CRATOS
Center for Research on the Applications of Telematics
to Organisations and Society

SUPRANETS

Domenico Ferrari and Luca Delgrossi

{dferrari, ldgrossi}@pc.unicatt.it

Quaderni del CRATOS

Serie di Telematica

CTR-T96-001

Università Cattolica del Sacro Cuore – Piacenza, Italy

Abstract

Many future collaborative environments targeted to facilitate group communications will need to provide, among the other services, a high security level and a high degree of control over the communications. The communication infrastructures of today are either inadequate to address these new requirements or too expensive for wide exploitation by small groups or associations. We introduce the notion of "supranet" as a proposed solution to these problems. Supranets are virtual networks - private to a group - that are built on top of a physical network by means of an appropriate software toolkit. This paper illustrates the motivations behind the design of supranets and discusses how supranets can offer their users the means to achieve the desired security level and to control internal traffic according to a user-defined set of rules.

1 Introduction

To become a true basis for human society in the next century and beyond, the Internet, or any other global information infrastructure to be created in the future, will have to be able to reproduce the most important features of the networkless societies of yesterday and of today. One of these features, whose exercise has even been recognised as a right for individual citizens in many modern constitutions, and which is widely used also in the world of organisations of all kinds, is the ability to form groups. Depending on their context and purpose, these groups are designated with different terms, e.g., association, alliance, congregation, conglomerate, corporation, company of companies, consortium, and so on.

Many of the current users of the Internet (for that matter, of any network) do undoubtedly belong to several groupings. However, the only differentiations that have been so far introduced in the Internet to reflect the existence of such groupings are extremely limited in scope. These include mailing lists, bulletin boards, intranets and administrative domains. A *mailing list* collects e-mail addresses of users with common interests, whose only privilege is receiving all the messages that are sent by other participants in the discussions. A *bulletin board* is an even looser group, in that the messages posted on it can be browsed by any users, and replied to by any users as well; in fact, it may be seen as a group only if we stretch the concept of group to its extremes. An *intranet* is a network internal to an organisation (often, to a geographically concentrated part of an organisation) in which the Internet protocols are run, and that may communicate with the Internet through one or a few gateways ("firewalls") protecting it from unwanted accesses and leaks of information. On the other side, the *administrative domains* group organisations and users having similar characteristics or belonging to the same entity, but this hierarchical arrangement is primarily motivated by addressing, routing, and network management concerns; being members of a domain does not give users any special duties or privileges.

In the future, groups of users of an internetwork will need more extensive support for group collaboration. We feel that a significant step in this direction will be the provision of appropriate tools that allow single groups of users to design, create, manage and tear down their own collaborative environment within a large network. The creation of a collaborative environment - private to a group - within a large network could be seen as equivalent to defining a virtual network on top of the physical one. This virtual network interconnects only the members of the group, carries the group-related traffic being exchanged among them, and may satisfy other requirements to be discussed in this paper. Since it is created on top of the physical network, such a network may be called a “supranet”. We feel that the concept of supranet has the potential to respond well to many of the user needs and requirements in a number of future group collaborative environments.

This paper illustrates the philosophy behind the design and realisation of supranets, and presents some of the guidelines that will influence our research efforts at *CRATOS* in the near future. The rest of this paper is organised as follows: Section 2 introduces a simple scenario and derives a number of user requirements that are the main motivations for the proposed supranet approach; Section 3 defines supranets in more detail and discusses issues related to their design and realisation; finally, Section 4 briefly illustrates the main areas for future work, and Section 5 concludes the discussion with a summary of the main ideas presented in the paper.

2 Motivations and Requirements

In this section, we discuss a number of requirements on communications environments designed to support group-related traffic. These requirements have been the main motivations that led us to the design of supranets. In order to introduce the discussion, we first present a simple business scenario that helps us highlight these requirements. This scenario has been centred around a business application because this is an area that can easily provide many real-life examples where groups of people need to communicate and interact with one another. However, this was certainly not the only possible choice, and we could have easily described a number of other scenarios based on education, health care, or other application areas.

