Associating Network Flows with User and Application Information

Ralf Ackermann [1], Utz Roedig [1], Michael Zink [1], Carsten Griwodz [1], Ralf Steinmetz [1,2]

1 - Darmstadt University of Technology - Industrial Process and System Communications (KOM)
Merckstr. 25 - 64283 Darmstadt, Germany

2 - German National Research Center for Information Technology - GMD IPSI
Dolivo-Str.15 - 64293 Darmstadt, Germany

{Ralf.Ackermann, Utz.Roedig, Michael.Zink, Carsten.Griwodz, Ralf.Steinmetz}@KOM.tu-darmstadt.de

ABSTRACT

The concept of authenticating users e.g. by means of a login process is very well established and there is no doubt that it is absolutely necessary and helpful in a multiuser environment. Unfortunately specific information about a user originating a data stream or receiving it, is often no longer available at the traversed network nodes. This applies to the even more specific question of what application is used as well. Routers, gateways or firewalls usually have to base their classification of data on IP header inspection or have to try to extract information from the packets payload.

We present an approach that works transparently and allows to associate user and application specific information with IP data streams by only slightly modifying components of the operating system environment and infrastructure components. On top of this framework we show usage scenarios for dedicatedly placing copyright information in media content and for an enhancement of the interoperation with the security infrastructure.

Keywords

Security, Network Traffic Marking, Watermarking, Firewalls

1. INTRODUCTION AND MOTIVATION

According to the Internet communication model, only the header information of a specific layer should be used by the network nodes to route, filter, interpret or otherwise process data. In reality though, the strict layered concept is weakened at many points and information normally assigned to different layers is used to process packets. An example is the implementation of QoS routing functions in network nodes. Information of the application layer is necessary in the router devices, which normally should only know about the network layer, to fulfill their tasks. A dedicated information, which should not only be available in its original (application) layer is the one describing the originator or receiver of a data stream. Usually only the application layer should be aware of users, but many processes within a network also benefit from this information. Example scenarios where such a knowledge is very helpful include authentication at firewall systems, logging, admission control, billing but also the placement of copyright information.

In this paper we will describe an approach to map additional information to network streams and show its implications.

2. REQUIREMENTS

The access to user and/or application relevant information at traversed network nodes is helpful for a number of scenarios. There is an existing classification of information types and several approaches to obtain it.

2.1 Availability of User Information

Some network nodes (e.g. watermarking gateways, firewalls) have to map knowledge about user identities to the data flows, to fulfill their tasks. A network node has basically two possibilities for doing that. The node can interrupt the communication path at the application layer and force the user to identify himself (explicitly, actively concerning the user part). Alternatively, the node can try to extract the information - if (still) present - by analyzing the application layer part of the communication data (implicitly, passively concerning the user part). Both methods have drawbacks:

- The active or passive gathering of user information is not always possible.
- The passive information retrieval is costly and may result in a reduction of performance. Additionally it involves a considerable implementation effort.

To avoid these drawbacks, out of band signalling can be used. The communication partners can signal user relevant information in advance before the communication data itself is sent. For using this method, a standardized protocol would be applicable and the drawbacks mentioned above could be avoided. Other problems occur though:

- Every endsystem and all the nodes that need access to the user information have to implement and support the signalling protocol.

- The network nodes have to remember the mappings between user information and data flows. This increases the complexity of such a network node.

The method that we will describe in this paper is to add the necessary information directly to a network data flow.

2.2 Availability of Application Information

Certain network nodes (e.g. QoS-enabled routers, firewalls) need information about the application that is generating a data stream to process the data. This information can usually be gained by analyzing the transport layer header (TCP/UDP header interpreting the port fields). For some scenarios it is necessary to consider that a logical session between two endpoints may consist of several flows[1]. In this case the first flow normally uses static ports and a traversed node can extract the information about the application type from the transport layer header. Subsequent flows are then negotiated dynamically by exchanging signaling data on the first channel. In many cases it is necessary to treat all the flows that an application uses uniformly as a single entity (for example a firewall wants to authenticate the involved parties of a whole session and needs to know about the dependencies of flows).

3. MARKING OF NETWORK PACKETS

3.1 Basic Approach

Our basic approach which is shown in Figure 1 assumes the deployment and use of a marking procedure for network data streams at dedicated network nodes (usually endsystems but also gateways) of an administrative domain which is under our explicit control (since modifications have to be done at least for one communication partner). Whenever a user is authenticated to the network node, it is possible to e.g. mark the data he is originating with the user-id that processes are identified with. The network nodes that are passed by the data packets (e.g. gateways or firewalls which form a dedicated crossing point for traffic entering or leaving the domain) make use of the information. It must be mentioned that this is also applicable for data streams that flows in the opposite direction (originating from sources that are outside) but can be associated with an original data stream (e.g. answers to retrieval requests/bi-directional TCP flows).

We will have to consider both cases - either that the user has a strong interest in supplying and passing this information or does at least not actively suppress it (e.g. because it allows for a better service or fair billing for him) or that we have to enforce the use of the mechanisms and prevent participants from mis-using or faking it.

