability

2.6 IPv6 and Internet accounting

IPv6 [RFC2460], also known as IP the Next Generation (IPng), was developed to enlarge the available IP address space to meet the increasing requirements on the Internet. In addition to providing greater addressing space than IPv4, IPv6 can also provide built-in QoS, better routing performance and services. As a new Internet protocol, IPv6 will certainly influence systems on the basis of IPv4 architecture. IP traffic accounting systems should also accommodate the new requirements of IPv6.

This section takes a brief look at the IPv6 protocol at first, and then discusses possible challenges to Internet accounting accompanied by the emergence of IPv6, and at the end draws some conclusions.

2.6.1 IPv6 in a nutshell

IPv4 was published by IETF in 1981 [RFC791], and it has become the most popular network layer protocol. During the rapid development of the Internet more challenges have arisen due to the inherent deficiencies of IPv4. A prominent insufficiency of IPv4 is its limited IP address space, which may slow down the further growth of the Internet. In order to break the IPv4 address space limitation, NAT has been developed to reuse IPv4 private address space through mapping private IP address and transport

layer port pair to public IP address and transport layer port pair. Although this solution can support more end-systems joining the Internet, it is still not perfect. Possible problems include: servers behind NAT may be unreachable if NAT and servers are not properly configured, the NAT servers become traffic bottlenecks because of burdensome IP address mapping for every IP packet, some types of traffic, e.g. IPSec packets, cannot pass through NAT due to the encrypted IP addresses. Therefore the NAT mechanism is regarded as a solution only to extend the life of IPv4 address space rather than a solution to solve the IPv4 address space problem [Davi03].

Besides the IPv4 address space problem, demands for simplifying router table and IP configuration, enhancing IP layer security and QoS support, etc. emerge progressively. All these improvements cannot be realized in the old IPv4 framework, hence IPv6, a new IP protocol, was developed to undertake all these tasks. Figure 2.19 below depicts the IPv6 header format.

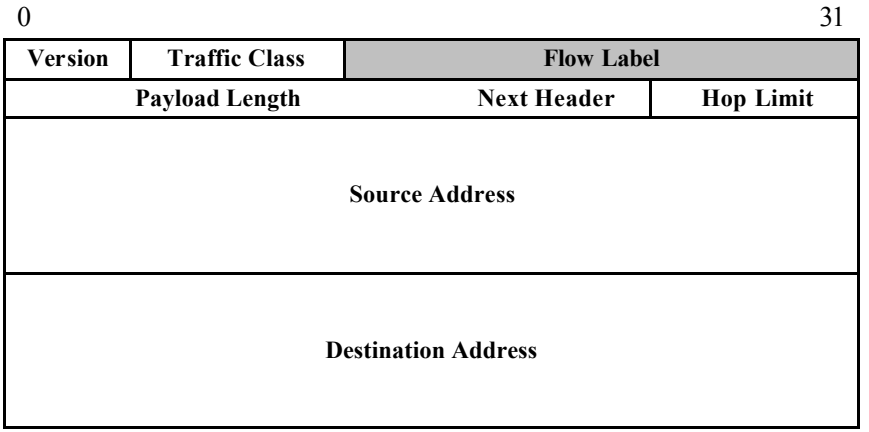


Figure 2.19 IPv6 header format

Below we summarize IPv6 features by comparing it with IPv4.

1. Header format simplification

Compared with IPv4, one of the most important improvements of IPv6 is its simple header format. The IPv6 header consists of eight fields and has a fixed size of 40 bytes versus 13 fields and 20 – 60 bytes in IPv4. Extensions and options in IPv6 are not included in the IPv6 header, and they belong to the payload of IPv6 packets.

The number of fields is 12 (without Options field) in IPv4 whereas it is 8 in IPv6. In addition, the number of fields that must be processed by an

intermediate router is decreased from 6 in IPv4 to 4 in IPv6. Therefore routers can forward packets more efficiently.

Fields such as Header Length, Identification, Flag, Fragment Offset, and Header Checksum in IPv4 header disappear from IPv6 header. Only Flow Label is introduced into IPv6 header as a new field. Since IPv6 header has a fixed size, the Header Length field is not necessary. In IPv6 payload fragmentation is suggested to be processed only by communication pairs instead of intermediate routers. The fragmentation related fields such as Identification, Flag, and Fragment Offset in IPv4 header are not included in the IPv6 header. A Fragment extension header can be used for this purpose. The Header Checksum field is also not contained in IPv6 header, since other layers instead of IP layer will be responsible for that. Through removing these unnecessary fields the efficiency of packets transmission can be improved.

2. Larger Address Space

Address space limitation in IPv4 was the main impetus for the emergence of IPv6. In IPv4, 32 bits are allocated for representing an IP address, which can support about 4 billion hosts. With the explosion of the Internet, the Internet population grows rapidly. It was estimated that there were about 1.08 billion Internet users till 2005, which is less than 20% of the world population [CIA2]. With the development of pervasive computing, more and more devices, e.g. mobile phones, palms, PDAs, even refrigerators, etc., may need IP addresses to join in the Internet. It seems that the IP addresses will be exhausted very soon. Although efforts such as PPP/DHCP IP address sharing, Classless Inter-Domain Route (CIDR), NAT, etc. have been made to enhance the IP address reusability, they cannot solve the IP address space limitation problem of IPv4.

IPv6 uses 128 bits to represent an IP address, which can theoretically support 3.4×10^{38} possible hosts. This is more than enough from the point of view of current knowledge and technology. With the enough IP address space in IPv6, the deployment of network elements like NAT for IP address reuse is no longer necessary.

3. Automatic address configuration

IPv6 brings not only larger address space but also an automatic address configuration mechanism [RFC3315]. With the IPv6 automatic address configuration mechanism, a host can obtain its IP address even without DHCP support [RFC2462].

4. Improved extensibility

Options in IPv4 header require all intermediate routers to check the existence of options and process them when they exist. This results in performance degradation of routers. IPv6 introduces a new extension header mechanism to improve both extensibility and efficiency. The extension

headers are not included in the IPv6 header but placed between the IPv6 header and upper layer header. In IPv6 only the Hop-by-Hop extension header needs to be handled by all intermediate routers. Other extension headers are processed by sender specified routers or destination host. This can improve routers' packet forwarding performance.

According to [RFC2460], the following six extension headers must be supported by IPv6 enabled nodes: Hop-by-Hop Options Header, Routing Header, Fragment Header, Destination Operations Header, Authentication Header, Encapsulating Security Payload (ESP) Header [RFC2402, RFC2460].

An IPv6 packet may contain zero, one or more extension headers. If more than one extension header is present in a packet, then these headers can construct a chain with the help of Next Header pointer in every extension header to indicate the immediate followed extension header. [RFC2460] recommends the following location order of extension headers in the case of more than one extension headers in a packet:

1) Hop-by-Hop Options header
2) Destination Options header (for the first destination defined in Destiantion address field and subsequent destinations specified in Routing header)
3) Routing header
4) Fragment header
5) Authentication header
6) Encapsulating Security Payload header
7) Destination Options header (for the final destination)

5. Built-in security

In order to amend the insufficiency of IPv4 in security, a security framework for the IP protocol layer, i.e. IPSec, is proposed by the IETF IP Security Protocol Working Group in [RFC2401]. IPSec provides security services in the IP layer, which is transparent to the upper layers and can improve the communication security without any modification in upper layers. The IPSec provides a flexible framework by supporting different services, algorithms and granularities. IPSec is supported by both IPv4 and IPv6. In IPv4, IPSec supporting is optional whereas in IPv6 it is mandatory.

Security is considered to be an important concern when IPv6 was developed. Many existing security threats or possible attacks happened in IPv4 are taken into consideration in IPv6's design. On the other hand, the new characteristics of IPv6 such as its expanded address space bring also additional functions against attacks. According to the analysis of S. Convery and D. Miller [CoMi04], some attacks in IPv4 become more difficult or even impossible in IPv6.

2.6.2 Challenges caused by IPv6 to Internet accounting

With the emergence of IPv6, new challenges are also brought to Internet accounting. The expanded IP address space requires corresponding modification in IP address related accounting attributes. Accounting protocols also require modifications to adapt to IPv6. Meter location consideration may also be different from that in IPv4. This section makes a survey on how IPv6 affects the traditional IPv4 based Internet accounting.

2.6.2.1 Meter Location

The 128 bits address space of IPv6 does not affect the choice of meter location. The possible IPv6 effect on choosing meter location may be the Routing extension header defined in IPv6. With the help of the Routing extension header, the address of a meter could be inserted in the Routing extension header. Through that, all the required IP packets can be sent to destination via the designated meter. In this situation, meters can be located in any place. Although a similar mechanism has been defined in IPv4 as Loose Source Routing option, two disadvantages prevent this option from being widely applied. One is the limited option size of IPv4 that restricts the routing list size, and the other is the poor IPv4 options implementation in routers.

2.6.2.2 Measurement technology: active and passive

IPv6 introduces little impact on active measurement. The simulation activities or injected packets should be compliant to the requirements for IPv6. ICMPv6 instead of ICMPv4, for instance, should be used for measuring responses in the IPv6 environment. Other measurement tools such as "ping", "traceroute", etc. should be replaced with the corresponding IPv6 version. Some existing active measurement tools may need to be modified to meet the requirements for IPv6.

For passive measurement methods, not only the expanded IPv6 address space but also the new IPv6 header should be taken into consideration.

The packet capturing method in IPv6 could generate an even larger volume of measurement data than in IPv4. Considering the calculation example in section 2.2.3.2, with the source and destination IP addresses expanded from 4 bytes in IPv4 to 16 bytes in IPv6, the record size grows from 200000 bytes/s to 500000 bytes/s, which is 2.5 times than that in IPv4. Consequently this means that more memory and storage space, and more powerful CPUs are required for IPv6 packet capturing. Furthermore, expanded IP address space will inevitably stimulate Internet traffic volume increasing as a consequence of more nodes joining into the Internet. This challenge must be confronted by the packet capturing method.

IPv6's impact on flow based measurement method lies in flow identification and its generation. The Flow Label field in the IPv6 header is a new field compared with the IPv4 header. It is used to identify flows. Rather than generating flow identification in routers or in flow based meters in IPv4, hosts sending flows are required to label flows by themselves in IPv6. Besides the advantage of informing routers to handle flow packets specially, the flow label in the IPv6 header simplifies the flow based measurement. In IPv4, an IP flow is usually identified by routers or flow meters through extracting source and destination IP addresses from IP header, source and destination port from transport layer header [RFC2063]. Analyzing both IP layer and transport layer protocol is inefficient and results in performance degradation on routers with meter functions. Another disadvantage of identifying flows by routers or meters is that the generated flows cannot reflect the application purposes of the flows, since flows are classified only according to IP addresses and transport layer ports and only for the accounting purpose. Therefore, flows in IPv4 accounting are usually considered as a sequence of meaningless packets between two communication entities. The third disadvantage of IPv4 flow based measurement is that different meters located between two communication entities may generate different flows for the same sequence of packets due to different flow classification rules. Sometimes, this inconsistency may make it difficult to correlate records from these different meters. In IPv6, the flow labels are identified by sending hosts according to their application purposes. A flow can simply be identified by a flow label. Therefore, flow based meters need only extract information such as Flow Label, source address, destination address from IPv6 headers to fulfill flow based measurement. It is not necessary to analyze the upper layer of IP packets. The Flow Label improves the efficiency and simplicity of the flow based measurement. Since the IP packets with Flow Label mean that they require special processing, this can facilitate usage based charging and billing. The inconsistency problem in IPv4 flow based measurement can also be solved.

The Flow Label can also enable the flow based sampling strategy that is suitable for measuring multimedia service such as video and audio transmission.

2.6.2.3 Measurement metrics

As a result of the new IPv6 header, the measurement metrics of meters are required to be adjusted accordingly.

First, meters should have the ability to distinguish IPv4 traffic from IPv6 traffic through simply checking the version field of the IP header.

The Flow Label in IPv6 is a new attribute that can be extracted directly from the IPv6 header. This parameter should be monitored and reported for the purpose of flow based traffic accounting.

Analyzing transport layer protocols such as TCP or UDP may become a little more complicated in IPv6 than in IPv4. The IPv4 header contains a protocol field used to identify the type of transport layer protocol header directly after the IPv4 header. Moreover, the location of the transport layer header can be obtained from checking the Header Length field in the IPv4 header. In IPv6, if extension headers exist, the TCP or UDP header is located after the extension headers chain. Therefore the only way to obtain transport layer protocol related information is to traverse the whole extension headers chain. This will certainly increase the overhead of analyzing transport layer information.

IPv6 128 bits address requires corresponding adjustment to be made in Raw Data Records (RDR) to accommodate the IPv6 address attribute. In order to be compatible with IPv4 traffic measurement, IPv4 address attribute can be considered in RDRs in two ways. One way is that the IPv4 address attribute can be explicitly defined in RDRs coexisting with IPv6 address attribute. Another way is that IPv4 address can also be represented by IPv4 compatible IPv6 address or IPv4 mapped IPv6 address. This requires only IPv6 address attribute and IPv4 address attribute is implicitly defined by IPv6 address attribute.

2.6.2.4 Mediation Layer

The Mediation Layer is affected by IPv6 mainly due to the IPv6 address. Accounting protocols have to be improved to accommodate the IPv6 address structure. RADIUS, DIAMETER and SNMP all extended the IPv6 related attributes to support IPv6 address [RFC3162, RFC3588, RFC3419]. The legacy mediation system should also be improved to have the ability of processing, generatin g the IPv6 related records.

2.6.3 Summary

IPv6 can provide not only a huge address space, but also efficient routing, better QoS support and enhanced security mechanisms. With the introduction of IPv6, traditional IPv4 based accounting mechanisms should be adjusted to meet the requirements for accommodating IPv6; and on the other hand the Internet accounting systems should also be improved to be able to utilize the new characteristics of IPv6 for the purpose of providing better accounting services.

The adjustments of IP accounting systems are mainly caused by the increased size of IPv6 address. Meter records and usage records should take

IPv6 address into consideration. When an accounting administrative domain contains both IPv4 and IPv6 networks, attention should be taken in processing these two different IP address formats. The new characteristics such as Flow Label, extension headers such as Routing header, Destination header, etc. can help to simplify IP accounting, and to provide more accurate accounting information. Because IPv6 is still not widely applied, it is expected that accounting mechanisms could be integrated into IPv6 to facilitate the implementation of simple, accurate and finer granular IP accounting systems.

Chapter 3 User based IP traffic accounting

Nowadays IP traffic accounting systems are based on the assumption that one IP address is uniquely associated with one user during a period of time. However, this assumption cannot be applied to multi-user systems, where an IP address can be shared by multiple users at the same time. In this case, an IP address cannot be uniquely mapped to a single user. Therefore, now existing IP traffic accounting systems can only provide coarser granular IP address level accounting information. In order to provide finer granular user level traffic accounting information, we propose a user based traffic accounting concept which can facilitate collecting and analyzing network resource usage information on the basis of users. With user based IP traffic accounting, IP traffic can not only be distinguished by IP addresses but also be distinguished by users. The user based IP traffic accounting mechanism can be regarded as an extension to the now existing IP address based traffic accounting mechanism. With user based IP traffic accounting not only an IP address attribute but also a user attribute is used as an aggregating index in collecting the network resource usage information.

In this chapter the principle of user based IP traffic accounting is introduced. At first the motivation for user based IP traffic accounting is explained through analyzing the flaws of the traditional IP address based IP traffic accounting architecture. Then the user based IP traffic accounting concept is introduced and the user model for IP traffic accounting in multi-user systems is proposed. After that, an overview about the user based IP traffic accounting technique is depicted. Issues such as the Accounting Agent and its location, IP traffic identification methods and user information transmission are briefly introduced. The related works concerning user based IP traffic accounting are also surveyed and analyzed. Finally the user based network access control mechanism with the help of user based IP traffic accounting technique is illustrated.

3.1 Motivation for user based IP traffic accounting

The Internet is now becoming a part of people's daily life. With the rapid development of the Internet, more and more services are provided by the Internet, more and more users are attracted to enjoy these Internet services, and consequently more and more traffic is produced. For example, according to the statistics from DFN, from 1997 till 1999 the IP traffic volume of the University of Kaiserslautern doubled every year [Muel00]. Figure 3.1 shows the traffic volume and corresponding cost forecasted after 1999. A similar statistic result was also found in [Odly03].

The increased traffic volume not only causes load on the network, which may decline the network performance due to congestion, but also means more money should be paid for this traffic. Users will not use the network resource responsibly if the usage is free of charge. In order to facilitate reasonable Internet usage and avoid unnecessary traffic generation, IP traffic accounting is a solution which can provide accounting information reflecting the behavior of users' network resource consumption. It is believed that traffic volume based charging will become dominant in Europe by 2008 [Anal03]. The traffic accounting information can be used not only for network performance trend analysis but also for charging and billing. Charging users for their network resource consumption can stimulate their reasonable network resource usage on the one hand, and allocate cost to users according to their network usage on the other hand. This was the motivation for the proposed user based IP traffic accounting project "NIPON" [Muel00, NIPO00].

Today billing is commonly used by Internet service providers (ISP). However, billing systems might be used even within LAN. This makes it possible to allocate the costs which are related to the network traffic of a single-user or an institution using a campus network. Nevertheless, billing has also an influence on the behavior of the users. Because of the rising costs, it is also reasonable to present bills to the end users. This aspect becomes very important when a network offers different Classes of Service (CoS).

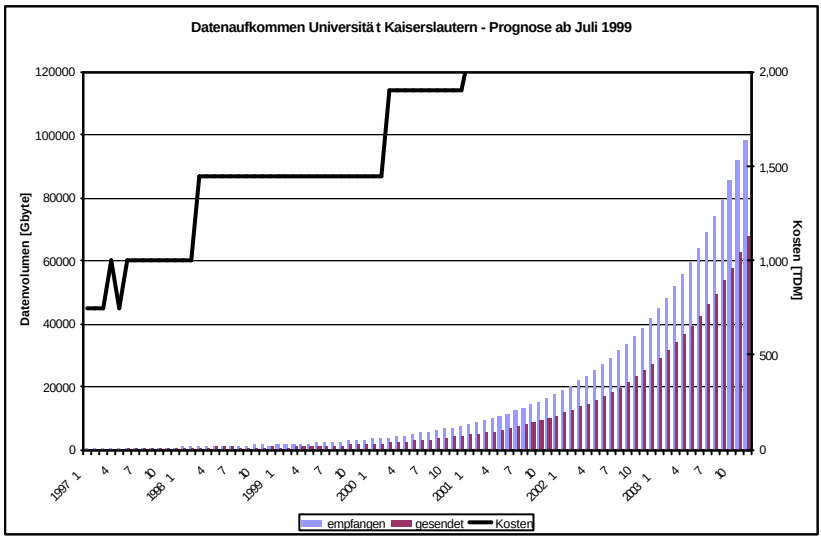


Figure 3.1 Statistic and prognosis of IP traffic volume and cost in university of Kaiserslautern made in 1999 [Muel00]

It is not relevant whether billing is used in a LAN to allocate costs or to motivate reasonable network usage. In both cases, it makes sense to be able to correlate the network traffic with the corresponding users, which are responsible for it. Despite the fact that there exist many IP accounting solutions today, almost all of these solutions correlate IP addresses, instead of users, with traffic. This is based on the simple assumption that an IP address is equal to one user. However there are several scenarios where an IP address is not associated with one user. For example, the multi-user computers which are commonly used in organizations like universities, institutes, etc. An IP address is shared by several users at the same time in these systems. The IP address based IP traffic accounting solution, which is referred to as the traditional IP traffic accounting mechanism in this dissertation, is not able to distinguish between different users of the IP traffic in those systems.

Therefore a more accurate IP traffic accounting mechanism is required to compensate the insufficiency of the traditional IP address based traffic accounting. The user based IP traffic accounting is the evolutional solution that meets the finer granularity requirement in multi-user systems.

A secure Internet requires also that users' network usage can be accountable. Instead of hosts, users' network usage should be accountable, traceable and controllable. Separating location information, host information and user information from each other is regarded to be a very important issue in designing future secure Internet by the GENI project [BCPS05].

3.2 The flaw of traditional IP address based IP accounting

Before introducing the user based IP traffic accounting concept, let us first take a brief look at the user identification process in traditional IP traffic accounting.

Traditional IP traffic accounting is regarded as IP address based because the IP address is used as the unique index to aggregate RDRs. This means that if two RDRs have the same IP address, they are assumed to be generated by the same user. Then these two RDRs will be merged together as one record, which is assigned to the user who owns the IP address. There are two reasons why the traditional IP traffic accounting systems can only provide IP address based IP traffic accounting information.

One is based on the assumption that one IP address is equal to one user. This assumption usually comes from the simplification of the accounting information collection. Since IP addresses can be extracted from IP packets directly, if an IP address is equal to one user, through simply mapping IP address to the user, the IP traffic associated user can be identified. An-

other reason supporting this assumption is the fact that the accounting systems are often served for the organizations which need only coarser granularity accounting information. For example, an ISP concerns only about how much traffic is inside and outside the networks of its costumers, e.g. universities, no matter whether the IP traffic is generated by single user systems or multi-user systems in the universities' networks, although this is of great interest to universities.

Another reason relates to the location of meters in traditional IP accounting systems. Meters are usually placed in the location where not only all needed IP traffic can be monitored but also cost and overhead can be minimized. For example, routers in the network boundaries are typically chosen as a place to integrate the meter function. However, the general placement policy of the meter limits its ability to IP address based IP traffic accounting, since only IP addresses rather than user information can be extracted from the IP traffic. The user information can be collected only from the end-system where the user has been authenticated via login. Outside the end-system no mechanism can distinguish users who generate the IP traffic.

3.2.1 User identification process with traditional IP accounting mechanism

We use an example in Figure 3.2 to illustrate how user information of single user systems is correlated to the corresponding IP traffic in a traditional IP traffic accounting system.

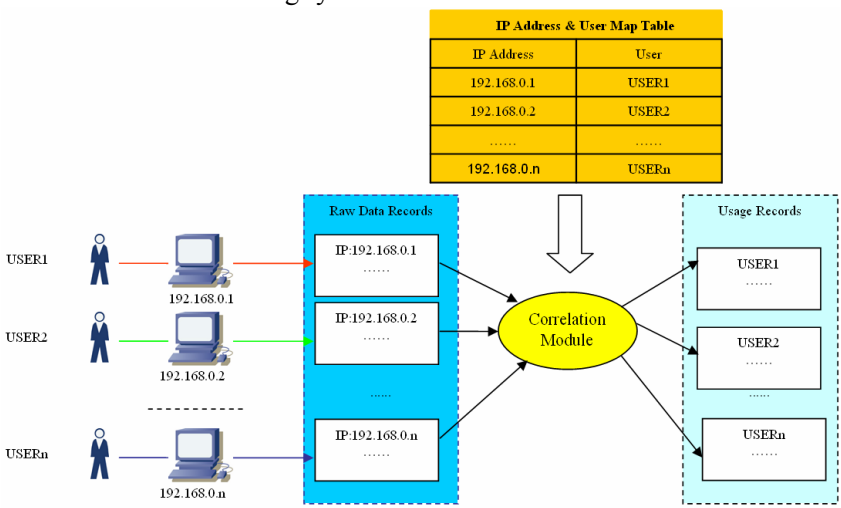


Figure 3.2 IP address – User mapping with traditional IP address based IP traffic accounting

- There are n single user computer systems[2] in the accounting administrative domain. Each computer has an IP address[3] and a corresponding user. The IP traffic associated with these computers is measured by meters.
- Meters store the measurement information of each computer in corresponding RDRs. Every RDR may include not only statistic attributes such as sent and received bytes and status attributes such as timestamp, but also identification attributes such as source and destination IP addresses.
- In the Mediation Layer, an IP Address & User Map Table records the relationship between IP addresses and users. This table may be created when each IP address is registered by the corresponding user. It can also be dynamically adjusted when IP addresses are allocated with a DHCP mechanism. Time duration may also be required in this table to record the valid period for the relationship between an IP address and a user. In this table every IP address corresponds to a single user.
- After the RDRs are collected by the Mediation Layer, the Correlation Module will use the IP Address & User Map Table to identify the user of every RDR. For example, an IP address 192.168.0.1 in a RDR can be found in the IP Address & User Map Table that it is allocated to " USER1". Therefore the Correlation Module identifies this RDR with its corresponding user ID "USER1".

Through this process all URs generated by the Mediation Layer are identified with the corresponding users. In this process the IP Address & User Map Table plays an important role in providing the information about the relationship between IP addresses and users, which can be used by the correlation processing.

3.2.2 User identification with traditional IP traffic accounting mechanism in multi-user systems

A multi-user system may have more than one user sharing the same IP address, which belongs to the multi-user system, during a period of time. The traditional user identification mechanism, which utilizes only IP address to identify the user of IP traffic, cannot distinguish the IP traffic generated by different users in the multi-user system. Figure 3.3 below shows

[2] In this example we use single user systems. The " single user" systems may be real single user systems, or they can also be multi-user systems which are regarded as single user systems.

[3] For simplicity a computer with more than one IP address is not discussed. The system with more than one IP address can be regarded as several single user systems used by the same user.

how user information is correlated to the IP traffic in a multi-user system with the traditional IP traffic accounting mechanism.

- The multi-user system has an IP address 192.168.0.100. It is shared by several users who use the multi-user system.
- Since this multi-user system possesses only one IP address, inbound or outbound IP packets of the multi-user system contain the IP address 192.168.0.100 as destination or source IP address. Consequently, RDRs, generated by the IP traffic accounting meters which measure the IP traffic related to this multi-user system, contain the same IP address attribute " 192.168.0.100".
- In the Mediation Layer, the IP address "192.168.0.100" is identified with a user " USER100" in the IP Address & User Mapping Table. Therefore, after the RDRs correlated to different users in this multi-user system are collected, the Correlation Module maps the IP address "192.168.0.100" in all RDRs to the same user "USER100".

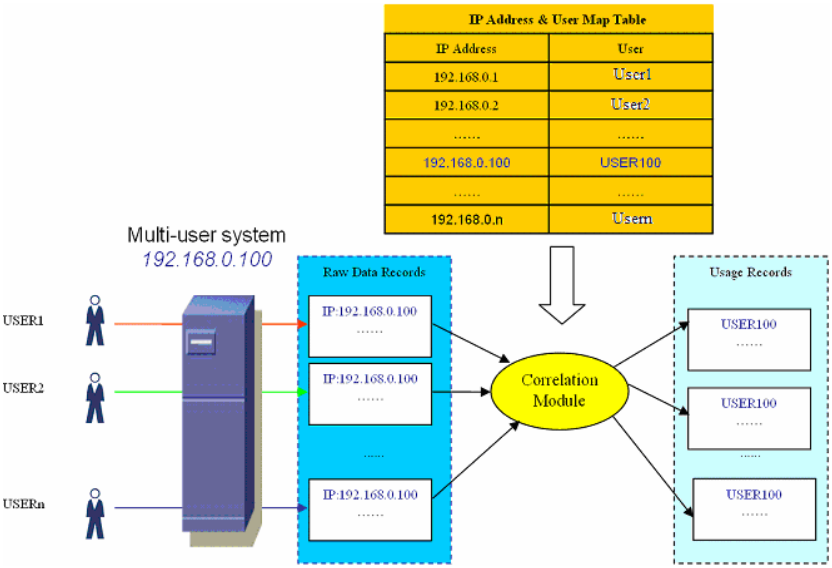


Figure 3.3 IP address – User mapping with traditional IP address based IP accounting mechanism in multi-user system

From the above described user identification process we can see that, although the IP traffic may be generated by different users in the multi-user system, the traditional IP traffic accounting system treats all IP traffic of the multi-user system as generated by the same user, i.e. "USER100". The traditional IP traffic accounting mechanism identifies the originators of the IP traffic by IP addresses, which are bound to hosts instead of users. Therefore, an IP address can only be used to identify a host uniquely. If

this host is a single user host, this IP address and the user of the host can be regarded as identical. However, for a multi-user host in which many users share the same IP address, this IP address cannot be used to identify the originators of the IP traffic related to this host without ambiguity.

This example shows that the IP address based traditional IP traffic accounting mechanism cannot meet the requirements for accurate IP traffic accounting in multi-user systems. The traditional IP address based accounting mechanism is a coarser granular mechanism. In order to distinguish the originators of the IP traffic generated by the same multi-user system accurately, a finer granular mechanism should be developed to provide more accurate accounting information.

3.3 User based IP traffic accounting concept

In order to compensate the insufficiency of the traditional IP traffic accounting mechanism, a user based IP traffic accounting concept was proposed to provide more accurate accounting information in multi-user systems [ZhRM03].

User based IP traffic accounting can be defined as the process of collecting and processing network resource consumption data with corresponding (on the basis of) user information. Before we discuss about it, we should first answer the question: what is a user?

Before introducing the principle of user based IP traffic accounting, some frequently used terms in user based IP traffic accounting are listed:

Accounting Agent

An Accounting Agent is a mechanism residing in the measured host to be responsible for user information collection, processing and transmission.

Dynamic User Traffic Relationship Table (DUTRT)

DUTRT is a table recording the relationship between IP traffic and corresponding users.

Inbound IP traffic

Inbound IP traffic denotes the IP traffic from a remote host to the measured host. From the measured host's point of view, inbound IP traffic is the incoming or received IP traffic.

Measured host

A measured host is the host in which the user based IP traffic accounting mechanism should be applied. Usually the IP traffic of the host is monitored and measured for the purpose of IP traffic accounting.

Outbound IP traffic

Outbound IP traffic denotes the IP traffic from a measured host to destination. From the measured host's point of view, outbound IP traffic is the outgoing or sent IP traffic.

Redirected IP packet

Redirected IP packets are the IP packets which arrive in the measured host and then are forwarded to the meter with user information by the Accounting Agent.

Traffic information

Traffic information indicates the attributes in an IP packet concerning identification attributes, statistic attributes, and status attributes[4].

3.3.1 User model for IP traffic accounting systems

The meaning of the term *user* depends on the context where the term is used. In the Meter Layer, a user is defined as a person who generates the network traffic from an end-system. When talking about traditional IP accounting systems, a user is a host that is the source or the sink of IP traffic. In a multi-user system, a login name or an identifier represents a user. In the Mediation Layer, the term user can denote a person or a group of persons who generate network traffic from the whole accounting administrative domain. In the Application Layer or Billing Layer, the term user denotes a person or a group of persons who should be responsible for the consumption of the network traffic resources. Specifically for the IP Billing, a user is a person or a group of persons who should pay for her/their consumption of resources. Because of this ambiguous usage of the term *user,* we present some definitions of terms, which will be used within this dissertation:

- *Host-Identifier* is a unique identifier for an end-system of the network layer. In the context of IP networks, an IP address can be used as a synonym for a Host-Identifier, since IP addresses are unique numbers for network layer devices, at least within an administrative domain.
- *User-Identifier* or UID is a unique identifier for an account on a measured host. This term is commonly used in the context of multi-user systems.
- *Traffic-Originator* ::= <*Host-Identifier*> [<*User-Identifier*>]. A Traffic-Originator (TO) is responsible for specific outbound and inbound traffic flows. A TO may be described only by a Host-Identifier which means that a TO is an exclusively used computer or by a combination of a Host-Identifier and a User-Identifier which means an account in a multi-user system.

[4] Please refer to chapter 2.2

- *User*::= 1*<*Traffic-Originator*> is a unique identifier for a real person or a group of persons which are associated with one or more TOs. Each TO is exactly associated with one user. Usually a user identifies one real person who has access to one or more single-user systems or accounts on multi-user systems. When a group of real persons shares an account or a single-user system, this group may be described by one user.
- *Purchaser*::= 1*<*User*> is the unique identifier of a person or an institution who will pay for the traffic that is originated by one or more users.

Figure 3.4 illustrates the user model for IP traffic accounting systems.

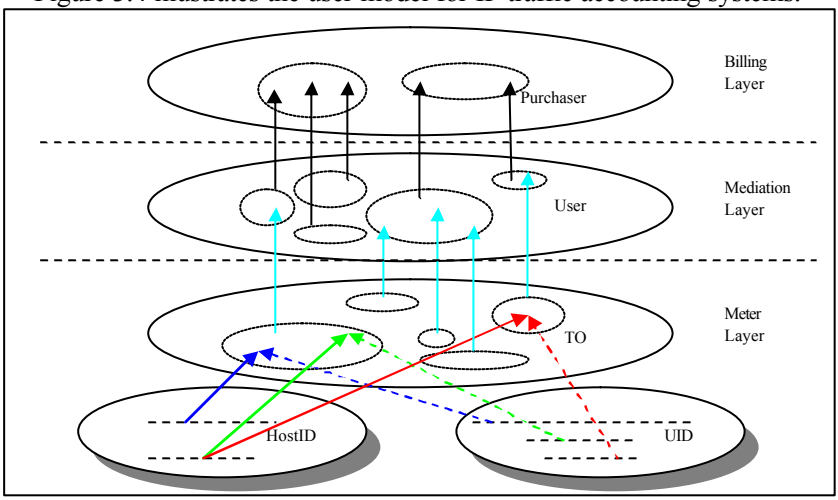


Figure 3.4 User model for IP traffic accounting systems

With this user model, the relationship between UID and HostID in a single user system can be characterized as 1:n (n>=1), which means all IP addresses of the single user system belong to only one user during a period. In this case, all HostIDs can be mapped to the single UID. The relationship between UID and HostID in a multi-user system can be characterized as m:n (m>1, n>=1), which means all IP addresses of the multi-user system are shared by many users at the same time.

A HostID can only be mapped to only one TO, which is the case in traditional IP traffic accounting systems. A HostID combining with different UIDs can be mapped to different TOs. User based IP traffic accounting utilizes this mechanism to identify different users in multi-user systems.

3.3.2 User based IP traffic accounting definition

According to the user model, traditional IP address based IP traffic accounting can be described as the process of collecting and processing network resource consumption information on the basis of the Host-Identifier, i.e. IP address. In this case, an IP address is the key index aggregating RDRs.

User based IP traffic accounting can be described as the process of collecting and processing information of network resource consumption on the basis of both Host-Identifier and User-Identifier. In this case, Host-Identifier and User-Identifier combined as TO are used as the key index aggregating RDRs.

Traditional IP traffic accounting distinguishes different users' IP traffic by different Host-Identifiers, i.e. IP-Addresses only. Moreover, within the Mediation Layer only Host-Identifiers are mapped to users directly. In contrast to traditional IP traffic accounting, the user based IP traffic accounting distinguishes different users of IP traffic by TOs, i.e. by both Host-Identifier and User-Identifier. One TO or a group of TOs in Meter Layer may be mapped to a user in the Mediation Layer. One user or a group of users, in turn, may be regarded as a purchaser in the Billing Layer. It is important to distinguish between TOs, Users and Purchasers, because of their different responsibilities. TO is only used to distinguish which User-Identifier in a host relates to the IP traffic, whereas the user identifies the real person or a group of persons who consume the network resources. If problems occur with some traffic flows or with the volume of traffic that is produced, then it is important to know who is responsible for the traffic. However, in order to send a bill to some person or institution it is only necessary to know who is responsible for paying for the produced traffic.

Table 3.1 and 3.2 give two examples to show how this user model is applied to traditional IP traffic accounting and user based IP traffic accounting, respectively.

Table 3.1 and 3.2 depict an example scenario in a university. Alice is a student. Bob and Charles are employees of the Computer Science (CS) department of the university. Bob is a member of the Work Group 1 (WG1), whereas Charles is a member of WG2. There are four computers with IP addresses IP1, IP2, IP3 and IP4, respectively. IP1 is allocated to Alice's private computer in student dormitory and IP3 is allocated to Bob's computer in his office. These two computers are single user systems. IP2 is allocated to a multi-user computer of the university. IP4 is also allocated to a multi-user computer which belongs to the CS department. Alice uses A1 as

UID for hosts IP1 and IP2. Bob uses different UID B1, B2 and B3 for hosts IP2, IP3 and IP4, respectively. Charles uses J1 as UID for host IP4[5].

From table 3.1 we can find that the traditional IP traffic accounting mechanism utilizes only IP address as TO. With IP address as TO, there is no problem for single user system IP1 and IP3 to identify the users Alice@Home and Bob@CS. But for multi-user systems IP2 and IP4, only two users User@UNI and Employee@CS can be identified respectively. Despite the fact that Alice and Bob have accessed network through computer IP2 as different users, they cannot be distinguished by traditional IP accounting user model, therefore the university, as the purchaser UNI, must pay for the network usage. Similar problem exists in multi-user system IP4.

Real Person	UID	HID	TO	User	Purchaser
Alice	A1	IP1	<IP1>	Alice@Home	Alice
	A1	IP2	<IP2>	User@UNI	UNI
Bob	B1		<IP2>	User@UNI	UNI
	B2	IP3	<IP3>	Bob@CS	CS
	B3	IP4	<IP4>	Employee@CS	CS
Charles	J1		<IP4>	Employee@CS	CS

Table 3.1 An example of the relationship among TO, User, and Purchaser in traditional IP traffic accounting system

Supposing that the accounting policies for network usage in this university are defined as follows:

1. Students must be responsible for all their network usage, e.g. they must pay for it[6].

[5] Here IP address is used as Host-Identifier.

2. Department CS pays for all network usage of their employees.
3. University charges students and faculty for all their network usage.

According to table 3.1, it is obvious that traditional IP traffic accounting system cannot meet the accounting policy because it is unable to distinguish users in the multi-user system IP2 and consequently the accounting policy 1 and 3 defined above cannot be met. Traditional IP traffic accounting systems can still meet the requirements of the accounting policies in multi-user system IP4. According to the accounting policy 2, i.e. CS department will simply pay for all network usage of their employees on this computer, there is no need to distinguish users in this system, and this multi-user system can be regarded as a single user system.

As showed in table 3.2, with user based IP traffic accounting, Host-Identifier and User-Identifier are combined to identify TO. Therefore traffic related with Alice in IP2 can be identified with <IP2, A1> which in turn can be mapped to user Alice@UNI, whereas traffic related with Bob in IP2 can be identified with <IP2, B1> which in turn can be mapped to user Bob@CS. Hence, according to accounting policy 1 and 3, Alice as the purchaser should pay for her network usage in computer IP2, and according to accounting policy 2 and 3, CS department as purchaser will pay for Bob's network usage (first sub column in Purchaser column). If the accounting policy 2 is changed to a more accurate accounting policy, e.g., every work group in CS department should pay for all network usage of their members, with the help of 2-tuple TO the different purchasers can still be distinguished (second sub column in Purchaser column). With this user based IP traffic accounting mechanism, TOs information can help to distinguish which users should be responsible for what traffic related to multi-user systems. Hence, the requirement for finer granular accounting can be met.

Comparing table 3.2 with table 3.1 we can find that, with the traditional IP traffic accounting mechanism, only the IP address is used as TO to identify the consumer of IP traffic. It can only distinguish IP traffic from different hosts. Distinguishing IP traffic from different users with this mechanism can be achieved only when only one user uses a host. If IP traffic related to a single user host should be identified with corresponding users, who may utilize this host in different time with different user names, the traditional IP traffic accounting mechanism can distinguish the IP traf-

[6] The meaning of " responsible" may depend on the management policy, e.g. when a user generates traffic volume less than 1GB in a month she pays nothing, otherwise she pays 1 cent for every extra 1MB or she still pays nothing but she is banned from accessing network for a period of time as a result of her excessive network resource consumption.

fic's originators with the help of timestamp combined with login information recorded in system log.

Real Person	UID	HID	TO	User	Purchaser	
Alice	A1	IP1	<IP1, A1>	Alice@Home	Alice	Alice
	A1	IP2	<IP2, A1>	Alice@UNI	Alice	Alice
Bob	B1	IP2	<IP2, B1>	Bob@CS	CS	WG1
	B2	IP3	<IP3, B2>	Bob@CS	CS	WG1
	B3	IP4	<IP4, B3>	Bob@CS	CS	WG1
Charles	J1	IP4	<IP4, J1>	Charles@CS	CS	WG2

Table 3.2 An example of the relationship among TO, User and Purchaser in user based IP traffic accounting system

Table 3.3 compares the ability of identifying IP traffic with corresponding users by traditional IP address based traffic accounting and user based traffic accounting under different situations.

According to Table 3.3, the following situations are considered:

1. One user has one or more User ID in a host.

   If this host is a single user host, IP address based IP traffic accounting can identify the user of IP traffic with the help of IP address and the log information about users' login and logout. User based IP traffic accounting can identify the user of the IP traffic with the 2-tuple TO <User ID, IP address>.

   However, if this host is a multi-user host, the IP address based IP traffic accounting can identify the user of the IP traffic only when this host is used by only one user. Otherwise, when there are more than one user that utilize the host at the same time, the users of IP traffic cannot be identified by the IP address based accounting mechanism. There is no problem for user based IP traffic accounting to identify users of IP traffic in the multi-user host.

Scenario	Number of Real Person	Number of User ID	Number of host	IP address based IP traffic accounting		User based IP traffic accounting	
				Single user host	Multi-user host	Single user host	Multi-user host
1	1	n	1	✓	✗	✓	✓
2	1	n	n	✓	✗	✓	✓
3	n	1	1	✗/✓	✗	✗/✓	✗/✓
4	n	1	n	✓	✗	✓	✓
5	n	n	1	✓	✗	✓	✓
6	n	n	n	✗/✓	✗	✗/✓	✗/✓

Table 3.3 A comparison of the user identification abilities between IP address based traffic accounting and user based traffic accounting under different conditions

2. One user has more than one User ID in different hosts. If these hosts are single user hosts, both accounting mechanisms can distinguish the users of IP traffic. For multi-user host, only user based traffic accounting can identify the user of the IP traffic. IP address based accounting has the same problem as in situation 1.
3. Several users have only one User ID in a host, i.e. these users share one User ID in a host. These users may share the User ID in a host in two different ways. One is that every user logs in with the shared User ID and accesses the network only when other users with the same User ID logout. Another is that after login with the User ID, one or several other users access the network from the same host at the same time.

   The first share manner is a time-share manner of the User ID. In this case, the user identification of IP traffic in single user system can be achieved with the help of the log information and timestamp by either

IP address based traffic accounting mechanism or user based IP traffic accounting mechanism. However, the user identification of IP traffic in multi-user systems can be achieved only by the user based traffic accounting mechanism.

With the second User ID share manner, no matter in single user system or in multi-user system, no matter using IP address based traffic accounting or using user based traffic accounting, the users of the IP traffic cannot be distinguished. For example, user A logs into a computer, uses a browser to access the Internet for 10 minutes and leaves without logout. After that, user B can use the same computer to access the Internet without login. Another case is that user A logs into a multi-user system with her UserID, and user B also logs into the same host with the UserID of user A from another terminal. In these two cases, it is difficult to distinguish the Internet access activities performed by user B. Only user A is regarded as the user of the IP traffic which is produced after her login in the computer. However, for different users running different applications under the same user ID, which is normal in distributed computing environments, the user model should be extended for distinguishing the users of IP traffic in this case. Chapter 8 will illustrate the details of the extended user model.

4. Several users have the same User ID in different hosts. This is the case when different users register in different hosts with the same User ID. Despite the fact that the User IDs used by different users are the same, these users login into different hosts. Therefore, their IP traffic can be identified with the corresponding users according to IP addresses in single user systems. However, in multi-user systems only user based traffic accounting can distinguish the user of the IP traffic.
5. Several users have different User IDs in one host. If the host is a single user host, these users must use this host exclusively at a time. Therefore, both IP address based traffic accounting and user based traffic accounting can identify the users of IP traffic in this case. However, if the host is a multi-user host, only user based traffic accounting can distinguish the user of the IP traffic.
6. Several users have different User ID in different hosts. This can be simplified as case 3, 4, or 5.

From the above we can see that traditional IP traffic accounting can only provide host level IP traffic accounting information, whereas user based IP traffic accounting can provide more accurate and finer granular accounting information.

The user based IP traffic accounting extends the concept of traditional IP accounting by considering User-Identifiers in addition to Host-

Identifiers. Traditional IP traffic accounting can be regarded as a special case of user based IP traffic accounting, since in traditional IP traffic accounting the TOs of the traffic flows related to the same host have always the same User ID. Compared with traditional IP traffic accounting, user based IP traffic accounting should provide information about User-Identifiers as a measurement metrics in RDRs, which must be correlated to the accounted IP traffic. Therefore, the following issues should be taken into consideration in user based IP traffic accounting:

- Identify traffic with corresponding User ID and IP address. The relationship between a TO and a traffic flow must be recorded. User and IP traffic relationship information can be collected only within the multi-user system, since this information is not available outside of the multi-user system.
- Store and transfer user traffic relationship information.
- Correlate RDRs with TOs for identifying users of RDRs.

Since user based IP traffic accounting concentrates on the user concept in Meter Layer, i.e. TO, hereafter the term user and TO are used to mean the same user term in this layer.

3.4 Related work

Till now, several standards and documents have been published by IETF concerning the Internet Accounting. However, all of them are IP address based IP accounting. [RFC1272] introduced the basic information about the Internet accounting architecture whereas [RFC2063] suggested the traffic flow measurement architecture. A flow based accounting system "NeTraMet" [Netr] has been implemented based on the [RFC2063] suggested the Internet accounting architecture. Although the user information has been defined as attributes "flowDataSourceSubscriberID" and "flowDataDestSubscriberID" in the MIB [RFC2720], these two attributes are neglected by the RTFM implementation "NeTraMet". No user identification and user information processing function was integrated in "NeTraMet". Actually, the IP packet capturing mechanism with the help of Libpcap and Winpcap cannot obtain any user information of the IP traffic. In this implementation, user information may be obtained simply by mapping the IP address to its owner.

Many reseaches have also been made in Internet Accounting and charging. M3I (Market Managed Multi-service Internet) project provided a framework in managing Internet resource usage through market forces, i.e. through charging and accounting multiple levels of network services. With this framework, customers can purchase their required services and QoS, whereas congestion can be reduced through pricing. However, despite that

the CAS system in M3I intends to provide network resource usage information on a per-customer basis, only IP address based user IP traffic accounting and charging can be realized due to the fact that routers are commonly supposed to be the location in which the traffic information is collected [M3I00] [KSSW00].

In ETSI's document three charging relationships between Internet service providers and customers are summarized [ETSI99]:

- Every provider chages the end user individually
- One provider chages the end user on behalf of the all providers
- One providers buys the service from the other providers, and the user is only a customer of one of the providers

But in user based IP traffic accounting, a new intermediate role exists, i.e. the host provider, between the users and ISPs. Therefore, except that the users are charged directly by ISPs, the host provider can also charge its users on behalf of the providers or buy services from ISPs. Currently, due to lacking of user based IP traffic accounting mechanism, host providers are difficult to distribute network costs to its users.

S. Blott et al proposed a user level billing and accounting mechanism in IP networks [BMBB99]. A special purpose network probe, called NetCounter, is applied to correlate network traffic with individual users that generated it in real-time. The principle of the NetCounter is illustrated in Figure 3.5.

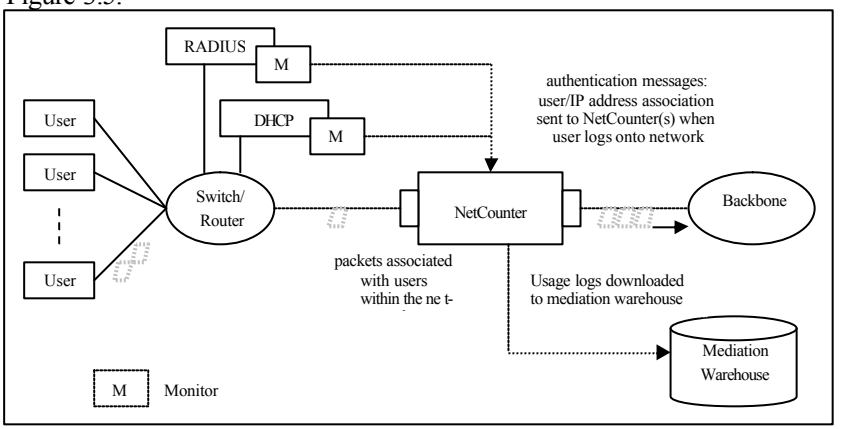


Figure 3.5 NetCounter – A Dual-Ported Accounting Device within an IP Network [BMBB99]

In Figure 3.5, the NetCounter is a network device with two network interfaces. It aims at collecting usage data for individual users with the help of RADIUS or DHCP servers. When a user logs on, she is assigned an IP address when she registers herself with either RADIUS or DHCP server.

Then the association of user and IP address is sent to NetCounter immediately. After that when this user sends and receives IP packets, the NetCounter can identify the user of these IP packets according to the registered user IP address association. This user identification mechanism is a typical user identification mechanism applied in traditional IP accounting systems. User identification operation is performed outside the users' hosts through checking the logs of user IP address association in authentication devices. This user identification mechanism is still IP address based due to the fact that the user IP address relationship is 1:1. When a multi-user system is allocated an IP address by a DHCP server, the IP traffic generated by different users in this system cannot be distinguished with this NetCounter mechanism.

Many commercial accounting systems such as XACCT [XACCT], NARUS [NARUS], etc. have also implemented their accounting systems with a similar Internet accounting architecture. However, they all provide IP address based accounting. Their user identification process is achieved with the same mechanism as NetCounter through applying logs of user IP address association.