2.1 The “Service Broker” Scenario

The “service broker” scenario is sketched in Figure 1. A service broker (B) would like to offer an extended number of services to his clients (C_1, …, C_N). Thus, he decides to team up with a tax assistant

(T), and two lawyers (L_1, L_2). Upon request, the broker's partners will provide consultant services in their specific areas of competence to the broker's clients. Since all partners are located at distant sites, they will make use of tools such as electronic mail and video-conferencing systems to facilitate communications with the broker himself and his customers. In this arrangement, the broker hopes to grow his business by being able to offer his clients a larger set of services, while the tax assistant and the two lawyers believe they will benefit from the possibility to reach new customers outside their regions of residence.

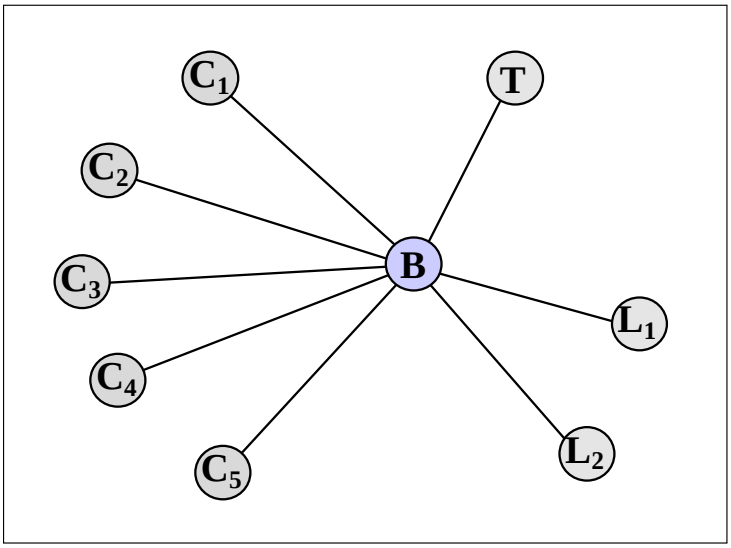

Figure 1: The "Service Broker" Scenario.

To be successful, the service broker needs a collaborative environment, that is, a network, that can facilitate his business communications. Ideally, he would like to have a restricted environment in which access is limited to his partners, his clients and himself. Clients will pay him appropriate fees to become "members" and get access to the services that are made available to them. Non-members are completely excluded and can have no access to the collaborative environment or the services. Since the conversations between a client and his lawyer or tax assistant are of a confidential nature, these communications need take place in a secure manner and be protected against the intrusion of outsiders, or even against the curiosity of other clients. The broker himself guarantees for the security of the environment and responds directly to his customers. By doing so, he feels he adds a value to the services he can offer. Customers should feel they have access to an exclusive environment, where their privacy and confidentiality are guaranteed and well preserved.

But this is not all. The broker would also like to achieve full control over the communications that occur over the network. In fact, he would like to be able to decide, for instance, that interactions among any two clients or two specific clients or the two lawyers are forbidden, or that each client is allowed to reach the tax assistant only through the service broker's node. In short, the broker would like to be able to define a structure that reflects his own business needs, with an appropriate topology and the possibility to control communications over each link.

Finally, but most importantly, given his limited power of investment, the service broker needs a low-cost solution that is also flexible. If the network is sufficiently flexible, it will permit low-cost maintenance and modifications, e.g., when new members have to be added or ex-members removed, or when new needs suggest modifications of the network's topology and connectivity rules. Note that in many applications, not only in this one, dynamic changes with relatively high frequency are the rule, not the exception.

2.2 Main Requirements

The "service broker" scenario highlighted a series of user requirements in this kind of future communications environments. These requirements are often of a "private" nature, that is, they reflect the need for *ad hoc* networks, that can be tailored to the needs of small organisations or even individuals. The scenario also showed a strong need for secure communications at different levels. The latter is a most common and very well known requirement, as demonstrated by the number of current

efforts targeted to the provision of secure commercial transactions over the Internet [1]. In synthesis, we feel that, to take care of the desires of these users, network designers need to address requirements of the following types:

(1) Membership: only "members" should be allowed to access the network, i.e., to make use of the available services. Restricted membership implies the existence of an admission control policy, possibly enforced by a central authority or committee ruling the network. Members may belong to one of several classes, differing from the others in privileges and obligations.

