

# Interconnection Protocols for Interorganization Networks

DEBORAH ESTRIN, MEMBER, IEEE

**Abstract**—This paper analyzes the technical implications of interconnecting networks across organization boundaries. Such *Interorganizational Networks (ION's)* are used increasingly to support exchange of CAD/CAM data between manufacturers and subcontractors, software distribution from vendors to users, customer input to suppliers' order-entry systems, and the shared use of expensive computational resources by research laboratories, as examples.

We begin by demonstrating that interorganization connections are not satisfied by traditional network design criteria of connectivity and transparency. A primary high-level requirement is access control, and therefore, participating organizations must be able to limit connectivity and make network boundaries visible. We summarize an approach to access control in ION's, based on nondiscretionary control, that allows interconnecting organizations to combine gateway, network, and system-level mechanisms to enforce cross-boundary control over invocation and information flow while minimizing interference with internal operations [6], [4]. The focus of this paper is on the underlying interconnection protocols that are needed to support these access control mechanisms.

We describe in detail a particular protocol, called a *visa* scheme [7]. The visa scheme uses access control servers to authorize a session request and *visas* to authenticate that successive packets belong to the authorized connection. Control is distributed among the ION participants and each may make its own design tradeoffs between performance and trust. In order to support interorganization communication two (or more) organizations must be able to communicate with one another's access control servers and their respective packet-level gateways and nodes (source/destination) must implement the visa scheme. The security of the proposed mechanism varies according to the security of an organization's components (access control server, gateway, and select hosts) and the encryption function used. The visa scheme's purpose is to allow an organization to modify and trust only those internal systems that require ION access; all other internal systems are inaccessible from and to the ION gateway.

We conclude by comparing and contrasting the visa approach to the use of higher level gateways.

## I. CONTROLS FOR INTERORGANIZATION NETWORKS

**I**NTERORGANIZATION Networks (ION's) support person-to-person communication via electronic mail; exchange of CAD/CAM data, software modules, or documents via file transfer; input to an order-entry or accounting system via a database query and update protocol; or use of shared computational resources via an asynchronous message protocol or remote log in. In most such

interorganization arrangements, the set of resources that an organization wants to make accessible to outsiders is significantly smaller than the set of resources that it wants to remain strictly internal (i.e., accessible to employees of the organization only); see Fig. 1. In addition, because the potential user is a person (or machine) outside the boundaries of the organization, the damage associated with undesired use can be high. Because of these characteristics, ION's have special access control requirements. These control requirements in turn place new requirements on the underlying network protocols.

This paper analyzes these requirements and proposes a protocol to support them. This Section I summarizes the control requirements, and Sections II-IV discuss the implications for network protocols, and our proposed visa scheme.

### A. Control Requirements

Organizations frequently make their internal computing facilities accessible to off-site employees via public switched or packet networks. These external connections sometimes require added security measures in order to prevent *outsiders* (i.e., persons who are not members or employees of the organization) from gaining access to the internal facilities. However, in an ION, the goal is not simply to prohibit access by outsiders; some outside access is explicitly desired. The goal is to support access to certain machines, services, and processes, while preventing access to all other internal facilities.

Unfortunately, the two obvious approaches—1) physical isolation of externally accessible systems from internally accessible systems (see Fig. 2), and 2) increased access controls/security on all internal systems (see Fig. 3)—are not generally acceptable solutions. Namely, because the function of the internal network predates and dominates that of the ION, interconnection and associated controls must not interfere with internal operations. Therefore, it is not acceptable that ION facilities be physically isolated from all strictly internal resources for this would interfere with internal access to information and resources that reside on ION systems. Similarly, requiring that *all* internal facilities adopt additional security measures to cope with an external connection may interfere with internal communication and resource sharing, and furthermore, assumes global knowledge of all interconnections. In general, we want to implement *logical*

Manuscript received February 2, 1987; revised May 6, 1987. This paper was presented in part at the 1987 IEEE Symposium on Security and Privacy, April 1987 and at SIGCOMM '86, ACM, August 1986.

The author is with the Department of Computer Science, University of Southern California, Los Angeles, CA 90089.

IEEE Log Number 8716983.

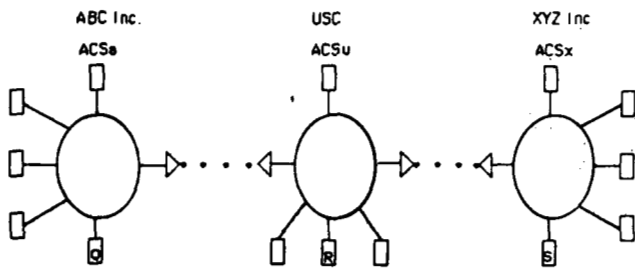


Fig. 1. A simplified ION among a university and two local companies. [Q], [R], and [S] represent externally accessible resources of the three respective participants. [ ] represents each organization's strictly internal resources.

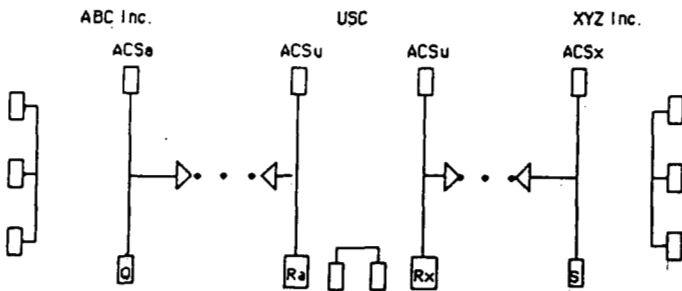


Fig. 2. Physical isolation of externally accessible and strictly-internal resources. [Q], [Ra], [Rx], and [S] represent externally accessible resources of the three respective participants. [ ] represents each organization's strictly internal resources.

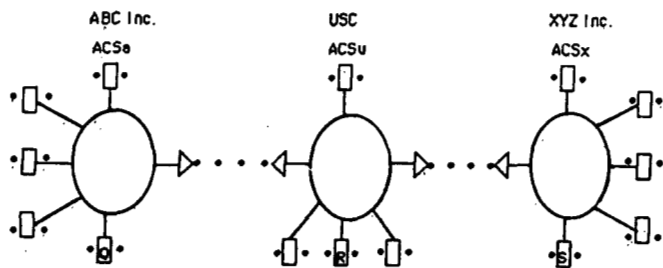


Fig. 3. Special system level access control on all internal systems. [Q], [R], and [S] represent externally accessible resources of the three respective participants. [ ] represents each organization's strictly internal resources. [\*] represents special access controls added to a system.

networks that can be isolated from one another yet share physical resources. Similarly, when two organizations, e.g., ABC and XYZ, interconnect, it may be inappropriate to create a connection between other organizations to which ABC and XYZ were interconnected previously. In other words, the new ION may overlap physically with the existing ION's, but it must not form a transit path between those organizations that desire to remain isolated from one another.

In [4] and [3], we analyze in detail these control requirements and the constraints that call for significantly different approaches to security mechanism design and implementation. To summarize, we conclude that only the administrators of the external link (i.e., the ION gateway) and the internal resources that are made explicitly accessible should be required to take security action in response

to an external connection. Owners of all other internal resources should be assured that their facilities are not accessible to outsiders.