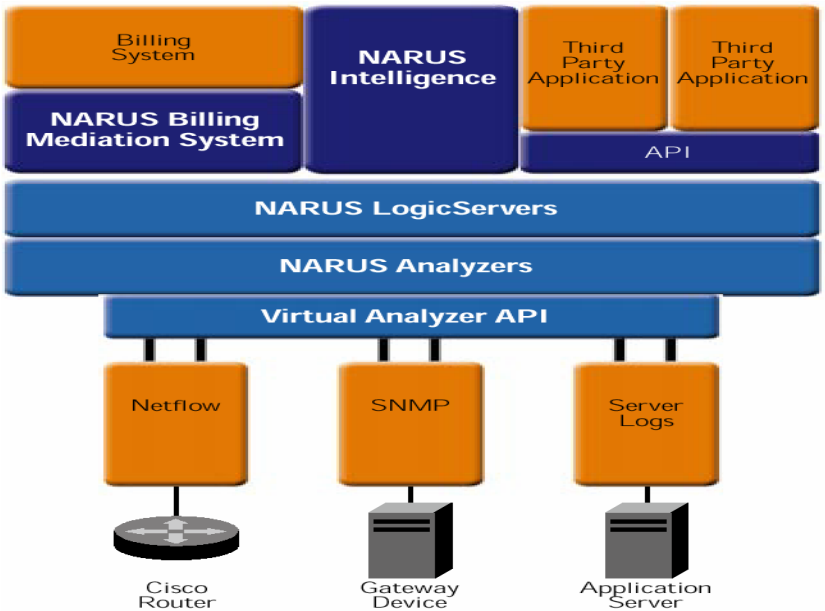


Figure 3.6 NARUS Internet Business Infrastructure (IBI) [NARUS]

Figure 3.6 illustrates the NARUS accounting system architecture, which is a typical traditional accounting system architecture. From this architecture, we can find that all IP traffic information is directed and captured

with the help of third party network elements such as routers, gateways. The meters of this architecture are located in key position devices such as routers and gateway in which user information cannot be collected directly. Whereas the Server Logs in an Application Server can only provide application related information which is usually used for enhancing and correlating traffic information gathered by network devices. This system is still IP address based and it concentrates mainly on accounting information processing.

There are not many researches concerning the theme of user based IP traffic accounting. R. J. Edell et al proposed a user based IP traffic accounting method only for TCP traffic [EdMV95]. The principle of this method is to intercept the TCP connection establishment request, and then to verify user's authorization of network resource usage to control the TCP connection establishment. If a user's TCP connection is allowed, corresponding measurement on this TCP flow will be recorded. Figure 3.7 depicts the process of TCP connection establishment with this method.

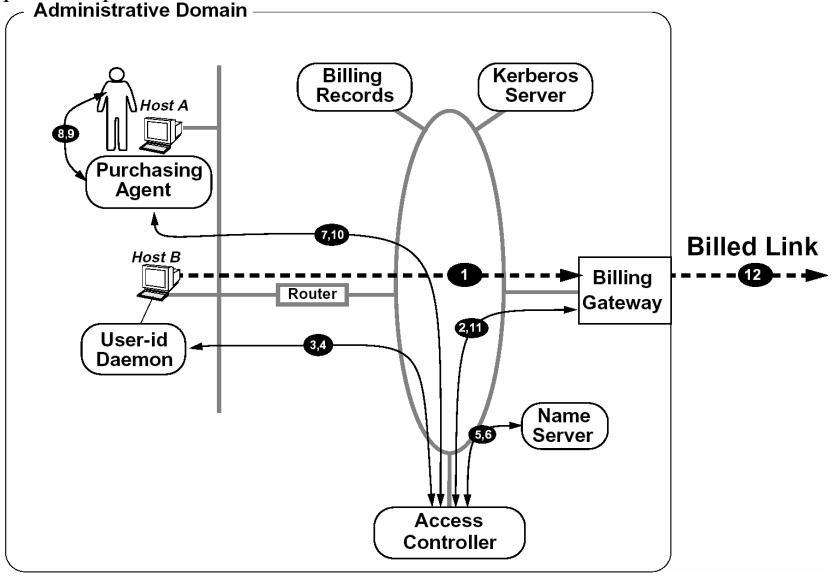


Figure 3.7 TCP connection establishment process in billing system suggested by [EdMV95]

According to the process illustrated in Figure 3.7, when a user's application on host B attempts to establish a TCP connection to a remote host through sending a TCP SYN message, the Billing Gateway (BGW) intercepts this message and holds it on to ask the Access Controller whether the user initiating this message is allowed to build this TCP connection with the outside world. At first, the Access Controller asks the User-id Daemon

to identify the user of the TCP SYN message. Then the Access Agent queries the Purchasing Agent of the user to verify whether the user is able to pay for this connection. After that, the Access Controller responds the payment information to the BGW. In the end, if the payment cannot be afforded, the BGW denies the connection. Otherwise, the BGW forwards the TCP SYN message to the remote host and creates an entry for this connection. The created new entry of the connection will be used for identifying the user of successive IP packets of this TCP connection. Through this mechanism, user based IP traffic accounting can be achieved for TCP traffic.

Although this user based IP traffic accounting mechanism works well for TCP traffic, it has several limitations. This method cannot be applied for metering non-connection oriented protocol traffic such as UDP, and an extra verification server is needed. Checking a TCP SYN message requires the TCP header to be examined, which may result in more CPU burden in BGW. When IPSec is applied, the TCP header may be encrypted by the ESP [RFC2406] protocol. In this case, TCP connection establishment packets cannot be detected by the Billing Gateway. Consequently, this method cannot be applied to communications with IPSec mechanism. How this mechanism coexists with tunnel mechanisms is also not explained.

[RFC1413] also suggested a simple Identification protocol to retrieve user information of established TCP connections. This protocol utilizes an identification server application to serve for requests querying about user identification of established TCP connections. Requests contain information of port number pair related to TCP connection. The identification server retrieves user information of the established TCP connection according to the port number pair. If the requested user information is found, it is sent with a response message to the querist. Otherwise, an Error message will be sent back. Figure 3.8 illustrates the principle of this protocol. This protocol may encounter errors in querying user information for short time lived TCP connections, and it is also impossible to meter the UDP traffic.

In the early stage of IPng development, B. Carpenter suggested utilizing IP packets to carry triplet <source, destination, transaction> for the purpose of facilitating traffic authentication, policy-based source routing and detailed accounting [RFC1671]. Further in [RFC1672] N. Brownlee proposed integrating an accounting tag into the transaction field of the aforementioned triplet.

An accounting tag can be an arbitrary string identifying the party responsible for the packet. This tag can be used by the meter to simplify identifying the party responsible for the traffic flow. Initially an accounting tag would be set by the host when a packet is sent. At that stage, the tag

would identify the user who sends it. The tag could be changed along the packet's path to the destination to identify the party responsible for this packet.

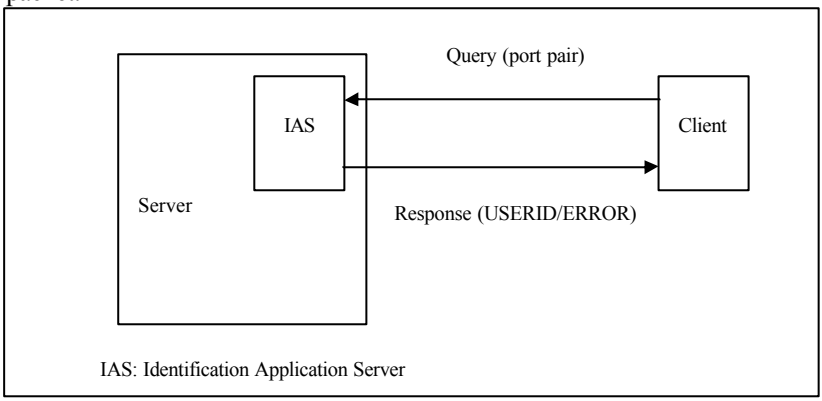


Figure 3.8 Identification protocol proposed by [RFC1413]

[RFC1672] does not explain the concrete accounting tag mechanism. How it is constructed, how it is transferred, how it is collected by meter, security and implementation issues, etc. are not specified in this document. Finally in the IPv6 protocol, the user information is not taken into consideration into the protocol design. With the development of the Internet, the issue arises again: should people have identities that cross application boundaries. This issue is now taken into consideration by GENI for designing new Internet architecture. "A Future Internet should include a coherent design for the various name-spaces in which people are named. This design should be derived from a socio-technical analysis of different design options and their implications. There must be a justification of what sort of identification is needed at different levels, from the packet to the application" [ClSF06].

The Host Identification Protocol (HIP) [RFC4423, MNJH06] is now an effort made by the IETF HIP WG for the purpose of decoupling of host identifiers and locators (i.e. IP addresses). HIP introduces a new name space, the "host identity" name space, to the Internet architecture. With this new name space, some challengers in mobility, multihoming, IPv6 transition and network level secrity can be solved. Till now, this protocol is still under developed. By design HIP is located between IP layer and transport layer in the TCP/IP stack. For user based IP traffic accounting, this protocol may be extended to convey both host identity and user identity information.

Considerations have been made on collecting, storing and transferring IP traffic related user information in [Baue00]. The focus of it was to study

the possibility of utilizing IP packets conveying user information. Concrete solutions and mechanisms were not suggested.

" UserIPAcct" [Smey01] has realized a per user IP statistic mechanism in Linux. It can provide information about users' bytes sent and received on the IP level. However, this implementation is limited in open source OS Linux.

The NIPON project was proposed to aim at improving the accuracy and granularity of IP traffic accounting technique in multi-user environment with user based IP traffic accounting mechanism [Muel00]. At first traditional IP traffic accounting technology and products were surveyed, analyzed and compared [Zhan01]. Then an Accounting Agent mechanism was suggested for user based IP traffic accounting [ZhRM02]. Based on the Agent mechanism, a user based IP traffic accounting system architecture with out-of-band scheme was suggested [ZhRM03]. In addition, a user based IP traffic accounting prototype system was developed [NIPO03]. In [ZhRM05], several user based IP traffic accounting schemes were discussed. The research results of user based IP traffic accounting described in this dissertation are based on the achievement of the NIPON project.

3.5 Overview of the user based IP traffic accounting technique

User based IP traffic accounting requires all IP traffic to be identified with the corresponding users, i.e. with corresponding TOs. Then this user information should be stored and transferred. With this user information, different RDRs can be correlated to their consumers. User information of IP traffic related to a multi-user host can only be obtained inside the host. In other words, inspecting IP traffic related to the multi-user host cannot acquire any accurate user information. Hence, a mechanism must be integrated into the multi-user host to perform the task of identifying IP traffic with the corresponding users. This mechanism is called Accounting Agent. The Agent can be located in different layers of the TCP/IP stack for collecting user information in the measured host. It can identify users of IP traffic on the basis of packets, flows or TCP connections. User traffic relationship information can be stored and transferred with the in-band or out-of-band method.

3.5.1 Accounting Agent

The key of user based IP traffic accounting is to identify the related user of IP traffic. Since IP header can only carry the IP address information, which is not enough to identify a user uniquely in multi-user systems according to previous discussion, an additional mechanism is needed to asso-

ciate the IP traffic with the corresponding users. This identification mechanism is called Accounting Agent mechanism.

An Accounting Agent, in short Agent, is responsible for recording the relationship between IP traffic and user. Its functions include:

- Identify IP traffic with corresponding user.
- Store the IP traffic and user relationship information temporarily.
- Transfer IP traffic and user relationship information. If needed, some security mechanisms should be applied to guarantee the confidentiality and integrity of data transmission.
- Perform access control according to network access control policies.

3.5.2 Agent location

Considering the meter location principle introduced in 2.2.2, in order to identify the user of IP traffic, the Agent should be located in the place where both IP traffic and user information can be gathered. Although IP traffic information can be obtained outside the measured host from monitoring the IP headers, user information cannot be extracted from IP headers. Therefore, the Agent must be integrated in the measured hosts. The TCP/IP reference model is composed of five layers: Application Layer, Transport Layer, Internet Layer, Data link Layer and Physical Layer [Tane03]. The Data link Layer and Physical Layer are not suitable for integrating Agent due to their hardware related characteristics. The other three layers are possible for integrating the Agent. Considering the fact that the Transport Layer and the IP Layer are usually implemented as a TCP/IP kernel in operation systems, figure 3.9 shows the possible locations for an Agent to be integrated in a multi-user system.

If the Agent is integrated into the Application Layer, it is easy to obtain the user information. This may be realized by providing network APIs which bind Agent functions. However, this cannot be applied to the legacy applications which did not bind these APIs. Another problem is that some users may avoid using these APIs because of its metering function. Since the measurement information may be used for charging and billing purposes, the information should be dependable. Agents integrated into every application instance may be easily attacked or cheated. Dependability in this environment is hard to be achieved.

Another problem is that if the Agent works in the Application Layer, the collected user information may be inserted into Application Layer PDUs for IP packet transmission. This is inefficient for a meter to extract accounting information from Application Layer PDU of IP packets.

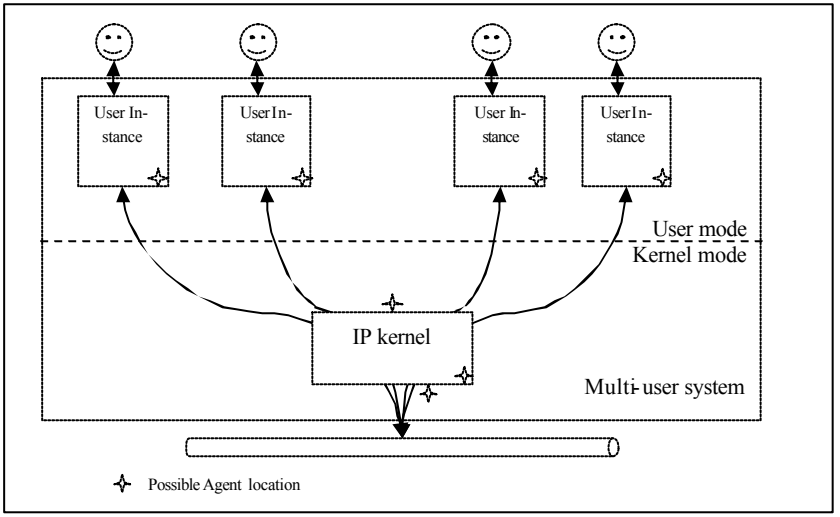


Figure 3.9 Possible locations for integrating Agent into a multi-user system

The second possible location is right above the IP kernel. In this position, the Agent can intercept all IP requests between the IP kernel and applications. This does not require the Agent mechanism to be integrated into every application. IP traffic generated by legacy applications can also be measured, since this Agent works in a non-intrusion position, i.e. the Agent is transparent to the applications and users. Since the Agent may cause system performance decline, therefore it should be carefully designed. Otherwise, the Agent will become system bottleneck. This position is suitable for implementing the Agent in non open source OSs. In the NIPON project, the user based IP traffic accounting prototype system realized the Accounting Agent at this position in Solaris and Windows 2000 Terminal Server, respectively [NIPO03]. These two OSs are not open source systems. With this method, the direct modification on OS can be avoided. However, this location is not suitable for in-band scheme which integrates user information in the IP packets, because the user information can only be inserted into transport layer's PDU at this location.

The third position is in the IP kernel. Agents integrated at this location can work efficiently. This is suitable for open source OSs like LINUX or OS producers. The disadvantage is that the Agent cannot be integrated into legacy OSs without source code or support from OS producers.

The fourth position is underneath the existing implementation of an IP protocol stack, between the native IP and the local network drivers. This position is called "Bump-in-the-stack" (BITS) in IPSec [RFC2401]. Inte-

grating the Agent in this position may not require to access source code. The disadvantage of this position is that user information may not be able to be obtained. According to our experience in the Agent implementation in Windows 2000 Terminal Server, if the Agent is located in this position, user information cannot be obtained directly. A mechanism must reside above the TCP/IP stack implementation to provide user information of IP packets.

Table 3.4 summarizes the characteristics of these four possible Agent locations according to above analysis.

	Effect on performance	Need source code of OS	Security	Coexistence with legacy applications
In Application	Low	No	Bad	No
Above TCP/IP	May be high	No	Good	Yes
In TCP/IP	Middle	Yes	Good	Yes
Below TCP/IP	May be high	No	Good	Yes

Table 3.4 Characteristics of possible locations for Agent implementation

3.5.3 IP traffic identification methods

An Agent identifies the users of IP traffic when the IP traffic passes through the Agent. The Agent can identify users of the IP traffic on the basis of IP packets, IP traffic flows or TCP connections.

3.5.3.1 IP packets based user identification

With the IP packets based user identification method, the Agent intercepts every IP packet to identify its corresponding user, and then generates records to store the user IP traffic relationship information. The advantage of this method is that it is very accurate. But its disadvantage is also obvious, i.e. its efficiency is very low, since every packet must be checked and identified, and, accordingly, the relationship between every IP packet and its user must be recorded. If a User IP Traffic relationship table is generated to store the IP traffic identification results, this table will become huge. Therefore, for IP packets based user identification method, inserting the user information into the IP packets is a better choice than the User IP Traffic relationship table mechanism (see chapter 4, 5).

3.5.3.2 IP traffic flow based user identification

The IP traffic flow identification method is an improved method compared with the IP packets based identification method. This method identifies the user of each flow instead of each packet. Especially with the help of the Flow Label in the IPv6 header, flow recognition becomes easier and simple. The Agent needs only to check the Flow Label rather than to extract source IP, source port, destination IP and destination port information from the IP header to distinguish flows as processing IPv4 flows.

Flow can be defined as a uni-directional or bidirectional traffic from the source end-point to the destination end-point during a period. The traffic flow based user identification process is described as follows:

1. When an IP packet passes through an Agent, the Agent identifies its user and extracts traffic flow attributes such as source IP, source port, destination IP and destination port from its IP header. For IPv6 packets with Flow Labels, the Agent simply extracts this information from the IP packets as flow attribute.
2. Then the Agent uses these flow attributes with the user information as a key index to search the user IP traffic flow relationship table.
3. If there exists an entry for this user IP traffic flow relationship, then this entry can be updated (or no update needs to be made on this entry if this table stores only relationship information).
4. If there is no entry for this user IP traffic flow relationship, a new entry is created for it. In user based IP traffic accounting, the 6-tuple <source IP, source port, destination IP, destination port, start time, end time>in IPv4 or <Flow Label, start time, end time> in IPv6, which identifies a flow uniquely, is combined with the User ID to construct an entry in the user IP traffic flow relationship table.
5. If the user traffic flow relationship information is stored locally, i.e. it is not transferred with IP packets, it should be reported to or collected by the correlation module regularly to correlate flows with the corresponding users.

The advantage of this method is that it greatly decreases the size of the user IP traffic relationship table. User information will be recorded only when the first IP packet of a flow appears. This can also save bandwidth for transferring user information. Flow Labels in IPv6 headers can further improve efficiency of the flow identification process. However, this method still needs to check every IP packet to decide to which flow the packet belongs.

3.5.3.3 TCP connection based identification

The TCP connection based user identification method monitors every TCP connection and finds the corresponding user of the TCP connection. This method can also be regarded as a special traffic flow based identification method. The principle of this method is that the Agent monitors and records only the connection establishment and release activities of a TCP flow to generate the user and the TCP flow relationship information. The external meter records all packets of this TCP flow and utilizes the user TCP flow relationship information to identify the user of the TCP packets during the correlation process. In this process, the Agent is called only when the TCP connection is established or released. Other packets of the TCP connection generated after connection establishment will not be inspected by the Agent. Therefore, the performance decline caused by the Agent can be reduced. Moreover, the performance for metering TCP traffic can be improved. [EdMV95] adopted a similar mechanism to control TCP connections. The limitation of the connection based identification is that it can only be applied to TCP traffic. For UDP, the connectionless transport protocol, every packet is still required to be checked to identify the user. The user identification process can be described as follows:

1. When a TCP connection request packet with a TCP SYN message comes into the Agent, it finds out the user related to this TCP connection and then extracts this TCP connection related information such as source IP, source port, destination IP, destination port from the request packet.
2. The Agent records the user TCP connection relationship information either into a local table or directly into the TCP connection packets. Then the IP packet is forwarded.
3. After that, other TCP packets of the same connection will not trigger the Agent.
4. When a TCP connection termination message TCP FIN comes into the Agent, it also finds out the corresponding user and updates the user TCP connection relationship information in the local table or inserts user information into this IP packet. Then the IP packet is forwarded.

3.5.4 User traffic relationship information storage and transmission

After the IP traffic is identified with the corresponding users, this relationship information should be stored and then transferred to the meters. Here we consider two different storage and transmission methods: in-band method and out-of-band method.

1. The in-band method utilizes the IP packets to carry their corresponding user information. According to the discussion in 2.2.4, accounting attributes can be classified into three categories: identification attribute, statistic attribute and status attribute. These attributes, such as IP address, bytes, etc., are already included in the IP header of the IP packet. With in-band method for user based IP traffic accounting, user information is inserted into IP packets to identify the users of corresponding IP packets. With this method, the IP packets can convey the user traffic relationship information to the meter actively. This method does not require the user and IP traffic relationship information to be stored in the measured host.
2. The out-of-band method does not utilize IP packets to convey their user information. The collected user and IP traffic's relationship information is encapsulated in dedicated user information message packets and transferred to the meter separately from the identified IP traffic. Before the user IP traffic relationship information is transferred, it can be stored in the measured host temporarily.

3.6 User based network access control

Collecting users' network resource consumption is not the goal of user based IP traffic accounting technique. Providing statistic information about users' network resource consumption is for the purpose of charging, billing, network access control, etc. Access control is a commonly applied mechanism to adjust network resource allocation through limiting or prohibiting network usage. Access control is usually applied by ISPs when a user's quota, such as duration, volume, deposit, etc., is exhausted.

With the help of traditional IP address based traffic accounting, access control can only control users' network access on the basis of IP address, i.e., access control can be performed only on the host level. This kind of access control is similarly called traditional IP address based access control. Access control must be user based, since quota allocation and calculation are based on users. For traditional IP address based access control, an IP address belongs to only one user in a period. Therefore, a general network access control architecture with the IP address based traffic accounting mechanism is shown in Figure 3.10.

In the Figure 3.10 illustrated traditional access control architecture, users' network consumption information is collected by meters, which are usually integrated in the network elements such as routers, switches, etc. Different users' network usage can be calculated and their corresponding quotas will be checked according to the accounting information. If a user's quota is reached, access control policy will be applied. For example, this

user will not be allowed to access the network any more. Therefore, the access control component will inform access devices such as routers to reject forwarding IP packets related to this user's IP address.

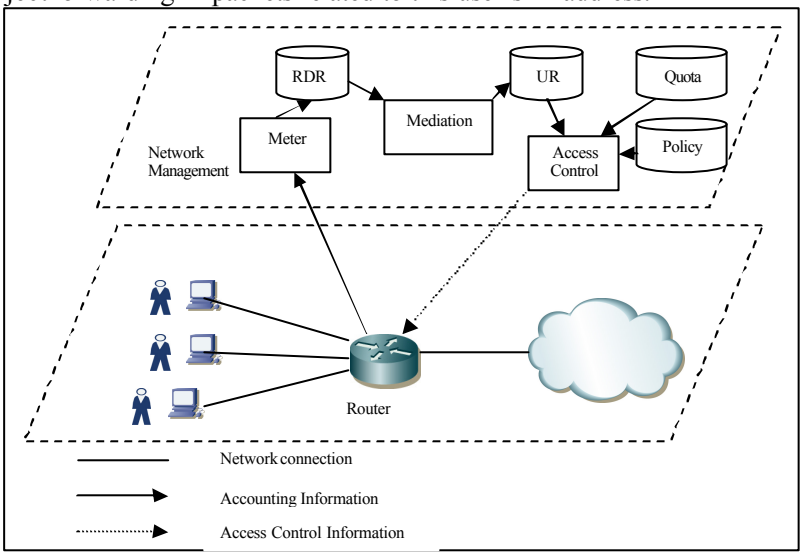


Figure 3.10 Traditional access control architecture with IP address based IP accounting

The traditional IP address based access control suffers also from the similar problem as the traditional IP address based IP traffic accounting, i.e. it cannot achieve access control on the basis of users in multi-user systems. The inability of controlling access on the basis of users by the traditional IP address based access control mechanism lies in two factors. One factor is that traditional IP address based accounting cannot provide user based IP traffic accounting information. This makes it impossible to calculate users' network resource consumption in multi-user systems. Another factor is that no access control mechanism can perform user level network access control, since usually access control can be applied only on the basis of IP addresses in the network access devices. Even if user based IP traffic accounting information can be provided for calculating different users' network resource consumption, without the corresponding user based access control mechanism the more accurate user based access control cannot be achieved.

With the help of the user based IP traffic accounting technique, not only user based IP traffic accounting information can be generated, but the

Agent can also play a user based access control role. The user based IP traffic access control architecture is depicted in Figure 3.11.

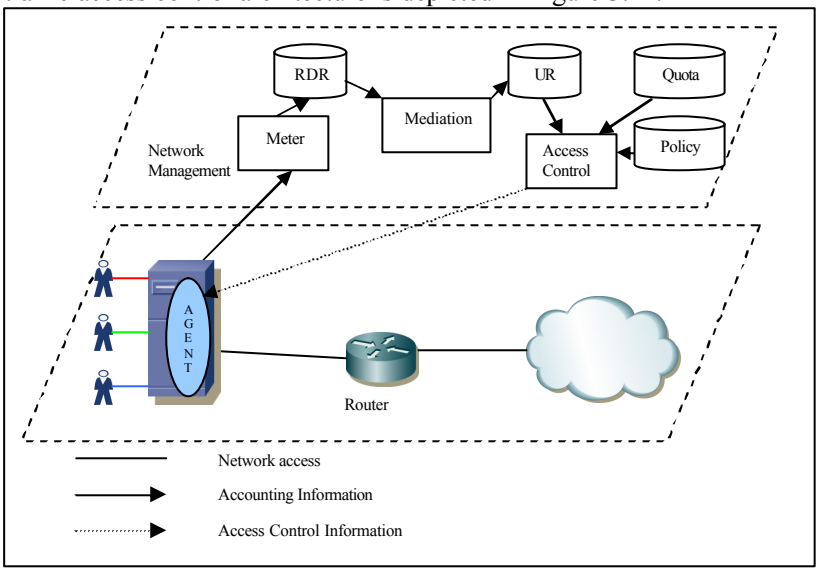


Figure 3.11 User based access control architecture with user based IP traffic accounting

The main difference between Figure 3.11 and Figure 3.10 is the Accounting Agent located in the multi-user host. With the help of the Accounting Agent, User based IP traffic accounting can be achieved. A meter located in the key position, e.g. in a router, cooperates with the Agent to collect network resource consumption information on the basis of users (user based IP traffic accounting schemes are discussed in chapter 4, 5, 6). Routers or other access devices can only provide IP address based access control. User level access control in multi-user systems can be made only in the measured hosts. With the Agents integrated in the multi-user systems, user based access control can be easily achieved.

In the user based access control architecture, the following two components are critical for performing access control: Accounting Agent and Access Control component.

The Agent performs the following functions for user based access control:

1. Monitor and validate different users' network access authorization.
2. Maintain and synchronize access block list. This includes downloading a block list from the access control component, processing block list synchronization between Agent and access control component by receiving update information from the access control component.

3. Block or allow users' network access according to the block list.

The access control component performs the following functions:

1. Calculate different users' network usage according to the accounting information provided by the user based IP traffic accounting system.
2. Compare different users' network usage with their quota.
3. Maintain block lists of different hosts. Update the block list when a user's quota is used up or when a blocked user obtains new quota.
4. Synchronize block lists with different hosts by informing Agents to update their block lists.

The Figure 3.12 shows the user based access control process with the help of the user based IP traffic accounting mechanism.

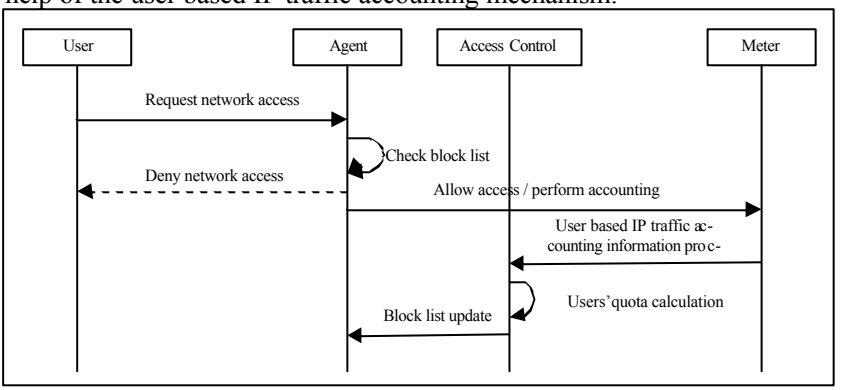


Figure 3.12 User based access control process

The user based access control process in Figure 3.12 is described as follows:

1. When a user's IP packet passes through the Agent, the Agent intercepts this IP packet.
2. The Agent checks the block list to verify if this user is allowed to access the network. The block list is stored in the measured host. Usually the block list is downloaded from the Access Control component when an Agent is started. The Access Control component keeps all block lists of the managed hosts.

   The block list in the Agent will be updated in time by the Access Control component when the corresponding host's block list stored in the Access Control component happens to be changed. In the following situations, the block list will be modified and the synchronization of the block lists in Agents and the Access Control component will be started accordingly:

   - A user just runs out of her quota

- A user is just allocated new quota, which makes this user be removed from the block list.

3. When the user of the IP packet is in the block list, the IP packet will be discarded by the Agent.
4. When the user of the IP packet is not in the block list, the IP packet will be forwarded by the Agent to its destination and a corresponding user based IP traffic accounting operation will be performed.
5. The Meter collects the user based IP traffic accounting information to provide user based network resource consumption statistic information for the Access Control component.
6. The Access Control component keeps checking the user based IP traffic accounting information to calculate different users' quota.
7. When a user's quota is used up, the Access Control component updates the corresponding block list and informs the user related access device to block this user.

3.7 Summary

This chapter analyses the flaw of the traditional IP traffic accounting mechanisms in providing finer granular accounting information in multi-user hosts. After surveying some related work concerning IP traffic accounting, we can conclude that most of the existing IP traffic solutions are IP address based. Even though there were a few works concerning user based IP traffic accounting, they were imperfect and concerned only some part of the user based IP traffic accounting technique. The IP address based accounting mechanism of traditional IP traffic accounting cannot meet the requirement of finer granular accounting in multi-user hosts. Therefore an innovative traffic accounting mechanism, the user based IP traffic accounting concept, is suggested in this dissertation.

This chapter introduces the principle of user based IP traffic accounting. The key component of user based IP traffic accounting is the Accounting Agent. It is responsible for identifying the users of IP traffic, recoding and storing the user IP traffic relationship, and transferring the information to the Mediation Layer. The Accounting Agent must be located in the measured host. Otherwise, the user information of IP traffic cannot be gathered. There are four possible implementation positions for integrating Accounting Agent into the measured host. These positions have different advantages and disadvantages. Integrating the Accounting Agent in which position is a trade-off among performance, efficiency and the difficulty of implementation. The user identification of IP traffic can be either IP packets based, IP traffic flow based, or even TCP connection based. IP traffic flow based user identification may be the best solution, but sometimes IP

packets based user identification is a supplementary to the IP traffic flow based user identification. With the help of user based IP traffic accounting, more accurate user based access control can also be achieved. This requires the user based IP traffic accounting information be calculated and the Accounting Agent is used to control the network access on the basis of users.

This chapter introduces only the general concept of user based IP traffic accounting. In the next three chapters, three different user based IP traffic accounting schemes will be explained in detail. These three schemes utilize different mechanisms to identify users of IP traffic, and to store and transfer the user traffic relationship information.

Chapter 4 In-band scheme

In this chapter, the in-band scheme for user based IP traffic accounting is introduced. The in-band scheme utilizes IP packets themselves to convey their corresponding user information. With this mechanism not only the accounting required statistic attributes such as sent bytes, received bytes but also the TO information can be extracted directly from IP packets.

This chapter is organized as follows: at first the principle of the in-band scheme is explained, then the User Information option format is defined and described, after that several possible locations for integrating the User Information option in an IP packet are discussed and compared, then security considerations are made in detail, and in the end several issues concerning implementation are analyzed.

4.1 Principle of the in-band scheme

According to the user model defined in chapter 3, the aim of user based IP traffic accounting is to identify the corresponding user of IP traffic with TO, i.e. IP address and User ID. From analysing IP packets we can find that not only the IP traffic accounting required statistic attributes, such as received bytes, sent bytes, etc., but also a part of the identification attributes for user based IP traffic accounting, i.e. IP address, are available in IP packets. For traditional IP address based IP accounting, all required accounting information can be obtained from the IP packet. Only the user information cannot be extracted from IP packets directly.

The idea of the in-band scheme is to integrate user information into the IP headers of corresponding IP packets. Through that, user based IP traffic accounting required accounting attributes can all be extracted from IP packets directly. With the in-band scheme, user information can be stored and transferred by IP packets. No local storage in the measured host is required.

The principle of the in-band scheme, which utilizes IP headers for storing and transferring user information to achieve user based IP traffic accounting, can be described as follows:

- For outbound IP traffic, the Accounting Agent identifies users of IP packets and then integrates user information into IP headers of the corresponding IP packets. After that, the IP packets are forwarded to their destinations. A meter located in key position intercepts the IP packets tagged with user information and extracts this information directly from IP packets for the purpose of user based IP traffic accounting.

- For inbound IP traffic, when an inbound IP packet passes through the Agent, it intercepts the inbound IP packet and identifies its corresponding user. Then the Agent forwards the IP packet to its receiving application on the one hand, and integrates user information into a copy of the IP packet and redirects the copy of the inbound IP packet tagged with user information to the meter.

Figure 4.1 illustrates the principle of in-band scheme.

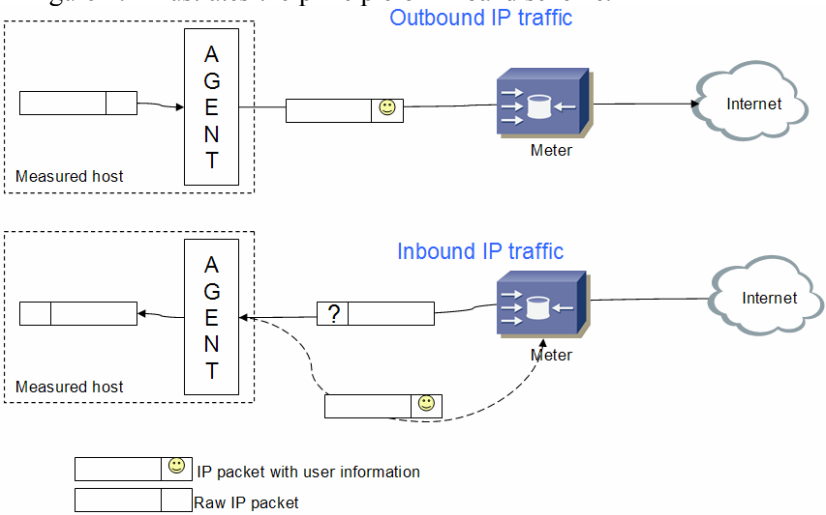


Figure 4.1 Principle of in-band scheme

4.1.1 Components in the in-band scheme

As shown in Figure 4.1, the following components exist in the in-band scheme:

- **Measured host**

The measured host can be either a single user system or a multi-user system which requires the user based IP traffic accounting mechanism to be applied.

For multi-user systems, it is obvious that user based IP traffic accounting is necessary for distinguishing different users' network usage as discussed in chapter 3. For single user systems, if different users' network usages are also required to be distinguished, with the in-band scheme user based IP traffic accounting can also simplify the accounting process, since user information and other accounting attributes can be directly extracted from IP packets. Consequently, log information about different users' login in single user systems does not need to be collected for correlating RDRs with corresponding users.

- **Accounting Agent**

An Accounting Agent is integrated into the measured host to fulfil the following functions:

  - Identify IP packets with corresponding users
  - Insert user information into IP packets
  - Forward IP packets tagged with user information to the meter
  - Maintain the dynamic user traffic flow relationship table (DUTRT) and keep synchronization with the DUTRT in the meter
  - If security mechanisms should be applied to the in-band scheme, the Agent is responsible for negotiating security parameters, encrypting and decrypting user information in the User Information option[7].
  - Perform access control

- **Meter**

In the in-band scheme, a meter can work similarly as in traditional IP accounting to extract all required accounting attributes from IP packets. For the in-band scheme, the meter has to fulfil the following extra functions:

  - Extract both user information and traffic information from IP packets.
  - For secure user information transmission between Agent and meter, it must negotiate security parameters with Agents and must encrypt and decrypt user information.
  - Maintain the dynamic user traffic flow relationship table (DUTRT) and keep synchronization with the DUTRT in the Accounting Agent.
  - Identify users of IP packets without user information with the help of DUTRT. Query the Agent for user identifiers of unrecognized IP packets.

With the in-band scheme, all required accounting attributes can be extracted from the IP packet directly. Therefore, meters can still be located in the key place of the network to collect the IP traffic information as in the traditional IP traffic accounting architecture. Usually meters are located in the network boundary to measure IP traffic into or out of the administrative domain.

4.1.2 User identification

User identification is the process of finding the users of IP traffic and integrating user information into the corresponding IP traffic.

[7] Please refer to chapter 5.2 about the User Information option format

In the in-band scheme, the Agent is responsible for identifying only the users in the measured host in which the Agent resides. This means that the Agent cares only about the local user of the two peers involved in a communication. For outbound IP traffic, the Agent is responsible for identifying the users who send the IP packets. In this case, the TO is identified by the 2-tuple <**Source IP, User ID**>. For inbound traffic, the Agent is responsible for identifying the users who receive the IP packets. In this case, the TO is identifed by the 2-tuple <**Destination IP, User ID**>.

User identification can be IP packet based, IP traffic flow based or hybrid:

- With the IP packet based user identification method, user information is inserted in every outbound and inbound IP packet.
- With the IP traffic flow based user identification method, user information is only inserted in the first IP packet of an IP traffic flow. A new entry about this flow and its user will be added in a DUTRT when this IP packet arrives into the meter. The successive IP packets of the same flow can be identified with their corresponding user by searching the DUTRT.
- With the hybrid user identification method, user information is inserted in every outbound IP packet. But for inbound IP packets, only the first IP packet of a flow is integrated with user information and is redirected to the meter by the Agent.

4.1.3 User identification for outbound IP traffic

After being identified with the corresponding users, outbound traffic and inbound traffic will be forwarded by the Agent in different directions: the outbound traffic will be sent to the remote host, whereas the inbound traffic will be forwarded to the receipt application in the measured host and also to the meter if necessary. Considering the different forwarding characteristics for outbound traffic and inbound traffic, the Agent should treat these two types of IP traffic differently.

How IP traffic is processed to carry user information with the in-band scheme depends on different user identification methods. Therefore the user identification of outbound IP traffic is discussed according to the IP packet based method, the IP traffic flow based method and the hybrid method, respectively.

4.1.3.1 IP packet based user identification method

With the IP packet based user identification method, the user based IP traffic accounting process for outbound IP traffic can be described as follows:

1. When an outbound IP packet passes through the Agent, the Agent finds out the corresponding user information of this IP packet from the system.
2. The Agent inserts the user information into this IP packet. How user information is formatted and how user information is integrated into the IP packet are discussed in 4.2 and 4.3, respectively.
3. The Agent forwards this IP packet tagged with user information to the meter.
4. When the meter receives an IP packet tagged with user information from the measured host, it extracts the user information and the traffic information from the IP packet, which is similar as in traditional IP accounting.
5. The meter forwards this IP packet to its destination. But before this IP packet is forwarded to its destination, the meter may delete the user information from the IP packet. This can prevent user information from being transported to the outside and reduce the extra traffic caused by user information.

4.1.3.2 IP traffic flow based user identification method

With the IP traffic flow based user identification method, the user based IP traffic accounting process for outbound IP traffic can be described as follows:

1. When an outbound IP packet passes through the Agent, the Agent finds out the corresponding user information of this IP packet from the system.
2. For IP traffic flow based user identification, the Agent checks the DUTRT in the measured host to verify if this IP packet belongs to an existing IP traffic flow. If there is no corresponding entry in the table, it means that this IP packet is the first IP packet of an IP traffic flow, therefore a new entry about this new flow and its corresponding user is created and added into the DUTRT. Then the Agent inserts the user information into the IP packet. If there is already an entry corresponding to the IP packet in the table, it means that the IP packet is not the first IP packet of the corresponding flow and the user information of this flow has been already sent to the meter in the first IP packet. Therefore, the user information does not need to be inserted into this IP packet.
3. Then the Agent forwards the IP packet to the meter.

   When an IP packet tagged with user information comes into the meter, at first, the meter extracts the user information and the traffic information from the IP packet, and then a new entry will be added into

the DUTRT of the meter to identify this flow and its corresponding user by the meter.

If an IP packet without user information comes from a measured host into the meter, at first, the meter extracts traffic information from the IP packet, and then the meter searches its DUTRT to check if this IP packet belongs to an existing flow. If there is an entry in the DUTRT, the statistic attributes value in this IP packet will be calculated with the statistic attributes value of the same flow in the DUTRT. Otherwise, this IP packet will not be forwarded to its destination. In this case, in order to improve the dependability, the meter should send back this IP packet to the Agent asking for the user information of an unrecognized flow. When the Agent receives this feedback query message, it finds out the user of this IP packet and forwards it with its user information to the meter again. Detailed explanation about the dependable DUTRT synchronization mechanism is described in 4.1.6. If no user information of this IP packet can be found, this IP packet will be discarded by the Agent.

4. The meter forwards the IP packet to its destination.

4.1.3.3 Hybrid user identification method

The user based IP traffic accounting process for outbound IP traffic with hybrid user identification method is the same as the IP packet based user identification.

4.1.4 User identification for inbound IP traffic

For the inbound IP traffic processing, things become a little more complex. When an IP packet from outside the administrative domain and destined to a measured host in the administrative domain comes into a meter, no user information about the receiver of the IP packet can be found in this IP packet. The receiving user of this IP packet can be identified only when this IP packet arrives in the Accounting Agent. But when this IP packet arrives in the destination host, because this host, in which the Agent resides, is the last hop of the IP packet, the IP packet does not need to be further forwarded outside the host. This means that, despite the fact that the IP packet can be identified with the corresponding user by the Agent, its user information cannot reach the meter outside the host actively. Therefore, after the Agent intercepts the IP packet and identifies the corresponding user, it should perform two operations: one is to hand this IP packet further on to the corresponding receiving instance in the host system, and the other operation is to forward the IP packet tagged with user information to the meter outside the host.

Just like the outbound IP traffic, how inbound IP traffic is processed with the in-band scheme also depends on different user identification methods. Below we explain them separately.

4.1.4.1 IP packet based user identification method

With the IP packet based user identification method, the user based IP traffic accounting process for inbound IP traffic can be described as follows:

1. When an IP packet destined to a measured host comes into a meter, the meter just forwards this IP packet to its destination. In this method, the user based IP traffic accounting meter is triggered only when IP packets from the measured hosts come in.
2. When an inbound IP packet comes into the measured host, the Agent intercepts the IP packet and finds out the corresponding user information of this IP packet from the system. Then it makes a copy of the IP packet.
3. The Agent forwards the IP packet to the receiving application in the measured host.
4. The Agent builds a User Information option which contains not only the user ID of the IP packet but also the original source IP address. Then it inserts the User Information option into the IP packet copy. After that, it modifies the source IP address to the IP address of the measured host and the destination IP address to the IP address of the meter. In the end, the Agent sends the modified IP packet copy to the meter.
5. When the meter receives an IP packet tagged with user information destined to it, it means that this IP packet is a redirected inbound IP packet. The meter extracts the user information and the traffic information from the received IP packet for further user based IP traffic accounting processing.

4.1.4.2 IP traffic flow based user identification method

With the IP traffic flow based user identification method, the user based IP traffic accounting process for inbound IP traffic can be described as follows:

1. When an IP packet destined to a measured host comes into a meter, the meter extracts traffic information from this IP packet. Then the meter searches the DUTRT to check if this IP packet belongs to an existing flow. If an entry in the table can be found, the user of this IP packet can be identified and the statistic attributes of this flow are recalculated.

If there is no entry in the table, it means that this IP packet is the first IP packet of a new inbound flow and the user of the flow is not identified by the Agent. In this case, the meter inserts a Query User Information message into the IP packet (about the format of Query User Information message please refer to 4.2.2).

2. The meter forwards this IP packet to the destination host.
3. When an IP packet comes into the measured host, the Agent intercepts the IP packet and finds out the corresponding user information of this IP packet from the system. Then the Agent searches its DUTRT to check if this IP packet belongs to one of this user's flows.
4. If an entry is found in the DUTRT indicating that this IP packet belongs to one of this user's flows and the inbound IP packet contains no Query User Information message, the Agent forwards the IP packet to the receiving application in the host.
5. If an entry is found in the DUTRT and the inbound IP packet contains a Query User Information message, the Agent performs two operations. One is to delete the Query User Information message from the IP packet and forward it to the receiving application in the host; the other is to insert the user information into a copy of the IP packet and redirect the new IP packet tagged with user information to the meter.
6. If no entry can be found in the DUTRT, or a flow is found in the DUTRT but it belongs to a different user, it means this is the first IP packet of an inbound flow, consequently a new entry is added into the DUTRT with the new flow information and its corresponding user information. After that, the Agent hands this IP packet on to the receiving application in the host. Then the Agent inserts the user information of this IP packet into its copy and redirects this new IP packet to the meter.
7. When the meter receives an IP packet tagged with user information destined to it, t means that this IP packet is a redirected inbound IP packet. The meter extracts the user information and the traffic information from the received IP packet and adds a new entry into the DUTRT with the new flow information and its corresponding user information. The corresponding statistic calculation will also be made according to this IP packet.

4.1.4.3 Hybrid user identification method

The user based IP traffic accounting process for inbound IP traffic with hybrid user identification method is similar to the IP traffic flow based user identification except that the flow in DUTRT is uni-directional, i.e. only inbound flows will be recorded in the DUTRT. The DUTRT can be

regarded as a user inbound IP traffic flow relationship table. A new entry is added into the DUTRT only when the meter receives a redirected IP packet tagged with user information.

4.1.5 Different user identification methods comparison

With the IP packet based user identification method, every outbound IP packet carries its user information and every inbound IP packet is redirected to the meter with added user information. Therefore, the user information integrated in each IP packet produces extra IP traffic in the network, and the inbound IP traffic volume of measured hosts is doubled. The advantage of this method is its simplicity and low processing burden onto both Agent and meter, since no DUTRT needs to be maintained by the Agent or the meter.

With the IP traffic flow based user identification method, only the first IP packet of a flow must be integrated with user information. Other IP packets of the same flow do not need to carry user information and not all inbound IP packets must be redirected to the meter. This can significantly reduce the extra IP traffic caused by user information transmission. Less processing overhead will result by integrating and extracting the User Information option. However, the method will result in more processing overhead on both Agent and meter for maintaining the DUTRT and searching IP packets' users in the DUTRT. In addition, the synchronization of DUTRTs in both Agent and meter should be carefully designed.

By analyzing the IP packet based user identification method we can find that most of the extra traffic generated by this method is due to inbound IP packets redirection. Te processing overhead of the IP traffic flow based user identification method lies in searching the corresponding user and flow information in the DUTRT for every IP packet. With the hybrid method, all outbound IP packets are integrated with their corresponding user information, whereas for inbound IP packets only the first IP packet of an inbound flow must be redirected to the meter. The hybrid method will cause less processing overhead than the IP traffic flow based method, since all outbound IP packets are equipped with user information and there is no need to search the DUTRT for these IP packets. In addition, the hybrid method will produce less extra IP traffic than the IP packet based user identification method, since only some of the inbound IP packets are required to be redirected to the meter.

Below is a comparison table of these three user identification methods according to above analysis:

	IP packet based method	IP traffic flow based method	Hybrid method
Outbound IP packet tagged with user information	Every IP packet	First IP packet of every bi-directional flow	Every IP packet
Inbound IP packet redirection with user information	Every IP packet	First IP packet of every bi-directional flow	First IP packet of every uni-directional flow
Extra traffic volume	Large	Small	Small
Processing overhead	Low	High	Medium

Table 4.1 Comparison of different user identification methods.

The three user identification methods have different advantages and disadvantages. Which method should be chosen for in-band scheme depends on the application environments and the trade-off between performance and extra traffic volume. Although the IP traffic flow based method and the hybrid method produce less traffic burden, in some situations the IP packet based method may be the best choice. For example, if IPSec is applied and the payloads of IP packets are encrypted, flow information cannot be extracted directly from the IP packet. In this case, the flow based user identification mechanism cannot be applied and the IP packet based user identification method is the only choice.

4.1.6 Dynamic user IP traffic relationship table (DUTRT)

In the in-band scheme, the dynamic user IP traffic relationship table (DUTRT) records IP traffic and its corresponding user for keeping history records of user IP traffic relationships to avoid integrating user information into all IP packets of the same flow.

In the in-band scheme, the IP packet based user identification method does not require a DUTRT. The DUTRT records user and IP traffic flow relationship information. A traffic flow is defined as a sequence of packets between given source and destination endpoints during a period of time [RFC2724, RFC3917]. Flows may be bi-directional or uni-directional and have different granularities (see 2.2.3.2). The direction and granularity of flows are usually decided by accounting policies.

According to the user model, several attributes in an IP traffic flow and the corresponding user information should be collected by the Accounting Agent to construct a record about this flow into the DUTRT. Whenever the first IP packet of an IP traffic flow is captured by the Agent, the identification attributes of the flow and its corresponding user information are collected, and the Agent creates a new entry into the table.