[1]A flow is a single data stream (channel), identified by a tuple of characteristic values (source address, source port, destination address, destination port, protocol number). A session describes the association of multiple flows that together form an application's data stream.

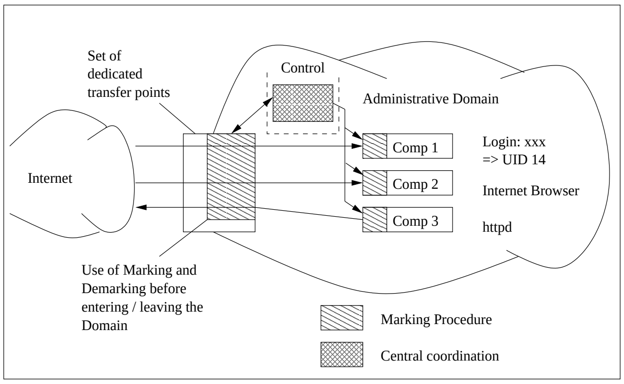

Figure 1: Basic description of the approach and concerned components

3.2 Placement of the marking information

There are a number of possible places at which the information can be placed for transmission. The approaches and their advantages and disadvantages are described in the following. By including the information either in the layer 2 or layer 3 header, a fast access to the information at the intermediate nodes (router, proxy, firewall) is possible. In comparison, storing the additional data in the payload is more costly and involves additional analysis of each packet in the network. Both approaches allow the insertion and removal at either the endsystems or intermediate nodes depending on the desired operating mode.

3.2.1 Placement as part of the MAC-Header

Placing the information in an additional field of the MAC header forms a very general approach. A technique like that is e.g. used for Label Switching [1]. This approach has the advantage that not only IP but also other layer 3 protocols (IPX, ...) can easily be supported. Additionally it has proven to perform well in terms of packet processing speed in routers or switches, since only the MAC header must be examined to gather the desired information.

3.2.2 Placement as part of the IP header

An alternative place to insert the information is the IP header. Since IPv6 is fully standardized and already used in some test beds we give a proposal for both of the IP versions IPv4 and IPv6.

- In IPv4 the information can be inserted after the IP header and before an upper layer protocol (e.g. TCP, UDP,...). [8] proposes how this is done for the authentication header and is applicable for additional extension headers as well.

- In IPv6 the information can be inserted in the packet as an extension header, like the ones already defined in [5].

The interoperability with standards-compliant existing network nodes that lack the proposed extensions is ensured, they are required to ignore the additional information.

4. IMPLEMENTATION CONSIDERATIONS

4.1 Marking Procedure

We distinguish between an endsystem- and an infrastructure- (e.g. by means of firewalls/gateways) -based packet marking approach.

4.1.1 Endsystem-based - Unix

To support the insertion of marks in a general way, the insertion should be transparent for the user. A convenient place for doing that is as part of a modification of the network stack or by passing all traffic through a dedicated (network) tunneling device. Implementation alternatives for different operating systems differ both in their granularity as well as in the way and necessary permissions for performing them.

Depending of the kind of the system and the availability of sources we could decide to modify and replace the kernel. Since different machines usually use different kernels, this approach is not very flexible and involves a remarkable additional effort. Therefore placing the functionality in a shared library that is pre-loaded in order to replace the systems libsocket whenever an application is started (or to replace the libsocket in general) is considered to be more convenient. Alternatively the use of a dedicated stream module [10] that can be dynamically pushed into the communication stack is an option for systems where this is supported, e.g. when using Solaris.

4.1.2 Endsystem-based - Windows

WinSock, the Microsoft Windows networking API, consists of a set of layers called "service providers". It is possible to install new service providers in the form of a Dynamic Link Library (DLL) between any two existing layers in the Winsock stack [6]. All programs using the Winsock API invoke the new service provider automatically then. This mechanism allows the creation of a new service provider, that is responsible for performing the necessary routines to put user and application information into a layer 2 or layer 3 header.

4.1.3 Infrastructure-based

In some cases, several or even all hosts may not be extended (or even only extendable) as described above. For a limited set of tasks the packet marking process can be handled by a (set of) marking gateway(s). Figure 2 shows a possible scenario. A marking gateway can be implemented in two different ways. First, the gateway can use active or passive information gathering (as described above) to get user specific information. Then this information has to be added to the flows before they leave the gateway. All the mentioned marking techniques can be used for this purpose.

The second method is to summarize the subnets, which include not marking capable hosts. By doing this, data which leaves the subnet is generally marked with an information which represents the subnet. In this case certain flows can be identified in the other networks as being originated from a dedicated part of the network (e.g. Subnet 1 in our example figure).