To realize these controls, each organization must control all exit and entry points to its internal network. We refer to these control points as ION gateways and assume that no direct external connections are permitted; i.e., all external connections are via specified ION gateways.

The intention of this approach is to minimize interference with the organizations' internal facilities and operations by putting controls in ION gateways. However, the approach does impose new requirements on the network interconnection machinery—namely the ION gateway and the communication protocols used to communicate with and through it. The remainder of this paper investigates the implications of the proposed control mechanisms for network interconnection protocols.

## II. IMPLICATIONS FOR NETWORK INTERCONNECTION

As summarized in the previous section, we propose an approach to the problem of access control in ION's based on controls in all entry and exit points to each organization's internal network, i.e., all ION gateways. The primary difference between such ION gateways and traditional, uncontrolled gateways is that the nondiscretionary control mechanisms in the ION gateway must have access to additional information about traffic, e.g., organization affiliation of source and destination, type of service, or amount of resource requested. According to this information the gateway determines which categories of internal information or resources the external entity may access. In other words, in addition to the traditional bindings between user (or service) and node, node and network attachment point, and network attachment points and path [13], the ION gateway needs a binding between user (or service) and organization affiliation.

If the logical information required for the access control decisions is available, then the controls can be implemented, for example, by assigning category sets to incoming and outgoing traffic, according to logical characteristics of the traffic, and enforcing invocation and information flow controls accordingly. In the remainder of this paper, we describe what is required of the underlying network protocol in order to make this logical information available to the gateway. We evaluate the alternatives and tradeoffs associated with designing ION gateways to implement these controls.<sup>1</sup> Although most of our examples focus on the issue of controlling ION flows to meet security policy concerns, similar mechanisms are needed to support other aspects of network management, e.g., accounting and resource usage.

### A. Level of Interconnection

As with any gateway, an ION gateway can be designed to operate at one of several levels. For the sake of sim-

<sup>1</sup>Some of these ideas were first presented in [5].

plicity, we classify gateways as either high or low level. A high-level gateway is an end-point in a message- or connection-based communication session, such as file transfer, remote login, or electronic mail. A low-level gateway forwards packets between machines that are the endpoints of higher level message- or connection-based communication sessions, but the gateway itself is not an endpoint.<sup>2</sup> In terms of the International Standards Organization, Open Systems Interconnect (ISO-OSI) reference model [14], high-level gateways operate at transport, presentation, session, or application layers, whereas low-level gateways operate at the *network* layer.

A major difficulty of applying traditional interconnection methods to interorganization connections is that most existing gateways operate almost exclusively at lower protocol levels and most of the low-level protocols (and therefore the gateways) do not have access to the information needed to make ION policy decisions. The absence of policy related information is not inherent to this level of connection, but is a result of the competing requirements that constrain the design of low-level protocols. Our proposed ION gateways must at least be able to identify the organization affiliation of the traffic destination and source. In low-level protocols, information about the source and destination is carried in the packet header in the form of network addresses. The following paragraphs describe some of the problems of relying on these addresses, and therefore, the traditional low-level protocols—for identification of organization affiliation.

Networks interconnected at the datagram level (e.g., Internet Protocol (IP) level in the DARPA TCP/IP family of protocols) must coordinate the assignment of network identifiers in order for datagram addresses to be meaningful throughout the internet. In addition, internetwork addresses provide information about efficient routing of a packet to its destination, e.g., which subnet on which network a particular host sits. This routing information pertains to the physical location of the destination. When networks cross organization as well as geographic boundaries, *logical* information is desired in addition to *topological* information. In other words, policy control mechanisms need to know the organization to which a message is being sent, and from where it came, in addition to the physical locations. One possibility is to interpret the physical address as a logical address. The DARPA Internet illustrates why this is not a generally useful approach.

Currently, network identifiers in the Internet are allocated to sites by a centralized number czar. Each site may then allocate addresses to hosts and even subnets that lie within its topological network. Most of these hosts and subnets are within the confines of a single organization, but some are not. For example, in Fig. 4, USC has direct network connections to two local companies. The net-

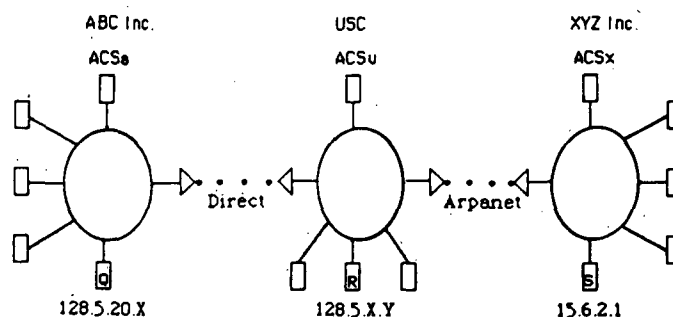


Fig. 4. An ION with internet addresses shown. ABC is connected as a subnet of USC; XYZ is connected to the Arpanet and thereby to USC. [Q], [R], and [S] represent externally accessible resources of the three respective participants. [ ] represents each organization's strictly internal resources. The numbers listed are *not* actual internet addresses.

work address of ABC corporation looks like the network addresses of other USC subnets because the address contains topological information for routing purposes. In order to discriminate between subnets and hosts that are part of USC's logical network (i.e., actually belong to USC) and those that lie outside of the logical network (i.e., facilities in the local companies which are accessible but do not belong to USC), XYZ's gateway must be able to bind the source and destination internetwork addresses in the datagram header to the organization affiliations. One might try to use the network and subnetwork numbers as a *hint* to organization affiliation. However, because of the decentralized manner in which networks and subnetworks may establish their own interconnections, over time these topological numbers may not map into meaningful logical groupings.

These issues were not among the many considered during design of the Arpanet/Internet protocols. At that time, the primary concern was to achieve connectivity and transparency and make network boundaries disappear.<sup>3</sup> Therefore, it makes sense that providing information needed to enforce organization boundaries was not a design requirement. Even if it had been a consideration, the number of competing requirements and constraints on the low-level protocol would probably have led the designers to leave such application specific information to higher levels. In particular, because routing table size is limited, there is pressure to be able to make routing decisions on the basis of a datagram's destination subnet number.

An example of a packet-level ION gateway that implements usage controls is the University College London (UCL) network connection to the Arpanet [2]. The UCL network employs two gateways to the Arpanet. One connection forwards packets via a private satellite network to the Arpanet. The second connection forwards packets via an X.25 connection over public packet-switched networks. The two separate gateways are needed because of

<sup>2</sup>Low-level gateways may operate on individual packets (datagrams) or virtual circuits, depending upon the protocol design of the interconnected networks.

<sup>3</sup>In many ways, the Arpanet is not a typical ION. In the past, the Arpanet was intended to encompass the internal networks of the participating organizations. Only recently, as the participating organizations have extended their internal networks to other internal communities, is the Arpanet manifesting many of the issues attributed here to ION's.

the different protocols used, and the division satisfies policy requirements. Due to PTT regulations, only Ministry of Defense traffic can be sent via the private satellite path, while civilians (such as many university researchers) must send traffic via the public network path. Because only routing information is available at the IP level, the restriction is enforced by making UCLnet appear as two separate networks, UCLnet and PSSnet. This is achieved by splitting the namespace in two and assigning addresses to MOD and civilian hosts, accordingly. Because there is a small and fixed number of user groups (i.e., two), under centralized control, the mechanism works.