The in-band scheme should define flows with application-to-application level granularity, since end-to-end level granularity cannot distinguish between different users in multi-user systems. Hence, 7-tuple <**Source IP Address, Destination IP Address, Source Port Number, Destination Port Number, Protocol number, Start Time, End Time**> is suitable to define a flow for DUTRT. Usually an entry in DUTRT may have the following attributes: <**User ID, Source IP Address, Destination IP Address, Source Port Number, Destination Port Number, Protocol number, Start Time, End Time**>.

A record in the DUTRT may include two kinds of attributes:

- Traffic-Originator attribute: it is used to uniquely identify a TO that is related to the traffic. As for the above example, a TO attribute includes these items: <**UserID, Source IP**> for outbound IP traffic flow or <**UserID, Destination IP**> for inbound IP traffic flow.
- Identification attribute: identification attributes are the information that can be used to identify an IP packet or an IP traffic flow uniquely. The identification attributes can be directly extracted from IP packets. The Agent extracts them from IP packets to generate user IP traffic flow relationship records.

  Since the identification attributes in IP traffic can also be extracted from IP traffic by meters to generate RDRs, the identification attributes are used as a key to associate DUTRT records with corresponding RDRs for the purpose of mapping RDRs to the corresponding users. [RFC2924] has defined the attributes and formats of RDRs. As for the above example, correlation attributes include these items: <**Source IP, Source Port Number, Destination IP, Destination Port Number, Protocol number, Start Time, End Time**>. These attributes uniquely identify an IP traffic flow.

In order to control the size of the DUTRT, inactive flows should be deleted from the table regularly. Through that, memory can be recycled and the searching performance can be improved. After an inactive flow's entry is deleted from DUTRT, when an IP packet belonging to this deleted flow appears, a new entry can be created into the table for this IP traffic flow.

Since both Agent and meter maintain their own DUTRT, these two DUTRT should be kept synchronized. Otherwise, the users of flows cannot be correctly identified.

Below we discuss DUTRTs synchronization mechanisms between Agent and meter according to outbound IP traffic flow and inbound IP traffic flow, respectively.

4.1.6.1 DUTRT synchronization with outbound IP traffic flow

If a flow is initiated by the measured host, the Agent finds out its user and creates a new entry into the DUTRT. Then the Agent sends the IP packet tagged with user information to the meter. The meter updates its DUTRT with this new flow and its user. This is a standard DUTRT synchronization process with outbound IP traffic flow. After that, all successive IP packets of the same flow will not be tagged with user information.

Maybe in some situations, the first IP packet tagged with user information of a flow is lost during its way to the meter. This may happen, for example, due to network failure or congestion. In this case, the DUTRT in the meter will not be updated. However, the successive IP packets of this flow may arrive in the meter without carrying user information, since the Agent supposes that the first IP packet of this flow has been received by the meter. In this case, the meter cannot find the user from its DUTRT. The worst case is, when there exists an old flow in the meter's DUTRT with the same flow characteristics as the new flow (for example, the same source and destination IP addresses, the same source and destination ports and the same protocol, except that the old flow belongs to another user), the meter will use this old flow from DUTRT to identify the user of the inbound IP packet. In this case, this inbound IP packet of the new flow is incorrectly treated as the old flow and the user of the new flow is also incorrectly identified with the user of the old flow.

In order to avoid the synchronization problems described above, the following acknowledgement mechanisms should be adopted:

- The Agent continues to insert user information of a flow in its successive IP packets until it receives a User Information Acknowledgement message from the meter. The acknowledgement message can be either in-band or out-of-band. With the in-band acknowledgement, the meter continues to insert the acknowledgement message in the inbound IP packets of the same flow until the Agent does not insert user information in the outbound IP packets of the flow. With the out-of-band method, the meter sends an acknowledgement message to the Agent through a special channel between Agent and meter. In order to avoid unnecessary acknowledgements and reduce extra traffic, the meter can choose to respond only to large volume flows. For example, meter responds acknowledgements to the Agent only when it receives more than five IP packets tagged with user information of the same flow.

  Figure 4.2 illustrates the information exchange processes for synchronizing DUTRTs between the Agent and the meter according to the mechanism described above:

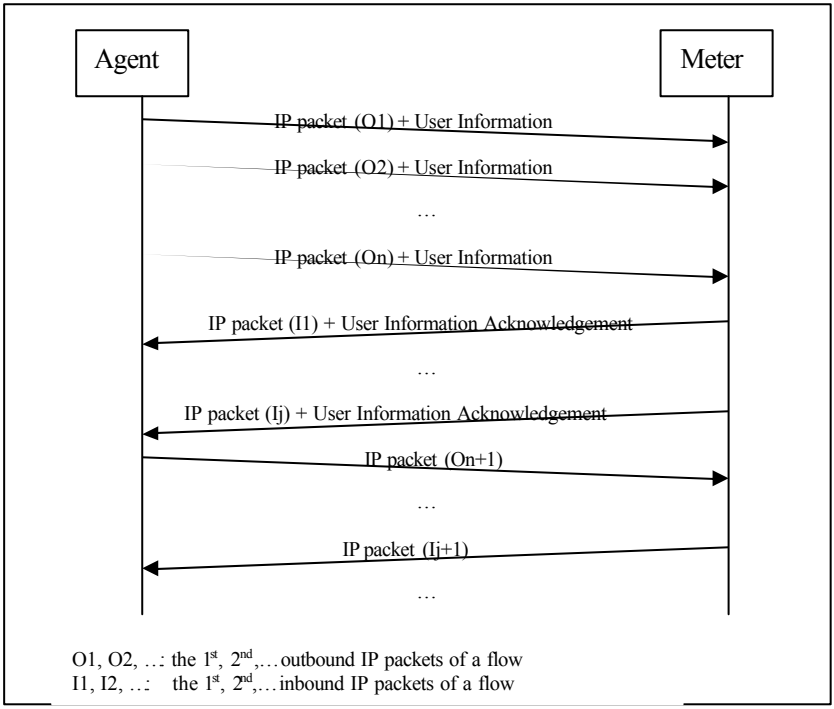


Figure 4.2 DUTRT synchronization with outbound IP traffic flow

- The mechanism described above can be improved to decrease unnecessary traffic. When a new flow is initiated, if the Agent can find out an old flow with the same flow characteristics as the new flow from the DUTRT, but the old flow and the new flow belong to different users. Only in this case, the Agent will continue to insert user information in successive IP packets of this new flow until an acknowledgement message from the meter is received.

  If there is not a flow with the same flow characteristics belonging to a different user, the Agent integrates the user information only in the first IP packet of the flow. After the first IP packet tagged with user information of the flow arrives in the meter without problem, the successive IP packets of the flow can be identified with the corresponding user.

  If the meter receives an outbound IP packet without user information, it will not forward this IP packet. The meter inserts a Query User Information message into this meter and sends it back to the Agent. When the Agent receives an IP packet with a Query User Information message, it finds out the user of this IP packet. Then it inserts the user

information into this IP packet and sends this IP packet to its destination again.

Figure 4.3 illustrates this improved DUTRT synchronization mechanism in case no transmission failure happens to the first IP packet of a flow. Figure 4.4 illustrates this improved DUTRT synchronization mechanism in case the first IP packet of a flow is lost:

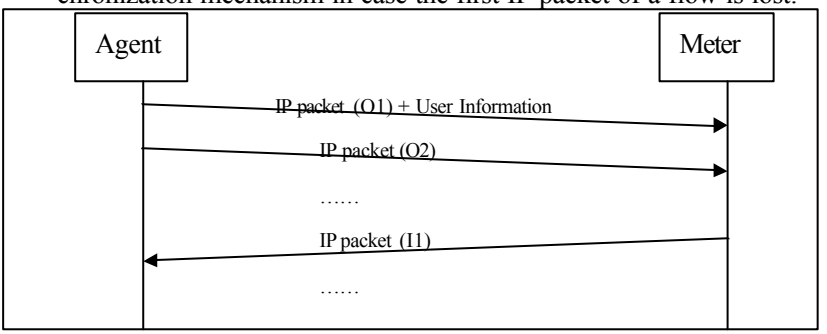


Figure 4.3 Improved DUTRT synchronization mechanism with outbound IP traffic flow in case that the first IP packet tagged with user information arrives in the meter

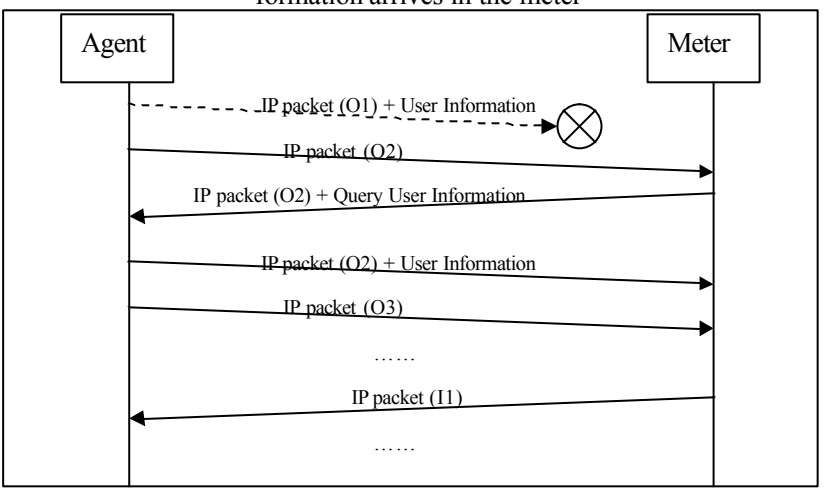


Figure 4.4 Improved DUTRT synchronization mechanism with outbound IP traffic flow in case that the first IP packet tagged with user information is lost

4.1.6.2 DUTRT synchronization with inbound IP traffic flow

If a flow is initiated by a remote endpoint, when its first IP packet arrives in the meter, the meter searches its DUTRT to check if this IP packet belongs to an existing flow.

If no entry is found in the DUTRT, the meter forwards this IP packet with a Query User Information message to the measured host.

If an entry can be found from the DUTRT, the meter forwards this IP packet without any change to the measured host. In this case, the found flow may be an old flow with the same flow characteristics belonging to a different user. However, the meter cannot distinguish that. Therefore the user of the new flow may be incorrectly identified with the user of the old flow. The correct user information of this new flow can be obtained only after the Accounting Agent redirects this IP packet tagged with user information to the meter. In order to prevent this kind of error, a rollback mechanism should be applied to recover the statistic calculation on the wrong user.

When an inbound IP packet arrives in the measured host, the Agent intercepts this IP packet and finds out the user information of this IP packet. If this IP packet carries a Query User Information message, the Agent inserts its user information into this IP packet and redirects the new IP packet to the meter. If this IP packet carries no User Information option, the Agent searches its DUTRT to verify if it belongs to an existing flow with the same user. If no entry can be found, or a flow is found but the user is not the same, the Agent inserts user information into this IP packet and redirects the new IP packet to the meter. The IP packet will not be redirected only when both flow and user information of a found entry match the packet.

When the meter receives the redirected IP packet tagged with user information, it checks if this user information is a response to a Query User Information message. If so, a new entry with this new flow information and its user information will be added into the DUTRT. If it is not a response to a Query User Information message, it means that the user of this IP packet was incorrectly identified and a rollback operation has to be made. At the same time, a new entry describing this new flow and its user is created into the DUTRT. In order to inform the Agent that the DUTRT is updated for this new flow, the meter should insert the new user information into the first successive inbound IP packet of this flow after updating DUTRT. When the Agent receives IP packets with correct user information, it knows that the DUTRT has been updated for this new flow, and the successive inbound IP packets of this flow without user information will not be redirected to the meter.

Figure 4.5 illustrates the information exchange processes for synchronizing DUTRTs in the Agent and in the meter when the flow is initiated by a remote host:

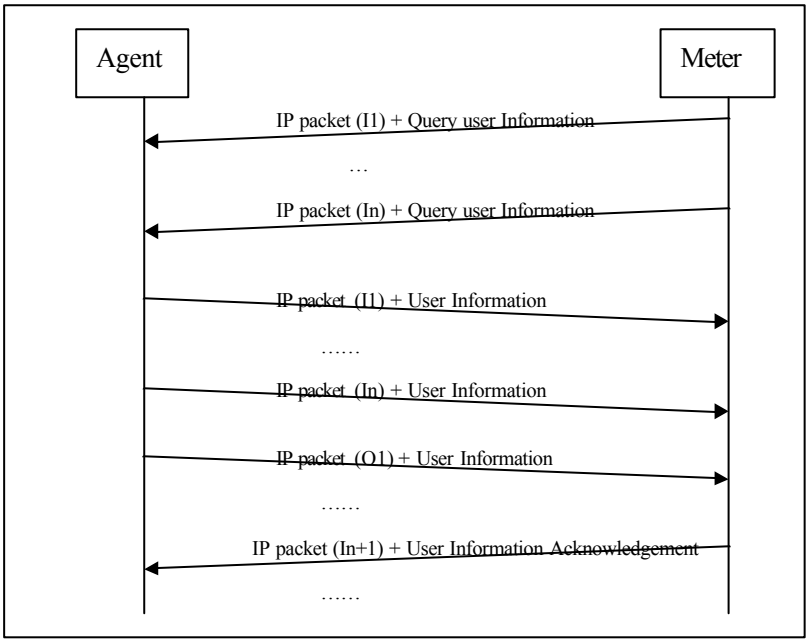


Figure 4.5 Message exchange for DUTRT synchronization with inbound IP traffic flow in case no entry is found in DUTRT for the inbound IP packets of a new flow

4.2 User Information option format

An IP packet consists of an IP header and a payload. The IP header contains information explaining the characteristics of the IP packet, how it should be routed, how it should be processed, etc. The payload contains the upper layer datagram. In order to utilize IP packets to carry user information, the IP header should be extended to accommodate the user information. When the IP header is extended for the purpose of user based IP traffic accounting, the following requirements should be met:

1. The IP header extension must be able to accommodate the required User Information option;
2. The IP header extension should support both IP packet based user identification and IP traffic flow based user identification;
3. The security and privacy of the user information should be taken into consideration.

For user based IP traffic accounting, a new option called User Information option is defined to be integrated into IP packets for the purpose of carrying user information. Figure 4.6 illustrates the general format of the User Information option.

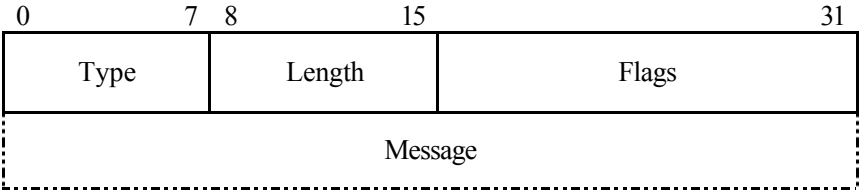


Figure 4.6 General format of User Information option

The fields in the general format of User Information option are:

- ***Type*** is a one byte field specifying the type of the option, like in other options. The value of this field depends on where the User Information option is positioned and what kind of option is this User Information option. For example, if the User Information option is defined as a type of IPv4 Options, the value of this field should be assigned by IANA [IANA1]. The Type of User Information option must not conflict with the now existing types.
- ***Length*** is an 8 bits field that indicates the length of this User Information option in bytes.
- ***Flags*** is a 16 bits field indicating the type of information in the following ***Message*** field, and how this information is organized in the ***Message*** field. Detailed definitions about this field are discussed in subsequent sections. The following types of flags are defined in this dissertation: User Information, Query User Information, User Information Acknowledgement.
- ***Message*** is a variable field that contains user information and control messages. Which kind of message is integrated in this field depends on the ***Flags*** field. A detailed description about this field is explained in subsequent sections.

4.2.1 User Information message

The User Information message is used to carry user information of corresponding IP packets. Figure 4.7 illustrates the format of the User Information message.

The ***Type*** and ***Length*** field in this format are the same as described in 4.2, the meanings of other fields are:

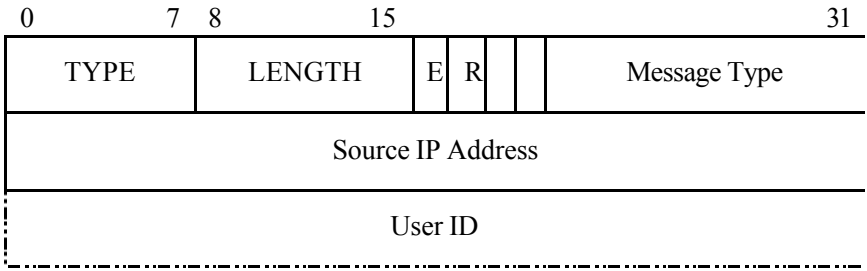


Figure 4.7 Format of the User Information message

- ***E*** is a one bit field indicating whether an encryption mechanism is applied to the User ID field. E=0 means that the user information stored in this option is not encrypted, whereas E=1 means that the user information stored in this option is encrypted. The encryption mechanism is explained in 4.4.
- ***R*** is a one bit field indicating whether this IP packet tagged with User Information option is a redirected inbound IP packet. R=0 means that this is not a redirected inbound IP packet, and the Source IP Address field does not exist. This type of user information is inserted into the outbound IP packets. R=1 means that this is a redirected inbound IP packet, and the Source IP Address field exists.
- ***Message Type*** is a 12 bits field indicating the type of message contained in the User Information option. For the User Information message, the Message Type is 1.
- ***Source IP Address*** is a 4 bytes field in IPv4 packets or 16 bytes field in IPv6 packet indicating the IP address of the sender. This field exists only in the redirected inbound IP packets. In this case, the ***R*** bit in the ***Flags*** field is set to 1. In other cases, the Source IP Address does not exist in the User Information option and the ***R*** bit in ***Flags*** field is set to 0.

  The redirected inbound IP packet, as explained in section 4.1, is generated like this: when an inbound IP packet arrives in the Agent and it cannot be identified with its user by the meter, the Agent finds out its user and inserts user information into the inbound IP packet to build a new redirected IP packet, then this new IP packet is redirected to the meter to inform it about the user of this IP packet.

  When the new redirected IP packet is built, the destination IP address of the inbound packet must be changed to the IP address

of the meter, because the meter is the receiver of the redirected IP packet. And the source IP address of this inbound IP packet is changed to the IP address of the measured host. After the modification of the source and destination IP addresses, the original source IP address of the inbound IP packet cannot be found from the new redirected IP packet. Considering that the original source IP address is necessary for metering, the Source IP Address field is defined in the User Information option. With that, the meter can recover the original inbound IP packet from the redirected IP packet.

- ***User ID*** is a variable length field which contains User ID information of this IP packet. When the ***E*** flag is set to 0, the User ID is not encrypted. In this case, the length of User ID field can be 32 bits. And when the ***E*** flag is set to 1, the User ID is encrypted. If the User ID is encrypted, the length of this field depends on the encryption mechanism.

  In this User Information option format, the User ID field can be either encrypted or not. Since encryption is a complex and CPU burden operation, this will cause performance decline in both Agent and meter by encryption and decryption. Without encryption, the user identification process can be speeded and less effect on performance will result in the Agent and the meter. Without encryption, this mechanism can only be applied in safe environments in which attacks are rare, or accounting information is collected for research purposes such as trend analysis, decision support, etc., which may not attract attacks. However, if the accounting information will be collected for the purpose of charging, billing, access controlling, etc., attacks must be taken into consideration. Therefore, a security mechanism must be integrated into the in-band scheme. The encryption of User ID field is designed for this purpose. The security mechanism applied in the in-band scheme is discussed in detail in 4.4.

4.2.2 Query User Information message

The Query User Information message is used by the meter to ask the Agent to identify the user of the IP packet which carries this message. The Query User Information message is inserted in the IP packet by the meter and is forwarded to the Agent. Figure 4.8 illustrate the format of the Query User Information message.

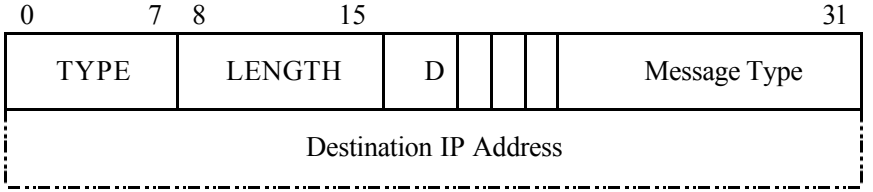


Figure 4.8 Format of the Query User Information message

The ***Type*** and ***Length*** fields in this format are the same as described in 4.2, the meanings of other fields are:

- ***D*** is a one bit field indicating whether this IP packet tagged with User Information option is an inbound IP packet or an outbound IP packet. D=0 means that this is an inbound IP packet, and the Destination IP Address field does not exist. D=1 means that this is an outbound IP packet, and the Destination IP Address field exists.
- ***Message Type*** is a 12 bits field indicating the type of message contained in the User Information option. For the Query User Information message, the Message Type is 2.
- ***Destination IP Address*** is a 4 bytes field in IPv4 packets or a 16 bytes field in IPv6 packets indicating the IP address of the destination. This field exists only in the redirected outbound IP packet which cannot be identified with the corresponding user by the meter. In this case, the ***D*** bit in the ***Flags*** field is set to 1. In other cases, the Destination IP Address does not exist in the Query User Information option and the ***D*** bit in the ***Flags*** field is set to 0.

4.2.3 User Information Acknowledgement message

The User Information Acknowledgement message is used by the meter to acknowledge to the Agent that it has received the User Information of a flow. This message is inserted into the inbound IP packets by the meter. When an Agent receives this message, it verifies if the flow information and the user information in this IP packet match an entry in the DUTRT. If the verification succeeds, the Agent will not insert any user information into the successive IP packet of the same flow. The format of the User Information Acknowledgement message is illustrated in Figure 4.9.

The ***Type*** and ***Length*** fields in this format are the same as described in 4.2, the meanings of the other fields are:

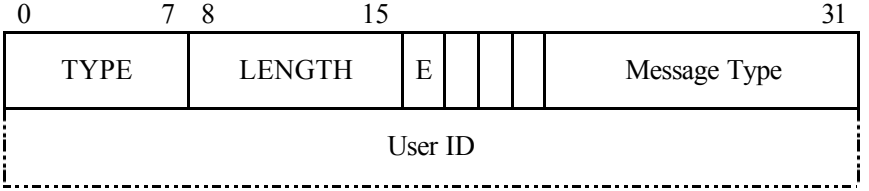


Figure 4.9 Format of the User Information Acknowledgement message

- ***E*** is a one bit field indicating whether an encryption mechanism is applied to the User ID field. E=0 means that the user information stored in this option is not encrypted, whereas E=1 means that the user information stored in this option is encrypted.
- ***Message Type*** is a 12 bits field indicating the type of message contained in the User Information option. For the User Information Acknowledgement message, the Message Type is 3.
- ***User ID*** field is a variable length field which contains User ID information of this IP packet. When the ***E*** flag is set to 0, the User ID is not encrypted. In this case, the length of the User ID field can be 32 bits. And when the ***E*** flag is set to 1, the User ID is encrypted. If the User ID is encrypted, the length of this field depends on the encryption mechanism.

4.3 User Information option location

This section discusses where and how the User Information option can be integrated into the IP packet. Considering the different formats and characteristics of IPv4 and IPv6 protocols, locating the User Information option in IPv4 packets and IPv6 packets are discussed separately.

4.3.1 Integrating User Information option in IPv4 packet

An IPv4 packet contains an IPv4 header and a payload. The IPv4 header consists of a fix 20 bytes length part and a variable length optional part. Figure 4.10 depicts the IPv4 packet format.

By checking the IPv4 packet format, we can find:

- The fix part of the IPv4 header is not extendable for inserting the User Information option;
- The Options field is extendable and may be chosen for carrying the User Information option;
- The User Information option may also be inserted into the position between IPv4 Options (if the Options field does not exist, then IPv4 header) and upper layer PDU.

- The payload with the upper layer PDU is not suitable for carrying the User Information option. If the transport layer datagram is utilized to carry user information, the TCP, UDP protocols must be extended correspondingly. However, the UDP header is very simple and not extendable. Another disadvantage is that inserting user information in the transport layer datagram will increase the overhead. Therefore, the transport layer datagram is not suitable for carrying user information.

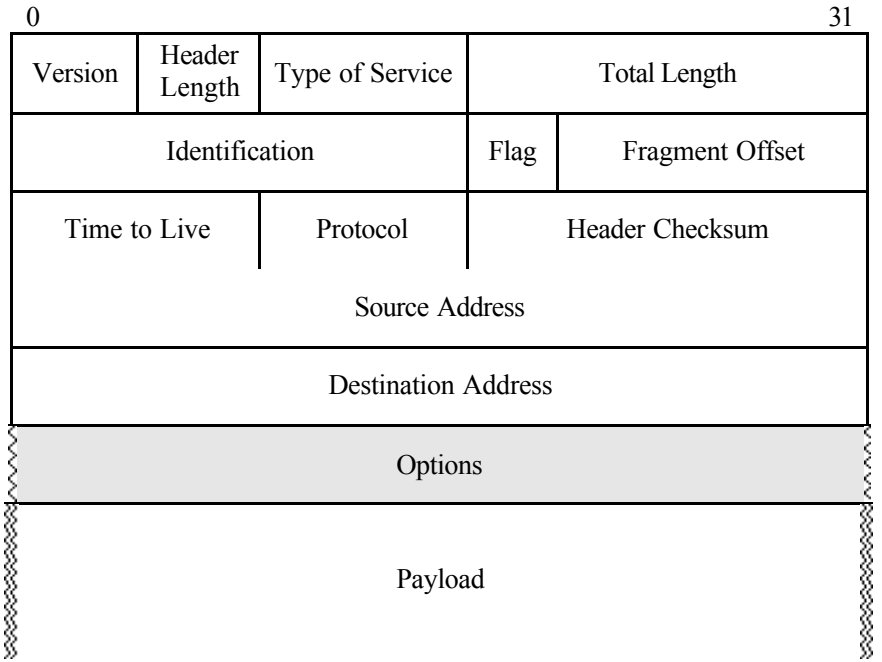


Figure 4.10 IPv4 packet format

Two possible User Information locations in IPv4 packets, i.e. in the IPv4 Options field or between IPv4 Options and upper layer PDU, are discussed below.

4.3.1.1 Positioning User Information option in IPv4 Options field

The IPv4 Options field was designed for possible reinforcement to the original design, or experimenting new ideas. Originally, five IPv4 options were defined accompanied with the publication of IPv4 standard [RFC791]. The five options are security, record route, strict source routing, loose source routing, and timestamp. The Options field can be chosen to be

extended for storing the User Information option to identify the user of the corresponding IP packet.

The positioning of the User Information option in an IPv4 packet's Options field is illustrated in Figure 4.11.

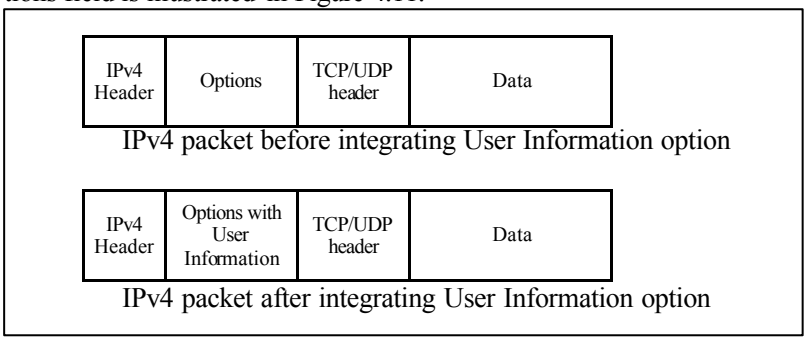


Figure 4.11 integrating User Information option in Options field of IPv4 packet

When the User Information option is integrated into an IPv4 Options field, the type value in the Type field should not conflict with the now existing type values.

Since the IPv4 options are required to be processed by all intermediate nodes, meters can be placed in routers in the key position of networks, as in the traditional IP accounting system, to intercept IP packets tagged with user information and extract user and traffic information.

In order to be able to handle this User Information option, routers with meter functions should be configured to support processing this User Information option. If a router without meter functions or having no interest in this option receives an IP packet tagged with user information, it should ignore this field and continue processing the IP packet according to the "Requirements for IP Version 4 Routers" standard [RFC1812].

Another issue that should be taken into consideration is that the User Information option integrated in the IPv4 Options field must take part in the calculation of the header checksum. If the User Information option is inserted into an IPv4 packet after the IPv4 packet construction, or accurately, after the header checksum is calculated, then the header checksum does not include the User Information option and therefore it must be recalculated. If the User Information option is inserted into an IPv4 packet before the IPv4 header checksum calculation, there is no need to care about the header checksum calculation. Whether the IPv4 header checksum should be recalculated or not depends on the Accounting Agent implementation. More considerations about Agent implementation are discussed in 4.7.1.

4.3.1.2 Positioning User Information option between IPv4 header and upper layer PDU

The IPv4 Options field is designed to be a part of IPv4 header and its size is limited to 40 bytes. Due to the limited space for IPv4 Options and its inflexibility, IPv4 options are rarely used [Hage02]. Some new IPv4 options have been suggested after the publication of the IPv4 protocol [IANA1], but none of them is applied as a standard. IPv4 options are not supported by all hosts and routers.

Considering the limited space in the IPv4 Options field, integrating the User Information option into this field may make things worse. For example, if a Loose Source Router option is inserted into an IP packet with 9 IP addresses, the space occupied by this option reaches 39 bytes, and only one byte is left for the User Information option, which makes it impossible to integrate the User Information option into the IPv4 Options field.

Positioning the User Information option after the IPv4 header and the Options field but before the upper layer PDU, though, will not be limited by space. When the User Information option is positioned in this place, no conflict with IPv4 Options will appear. Figure 4.12 depicts positioning the User Information option between IPv4 the header and the upper layer PDU.

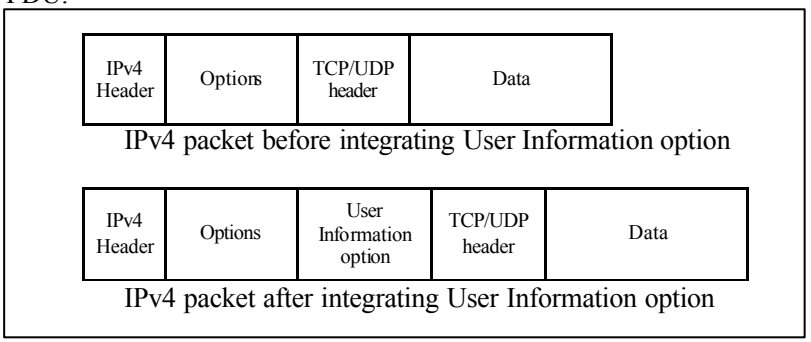


Figure 4.12 positioning User Information between IPv4 header and upper layer header

If the User Information option is positioned in this location, the Type field in the User Information option is used to identify the type of the next payload after the User Information option. The value of this Type field is chosen from the set of protocol numbers defined in [IANA2]. For example, if a TCP header follows the User Information option, the Type field will contain the value 6 which indicates the protocol number of TCP protocol. When the User Information option is positioned after the IP header, the Protocol field of the IPv4 header will contain a value identifying the type

of the User Information option. Therefore, a protocol number must also be assigned to the User Information option. In order to avoid the protocol number assigned to the User Information option conflicting with an existing protocol number, the new protocol number for the User Information option must be assigned by IANA.

Since only the IPv4 header and options will be processed by all intermediate nodes, a User Information option positioned between the IPv4 header and the upper layer PDU will not be checked by any intermediate nodes. In order to trigger meters to check the User Information option for outbound IP packets, every measured host with the Agent must register to a meter before the user based IP traffic accounting mechanism with the in-band scheme is applied. After a measured host is registered to a meter, when the IP packets related to this measured host pass through the meter, the User Information options in the IP packets will be checked by the meter.

Although the User Information option positioned after the IPv4 header and Options will not be processed by intermediate routers, when the IP packets arrive in destinations, the User Information option may cause trouble. This is because the User Information option cannot be recognized by receivers at the destinations. In order to avoid the confusion, the meter must remove the User Information options from the IP packets before they are forwarded to their destinations. This can limit the extension to the IP protocol only between the Agent and the meter. Through that, the modification to the IP packets is transparent to both the sender and the receiver. This will also result in less overhead to the network.

4.3.2 Integrating User Information option in IPv6 packets

IPv4 has been applied as Internet standard for a long time. With the exponential growth of the Internet, IPv4 addresses are becoming scarce. 32 bits IPv4 address cannot provide enough IP address space for the increasing requirements. This is the main reason why IPv6 was proposed. With IPv6 protocol, not only a huge IP address space can be supplied but also the Internet performance can be improved, and new features will be supplemented.

As IPv4 already exists for a long time, many IPv4 products have been developed and deployed. Extending the IPv4 Options field for user based IP traffic accounting may cause modifications to these existing implementations and products. Compared with IPv4, IPv6 is a new Internet standard that is still under development. Moreover, it is also not widely applied. It is expected that there will be less difficulties when the user based IP traffic accounting issues are considered in the IPv6 standard.

By analyzing the IPv6 protocol, we find that the IPv6 extension header can be utilized as the suitable place for integrating the User Information option.

The IPv6 protocol provides extension headers instead of the Options field in IPv4 header. This can improve the efficiency of forwarding IP packets by the routers. With the extension headers, IPv6 can also be extensible beyond a limited Options field. The size of the IPv6 extension header is not limited to 40 bytes like the IPv4 Options field.

Some extension headers have already been defined in IPv6. These extension headers, except the Hop by Hop Options header, are designed to be processed only by destination nodes [RFC2460] due to the consideration of the IP packet forwarding efficiency. Considering this characteristic of the IPv6 extension headers, if User Information option is integrated in an IPv6 packet as an extension header, it might be neglected by the router with meter function, which is located between source and destination endpoints. Due to the fact that the router is only an intermediate node along the path of the packet, the router will ignore all extension headers except the Hop by Hop Options header, the Routing Option header and the Destination Options header.

The ***Hop by Hop Option header*** can be utilized for carrying the User Information option. Since this type of extension header in an IP packet must be processed by all nodes along this IP packet's path to its destination, the meter located in the path can be activated to extract user information from the Hop by Hop Option header. However, considering the fact that user information is required to be processed only by interested routers, i.e. routers with meter functions, the Hop by Hop Option header is not a good choice for integrating the User Information option, because it will cause every node in the path of an IP packet from source to destination to process this extension header. Even though most of the intermediate routers have no interest in the user information in this IPv6 packet, they have to process the Hop by Hop Option header. Therefore, integrating the User Information option into the Hop by Hop Option will increase the processing overhead to the intermediate routers. Moreover, it does not conform to the design principle of the IPv6 extension header.

The ***Destination Options header*** was designed to deliver parameters for destinations. The destinations can be either the final destination of an IP packet or intemediate destinations specified in the Routing Option header. The Destination Options header is identified by the value of 60 in the previous header's Next Header field. A typical application of the Destination Options header is the Mobile IPv6 [RFC3775].

The ***Routing Option header*** contains a list of intermediate nodes that must be visited on the packet's path to its destination. [RFC2460] defines

only the Type Zero Routing Option header which corresponds to the Loose Source Routing option in IPv4. This header is identified by the value of 43 in the previous header's Next Header field.

Whether a Destination Options header is required or not to be processed by intermediate destinations is decided by the existence of the Routing Option header on the one hand, and also by the appearance order of the Destination Options header and the Routing Options header in the IPv6 packet on the other hand.

If a Routing Option header is presented after the Destination Options header, the Destination Options header should be processed not only by the final destination but also by the intermediate destinations specified in the Routing Options header. Otherwise, if no Routing Option header exists or a Destination Options header occurs after the Routing Option header, the Destination Options header will be processed only by the final destination, and all intermediate nodes will ignore this header.

The analysis on the characteristics of the Routing Option header and the Destination Options header shows that, the Routing Options header can be used to designate the meters as the intermediate nodes in the Routing Options header, and the Destination Options header is suitable for carrying the User Information option which can be extracted by meters specified in the Routing Options header.

In order to integrate the User Information option into IPv6 packets, the User Information option should be designed as a type of Destination Options. The Type field of the User Information option should be assigned a unique value for his User Information option, and the IP addresses defined in different messages of User Information option must all be 128 bits IPv6 addresses.

According to the IPv6 specification, winthin the Option Type field of any option in Destination Options header, the two highest-order bits specify how the option should be handled when the option type cannot be recognized by a processing node. Actions should be taken according to different values in the two highest-order bits:

- 00 - Skip the option and continue processing the header
- 01 - Silently discard the packet
- 10 - Discard the packet and send an ICMPv6 Parameter Problem message to the packet's source address no matter the Destination Address field in the IPv6 header is a unicast or multicast address
- 11 - Discard the packet and send an ICMPv6 Parameter Problem message to the packet's source address only if the Destination Address field in the IPv6 header is not a multicast address

The third-highest-order bit of the Option Type specifies whether the option data may be changed (= 1) or not (= 0) along the way to the packet's final destination.

For the User Information option, when an IPv6 node cannot recognize this option, it must skip it and continue to process the rest part of the header. And this User Information option may also be changed along the way to the packet's final destination. Consequently, the three highest-order bits of the Type field in the User Information option must be set to 001. The other five bits of the Type field of the User Information option should be assigned a value that does not conflict with the other existing Destination Options types.

As mentioned above, through the cooperation between the Routing Option header and the Destination Options header, an IPv6 packet with a Destination Options header can traverse the pre-arranged nodes which can handle the information in the Destination Options header. Hence the User Information option in the Destination Options header should cooperate with a Routing Option header for the purpose of delivering the user information to the designated meters.

For an inbound IP packet, the User Information option may be inserted into the Destination Options header by the meter before this IP packet arrives in the measured host. In this case, the final destination is the measured host. Therefore no Routing Options header is required. When a Routing Options header already exists in this IP packet, the User Information option must be inserted after the Routing Options header. Otherwise the nodes specifed in the Routing Options header will try to process the User Information option, which is unnecessary for the User Information option. Figure 4.13 illustrates the position of the User Information option in an inbound IPv6 packet.

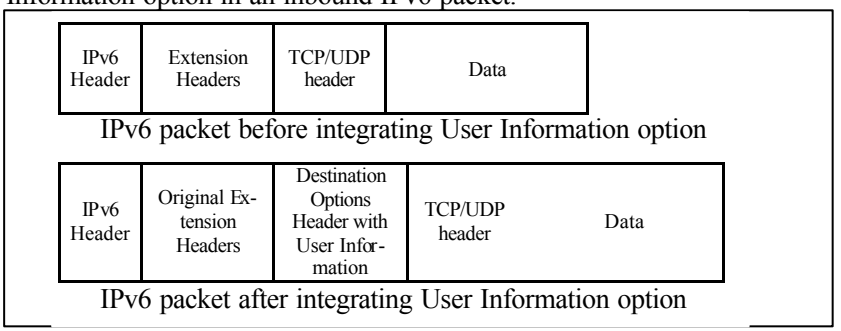


Figure 4.13 integrating User Information in inbound IPv6 packet

For an outbound IP packet, User Information option may be inserted into the Destination Options header by the Agent before this IP packet is

sent to its destination. If this IP packet is a redirected IP packet, its final destination must be the meter. In this case, the Routing Options header is not necessary, since the User Information option is required to be processed by the final destination node, i.e. the meter. But if this IP packet is a normal outbound IP packet, the meter should be an intermediate node along the way to this packet's final destination. In this case, a Routing Options header is required to specify the meter's IP address in it. With that, the routers with meter function will be triggered to process the user information in the Destination Options header. Figure 4.14 illustrates the position of User Information option in an outbound IPv6 packet.

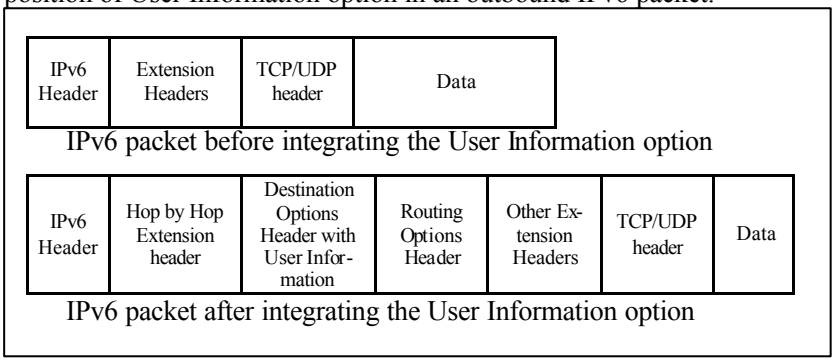


Figure 4.14 integrating User Information in outbound IPv6 packet

Below we use an example to illustrate how the User Information option is integrated into an IPv6 packet as a type of Destination Options and how user information in an IPv6 packet is collected by a meter. In this example an IPv6 packet is sent by an application in a measured host with IP address S to the destination D. In order to perform user based IP traffic accounting, a meter with IP address M is located between the source S and the destination D. In order to make things simple, it is assumed that no other intermediate node exists along the IP packet's path to destination. If the User Information option is designed as a type of Destination Options, the process of user based IP traffic accounting is:

1. At first, the measured host is equipped with the accounting Agent which can intercept inbound and outbound IPv6 packets.
2. When an IPv6 packet sent by an application passes through the Agent, the Agent retrieves the corresponding user information of the IPv6 packet from the system. Then the Agent builds a User Information option. The Message Type field in the User Information option is set to User Information message type. The E bit is set to 0 indicating no encryption. The R bit is set to 0 indicating this is not a redirected IP packet and no Source IP address field

exists in the User Information option. And the user information is placed in the User ID field. After that, the User Information option is encapsulated into a Destination Options header. The length of the Destination Options header must be updated accordingly.

3. In this example, there is no other intermediate node except the meter in the path of this IP packet from source to destination. In order to guarantee that this IP packet passes through the meter, a Routing Option header must be created. The Agent adds the IP address of the meter into a Routing Option header. The Routing Type field should be set to 0 indicating loose source routing. If there are other intermediate nodes defined in the route list, and if the meter is the first intermediate node that this IPv6 packet must access, then the IP address of the meter is placed in the destination IP address field of the IPv6 packet. Otherwise, the IP address of the meter should be inserted into the route list in a suitable place. And the Segments Left field is correspondingly incremented by 1. In this example, the destination IP address field of the IPv6 packet is set to the IP address of the meter and the final destination IP address is moved into the route list in the Routing Option header. Then the Segments Left field is set to 1 indicating that there is one intermediate node left to be accessed.
4. According to the IPv6 extension header place order rule [RFC2460], the Routing Option header must be placed after the Destination Options header. Through that, the meter as an intermediate destination node will be activated to process the Destination Options header.
5. The Agent forwards the IPv6 packet with the User Information option in the Destination Options extension header to its destination.
6. When the IPv6 packet with User Information option reaches the meter M, which is an intermediate destination of the IPv6 packet, the meter extracts user information from the User Information option in the Destination Options header as well as traffic information, e.g. source IP, packet length, etc. from the IPv6 header. Traffic information combined with the corresponding user information can be used by the meter to generate user based RDRs.
7. The meter finds out the next destination from the route list in the Routing Option header according to the Segments Left pointer. Then it places the final destination IP address D into the destination IP address field of the IPv6 packet. At the same time, the current intermediate destination IP address, i.e. the meter's IP

address M, is placed into the position of the final destination D in the route list. At last, the Segments Left field is decremented with 1.

In order to keep users' privacy, or avoid the User Information option to be processed by the final destination, the User Information option should be deleted from the IPv6 packet after being processed by the meter.

8. After processing the User Information option, the meter forwards the IPv6 packet to its final destination D.

Table 4.2 records the changes in the IPv6 packet according to the above described process. In order to make things simple, it is assumed that no other extension header exists in this example.

	IP Header	Destination Options Header	Routing Options Header
From sender instance to Agent	Source IP = S Destination IP = D		
From Agent to meter	Source IP = S Destination IP = M	E = 0 R = 0 Message Type = User Information message User ID = Sender	Segment Left 1 Address (1) = D
From meter to destination	Source IP = S Destination IP = D		Segment Left 0 Address (1) = M

Table 4.2 Changes in an IPv6 packet during the user based IP traffic accounting process

4.4 Security mechanism of in-band scheme overview

Accounting systems are always the targets of attackers. One of the most important reasons is that accounting information is usually related with charging and billing, in other words it always concerns money. In order to prevent this information from attacks such as cheating, modification, forging or even erasing, security measures must be applied when accounting information is processed.

With the in-band scheme, the user information in IP packets may be under attacks such as modification, deletion, spoof and masquerade. For example, users who do not want to pay for their network resource consumption may try to replace their user information in the IP packets with other people's user information or even delete it. Therefore, keeping confidentiality and integrity of the user information to prevent from these attacks is critical for the in-band scheme. In the in-band scheme, a security mechanism is introduced to guarantee that the user information in IP packets is transferred securely.

The security mechanism of the in-band scheme consists of two phases:

1. Negotiation between Accounting Agent and meter to prepare for secure user information transmission. In this phase, the Agent and the meter authenticate each other and exchange security parameters required for the subsequent user information transmission.

2. User information encryption and decryption. In this phase, the user information is encrypted and decrypted according to the security mechanism negotiated in the first phase. With encryption and decryption mechanisms, the user information can be transferred securely to achieve confidentiality. The integrity validation mechanism in this phase can also prevent from spoof and masquerade attacks with the help of encryption.

4.4.1 Negotiation between Accounting Agent and meter for secure user information transmission

Before an Accounting Agent is configured to transfer the encrypted user information, at first it must contact with the meter to authenticate each other and then negotiate parameters about secure user information transmission. The negotiation process for secure communication in in-band scheme simulates the SSL protocol handshake [FrKK96]. Figure 4.15 depicts the message exchange process between Agent and meter for negotiating secure information transmission.

Below the secure communication handshake negotiation process is explained in detail:

1) The Accounting Agent sends an Agent Hello message to the meter. Before the Agent sends this message, the meter should be listening on the predefined port waiting for incoming user based IP traffic accounting requests from Accounting Agents. This message includes a list of supported encryption algorithms and corresponding key sizes, a random number, and a session ID. The encryption algorithms in the list relate to this handshake process. The cryptographic algorithm for user information encryption will be decided in step 7). The random number will be used to seed the cryptographic calculations. The session ID identifies this handshake of secure communication. It can be reused by the same Agent to facilitate speeding up the future handshake process with this meter. In addition, it can also be used to speed up future shared secret update.

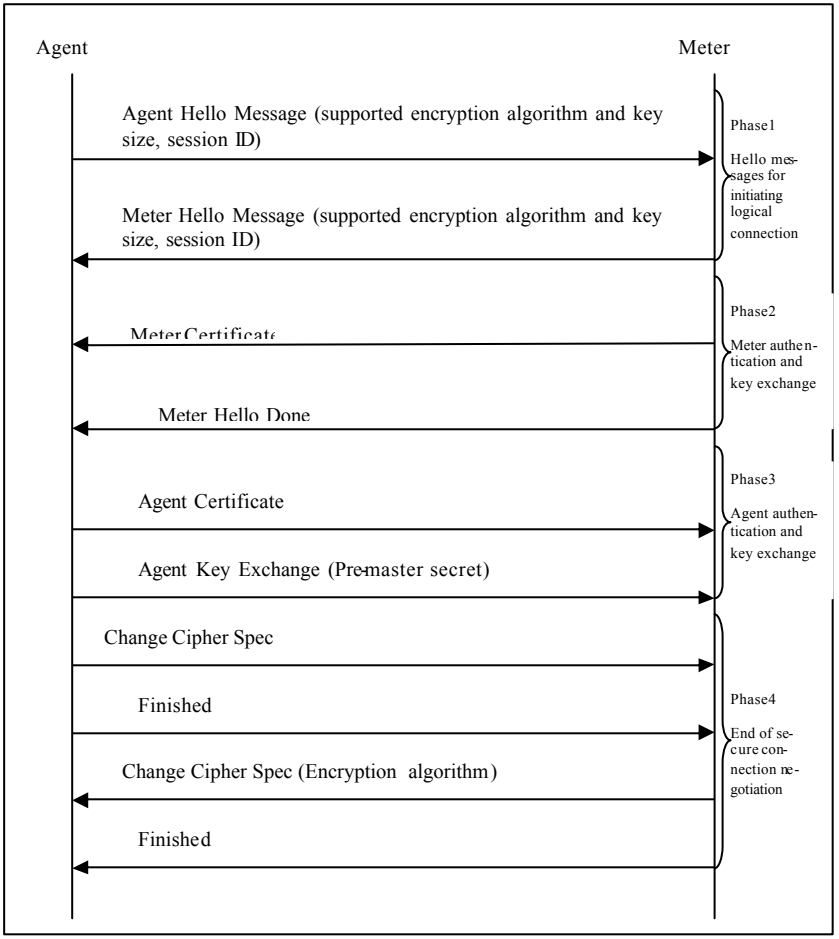


Figure 4.15 Negotiation for secure transmission of user information between Agent and meter

2) When the meter receives the Agent Hello message, it feeds back a Meter Hello message to the Agent. In this Meter Hello message a random number, supported cipher suite and the session ID are included. The new random number, along with the random number that the Agent creates, provides the seed for critical cryptographic calculations. The meter chooses a cipher suite, which it can support from the cipher algorithms list provided by the Agent. If no cipher suite in the list provided by Agent can be supported by the meter, the handshake negotiation fails. If the Agent and the meter had a preexisting session with the same session ID in the Agent

Hello message, the meter returns the Agent's session ID value in the Meter Hello message. If not, the meter responds with a different, meter-generated random number that indicates a new session.