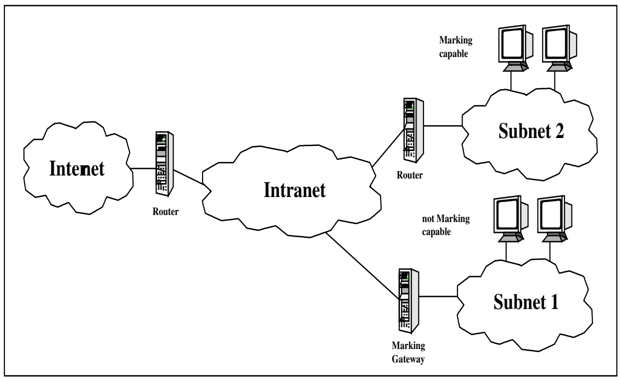

Figure 2: Infrastructure Based Marking Approach

4.2 Security of the mapping information

Since the mapping of user or application information to data streams might often be sensitive to spoofing, we consider that it can be protected in a secure manner based on cryptographic algorithms if the operation environment (such as e.g. working outside an "internally trusted company network") demands that. Mechanisms for doing so exist with a message authentication code based on a (e.g. predefined) shared secret. We refer to the mechanisms Secure ONC RPC [4] and Security Enhanced SNMPv2 [9] for now.

Our approach must and will be enhanced concerning its security but is already viable for a number of environments with "cooperative participants".

5. USAGE SCENARIOS

In this section we present usage scenarios that show how packet marking can be used for the placement of copyright or originator/retriever information in media content and for enhancing the interaction with the security infrastructure. The description is not comprehensive nor even fully representative and can be extended by e.g. billing support as well as by support for the enforcement of single login and (user- or application-) class-based security policies.

5.1 Support for the placement of originator or copyright information

In this scenario we assume library servers for pictures, audio/video data or special documents. These servers can e.g. be accessed via HTTP or by means of a streaming protocol. In the case of many user request (e.g. for an electronic public library) there might be several servers for scalability reasons.

Watermarks [7] are one of the possibilities to add copyright or originator/retriever information in the data that is down- or uploaded by the users. To be able to track a user in case of a copyright violation, the watermark should include user specific information.

With our approach it is possible to deploy the placement of the watermarking information at dedicated points that the data traffic passes through without modifying the original servers. The approach is not targeted at implementing dedicated watermarking mechanisms (which significantly differ e.g. for packaged vs. streamed content and different media) but makes use of those and parameterizes them. The parameterization info can be gathered either explicitly (e.g. because a user logs on to the service first) or implicitly by means of the analysis of network traffic dependencies (e.g. TCP requests/replies).

The scenario can be adapted for many other use cases e.g. for tracing who brought certain data (pictures, documents) into an administrative domain. In this case the mechanism has to be deployed at the receiver instead of the server side.

5.2 Firewall Interaction

Firewalls [2], [3] are specialized network nodes, which perform access control at network borders. They consist of packet filters, stateful filters, proxies or a combination of all these. Based on the analysis of the specifics of data traffic (using passive or active information retrieval), a firewall system decides whether packets may be passed through.

If the marking approach is used, a firewall could benefit from the information included in the flows in the following ways:

- User information: Normally, the authentication at a firewall system is performed by application-specific proxies. By using the user information, authentication proxies would not be necessary. The firewall would be able to use a generic method (uniform for all applications), to perform authentication.
- Application information: The firewall can use this information, to determine which flows together form a session. With this information, the firewall is able to handle applications which use different flows for one logical session in a generic way.

As we have shown, firewall systems can benefit from the marking approach in many aspects. Using it it would be possible to build firewalls which have a better performance than existing systems, without compromising security.

6. CONCLUSION AND FUTURE WORK

We have described an approach which attaches and transmits user or application specific information to network data streams. Systems do definitely have a remarkable benefit from the availability of that additional information. We consider the approach an "enabling mechanism" that can fulfill its potential especially in interaction with other existing and emerging technologies, which can be parameterized using it. The viability of the mechanisms has been determined by means of prototype implementations for the main components and will furthermore be enhanced.

7. REFERENCES

[1] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. *Internet Request for Comments Nr. 2702*, September 1999.

[2] D. B. Chapman. *Building Internet Firewalls.* O'Reilly, Cambridge, 1995.

[3] W. R. Cheswick and S. M. Bellovin. *Firewalls and Internet Security.* Addison Wesley, 1994.

[4] A. Chiu. Authentication Mechanisms for ONC RPC. *Internet Engineering Task Force*, May 1999.

[5] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. *Internet Request for Comments Nr. 2460*, December 1998.

[6] W. Hua, J. Ohlund, and B. Butterklee. Unraveling the Mysteries of Writing a Winsock 2 Layered Service Provider.

[7] S. Katzenbeisser and F. (Editors). *Information hiding techniques for steganography and digital watermarking.* Artech House Books, 1999.

[8] S. Kent and R. Atkinson. IP Authentication Header. *Internet Request for Comments Nr. 2402*, November 1998.

[9] W. Stallings. SNMP and SNMPv2 - The Infrastructure for Network Management. *IEEE Communications Magazine*, 36(3):37–43, March 1998.

[10] SunSoft. *STREAMS Programmers Guide.* SunSoft, November 1995.