The above example demonstrates that it is possible to identify individually the various subnet numbers that are assigned to external entities and add such information to the gateway filter explicitly. However, this is not a *general* solution because such interconnections are established in an incremental and decentralized manner, and therefore there is no good way of tracking the exception cases without centralizing the interconnection and address allocation process in some way. One approach might be to establish guidelines that set aside blocks of network addresses to be used for external sites (e.g., XYZ corporation in Fig. 4). Unfortunately, because of the nature of the namespace, it is hard to know *a priori* how many such numbers to set aside, and exactly what groupings one will want to be able to distinguish between, i.e., USC/non-USC, is only one relevant distinction. If such guidelines do not exist, and the connection is not centrally managed, it is not feasible for the gateway to maintain a list of allowable host and/or subnet addresses with which to implement packet-level controls.

In conclusion, traditional gateways, and the packet-level protocols that they speak, do not carry the information in each packet header that an ION gateway needs to make policy decisions. Provided with access to ad hoc mappings of network address to policy related information, specific ION gateway requirements can sometimes be met. However, the following sections describe techniques for network interconnection that are better suited to interconnection across organization boundaries.

Section III describes our proposed visa scheme for augmenting a network level protocol in order to accommodate policy controls. Section IV provides a basis for comparison by describing alternative approaches that involve interconnecting at higher protocol levels.

### III. VISA SCHEME

This section describes an access control protocol, called a *visa* scheme for use in Interorganization Networks (ION's).<sup>4</sup>

Ordinarily, gateways forward packets between networks indiscriminantly, i.e., based on routing information only. If such a gateway is used for an interorganiza-

tion connection, all internal resources are potentially accessible to all external machines, and all internal machines can potentially gain access to external resources. Some organizations address the need for control of such connections by implementing high-level gateways with access control functions; for example, an electronic mail relay that forwards mail to and from registered users only. While suitable for some ION's, high-level gateways suffer from performance overhead of the gateway's high-level processing, and reduced generality and flexibility, since special high-level gateway software must be constructed for each high-level protocol supported. The purpose of our visa scheme is to implement access control in ION gateways without incurring the costs inherent to high-level gateways [5].

One simple way of implementing access control is to place a source destination filter in the packet-level gateway, i.e., to maintain an access control list based on internet addresses. However, this approach works only if the access control list is static or if the source and destination ID's carry sufficient information to inform access decisions. If there is a well-defined set of resources that are to be accessible by a well-defined set of entities, then the access control list could be managed manually. Alternatively, if internet addresses are structured in such a way that the gateway can classify a node according to the range into which its internet address falls, the gateway could maintain an access control list by node classes (internet address regions), and thereby achieve greater flexibility.

In this paper, we are interested in the more general case of a dynamic environment where network addresses by themselves do not provide sufficient information for the gateway to make a policy decision about whether or not to permit access; the DARPA Internet is one such environment. As described in [5], internet numbers are assigned to carry topological, not logical information, while policy decisions are generally based on the latter. Because internet numbers do not carry enough information to assist access control decision making, our first proposal is that before an ION packet-level gateway starts passing packets between internal and external machines, it should require both internal and external participants to carry out a high-level conversation with an access control server (ACS). The ACS would decide whether or not the connection is authorized based on the high-level information provided. After authorization, the ACS could inform the gateway that the connection between that source and destination was approved and the gateway could then check all address fields of arriving packets and reject packets whose source destination pair was not registered.

Unfortunately, if the gateway relies solely on a list of approved address pairs provided by the ACS, the gateway, as well as the ACS and authorized internal nodes, must trust all internal nodes to not masquerade as other nodes, i.e., not to fake their internet addresses. In a decentralized environment with many personal computers and workstations, it is not hard to modify one's internet address. As a result, this simple scheme does not provide

<sup>4</sup>Adapted from [7].

internal nodes and gateways with enough of a mechanism to protect themselves from malicious or fraudulent traffic. Without additional control it would be unwise for an organization to accept liability for outgoing ION traffic or for a particular internal node to accept responsibility for its own outgoing ION traffic. This led us to develop a more sophisticated mechanism, referred to here as a visa scheme and described in the following pages. In summary, the visa scheme is developed to address two limitations of relying on internet addresses alone for access control: 1) internet addresses are bound to topological information, and 2) machines on a local network can claim a false address rather easily.

The visa scheme described below implements controls in a packet-forwarding gateway by working in concert with an ACS. The ACS carries out the high-level evaluation of communication requests and the gateway enforces the ACS's decision using the visa scheme. The visa scheme allows an organization to trust only those internal and external nodes that it explicitly provides with unique visas. If an authorized connection is abused or a visa is passed from an authorized user to an unauthorized user, the responsibility can be isolated to a specific node and session. Without such a mechanism an organization, and the authorized machines within that organization, have inadequate means of protecting their liability for ION traffic.

In the following section, we present the visa mechanism and several design goals.

#### A. Overview of Design Goals

The visa analogy was first suggested by Reed (M.I.T.) and developed by Mracek [9] and Estrin [5]. It is referred to as a visa scheme because gateways are analogous to border control stations, access control servers to government offices, packets to tourists, and keys to visas.

In this scheme, in order for a host to send a packet via an ION gateway, it must obtain keys (visas) from the ACS's of the visa networks it wishes to exit and enter. If the host passes an ACS's policy filter, the ACS gives its local gateway the source and destination hosts' internet-network address and a visa with which to authenticate packets coming from or to the source host as they pass through the gateway. The same visa is given to the source host to stamp all outgoing packets for the duration of the session. To prevent or inhibit (depending on the strength of the stamping function) the acquisition of visas through interception of packets the stamp included in each packet is a function of the visa and the packet checksum. Abuse of a visa is therefore possible only if 1) the source or gateway machine releases the visa value or does not protect it adequately or 2) the attacker is able to invert the function used to stamp packets.

1) *Liability*: The visa mechanism is designed to allow an organization to connect to the outside world without modifying *all* internal systems to defend themselves from external access, and without having to trust all internal systems to not abuse the external connection in the name

of the organization. In other words, our goal is for an organization to *modify* and *trust* only those internal systems that explicitly request or require ION access. All other internal systems (the majority) would be unreachable by external packets and would not be able to export packets.<sup>5</sup>

The requirement for control of incoming traffic (i.e., external access to internal information and resources) is rather straightforward, namely, controlled access to proprietary resources. In addition to incoming flows, we are also concerned with outgoing traffic because generally when an organization *A* connects to an external organization *B*, *A* must agree to assume responsibility for the actions of persons and machines within its organization boundaries (e.g., to stand by purchase orders or other contracts written by its employees). In particular, *A* must vouch for the authenticity of internal entities that are able to export packets to *B*. If *A* is not confident as to the identity of an internal entity, then *A* should not allow it to use the gateway. Alternatively, *A* should not agree to ION connections for which the liability exceeds the level of confidence that *A* has in its internal access control mechanism.