3) Then the certificate of the meter is sent to the Agent. This certificate is used to authenticate the meter's identity. It includes the public key of the meter. After receiving the meter's certificate, the Agent verifies the meter's certificate through validating the signature, checking the validity period, making sure the certificate was signed by one of the CAs the Agent trusts.
4) The meter sends a Meter Hello Done message to the Agent to inform the Agent to start the next phase.
5) After receiving the Meter Hello Done message, the Agent sends its certificate to the meter. This certificate includes the public key of the Agent. The meter also needs to verify the certificate of the Agent.
6) Then the Agent generates a pre-master secret value and encrypts it with the public key from the meter's certificate. This encrypted pre-master secret will be encapsulated into an Agent Key Exchange message which will then be sent to the meter. Both Agent and meter convert the pre-master secret into the master secret by computing a series of hashes using the pre-master secret, the random number in Agent Hello message and the random number in the Meter Hello message. The generated master secret is then used to derive keys for encrypting user information.
7) The Agent sends a Change Cipher Spec message to the meter to inform it to activate just negotiated read cipher suite. And the write cipher suite of the Agent is also activated. In this step the encryption algorithm of the Agent and decryption algorithm of the meter are settled.
8) The Agent sends a Finish message to the meter. The Finished messages allow both systems to verify that the negotiation has been successful and that security has not been compromised.
9) The meter sends also Change Cipher Spec message to the Agent to inform it to activate its just negotiated read cipher suite. And the write cipher suite of the meter is also activated. In this step the encryption algorithm of meter and decryption algorithm of the Agent are settled.
10) At last, the meter also sends a Finish message to the Agent.

Through the negotiation process, the Accounting Agent and the meter are authenticated with each other. The shared session key for encrypting and decrypting user information in IP packets is generated by the Accounting Agent and the meter. The cryptographic algorithm is

also agreed upon by both parties. After that, the User Information option can be transferred between Accounting Agent and meter according to the negotiated secure communication mechanisms.

4.4.2 Encrypting user information

The encryption algorithm and the key applied for encrypting user information are determined by means of the secure communication handshake negotiation. In the in-band scheme, a symmetric encryption algorithm [MeOV96] is chosen for encrypting user information, since symmetric encryption produces less computational overhead than asymmetric encryption [Thom00].

The user information is usually presented in string form or ID number form. For example, user information about Alice may have the form of a string like " Alice" or ID number form like "12345". Formally, let ***u*** stand for user information without encryption. ***E*** is the encryption algorithm, ***D*** is the decryption algorithm, ***c*** is the ciphertext of ***u*** after encryption, ***and k*** is the session key. Let us consider the following situations:

1) If there is no security mechanism applied to the user information, all IP packets related to the same user contain the same content in the User Information option. That means, all the IP packets related to the same user will include the same ***u*** in their User Information option which is not encrypted. It is easy to be attacked by replacing this ***u*** with another user's information ***u´.*** And the privacy of this user can also be simply peeped from the unencrypted user information.
2) Now let us encrypt the user information simply with the session key generated in the secure communication negotiation, i.e. ***c=E(u, k)***. The ciphertext ***c*** is then inserted into the User Information option of user ***u***'s IP packets. The meter can decrypt the ciphertext with the session key to recover the user information: ***u=D(c, k)***.

   It seems that user information could be protected against attacks with this simple encryption mechanism. However, this simple encryption is almost as insecure as without encryption. It is impossible to obtain user information such as user name or user ID from the encrypted user information in the IP packets. Even so, attackers can classify IP packets according to the ciphertext of the encrypted user information to identify one user's network activities without knowing exactly the content of user information. This is because the encryption result of one user's user information ***u*** is always the same ciphertext ***c***. In this case, privacy protection is as weak as without encryption. Furthermore, the possible spoof and masquerade attacks still can be achieved. Although the user information is not readable due to encryption, an at-

tacker can achieve spoof and masquerade without knowing about ***u***. An attacker can use another user's encrypted user information ***c´*** to replace the ***c*** in the User Information option. The ***c´*** is the encrypted user information of ***u´*** (***c´=E(u´, k)***), which can be obtained from sniffing other user's IP packets. After the meter decrypts ***c´***, user ***u*** related IP packets will be falsely regarded as user ***u´*** related IP packets. One reason for this security leak is due to the fact that the encryption of user information is not variable, so the relationship between the ciphertext and user can be easily guessed. Another reason is that this simple encryption cannot provide an integrity mechanism protecting user information from being modified.

According to the above discussion, simple encryption cannot meet the requirements for security. Therefore, a different security mechanism must be introduced. This security mechanism should meet the following requirements:

- It can prevent user information from being leaked, i.e. providing confidentiality to protect privacy.
- It can protect user information against spoof and masquerade attacks. In other words, it can provide integrity validation mechanism.
- It should be simple and efficient. This security mechanism should not result in too much overhead. The overhead lies in two aspects: one is that the encryption should not cause too big performance decline on the measured host in which the Agent resides; the second is that the encryption result, i.e. the ciphertext, should not take too much space for storing it in the IP packet.

4.4.2.1 Principle of encrypting user information

In the in-band scheme, a symmetric encryption algorithm is employed to encrypt user information combined with a Digest, which is the fingerprint of an IP packet. With an IP packet related Digest, the integrity validation can be achieved. The symmetric encryption algorithm can provide more efficient encryption function than asymmetric algorithms. The principle of user information encryption mechanism is illustrated in Figure 4.16.

The user information encryption mechanism consists of five components:

- User information (***u***) is the User ID of a user in the measured host. The size of user information must be fixed.
- Digest (***d***) is the fingerprint of an IP packet. It may be hash value of the payload of the IP packet, or some selected data from the payload. The length of the Digest combined with user information (***u***) must be compatible with the encryption algorithm.

- Session Key (***k***) is the key used to encrypt ***u*** and ***d***. It is generated during the process of secure communication handshake negotiation.
- The encryption algorithm (***E***) is settled during the negotiation phase between Accounting Agent and meter.
- Ciphertext (***c***) is the encryption result after encrypting ***u*** and ***d*** with Session Key ***k***. The ciphertext is inserted into the User ID field of the User Information option and transferred to the meter which will then decrypt the ciphertext to extract the user identifier and to verify the integrity.

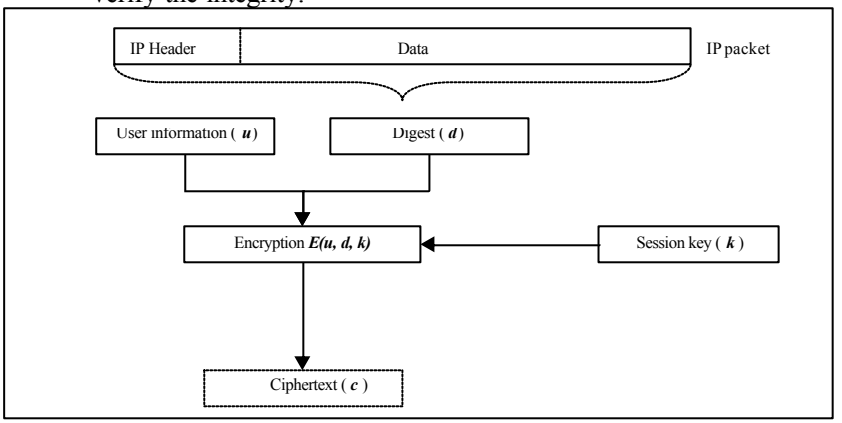


Figure 4.16 user information encryption mechanism

4.4.2.2 Accounting ID

The Agent provides the user information ***u*** of the IP packet. The user information ***u*** may be a user name in string form or a user ID in number form which is used in the measured host. However, user names or user IDs can be easily obtained by attackers, since this information is open in the measured host.

In order to prevent the user identifier from being easily guessed, a special string or random number, called Accounting ID, can be assigned to every user. Therefore, Accounting ID, instead of user name or user ID, will be used as user identifier to be encrypted and inserted into the User Information option. Since the Accounting ID is generated for accounting purpose and only the Agent and the meter know the relationship between Accounting ID and user, it is hard for attackers to guess the Accounting ID.

Hash or random number generation mechanisms can be used to generate Accounting IDs. The special Accounting ID is only used for the account-

ing purpose and is generated by the Accounting Agent when it is started. An Accounting ID – User Mapping table should be transferred from the Accounting Agent to the meter during the secure communication handshake negotiation process. This table will be used by the meter after decrypting the encrypted Accounting ID to identify the corresponding user.

The length of the Accounting ID depends not only on the encryption algorithm but also on the strategy for the Digest ***d*** from the IP packet. For example, the block encryption algorithm DES uses 64 bits (8 bytes) plaintext as an input block, therefore the length of the User Information plus the Digest should take this factor into consideration. It is suggested that the length of the user information part and the Digest part are fixed, which can make the meter separate these two parts from the decrypted text easily and quickly.

4.4.2.3 Digest

As discussed in 4.4.2, simple encryption on user information cannot meet security requirements for the in-band scheme. This can be improved by combining the variable text with user information to generate variable ciphertext for the same user's different IP packets. This variable text is called Digest. The Digest is the fingerprint of an IP packet. Here the fingerprint means that it represents the unique characteristic of an IP packet. The corresponding bytes in an IP packet chosen for generating Digest must be immutable along the path from source to destination. Otherwise, the integrity validation will fail.

Here we propose two possible Digest generation mechanisms:

1. The first Digest generation mechanism is to take advantage of the similar integrity mechanism used in IPSec. At first, the Integrity Check Value (ICV) is calculated on the basis of the whole IP packet except the mutable fields. Then the first several bytes of the ICV are chosen as Digest combined with user information for the purpose of encryption. By decryption, the ICV of the IP packet will be recalculated to verify whether the decrypted Digest matches the recalculated ICV. This mechanism can provide a stronger integrity validation mechanism. However, the overhead of this mechanism is very high due to the ICV calculation and recalculation.
2. The second Digest generation mechanism is to select some bytes from the payload of an IP packet randomly to construct the Digest. The information about which bytes are selected must be a part of the plaintext. In this case, the structure of the plaintext for encryption is like Figure 4.17. In this structure, the "References" field lists all positions of selected bytes for constructing Digest. After decryption, the position

information in the "References" field is used to gather the bytes in the payload of the IP packet to compare with the decrypted Digest. Since this mechanism does not need to calculate Digest, the overhead for generating Digest by this mechanism is lower than the first mechanism. Since the Digest is randomly constructed, even the same payload may have different References. This mechanism can provide enhanced security for protecting the User Information.

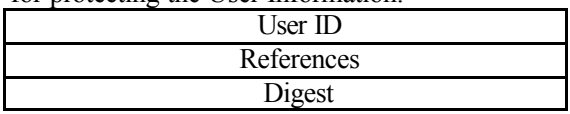


Figure 4.17 Structure of the user information for encryption

Which mechanism for constructing Digest should be applied may be a trade-off between performance and security. The information about the strategy of constructing Digest should also be exchanged between the Accounting Agent and the meter during the secure communication negotiation process. If the Digest pluses user information cannot reach the size requirement for encryption, e.g. 64 bits for DES, the rest part should be padded with padding data.

Digest is a part of the IP packet and it corresponds to a part of the IP packet. The meter side requires this Digest to verify the integrity of the encrypted User Information. The verification is accomplished by comparing decrypted Digest with corresponding bytes in the IP packet. If the ciphertext is changed along the way between measured host and meter, the Digest cannot match its corresponding bytes in the IP packet and the verification will fail. In other case, if the corresponding bytes of Digest are changed during the transmission, despite the fact that the ciphertext contains the User Information and the Digest is not changed, the integrity validation will also fail.

The possible situations of changing IP packets along their way to destinations may be attack related or protocol related.

For attack related modification on the IP packets[8], if the integrity verification on an IP packet fails, this packet can be discarded. If the modification is not on the bytes corresponding to Digest, the integrity verification will succeed. In the later case, it only means that the User Information is not modified during transmission, but that does not mean that the integrity of the whole IP packet is verified. The IP packet will still be forwarded. If the whole IP packet should be protected from attacks, security mechanisms such as IPSec, SSL/TLS, HTTPS, etc. should be applied to different layers of TCP/IP protocol stack. The security mechanism of the in-band scheme cares only about the security of the User Information option.

[8] Here those attacks on User Information option are not included.

Another possible modification on IP packets is protocol related. Examples are, IPSec, NAT, transition between IPv4 and IPv6, VoIP Signalling conversion [SiSc00, RFC3372, ZhHM05], Mobile IP [RFC3344, RFC3775], etc. Figure 4.18 illustrates the possible positions where modification on an IP packet may happen.

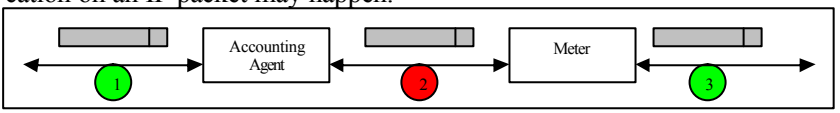


Figure 4.18 Possible positions where IP packets may be modified

The modification made on IP packets in position 1 will not affect the integrity verification of User Information. IP packets can enter Accounting Agent from two directions: one direction is from the kernel of the measured host as outbound IP packets; the other direction is from outside the measured host as inbound IP packets. No matter how the outbound IP packets were modified in upper layer modules, e.g. HTTPS, IPSec, etc., in the kernel before they enter the Accounting Agent, obviously these modifications will not result in the Digest bytes being modified after the encryption of User Information. What should be noticed is that when the Accounting Agent is integrated into the system kernel, it should be placed in the last position of IP packets modification chain for outbound IP packets and also the first position of IP packets modification chain for inbound IP packets. For example, for outbound IP packets IPSec module should process the IP packets before the Agent module. Otherwise, after being processed by Accounting Agent the IP packets will be modified by IPSec module, which may cause the integrity verification of User Information option failure.

The modification made on IP packets between the Accounting Agents and the meter may cause Digest related bytes in IP packets inconsistent with the Digest. For example, integrating IPSec related header into an IP packet will bring change to the payload of the IP packet. Modifications on payload may cause User Information integrity verification failure. In order to avoid this problem, the Agent and the meter should be located in the place before the IP packets modification happens. When necessary, the Agent must be implemented in the last position of the modification chain on IP packets and the meter may need to be located directly next to the multi-user system as the first intermediate node. Through that, no modification will be made on IP packets between Accounting Agent and meter.

The modification made on IP packets in position 3 will not affect the integrity verification mechanism of User Information option, since the IP packets in this position will not be checked by meter.

By encrypting User Information and Digest, the User Information carried in IP packets cannot be peeped by attackers, and the spoof and masquerade attacks can also be prevented. The Digest can help keeping the integrity of the encrypted user information.

4.4.3 Decrypting User Information

When a meter receives an IP packet with encrypted User Information option, it performs the following operations to decrypt it:

1. Session key is used to decrypt the ciphertext to recover the User Information and the Digest, i.e. $(\boldsymbol{u}, \boldsymbol{d}) = \boldsymbol{D}(\boldsymbol{c}, \boldsymbol{k})$.
2. The meter rebuilds the Digest according to the received IP packet. Then it compares the newly built Digest with the decrypted Digest to verify whether they are identical.
3. If they are not identical, this means that either the Digest related bytes in the IP packet or the encrypted User Information option is changed during the transmit of the IP packet. Therefore, this IP packet should be discarded. This verification error information should be recorded into a security log.
4. If they are identical, the meter extracts the user based IP traffic accounting required information such as User Information, source IP, destination IP, bytes, etc. from IP packet to generate a RDR. Then meter forwards this IP packet to its destination.

The above described user information decryption process is depicted in Figure 4.19.

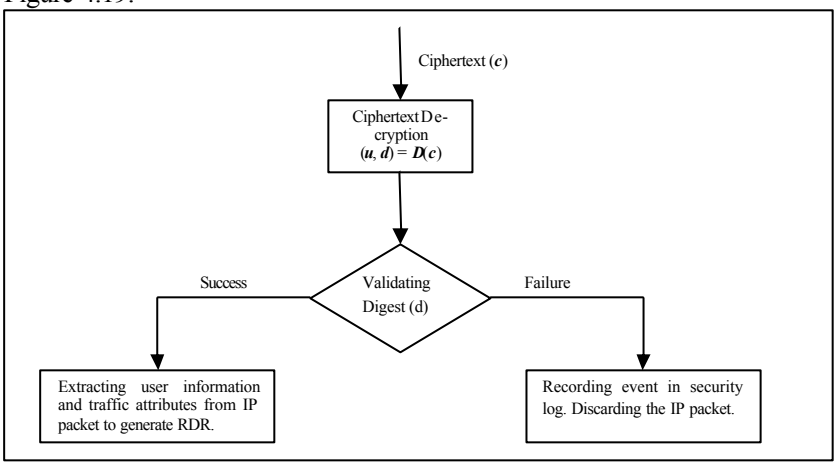


Figure 4.19 User Information decryption process in meter

4.5 Implementation considerations

In order to realize user based IP traffic accounting with the in-band scheme, the Accounting Agent must be implemented in the measured host. The traditional meter should also be extended to support the protocol defined in the in-band scheme. Before user based IP traffic accounting with the in-band scheme can be implemented, there are some issues must be taken into consideration:

- Where should the Accounting Agent be implemented?
- How to handle fragmentation.
- How does the in-band scheme coexist with IPSec?
- Performance issues.
- Dependability issues.

4.5.1 Where should the Accounting Agent be implemented

The Accounting Agent must have the ability of intercepting IP packets, collecting user information and integrating User Information options into IP packets. Therefore, the Agent must be implemented in the position where all IP packets and their corresponding user information can be obtained. According to the four theory positions discussed in 4.5.2, considering the characteristics of user based IP traffic accounting with the in-band scheme, the following two positions are possible for implementing the Agent for the in-band scheme:

- Realizing the Agent in the native IP protocol implementation. This position requires the source code of the native IP protocol implementation to be modified.
- Realizing the Agent below the native IP protocol implementation, between the native IP protocol implementation and network drivers, i.e. "Bump-in-the-stack" (BITS) [RFC2401]. This position does not require source code of the native IP protocol implementation to be accessed.

It may be difficult for the Agent to insert User Information options into IP packets when the Agent is implemented above the native IP protocol implementation, since in this position IP packets are not completely constructed.

4.5.2 Fragmentation

In the in-band scheme of user based IP traffic accounting, two fragmentation related issues should be taken into consideration:

- One issue is that inserting User Information option into an IP packet may cause the size of this IP packet to exceed the MTU of the trans-

mission path. In this case, the IP packet must be fragmented after User Information option is integrated into IP packets.

- Another issue is that, if an IP packet tagged with user information must be fragmented on the way to destination, how should the User Information option be processed during packet fragmentation?

Fragmentation of IP packets may be affected by different factors. It can take place in different positions: in the measured host, in intermediate nodes between the measured host and the meter, in the meter, and in intermediate nodes which locate between the meter and the remote endpoint. Different user identification methods also play important roles in the IP packet fragmentation. IPv4 and IPv6 packets should be treated differently in fragmentation. Inbound IP packets fragmentation and outbound IP packets fragmentation may have different characteristics.

4.5.2.1 Fragmentation in measured host

Because the data link layer normally has a limitation on the size of the frame that can be transmitted, the IP layer must fragment IP packets into suitable sizes according to the MTU of the network interface. When an IP packet is to be sent, the size of this IP packet and the MTU will be compared. Then fragmentation will be made in the IP layer when the size of this IP packet is larger than the MTU.

In IPv6, fragmentation only occurs on the source host sending the packet. The destination host handles reassembly. The Path MTU discovery mechanism [RFC1191] is used by source host to determine the maximum packet size that can be used on the way to destination. If an IP packet's size is larger than the MTU, it will be fragmented in the source host before being sent to the destination. A Fragmentation extension header will be added into every fragmented IPv6 packet. Unlike IPv4, IPv6 packets will not be fragmented along the path to destinations.

If the Accounting Agent function is integrated in the native IP protocol implementation, the User Information option will be built into IP packets when IP packets are constructed. When an IP packet tagged with user information must be fragmented before being sent, the User Information option should be processed according to different user identification methods:

- When the IP packet based user identification method is adopted, the User Information option must be integrated into every fragmented IP packet.
- When flow based user identification method is adopted, if the fragmented IP packet is the first IP packet of an IP traffic flow, the User Information option must be integrated into the first fragmented IP

packet. Otherwise, no User Information option is required to be integrated into the fragmented IP packets.

If the Accounting Agent is realized underneath the native IP protocol implementation, when an IP packet sent by the IP layer comes into the Agent, the fragmentation operation should be made by the Agent. The Agent calculates the size of this IP packet together combining with User Information option and then compares it with the MTU.

- If the size of this IP packet combined with the User Information option does not exceed the MTU, no fragmentation will be made.
- If the size of this IP packet combined with the User Information option exceeds the MTU, and the IP packet is not a fragmented IP packet, then the Agent fragments this IP packet and inserts User Information into the fragmented IP packets according to different user identification methods. In IPv4 environment, the fragmentation flag is set. In IPv6 environment, the Fragmentation extension header is integrated into the fragmented IP packets.
- If the size of this IP packet combining with the User Information option exceeds the MTU, and the IP packet is a piece of a fragmented IP packet, the Agent reassembles all pieces of the original IP packet together. After that, the Agent fragments the reassembled original IP packet again so that the sizes of new fragmented IP packets combined with the User Information option will not exceed the MTU. Finally, the Agent inserts User Information options into new fragmented IP packets according to different user identification methods. A new fragmentation flag or Fragmentation extension header must be integrated into the fragmented IP packets.

4.5.2.2 Fragmentation between measured host and meter

Fragmentation between the measured host and the meter may take place in intermediate nodes such as routers. As mentioned above, this kind of fragmentation can only be applied to IPv4 packets, since IPv6 fragmentation is recommended to be made only in source hosts. The IP packet can be either an inbound or an outbound packet.

When an IP packet must be fragmented by an intermediate router, the issue concerning the in-band scheme is how to process the User Information option in this IP packet. The solution depends not only on the user identification methods but also on the position of User Information options in IP packets.

- For IP packet based user identification, the User Information option is assumed to be attached to every IP packet. Therefore, the same User

Information option should be copied to all fragmented IP packets during the IP packet fragmentation process.

If the User Information option is positioned in the IPv4 Options field as an option, it will be automatically copied to all fragmented IP packets during IP packet fragmentation process.

However, if the User Information option is positioned between IPv4 header and upper layer PDU, usually it will not be copied to all fragmented IP packets. This User Information option will be treated as normal IP packet payload and is probably inserted into the first fragmented IP packet. Other fragmented IP packets will carry the rest part of the payload. In order to handle this problem, the Agent or the meter can aggregate fragmented IP packets together according to the value in the Identification field of IPv4 header and then identifies the user of these fragmented IP packets with the User Information option in the first fragmented IP packet.

- For IP traffic flow based user identification method, the User Information option exists only in the first IP packet of a traffic flow. When the first IP packet of a traffic flow needs to be fragmented by the intermediate router, let's consider two different positions of User Information option:

  If the User Information option is positioned in the IPv4 Options field, it will be copied to all fragmented IP packets. Therefore, the meter and the Agent should have the ability of handling redundant occurrence of the User Information option correctly and efficiently

  If the User Information option is positioned between the IPv4 header and the upper layer PDU, by fragmentation, it will be integrated into the first fragmented IP packet and the other fragmented IP packets will contain no User Information option. In this case, it still conforms to the requirement for IP traffic flow based identification method, i.e. the first IP packet conveys the User Information option. Therefore, no other extra operation should be performed in meter or Agent.
- For hybrid user identification method, all outbound IP packets are attached with User Information option. For inbound IP packets, only those IP packets that are the first IP packets of IP traffic flows will be redirected to meter. The principle discussed in IP packet based user identification is suitable for outbound IP packets fragmentation, whereas the principle discussed in IP traffic flow based user identification can be applied to the redirected IP packets.

4.5.23 Fragmentation in the meter

Fragmentation may take place in the meter when a User Information option must be inserted into an inbound IP packet. This may happen only when the IP traffic flow based user identification method is applied. In IP packet based user identification, User Information options are integrated into outbound IP packets or redirected IP packets only by the Agent.

In the meter, User Information options should be integrated into IP packets in two situations:

- When an inbound IP packet without User Information option comes into the meter and it cannot be identified with the corresponding user by the meter through searching DUTRT, the meter encapsulates a Query User Information message into the IP packet and sends it to the Agent.
- When an Agent continues sending IP packets tagged with user information of the same flow to the meter, the meter may send the User Information Acknowledge message to inform the Agent about its receipt of this User Information option of the flow.

Therefore, when an IP packet should be integrated with User Information option in the above situations, the size of this IP packet with integrated User Information option may be larger than the MTU, and consequently the meter has to fragment the new IP packet.

For an IPv4 packet which is not a fragmented IP packet, the meter fragments it into different pieces and inserts a User Information option into the first fragmented IP packet. Then the meter sends them to the Agent.

For an IPv4 packet which is a fragmented IP packet, the meter should at first aggregate all fragmented IP packets to reconstruct the original non-fragmented IP packet. Then the meter fragments the new IP packet into different pieces and inserts a User Information option into the first fragmented IP packet. After that, the meter sends these fragmented IP packets to the Agent.

For IPv6 packets, since fragmentation is not allowed by intermediate nodes, the following measures can be taken:

- The meter inserts a User Information option into an IP packet and sends it to the Agent. If the new IP packet is larger than MTU, an ICMPv6 Packet Too Big message will be sent back to the source endpoint. Since meter is usually located in the key place where all traffic will pass through, the meter can capture the ICMP message. Figure 4.20 shows the structure of ICMPv6 Packet Too Big message.

  In the Packet Too Big message, the Type field is set to 2 and the Code field is set to 0. The 32-bit MTU field stores the link MTU of the interface over which the IP packet was forwarded. The Data field

contains the leading portion of the discarded packet. The size of this field is variable. As much as possible leading portion of the discarded packet can be contained in the Data field, if the ICMPv6 message is no larger than the Path MTU. Considering that the size of IPv6 header is fixed to 40 bytes, the maximum size of the Data field can be calculated as follows in case the minimum IPv6 MTU is applied and no IPv6 extension header is included:

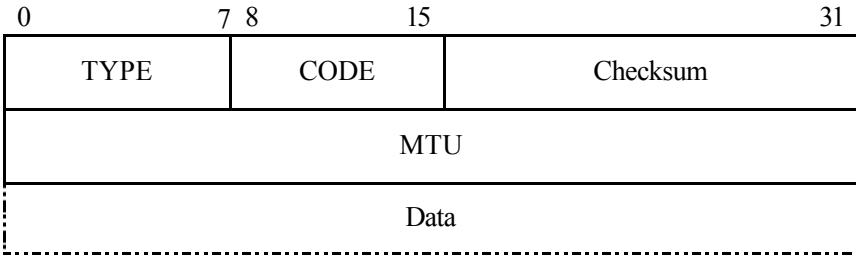


Figure 4.20 structure of ICMPv6 Packet Too Big message [RFC2463]

Data field size = Minimum MTU – IPV6 header size – TYPE size – CODE size – Checksum size – MTU size = 1280 – 40 – 1 – 1 – 2 – 4 = 1232 (bytes)

The User Information option is designed as a type of Destination Extension header which is located in the front part of the IPv6 packet. Considering the size of Data field is at least 1232 bytes, the User Information option will be included in the Data field of the ICMPv6 packet as the leading portion of the discarded packets. When an ICMPv6 Packet Too Big message as an outbound IP packet comes into the meter, the meter checks if the User Information option field exists in the Data field. If so, this means that the Packet Too Big error is caused by the integrated User Information option. In this case, the meter should calculate a new MTU. The new MTU is the value in the MTU field minus the size of the User Information option. Then the meter inserts the new MTU in the MTU field of the ICMPv6 Packet Too Big message and then forwards it to its destination. After the ICMPv6 Packet Too Big message is received, the source host will adjust its sent IP packets' size according to the new MTU.

- Another method is to fragment IPv6 packets in the meter, if necessary. Although fragmentation is not recommended by intermediate nodes in IPv6 considering the forwarding performance of routers,

fragmentation may still be made in intermediate nodes if necessary. Example is the IPv6 to IPv4 translator [RFC2460].

When the size of an IPv6 packet plus the size of a User Information option is larger than the PMTU from the meter to the Agent, the new IP packet tagged with user information must be fragmented. If the IP packet is not a fragmented IP packet, the meter should construct a new Fragmentation extension header for each fragment. If the IP packet is a fragmented IP packet, the meter reassembles all fragments together and inserts User Information into new fragments. The Fragmentation extension header should also be modified according to the new fragmentation.

- Sending back User Information option to Agent with out-of-band mechanism may be also a choice, i.e. the Acknowledge or Query User Information message is sent to Agent in a dedicated communication channel between Agent and meter rather than utilizing IP packets to convey their User Information option.

4.5.2.4 Fragmentation between meter and remote endpoint

If fragmentation is made between the meter and the remote endpoint, it is irrelevant to the in-band scheme.

4.5.3 Coexistence with IPSec

IPSec is a security framework designed to provide cryptographically based security for IPv4 and IPv6 [RFC2401, RFC2402, RFC2406]. Two traffic security protocols are applied in IPSec to achieve IP level security: the Authentication Header (AH) and the Encapsulating Security Payload (ESP). The AH Protocol provides connectionless integrity, data origin authentication, and an anti-replay protection service. However, AH does not provide any confidential services: it does not encrypt the packets it protects, whereas the ESP protocol provides data confidentiality and limited traffic flow confidentiality. Data origin authentication, connectionless integrity and anti-replay service are also provided by the ESP protocol. Since in AH and ESP protocols relevant portions of the IP packet must be taken into consideration in integrity, authentication, encryption, etc. process, how User Information option coexists with IPSec should be carefully analyzed before user based IP traffic accounting in-band scheme is implemented.

When User Information options are integrated into IP traffic for which the IPSec mechanism is applied, two possible issues may arise:

- One issue is that the appearance and transformation of User Information options in IP packets may affect the integrity verification and au-

thentication of IPSec. For example, if a User Information option is inserted into an IP packet after the AH header is integrated into the IP packet, the integrity verification of IPSec may fail in the destination due to the absence of User Information option when the AH header was built.

- Another issue is that the encryption of the ESP protocol may affect the operation of the in-band scheme. For example, if the User Information option in an IP packet is encrypted by ESP, the meter will not be able to extract it from the IP packet due to the end to end encryption characteristic of IPSec.

4.5.3.1 Conforming to IPSec integrity

In order to avoid the in-band scheme influencing the integrity and authentication ability of IPSec, user based IP traffic accounting mechanism should be implemented conforming to IPSec integrity requirements. The principle of in-band scheme coexisting with IPSec is to avoid User Information option being taken into consideration in calculating the ICV (Integrity Check Value).

For IPSec transport mode, the implementation position of the Accounting Agent in a host may be the same as IPSec, i.e. they may all be integrated in the native IP implementation or underneath an existing implementation of the IP protocol stack ("Bump-in-the stack") [RFC2401]. No matter in which position, the Agent must be implemented underneath the IPSec implementation. Agent and meter should remove User Information option from IP packets before they are forwarded to the IPSec endpoints. With that, the operations, such as integration, deletion, etc., on User Information options in IP packets are transparent to IPSec integrity and authentication mechanisms, since the transformation on IP packets for the purpose of user based IP traffic accounting happens only between the two IPSec communication endpoints. Figure 4.21 shows the relative positions between IPSec and in-band scheme implementations.

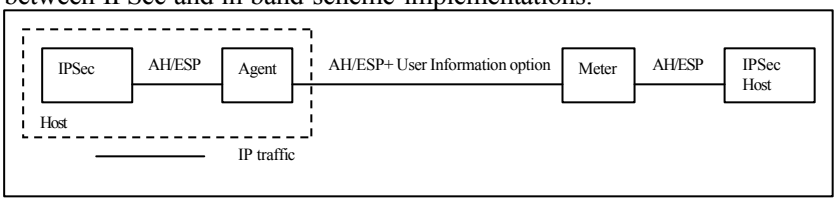


Figure 4.21 Relative positions between IPSec and in-band scheme implementations for transport mode

For IPSec tunnel mode, the meter is not allowed to be placed between two security gateways, i.e. there is no security gateway between the Agent

and the meter. Otherwise, the User Information option may become a part of the inner IP packet for outbound traffic, which makes it difficult for the meter to find it. Integrating User Information option in IP traffic by the meter for inbound traffic may result in the integrity check failing in security gateway.

4.5.3.2 In-band scheme and ESP encryption

The ESP protocol provides traffic flow confidentiality using an encryption mechanism. In transport mode, only the upper layer PDUs of IP packets are encrypted. In tunnel mode, IP packets are encrypted and encapsulated into new IP packets as inner IP packets. The principle of the in-band scheme coexisting with ESP encryption is to avoid the User Information option from being encrypted by ESP implementation.

For ESP transport mode, if the implementation position of the Accounting Agent is underneath IPSec in a host, integration of User Information options in IP packets happens after ESP encryption. Therefore, the User Information option will not be encrypted by ESP. In this case, User Information options are required to be removed from IP packets by the meter or the Agent before they are forwarded to destinations for decryption. If the implementation position of the Accounting Agent is above IPSec, the User Information option inserted between IPv4 header and upper layer PDU will be encrypted. Because of this, the User Information option cannot be encrypted by the meter. Hence, implementing the Agent underneath IPSec is the right choice in the case of ESP encryption.

For ESP tunnel mode, in order to avoid the User Information option from being encrypted by the security gateway, the only choice is to place the meter in the position nearer to the Agent than to the security gateway. With this strategy, IP traffic between Agent and meter will not be disturbed by the IPSec mechanism.

Summarizing the above discussed principles, the location of IPSec and the Agent of the in-band scheme implementation is illustrated in Figure 4.22.

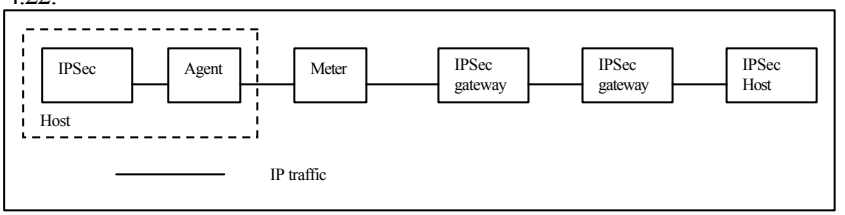


Figure 4.22 Relative positions between IPSec and in-band scheme implementations

The meter location strategy for IPSec tunnel mode can also be applied to other tunnel situations such as IPv6-to-IPv4 tunnels. In all these tunnel cases, original IP packets are encapsulated in new IP packets as inner IP packets. Therefore, the inner IP packets are treated as payload of IP packets by the meter. The User Information option in inner IP packets may usually be neglected by the meter as normal IP packet payload. Otherwise, the meter must check the protocol field in the IP header of every IP packet to find out the User Information option in inner IP packet. This will certainly increase the overhead for user based IP traffic accounting. Hence locating the meter in the place between the Agent and the tunnel server, as showed in Figure 5.24, is a good solution to avoid the problems described above.

4.5.4 Performance issues

The usage of the Accounting Agent imposes computational performance overhead on the measured host. If the meter is implemented in a router, it affects the performance of the router. Possible overhead related to the in-band scheme may include extra CPU burden caused by per-packet handling for the User Information option in the measured host and the meter, memory occupation for the DUTRT, and extra bandwidth consumption for User Information option transmission.

With the in-band scheme every IP packet must be intercepted for validating the corresponding user information, inserting or extracting user information, encrypting or decrypting user information (if necessary). The per-packet computational costs will be manifested by increased latency and, possibly, reduced throughput. Especially the encryption and decryption computation may delay TCP connection establishment. Flow based user identification can improve the performance in reduced handling costs and less bandwidth consumption. However, maintaining DUTRT requires extra memory, whereas searching information in DUTRT also results in extra CPU burden.

The application of the in-band scheme imposes bandwidth consumption costs on transmission due to the increased packet size resulting from the addition of User Information options. The packet based user identification method may consume more extra bandwidth compared with the flow based or the hybrid user identification method.

4.5.5 Dependability considerations

Dependability is an important issue that must be taken into consideration in designing accounting systems. In user based IP traffic accounting systems realized with the in-band scheme, the reliability of the Agent and

the meter is critical for dependability. The dependability related issues in the in-band scheme are: when the Agent or the meter is down, how to detect it, how to handle user based IP traffic accounting in this case; and when the Agent or the meter lives again, how to recover the user based IP traffic accounting process.

After an Agent is started, it should be registered to a meter for the purpose of user based IP traffic accounting with the in-band scheme. From that on, the Agent and the meter should communicate with each other by sending keep-alive messages to monitor the state of each other.

If an Agent detects that the meter is down, it should stop sending and receiving IP packets which must be accounted according to the accounting policy. The Agent should record the detection of meter crash into an error log and should also prompt information to the system administrator about it. The Agent should keep sending the Agent Hello message (see 5.4.1) regularly to register it again to the meter after the detection of meter crash. When the meter is newly started, Agent can try to use the previously negotiated session ID to resume predefined secure setting. If these settings are still available in the meter, it can resume previous secure configuration. Otherwise, a new secure handshake must be made.

If a meter finds that an Agent is down, it should stop forwarding any IP traffic from or to the corresponding measured host on the one hand, and it should also stop recording accounting information related to all users in the corresponding measured host on the other hand. Through that, free network resource usage can be prevented when the accounting system crashes. Moreover, this can also discourage some attackers from trying to attack meter using Denial of Service (DoS) to achieve illegal network resource consumption.

4.9 Summary

In this chapter, the user based IP traffic accounting in-band scheme is discussed in detail. With the in-band scheme, IP packets are utilized for storing and transferring their user information. Through that, a meter can directly extract user information, IP address and other traffic information from IP packets for the purpose of user based IP traffic accounting.

In the in-band scheme, the Agent is a key component in identifying users of IP packets, integrating User Information options in IP packets, and negotiating a secure communication channel with the meter. In order to achieve user based IP traffic accounting, the meter should also be adaptable to the user information in the IP packets. In the in-band scheme, three user identification methods, IP packet based method, IP traffic flow based method and hybrid method, are suggested and analyzed. The user based IP

traffic accounting process and mechanism are explained on the basis of these user identification methods. If the flow based user identification method should be applied, DUTRT in the Agent and the meter should be maintained to be synchronized. These three methods have different advantages and disadvantages. They can be applied in different application environments.

A User Information option is proposed for the purpose of carrying user information and other user based IP traffic accounting related control information. This User Information option may be integrated in different positions of IPv4 and IPv6 packets. In IPv4, the User Information option can be defined as a type of Options to be integrated into the Options field. Considering the limited space in the IPv4 Options field, positioning the User Information option between IPv4 header and upper layer PDU may be a better choice. In IPv6, designing the User Information option as an option in the Destination Options extension header is a suitable choice. In order to trigger the meter to check the User Information option in the Destination Options extension header, the Routing Options header is also required to direct IP packets passing through designated meter.

In order to protect the in-band scheme from potential attacks and to keep privacy, security mechanisms should be applied in the in-band scheme. This chapter suggests building a secure communication mechanism between the Agent and the meter. User Information can be encrypted for transmission. The security mechanism for the in-band scheme is explained comprehensively. With this security mechanism, the in-band scheme can be protected from potential attacks such as spoofing, masquerade, man in the middle, etc. The encryption mechanism can also help to keep privacy through hiding user information.

Figure 4.23 depicts how user based IP traffic accounting is achieved with the in-band scheme:

In Figure 4.23 we can find that, although different users in the multi-user system generate IP traffic with the same IP address (192.168.0.100), the IP traffic can be differentiated by user information in the IP packets. Therefore, the RDRs generated by the meter are identified with the corresponding user information. The correlation module in the Mediation Layer utilizes a TO & User Map Table instead of an IP address & User Map Table, which is used in traditional IP accounting systems, to translate <IP address, User ID> into the corresponding user. With the help of the 2-tuple TO, the correlation module can generate URs of different users in the multi-user system without any ambiguity.

This method can facilitate gathering user traffic relationship information outside the measured hosts by the meter. Another advantage is that no temporary user traffic relationship information needs to be stored in the

measured hosts. Since user information is combined with traffic information, real-time user based IP traffic accounting can be easily achieved.

TO & User Map Table		
IP Address	User ID	User
192.168.0.100	UID1	USER1
192.168.0.100	UID2	USER2
.....	.....	.....
192.168.0.100	UIDn	USERn

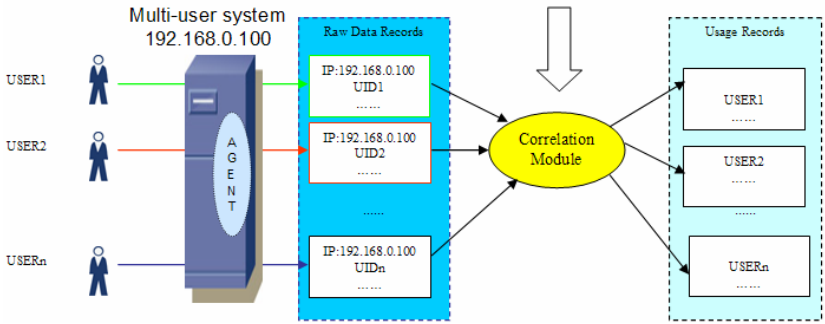


Figure 4.23 User Information processing with in-band scheme

In realizing the in-band scheme, several issues should be taken into consideration: how fragmentation should be handled after integrating User Information option into IP packets, how in-band scheme coexists with the IPSec protocol, how the Agent and the meter affect the system performance and how dependability is achieved in in-band scheme. The discussions in this chapter provide some suggestions about ways of solving these problems.

Chapter 5 Out-of-band scheme

The in-band scheme utilizes the IP header to convey the user information of corresponding IP traffic. Through that, the meter can intercept IP packets and extract both user information and traffic information directly from them. However, this scheme requires the IP protocol to be extended to accommodate the user information. In this chapter, a different user information transmission scheme is introduced. This scheme is called out-of-band scheme. Here the concept of out-of-band is a contrast to the concept of in-band. Out-of-band means that the user information is transferred through a dedicated channel which is different from the channels for normal IP packets transmission. With the out-of-band scheme, the user information is not transferred with the corresponding IP packet synchronously. The IP traffic and its corresponding user information are transferred separately and asynchronously.

In this chapter, three different mechanisms for implementing the out-of-band scheme are introduced: IP packet based user identification mechanism, IP traffic flow based user identification mechanism, standalone meter mechanism. The first two mechanisms are similar to that in the in-band scheme except that the user information of IP packets is transferred to the meter independently from the transmission of the IP packet, i.e., the user information is not integrated in the IP packet. In the standalone meter mechanism, the Accounting Agent and the meter are tightly coupled as a standalone meter running in the measured host. Thus, the Agent can generate RDRs with user information like a meter and store the RDRs in the measured host temporarily. No external meter is required for monitoring IP traffic from this measured host for the purpose of IP traffic accounting.

This chapter is organized as follows: at first the principle of out-of-band scheme is introduced, then the IP packet based user identification mechanism is illustrated, after that the principle and the process of the IP traffic flow based user identification mechanism is explained, subsequently the principle of the standalone meter mechanism is analyzed, at the end some considerations concerning implementation, dependability, security of the out-of-band scheme are discussed.

5.1 Overview of the out-of-band scheme

In the in-band scheme when an IP packet is identified with its corresponding user, the user information is accompanied with this IP packet to be forwarded to the meter, which can extract both user information and traffic information from the IP packet and generate an RDR with user information for the purpose of user based IP traffic accounting. User infor-

mation integrated in IP packets is from where the name of the in-band scheme comes. However, utilizing the IP header to carry user information requires extension to be made on the IP protocol. This may be especially difficult in the IPv4 environment and legacy systems.

The out-of-band scheme, in contrast to the in-band scheme, does not integrate user information into the IP header for the purpose of storing and transferring user information. With the out-of-band scheme, the user information is not transferred with the corresponding IP packet. Instead, the user information is transferred separately from the corresponding IP packets.

In the out-of-band scheme, the user identification of IP traffic can be either IP packet based or IP traffic flow based. With the IP packet based user identification method, every IP packet is identified with its corresponding user by the Agent, then the user IP packet relationship information is recorded and transferred to the meter without being integrated into corresponding IP packet. No Dynamic User Traffic Flow Relationship Table (DUTRT) is required for this method to store user IP packet relationship information temporarily.

With the IP traffic flow based user identification method, the first IP packet of a flow will result in a user IP traffic flow relationship record. This record will be stored in the measured host by the Agent for the purpose of verifying the successive IP packets of the same flow. User and IP traffic flow relationship is transferred to the meter through a dedicated communication channel separated from IP traffic transmission. And this record will be sent to the meter for the purpose of identifying the user of IP packets and generating RDRs with user information. Both Agent and meter should maintain a DUTRT. Synchronization should be kept between these two DUTRTs.

The Agent may also function as a standalone meter generating RDRs with user information and storing them temporarily in the measured host. In this case, no user IP traffic relationship information is required to be sent to the meter. RDRs with user information stored in measured host can be collected with different accounting protocols.

5.2 Principle of the out-of-band scheme

The principle of the out-of-band scheme is that, when IP traffic passes through an Agent, it is identified with its user by the Agent, and then the user information of the IP traffic is transferred to the meter through a dedicated communication channel between Agent and meter. Instead of utilizing IP packet itself to convey its user information to the meter, user infor-

mation is encapsulated in a special User Information message packet to be transferred to the meter.

User identification in the out-of-band scheme can be defined as the process of finding the users of IP traffic and recording the user and IP traffic relationship information. The user identification can be either IP packet based or IP traffic flow based.

Considering the differences between IP packet based user identification method and IP traffic flow based identification method, they are discussed separately below.

5.2.1 IP packet based user identification

The principle of the IP packet based user identification method is illustrated in Figure 5.1.

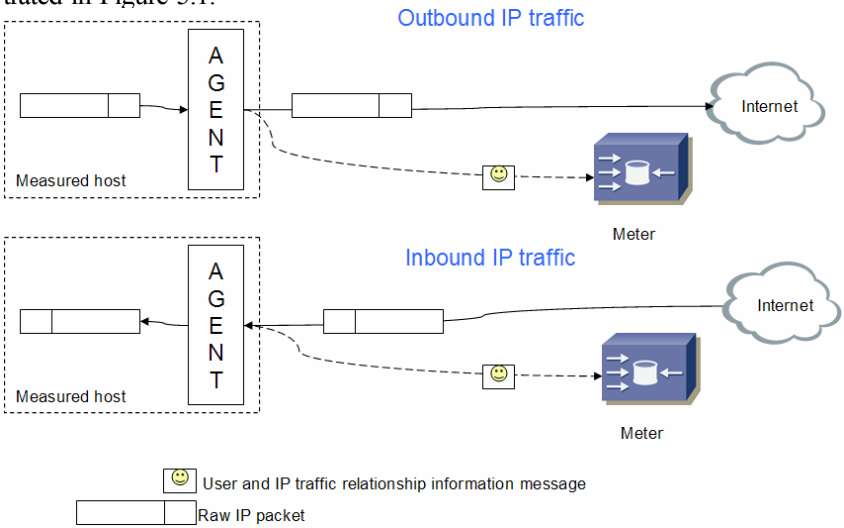


Figure 5.1 Principle of the IP packet based user identification

The IP packet based user identification method of out-of-band scheme is similar to that of the in-band scheme. However, the IP packet based user identification method of the out-of-band scheme is different from the IP packet based user identification method in the following aspects:

- The user information is transferred in a separate channel, rather than in IP packets with the in-band scheme.
- User information arrives at the meter, whereas the IP traffic does not need to pass through the meter.
- The meter does not capture any outbound or inbound IP traffic of the measured host for gathering IP traffic accounting information. All

user based IP traffic accounting information is reported to the meter by the Agent actively.

- Traditional accounting protocols such as SNMP, RADIUS and DIAMETER, transport layer protocols such as TCP and UDP can all be used to transfer the RDRs with user information.

5.2.1.1 Components in IP packet based user identification

In Figure 5.1, the following components exist for IP packet based user identification: Accounting Agent and meter.

- **Accounting Agent**

The Accounting Agent is integrated into the measured host to fulfil the following functions:

  - Identify IP packets with their corresponding users
  - Extract the IP header of every IP packet, encapsulate it with the corresponding user information into User Information message[9] packet, and send it to meter.
  - If a security mechanism should be applied to the out-of-band scheme, the Agent is responsible for negotiating security parameters and encrypting User Information messages.
  - Perform access control.

- **Meter**

  In IP packet based user identification, the meter is simply responsible for extracting user information and IP traffic information from the User Information messages to generate RDRs with user information. It does not need to be located in the place where IP traffic passes through.

5.2.1.2 Process of IP packet based user identification

In IP packet based user identification, the Agent identifies users of IP packets and extracts IP headers from IP packets. Then it generates User Information messages by combining IP packets' user information with IP headers. After that, the Agent sends the User Information messages to the meter.

The process of IP packet based user identification is:

1. When an outbound or inbound IP packet passes through the Agent, the Agent finds out the corresponding user information of this IP packet from the system.

[9] About the format of the User Information message please refer to 6.2.3.1

2. The Agent extracts the IP header from the IP packet. Then the Agent uses the IP header with user information to construct a User Information message packet.
3. The Agent sends the User Information message packet to the meter.
4. The Agent forwards the outbound IP packet to its destination or the inbound IP packet to the receiving application in the measured host, respectively.