The visa mechanism allows an organization to isolate trust and identify fault but it *does not* in and of itself provide any particular level of security. The security of the mechanism depends upon each organization's internal security, in particular, the ability of the source and gateway machines to prevent access to their visa values, the protection of visas during distribution, and the strength of the stamping function. The value of the visa mechanism is that it allows an organization to exert control over ION connections in a way that is consistent with its security guidelines. Moreover, an organization does not have to trust all its internal entities. It only trusts those that it explicitly permits to use the connection to the outside (see Section III-D).

2) *Flexibility*: One of the main benefits of this scheme is its flexibility. Each organization employing the visa scheme should be able to tailor it to reflect that organization's policy regarding incoming and outgoing traffic and to make its own tradeoffs in performance and security. The scheme is designed to support this diversity in addition to minimizing requirements for trust and *a priori* agreements across network boundaries. Where such requirements remain, the placement of trust is explicit and well isolated.

3) *Transparency*: In addition to flexibility, transparency of the underlying mechanism is an important design goal. This scheme must allow an organization to connect some subset of its internal resources to some subset of the outside world without endangering or tampering with any

<sup>5</sup>Many workstations and personal computers may be designated to receive electronic mail from external sources. However, for such applications, these hosts need not be directly connected to the ION gateway; rather a mail server would be one of the ION accessible machines and it would in turn forward mail to individual hosts after applying appropriate controls.

other internal facilities. The scheme must allow each ION participant to define the terms of liability that it and external parties must agree to. At the same time, interoperability with nonvisa users must be maintained for those systems that are globally accessible, i.e., impose no ION access control.

Another issue related to transparency is that the interconnection of two organizations may traverse other networks which may or may not be using the visa scheme. In such cases, the presence of the visa mechanism at the endpoint(s) must be transparent to the nonvisa, transit gateways.

### B. Visa Scheme Components

This section describes the main components of the visa scheme—hosts, visas, access control servers (ACS's) and gateways (GW's). A host that wants to communicate across its organizational boundary engages in a high level authorization and authentication procedure with the ACS's on the visa networks traversed. The need for ACS communication is determined individually by the owners of each participant network. After the source destination session has been approved by an ACS on each network, the ACS's allocate visas to their respective gateways and to the requesting host. The host uses the visa to stamp all ION packets. The gateways check all packets for appropriate stamping and pass packets until the visa expires or is terminated. If system processes are programmed to carry out the authorization procedure on behalf of the user, the entire process can be transparent to end users.

Our initial implementation of the visa scheme is based on the DOD internet protocol (IP) [11]. IP supports connectionless datagram service between hosts. It was designed to flexibly operate over a range of network types, and to adapt to changes in topology and congestion. Both connection and connectionless transport protocols run on top of IP. Although we have designed this scheme to work within IP, the fundamental concepts could also be applied to other protocols.

1) *Visas*: In the context of this scheme, a visa is a unique value (e.g., a cryptographic key) assigned to a session between two hosts on distinct networks. Each packet that is part of an authorized session carries a special stamp value in the IP header option field that includes the visa and packet checksum in its calculation (see Fig. 5). In our implementation, each visa packet carries two visas—one for the visa gateway that it is exiting, and one for the visa gateway that it is entering. This approach was selected because it provides flexibility in the future for different networks to employ different stamping functions (e.g., stronger functions than the simple IP checksum). The packet header format is described further below. The stamp is therefore a function of the visa, packet header, and packet data. Initially, while we work out the protocol details, we use the IP checksum as our stamping function. Because this checksum algorithm is not secure, stronger one-way, trapdoor functions will be employed in future

prototypes. This option for upgrading and tailoring the mechanism is one of the features of the visa scheme.

Each host that makes use of the ION maintains an active visa list (VL). Each entry in the VL consists of a visa, the addresses of the machines involved in the session, and any restrictions that may apply (e.g., time limit). Gateways and hosts also maintain records of which ACS provided each visa. Likewise, for an ACS, the VL includes the address of the GW to which a visa was allocated. Also, an ACS associates with each entry in its VL an address of the ACS on the source or destination network.

In some cases, it would be desirable to allocate visas to particular processes, not to entire hosts. However, packets do not carry process ID's or even port numbers. Consequently, in our implementation, the gateway maintains a visa list that maps visas to host ID pairs (i.e., source and destination host ID) and relies on the source host's visa-IP implementation not to share visas among processes. Therefore, when more than one process on a host obtains visas to communicate with a common destination host, the gateway accepts packets stamped with either visa. The gateway is, therefore, trusting the source host's visa-IP implementation to only employ a visa for the particular process to which it was allocated. For further discussion of how finer granularity could be achieved, see Section III-D.

When a host explicitly terminates a session, visa-IP sends a special ENDING packet to its ACS. The ACS deletes the visa from its VL and forwards the packet to its gateway and to the next network's ACS. The ACS of the next network deletes the visa(s), informs its gateway(s), and sends the ENDING packet to the next network's ACS, and so on. When the ENDING packet finally arrives to the destination network's ACS, it sends the ENDING packet to the local gateway and to the local host. At that time, all visas issued for that connection are invalidated by all parties involved. In addition, any GW or ACS may at any time decide to stop honoring a certain visa, e.g., timeout. In that case, it will send ENDING packets to its GW's, as well as to the local hosts and neighboring ACS's that are part of the session. The next packet bearing a stamp corresponding to the invalidated visa will be rejected. In the best possible case, the ACS of the nearest network still honoring that visa will be able to recover the connection. In the worst case, the rejected packet will propagate all the way back to the source and the whole visa issuing procedure will have to be repeated. This visa expiration mechanism is also needed to terminate visas that are associated with connectionless protocols; in this case, the participating host(s) will not generate explicit ENDING packets themselves. Similarly, when topological changes cause rerouting of packets, new visas will be required to pass through any new visa GW's or any old visa gateways that have crashed and resumed without previous state information.

The headers of visa related packets are illustrated in Figs. 5 and 6.

2) *Access Control Server*: An ACS is assumed to be a

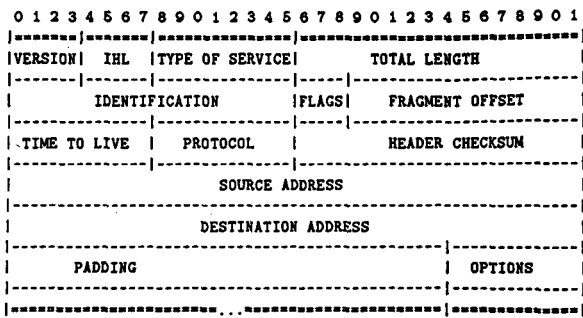
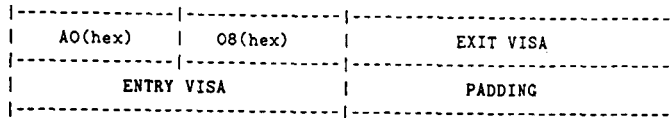


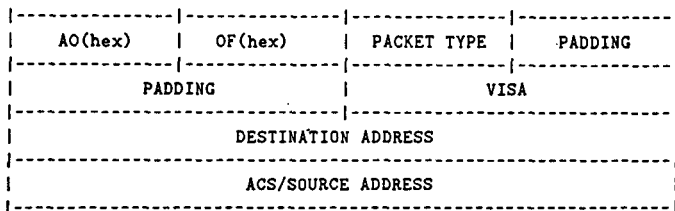
Fig. 5. Standard IP header. Options: one-byte options type and one-byte options length, followed by options data.



EXIT VISA: for exiting the current network

ENTRY VISA: for entering the next network

(a)



PACKET TYPE: VISA, LAST, REJECT, REQUEST.

ACS/SOURCE ADDRESS: reflects the address of the ACS in a REJECT packet, and the address of the source in a VISA, REQUEST or LAST packet.

VISA: only used in LAST and VISA packets.

(b)

Fig. 6. Visa scheme packet headers. (a) Visa option as it occurs in data packet. (b) Visa option as it occurs in a control packet.

host on the network (usually dedicated to ACS functions for security reasons) whose primary concern is access control. Each ACS knows of a number of local GW's to which it issues visas. Its presence, however, is not mandatory and levels of control can vary across organizations. If a participant network does not have an ACS, the scheme will still work, although the network in question will be subject to risk associated with uncontrolled access. ACS's are trusted and assumed to be defensive against attempted abuse from external entities. This assumption is critical because visa gateways allow any packet to flow to or from trusted internal ACS's.

The choice of the authorization and authentication procedures used by an ACS is the decision of each individual organization. The procedure may involve establishing a high-level conversation with the host, in which a pass-

word, biographical data, or other authenticating information is requested. Some ACS's will require end-user provided data; others will require information that the user's system can provide on its behalf. As described in [4], access control decisions may be most appropriately made according to group or class affiliation and associated category sets that determine access rights. The visa scheme itself does not dictate or constrain the particulars of the authorization schemes. One example of an ACS that could serve this function is the Kerberos Authentication Server developed at M.I.T. [8]. Regardless of the approach used, the visa scheme assumes only that a YES/NO decision is passed to the visa software. In this paper we describe the visa interface of the ACS, not the ACS design itself. Finally, significant application-specific access control is left to the end-point hosts and applications; our scheme addresses only control of access to the hosts on a network.

The ACS' functions can be summarized as follows:

- On receipt of a request-to-connect from a host, authenticate, and authorize that host.
- Issue new visas and send visa packets to participating GW's, and hosts.
- Expire visas upon termination request by participant host or ACS, or upon timeout, and notify all parties involved—hosts, other ACS's gateways.

Regardless of the authentication and authorization procedure used, when an ACS carries out a higher level protocol via which it authorizes a host, it must have access to more than just the network addresses or ids. This does require each participant host to understand the higher level protocol used by a particular ACS whose gateway that host wants to traverse. There are two options for dealing with this requirement: either the source host itself must have the ability to "speak" the higher level protocols or a local ACS must act on behalf of the source. In other words, one of the necessary, and unfortunately constraining, conditions for visa scheme implementation is that the ION participants' ACS's must satisfy one another's idiosyncratic higher level protocols or must have agreed upon a common mechanism *a priori* (e.g., a public key scheme).

ACS's play a critical role in this proposed scheme. Consequently, the availability of network service is a direct function of the availability of the ACS service. It, therefore, becomes worthwhile to designate backup ACS's within a single organization. In this case, each gateway would be initialized with the address of backup ACS's in case the primary ACS becomes unavailable. Similarly, the security of the scheme is dependent upon the security of the ACS. This suggests that the ACS reside on a dedicated trusted machine, and that the ACS employ a secure mechanism for communication with hosts; see paragraph 3.4 for further discussion. In addition, as is described in Section III-D, ACS's should employ mechanisms to ensure secure distribution of keys, i.e., visas.

3) *Gateway*: An ION GW is assumed to be a machine

on the network (usually dedicated to performance, and in this case security, reasons) concerned primarily with packet forwarding. Each GW knows of some number of trusted, local ACS's. By trusted we mean that the GW is willing to accept visa assignments from these ACS's and thereby trusts their decisions about authorizing sessions. Moreover, the GW allows any external party to communicate with (send packets to and receive packets from) any registered, internal ACS; similarly the GW allows all registered, local ACS's to communicate with any external party. In other words the GW trusts the ACS to protect itself from any external access and to not abuse the ION connection. This trust is reasonable because ACS's are special machines explicitly designed to be defensive and to enforce organization policy.

The gateway's functions include:

- Trap all packets, extract visa stamp, search for source, destination, and visa in VL.
- Reject packets not possessing a valid stamp and return them to source along with the address of a local Access Control Server (ACS). If a packet does not possess any stamp option field, the gateway knows that the packet originated from a host that is not equipped to participate in the visa protocol. In such a case, the gateway simply drops the packet and leaves it to the source to timeout and diagnose why the connection was not established.
- Forward packets bearing a valid visa through.
- Accept special visa packets from the trusted ACS and add new visa entries to the VL.
- Accept special ENDING packets from trusted ACS and delete visa entries from the VL.
- Upon visa expiration, notify the corresponding ACS.

4) *Network Environment*: The particular visa scheme described here is designed to operate on IP networks such as the DARPA Internet as well as privately operated internets [11]. The general approach is applicable to other protocols, but implementation is protocol dependent. Moreover, the need for such a scheme is greatest in datagram networks where there exist no closed user group mechanisms such as are provided by X.75 [1]. Visa software is being integrated into IP code. We chose to implement the visa mechanisms at the IP level to exploit the use of this protocol for efficient network interconnection. In the future, to evaluate the relative value of this approach, we will compare its performance to that of transport and higher level gateways.

### C. Illustration

The following example illustrates how the visa scheme is applied in a sample Interorganization Network. This example illustrates a pairwise connection. The scheme conceptually works in a multinet network case where intermediate networks also employ visa gateways. However, due to the overhead per visa gateway transited, we suggest the scheme is most practical for the end-to-end case (i.e., only gateways on the source and destination net-

work enforce visa requirements). See Section III-C for further discussion.

Fig. 7 shows the interconnection of a university department and a research division of a manufacturing company. Suppose that department *A* was contracted to do some research for company *B*. Furthermore, *B* is allowing a certain number of faculty to use some of its resources in order to assist with ongoing research. However, being understandably protective about its assets, *B* is very much concerned with security and requires restricted access to internal systems. At the same time, physical isolation is not an acceptable solution because it limits the functionality of the connection by preventing communication between ION-accessible and strictly internal machines. Instead, *B* "screens" all incoming and outgoing connections and imposes time limits on sessions. *A*, on the other hand, is only concerned with the appropriate usage of its gateway and external machines (i.e., *A* is more concerned with liability than with protection) and requires anyone requesting a remote connection be authorized to do so.