5.2.2 IP traffic flow based user identification

The IP packet based user identification produces extra IP traffic, since every IP packet will result in a RDR to be sent to the meter. IP traffic flow based user information transmission requires only the first IP packet's user information of an IP traffic flow to be transferred to the meter. After that, the successive IP packets of the same flow can be identified with the corresponding user through finding out the flow's user information from the DUTRT. Therefore the user IP traffic flow relationship information will be generated and transferred only once for the first IP packet of every flow. Consequently, less extra IP traffic may result. For flow based user identification, DUTRT is required to record the user IP traffic relationship information.

Figure 5.2 illustrates the principle of IP traffic flow based user identification.

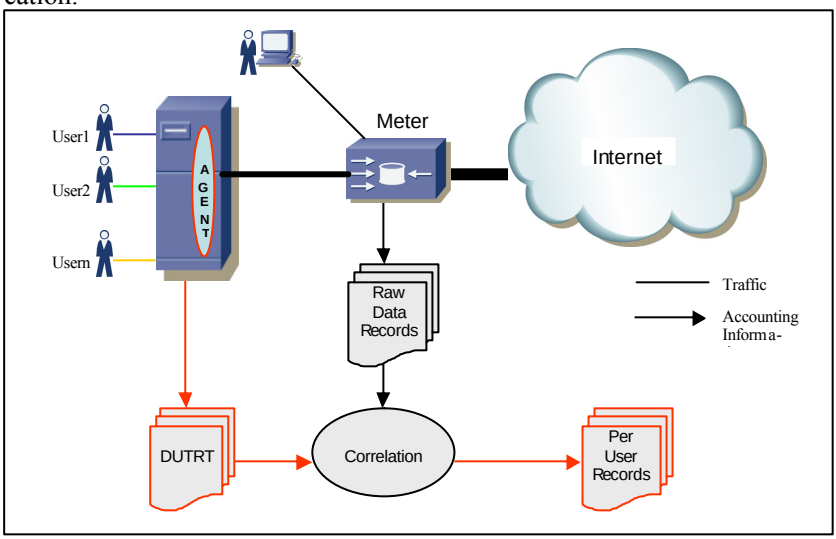


Figure 5.2 Principle of IP traffic flow based user identification

5.2.2.1 Components in IP traffic flow based user identification

The following components exist for the purpose of the IP traffic flow based user identification:

- **Accounting Agent**

The Accounting Agent is integrated into the measured host to fulfil the following functions:

- Find new IP traffic flows by checking IP packets and their corresponding users.
- Gather accounting information and user information of every new IP traffic flow to generate a user IP traffic flow relationship record. The user IP traffic flow relationship record is stored in the DUTRT temporarily.
- Encapsulate the newly generated DUTRT record into a User Information message packet and send it to the meter.
- Maintain DUTRT synchronization between Agent and meter.
- Answer the Query User Information messages[10] from the meter to find out the users of the queried IP packets.
- If a security mechanism should be applied to the out-of-band scheme, the Agent is responsible for negotiating security parameters and encrypting the User Information messages.
- Perform access control.

- **Meter**

The meter is required to be located in the key position of the network to measure IP traffic flow. Its main functions include:

- Intercept IP packets related to measured hosts.
- Identify users of IP packets by searching the DUTRT. When an IP packet's user cannot be found in the DUTRT, a Query User Information message will be sent by the meter to the Agent.
- Generate flow based RDRs with user information by combining IP traffic flows' accounting information and their user information.
- Accept user IP traffic flow relationship records from the Agent.
- Maintain DUTRT synchronization between Agent and meter.

The IP traffic flow based user identification method is different from the IP packet based user identification method in the following aspects:

- A DUTRT is required to record user and IP traffic flow relationship. Both Agent and meter must maintain a DUTRT.

[10] About the format of Query User Information message please refer to 6.2.3.2

- The meter is required to intercept IP packets related to measured hosts for the purpose of generating flow based RDRs with user information.

5.2.2.3 Process of IP traffic flow based user identification

Considering the different characteristics of inbound and outbound IP traffic flow based user identification, the process of IP traffic flow based user identification is explained according to outbound and inbound IP traffic flows, respectively.

The process of IP traffic flow based user identification with outbound IP traffic is:

1. When an outbound IP packet passes through the Agent, the Agent finds out the corresponding user information of this IP packet from the system.
2. The Agent extracts identification attributes from the IP header of the IP packet. Then the Agent checks the DUTRT in the measured host with both user information and identification attributes to verify if this IP packet belongs to an existing IP traffic flow. If there is no corresponding entry in the table, it means that this IP packet is the first IP packet of an IP traffic flow. Hence, the Agent generates a user IP traffic relationship record through combining user information and identification attributes of the flow. Then the new record about this new flow and its corresponding user is created into the DUTRT. After that, the Agent encapsulates the user IP traffic relationship record in a User Information message packet and sends it to the meter.

   If there is already an entry corresponding to the IP packet in the table, it means that the IP packet is not the first IP packet of the corresponding flow and the user information of this flow has already been sent to the meter. Therefore, the user information of this IP packet does not need to be sent to the meter.
3. The Agent forwards the IP packet to its destination.
4. When the meter captures an outbound IP packet, it extracts identification attributes and statistic attributes from the IP packet.

   The meter also maintains a DUTRT which is updated when User Information message packets are received. The identification attributes of the IP packet will be used by the meter to search the DUTRT to check if this IP packet belongs to an existing flow. If this IP packet belongs to an existing flow, the user information of the flow found in the DUTRT will be used to generate a RDR with user information.

   If this IP packet does not belong to an existing flow, the IP packet should be moved into a waiting queue. After a short time, the user information of this packet should arrive. Otherwise, the meter should

send a Query User Information message to the Agent in the measured host querying information about the user of the flow. The IP header of the IP packet is inserted into the Query User Information message.

5. When the Agent receives the Query User Information message, at first it finds out the user of the queried IP packet, and then generates a User Information message through combining the user information with the identification attributes of this flow. The '*A*' bit of the User Information message should be set to 1 indicating that this is a response to the Query User Information message. After that, the Agent sends the User Information message to the meter. And at the same time, the Agent updates the DUTRT if the information about this queried IP packet related flow and its user is not recorded in the DUTRT.

   If no user can be found for the queried IP packet, the User ID field of the User Information message must be set to "NULL" indicating that no user can be found.

6. When a meter receives a User Information message, the user IP traffic flow relationship information in the message is used to create a record in the DUTRT. If the '*A*' bit of the User Information message is set to 1, then the identification attributes of the flow in the message are used to compare with all IP packets in the waiting queue for the purpose of finding out IP packets belonging to the same flow. The matched IP packets will be used to generate RDRs with user information. If the User ID field of the User Information message is "NULL", the matched IP packet will be discarded. Otherwise, the matched IP packets will be sent to their destinations. IP packets in the waiting queue will be deleted when they stay in the queue over the time limitation.

The process of IP traffic flow based user identification with inbound IP traffic is:

1. When a meter captures an inbound IP packet destined to a measured host, it searches its DUTRT to check if this IP packet belongs to an existing IP traffic flow.

   If the IP packet does not belong to an existing flow, it means that this IP packet is the first IP packet of a new flow. Therefore, the meter sends a Query User Information message to the Agent to inquire the user of the IP packet. At the same time, the IP packet will be moved to a waiting queue.

   If the IP packet belongs to an existing flow, the statistic attributes of the RDR of this flow are calculated with the statistic value of the IP packet. Then the IP packet is forwarded to the measured host.

   When the Agent receives the Query User Information message, at first it finds out the user of the queried IP packet, and then generates a User Information message through combining the user information

with the identification attributes of this flow. If no user can be found for the queried IP packet, the User ID field of the User Information message must be set to "NULL" indicating that no user can be found. The '*A*' bit of the User Information message must be set to 1 indicating that this is a response to the Query User Information message. After that, the Agent sends the User Information message to the meter. And at the same time, the Agent updates its DUTRT if the information about this queried IP packet related flow and its user is not recorded in the DUTRT.

2. When an inbound IP packet passes through the Agent, the Agent finds out the corresponding user information of this IP packet from the system.

   The Agent extracts identification attributes from the IP packet. Then the Agent checks the DUTRT in the measured host to verify if this IP packet belongs to an existing IP traffic flow.

   If there is already a record with the same user information and the same identification attributes of the IP packet in the table, it means that the IP packet is not the first IP packet of the corresponding flow and the user information of this flow has already been sent to the meter.

   If there is a record with the same identification attributes of the IP packet but with different user information, this means that this IP packet was incorrectly identified with another user by the meter. In this case, the Agent generates a new user IP traffic relationship record through combining user information and identification attributes of this IP packet. Then the new record about this new flow and its corresponding user is added into the DUTRT. After that, the Agent encapsulates the user IP traffic relationship record in a User Information message packet and sends it to the meter.
3. The Agent forwards the IP packet to the receiving application in the measured host.
4. When the meter receives a User Information message with the '*A*' bit set to 1, it extracts the user IP traffic flow relationship record from the message and inserts the record into the DUTRT. Then the meter uses the identification attributes in the message to match the IP packets in the waiting queue. If the IP packets of the flow can be found in the waiting queue, a RDR with user information for this IP traffic flow is generated and the statistic attributes of the matched IP packets will be calculated into this flow's RDR. If the User ID field of the User Information message is "NULL", the matched IP packets should be discarded. Otherwise, these matched IP packets will be forwarded to the measured host.

When the meter receives a User Information message with the identification attributes of an inbound IP packet, it means that this IP packet was incorrectly identified by an old flow's user. Therefore, the user IP traffic relationship record in the message is added into the DUTRT for the purpose of identifying the new flow and its user. A new RDR with user information for this flow will be generated. Accordingly, rollback operations should be performed to recover the statistic calculation on the old flow. In order to avoid incoming IP traffic being incorrectly identified, the DUTRT records should be updated as soon as possible. Maybe it is better that the Agent can inform the meter to delete the closed flows through monitoring the status of the flows.

5.2.3 Format of messages exchanged between Agent and meter

In IP traffic flow based user identification, two types of messages may be exchanged between Agent and meter for the purpose of identifying users of flows. These two types of messages are: User Information message, Query User Information message. For IP packet based user identification only the User Information message is required.

The differences between the format of User Information message in the out-of-band scheme and the User Information option in the in-band scheme are:

1. Since in out-of-band scheme the User Information messages can be transferred by accounting protocols or transport protocols, the existed security mechanisms such as IPSec, SSL, TLS etc. can be applied for transmission. Therefore, the encryption mechanism is not required to be applied into the User Information message.
2. An '***H***' field is used in the format of User Information message to identify the attached attributes of IP traffic in the message. The '***D***' field and '***R***' field in the in-band scheme are not necessary in the out-of-band scheme.

5.2.3.1 User Information message

The User Information message is used to carry user and IP traffic relationship information. The format of User Information message is illustrated in Figure 5.3.

The meanings of the fields in User Information message are:

- ***Type*** is a one byte field. It is reserved here to be compatible with the format of User Information option defined in in-band scheme.

- ***Length*** is an 8 bits field that indicates the length of this User Information option in bytes.

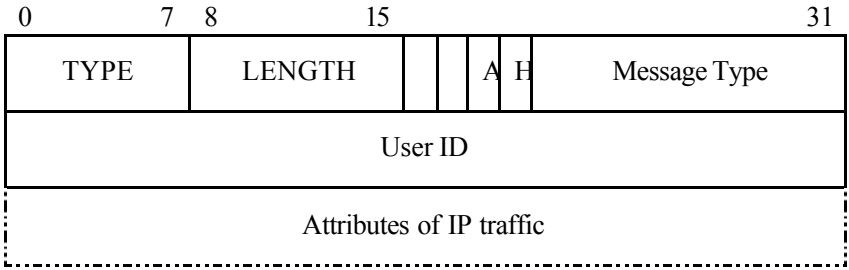


Figure 5.3 Format of User Information message

- ***A*** is a one bit field indicating whether this IP packet tagged with user information is a response to the Query User Information message. A=0 means that this is not a response to the Query User Information message. A=1 means that this is a response to the Query User Information message.
- ***H*** is a one bit field indicating whether the "***Attributes of IP traffic***" field contains an IP header or attributes of an IP packet or a flow in TLV format. H=0 means that the corresponding field contains attributes of IP packet or flow with TLV format. H=1 means that the corresponding field contains IP header.
- ***Message Type*** is a 12 bits field indicating the type of message contained in the User Information option. For the User Information message, the Message Type is 1.
- ***User ID*** field is a 32 bits field which contains User ID information of the IP traffic.
- ***Attributes of IP traffic*** is a variable length field. If H=1, then it contains an IP header. If H=0, then it contains attributes of an IP packet or a flow. The attributes include identification attributes such as source IP, destination IP, source port, destination port, etc. Statistic attributes such as sent bytes, received bytes, etc. may also be included in this field. Every attribute may have the Figure 5.4 depicted TLV format.

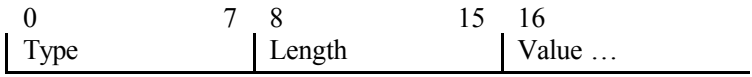


Figure 5.4 TLV format of attributes

The ***Type*** filed indicates the type of the attribute, ***Length*** field indicates the length of the ***Value*** field, and the ***Value*** field stores the value of the attribute.

5.2.3.2 Query User Information message

The Query User Information message is used by the meter to ask the Agent to identify the user of an IP packet. The IP header of this IP packet is carried by this message. The format of a Query User Information message is illustrated in Figure 5.5.

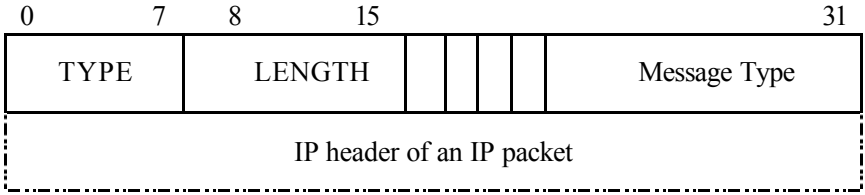


Figure 5.5 Format of Query User Information message

The ***Type*** and ***Length*** field in this format are the same as described in the User Information message, the meanings of the other fields are:

- ***Message Type*** is a 12 bits field indicating the type of message contained in the User Information option. For the Query User Information message, the Message Type is 2.
- ***IP header of an IP packet*** is a variable length field. It contains an IP header of an IPv4 or IPv6 packet which is required to be identified with the corresponding user.

5.2.4 Dynamic user IP traffic relationship table (DUTRT)

In the out-of-band scheme, the Dynamic User Traffic Relationship Table (DUTRT) is also used for the purpose of recording the user and IP traffic flow relationship information as in the in-band scheme. It is generated only in the IP traffic flow based user identification process.

The DUTRT is generated and maintained by the Accounting Agent. The meter maintains also a DUTRT which is updated by the Agent continuously. It is used by meter to identify the user of every IP traffic flow for the purpose of generating flow based RDRs with user information.

The format of the DUTRT records is the same as that described in the in-band scheme. A DUTRT record contains two types of attributes: Traffic-Originator (TO) attribute and identification attribute.

Since in the out-of-band scheme the user IP traffic flow relationship record is transferred separately from the IP packet, the DUTRTs' synchroni-

zation mechanism in the out-of-band scheme is not the same as that in the in-band scheme.

The DUTRT in the meter can be updated by the Agent either actively or passively. The active updating is that the meter sends a Query User Information message to the Agent to inquire for unknown IP packets, and the Agent sends a User Information message to the meter as a response. This happens usually when the first IP packet of an inbound IP traffic flow comes into the meter, or when the User Information message of a new flow is lost during its way to the meter, or when a User Information message carrying a user IP traffic flow relationship record of a flow comes later as the IP packets of the flow into the meter. The passive updating is that the Agent sends a User Information message about a new flow actively. This happens usually when the Agent identifies a new flow. Before an IP packet can be identified with its corresponding user, meter should not forward the IP packet. The IP packet should be kept in a waiting queue until it can be identified by its user, or time out is reached. Other synchronization problems in the out-of-band scheme can be solved by similar mechanisms as the ones described in the in-band scheme.

5.2.5 Comparisons between IP packet based user identification and IP traffic flow based user identification

The advantage of IP packet based user identification mechanism is its simplicity. It does not store any user IP traffic relationship information in the measured host, and it does not require maintaining a DUTRT. The disadvantage of this mechanism is that it produces very much traffic, since every IP packet will cause a User Information message to be sent to the meter.

The advantage of IP traffic flow based user identification is that it produces less network traffic in transporting user information and it is efficient. However, it has to maintain a DUTRT. The synchronization of DUTRT between Agent and meter is complicated. Another disadvantage is that in some cases flow identification is difficult to be achieved. An example is the encrypted flows. Identification attributes of encrypted flows may be difficult or even impossible to be gathered from IP packets.

5.3 The Accounting Agent as standalone meter

The Accounting Agent described above plays only the role of gathering and transferring user and IP traffic relationship information. In IP traffic flow based user identification, the Agent records user information and IP traffic's identification attributes in the DUTRT. RDRs are generated by the meter, whereas user mapping is performed by the correlation module.

When an Agent identifies users of IP traffic, it can also record the statistic information such as sent bytes, received bytes, etc. in the DUTRT. In this case, the Agent can be regarded as a standalone meter. And DUTRT records are RDRs with user information.

When an Agent works as a standalone meter, TO information and statistic attributes must be recorded in RDRs. Whether or not other identification attributes should be recorded in RDRs depends on the accounting policy. If detailed accounting information should be recorded to provide flow based or even IP packet based network usage information to the user, then other identification attributes should be recorded in RDRs. However, this excessive detailed accounting information offering may result in extra overhead to the measured host in which the Agent resides. Therefore, the metrics of RDRs generated by the Agent should be a trade-off between performance and the granularity of detail. For example, the simplest metrics of a RDR for user based IP traffic accounting may look like this: <**User ID, Measured Host IP, Sent bytes, Received bytes**>.

As a standalone meter, the Agent intercepts inbound and outbound IP packets, identifies the users of IP packets, extracts statistic attributes from IP packets, calculates the users' corresponding statistic attribute value, generates RDRs and stores them temporarily in the measured host. This process of generating RDRs with user information by the Agent is illustrated in Figure 5.6.

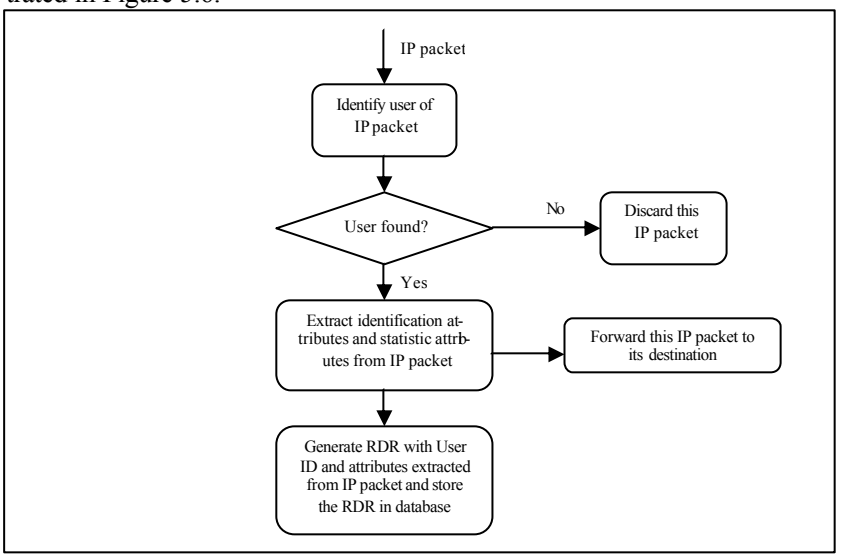


Figure 5.6 RDRs generation process when the Agent functions as a standalone meter

The RDRs with TO information and statistic information are gathered by the collector in the Mediation Layer with accounting protocols such as RADIUS, DIAMETER, SNMP, etc. or transport protocols. RDRs generated by the Agent may be collected in batch. This can improve the transmission efficiency and reduce the effect on performance caused by the Agent. The RDRs may also be compressed before being collected, which can save bandwidth in transporting RDRs. In the correlation module, these RDRs are mapped to corresponding users directly with TO & User Mapping Table. In this case, an external meter outside the measured host is not required to generate RDRs related to IP traffic from this measured host.

Compared with the IP traffic flow based user identification mechanism, the advantage of the standalone meter mechanism is its simplicity. No DUTRT synchronization is required. Compared with the IP packet based user identification mechanism which generates RDR for each IP packet, the standalone meter mechanism can transfer compressed RDRs in batch. The processing efficiency and bandwidth utilization rate can be improved. The disadvantage of the standalone meter mechanism is the performance decline on the measured host when performing user based IP traffic accounting operations by the Agent. Extra storage space is required for storing RDRs. According to the experiences of implementing user based IP traffic accounting prototype system with standalone meter mechanism, if the Agent is carefully designed and the CPU power of the measured host is strong enough, low performance decline will result in the measured host [ZhRM05].

5.4 Security considerations in the out-of-band scheme

An IP traffic accounting system, especially when it is used for the purpose of charging network resource consumption, will be potentially under attacks. Considering the characteristics of storing and transferring user information with the out-of-band scheme, potential attacks may happen in two positions: one is in the measured host and the meter in which DUTRTs are stored; the other is during the transmission of user information messages between the measured host and the meter.

For IP traffic flow based user identification, in the measured host potential attacks may include tampering the user IP traffic relationship records or even deleting them. Therefore, authentication and authorization mechanisms should be applied on accessing and modifying the DUTRT. Another issue that should be taken into consideration is privacy. If detailed user IP traffic relationship information is stored in the DUTRT, users' network access activities can be concluded from these DUTRT records. This information may reveal users' private infomation. Therefore, the DUTRT should

be protected from unauthorized access. And the information in the DUTRT should be used only for the purpose of validating users' network access activities.

The transmission of User Information message between the measured host and the meter over the insecure network can also be attacked easily. The main potential attacks may include information modification, deletion, and masquerade. Attackers may intercept in-transit User Information messages and Query User Information messages, alter the content and forward them to the destinations. Attackers may even send fake User Information messages to the meter on behalf of the Agent. The privacy related attack might be passive eavesdropping.

Therefore, for IP packet based user identification and IP traffic flow based user identification mechanisms, the messages exchanged between Agent and meter should be transferred through a secure communication channel. The following security mechanisms can be applied to prevent the attacks described above:

- Authentication between Agent and meter to prevent from unauthorized entities and masquerade.
- Utilize IPSec, SSL, or TLS mechanisms to secure the DUTRT records transmission, and apply confidentiality and integrity validation mechanisms. Since in the out-of-band scheme the user information is transferred in special packets, now existing security mechanisms can be directly applied on these user information packets to provide confidentiality, integrity and consequently to protect user information transmission from attacks such as masquerade, user information modification, passive eavesdropping, etc.

For an Agent functioning as standalone meter, the RDRs stored in the measured host should be protected from unauthorized access. The security mechanisms defined in the accounting protocols should also be applied to facilitate the RDRs to be sent to meter securely.

5.5 Dependability considerations

Trustability of the user information is an issue concerning the dependability of the out-of-band scheme. The user information of IP traffic gathered by the Agent should be trustable. Fake user information must be detected by the Agent. The user information sent to the meter should also be trustable. The security mechanisms applied in the out-of-band scheme can help to provide the trustability of the user information.

For IP packet based user identification, reliability should also be taken into consideration to guarantee that the User Information message of every IP packet arrives in the meter. For this sake, the reliable transport protocol

TCP is suitable for transferring User Information messages. Otherwise, application level reliability mechanisms must be applied to guarantee the security of User Information messages by the meter. For example, every User Information message contains a sequence number. Through that, the meter can detect how many and which User Information messages were lost. Agents should then resend these messages according to the meter's requests.

For an Agent as standalone meter, RDRs are stored in the measured host at first, and then they will be sent to meter or collector. Considering that the space for storing RDRs may be limited in the measured host, RDRs should be deleted regularly. Before RDRs are deleted, they must have been collected by a meter or a collector. If these RDRs cannot be collected in time, other dependability mechanisms should be applied. For example, these RDRs should be archived for future collection, or the granularity of accounting information should be adjusted to coarser level to reduce the size of RDRs, or the RDRs should not be deleted but the network access of the measured host should be stopped when no space is available for storing RDRs. Which dependability mechanism should be taken depends on the applied dependability policies. Without the dependability mechanisms, network resource usage of the measured host may be out of control.

The fault tolerance of the Agent or the meter is also an important factor in the out-of-band scheme. A similar mechanism as the one in the in-band scheme can be applied. Network access must be stopped when either the Agent or the meter collapses. Keep alive messages should be sent between Agent and meter to monitor each other's status. However, the keep-alive message may be different. For an Agent as standalone meter, when the Agent or the collector crashes, RDRs, which are not collected by the collector, should be protected from being deleted so that the RDRs collection can be recovered from the right position after the Agent or the meter is restarted again.

5.6 Implementation issues

According to the principle of the out-of-band scheme, the key for implementing user based IP traffic accounting is the realization of the Agent, which can gather user information of IP traffic. In order to collect the corresponding TO information of IP traffic, the Agent must be located in the measured host. Outside the measured host no mechanism can obtain the TO information of the IP traffic alone.

Because the Agent must have the ability of obtaining the TO information of the IP traffic from the measured host, usually the realization is OS dependent, in other words it is OS kernel dependent. For example, usually

the TCP/IP drivers are implemented in kernel mode. Here we consider two realization methods:

1. Kernel modification method

The principle of this method is: modifying the TCP/IP driver directly and inserting the Agent function of the user oriented IP accounting into the driver. By this means, the built-in user based IP traffic accounting Agent can generate the DUTRT. Because the Agent is located in the TCP/IP driver, it can check all IP traffic and obtain the corresponding TO information. This method is depicted in Figure 5.4.

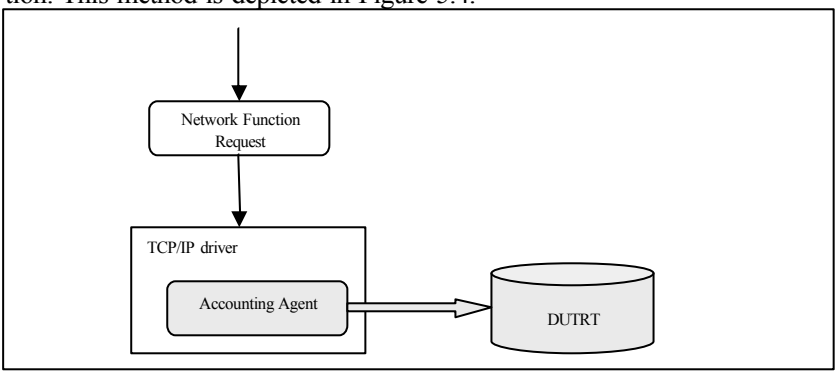


Figure 5.4 Principle of kernel modification method

This method is based on this precondition: the OS source code can be obtained and modified. It is suitable for OS producers to make this modification, or for open source code OS (e.g. Linux).

2. Kernel patch method

The principle of this method is, intercepting the network requests to the TCP/IP driver and redirecting them to the Accounting Agent. This method does not need to modify the system kernel. The Agent can be realized as a kernel patch. Figure 5.5 illustrates the principle of this method.

With the redirection technique, the network function requests to the original network function are redirected to the newly defined network system call, in which the Accounting Agent function is integrated to extract IP traffic information and record the traffic and corresponding TO information into the DUTRT. This method is suitable for the non-OS producers, who cannot get the OS source code.

Comparing the two above described user based IP traffic accounting realization methods with each other, the kernel modification method might be a better solution. Because in this method the Agent works in the TCP/IP driver, all the IP traffic related operations can be traced and recorded. However, for the kernel patch method, since it works above the TCP/IP driver, some in the TCP/IP driver performed IP traffic related operations

cannot be recorded. For example, the three-way handshake of the TCP connection is completed in the TCP/IP driver, the kernel patch method cannot capture the packages related to this process. The kernel patch method can meter most of the IP traffic. In addition, it is a simple method that does not require modifying the kernel code.

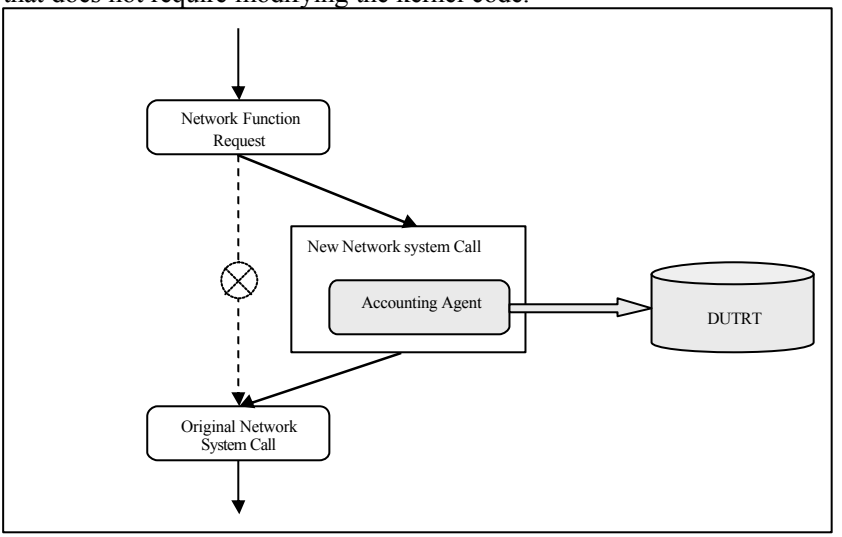


Figure 5.5 Network system call redirection

The collection and transmission of the TO information of IP traffic by Agent will cause overhead to the measured host and the network. There are some ways to reduce the performance decline:

1. Utilizing IP traffic flow based user identification instead of IP packet based user identification. This can reduce the network volume in transporting User Information message.
2. For an Agent as standalone meter, RDRs should be compressed before transmission, and it is better to collect RDRs in batch.
3. User based IP traffic accounting may be configured as an optional function for the measured hosts. If this function is not needed, or an IP address can be regarded as user, the User based IP traffic accounting Agent needs not be started.

5.7 Summary

The out-of-band scheme is different from the in-band scheme mainly in the processes of storage and transmission of user IP traffic relationship information. The out-of-band scheme transfers user information through a special channel which is separated from the normal IP packets transmis-

sion. This mechanism does not require the IP packet to be exploited for the purpose of conveying its user information. Therefore, the IP protocol is not necessary to be extended.

In the out-of-band scheme, three different mechanisms are introduced. With the IP packet based user identification mechanism, a User Information message carrying IP header and user information is sent for every IP packet. The IP traffic flow based user identification mechanism requires the Agent to send the user IP traffic flow relationship information to the meter when the first IP packet of a flow is detected by the Agent or the meter. With the DUTRT records, the meter can identify the user of successive IP packets of the same flow. The Agent can also function as a stand-alone meter. It gathers user information and accounting information of IP traffic to generate RDRs with user information in the measured host.

The three different mechanisms of out-of-band scheme have different advantages and disadvantages. Which mechanism should be selected for user based IP traffic accounting is based on issues such as performance, accounting information availability, hardware capability of the measured host.

Chapter 6 Multi-IP scheme

According to the analysis in chapter 3, the reason why the traditional IP address based accounting mechanism cannot be applied to multi-user systems is that the number of users in a multi-user system is more than the number of available IP addresses of the multi-user system. Hence, these IP addresses are shared by different users. No user owns a unique IP address in a multi-user system. The idea of the Multi-IP scheme is to allocate every user a unique IP address in a multi-user system. Through that, the traditional IP address based accounting mechanism can still be applied in multi-user systems.

In this chapter, the Multi-IP scheme for user based IP traffic accounting is introduced. At first the principle of Multi-IP scheme is explained. Then the IP address allocation mechanism in Multi-IP scheme is illustrated. After that, some issues concerning Multi-IP scheme are discussed. At the end, a comparison among three different user based IP traffic accounting schemes are made.

6.1 Overview of the Multi-IP scheme

User based IP traffic accounting is developed to improve the measurement granularity and accuracy of the traditional IP traffic accounting mechanism. User based IP traffic accounting can be regarded as a complementary mechanism to the traditional IP traffic accounting. The traditional IP traffic accounting mechanism can still work in providing user based IP traffic accounting for single user systems. IP traffic generated by different users in a single user system can be identified with corresponding users with the help of users' login information. Therefore, in single user systems, the traditional IP address based accounting mechanism is enough to achieve user based IP traffic accounting. In multi-user environment, however, the user based IP traffic accounting ability of the traditional IP address based accounting must be enhanced. In a network environment with single user systems and multi-user systems, the traditional IP address based accounting mechanism and the user based IP traffic accounting mechanism can work together to realize user based IP traffic accounting in single user systems and multi-user systems, respectively. The improvement of user based IP traffic accounting focuses mainly on the Meter Layer, in which the users of corresponding IP traffic are identified. In the Mediation Layer, only the correlation module needs to be modified to process user information.

The reason why traditional IP address based mechanism cannot be applied to multi-user systems lies in the fact that one or several IP addresses

of a multi-user system are shared by different users at the same time. User information of the IP traffic cannot be extracted from IP headers of IP traffic directly. For a single user system with many accounts for different users, the users cannot share the computer, or more precisely the IP address of the computer, at the same time. The relationship between a user and the IP address of this computer is 1:1 or 1:n during a period of time. This means that the IP address of the single user system belongs to only one user during this period of time. Therefore, the users of this computer can be distinguished by different login time, which can be obtained from the system log information of the computer.

In a multi-user system, the relationship between users and IP addresses can be described as n:m, where n>1, m>=1.

If m<n, it means that the number of IP addresses allocated to the multi-user host is less than the number of users using this multi-user system. This is the usual case of multi-user systems. The aforementioned in-band or out-of-band scheme can provide user based IP traffic accounting solution in this environment. These two schemes achieve user based IP traffic accounting though providing additional user information to identify corresponding IP traffic.

If m>=n, it means that the number of IP addresses allocated to a multi-user system is larger than the number of users using this multi-user system. In this case, if every user can be allocated a unique IP address in this multi-user system, then an IP address is equal to a user. And an IP address can be used to represent a user uniquely. Under this condition, a multi-user system can be viewed as a group of single user systems. Therefore, in this situation, the traditional IP address based mechanism can be applied for the purpose of user based IP traffic accounting. This is where the idea of Multi-IP scheme comes from. The Multi-IP scheme allocates each user a unique IP address in a multi-user system. Figure 6.1 depicts the view of Multi-IP scheme on a multi-user system.

With the Multi-IP scheme, the meter can identify the user of every IP packet through mapping an IP address to the corresponding user directly. In order to achieve user based IP traffic accounting with the Multi-IP scheme, the following issues should be taken into consideration:

- How to allocate IP addresses to different users?
- How can the meter correlate IP addresses to corresponding users?
- How to protect this scheme from attacks?

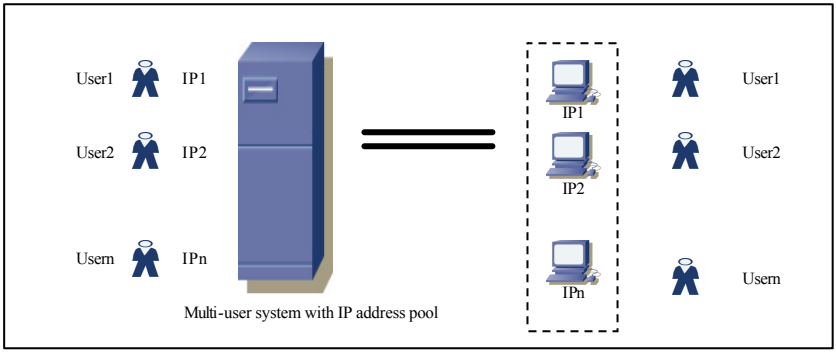


Figure 6.1 A multi-user system with the Multi-IP scheme can be regarded as a group of single user systems

6.2 User based IP traffic accounting architecture with multi-IP scheme

The Multi-IP scheme requires enough IP addresses to be allocated to every user in multi-user systems. Since IP addresses resource supplies are different in the IPv4 and IPv6 world, the Multi-IP scheme is discussed below under two different situations respectively:

- IP addresses are enough to equip every user with a unique IP address in a multi-user system as in IPv6 world
- IP addresses are scarce resource as in IPv4 world

6.2.1 Multi-IP scheme with public IP address pool

Let's at first consider an ideal situation in which IP addresses are enough so that every user in a multi-user system can obtain a unique IP address. Here IP addresses are public IP addresses, not private IP addresses [RFC1918]. For IPv4 it might be practically impossible to allocate every user in the multi-user system a unique public IP address, especially in the IP address scarce areas, e.g. Asia. However, this would be a reality in the IPv6 world from the current point of view[11]. Under this condition, every user can be allocated a public IP address which can be used to identify this user uniquely. If every user can be distinguished by the IP address, the traditional IP address based traffic accounting mechanism can meet the requirements of user based IP traffic accounting in multi-user systems without any modification. Figure 6.2 shows how a multi-user system is directly

[11] An interesting calculation [Davi03] indicates that 128 bits IPv6 address space can even provide 6.65 x 10^{23} addresses for every square meter of the Earth's surface.

integrated into a traditional IP address based accounting architecture to achieve user based IP traffic accounting with the help of allocating a unique public IP address to every user.

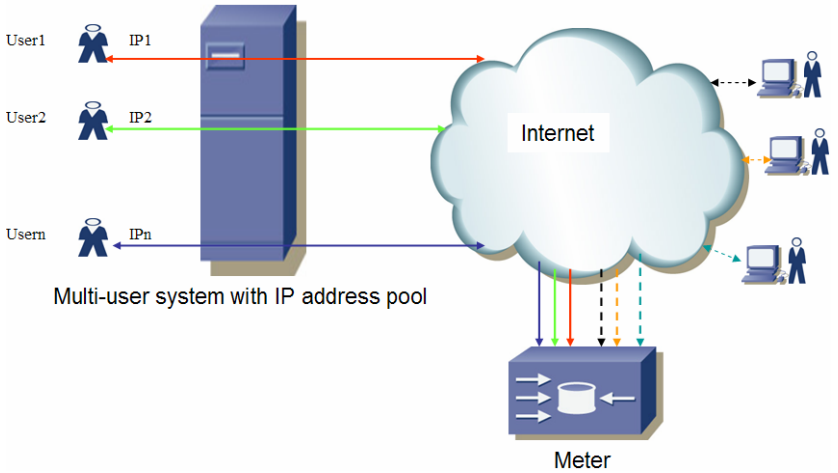


Figure 6.2 Realizing user based IP traffic accounting with the IP address based accounting mechanism through allocating every user with a unique public IP address

From Figure 6.2 we can see that, USER1, USER2, …, USERn are allocated different public IP addresses IP1, IP2, …, IPn, respectively, from the multi-user system's IP address pool for the purpose of accessing the Internet. IP packets related with these users contain their own IP addresses which belong to different users without ambiguity. The metering process for multi-user system related IP traffic is the same as that for single user system related IP traffic. Therefore, when IP packets related to the multi-user system pass through the meter, it extracts IP traffic accounting information from the IP headers to generate RDRs. The correlation module can then use IP addresses, for example, IP1, IP2, …, IPn, etc. in the RDRs to find the corresponding users: USER1, USER2, …, USERn. IP packets related to different users have different identification attributes, i.e. IP addresses of different users. The IP addresses are the key of identifying users of IP traffic by the traditional IP traffic accounting meter. Under this condition, all hosts, no matter multi-user hosts or single user hosts, are treated as single user systems by the meter. User based IP traffic accounting can be achieved through mapping IP addresses to users without any ambiguity.

6.2.2 Multi-IP scheme with private IP address pool

Since IP addresses are scarce nowadays in the IPv4 world, it is almost unpractical to allocate all users in a multi-user system with different public

IP addresses. However, private IP addresses can be utilized for this purpose.

According to [RFC1918], private IP addresses are reserved IP addresses which can be used by any organizations inside their LANs. The private IP address space is designed to be used for intranet usage, for example in the LAN of an organization. These IP addresses have no global meaning. IP traffic with private IP addresses is constrained to the intranet and cannot be forwarded to the Internet by routers. Routing information about private networks will not be propagated on links outside the intranet. This means private IP addresses are valid only when they are used inside the organizations. There are three reserved IPv4 address ranges for private IP addresses:

10.0.0.0 - 10.255.255.255 (10/8 prefix) (>16 million hosts)

172.16.0.0 - 172.31.255.255 (172.16/12 prefix) (>1 million hosts)

192.168.0.0 - 192.168.255.255 (192.168/16 prefix) (> 65 thousand hosts)

The private IP address mechanism provides a possibility to reuse IP address space, although it is usually regarded as a good but short-term solution to IP address shortage. Due to the localization constraint of the private IP addresses, hosts with private IP addresses cannot connect to the Internet directly. In order to facilitate the hosts in the intranet with private IP addresses to access the Internet without allocating them public IP addresses, a Network Address Translation (NAT) [RFC3022] mechanism is introduced to solve the problem. NAT server translates a private IP address to a public IP address for outbound IP packets and vice versa for inbound IP packets. With the NAT mechanism, hosts within an intranet can communicate with other hosts outside the private network transparently. Figure 6.3 illustrates the principle of the NAT mechanism.

In the company's LAN depicted in Figure 6.3, every host owns a unique private IP address in the form of 10.0.2.x. Within the LAN, hosts can communicate with each other using private IP addresses. A NAT server is located at the edge of the LAN with a private IP address 10.0.2.1 connecting LAN and a public IP address 131.246.123.45 connecting the Internet. When a host sends an IP packet to the Internet with the source IP address 10.0.2.123 and the transport layer port number 2345, this IP packet is forwarded to the NAT server. The NAT server translates the private source IP address into the public IP address of the NAT server, i.e. 131.246.123.45. And then it finds a free transport layer port 7890 to replace the original port number 2345. After that, this IP packet with the modified source IP address and port is forwarded to the Internet. At the same time, the private

IP address, the public IP address and port mapping information is recorded in the NAT server. When the NAT server receives an inbound IP packet using 131.246.123.45:7890 as destination endpoint reference, it changes the IP address and port in the IP packet into 10.0.2.123:2345 according to the mapping record. Then this IP packet is forwarded to the host in the intranet.

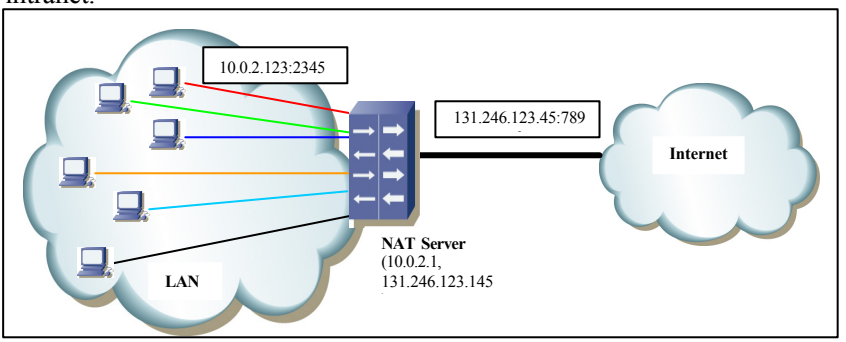


Figure 6.3 NAT mechanism

In order to achieve user based IP traffic accounting in multi-user systems, a private IP address pool, which is a group of private IP addresses, can be allocated to the multi-user system. In this situation, a NAT server is necessary to provide IP address translation support to the multi-user system for accessing the Internet. With the introduction of the NAT server, which is located between the multi-user system and the Internet, the meter must be placed between the multi-user system and the NAT server. Accurately, the meter must work before IP address translation for outbound IP packets and after IP address translation for inbound IP packets. All IP traffic between the multi-user system and the NAT server contains the private IP addresses to identify the senders or receivers in the multi-user system, whereas IP traffic between the NAT server and the Internet contains only the public IP address of the NAT server. Therefore, if the meter is not located between the multi-user system and the NAT server, different users' private IP addresses cannot be extracted from IP headers. And this means that the related users of IP traffic cannot be identified. Figure 6.4 illustrates the user based IP traffic accounting system architecture with multi-IP scheme using NAT mechanism.

In this architecture, the multi-user system owns a private IP address pool. The multi-user system may be located in an intranet or the multi-user system with private IP address pool can be regarded as a LAN containing only single user systems. Every user of this multi-user system uses its own private IP address to access network. The meter locates between the multi-user system and the NAT server. It can also be integrated into the NAT

server. Outbound IP packets are monitored by the meter to extract accounting information before they are processed by the NAT server, whereas inbound IP packets are processed by the NAT server at first to translate public IP address and port into corresponding private IP address and port before they are processed by the meter. This way, users in the multi-user system can access the Internet and different users' network resource usage can be measured according to the private IP addresses without ambiguity.

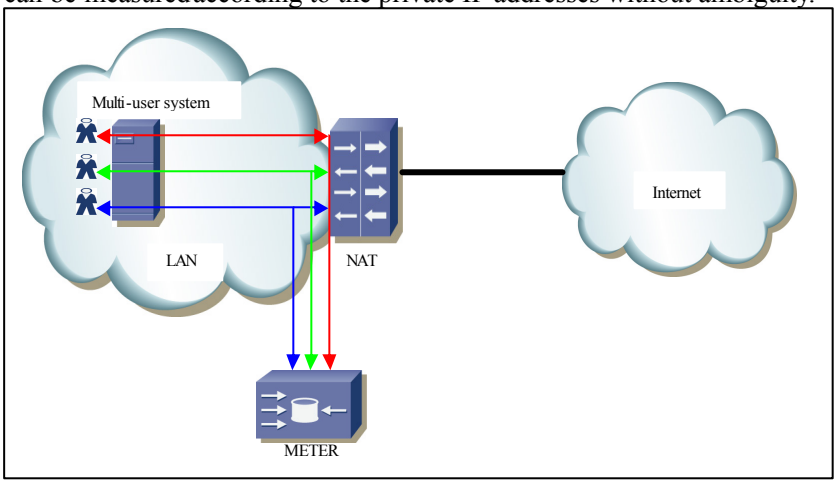


Figure 6.4 Multi-IP scheme using NAT mechanism

From the implementation point of view, the meter can either be implemented as a standalone system or be integrated into the NAT server. The advantage of the standalone method is that the meter functions will not affect the performance of the NAT server, but the disadvantage is that extra hardware is needed. The advantage of integrating meter functions into NAT server is that no extra hardware is needed, but the disadvantage is that it will cause performance decline in the NAT server.

If user based IP traffic accounting with Multi-IP scheme should be applied for an organization, every multi-user system must be allocated an IP address pool, and each multi-user system must have a NAT server responsible for its Internet access. In order to reduce hardware investment for NAT servers, all multi-user systems or several multi-user systems in an organization can be grouped into an intranet with a private IP address pool. Therefore, only one NAT server may be required to serve all multi-user systems.

After the meter collects IP traffic accounting information with the Multi-IP scheme, the traditional IP address based IP traffic accounting

mechanism can be directly applied to the generated RDRs without any change to achieve the goal of user based IP traffic accounting.

An example below explains how user based IP traffic accounting is achieved with the Multi-IP scheme.

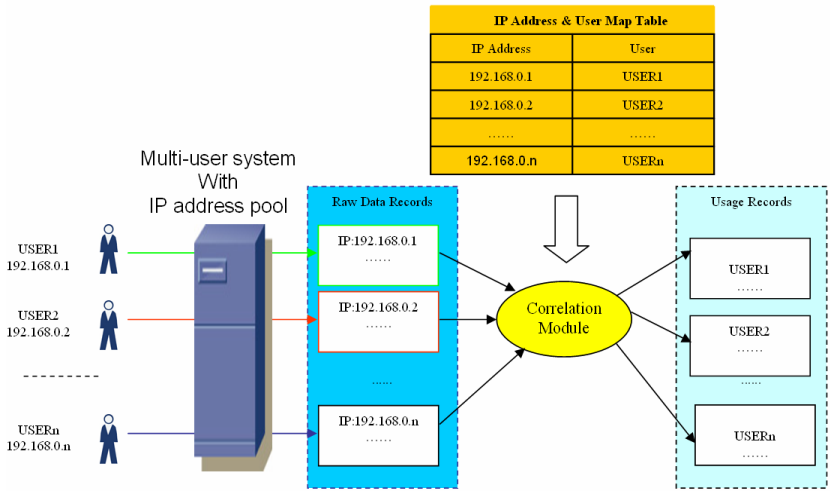


Figure 6.5 User identification process with Multi-IP scheme in multi-user system

From figure 6.5 we can find that, despite the fact that USER1, USER2, …, USERn share the same multi-user system, they do not share any IP address. Every user in the multi-user system uses its own IP address to access the network. These IP addresses can be either public or private IP address according to the IP address resource supply. In this situation, one IP address is equal to one user in the multi-user system, which is the same as in single user systems. Therefore, by simply checking the IP address attribute in a RDR, the user of IP traffic can be identified. And with the IP address & User Mapping Table which is updated by the Accounting Agent, the IP address attribute in RDRs can be correlated to the corresponding users in the multi-user system without ambiguity.

6.2.3 IP address & User Map table

The IP address & User Map table records the relationship between users and their IP addresses. When an IP address is allocated to a user, an entry is generated in the table to record the new created relationship. When an IP address is revoked from a user, the corresponding entry in the IP address & User Map table is deleted. This table is updated by the Accounting Agent. The meter and the correlation module should also keep the same table. Therefore, the IP address & User Map table in the Accounting Agent, the

meter and the correlation module must be kept synchronized. Every time when the table is updated, the Accounting Agent notifies the meter and the correlation module to update corresponding entries.

6.2.4 Accounting Agent in Multi-IP scheme

In the Multi-IP scheme, the Accounting Agent plays the role of allocating and validating IP addresses. The main functions of the Accounting Agent the in Multi-IP scheme include:

- Manage IP address allocation. Information about users' IP addresses allocation needs to be recorded.
- Report IP address and user relationship information to the meter and the correlation module.
- The Accounting Agent should provide a security mechanism to guarantee that the IP addresses in IP packets match their corresponding users. The Accounting Agent should verify the relationship between IP addresses and users. IP packets with IP addresses that cannot match the corresponding users must be discarded.

The Accounting Agent can be implemented as a part of OS kernel. It can allocate different users' IP packets with different IP addresses. Another implementation method is that the Agent is realized as a patch. The original network system allocates the single IP address of the host to all IP packets. The Agent intercepts the IP packets and changes the IP addresses of the IP packets to the corresponding users' IP addresses.

6.3 Static versus dynamic IP address allocation

With the help of private IP addresses or IPv6 addresses, it might be enough to allocate each user in a multi-user system a unique IP address. IP addresses can be allocated to users in a multi-user system through two different ways: one is static IP address binding, which allocates a permanent IP address to a user when her account is first created; another method is dynamic IP address binding which allocates an IP address to a user only when she logs in. After the user logs out, her allocated IP address will be recycled and it will be allocated to other users again.

The advantage of the static IP address allocation method is its simplicity in IP address management. A user obtains her IP address when she registers to a multi-user system and her account is created. Consequently, the IP address & User Map Table in the accounting system needs to be updated for newcomers. The allocated IP address will be taken back only when its corresponding user's account is deleted from the multi-user system. At this point, the IP address & User Map Table in the accounting system needs to be updated again. Except a user's account being created or deleted, user's

login or logout will not cause any change to the IP address & User Map Table.

The disadvantage of the static IP address allocation method is that the IP addresses are not used efficiently. An IP address cannot be allocated to other users even if its owner does not log into the multi-user system for a long time. This will certainly cause waste of IP address resource. Considering that private IP address space can provide more than 16 million IP addresses for users, this kind of "waste" is not a big problem. For public IP addresses, however, money may have to be paid for ownership of public IP addresses. Therefore, this method may cause a "waste" of money for the rarely used IP addresses.

Dynamic IP address allocation, on the contrary to static IP address allocation, can utilize IP address resource reasonably. Users acquire IP addresses only when they log into the multi-user system and release the allocated IP addresses when they log out the system. With dynamic IP address allocation, IP addresses are not allocated to users permanently. Hence, IP addresses can be reused. Therefore, the actually required IP address space for the multi-IP scheme can be reduced. This may still make sense in the IPv6 environment.

The advantage of the dynamic IP address allocation method is that IP addresses can be used efficiently. But the disadvantage of this method is the cost for managing the IP address allocation and recycling. Another side effect of this method is its effect on the IP address & User Map Table. Every time a user logs into the multi-user system, a new entry must be added to the table. And when the user logs out, this entry should be deleted from the table. All these information should also be reported to the meter or the Mediation Layer regularly, or even in real time. This will certainly increase the complexity of the correlation process and the performance of multi-user system will also be affected.

If IP address resources are not a problem or IP address space is enough for all possible users in multi-user systems, the static IP address allocation method can be chosen because of its simplicity. Otherwise, the dynamic IP address allocation method should be chosen.

The IP address & User Mapping Table in the correlation module should be updated by the Accounting Agent according to the change of the relationship between users and IP addresses. As the aforementioned static and dynamic IP address allocation strategies, the relationship between user and IP address is changed when a user's account is created or when a user logs in. This change should be reported to the correlation module in time. The report can be made in different ways: regularly, in real time, or according to the request from the correlation module. If IP traffic accounting should be made in real time, the change in the relationship between user and IP

address must be reported to the correlation module at once. Otherwise, the Accounting Agent can send the new status of the relationship between user and IP address to the correlation module at a fixed interval or when the correlation module requests for it.

6.4 Considerations on user based IP traffic accounting with the Multi-IP scheme

In order to realize user based IP traffic accounting with the Multi-IP scheme, the following operations should be performed:

1. Allocating multi-user systems with IP address pools. These IP address pools can be either public IP addresses or private IP addresses.
2. Multi-user systems should be modified to integrate the Accounting Agent for IP address allocation management and for User IP address relationship table update.
3. The IP address allocation policy can be either static or dynamic. If the static IP address allocation policy is applied, when a new user account is created, she is allocated with a unique IP address which will not be changed until her account is deleted. The IP address & User Mapping Table needs to be changed only when a user's account is created or deleted. If a dynamic policy is applied, when a user logs into the multi-user system, she is allocated with a unique IP address which will not be changed until she logs out. The IP address & User Map table needs to be changed when a user logs in or logs out.
4. For private IP address pools, a NAT server must be installed and meters must be placed between the multi-user systems and the NAT server.

The Multi-IP scheme is an elegant solution for realizing user based IP traffic accounting with traditional IP accounting mechanisms. Every user in a multi-user system acquires a unique IP address which can be used as User Identifier to identify the user without ambiguity. Therewith, a multi-user system can be viewed as a cluster of single user systems. Therefore, the traditional IP address based IP accounting mechanism can be applied for user based IP traffic accounting.

Despite of the simplicity of the Multi-IP scheme, several issues should be taken into consideration when this scheme is chosen for user based IP traffic accounting:

- This scheme requires the multi-user systems to be modified to support IP address allocation and IP address pool management. Usually the system kernel needs to be modified to accommodate these functions. This may require help from OS producers for non-open source OSs. However, how to modify legacy systems to integrate Multi-IP scheme

is still a challenge. The Accounting Agent can also be implemented as patch. In this case, the OS needs not be modified.

- Reserving an IP address pool for a multi-user system may be difficult even in the IPv6 world, since public IP address allocation is usually managed by some management organizations, e.g. ICANN [ICANN]. Owning more public IP addresses may also mean that more money has to be paid.
- Since IPv6 is still not widely applied for normal Internet usage, most multi-user systems are connected to the IPv4 network. Therefore, the above described private IP address pool might be used for the Multi-IP scheme, and consequently the NAT servers must be installed. Although the NAT mechanism provides a solution for reusing private IP address space, it has some limitations. All traffic between the multi-user systems and the Internet must be processed by NAT servers. Hence, NAT servers may become application and performance bottlenecks. NAT servers work best for reusing the private address space for client computers. But most server computers still need unambiguous public addresses, since some applications cannot work properly behind the NAT servers. A server behind a NAT may require the NAT to be configured manually with a static translation table entry to translate the inbound connection request packets to the server's private address and port. In peer-to-peer communications, peers separated by NATs might not operate correctly and must be modified for NAT awareness. NAT can also not be applied for translating encrypted packets. For IPSec packets, address and port translation invalidates the packet's integrity. These shortcomings of the NAT mechanism may limit the application of Multi-IP scheme.

6.5 Security consideration on the Multi-IP scheme

Using the Multi-IP mechanism, a multi-user system can be logically regarded as a group of single user systems. In this case, the user based IP traffic accounting mechanisms applied to the multi-user system are the same as the traditional IP traffic accounting mechanisms to a group of normal single user systems. Therefore, the security considerations and solutions for traditional IP accounting can also be applied to the user based IP traffic accounting for multi-user systems.

Despite the logical similarity between multi-user system with the Multi-IP scheme and single user system, some new issues concerning security must be taken into consideration in the Multi-IP scheme:

1. One of the probable attacks against the Multi-IP scheme is spoof or masquerade. A user who does not want her network resource usage to

be measured may try to use another user's IP address instead of her own IP address for communication. Since the multi-user system possesses an IP address pool, the spoofer may choose any IP address except her IP address from the IP address pool for accessing the Internet. If there is no measure taken to prevent from this spoof attack, there is no problem for the attacker to send and receive IP packets with other user's IP address.

In order to prevent from spoof or masquerade attacks, a validation mechanism must be introduced into the Accounting Agent to check whether the local related IP addresses in the sent IP packets match their corresponding users' IP address or not. Inbound IP packets with fake destination IP addresses will be rejected by the measured host. The spoof attacks with the "man-in-the-middle" method can be prevented by the IPSec mechanism. Here the Accounting Agent concentrates only on the attacks made in the measured host. Figure 6.6 explains the algorithm of the validation mechanism for outbound IP packets.

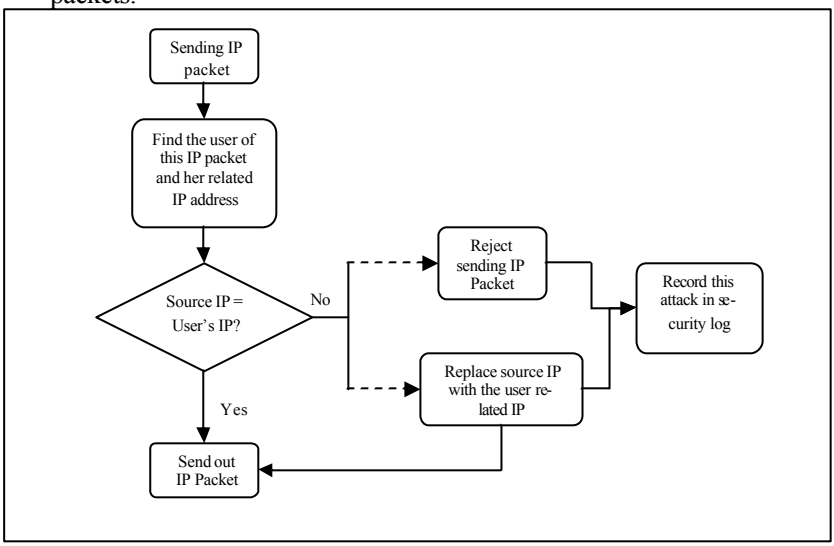


Figure 6.6 IP address validation algorithm for outbound IP packets

Whenever an IP packet is sent by an application, the anti-spoof mechanism checks whether the source IP address of the IP packet is the same as that bound to the user of the application. If the validation succeeds, the IP packet will be sent, otherwise this IP packet will be rejected to be sent or the source IP address in the IP packet will be replaced with the actual user's IP address before the IP packet is sent

out. And this spoof attack must be recorded in the security log. This mechanism can prevent the spoof attack from happening in the multi-user system.

2. Another issue is privacy. For a single user system, its IP address can leak the user information because the IP address is always bound to one user during a period of time. For example, a peeper can sniff IP packets related to one single user system and analyze the IP packets according to the IP address of the single user system to peep the privacy of the user who uses the single user system. Usually it is difficult for this kind of peeper to peep the privacy of users in multi-user system because its IP address is not bound to only one user and it is hard to ascertain which user owns this IP packet. But with the introduction of the Multi-IP mechanism, every user in a multi-user system is bound to a unique IP address. Therefore, it can become as easy for a prier to peek the privacy of users in a multi-user system as it is in a single user system.

   For the aforementioned static and dynamic IP address binding strategies in the Multi-IP scheme, the dynamic IP address binding strategy works somewhat better than static ones in against privacy leaking. However, both of them are not secure against the peepers. Therefore in order to avoid privacy to be leaked by the bound IP address, it is reasonable for users in this kind of multi-user systems to communicate with security measures such as IPSec, SSL, etc. as in single user systems to prevent their important private information from being peeped.

6.6 Comparison of user based IP traffic accounting schemes

Chapters 4, 5, 6 illustrate three different schemes of user based IP traffic accounting. These schemes have different advantages and disadvantages, and they are suitable for implementation and application under different conditions. This section makes some comparisons among these three schemes.

Table 6.1 below summarizes the comparison results among the three above described user based IP traffic accounting schemes.

In these three schemes, only the in-band scheme takes advantage of IP packets to convey their own user information. Therefore, IP protocol must be extended to accommodate the User Information option.

The Accounting Agent plays a key role in user based IP traffic accounting mechanism. It must be integrated into the measured host to handle user information related issues. Hence, these three schemes require Operation Systems (OS) of the measured hosts to be modified to integrate the Ac-

counting Agent. Although OS modification can be realized either with kernel modification method or with kernel patch method, these two types of realization methods should be regarded as modification to the OS.

For flow based user identification, control messages such as Query User Information, Query User Information Acknowledgement must be transferred between Accounting Agent and meter for the purpose of DUTRT synchronization. Hence, the meter of traditional IP accounting system needs to be improved to support the control message exchange and maintain the DUTRT. For packet based user identification in in-band scheme, since user information is in corresponding IP packets, the traditional meter needs to be improved to support extracting user information from IP packets. But for packet based user identification in the out-of-band scheme, the traditional meter can still be applied without any change to achieve user based IP traffic accounting. If the Accounting Agent is realized as a standalone meter, it might be required to be tightly coupled with the meter. For the Multi-IP scheme, user information is still implicit in the IP address. Therefore the traditional meter needs not be modified.

	In-band scheme		Out-of-band scheme			Multi-IP scheme	
	Packet based	Flow based	Packet based	Flow based	Standalone meter	Public IP	Private IP
IP Protocol extension	Yes	Yes	No	No	No	No	No
OS modification	Yes	Yes	Yes	Yes	Maybe	Yes	Yes
Meter modification	Yes	Yes	No	Yes	Maybe	No	No
Impact on local system's performance	Moderate	Moderate	Moderate	Moderate	High	Low	Low
Extra traffic volume of accounting information	High	Low	High	Low	Moderate	No	No
Impact on router	Maybe	Maybe	Maybe	Maybe	No	Maybe	Maybe
Additional hardware	No	No	No	No	No	No	Yes
Realization difficulty	High	High	Low	Low	Low	High	Moderate

Table 6.1 Comparison of user based IP traffic accounting schemes

Due to the fact that the Accounting Agent works in the measured host, it will certainly affect the performance of the measured host in which the Accounting Agent resides. The standalone meter mechanism in the out-of-band scheme may cause the highest impact on the measured host's performance, since both Accounting Agent and meter functions are integrated in the measured host. However, this mechanism may produce less extra

traffic for accounting information transmission than IP packet based user identification mechanism, since compression and batch transmission mechanism can be applied for RDR transmission. The IP packet based user identification mechanism requires every IP packet to be processed and integrated with user information. Therefore, the processing overhead may be high, and the extra traffic generated by this mechanism is the highest. The IP traffic flow based user identification mechanism requires only the first IP packet of a flow to be integrated with user information. Therefore, less extra traffic will be generated for the user information transmission. The processing overhead caused by this mechanism lies in the DUTRT search and maintenance. The Accounting Agent in the Multi-IP scheme is usually responsible for assigning IP addresses to or recycling IP addresses from different users when users login or logout. It does not intercept IP packets to gather user information. Therefore, less performance impact will be caused by this mechanism. Since user information is contained in IP addresses implicitly, no extra traffic will be generated for accounting purpose.

In traditional IP traffic accounting, meter functions are usually integrated into routers. In this case, the IP packet based user identification mechanism, the IP traffic flow based user identification mechanism, and the Multi-IP scheme will influence the performance of the router. However, if the meter is a dedicated device independent from any router, the performance of routers will not be affected. If an Accounting Agent is used as a standalone meter, no impact will be made on router.

For the Multi-IP scheme, if the IP address pool contains only private IP addresses, NAT devices are required to make this scheme applicable. Other schemes do not require additional hardware to help achieving user based IP traffic accounting except the meter.

In realizing these three different schemes, the Accounting Agent may be a bottleneck. Kernel modification method and kernel patch method can be chosen for realizing the Accounting Agent in open source OS or legacy systems environments, respectively. The aforementioned three schemes have different characteristics which may influence the realization of user based IP traffic accounting systems. The in-band scheme combines user information tightly with the corresponding IP packets. The meter can easily gather accounting information and user information from IP packets directly. Since this scheme requires the IP protocol to be extended, the difficulty in realizing it may be high. Especially for IPv4 packets, if the User Information option is implemented as an IPv4 option, the space limitation of IPv4's Options field may be an obstacle. IPv6 provides a better possibility for integrating the User Information option in IPv6 headers. Since IPv6 is still not widely deployed, we expect that the User Information option

may have a chance to be defined as an option of IPv6 extension header for user based IP traffic accounting. However, it may take a long time before the User Information option becomes a part of IPv6 protocol. For the Multi-IP scheme, public IP addresses may be impossible to be reserved for user based IP traffic accounting purposes in most current environments because of scarcity of IP address resource. The private IP address pool can solve the problem of IP address scarcity. For private IP address pool, difficulties reside in configuring the IP address pool and coordinating the NAT servers to support applications in the measured hosts. The out-of-band scheme does not require the IP protocol to be modified. One difficulty in realizing IP traffic flow based user identification mechanism is how to maintain the synchronization between DUTRTs. Sometimes, the IP packet based user identification may be realized as a supplementary mechanism in case a flow cannot be identified by extracting identification attributes from IP packets. Compared with the IP packet and IP traffic flow user identification mechanisms, the standalone meter mechanism simplifies the user information storage and transmission. In the NIPON project [NIPO03], this mechanism has been applied to realize user based IP traffic accounting systems in Solaris and Windows 2000 Terminal Server. In the next chapter the implementation of the user based IP traffic accounting prototype system will be introduced.

Chapter 7 User based IP traffic accounting prototype system implementation

This chapter introduces the implementation of a user based IP traffic accounting prototype system according to the out-of-band scheme. This prototype system is built on the basis of NeTraMet [Netr], which is a realization of the IETF Real-time Traffic Flow Measurement (RTFM) architecture [RFC2722]. The key of the implementation of user based IP traffic accounting with the out-of-band scheme is the realization of the Accounting Agent. In order to verify the user based IP traffic accounting architecture in legacy system, the Agent is implemented as a kernel patch in two typical non-open source multi-user systems: Windows 2000 Terminal server and Solaris. The Accounting Agent instead of the Winpcap in Windows and the Libpcap in UNIX is applied to capture IP packets for providing user and flow relationship information to the meter: NeTraMet. The Meter, the Reader and Manager of NeTraMet are improved to accommodate the user based IP traffic accounting ability. A Web based display application is also developed to provide statistical information of user based IP traffic accounting. Detailed implementation mechanisms are introduced in this chapter. In order to examine how the Accounting Agent affects the performance of the measured host in which it resides, the effect on performance caused by the Agent on throughput and delay of the measured host is analyzed.

This chapter is organized as follows: at first the system architecture of the prototype is introduced, and then the components of the prototype system are explained, after that the detailed implementation mechanisms about the Agent, Reader and Manager are described, at the end the performance analysis is made.

7.1 Prototype system architecture

In the "NIPON" project [NIPO00] a user based IP traffic accounting prototype system is implemented. The purposes of the implementation of the user based IP traffic accounting prototype system are:

1. Verify the user based IP traffic accounting mechanism with the out-of-band scheme suggested architecture
2. Clarify the user based IP traffic accounting records format
3. Provide a realization for analyzing the performance of the user based IP traffic accounting system

The user based IP traffic accounting prototype system is developed on the basis of the previously suggested user based IP traffic accounting system architecture. This IP traffic accounting system also conforms to the

IETF suggested IP traffic accounting architecture, and consequently this prototype system can be easily integrated into the now existing traditional IP traffic accounting system.

In this prototype system, the Accounting Agent is implemented to collect user and IP traffic flow relationship information. The meter is implemented to be responsible for RDRs generation. The open source implementation of the RTFM architecture "NeTraMet" was selected and improved to be integrated with user based IP traffic accounting functions. NeTraMet has implemented a standard traditional IP traffic accounting architecture. It follows the Internet Accounting Architecture for the traffic flow measurement architecture [RFC 2722] and the meter MIB standard [RFC 2720].

For the implementation of the user based IP traffic accounting prototype system, NeTraMet is chosen for the integration of user based IP traffic accounting techniques into traditional IP accounting systems. There are three reasons to choose NeTraMet:

- It is an open source software, and it realizes the traditional flow based IP traffic accounting.
- It conforms to the IETF suggested standard, and it has implemented the [RFC 2722] suggested IP accounting architecture.
- Most commercial IP accounting products use proprietary interfaces. However, the user based IP traffic accounting prototype system was expected to be independent of any existing commercial products.

Utilizing the suggested IETF standards can help the user based IP traffic accounting architecture to be easily integrated into now existing accounting systems.

Figure 7.1 illustrates the architecture of the prototype system.

In this architecture, the user based IP traffic accounting prototype system includes four components.

- Accounting Agent: The Accounting Agent is designed for user based IP traffic accounting according to the out-of-band scheme. It observes every IP traffic flow, and collects traffic information and the corresponding user information. The collected information will be stored into the user traffic flow relationship table. Then the information will be collected by the Meter. With this component, the user based IP traffic accounting technique can be integrated into the traffic flow measurement architecture (which belongs to the traditional IP accounting system architecture).
- Meter: The Meter receives the user IP traffic flow relationship information collected by an Accounting Agent. Then this information is processed to generate the raw accounting records with user information. These records will be stored in the MIB. The SNMP protocol

will be used to transfer the MIB records to the Reader. In this implementation, the aforementioned standalone meter mechanism is applied. The Accounting Agent and the Meter are tightly coupled in the measured host. The Meter is placed within the measured host, and its activities are controlled by rule sets which are downloaded from the Manager.

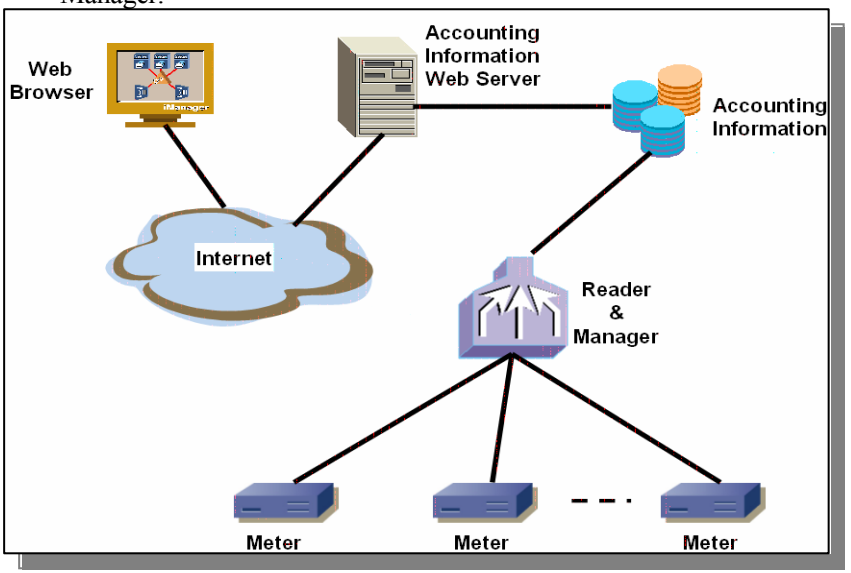


Figure 7.1 Architecture of user based IP traffic accounting prototype system

- Reader & Manager: This component implements all functions of the Reader and the Manager in the RTFM architecture[12] [RFC2722]. It requests MIB records from Meters using the SNMP protocol. The records collected from Meters will be stored into the Accounting Information database, which can be used for further processing. (In our prototype system, we simply display the accounting information). The Manager configures the Meters with rule sets and it can also control the Meters with SNMP commands.
- Display Application: This is a server side application which organizes accounting information and presents it to the user who accesses the user based IP traffic accounting information with web browsers.

[12] Please refer to Figure 2.3

7.2 Implementation environment

According to the user based IP traffic accounting architecture, the Accounting Agent must be placed within the measured host, because outside the measured host the user and IP traffic flow relationship information cannot be obtained. We choose the kernel patch method to implement the Accounting Agent. The implementation of the Accounting Agent is operating system dependant. In our NIPON prototype system, the Meter runs in the same host as the Accounting Agent. UNIX, Linux, Windows Terminal Server are three popular operating systems providing multi-user environment. There are many varieties of UNIX systems. Since Solaris is one of the typical UNIX system, and Windows 2000 Terminal Server is a different type of OS compared with UNIX and Linux, for the NIPON prototype system, Solaris and Windows 2000 Terminal Server were chosen as the implementation platforms for the Accounting Agent and the Meter. The experiences and standards used in the implementation of the user based IP traffic accounting prototype system in these two platforms can be propagated to other OSs.

The Reader & Manager and Analysis Applications are OS independent. Here we choose Windows as the implementation platform for the Reader & Manager and Analysis Applications, since it can provide a friendly GUI and is widely used.

We choose c and c++ as the implementation language. In Solaris, the standard c is used, and in Windows, c and Visual c++ are used together. Further ASP is used to implement the server side accounting information display application.

The Reader uses Microsoft ACCESS database to store records collected by Meters. Other implementations may use more sophisticated databases, such as Oracle, SQL Server, etc.

7.3 Accounting Agent

According to the user based IP traffic accounting architecture, the Accounting Agent is the key of the suggested architecture. The function of the Agent in the user based IP traffic accounting is to collect the user and IP traffic flow relationship information, in other words, the Agent is used to identify each traffic flow with its corresponding user. Since the traffic flows are changed in real time and dynamically, the Agent's user identification process is also in real time. User and traffic relationship information cannot be obtained outside the end system which produces the traffic. Therefore, the Agent must run within the measured system.

In the NIPON project, a kernel patch method is used to realize the Agent. The reasons we choose kernel patch method are:

1. The main reason we choose kernel patch method instead of kernel modification method for implementation is: the kernel modification method implementation needs to modify the OS, whereas for most operating systems the source code is not generally available.
2. The kernel patch method does not need to modify the OS. This method can log all send or receive activities of user applications using the protocols such as TCP and UDP, which are usually sufficient for the purpose of IP accounting, since most of the Internet traffic is produced by these two protocols.
3. The experiences and rules in the implementation of kernel patch method can also be applied to the implementation of kernel modification method.

The main functions of the Agent include:

1. Capture packets;
2. Extract Packets information and identify the corresponding user;
3. Generate, store and transfer user and IP traffic flow relationship information.

The Accounting Agent is operating system dependant. That means, different Agents must be implemented for different operating systems. In our NIPON project, two different packet capturing methods are utilized to implement the Agent in Solaris and Windows 2000 Terminal Server, respectively.

7.3.1 Packet capturing methods

7.3.1.1 Packets capturing method in Windows 2000 Terminal Server

A ***TDI (Transport Driver Interface)*** [TDI] call redirection method is utilized for the implementation of the Accounting Agent in Windows 2000 Terminal Server. The Agent is implemented as a TDI client driver. It runs in kernel mode.

Figure 7.2 illustrates the Agent in the Windows network architecture. The dashed grey box is the Agent.

According to [WinDDK], the Transport Driver Interface (TDI) defines a kernel-mode network interface that is exposed at the upper edge of all transport protocol stacks. The highest level protocol driver in each such stack supports the TDI interface for higher level kernel-mode network clients.

Windows 2000 includes interface modules for several popular network interfaces, such as Windows Sockets and NetBIOS. Each of these interface modules exposes a native set of primitive functions, which are accessible through standard calls from user mode. When called, the interface module maps the native function call and its associated parameters and procedural rules, to one or more calls to the underlying TDI transport driver.

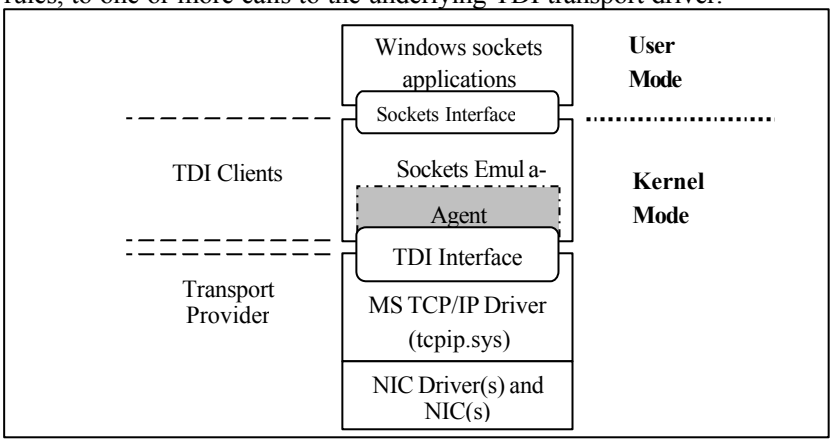


Figure 7.2 Accounting Agent in the Windows network architecture

TDI clients, which are kernel-mode drivers such as Redirector and Server, interface with the transport provider through TDI. TDI simplifies the task of developing transport drivers in that only the TDI interface needs to be coded. It also simplifies the task of developing clients by minimizing the amount of transport-specific code that must be written.

Transport drivers that expose only the TDI interface can be used only by TDI clients. To provide increased access to such transports, Windows 2000 includes an emulator module for Windows Sockets. This emulator module exposes its native set of functions, which are accessible through standard call mechanisms in user mode. When called, the emulator module maps the native functions and their associated parameters and procedural rules to TDI functions, and then calls the indicated transport driver through TDI.

The Accounting Agent lies between the Socket Emulator and the TDI interface. It captures all Send and Receive TDI calls, which are mapped from user mode calls by the emulator module. If a TDI call returns successfully, then the Agent collects accounting related information, such as source IP address and port, destination IP address and port, sent or received bytes, etc., from the TDI call. Meanwhile the Agent extracts the user information of the TDI call from the calling thread's information.

During this process, only the successful TDI send or receive related calls will be processed.

In the implementation of the Accounting Agent with this method, the captured TDI IOCTLs include:

- TDI_SEND
- TDI_SEND_DATAGRAM
- TDI_RECEIVE
- TDI_RECEIVE_DATAGRAM
- TDI_EVENT_RECEIVE
- TDI_EVENT_CHAINED_RECEIVE
- TDI_EVENT_RECEIVE_EXPEDITED
- TDI_EVENT_CHAINED_RECEIVE_EXPEDITED
- TDI_EVENT_RECEIVE_DATAGRAM
- TDI_EVENT_CHAINED_RECEIVE_DATAGRAM

7.3.1.2 Packets capturing method in Solaris

A system call redirection method is utilized for the implementation of the Accounting Agent in Solaris. The Agent is implemented as a Loadable Kernel Module.

Figure 7.3 illustrates the principle of the system call redirection mechanism.

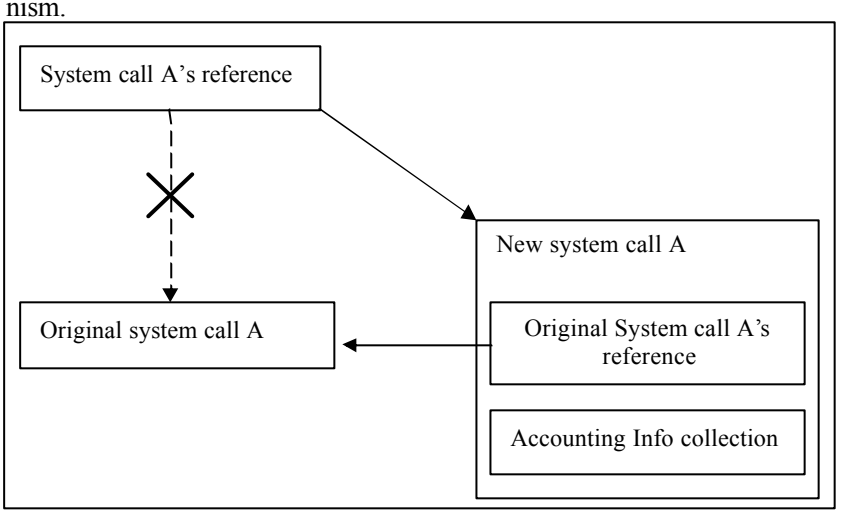


Figure 7.3 principle of the system call redirection mechanism

System calls' reference addresses under Solaris are stored in an array "sysent[]". Each entry of the array is a structure that holds information about a system call. The Agent redefines the accounting related new system calls, and then modifies the original system calls' reference addresses

in "sysent[]" to the new defined system calls' reference addresses. For example, the "sendto" system call can be redirected as below:

```
ssize_t (*oldsendto) (int s, const void *msg, size_t len, int flags,
const struct sockaddr *to, int tolen);

ssize_t newsendto(int s, const void *msg, size_t len, int flags, const
struct sockaddr *to, int tolen);

//Keep the old sendto reference address
oldsendto = (void *) sysent[SYS_sendto].sy_callc;

//replace sendto syscall with new sendto's reference address
sysent[SYS_sendto].sy_callc = (void *) newsendto;
```

When a system call is called, its reference address will be found from the "sysent[]" array. After the redirection of the system call by the Agent, the newly defined system call will be called. The newly defined system call calls the corresponding original system call at first. If the original system call returns successfully, the accounting information will be collected from this system call, and the user information will also be extracted from the calling thread. Otherwise, the syscall will be returned without further processing.

In the implementation of the Agent with this method in Solaris, the redirected accounting related system calls include:

- sendto
- receivefrom
- send
- receive
- write
- read
- writemsg
- readmsg
- readv
- writev

7.3.2 Agent workflow

Although the packet capturing methods of the Accounting Agent are different in Windows and Solaris, the workflows of the Agents in these two different systems are the same.

Figure 7.4 below illustrates the Agent's flow chart.

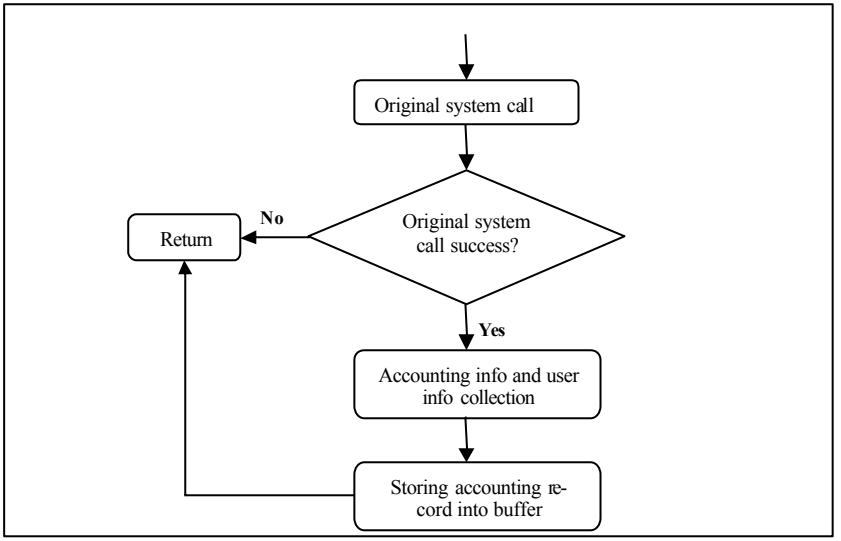


Figure 7.4 Flow chart of the Agent

The collected packet accounting information and user information are stored in the following record structure:

```
struct AccountingRecord
{
        IPAddr sourceIPAddress;
        unsigned short sourcePort;
        IPAddr destinationIPAddress;
        unsigned short destinationPort;
        unsigned short ProtAct; //protocol & Action(connect, accept, send, receive)
        unsigned long Bytes;    //number of bytes sent or received
        USERID User;        //ID of the user
};
```

These records will be collected by Meters for further processing.

7.3.3 Agent implementation consideration

The Accounting Agent runs in the kernel environment, therefore it should be carefully designed. According to the experiences of implementing the Accounting Agent in the prototype system, the following issues should be taken into consideration:

1. Reliability. The Accounting Agent runs in kernel mode. It should be designed reliably. It intercepts system calls, which will be called by threads concurrently, therefore synchronization and

mutex techniques need to be applied. In addition, the Agent needs to be designed as reentry-able. After the Agent is added to the system kernel, it should not affect the system's stability.

2. Efficiency. Since the Agent redirects the system calls, and the new system calls may be called frequently, the Agent may become a bottleneck of the system. Therefore, high efficiency should be an important goal when designing the Agent. In our prototype system, the multi-thread technique is used to improve efficiency.
3. Buffer overflow control. The collected user traffic information of the Agent is stored in the kernel buffer at first and then it is collected by the Meter. It may happen that the speed of generating user IP traffic flow information records is faster than that of collecting the records by the Meter due to busy network sending and receiving. This may soon exhaust all allocated kernel buffer. Consequently, the accounting information may be lost. Two measures have been taken to slow down the speed of the buffer exhaustion. On the one hand, we can use history records comparing technique to reduce the number of generated records. This may slow down the intercepted network system call execution. On the other hand, we can increase the records collection frequency of the Meter. However, these two methods cannot totally solve the problem. They can only slow down the speed of buffer exhaustion. Nevertheless, buffer overflow must be avoided since otherwise accounting data will be lost. In order to prevent loss of accounting information the Agent will block the send and receive function calls, when no more buffer space is available. It is unlikely that the Agent will run out of buffer space, but if it happens this will certainly affect the network related performance of the measured system.

7.4 Meter

The Meter collects the traffic and corresponding user information from the Agent and classifies them into certain flows. For each flow, the Meter accumulates certain attributes, for example the numbers of packets and bytes observed for the group. It can also aggregate, transform and further process the recorded attributes before the data is stored.

The Meter's functions include:

- accept and execute the Manager's configure command and rule sets
- collect user traffic relationship records from the Agent

- process the collected records according to accounting rules and classify them into different flows
- convert processed records into SNMP records, and store them in the MIB database
- respond to the Reader's query requests and sending MIB records to the Reader

The Meter is designed as an SNMP agent. All accounting records collected from the Agent are processed (accumulating, classifying, aggregating, etc.) according to the rule sets downloaded from the Manager. The processed records are stored into the SNMP MIB database. The MIB format conforms to the [RFC 2720] specification. The "sourceSubcriberID", which is one of the flow attributes defined in [RFC 2720], is used to identify the originator of the flow. Reader can send SNMP query requests to Meter for collecting the MIB records.

In the NIPON prototype system, traffic packets are classified into different flows. Rule sets are used to control the collection of the flow records. The rule sets can be edited for different usage purposes. The in [RFC 2722] suggested packets match algorithm is used to match the packets to the rule sets.

Figure 7.5 illustrates the workflow of the Meter.

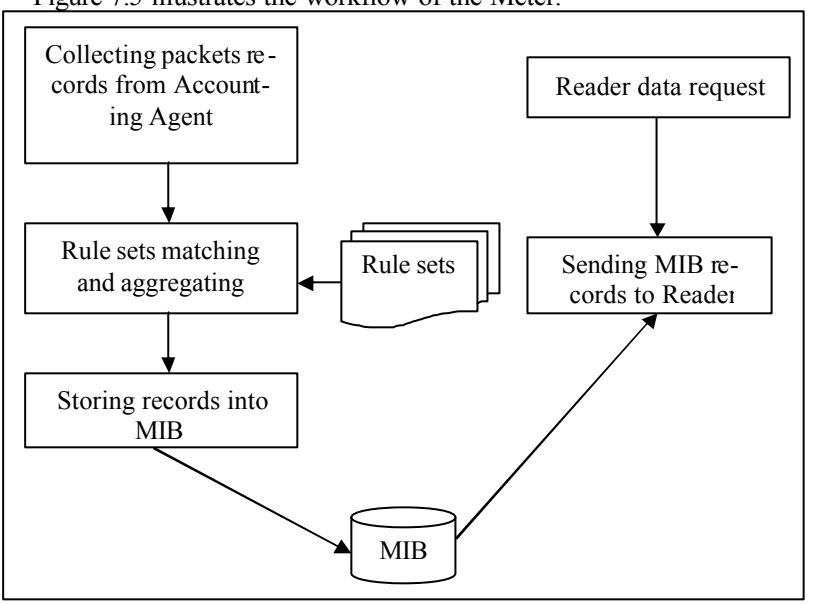


Figure 7.5 Workflow of the Meter

1. At first the rule sets are downloaded from a Manager;
2. The Meter collects traffic user relationship records from the Ac-

counting Agent, the record format is as described in 7.3.2;
3. The rule set matching machine matches every packet with the rule sets, the matched records are classified and aggregated, the not matched records are discarded;
4. Matched records will be stored into the MIB after being classified and aggregated, the record format follows the specification defined in [RFC 2720];
5. The Reader sends SNMP data requests to a Meter at regular intervals The Meter interprets the SNMP requests and executes the corresponding operations;
6. The Meter responds to the Reader's SNMP request, collects the corresponding MIB records and sends them to the Reader.

Both the Meter and the Accounting Agent run within the same host. This can reduce the overhead of transferring records from the Agent to the Meter. Because the Meter can perform the classification and aggregation, this can help to reduce the generated data volume, which consequently can alleviate the overhead to the network.

In Windows 2000 Terminal Server, the Meter runs as a Windows service. In Solaris system, the Meter can run as a normal application or as a daemon. Both of them can be started or stopped manually.

7.5 Reader and Manager

After the Meter has accumulated the users' network usage information into the MIB, the Reader collects the MIB records at regular intervals with the SNMP protocol. The collected records will then be stored into the database.

The function of the Manager is to configure Meters and to control Readers.

In the NIPON prototype system, the Reader and the Manager are implemented in one software component. The Manager is capable of managing more than one Meter. Different Meters can be controlled by different rule sets. The RDRs records in MIB can be collected by the Reader at different intervals.

The functions of the Reader and Manager component include:

1. Start or stop the data collection of a Meter
2. Edit the configuration parameters of each Meter
3. Download rule sets and configuration parameters to a Meter
4. Collect MIB records from Meters with the SNMP protocol
5. Store collected records in the database

Figure 7.6 illustrates the workflow of the Reader & Manager.

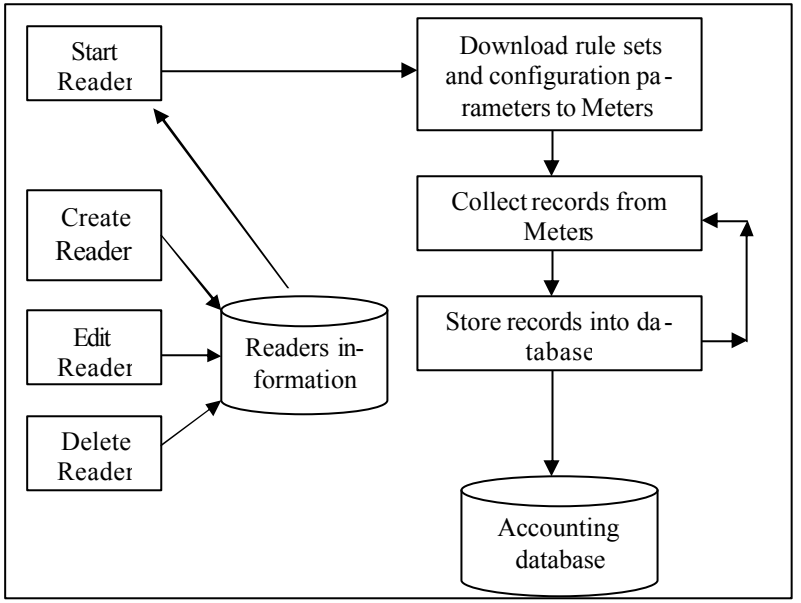


Figure 7.6 Workflow of the Reader & Manager

The workflow is described below:

1. Create a new Reader. This will add a new Reader into the Reader Table. The new Reader is created with default control parameters. These control parameters include:
   - Meter Name
   - Collection Interval
   - Configuration File
   - Garbage Collection Interval
   - High Water Mark
   - Inactive Timeout
   - Keep Alive Interval
   - Lag Time
   - Meter IP
   - Meter Port
   - MIB File
   - Rule File
   - Sample Rate

   These default parameters can be changed by the accounting system administrator. A Reader table will be used to store each Reader's parameters. Each entry in the table corresponds to one Reader.

2. Edit a Reader. The parameters of a selected Reader are displayed for editing. After modifying the Reader's parameters, the modified results will be stored into the database.
3. Delete a Reader. A record corresponding to the selected Reader will be deleted from the Reader table.
4. If a Reader is chosen and started, a Reader thread will be created. Each thread processes one Meter's information. Then the Reader downloads the rule sets and configuration parameters to the predefined Meter.
5. The Reader collects MIB records from the Meter at regular intervals, which are defined by the parameter "Collection Interval". The SNMP protocol is used to retrieve MIB records.
6. The collected records will be stored into an accounting record database. Each Meter has a table in this database. Each entry of the table represents a flow. A flow record includes the following attributes:
   - Flow Index
   - User ID
   - First Time
   - Last Time
   - Source IP Address
   - Source Port
   - Destination IP Address
   - Destination Port
   - Sent Bytes
   - Received Bytes

After a MIB record of a flow is collected from a Meter, it will then be matched with old flows to check if an entry with the same flow attributes exists. Three attributes, i.e. Flow Index, User ID, First Time, can be used to identify a flow uniquely. If there is no identical flow record, a new entry will be created for this flow. Otherwise, the accounting information (Sent Bytes, Received Bytes) of the new record will be aggregated with the old ones.

7.6 User based IP traffic accounting information display application

Since this is a prototype system only, the Billing application was not implemented. A web based display application, which can display each user's IP accounting information of one Meter, was developed. It can also display detailed information of all flows which are produced by different users. This application can be easily extended for billing purposes if pricing modules are added.

The display application includes the following functions:

1. Provide selection of different Meters.
2. Display different users' statistic accounting information in a Meter
3. Display the detailed accounting information of each flow in a Meter

The architecture of the display application is illustrated as Figure 7.7:

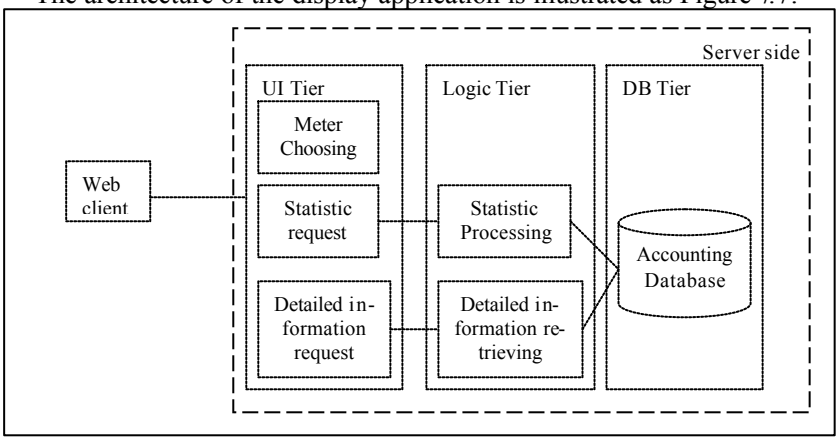


Figure 7.7 Architecture of display application

The display application runs in server side, and the clients can access the application via web browsers. Microsoft IIS was chosen as the server platform.