If a professor operating machine *X* located on the network *A* wants to query a database (host *Y*) located on the network *B*, the following procedure takes place:

- 1) *X* sends a packet addressed to *Y*.
- 2) The packet is trapped by *GW<sub>a</sub>*. The packet does not have a valid stamp. *GW<sub>a</sub>* sends a REJECT packet to *X* along with the address of the local ACS and *ACS<sub>a</sub>*.
- 3) *X* sends a REQUEST packet to *ACS<sub>a</sub>*. *ACS<sub>a</sub>* carries out an authorization and authentication procedure with *X*, the particulars of which will vary across organizations (and across different ACS's within an organization). The procedure may be executable by *X*'s local ACS or operating system, or may require *X*'s direct input.
- 4) a) If the ACS decides that *X* is not authorized to communicate with *Y* then the packet is dropped and it is left to the higher level protocol to time out and diagnose the problem. b) If *ACS<sub>a</sub>* does not reject *X* it sends a REQUEST packet to *Y* (on behalf of *X*). *GW<sub>a</sub>* passes the packet since it originated from a local ACS. But the packet is trapped by *GW<sub>b</sub>* and as in Step 2), a REJECT packet is sent to the source, this time *ACS<sub>a</sub>*, along with the destination's local ACS address, in this case *ACS<sub>b</sub>*.
- 5) On receipt of a REJECT, *ACS<sub>a</sub>* sends a REQUEST packet to *ACS<sub>b</sub>*. That packet passes through both *GW<sub>a</sub>* and *GW<sub>b</sub>* and gets to *ACS<sub>b</sub>* because both gateways are passing packets to or from recognized ACS's. Upon receipt of this packet, *ACS<sub>b</sub>* knows that someone wants a session with *Y*.
- 6) *ACS<sub>b</sub>* initiates its own authentication and authorization procedures with the requesting source *X* just as *ACS<sub>a</sub>* did. The conversation is carried out via *ACS<sub>a</sub>*, since *GW<sub>a</sub>* will only accept unstamped packets destined for an ACS.
- 7) After *ACS<sub>b</sub>* has authorized and authenticated *X*, it issues visa *XY<sub>b</sub>* and sends a special visa packet to *GW<sub>b</sub>* and *ACS<sub>a</sub>*. The gateways store the visa and associated information in their VL's.
- 8) When *ACS<sub>a</sub>* receives visa *XY<sub>b</sub>* it issues visa *XY<sub>a</sub>*

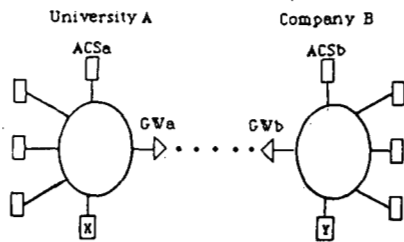


Fig. 7. Example ION between a university and company.

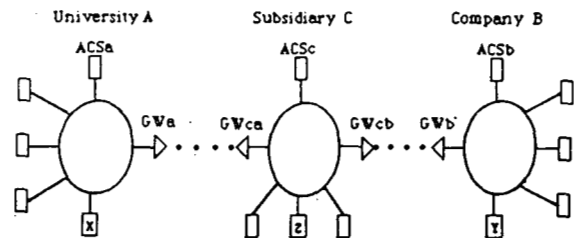


Fig. 8. Example ION between a university and company when the networks are connected indirectly. Network C operates as a transit network.

and sends it to  $GW_a$ . Then, it sends visa  $XY_b$  and visa  $XY_a$  to  $X$ . Now,  $X$  is armed with a visa for exporting packets from  $A$  to  $B$ .

9)  $X$  sends its first properly stamped packet (with  $XY_a$  and  $XY_b$ ) which passes through  $GW_a$  and  $GW_b$  and arrives at  $Y$ .

If  $ACSa$  and/or  $ACSB$  deploy symmetric policies regarding communication between  $X$  and  $Y$  (i.e., if  $X$  is authorized to send packets to  $Y$ , then  $Y$  is authorized to send packets to  $X$ ), then they can allocate two-way visas during the procedure described above. If  $ACSa$  or  $ACSB$  does not allocate two-way visas, then when  $Y$  attempts to reply to  $X$ 's communication, its first packet triggers the same procedure as was just described for  $X$ . This time, the first gateway and ACS involved is  $B$ 's followed by  $A$ 's. During this process if all participant ACS's authenticate and authorize  $Y$ , then they allocate visas to their respective GW's, and to  $Y$ , for the  $Y$  to  $X$  path.

In conformance with the spirit of IP, should any intermediate gateway or network go down, the session will resume automatically (albeit with additional overhead) just as when a gateway or ACS decides that a visa has expired or become suspect (see previous discussion).

1) *Transit Case:* For communication between  $X$  and  $Y$  when the networks of  $X$  and  $Y$  are not directly connected (see Fig. 8) the procedure may involve an additional set of steps.

If the intermediate network (e.g., belonging to an organization, "C") does not employ visa gateways, then the procedure would not change. The packets would simply be routed via an additional network before being processed by the participants' visa gateways; i.e.,  $C$ 's gateways would route the visa packets just as it does regular IP packets since the packets are not detectably different to a regular IP gateway (or host). If  $C$  employs visa gateways, it can elect to require visas for transit packets or to allow transit packets without special visas. In the former case,  $C$  may use the visa mechanism to discriminate in its provision of transit service. In the latter case,  $C$  is agreeing to a policy whereby it will either allow or restrict ALL transit packets, independent of the source and destination, etc. In the latter case,  $C$ 's gateways would recognize that the packets are transit packets and would pass them on without adding any steps to the visa setup phase. However, if  $C$  chooses to implement controls on transit traffic also, several additional steps are added to the visa setup phase. Steps 1-6 would continue as before, al-

though this time between  $ACSa$  and  $ACSc$ . However, instead of issuing a visa as  $ACSB$  did in Step 7 in the previous example,  $ACSc$  would continue the setup chain by attempting to send a REQUEST packet on to  $Y$  in network  $B$ . At that point,  $ACSB$  would get into the act as  $ACSc$  did before. Only after  $ACSB$  had assurance of authorization and authentication (via  $ACSa$  and  $ACSc$ ), would it then issue a visa. The visa issuing process would propagate back through  $C$  and  $A$  just as it did from  $B$  to  $A$  in the previous example. In this way, conceptually, the visa schemes are extendible to an internet in which any number of participating gateways and networks employ visa-based access control. However, the greater the number of visa gateways on the path between two points, the greater the overhead for that particular conversation. Consequently, we expect implementation to be practical when visa gateways treat transit packets differently from packets that are destined or originating from hosts on their network. This is accomplished by having each visa gateway on a network know about the other visa gateways on a network and allow transit packets to pass unchecked.

#### D. Implementation Issues

This section is devoted to the issues involved in implementation of the visa mechanism.

1) *Performance:* Our first and foremost goal in implementing the visa scheme is to analyze and evaluate trade-offs among performance, flexibility, and security. The extent to which we can actually meet our goals of transparency and flexibility and, yet, incur relatively low performance overhead will determine the usefulness of this security mechanism in a dynamic ION environment.

For the illustration given above, a minimum of 15 extra packets are generated before the first user packet gets through, not including the packets that comprise the authorization/authentication conversations between hosts and ACS's. Note that all of these control packets will be of minimal length. These ACS conversations may involve as few as two packets, but may involve many more depending upon the particular ACS design. Once the visa is allocated, successive user packets do not entail additional overhead (other than the added IP option field containing the stamp) unless a visa is expired or lost or the network state changes and a new gateway must be used.