The following screenshots illustrate the process of accessing the user based IP traffic accounting information:

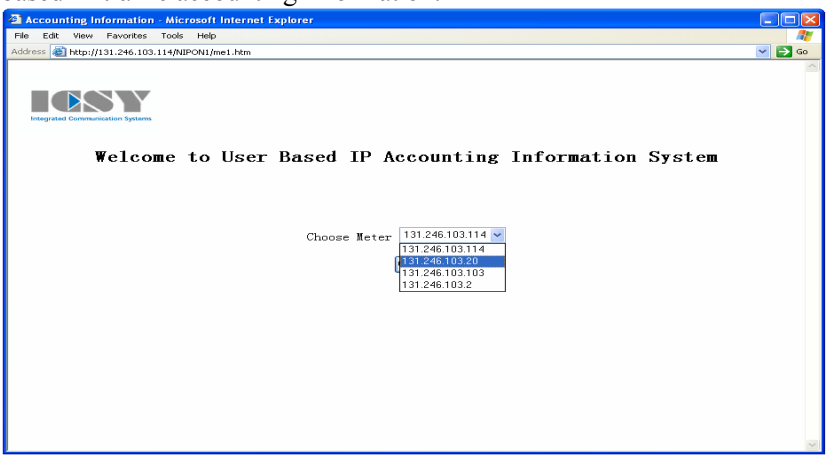
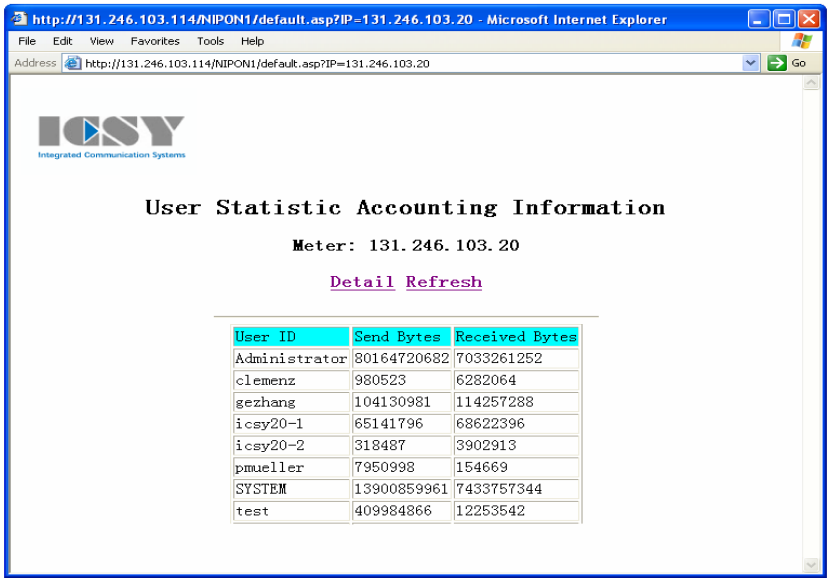
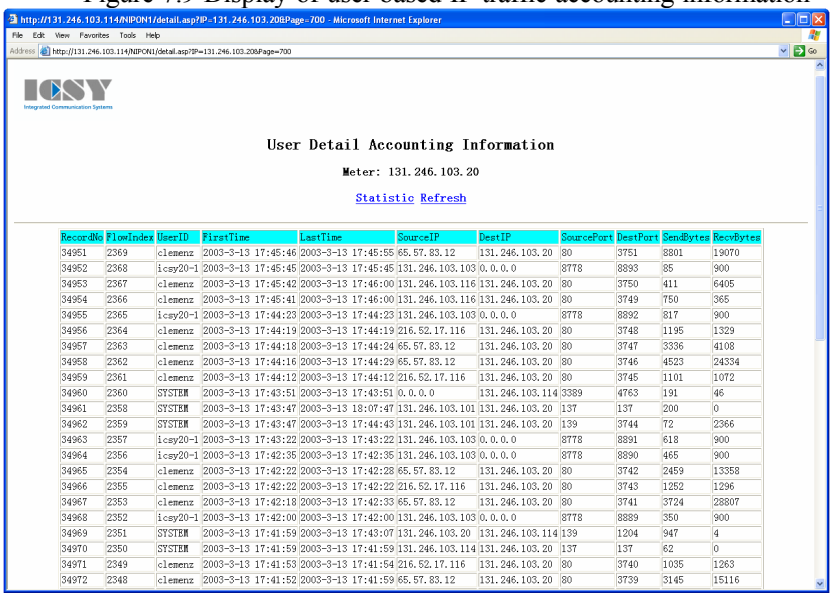


Figure 7.8 Choosing Meter



User ID	Send Bytes	Received Bytes
Administrator	80164720682	7033261252
clemenz	980523	6282064
gezhang	104130981	114257288
icsy20-1	65141796	68622396
icsy20-2	318487	3902913
pmueller	7950998	154669
SYSTEM	13900859961	7433757344
test	409984866	12253542

Figure 7.9 Display of user based IP traffic accounting information



RecordNo	FlowIndex	UserID	FirstTime	LastTime	SourceIP	DestIP	SourcePort	DestPort	SendBytes	RecvBytes
34951	2369	clemenz	2003-3-13 17:45:46	2003-3-13 17:45:55	65.57.83.12	131.246.103.20	80	3751	8801	19070
34952	2368	icsy20-1	2003-3-13 17:45:45	2003-3-13 17:45:45	131.246.103.103	0.0.0.0	8778	8893	85	900
34953	2367	clemenz	2003-3-13 17:45:42	2003-3-13 17:46:00	131.246.103.116	131.246.103.20	80	3750	411	6405
34954	2366	clemenz	2003-3-13 17:45:41	2003-3-13 17:46:00	131.246.103.116	131.246.103.20	80	3749	750	365
34955	2365	icsy20-1	2003-3-13 17:44:23	2003-3-13 17:44:23	131.246.103.103	0.0.0.0	8778	8892	817	900
34956	2364	clemenz	2003-3-13 17:44:19	2003-3-13 17:44:19	216.52.17.116	131.246.103.20	80	3748	1195	1329
34957	2363	clemenz	2003-3-13 17:44:18	2003-3-13 17:44:24	65.57.83.12	131.246.103.20	80	3747	3336	4108
34958	2362	clemenz	2003-3-13 17:44:16	2003-3-13 17:44:29	65.57.83.12	131.246.103.20	80	3746	4523	24334
34959	2361	clemenz	2003-3-13 17:44:12	2003-3-13 17:44:12	216.52.17.116	131.246.103.20	80	3745	1101	1072
34960	2360	SYSTEM	2003-3-13 17:43:51	2003-3-13 17:43:51	0.0.0.0	131.246.103.114	3389	4763	191	46
34961	2358	SYSTEM	2003-3-13 17:43:47	2003-3-13 18:07:47	131.246.103.101	131.246.103.20	137	137	200	0
34962	2359	SYSTEM	2003-3-13 17:43:47	2003-3-13 17:44:43	131.246.103.101	131.246.103.20	139	3744	72	2366
34963	2357	icsy20-1	2003-3-13 17:43:22	2003-3-13 17:43:22	131.246.103.103	0.0.0.0	8778	8891	618	900
34964	2356	icsy20-1	2003-3-13 17:42:35	2003-3-13 17:42:35	131.246.103.103	0.0.0.0	8778	8890	465	900
34965	2354	clemenz	2003-3-13 17:42:22	2003-3-13 17:42:28	65.57.83.12	131.246.103.20	80	3742	2459	13358
34966	2355	clemenz	2003-3-13 17:42:22	2003-3-13 17:42:22	216.52.17.116	131.246.103.20	80	3743	1252	1296
34967	2353	clemenz	2003-3-13 17:42:18	2003-3-13 17:42:33	65.57.83.12	131.246.103.20	80	3741	3724	28807
34968	2352	icsy20-1	2003-3-13 17:42:00	2003-3-13 17:42:00	131.246.103.103	0.0.0.0	8778	8889	350	900
34969	2351	SYSTEM	2003-3-13 17:41:59	2003-3-13 17:43:07	131.246.103.20	131.246.103.114	139	1204	947	4
34970	2350	SYSTEM	2003-3-13 17:41:59	2003-3-13 17:41:59	131.246.103.114	131.246.103.20	137	137	62	0
34971	2349	clemenz	2003-3-13 17:41:53	2003-3-13 17:41:54	216.52.17.116	131.246.103.20	80	3740	1035	1263
34972	2348	clemenz	2003-3-13 17:41:52	2003-3-13 17:41:59	65.57.83.12	131.246.103.20	80	3739	3145	15116

Figure 7.10 Display of detailed user based IP traffic accounting information

7.7 Performance analysis

As described in 7.2 the Accounting Agent and the Meter are placed in the measured system. This will certainly affect the performance of the measured system. Therefore, the Accounting Agent and Meter should be carefully designed to reduce the performance decline of the measured host in which they reside.

In order to analyze the influence on performance caused by the user based IP traffic accounting prototype system in the measured host, a performance test has been made on the prototype system - NIPON. This test emphasized mainly on the prototype system's influence on the network throughput caused by the user based IP traffic accounting mechanism in the measured host.

7.7.1 Test environment

The performance test bed is illustrated in Figure 7.11:

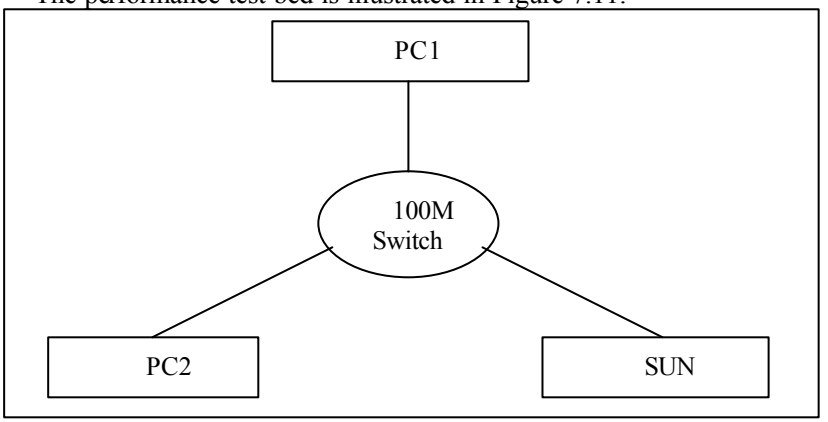


Figure 7.11 Performance Test bed

The hardware parameters are listed below:

- PC1: AMD 800 MHz, 512M, 100M Ethernet card. Windows 2000 professional.
  It is used as Reader and Manager.
- PC2: Pentium III 500 MHz, 256M, 100M Ethernet card. Windows 2000 Terminal Server.
  It is used as the measured host with Windows version Accounting Agent and Meter.
- SUN: UltraSparc 143MHz, 100M Ethernet card. Solaris 8.8.
  It is used as the measured host with UNIX version Accounting Agent and Meter.

The network performance test tool Netperf [Netp] is used to simulate an application generating network traffic and to record performance statistic data.

7.7.2 Test Procedure

The throughput test procedures are:

1. Set up the test environment according to Figure 7.11.
2. At first the NIPON system is not loaded. This is used to test the basis performance of the system without the NIPON prototype system.
3. Run the netperf-server in PC2, and running the netperf-client in SUN.
4. Use the netperf-client to send UDP, TCP packets to netperf-server with different packet sizes respectively. Record the throughput value.
5. Run the netperf-server in SUN, and run the netperf-client in PC2. Repeat step 4.
6. Load and start the Accounting Agent and Meter in the Windows 2000 Terminal Server of PC2. Repeat steps 4, 5.
7. Stop and unload the Accounting Agent and Meter from the Windows 2000 Terminal Server of PC2. Load and start the Accounting Agent and Meter in the Solaris of SUN. Repeat 4, 5.
8. The tests are repeated several times to obtain the average result values.

7.7.3 Test results and analysis

After the above described test process, we obtained the figures of performance test result in Appendix B. Every figure shows the test results in environments with and without the Accounting Agent for comparing the effect on performance caused by the Accounting Agent.

Two factors can affect the throughput: one is the network capacity and the other is the CPU speed of the host. In this test the network capacity of the three hosts are the same: 100Mbps. However, PC2 and SUN have different CPU speed.

Figure B.1 and Figure B.2 in Appendix B depict the throughput of TCP send and UDP send in case with and without the Accounting Agent in the Solaris system, respectively. These two figures show that the smaller the size of the sent packet is, the more performance impact results from the Accounting Agent. The frequency of activating the Accounting Agent depends on the number of packets rather than the size of the packets, i.e. no matter for a packet with 1 byte or for a packet with 1000 bytes the Ac-

counting Agent will be triggered for only one time. During the same period of time, the number of sent smaller sized packets is larger than that of sent bigger sized packets. Therefore, the Accounting Agent will be more frequently called in sending smaller sized packets. Consequently, this will cost extra CPU time in processing these smaller sized packets by the Accounting Agent, and the accumulated sending delay will increase Hence, the throughput of the measured host will decline.

Figure B.1 and B2 also show that, when the packet size is increased to about 300 bytes per packet, the throughput of the measured system with NIPON is almost the same as that without NIPON. During a fixed transmission period, more small IP packets can be sent than larger ones. This is due to the fact that larger IP packets need more CPU time for processing. Therefore, the frequency of activating the Accounting Agent is reduced during this fixed period of time, and consequently less performance decline results from the Accounting Agent. This result indicates that the performance impact caused by the NIPON Accounting Agent on the throughput is mainly in sending of IP packets of smaller size.

The throughput results of TCP send and UDP send in Windows system are shown in Figure B.5 and B.6. These two figures show that the Accounting Agent causes almost very little effect to the throughput in sending smaller size packets comparing with that in Solaris. The reason is the higher CPU power of the PC2. Compared with the UltraSparc 143 MHz CPU of the SUN, the Pentium III 500 MHz CPU of the PC2 is more powerful. Therefore, less processing overhead will result from the Accounting Agent in the PC2. From the above observations we can conclude that the Accounting Agent causes less impact on the performance of the measured host if the CPU of the measured host is powerful enough.

The test results of the receive operation in Figure B.3, B.4, B.7, and B.8 show that the Accounting Agent has only a slight impact on the throughput. The reason is that: although the sending host sends IP packets of small size, when the packets arrive in the receiving host, they will be accumulated to one or more big-sized packets before they are forwarded to the receiving instances.

Usually the receiving host will not use the same size buffer to receive the incoming packets as the sending host. Instead, a bigger-sized buffer will be used to receive the packets. Therefore less receive operations will be made. Considering that an Accounting Agent will be activated only when a receive operation is performed, the frequency of activating the Accounting Agent in the receiving host is lower than that in the sending host. Therefore, the Accounting Agent causes less impact to the throughput in receiving packets. Certainly if the receiving site receives the packets with a

small-sized buffer, this will cause similar throughput decline as sending operations.

From the test results we can conclude that NIPON user based IP traffic accounting prototype system does not produce very much overhead to the measured system. Its performance decline becomes obvious only in the case of sending a large number of small-sized packets in hosts with lower power CPU.

Using user based IP traffic accounting will certainly affect the performance of the measured host. We cannot avoid this performance decline. What we can do is to alleviate it. This can be done by carefully designing the accounting system, or choosing implementation methods with better performance. If the user based IP traffic accounting is needed, performance reduction must be taken into consideration. Just like the accounting function in Cisco routers, if the Netflow is chosen to run for the purpose of IP traffic accounting, the performance decline of the routers have to be accepted [LuCo99].

Chapter 8 User based IP traffic accounting in distributed computing environments

In the previous chapters, the user based IP traffic accounting mechanisms are discussed and introduced in non-distributed computing environments. Users, applications, and measured hosts are regarded tightly coupled in these environments. In distributed computing environments, users, applications and hosts distributed in different locations are joined together to form a virtual organization (VO) [FoKT01] to fulfill a task. A host that supports distributed computing can execute the jobs submitted by VO users on behalf of one or more local users. Hence, a local user account may not be used by a locally registered user. Instead, a local user account may be used by one or more VO users, and the local user identifier may become ambiguous. Under the distributed computing environments, even a single user system may become a multi-user system. These characteristics of distributed computing make the traditional IP address based IP traffic accounting impractical to achieve accurate IP traffic accounting. Therefore, it is necessary to introduce the user based IP traffic accounting mechanism into distributed computing environments for the purpose of providing more accurate and finer grained IP traffic accounting information.

In this chapter, the new challenges on IP traffic accounting brought by distributed computing will be analyzed. Accordingly, the user based IP traffic accounting solution is introduced to meet the new requirements on IP traffic accounting in distributed computing environments. VO users can be mapped to local user accounts with different methods. This requires different user based IP traffic accounting mechanisms to be applied correspondingly. The user model described in chapter 3 is extended for providing finer grained IP traffic accounting information in distributed computing environments. User based network access control according to QoS requirements in distributed computing environments is also discussed in this chapter.

8.1 Challenges on IP traffic accounting in distributed computing environments

As a consequence of the widespread usage of computers and the development of Internet technology, distributed computing is becoming popular. On the one hand, many resources required by an application may no longer exist only in a single computer. On-line shopping, as an example, might need a customer's authentication and purchasing information to be provided by the customer's computer. Whereas authentication database and goods related database may be in different systems of the on-line shopping

company. On the other hand, geographically distributed computers can join together dynamically to accomplish a task. For example, the SETI@home [SETI] project distributes a radio telescope data analysis program on volunteers' computers all over the world for the purpose of searching for extraterrestrial intelligence.

A distributed computing system is the system architecture that makes a collection of heterogeneous computers, workstations, or servers act and behave as a single computing system. In such a computing environment, users can uniformly access and name local or remote resources, and run processes from anywhere in the system, without being aware of which computers their processes are running on [HaPa04].

Especially with the emergence of Web Services [BHMN04] and Grid Computing [FoKT01, FKNT02], distributed computing is attracting more attention from not only academies and institutes but also from governments and the commercial world [BeFH03].

8.1.1 Necessity of user based IP traffic accounting in distributed computing environments

In traditional computing systems, users are tightly coupled with the underlying hardware and the administrative domain. For example, users are bound to individual computers by means of user accounts. Users must own "real" accounts on every single machine to which they access. In distributed computing systems, users, applications and data are decoupled from individual computers and administrative domains. A user in a distributed computing environment may be not bound to an individual machine, or in other words, a user may not belong to any concrete host. A user can exist in a virtual organization (VO) which is constructed with distributed computing technology, e.g. Grid, without binding to any host physically. If this user needs to submit jobs to a host, her VO user account must be mapped to a local user of the host at first, and then this job can be run on behalf of the local user in the host. With this mechanism, a user can execute applications in remote machines.

With the distributed computing technology, the computing resource of a computer can not only be utilized by people registered as local users in this host but also be shared by VO users registered in a VO. VO users can submit jobs to the distributed computing node (DCN), execute applications in the DCN, and utilize resources such as CPU, memory, storage and network of the DCN. All these distributed computing activities performed in the DCN according to the requests of the VO users are executed on behalf of one or more local user accounts of the node. A VO user must be mapped to a local user before her job can be executed in the node. There-

fore, the IP traffic of this DCN may be related not only to local users but also to VO users, or accurately to the mapped local user accounts of VO users. In this condition, it is apparently that the IP traffic related to this node cannot be identified uniquely with corresponding users through IP address, since the IP address of this node is in fact shared by several users.

This coexistence of local users and VO users in a host makes network access control and network resource consumption management more difficult and complex. In a distributed computing environment, a host delivering computing services should be regarded as a multi-user host from the IP traffic accounting point of view, since the IP address of this host is shared by more than one user. Considering the discussion in chapter 3 about the flaw of the traditional IP address based IP traffic accounting mechanism in multi-user systems, it is necessary to introduce the user based IP traffic accounting mechanism in the distributed computing environments for the purpose of providing more accurate and finer grained IP traffic accounting information. With user based IP traffic accounting, accurate information about network resource consumption can be provided and finer grained access control can be achieved in distributed computing nodes.

8.1.2 Users in distributed computing environments

In distributed computing there exist not only normal users bound to user accounts in a computer but also VO users that can execute jobs in remote hosts. Here user has the same definition as in chapter 4 and it means the real person who submits the distributed computing jobs. A VO user account or VO user is the user account registered in a VO for consuming distributed computing services. A user account is an account created in a DCN or a host. For example, a UNIX account can be a normal user account. An application or a job is allowed to be executed in the DCN only when it runs on behalf of a user account of the node.

Traditionally, in a host a user account is directly related to a user. For example, a user account with the name "alice@192.168.0.100" relates to student Alice who has registered to the host 192.168.0.100. From the registration information of the user account "alice@192.168.0.100" we can find the real user Alice. However, in distributed computing environments a user may not be bound to a user account in a single computer, instead, she may be bound to a VO user account that will be mapped to user accounts in DCNs. Figure 8.1 illustrates the relationship among user, VO user and user account in non-distributed computing and distributed computing environments, respectively.

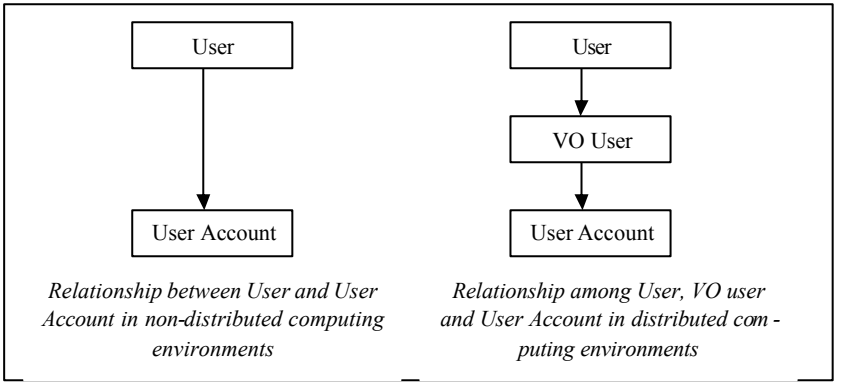


Figure 8.1 Relationship among user, VO user and user account in non-distributed computing and distributed computing environments

In a distributed computing environment VO users can be mapped to local users with different methods:

1. Mapping VO users to a single user account

   For this user mapping method a special user account is reserved for distributed computing. All submitted remote jobs are executed on behalf of this reserved user account. For example, Condor utilizes a reserved user ID " nobody" to execute jobs for VO users that do not have an account in the Condor flock [ELDE96]. Another example is the PUNCH system. It allocates different logical user accounts to VO users for executing jobs. All these logical user accounts are mapped to a single physical account [KaFo99]. Legion [Legi06, GrWu97] treats default Legion accounts as a "guest" UNIX account in the local host. This VO user to local user mapping method is very simple, since there is no need to create user accounts for VO users for executing jobs. This method also requires less resource in managing the account for VO users. However, this single user ID makes it difficult to distinguish different VO users' resource usage so that accurate accounting information is difficult or even impossible to be collected. Furthermore, the accounting system needs to store the VO wide identifiers, such as X.500 [CCIT93] distinguished names (DN) along with the usual account information. IP traffic metering and accounting will still be done locally on the basis of the local UID, after that the second mapping from UID to VO UID must be made.

   Another situation in which applications are executed under the same user account is, when a user provides distributed computing services, the distributed jobs can be run under her user ID, i.e. under her own user environment. For example, in SETI@home each participant can

install and run the SETI@home client program under her/his user environment. In this case the SETI@home client is not explicitly run for a VO user, therefore the user mapping between VO users and local user accounts is not required. The problem of this case is that it is difficult to isolate the user's local execution environment from the remote jobs, which may make privilege control and security difficult.

Figure 8.2 illustrates the principle of this user mapping method.

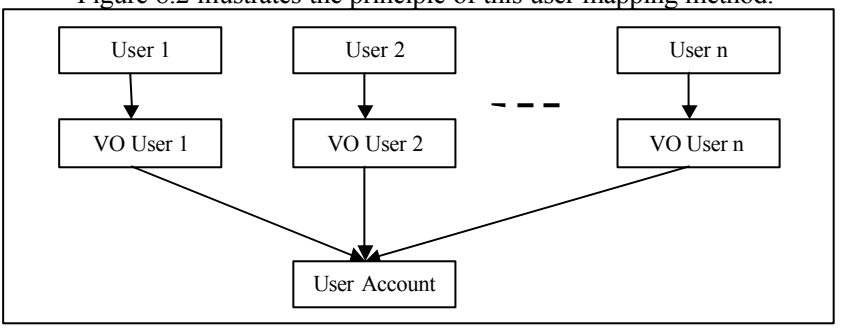


Figure 8.2 Mapping VO users to the same user account

2. Assigning every VO user a unique local user account statically

This user mapping method, in contrast to the first, does not reserve a single user account for all VO users. Every VO user, who has the authority to execute applications in a DCN, will obtain a unique local user account from the host. The VO users can be mapped to local users statically or dynamically.

The static user mapping is, when a VO user account is created in a virtual organization, correspondingly user accounts of this new VO user will be created in DCNs. When this user submits jobs to a DCN, her VO user account will be mapped to her related "real" user account in this node. The Globus [FoKe97, BEFK00] utilized this method to map a X.509 [RFC2459, RFC3480] identity to a single local user account statically. In Globus Tool a so-called "gridmap" file records the relationship between grid identifiers and corresponding user accounts. When a job of a grid user is received for execution, it is forked as a process owned by the corresponding local UNIX user, which can be found from the "gridmap" file. In the European Datagrid (EDG) Testbed 1, the "gridmap" file is updated by periodically querying authorization information in LDAP servers [Alif03]. The process of binding VO users to local users in Legion is the same as the process in UNIX where an administrator must create a user account for a new user. A Process Control Daemon (PCD) with administrator privilege is used to regulate the ownership of every remote job.

The disadvantage of these static local user accounts is that they cannot reflect the dynamic characteristic of distributed computing, in which users all over the world may submit their jobs. Static user mapping cannot meet the requirement for different dynamically joining VO users, and preconfigured user accounts cannot reflect the dynamically changing policies. Statically allocated user accounts may also result in waste of resources in maintaining rarely used user accounts. Figure 8.3 illustrates the principle of allocating every VO user a unique user account statically.

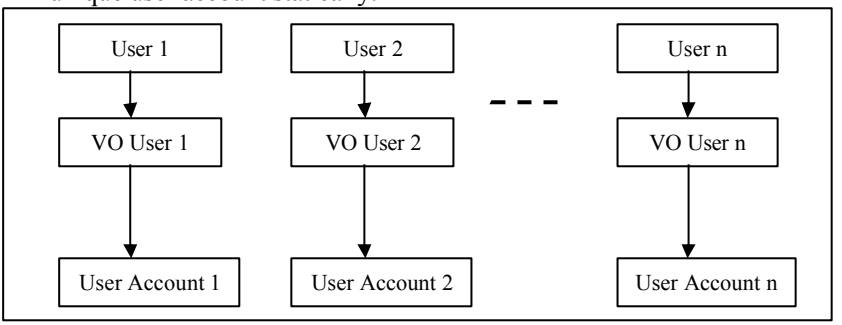


Figure 8.3 Allocating every VO user a unique user account statically

3. Assigning user accounts to VO users dynamically

Dynamic user mapping does not require the binding of a local user account to a VO user to be predefined. The local user account is created for or assigned to a VO user dynamically. The single local user account mapping method and the static user account mapping method cannot reflect the dynamic characteristics of the users in distributed computing. Especially in Grid computing the number of Grid users may be tens of thousands, a single local user account will make management on Grid users very difficult whereas pre-defined user accounts will consume or even waste a great number of resources for maintaining these user accounts in local systems.

Many efforts have been made in providing dynamic local user accounts to Grid users [McNa03, HaAt01, KeRD03, KaFF01, Pool05, TaBK03]. For dynamic user account mapping method, a local user account pool is usually built containing preconfigured local user accounts for distributed computing purpose. These dynamic user accounts in the pool may be grouped according to their different privileges. Unlike normal user accounts that belong permanently to their real world owners, a dynamic user account is bound or leased to a VO user temporarily. The selection of a pool and the binding of the VO user to an available dynamic user account from that pool are based

on the VO user account management policies. The temporary relationship between dynamic user and VO user must be written down in a log file to record the status of dynamic user accounts allocation and help future VO user accounts related management. After the binding of a dynamic user account to a VO user, the VO user's activities are subjected to the dynamic user in the local system.

The dynamic user account is freed and reclaimed to the user accounts pool according to the VO user account management policies. For example, when a job execution is finished, pre-configured termination time is reached, a VO user's Quota is used up, etc. all can cause the temporary relationship between dynamic user and VO user to be terminated. At the termination time, all this dynamic user account related resources will be released. For example, the user account related processes are killed and files owned by this user account are deleted. Then the user account is returned to the user accounts pool. This activity should also be recorded in the log file.

Figure 8.4 illustrates how a local user account "x" is bound to different VO users dynamically.

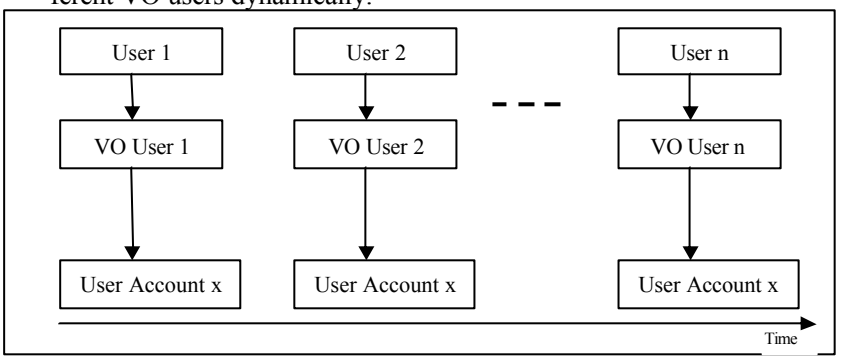


Figure 8.4 A user account is repeatedly allocated to and reclaimed from different VO users

The advantages of this dynamic user accounts mapping method are:

- There is no need to allocate every VO user a real user account in the local system. This can prevent unnecessary resource allocation for temporary user accounts.
- Therefore the scalability of this method is very good, which can meet the requirement for a potential huge number of VO users.
- Different dynamic user accounts may be preconfigured with different privileges. VO users with different authorities can be mapped to their corresponding dynamic user accounts. This can

meet dynamically changed requirements on resources from different VO users.

8.1.3 Issues concerning IP traffic accounting in distributed computing environments

In distributed computing environments, user based IP traffic accounting is necessary for providing more accurate IP traffic accounting information. The variety of the relationships among user, VO user and user account in distributed computing environments requires different mechanisms to identify the user of IP traffic. The user model applied in non-distributed computing environments should be improved to meet the requirements of identifying different users' IP traffic in distributed computing environments. For example, if all VO users are mapped to a reserved user account in a DCN, all applications, which are executed for different VO users, are run on behalf of the same reserved user in the DCN. Under this situation, according to the user model defined in chapter 3, the Traffic-Originators of the IP traffic related to these applications are all identified by the same reserved user and the IP address of the DCN. Therefore, with the user model defined in chapter 3, IP traffic of different VO users cannot be distinguished. That means that the IP traffic of all these different VO users' applications might be treated as being related to the same user.

In non-distributed computing environments, a user account in a host is directly related to a user. In distributed computing environments, a user account in a host may be related to one or more VO users, statically or dynamically, temporarily or permanently. A VO user account is directly related to a user account, whereas the user is directly related to the VO user. Figure 8.1 – 8.4 illustrate the relationship among user, VO user and user account in different conditions.

This variety of the relationships among user, VO user and user account in distributed computing environments also makes the correlation between Traffic-Originator and user becoming complex. Since the relationships may be temporary or permanent, different measures should be taken to record the relationship among user, VO user and use account for the purpose of correlating TOs to corresponding users.

8.2 Improved user model for user based IP traffic accounting in distributed computing environments

The user model defined in chapter 3 is based on the assumption that different users have different user accounts in a host, and different users' applications are executed on behalf of the corresponding user accounts regis-

tered in the host. Therefore, the Traffic-Originator can be identified with the 2-tuple <***Host-Identifier, User Identifier***> without ambiguity. Considering that different VO users may be mapped to only one user account in a DCN, the 2-tuple Traffic-Originator may be not able to identify the traffic originator of IP traffic. For example, VO user 1 and VO user 2 submit two jobs to a DCN which has the IP address 192.168.0.100. This node provides a reserved user account, e.g. "dummy", for running applications of VO users. Therefore, VO user 1's job related application 1 and VO user 2's job related application 2 are executed on behalf of user account "dummy". According to the user model in chapter 3, IP traffic related to application 1 of VO user 1 is identified with TO <192.168.0.100, dummy>, whereas the IP traffic related to application 2 of VO user 2 is also identified with the same TO <192.168.0.100, dummy>. Therefore, this TO cannot be mapped to the corresponding VO users correctly.

Considering the dynamic VO users and user accounts mapping method in a DCN with IP address 192.168.0.100, VO user 1 is mapped to user account x for executing applications in this computer, consequently the IP traffic related to applications are identified with TO <192.168.0.100, x>. After VO user 1's jobs are finished, the user account x is not bound to VO user 1 and the user account is reclaimed. When VO user 2 submits jobs to this host, supposing user account x is allocated again to VO user 2 for running applications, the IP traffic related to VO user 2's applications are also identified with TO <192.168.0.100, x>. When RDRs related to VO user 1 are not correlated to the corresponding user in time, this TO will also make confusion in distinguishing RDRs related to VO user 1 and VO user 2.

8.2.1 Improved user model for user based IP traffic accounting in distributed computing environments

In order to solve above described problems, the user model described in chapter 3.3.1 can be improved as follows:

- *Host-Identifier* is a unique identifier for an end-system of the network layer. In the context of IP networks, an IP address can be used as a synonym for a Host-Identifier, since IP addresses are unique numbers for network layer devices, at least within an administrative domain.
- *User-Identifier or UID* is a unique identifier for an account on a measured host.
- *Application-Identifier* is a unique identifier of an application on a measured host during a period of time. An example is the Process ID (PID).
- *Timestamp* identifies the time when TO information is collected.

- *Traffic-Originator* ::= <*Host-Identifier*> [<*User-Identifier*> <*Application-Identifier*> <*Timestamp*>]. A Traffic-Originator (TO) is used to uniquely identify the entity which is responsible for specific outbound and inbound IP traffic flows.
- *VO User* ::= 1*<*Traffic-Originator*> is a unique identifier for a user account in a Virtual Organization (VO).
- *User* ::= 1*<*VO User*> is a unique identifier for a real person or a group of persons which are associated with one or more VO Users. Each VO User is associated with exactly one user. Usually a user identifies one real person who has one or more VO user accounts. When a group of real persons share a VO user account, this group may be described by one user.
- *Purchaser* ::= 1*<*User*> is a unique identifier of a person or an institution who will pay for the IP traffic related to one or more users.

Figure 8.5 illustrates the improved user model for user based IP traffic accounting in distributed computing environments.

With this improved user model the TO of IP traffic can be identified with the 4-tuple <IP Address, UID, PID, Timestamp>. IP address is used to distinguish different DCNs. UID is the user account in the node for running distributed computing applications. The PID is the process ID of the application used for the purpose of distinguishing different users' applications. The Timestamp is the time the above three identification information is collected. Through that, IP traffic related users can be identified without ambiguity.

8.2.2 User identification of IP traffic with improved user model

In this section, we take three different user mapping methods into consideration to illustrate how user based IP traffic accounting can be achieved with the improved user model in distributed computing environments.

1. Mapping VO users to a single user account

   With this user mapping method, different VO users' applications are executed under the same user account. The PIDs of different applications can be used to distinguish the corresponding VO users running jobs in a host. Below an example is used to depict how the users of IP traffic are identified.

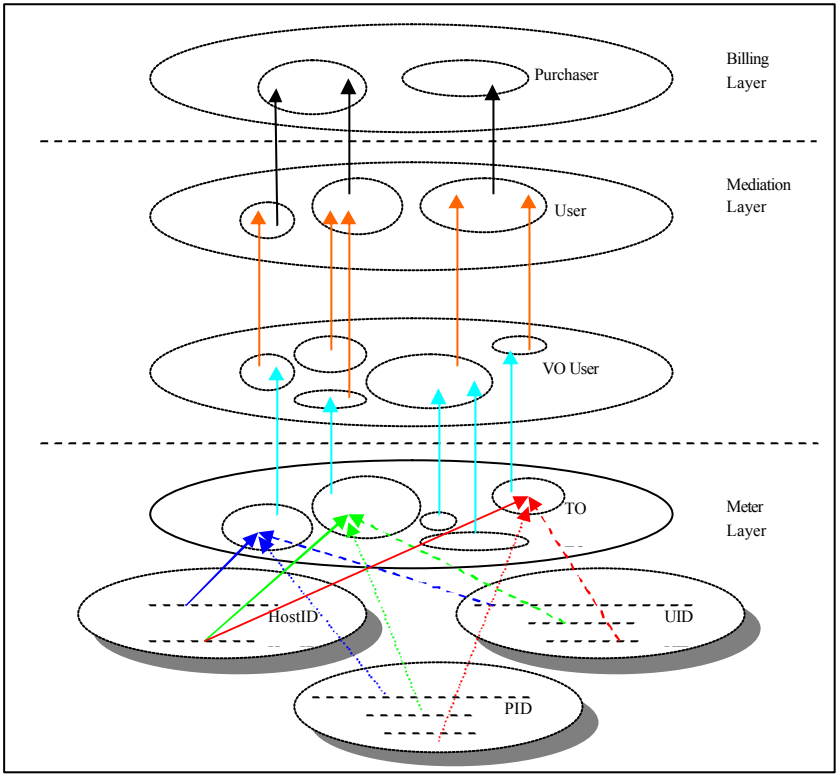


Figure 8.5 Improved user model for user based IP traffic accounting in distributed computing environments

Supposing student Alice and employee Bob of a university have registered accounts in a VO for the purpose of using distributed computing services. Alice registers a VO user account as alice@VO, and Bob registers as bob@VO. A DCN N owns an IP address 192.168.0.100 and provides a reserved user account "dummy" for executing distributed computing jobs.

Alice submits a job to the distributing resource manager, which allocates this job to node N for running this job, and a corresponding application with PID 1234 is. Bob submits a job which causes a corresponding application with PID 1678 to be started also in node N by the distributed resource manager. The user of IP traffic related to Alice's job and Bob's job can be identified as Table 8.1.

From the above example we can find that, despite the fact that different users' different distributed computing jobs are executed under the same UID "dummy" in the same node N, their jobs own different

PIDs. The 2-tuple <192.168.0.100, dummy> cannot uniquely identify the traffic originator of Alice and Bob related IP traffic. The PID combined with UID and HostID distinguishes the TOs. IP traffic related to PID 1234 (Alice's application) and PID 1678 (Bob's application) can be identified with the 4-tuple <192.168.0.100, dummy, 1234, 06/01/01-10:10:12> and <192.168.0.100, dummy, 1678, 06/01/01-10:11:37>, respectively. Through that, different VO users' IP traffic can be distinguished without ambiguity.

UID	HostID	PID	Time	TO	VO User	User	Purchaser
dummy	192.168.0.100	1234	06/01/01 - 10:10:12	<192.168.0.100, dummy, 1234, 06/01/01-10:10:12>	alice@VO	Alice@Home	Alice
dummy	192.168.0.100	1678	06/01/01 - 10:11:37	<192.168.0.100, dummy, 1678, 06/01/01-10:11:37>	bob@VO	Bob@CS	UNI

Table 8.1 An example of identifying user of IP traffic when VO users share a user account

In non-distributed computing environments, a similar situation occurs. Some applications or services require special privileges to be run in a host properly. Usually these applications or services run on behalf of the same user, e.g. root, but they belong to different users of the host. In this case, the 4-tuple TO can be used to distinguish between the users of the IP traffic related to different applications or services.

2. Assigning each VO user a unique local user account statically

With this user mapping method, a user account in a DCN belongs to a VO user permanently. The relationship between a user account and a VO user can be described as n:1 (n>=1). It means that a VO user may be mapped to different user accounts in DCNs. Therefore, the 2-tuple *<IP address, UID>* can be used to identify the traffic originator that can be correlated to a VO user without ambiguity. This case is the same as the one of normal multi-user systems discussed in chapter 4.

Still using the aforementioned example, but this time alice@VO is mapped to a user account "user1", whereas bob@VO is mapped to a user account "user2". In this case, the PID is not necessary for distinguishing VO users, since UIDs of different users are unique. Therefore, user identification of the IP traffic related to Alice and Bob in the node 192.168.0.100 can be achieved as Table 8.2.

UID	HostID	TO	VO User	User	Purchaser
user1	192.168.0.100	<192.168.0.100, user1 >	alice@VO	Alice@Home	Alice
user2	192.168.0.100	<192.168.0.100, user2>	bob@VO	Bob@CS	UNI

Table 8.2 Example of identifying user of IP traffic when VO users are mapped to unique user accounts statically

3. Assigning user accounts to VO users dynamically

With this user mapping method, a user account is not allocated to a VO user permanently. A user account belongs to a VO user only for a period of time, for example, during the lifetime of the VO user's application. After that, the user account may be allocated to another VO user. For this case a 3-tuple < *IP address, UID, Timestamp* > can be used to identify the traffic originator of IP traffic uniquely. In this 3-tuple, Timestamp identifies the time when the UID information is collected.

Considering the similar situation as in the above described examples, the DCN 192.168.0.100 provides a reserved user account pool for executing distributed computing applications. At first, the user Alice submits a job to the node and the job is executed under the user account " user1". After Alice's job is finished, the user account "user1" is reclaimed by the user account pool. Then the user Bob submits a job to the node, and the user account "user1" is chosen from the user account pool to be allocated to the VO user bob@VO again for executing Bob's job. With the 3-tuple the user identification of IP traffic in this case can be described as Table 8.3.

UID	HostID	Timestamp	TO	VO User	User	Purchaser
user1	192.168.0.100	06/01/01-10:10:12	<192.168.0.100, user1, 06/01/01-10:10:12>	alice@VO	Alice@Home	Alice
user1	192.168.0.100	06/01/01-10:11:37	<192.168.0.100, user1, 06/01/01-10:11:37>	bob@VO	Bob@CS	UNI

Table 8.3 Example of identifying user of IP traffic when user accounts pool is used

The improved user model provides the mechanism of distinguishing users of IP traffic related to applications executed under the same user account in DCN. Therefore, the improved user model can facilitate supplying

more accurate IP traffic accounting information in distributed computing environments.

8.3 User based IP traffic accounting solutions in distributed computing environments

Since in distributed computing environments the DCNs, which provide distributed computing services, may become multi-user systems, the IP traffic related to these nodes maybe generated by either normal local users or different mapped virtual users. In order to distinguish between the users of IP traffic in the DCNs, in the previously illustrated user based IP traffic accounting principles, schemes such as in-band scheme and out-of-band scheme can be applied to realize user based IP traffic accounting systems in distributed computing environments. The Multi-IP scheme can also be applied for user based IP traffic accounting in distributed computing environments through assigning different IP addresses to different VO users in a DCN.

Considering the new characteristics in distributed computing, some improvements should be made to the previous mechanisms and some new issues should be taken into consideration when integrating user based IP traffic accounting systems in distributed computing environments.

1. Accounting Agent

   One of the Accounting Agent's functions is to extract TO information and to identify the traffic originator of IP traffic. In distributed computing environments, the Accounting Agent is still required to be integrated into every measured host, i.e. the DCN, for the purpose of collecting the TO information and using it to identify TO of IP traffic. The Accounting Agent should gather TO information according to the user mapping mechanism used by the DCN. For example, the Accounting Agent may be required to gather not only IP address, User ID, but also PID of the application which is related to the IP traffic. The user identification methods are introduced in 8.2.2.

2. Correlation

   In a normal multi-user system, user accounts are related to users directly. Hence, from a user account in a host the corresponding user can easily be identified. In distributed computing environments, a user account is not directly related to one or more users, instead, it is directly related to one or more VO users. A user account may be allocated to different VO users at the same time or at different times. This makes the user correlation process more complex than that in non-distributed computing environments.

For the user mapping method that several VO users are mapped to a single user account, a TO & VO user Mapping Table is required to record the Host ID of the host which provides the distributed computing service, the User ID which is the user account allocated to different VO users, the PIDs of the executed jobs submitted by the VO user, and the Timestamp when the three attributes are collected. With the TO & VO User Mapping Table, the TO information recorded in the RDRs can be correlated to the corresponding VO users. In addition, the VO User & User Mapping Table can correlate the VO user to the corresponding user. Figure 8.6 illustrates the correlation process with the TO & VO user Mapping Table on the basis of the example in Table 8.1.

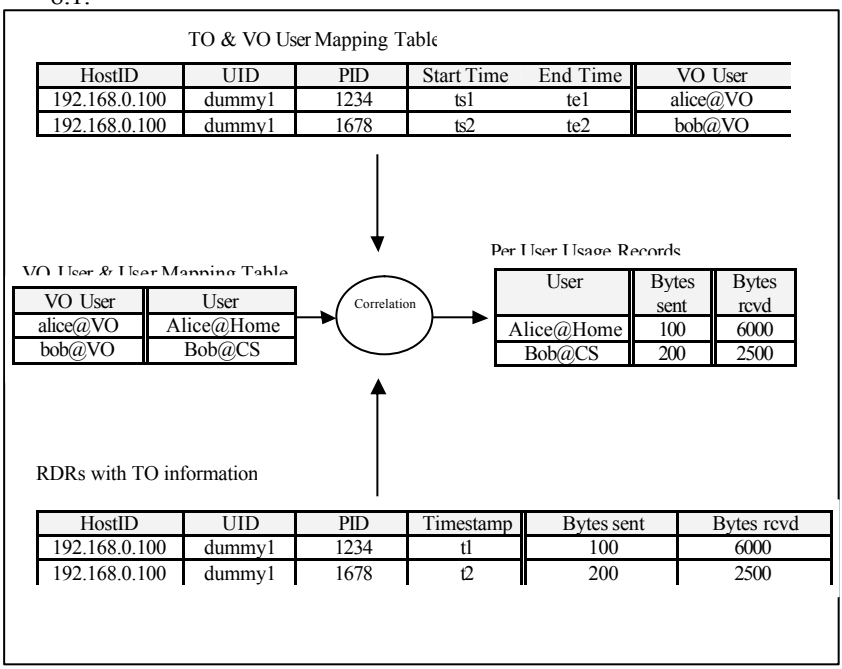


Figure 8.6 Example of correlating IP traffic meter information to the corresponding user in case several VO users are mapped to a single user account

For the dynamic user mapping method, the correlation process is similar to the single user account mapping method except that the PID information is not necessary to be collected as a part of the TO information. The Timestamp attribute must be recorded to indicate at which time the UID is recorded, so that the VO users sharing the same UID

at different time can be distinguished. For the static user mapping method, the correlation process is similar to the dynamic user mapping method except that the time information is not necessary to be recorded in the TO. Figure 8.7 gives an example of the user correlation process under the dynamic user mapping method.

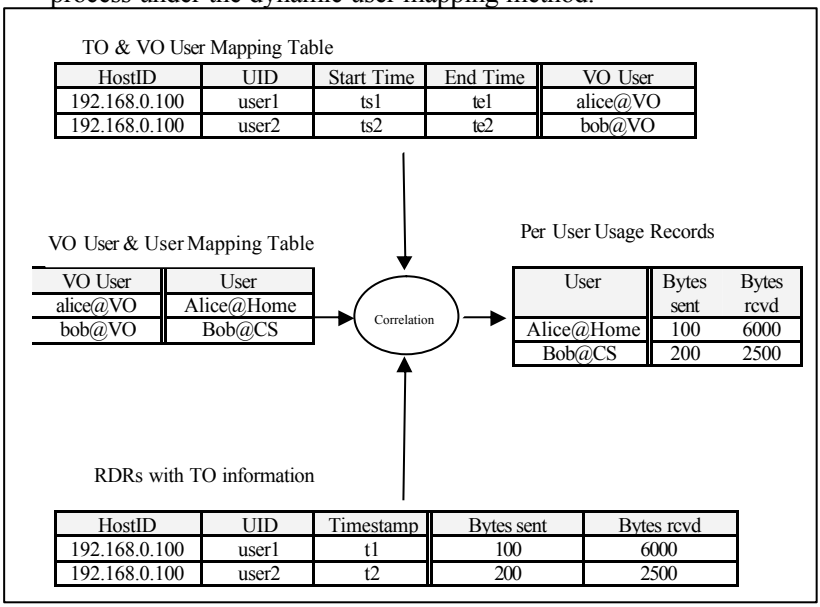


Figure 8.6 Example of correlating IP traffic meter information to the corresponding user in case the dynamic user mapping method is applied

8.4 User based network access control in distributed computing environments

With the user based IP traffic accounting mechanism, finer grained IP traffic accounting information can be provided. The user based IP traffic accounting information can facilitate more accurate network access control on the basis of VO users. Using the single user account mapping method as an example, although different VO users may have different network QoS such as bandwidth, IP traffic quota, etc., it is impossible to control these users' network usage according to their QoS requirements with the traditional IP traffic accounting mechanism which can only provide IP address based IP traffic accounting information. With the user based IP traffic accounting mechanism, the network usage information of every VO user's applications in a DCN can be gathered and calculated. When a VO user's IP traffic quota runs out, the Accounting Agent can block this VO

user's applications from accessing the network. Figure 8.7 illustrates the process of user based network access control with the help of the user based IP traffic accounting mechanism.

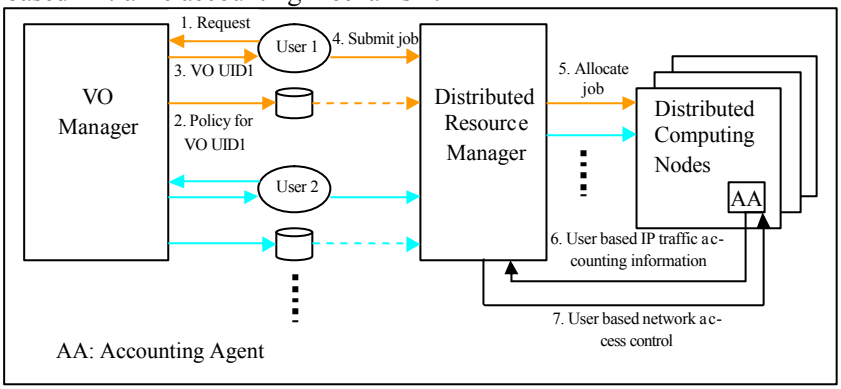


Figure 8.7 User based network access control in a distributed computing environment

1. When a user wants to use a distributed computing resource, she registers to the VO Manager to obtain a VO UID.
2. When this user's corresponding VO user is created, corresponding policies such as network access control, network resource quota, etc. are set for this VO user.
3. The user obtains a unique VO UID identifying the corresponding VO user.
4. After the user is authenticated with the corresponding VO user credential, she can submit jobs to the Distributed Resource Manager (DRM).
5. The DRM finds out a suitable Distributed Computing Node (DCN) from the resource pool, and allocates the user's job to the DCN. If this DCN owns enough resources to run more jobs, other users' jobs may also be allocated to this DCN. As in Figure 8.7, both User 1 and User 2 's jobs are allocated to the same DCN.
6. If the DCN utilizes the single user mapping method or the dynamic user mapping method for mapping VO users to local user accounts, a user based IP traffic accounting mechanism must be introduced to achieve user based network access control. With the user based IP traffic accounting mechanism, the Accounting Agent integrated in the DCN monitors the IP traffic and collects user based IP traffic accounting information, which is then sent to the meter in the DRM. The meter in the DRM aggregates and calculates the network resource consumption on the basis of VO users according to the user based IP

traffic accounting information. Then the calculated results are compared with the policy data.

7. When a user's resource consumption violates the policy, the DRM will inform the Accounting Agent in the DCN to perform corresponding operations according to the network access control policy to control this user's network access.

8.5 Summary

This chapter illustrates how the user based IP traffic accounting concept can be applied in the distributed computing environments to provide more accurate IP traffic accounting information, and how user based network access control can be achieved with the help of the user based IP traffic accounting mechanism.

The users in distributed computing environments may be different from the ones in non-distributed computing environments. A user needs to be registered as a VO user before she can access any distributed computing resource. In a DCN, a user account can stand for a local user, a VO user, or even different VO users. Moreover, a user account may be used by different VO users at the same time. Therefore, in this chapter the user model defined in chapter 3 is extended to accommodate the new characteristics of users in distributed computing environments. With this new user model, application identifier and timestamp may be required as a part of Traffic-Originators. Through that, different users' IP traffic can be distinguished in a distributed computing environment.

The in-band scheme or the out-of-band scheme can be applied for user based IP traffic accounting in distributed computing environments. The Accounting Agent and the correlation process of these two schemes should be improved to accommodate the new user model. With the user based IP traffic accounting mechanism, accurate user based network access control can be achieved in distributed computing environments.

Chapter 9 Conclusion and future work

In this dissertation, a user based IP traffic accounting model is proposed and discussed. For user based IP traffic accounting, a user model is necessary to distinguish the meanings and functions of a user in Meter Layer, Mediation Layer and Billing Layer. According to the user model suggested in this dissertation, a Traffic-Originator (TO) is used to specify who is responsible for generated IP traffic. A TO consists of a Host-identifier and a User Identifier. In a distributed computing environment, a TO may even be extended to contain HostID, UID, PID and Timestamp. In IP address based IP traffic accounting, a TO contains only a Host-Identifier, i.e. an IP address. Therefore traditional IP address based IP traffic accounting can be described as the process of collecting and processing network resource consumption information on the basis of the Host-Identifier, i.e. IP address, whereas in user based IP traffic accounting, IP address, User ID (User Identifier) and even process ID are required for identifying who generates the IP traffic. Hence, user based IP traffic accounting can be described as the process of collecting and processing network resource consumption information on the basis of the Traffic-Originator.

The key of user based IP traffic accounting is to identify the IP traffic with its corresponding TO information. The IP address of a TO can easily be obtained by checking the IP header of an IP packet. However, the User ID or the Process ID cannot be extracted from an IP packet directly. An IP packet's User ID and Process ID cannot be obtained outside the measured host, e.g. a multi-user host. Hence, the Accounting Agent mechanism is suggested to be integrated into the measured host for the purpose of gathering user information of IP traffic. This Accounting Agent can not only identify the user of IP traffic, but also store and transfer the user IP traffic relationship information to the meter. The Accounting Agent can also provide the access control ability to control users' network usage according to access control policies, which may be triggered on the basis of the user based charging information.

In this dissertation three different schemes, which can achieve the user based IP traffic accounting with the Accounting Agent mechanism, are proposed and discussed.

The in-band scheme utilizes the IP header to convey the user information of the corresponding IP packet. The Accounting Agent residing in the measured host intercepts IP packets passing through it, and then it identifies the users of these IP packets. After that, it inserts the user information into the IP packets. With this mechanism, a meter located in a key position of the network can intercept the IP packets tagged with user information,

and then it extracts not only statistic information such as bytes sent, bytes received, but also IP addresses and User IDs from the IP packets. Through that, the meter can generate accounting records with user information easily. The Accounting Agent can identify users of IP traffic on the basis of IP packets or IP traffic flows. With the IP packet based user identification, the Accounting Agent identifies the user of every IP packet, and then it integrates corresponding user information into every IP packet. With the IP traffic flow based user identification mechanism, the Accounting Agent identifies only the user of the first IP packet of a flow, and then the user information of this flow is transferred to the meter with this IP packet. Both Accounting Agent and meter build a Dynamic User Traffic Relationship Table (DUTRT) to record the user and flow relationship information. With that, the successive IP packets of the same flow can be identified by the meter directly by searching the DUTRT. The Accounting Agent is not required to identify the user of the successive IP packets of the same flow. This scheme requires the IP protocol to be extended to accommodate user information.

The out-of-band scheme is a contrast scheme to the in-band scheme. It also uses an Accounting Agent to intercept IP packets and identify the users of IP traffic. However, the user information of IP packets is not transferred together with the corresponding IP packets. Instead, the user information is transferred through a separated channel which is different from the corresponding IP packets' transmission. IP packets and their corresponding user information are transferred separately and independently. With the IP packet based user identification mechanism, the Accounting Agent generates accounting records with user information directly from IP packets and corresponding user information. Then it sends each IP packet's accounting record with user information to the meter at once. In this case, the accounting records are IP packet based and they are not stored in the measured host. The IP traffic flow based user identification mechanism requires the Accounting Agent to identify the user of the first IP packet of a flow, and then sends the user and IP traffic flow relationship information to the meter. The successive IP packets of the same flow will not be processed by the Accounting Agent. For this mechanism, a Dynamic User Traffic Relationship Table (DUTRT) is required to be maintained in both Accounting Agent and meter. The DUTRT is updated and synchronized by messages exchanged between Agent and meter. The Accounting Agent can also function as a standalone meter, which measures IP traffic and generates accounting records with user information in the measured host. This standalone mechanism can reduce the network traffic by compressing the accounting records before they are transported and the accounting records can be transferred in batch.

The Multi-IP scheme provides a different solution for identifying users of IP traffic. It assigns each user in a measured host a unique IP address. Through that, an IP address can be used to identify a user uniquely without ambiguity. With this scheme, traditional IP address based accounting techniques can be applied to achieve the goal of user based IP traffic accounting. In this scheme, the Accounting Agent is responsible for assigning IP addresses to users when they login or register, and reclaiming IP addresses from users when they logout or their accounts are deleted. This scheme requires a reserved IP address pool. In case IP address resource is scarce, the NAT mechanism may be required to guarantee that enough IP addresses are available for users.

In the NIPON project, a user based IP traffic accounting prototype system was developed in Solaris and Windows 2000 Terminal Server according to the standalone meter mechanism of the out-of-band scheme. The Agent was implemented with a kernel patch method in both Solaris and Windows 2000 Terminal Server systems. The traditional meter NeTraMet [Netr] was improved to support the user based IP traffic accounting function. The prototype system successfully provided IP traffic accounting information of different users in both Solaris and Windows 2000 Terminal Server systems. The prototype implementation shows that, although the Accounting Agent mechanism will cause performance decline in the measured host, if it is carefully designed and if the CPU of the measured host is powerful enough, low performance decline will result in the measured host.

The user based IP traffic accounting technique can not only be applied in multi-user systems, it is also meaningful that the user based IP traffic accounting technique is applied in distributed computing environments to provide finer granular IP traffic accounting information. A potential application is in Grid computing. In a distributed computing environment, a local user account may be shared by different remote users at the same time or at different times. In this case, user based IP traffic accounting can provide more accurate IP traffic accounting information. The possible policies are: assigning every user's job an IP address, assigning every job a user name, extending TO to <IP address, User ID, Process ID, start time, end time> and with standalone meter mechanism. Maybe the real time User ID correlation is required to map the User ID from a temporary ID to a fix ID.

With the rapid development of the Internet, the traditional time based flat rate charging and billing model will be replaced with IP traffic based or even service based charging and billing models in the future. Therefore, accurate accounting mechanisms must be applied to provide accurate and finer granular accounting information of IP traffic generation or network service consumption on the basis of users, instead of current coarser

grained IP address based accounting information. The user based IP traffic accounting mechanism suggested in this dissertation can facilitate solving the insufficiency problem of traditional IP address based IP traffic accounting. The user based IP traffic accounting technique is not only significant for academies or institutes which own a lot of multi-user systems but also meaningful for the increasing number of distributed computing applications. As discussed in this dissertation, despite that the necessity for user based IP traffic accounting is increasing, currently user based IP traffic accounting solutions are rare and no standard exists. More accurate and finer granular accounting on Internet usage will be a trend, which may require IP traffic, or even service and content based accounting mechanisms to provide technique supporting. In order to help the user based IP traffic accounting systems to be widely applied, standardization of the proposed techniques and methods in this dissertation should be made for user based IP traffic accounting. For example, considering that the IPv6 is still not widely applied, we expect that the user based IP traffic accounting mechanism will be taken into consideration in its design. With the user based IP traffic accounting technique, the network resource consumption can be measured accurately and cost allocation can be fair and reasonable.

Appendix A Bibliography

[A4RCH] IRTF Authentication Authorisation Accounting ARCHitecture Research Group (AAAARCH), http://www.irtf.org/charter?gtype=old-rg&group=aaaarch

[AAAWG] IETF Authentication, Authorization and Accounting (aaa) Working Group, http://www.ietf.org/html.charters/aaa-charter.html

[AbLi01] Bernard Aboba, Dave Lidyard, The Accounting Data Interchange Format (ADIF), IETF Internet Draft draft-aboba-roamops-adif-00.txt, 19 January 2001

[ACPZ99] J. Arkko, P. R. Calhoun, P. Patel, G. Zorn, DIAMETER Accounting Extension, IETF Internet Draft draft-calhoun-diameter-accounting-01.txt, http://ing.ctit.utwente.nl/WU 5/D5.1/Technology/diameter/draft-calhoun-diameter-accounting-01.txt

[Alif03] R. Alifieri et al., Managing dynamic user communities in a Grid of autonomous resources, Computing in high energy and nuclear physics, 24-28 March 2003

[Anal03] Analysys Research, Maximising Revenues from Broadband: new pricing structures for European operators,
http://research.analysys.com/default.asp?Mode=article&iLeftArticle=1238&m=&n=

[Anal06] Analysys Research, http://research.analysys.com/

[Apog] Apogee Networks Documentation, http://www.apogeenetworks.com

[Baue00] Volker Bauer, Analyse von Netwerk-Abrechnungs-Systemen bezü glich nutzerorientierter Datanerfassung, Diplomarbeit, Univerität Kaiserslautern, September 2000

[BCPS05] Steven M. Bellovin, David D. Clark, Adrian Perrig, Dawn Song (Eds), "Report of NSF Workshop on A Clean-Slate Design for the Next-Generation Secure Internet," GENI Design Document 05-05, July 2005, http://www.geni.net/GDD/GDD-05-05.pdf

[BeFH03] F. Berman, G. Fox, T. Hey, The Grid: past, present, future, in Grid Computing - Making the Global Infrastructure a Reality, pp 9-50, John Wiley & Sons, Ltd, 2003

[BEFK00] R. Butler, D. Engert, I. Foster, C. Kesselman, S. Tuecke, J. Volmer, V. Welch, A National-Scale Authentication Infrastructure, IEEE Computer, 33(12):60-66, 2000

[BeSy] Belle Systems Documentation, http://www.bellesystems.com

[BHMN04] D. Booth, H. Haas, F. McCabe, E. Newcomer et al., Web Services Architecture, W3C Working Group Note 11 February 2004, http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

[CCIT93] CCITT, The Directory --- overview of concepts, models and services, CCITT X.500 Series Recommendations, 1993

[CIA1] Computer Industry Almanac Inc., Worldwide Internet Users Top 1 Billion in 2005, http://www.c-i-a.com/pr0106.htm

[ClPB93] K. C. Claffy, G. C. Polyzos, H. Braun, Application of Sampling Methodologies to Network Traffic Characterization, Proceedings of ACM SIGCOMM'93, San Francisco, CA, USA, September 13 - 17, 1993

[ClSF06] David D. Clark, Scott Shenker (Chairs), Aaron Falk (Ed), "GENI Research Plan," GENI Design Document 06-28, Research Coordination Working Group, September 2006, http://www.geni.net/GDD/GDD-06-28.pdf

[CoGi98] I. Cozzani, S. Giordano, Traffic Sampling Methods for end-to-end QoS Evaluation in Large Heterogeneous Networks. Computer Networks and ISDN Systems, 30 (16-18), September 1998

[CoMi04] S. Convery and D. Miller, IPv6 and IPv4 Threat Comparison and Best Practice Evaluation (v1.0), Cisco Systems Technical Report, March 2004, http://www.cisco.com/security_services/ciag/documents/v6-v4-threats.pdf

[DaRa99] C. Darst, S. Ramanathan, "Measurement and management of Internet Services", Proceedings of the Sixth IFIP/IEEE International Symposium on Integrated Network Management, 24-28 May 1999

[Davi03] Joseph Davies, " Understanding IPv6" , Microsoft Press, 2003

[DuGr00] N. Duffield, M. Grossglauser, Trajectory Sampling for Direct Traffic Observation, Proceedings of ACM SIGCOMM 2000, Stockholm, Sweden, August 28 - September 1, 2000

[EdMV95] R. J. Edell, N. McKeown, P. P.Varaiya: " Billing Users and Pricing for TCP", IEEE Journal on selected areas in communications, Vol. 13. No. 7. September 1995

[EdVa99] R. Edell, P.Varaiya: " Providing Internet Access: What We Learn from INDEX" , IEEE Network, September/October 1999

[EkSP00] R. Ekstein, B. Sales, O. Paridaens, AAA Protocols: Comparison between RADIUS, DIAMETER and COPS, IETF AAA WG, Internet Draft, April 2000, draft-ekstein-aaa-protcomp-00.txt , http://ietfreport.isoc.org/idref/draft-ekstein-aaa-protcomp/

[ELDE96] D. H. J Epema, M. Livny, R. van Dantzig, X. Evers, and J. Pruyne, "A Worldwide Flock of Condors: Load Sharing among Workstation Clusters" , Journal on Future Generations of Computer Systems Volume 12, 1996.

[ETSI] European Technical Standards Institute, http://www.etsi.org/

[ETSI99] European Technical Standards Institute, Internet Protocol (IP) based networks: Parameters and mechanisms for charging, ETSI TR 101 734 V1.1.1, September 1999

[ETSI98] European Technical Standards Institute, Network Aspects: Considerations on Network Mechanisms for Charging and Revenue Accounting (ISBN 2-7437-2694-6), TR101 619 V1.1.1, November 1998.

[Exte] Extent Technologies Documentation, http//www.extent.com

[FaSP98] G. Fankhauser, B. Stiller, B. Plattner, Arrow: A Flexible Architecture for an Accounting and Charging Infrastructure in the Next Generation Internet, Netnomics: Economic Research and Electronic Networking, Vol. 1, No. 2, 1998

[FDLM01] C. Fraleigh, C. Diot, B. Lyles, S. Moon, et al, Design and deployment of a Passive Monitoring Infrastructure, In Proceedings of the Workshop on Passive and Active Measurements (PAM), April 2001

[FKNT02] I. Foster, C. Kesselman, J. Nick, S. Tuecke, The Physiology of the Grid: Open Grid Services Architecture for Distributed Systems Integration, presented at GGF4, February, 2002, http://www.globus.og/research/papers/ogsa.pdf

[FoKT01] The Anatomy of the Grid: Enabling Scalable Virtual Organizations, I. Foster, C. Kesselman, S. Tuecke, International Journal of Supercomputer Applications, 15(3), 2001.

[FoKe97] I. Foster, C. Kesselman, Globus: A Metacomputing Infrastructure Toolkit, International Journal of Supercomputing Applications, 11(2): 115-128, 1997

[FrKK96] A. O. Freier, P. Karlton, P. C. Kocher, The SSL Protocol Version 3.0, Netscape Communications Corporation, March 1996

[GDS1] Statistic data of Internet users, PCs, http://www.gdsourcing.ca/works/Almanac.htm#

[GLOP99] Internet Accounting – State of the Art
http://ing.ctit.utwente.nl/WU5/D5.1/index.html

[GrWu97] A. S. Grimshaw, W. A. Wulf, The Legion vision of a worldwide virtual computer, Communications of the ACM, 40(1), pp39-45, 1997

[HaAt01] T. J. Hacker, B. D. Athey, A Methodology for Account Management in Grid Computing Environments, Proceedings of the 2nd International Workshop on Grid Computing, November 2001

[Hage02] S. Hagen, IPv6 Essentials, O'Reilly & Associates, Inc., July 2002

[HaPa04] S. Hariri, M. Parashar, Tools and Environments for Parallel and Distributed Computing, John Wiley & Sons, 2004

[HaRa01] R. Hauck, I. Radisic, Service Oriented Application Management – Do Current Techniques Meet the Requirement?, In New Developments in Distributed Applications and Interoperable Systems, Proceedings of the 3rd IFIP International Working Conference (DAIS 2001), 295–304, Kluwer Academic Publishers, Krakow, Poland, September, 2001

[Hass03] J. Hassell, RADIUS, O'Reilly & Associates Inc, 2003

[Hobb05] R. H. Zakon, Internet Timeline, http://www.zakon.org/robert/internet/timeline/

[Hong03] J. W. Hong, " Internet Traffic Monitoring and Analysis: Methods and Applications", IM2003, March 2003

[IANA] IANA, http://www.iana.org/

[IANA1] IPv4 option numbers, http://www.iana.org/assignments/ip-parameters

[IANA2] Protocol numbers, http://www.iana.org/assignments/protocol-numbers

[ICANN] Internet Corporation For Assigned Names and Numbers, http://www.icann.org/

[IETF] Internet Engineering Task Force, http://www.ietf.org

[IPDR02] ipdr.org: "Network Data Management - Usage(NDM-U) for IP-Based Services", version 3.1.1, October 9, 2002

[IPDR03] ipdr.org, Service Specification, http://www.ipdr.org/service_specs/index.html, December 26, 2003

[IPDR06] IPDR organization, http://www.ipdr.org

[IPFIX] IETF IP Flow Information Export (ipfix) working group,

http://www.ietf.org/html.charters/ipfix-charter.html
[ISO] International Organization for Standardization, http://www.iso.org
[ISO74984] ISO Standard Document, Information processing systems – Open Systems Interconnection -- Basic Reference Model -- Part 4: Management framework, ISO/IEC 7498-4, 1989, http://standards.iso.org/ittf/PubliclyAvailableStandards/s014258_ISO_IEC_7498-4_1989(E).zip
[ISO8824] Information processing systems - Open Systems Interconnection, "Specification of Abstract Syntax Notation One (ASN.1)", International Organization for Standardization, International Standard 8824, December 1987
[ITU-T] International Telecommunications Union - Telecommunication Standardization Sector, http://www.itu.int/ITU-T/
[KaFF01] N. H. Kapadia, R.J. Figueiredo, J. Fortes, Enhancing the Scalability and Usability of Computational Grids via Logical User Accounts and Virtual File Systems, 10th Heterogeneous Computing Workshop, 2001
[KaFo99] N. Kapadia and J. Fortes, " PUNCH: An Architecture for Web-Enabled Wide-Area Network-Computing" , Cluster Computing: The Journal of Networks, Software Tools and Applications, September 1999.
[KeRD03] K. Keahey, M. Ripeanu, K. Doering, Dynamic Creation and Management of Runtime Environments in the Grid, Workshop on Designing and Building Web Services (GGF 9), October, 2003
[KLAD05] Frans Kaashoek, Barbara Liskov, David Andersen, Mike Dahlin, Carla Ellis, Steve Gribble, Anthony Joseph, Hank Levy, Andrew Myers, Jeff Mogul, Ion Stoica, Amin Vahdat , "Report of the NSF Workshop on Research Challenges in Distributed Computer Systems," GENI Design Document 05-06, December 2005, http://www.geni.net/GDD/GDD-05-06.pdf
[KSSW00] Martin Karsten, Jens Schmitt, Burkhard Stiller, and Lars Wolf. Charging for Packet-Switched Network Communication - Motivation and Overview. Computer Communications, 23(3):290–302, February 2000. ISSN 0140-3664
[KSWS98] Martin Karsten, Jens Schmitt, Lars Wolf, and Ralf Steinmetz. An Embedded Charging Approach for RSVP. In Proceedings of the 6th IEEE/IFIP International Workshop on Quality of Service (IWQoS'98), Napa, USA, pages 91–100. IEEE, May 1998. ISBN 0-7803-4482-0
[KSWS99] Martin Karsten, Jens Schmitt, Lars Wolf, and Ralf Steinmetz. Provider-Oriented Linear Price Calculation for Integrated Services. In Proceedings of the 7th IEEE/IFIP International Workshop on Quality of Service (IWQoS'99), London, UK, pages 174–183. IEEE, June 1999. ISBN 0-7803-5671-3
[Legi06] Legion: Worldwide Virtual Computer, http://legion.virginia.edu/, January 2006
[Libp] Libpcap: packet capture library, http://www.tcpdump.org/
[LuCo99] M. Lucas, O. Cohen: " Usage Collection and Analysis in an IP OSS", Billing World, March 1999

[M3I00] B. Stiller Ed., Charging and Accounting System (CAS) Design Version 1.01, The Market Managed Multi-service Internet (M3I) Consortium, July 2000

[MaSM03] V. Machiraju, A. Shai, A.van Moorsel, "Web Services management Network"

[McNa03] A. McNab, Grid-based access control for UNIX environments, Filesystems and Web Sites, Computing in High Energy and Nuclear Physics, 24-28 March 2003

[MeOV96] A. Menezes, P. van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996

[MNJH06] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, Host Identity Protocol, June 15 2006, http://tools.ietf.org/wg/hip/draft-ietf-hip-base/draft-ietf-hip-base-06.txt

[Muel00] P. Mueller, "Antrag fuer Nutzerbasietes IP Accounting – NIPON", 2000

[NARUS] Narus Documentation, http://www.narus.com

[NeEy] Neteye Documentation, http://www.neteyecorp.com

[Netf] NetFlow Documentation, http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html

[Netr] NeTraMet, http://www2.auckland.ac.nz/net/NeTraMet/

[Netm] Microsoft NetMeeting, http://www.microsoft.com/windows/netmeeting/

[Netp] NetPerf tools, http://www.netperf.org/netperf/NetperfPage.html

[NIPO00] NIPON Project, NIPON: Nutzerbasiertes IP accounting, http://www.icsy.de/forschung/nipon/index

[NIPO03] ICSY AG, NIPON Project Report, October 2003

[Odyl03] A. M. Odlyzko, Internet traffic growth: Sources and implications, Optical Transmission Systems and Equipment for WDM Networking II, B. B. Dingel, W. Weiershausen, A. K. Dutta, and K.-I. Sato, eds., Proc. SPIE, vol. 5247, 2003, pp. 1-15

[OpSy] Opencon Systems Documentation, http://www.opencon.com

[PBSP01] A.Pras, B. v. Beijnum, R. Sprenkels, R. Parhonyi, Internet Accounting, IEEE Communications Magazine, May 2001

[PHGG04] D. H. Pezaros, D. Hutchison, F. J. Garcia, R. D. Gardner, J. S. Sventek, In-line Service Measurements: An IPv6-based Framework for Traffic Evaluation and Network operations, Proceedings of the IEEE/IFIP Network Operations and Management Symposium, Seoul, Korea, April 2004

[Pion06] Pioneer Consulting LLC, Global Sales of IP Convergence Equipment to Reach 10.6 Billion (USD) by 2005, http://www.pioneerconsulting.com/pressrelease.php3?report=30

[Pool05] Pool Accounts patch for Globus, http://www.gridpp.ac.uk/gridmapdir/

[RFC791] J. Postel, Internet Protocol, September 1981

[RFC792] J. Postel, Internet Control Message Protocol, September 1981

[RFC1155] M. T. Rose, K. McCloghrie, Structure and identification of management information for TCP/IP-based internets, May 1990

[RFC1157] J. D. Case, M. Fedor, M. L. Schoffstall, J. Davin, Simple Network Management Protocol (SNMP), May 1990

[RFC1191] J.C. Mogul, S.E. Deering, Path MTU discovery, November 1990
[RFC1212] M. T. Rose, K. McCloghrie, Concise MIB definitions, Mar 1991
[RFC1215] M. T. Rose, Convention for defining traps for use with the SNMP, Mar 1991
[RFC1272] C. Mills, D. Hirsh, G.R. Ruth, Internet Accounting: Background, Nov, 1991
[RFC1413] M. St. Johns, Identification Protocol, February 1993
[RFC1631] K. Egevang, P. Francis, The IP Network Address Translator (NAT), May 1994
[RFC1671] B. Carpenter, IPng White Paper on Transition and Other Considerations, August 1994
[RFC1672] N. Brownlee, Accounting Requirements for IPng, August 1994
[RFC1812] F. Baker, Ed., Requirements for IP Version 4 Routers, June 1995
[RFC1825] R. Atkinson, Security Architecture for the Internet Protocol, August 1995
[RFC1918] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, E. Lear, Address Allocation for Private Internets, February 1996
[RFC2058] C. Rigney, A. Rubens, W. Simpson, S. Willens, Remote Authentication Dial In User Service (RADIUS), January 1997
[RFC2059] C. Rigney, RADIUS Accounting, January 1997
[RFC2063] N. Brownlee, C. Mills, G. Ruth, Traffic Flow Measurement: Architecture, January 1997
[RFC2123] N. Brownlee, Traffic Flow Measurement: Experiences with NeTraMet, March 1997
[RFC2138] C. Rigney, A. Rubens, W. Simpson, S. Willens, Remote Authentication Dial In User Service (RADIUS), April 1997
[RFC2139] C. Rigney, RADIUS Accounting, April 1997
[RFC2212] S. Shenker, C. Partridge, R. Guerin, Specification of Guaranteed Quality of Service, September 1997
[RFC2246] T. Dierks, C. Allen, The TLS Protocol Version 1.0, January 1999
[RFC2401] S. Kent, R. Atkinson, Security Architecture for the Internet Protocol, November 1998
[RFC2402] S. Kent, R. Atkinson, IP Authentication Header, November 1998
[RFC2406] S. Kent, R. Atkinson, IP Encapsulating Security Payload (ESP), November 1998
[RFC2459] R. Housley, W. Ford, W. Polk, D. Solo, Internet X.509 Public Key Infrastructure Certificate and CRL Profile, RFC 2459, January 1999.
[RFC2460] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, December 1998
[RFC2462] S. Thomson, T. Narten, IPv6 Stateless Address Autoconfiguration, December 1998
[RFC2463] A. Conta, S. Deering, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, December 1998
[RFC2486] B. Aboba, M. Beadles, The Network Access Identifier. January 1999
[RFC2512] K. McCloghrie, J. Heinanen, W. Greene, A. Prasad, Accounting Information for ATM Networks, February 1999

[RFC2578] K. McCloghrie, D. Perkins, J. Schoenwaelder, Structure of Management Information Version 2 (SMIv2), April 1999
[RFC2579] K. McCloghrie, D. Perkins, J. Schoenwaelder, Textual Conventions for SMIv2, April 1999
[RFC2580] K. McCloghrie, D. Perkins, J. Schoenwaelder, Conformance Statements for SMIv2, April 1999
[RFC2720] N. Brownlee, Traffic Flow Measurement: Meter MIB, October 1999
[RFC2722] N. Brownlee, C. Mills, G. Ruth, Traffic Flow Measurement: Architecture, October 1999
[RFC2723] N. Brownlee, SRL: A Language for Describing Traffic Flows and Specifying Actions for Flow Groups, October 1999
[RFC2724] S. Handelman, S. Stibler, N. Brownlee, G. Ruth, RTFM: New Attributes for Traffic Flow Measurement, October 1999
[RFC2865] C. Rigney, S. Willens, A. Rubens, W. Simpson, Remote Authentication Dial In User Service (RADIUS), June 2000
[RFC2866] C. Rigney, RADIUS Accounting, June 2000
[RFC2881] D. Mitton, M. Beadles, Network Access Server Requirements Next Generation (NASREQNG) NAS Model, July 2000
[RFC2882] D. Mitton, Network Access Servers Requirements: Extended RADIUS Practices,
July 2000
[RFC2924] N. Brownlee, A. Blount, Accounting Attributes and Record Formats, September 2000
[RFC2960] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, V. Paxson, Stream Control Transmission Protocol, October 2000
[RFC2975] B. Aboba, J. Arkko, D. Harrington, Introduction to Accounting Management, October 2000
[RFC2989] B. Aboba, P. Calhoun, S. Glass, T. Hiller, P. McCann, H. Shiino, G. Zorn, G. Dommety, C. Perkins, B. Patil, D. Mitton, S. Manning, M. Beadles, P. Walsh, X. Chen, S. Sivalingham, A. Hameed, M. Munson, S. Jacobs, B. Lim, B. Hirschman, R. Hsu, Y. Xu, E. Campbell, S. Baba, E. Jaques, Criteria for Evaluating AAA Protocols for Network Access, November 2000
[RFC3022] P. Srisuresh, K. Egevang, Traditional IP Network Address Translator (Traditional NAT), January 2001
[RFC3127] D. Mitton, M. St.Johns, S. Barkley, D. Nelson, B. Patil, M. Stevens, B. Wolff, Authentication, Authorization, and Accounting: Protocol Evaluation, June 2001
[RFC3162] B. Aboba, G. Zorn, D. Mitton, RADIUS and IPv6, August 2001
[RFC3280] R. Housley, W. Polk, W. Ford, D. Solo, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC 3280 April 2002
[RFC3315] R. Droms, Ed., J. Bound, B. Volz, T. Lemon, C. Perkins, M. Carney, Dynamic Host Configuration Protocol for IPv6 (DHCPv6), July 2003
[RFC3334] T. Zseby, S. Zander, C. Carle, Policy-Based Accounting, October 2002

[RFC3344] C. Perkins, Ed., IP Mobility Support for IPv4, August 2002
[RFC3372] A. Vemuri, J. Peterson, "Session Initiation Protocol for Telephones (SIP-T): Context and Architecture", RFC 3372, IETF, September 2002
[RFC3411] D. Harrington, R. Presuhn, B. Wijnen, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks, December 2002
[RFC3412] J. Case, D. Harrington, R. Presuhn, B. Wijnen, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), December 2002
[RFC3413] D. Levi, P. Meyer, B. Stewart, Simple Network Management Protocol (SNMP) Applications, December 2002
[RFC3414] U. Blumenthal, B. Wijnen, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), December 2002
[RFC3415] B. Wijnen, R. Presuhn, K. McCloghrie, View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), December 2002
[RFC3416] R. Presuhn, Ed., Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP), December 2002
[RFC3417] R. Presuhn, Ed., Transport Mappings for the Simple Network Management Protocol (SNMP), December 2002
[RFC3418] R. Presuhn, Ed., Management Information Base (MIB) for the Simple Network Management Protocol (SNMP), December 2002
[RFC3419] M. Daniele, J. Schoenwaelder, Textual Conventions for Transport Addresses, December 2002
[RFC3423] K. Zhang, E. Elkin, XACCT's Common Reliable Accounting for Network Element (CRANE) Protocol Specification Version 1.0, November 2002
[RFC3430] J. Schoenwaelder, Simple Network Management Protocol Over Transmission Control Protocol Transport Mapping, December 2002
[RFC3539] B. Aboba, J. Wood, Authentication, Authorization and Accounting (AAA) Transport Profile, June 2003
[RFC3546] S. Blake-Wilson, M. Nystrom, D. Hopwood, J. Mikkelsen, T. Wright, Transport Layer Security (TLS) Extensions, June 2003
[RFC3584] R. Frye, D. Levi, S. Routhier, B. Wijnen, Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework, August 2003
[RFC3588] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko, Diameter Base Protocol, September 2003
[RFC3775] C. Perkins, J. Arkko, Mobility Support in IPv6, D. Johnson, June 2004
[RFC3917] J. Quittek, T. Zseby, B. Claise, S. Zander, Requirements for IP Flow Information Export (IPFIX), October 2004
[RFC4282] B. Aboba, M. Beadles, J. Arkko, P. Eronen, The Network Access Identifier, December 2005
[RFC4301] S. Kent, K. Seo, Security Architecture for the Internet Protocol, December 2005

[RFC4423] R. Moskowitz, P. Nikander, Host Identity Protocol (HIP) Architecture, May 2006
[RTFM] IETF Real-time Traffic Flow Measurement Working Group, http://www.auckland.ac.nz/net/Internet/rtfm/
[RRRN04] White Paper: Mediation, Rating and Billing in an IP Services Enviroment - architecture and Approach
[SaMW01] A. Sahai, V. Machiraju, K. Wurster, " Monitoring and Controlling Internet based E-Services"
[SETI] SETI@home project, http://setiathome.ssl.berkeley.edu/
[SFPW98] B. Stiller, G. Fankhauser, B. Plattner, N. Weiler, Pre-study on Customer Care, Accounting, Charging, Billing, and Pricing, TIK, ETH Zurich, February 1998
[SiSc00]K. Singh, H.Schulzrinne, "Interworking Between SIP/SDP and H.323", Proceedings of the 1st IP-Telephony Workshop (IPTel'2000), April 2000
[Smey01] R. Smeyers, UserIPAcct - a program to do per user ip accounting, http://ramses.smeyers.be/homepage/useripacct/
[TaBK03] V. Talwar, S. Basu, R. Kumar, An Environment for Enabling Interactive Grids, IEEE International Symposium on High Performance Distributed Computing (HPDC-12), 22-24 June 2003
[Tane03] Andrew S. Tanenbaum, Computer Networks, Fourth Edition, Prentice Hall, March 17, 2003
[TDI] Transport Driver Interface, http://www.microsoft.com/resources/documentation/Windows/2000/server/reskit/en-us/Default.asp?url=/resources/documentation/windows/2000/server/reskit/en-us/cnet/cnad_arc_GHYW.asp
[Thom00] S. A. Thomas, SSL and TLS Essentials: Securing the Web, John Wiley & Sons, Inc., 2000
[TIA05] Telecommunications Industry Association (TIA), International Telecom Market to Reach Over $2T by 2008, July 15 2005, http://electronics.ihs.com/news-05Q3/tia-telecommunications-market.jsp
[TIPHON] ESTI Telecommunications and Internet Protocol Harmonization Over Networks, http://portal.etsi.org/tb/closed_tb/tiphon.asp
[TMPE04] V. Tosic, W. Ma, B. Pagurek, B. Esfandiari, "Web Service Offerings Infrastructure (WSOI) – A Management Infrastructure for XML Web Services"
[Winp] Winpcap: the packet capture library for Windows, http://winpcap.polito.it/default.htm
[XACCT] XACCT Documentation, http://www.xacct.com
[Zseb03] T. Zseby, Stratification Strategies for Sampling-based Non-intrusive Measurements of One-way Delay, PAM2003, March 30, 2003
[Zhan01] G. Zhang, Comparison and Analysis of IP Billing Technologies, internal report, Nov. 2001
[ZhHM05] G. Zhang, M. Hillenbrand, P. Mueller, Facilitating the Interoperability among Different VoIP Protocols with VoIP Web Services, 1. International

Conference on Distributed Frameworks for Multimedia Applications DFMA'2005, Besancon France, Februar 2005

[ZhRM02] G. Zhang, B. Reuther, P. Mueller, "Distributed Agent Method for User based IP traffic accounting", 7th. CaberNet Radicals Workshop, 13-16 October 2002

[ZhRM03] G. Zhang, B. Reuther, P. Mueller, "User Oriented IP Accounting in Multi-user Systems", The 8th IFIP/IEEE International Symposium on Integrated Network Management, 24-28 March 2003

[ZhRM05] G. Zhang, B. Reuther, P. Mueller, A Model for User Based Traffic Accounting, 31st EUROMICRO Conference on Software Engineering and Advanced Applications, 2nd September, 2005

Appendix B – Performance Test Results

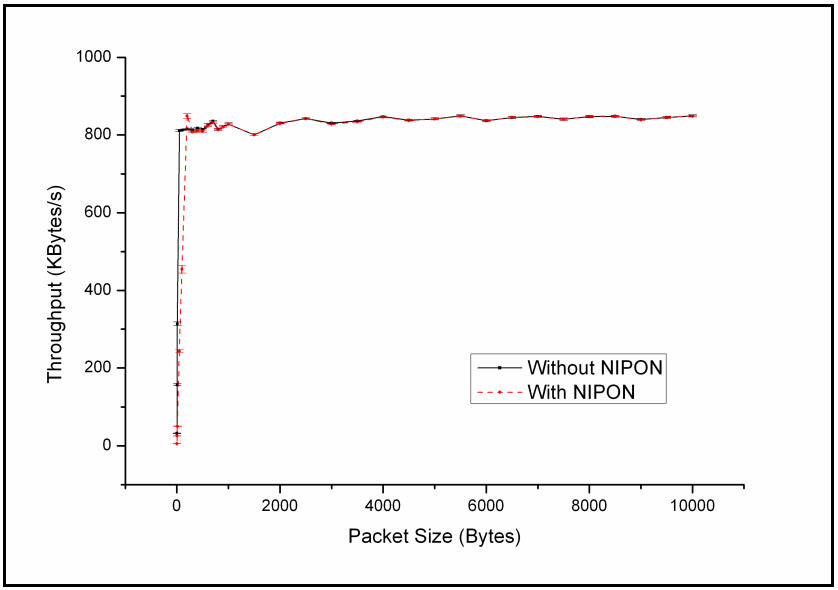


Figure B1 TCP send in Solaris

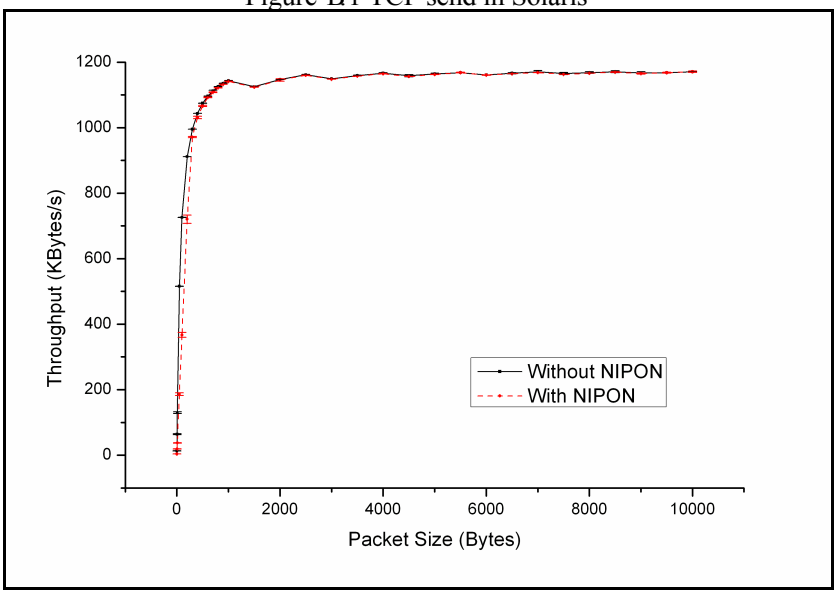


Figure B2 UDP send in Solaris

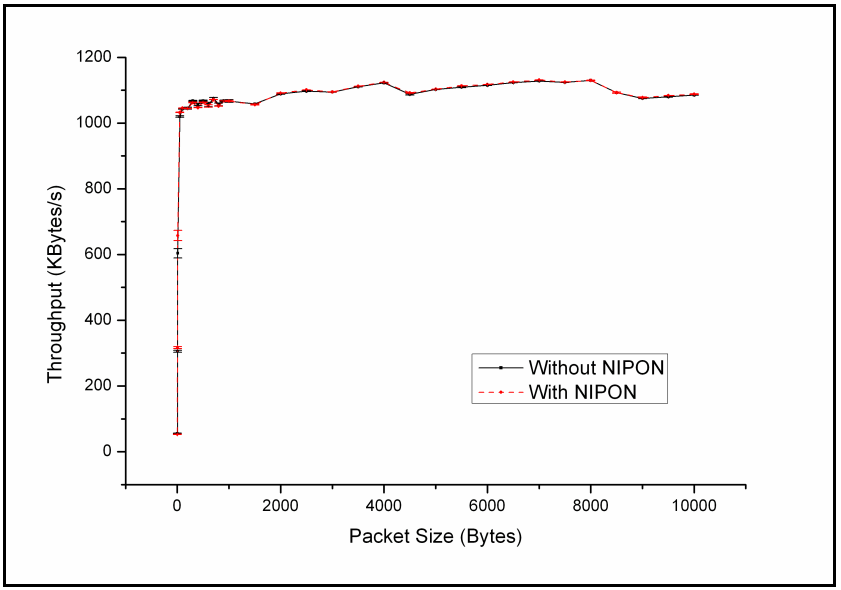


Figure B.3 TCP receive in Solaris

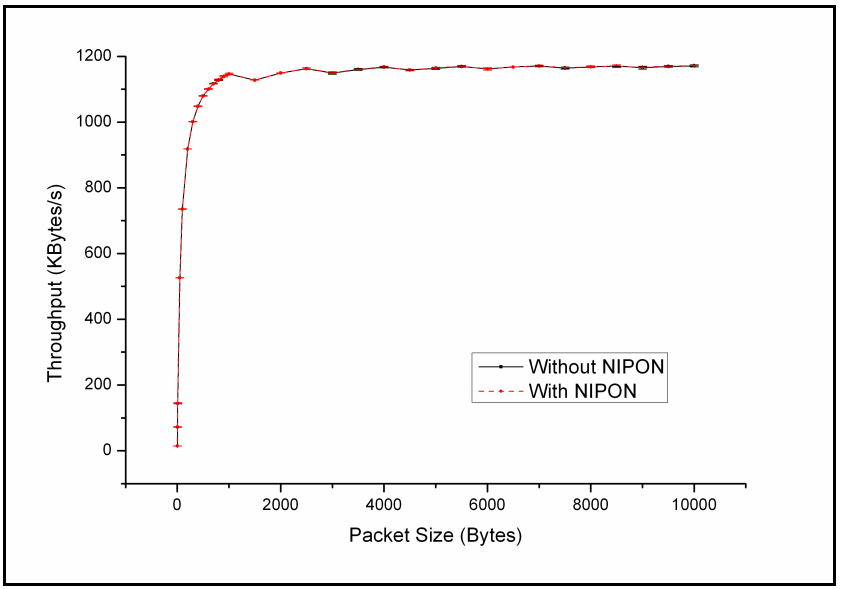


Figure B.4 UDP receive in Solaris

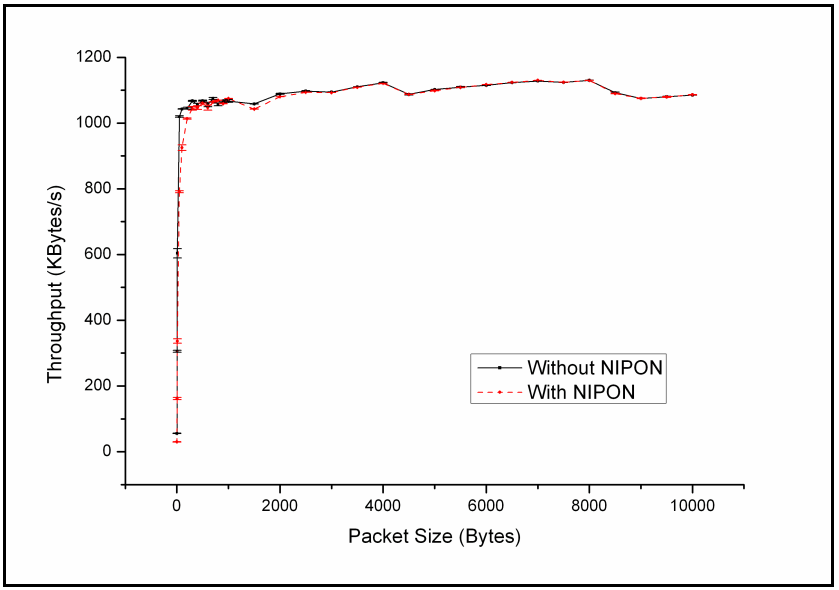


Figure B.5 TCP send in Windows

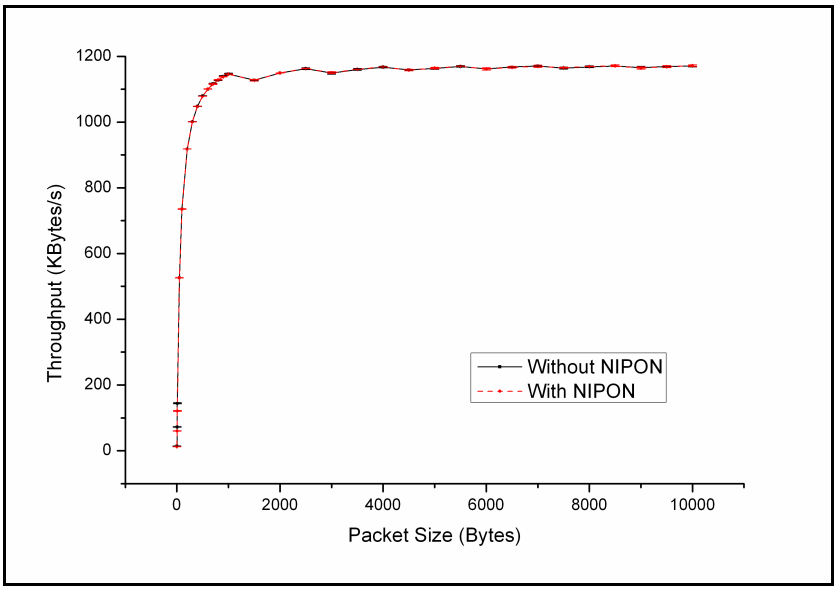


Figure B.6 UDP send in Windows

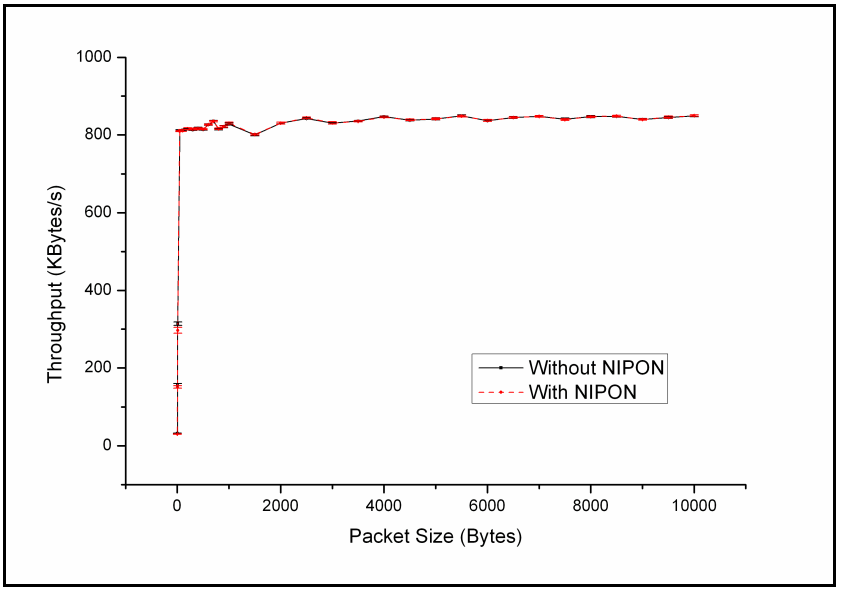


Figure B.7 TCP receive in Windows

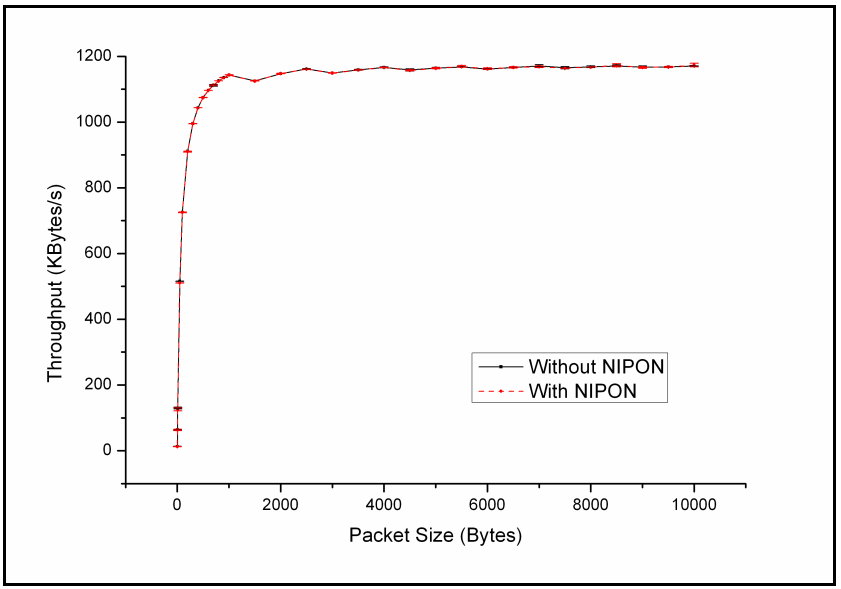


Figure B.8 UDP receive in Windows

Appendix C Glossary

ASP----------Active Server Page
BSS----------Business Support System
DCN----------Distributed Computing Node
DN----------Distinguished Name
DNS----------Domain Name Service
DHCP----------Dynamic Host Configuration Protocol
IETF----------Internet Engineering Task Force
IP----------Internet Protocol
IPDR----------Internet Protocol Data Record
ISDN----------Integrated Services Digital Network
ISP----------Internet Service Provider
LDAP----------Lightweight Directory Access Protocol
MIB----------Management Information Base
NE----------Network Element
OSS----------Operation Support System
QoS----------Quality of Service
RADIUS----------Remote Access Dial-In Usage Server
RAS----------Remote Access Server
RDR----------Raw Data Record
RMON----------Remote Network Monitoring
RSVP----------Resource ReSerVation Protocol
SMTP----------Simple Mail Transfer Protocol
SNMP----------Simple Network Management Protocol
TDI----------Transport Driver Interface
VO----------Virtual Organization
VoIP----------Voice over IP
VPN----------Virtual Private Network
XML---------- eXtensible Markup Language

LEBENSLAUF

Name:	Ge Zhang
Geburtsdatum:	04. November 1966
Geburtsort:	Hunan, V. R. China
Familienstand:	Verheiratet
Staatsangehörigkeit:	V. R. China

Ausbildung

1973 - 1978 Hong She Grundschule in Changsha.

1978 – 1984 Shi Da Hu Zhong Gymnasium in Changsha, Abschluß : Abitur.

1984 - 1988 Studium der Informatik an der Universität Hunan, Abschluß : Bachelor.

1990 - 1993 Studium der Informatik an der National Universität für Abwehr Technologie, Abschluß : Master.

Beruf

1988 - 1990 wissenschaftlicher Mitarbeiter für Information System Management an der Hunan Medizin Test Zentrum.

1993 - 2000 wissenschaftlicher Forschungsmitarbeiter an Taiji Computer Company, Beijing, V. R. China.

2000 – 2007 wissenschaftlicher Mitarbeiter an der Universität Kaiserslautern im Fachbereich Informatik AG ICSY.