There are several shortcuts that organizations can take that trade off trust for performance. For example, an or-

ganization may choose to allocate two-way visas automatically so that  $Y$  would not have to go through an explicit visa allocation process. Although this assumes greater trust in the remote organization, it would eliminate several steps and corresponding overhead. Another widely applicable example is passing transit packets without visas, as described earlier. In the future, the performance of this scheme must be compared to equivalent access control functions implemented in transport and higher level gateways.

2) *Security*: As mentioned previously, there are three points of potential vulnerability in the proposed scheme. The first is in the distribution of visas. If visas are distributed in the clear, then packets emanating from a local ACS can be monitored by an attacker on the local network and visas can be illicitly acquired. Assuming the attacker can modify its network address, the stolen visa could be used to send and receive unauthorized ION packets. We assume that in the future, most ACS's will have to carry out various kinds of key distribution functions and therefore will have an existing, local, mechanism by which to pass private information to hosts on the network, i.e., via encryption with the host's private key (e.g., as described in [10] and [8]). The second point of vulnerability is in the storage of visa lists by hosts and gateways. Once again, the vulnerability depends upon the level of security mechanism available on particular hosts within an organization. If an organization does not trust a particular host or gateway to have adequate protection mechanisms, the ACS would be programmed not to allocate visas to that host or gateway. Similarly, the gateway must trust the visa-IP software belonging to a particular host to not use a visa belonging to an authorized process for stamping a visa belonging to an unauthorized process when both processes are communicating with a common destination.

The third point of vulnerability is the stamp itself. The stamping function used must not allow a wiretapper to obtain the visa through analysis of the stamp and other packet data. Therefore, implementations should employ a strong one-way function for computing the packet stamp as a function of the visa and packet data or checksum. The function we are currently using is quite vulnerable to such attacks. Our rationale for beginning with a simple checksum is to investigate the other performance issues associated with our general protocol design. Future versions will experiment with more sophisticated stamping functions. In general, the more secure the scheme, the greater the computational overhead and the greater the need to employ special hardware. This might result in visa gateways being more expensive than traditional gateways. However, the relative number of ION gateways to internal gateways should be small and the expense justifiable. Once a host is registered as being accessible via the visa gateway, it is then up to that host to protect itself from abuse and to not allow transit traffic to other internal, non-ION, hosts.

3) *Implications for IP*: There are two significant implications for the use of the IP and other datagram pro-

ocols. The first is that this scheme imposes a kind of single path behavior on IP. Packets can travel via multiple paths only if the gateways coordinate sharing of visas. Therefore, we make use of the IP strict source routing option. The second issue is that fragmentation is a problem since the packet stamp is a function of the packet data (i.e., checksum). Consequently, stamps would have to be recalculated at all fragmenting gateways.

4) *Transport and Higher Level Protocols*: Although the visa scheme is being implemented at IP level, the choice of a higher level protocol is not arbitrary. At this time, the scheme is being experimented with under TCP [12]. Since TCP is a connection-oriented protocol our software can detect when a session is terminated and visas should be invalidated (check for FIN flag in TCP header). In the presence of a connectionless transport protocol (e.g., UDP), detecting the end of an application level session becomes not possible; for such applications timeouts must be used to expire visas. Further research is needed to determine the role that the visa scheme can play in support of connectionless protocols. The source of the problem is that we are modifying the connectionless IP protocol to be "aware" of connections in the sense of expiring visas when transport level connections are closed.

It is sometimes necessary to issue visas to specific users or user processes, not to entire hosts. Although this issue may not arise in a PC environment where a machine is usually associated with a single user, in a multiuser environment the internet address (common to all users on a host) is not fine-grained enough to provide process-level control. In that case, higher level ID's are needed to distinguish among user processes. Both UDP and TCP provide such information in their headers (port numbers). Thus, visas could be issued to specific user processes if the visa-IP code is programmed with knowledge of specific transport protocols (e.g., where to find the port information in the UDP header of each IP encapsulated UDP packet).

5) *Outstanding Design Issues*: We conclude our discussion with a list of several outstanding design issues.

- In our experiments we are investigating the tradeoffs associated with implementing functions in the ACS or gateway. We need to offload as much as possible from the gateway to maximize gateway performance while not exporting so much as to degrade performance through excessive communication requirements.

- We have designed this scheme to work within IP. However, the fundamental concepts could also be applied to other protocols. The analysis and implementation of visas in other protocols is left for future investigation.

- More experience with the protocol is needed before we can evaluate the practicality of this scheme in the transit case, i.e., where networks enforce visa-based control over transit traffic.

- Finally, there are questions associated with the interaction of our modifications and existing transport and higher level protocol mechanisms such as timeouts. Visa

protocol performance must allow us to operate within the timeout periods of higher level protocols.

#### IV. COMPARISON OF VISA AND HIGH-LEVEL ION GATEWAY IMPLEMENTATION

As described, the ION gateway's primary purpose is to associate packets, messages, or connections with access rights and either forward or reject them according to the designated policies. We, therefore, compare visa and high-level gateways based upon the cost and ease of implementing the following functions without violating the design constraints outlined in Section I. The two approaches differ from one another with respect to several of the ION gateway tasks. The most significant difference is in associating communications with logical information. The approaches also differ with respect to several performance parameters.

Higher level gateways (sometimes called application relays) terminate the higher level communication protocols and thereby gain access to more information about the application of the connection. These protocols deal with aggregated units of traffic that contain more semantic information in the headers and control fields (e.g., electronic mail messages, remote log in, or file transfer connections, etc.). Depending upon the level of connection and application, this information may include the logical affiliation of source and destination, the actual service being performed, and the amount of communication resources requested, for example. Although a key security issue remains with regard to the authentication of this information, the point is that the information is available for evaluation, and authentication mechanisms can be employed as needed. Note that even with a higher level ION gateway, some controls are best implemented in the endpoint applications themselves; in particular, controls that discriminate according to the content of a message, e.g., the size of a purchase order or the name of a file requested. In addition, these applications-level controls may be required to isolate the ION processes and applications from the non-ION ones.

A high-level gateway can associate communications with logical information directly, whereas the logical information needed to implement intelligent filtering in an ION gateway is not available at the packet level. Therefore, the visa scheme must employ a high-level dialog with an ACS to associate packets with logical information. The visa scheme described above can effect higher level control for many simple usage control policies. However, *when policy decisions are dependent on higher level information that cannot easily be bound to packet-level information and represented in the form of a key, higher level connections may be more suitable.*

The packet-level gateway must be able to evaluate the legitimacy of each packet based solely on the packet header and the visa. It is difficult, and sometimes impossible, to represent complicated policies in this manner. For example, there is no way for a packet-level gateway

to discriminate on the basis of mode of access (e.g., mail, file transfer, remote log in, etc.) because no information about higher application levels is available in the packet headers.<sup>6</sup> Consequently, even if mode of access is indicated in the visa, there is no way for the gateway to verify that a particular packet is supporting one mode of access and not another since this information is not carried in the packet header. The gateway must then rely on the endpoint node to use this visa only for the application originally authorized. The same problem arises if the gateway needs to discriminate on the basis of user ID. For this reason, higher level gateways are better suited to implementation of some types of policies.

On the other hand, high-level gateways evaluate each connection or message according to programmed control policies. They may or may not apply some check to each successive packet in a connection. In addition to employing an ACS to apply a high-level control algorithm to each connection or message request, a visa-based gateway always checks each and every packet against the visa. The same problem arises if the gateway needs to discriminate on the basis of user ID. For this reason, higher level gateways are better suited to implementation of some types of policies.

The two schemes are comparable in terms of several cost and performance criteria—storage and trusted components—but differ significantly in terms of others—end-user performance and protocol modification. Storage requirements are the same for both, although a high-level gateway may store control information locally instead of in an Access Control Server (ACS) and a visa gateway by definition stores it in an ACS. In addition, the visa gateway stores locally a small number of currently-in-use keys, whereas the high-level gateway maintains more state information about the connections passing through it. In both cases, the amount of storage used for access control information depends on the grain of control, i.e., user, host, network, organization. The two approaches are also similar in terms of the number and extent of components that must be trusted. In both cases, security depends upon the authentication of header and connection request information, the evaluation program in the gateway and ACS, and the ability to subvert the access control mechanisms used to approve connections or messages. The latter risk is somewhat higher for gateways that do not authenticate each packet.

The two schemes differ in performance overhead and the modification required of existing protocols. Each of the methods exacts a performance cost. The visa gateway is costly because of the required dialog with the ACS and the checks applied on a per packet basis. The high-level gateway is costly because protocols are terminated and because the gateway must be programmed with each

<sup>6</sup>In the Arpanet, packet-level gateways can make a pretty good guess at the higher application because of the use of well-known ports. Specific port numbers are routinely used for specific applications throughout the Arpanet. Packet headers do contain the port number and therefore the higher level application can often be determined.

higher level protocol that it supports. The tradeoff depends much on traffic patterns, in particular, the number of packets per session or message, the volume of traffic, and the number of communication service types. On the other hand, protocol conversion is hardest for the lower level protocol because of the tighter real-time constraints; for the same reason protocol conversion is harder for connection-based, higher level gateways than it is for message-based.

A second significant difference between the two methods is that visa gateways require that all internal systems that use the ION add the visa to the header or checksum calculation. This requires that each machine modify its low-level communication protocols. In contrast high-level gateways require that application-level procedures be changed; or in some cases, only that name tables be updated. Although the latter is less transparent to the end user, the cost and inconvenience of software modification is avoided. This cost can be quite high if it implies incompatibility with existing and future equipment. On the other hand, an additional cost associated with higher level gateways is the need to program the gateway separately for each higher level protocol that the organization wants to support; in contrast, the packet-level gateway supports all higher level applications.

In summary, the most difficult aspect of implementing ION gateways is the association of communications with logical information. Aside from this difficulty, the major implementation decision is whether to interconnect at the packet level and employ an ACS and visa scheme or whether to interconnect at higher levels and employ structured naming. Each approach is well suited to different environments and may be used in conjunction with one another in some cases. Finally, these controlled connections should be placed as close as possible to the administrative boundary being enforced.

## V. SUMMARY

This paper has characterized ION's and the access control and network interconnection issues raised therein. We began by characterizing a set of control requirements that are not addressed by traditional security mechanisms. We then identified and characterized these applications for which transparency, connectivity, and performance criteria alone are not adequate for selecting the interconnection method. Finally, we proposed and evaluated visa-based and higher level interconnections as alternatives to traditional, uncontrolled packet-level interconnection.

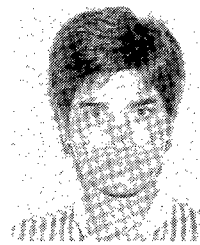
## ACKNOWLEDGMENT

The author would like to thank G. Tsudik for invaluable contributions to the design and implementation of the visa scheme. In addition, we thank M. Bishop, J. Mogul, B.

Leiner, J. Saltzer, L. Zhang, and anonymous reviewers for comments on a previous draft.

## REFERENCES

- [1] Anonymous, "Recommendation X.75, terminal and transit call control procedures and data transfer system on international circuits between packet-switched data networks," Tech. Rep., CCITT, 1980.
- [2] R. Braden and R. Cole, "Some problems in the interconnection of computer networks," in *Pathways to the Information Society: Proceedings of the 6th International Conference on Computer Communications*, W. Williams, Ed., North-Holland, Sept. 1982.
- [3] D. Estrin, "Access to interorganization computer networks," Ph.D. dissertation, Dep. Elec. Eng. Comput. Sci., Massachusetts Institute of Technology, Aug. 1985.
- [4] —, "Controls for interorganization networks," *IEEE Trans. Software Eng.*, Feb. 1987.
- [5] —, "Interorganization networks: Implications of access control requirements for interconnection protocols," in *ACM SIGCOMM '86 Symp. Commun. Arch. Protocols*, Association for Computing Machinery, New York, NY, 1986.
- [6] —, "Nondiscretionary controls for inter-organization networks," in *Proceedings of the 1985 Symposium on Security and Privacy*. Silver Spring, MD: IEEE Computer Society Press, 1985.
- [7] D. Estrin and G. Tsudik, "Visa scheme for inter-organization network security," in *Proceedings of the 1987 Symposium on Security and Privacy*. Silver Spring, MD: IEEE Computer Society Press, 1987, pp. 174-183.
- [8] S. Miller and B. Neuman, "Kerberos: Athena authentication, authorization, and accounting plan. Draft 3," Massachusetts Institute of Technology, Project Athena, Cambridge, MA, Tech. Rep., July 1985.
- [9] J. Mracek, "Network access control in multinet internet transport," Dep. Elec. Eng. Comput. Sci., B.S. thesis, Massachusetts Institute of Technology, Cambridge, MA, Tech. Rep., June 1983.
- [10] R. Needham and M. Schroeder, "Using encryption for authentication in large networks of computers," *Commun. Ass. Comput. Mach.*, vol. 21, no. 12, pp. 993-999, Dec. 1978.
- [11] J. Postel, "Internet protocol: DARPA internet program protocol specification," USC Inform. Sci. Inst., Marina del Rey, CA, Tech. Rep., Sept. 1981.
- [12] J. Postel, "Transmission control protocol: DARPA internet program protocol specification," USC Information Sciences Institute Tech. Rep. RFC 793, Marina del Rey, CA 90291, Sept. 1981.
- [13] J. Saltzer, "On the naming and binding of network destinations," in *Local Computer Networks*. New York: North-Holland, 1982.
- [14] H. Zimmerman, "OSI reference model—the ISO model of architecture for open systems interconnection," *IEEE Trans. Commun.*, vol. COM-28, pp. 425-432, Apr. 1980.



**Deborah Estrin** (S'78-M'80-S'81-M'85) received the B.S. degree in electrical engineering from the University of California, Berkeley, in 1980, and the M.S. degree in technology policy and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1983 and 1985, respectively.

In 1987, she was chosen as a National Science Foundation Presidential Young Investigator for her research in network interconnection and security. Her current interests are in technical and organizational issues related to the interconnection of computer networks across administrative boundaries. Her general research and teaching interests include computer networks, computer security, open systems design, and social and organizational impacts of computer systems.

Dr. Estrin is a member of ACM, AAAS, and Computer Professionals for Social Responsibility.