(2) Topology: the paths (consisting of nodes and links) that connect members in the network may be explicitly designed and tailored to specific user needs. This provides the ability to decide what paths should messages follow for security, control, or other reasons. It is desirable that the network's topology be dynamically adjustable based on new needs or convenience.

(3) Capacity: as for paths, it should be also possible for the network designer to define the capacity of the resources involved in the communications. Such resources include the bandwidth of the links, as well as the CPU power and memory size of each node. Resource management functions that enable a dynamic adjustment of the resources' capacity are also desirable.

(4) Security: communications among members may need to be kept private. In such cases, both the intrusion of outsiders and the eavesdropping by other members must be avoided. Network users may need to be guaranteed on the identity of their interlocutors. Different applications may have different requirements: some may request services that guarantee the authentication or the anonymity of the sender, others may need mechanisms that help avoid the denial or the repudiation of service.

(5) Connectivity: this requirement consists of the ability to control which services a member is allowed to benefit from and which other members he can reach and communicate with. This allows the network authority to "hide" services from a member, e.g., when he does not possess appropriate privileges, or to hide the presence of other members, e.g., for confidentiality reasons. Ideally, it should be possible to dynamically establish or tear down connectivity among members.

(6) Multicast: this is the ability to send messages to a subgroup of the members in a secure manner, so that the messages cannot reach a member which is not included in the subgroup. As an alternative, a message reaching an undesired destination should be useless, e.g. indecipherable, for that destination.

To be adopted, a solution of these problems ought to have the following two characteristics:

- Cost-Effectiveness: small organisations or individuals cannot in general afford to build their own physical network. However, also large organisations with greater investment power are normally interested in cost-effective solutions to the problems related to their communications infrastructure.
- Flexibility: this refers not only to the ability to control the network's connectivity, topology, and resources' capacity in a dynamic way, but also to the ability to create and delete the whole environment in a very short time. In a physical network, adding a new link, node or host may be an expensive operation.

2.3 Discussion

If we consider the Internet as it is today, we easily realise that it does not satisfy the set of requirements we have listed in the previous section. This does not come as a big surprise, because the Internet was originally designed to meet very different objectives and, as a consequence, it lacks strict controls of all kind. In fact, there is no authority that can exert those controls. For this very reason, Internet users are often requested to "behave", and control is delegated to individual users rather than built into the network itself. The Internet cannot be considered a "restricted membership" environment: no particular admission policies are applied to decide whether new members are allowed to join. As the Internet is a "democratic" environment, anybody can get access to the available services, and everybody is, at least initially, trusted. As the Internet makes, among other things, eavesdropping and impersonation of other users possible, the confidentiality of conversations is at high risk; thus, the Internet is not considered sufficiently secure for business transactions or other private communications requiring a reasonable security level.

The notion of "connectivity control" is not present either, and it is possible for any user to contact any other user of whom the Internet address is known. Tools are usually available that help

retrieve the Internet address correspondent to a user name. The Internet topology is, for any practical purposes, out of the users' control, and, even if the means are provided to enforce routes and paths, the mechanisms that are destined to this purpose (such as the Source Route option in IP [2]) have been so far used only by network specialists, mainly for testing. As far as multicast is concerned, although the multicast extensions of IP used, e.g., over the Mbone [3] allow for a broad distribution of the data, this does not occur in a controlled and secure fashion; in particular, any user who uses the multicast extensions of IP has access to all multicast communications in progress on the Mbone. In conclusion, the Internet, as it is today, does not respond to the needs of a private and secure environment like the one we feel would support group-related communications well. This is mainly due to its "democratic" (or, more precisely, "anarchic") design philosophy.

As an alternative, a Private Network (PN) or a Virtual Private Network (VPN) [7] [8] can be considered. In a PN or VPN, the network has physical resources associated with it for exclusive use. A VPN is a feasible solution that meets some of the above requirements. However, it is not a cost-effective solution. In fact, VPNs are currently used mainly by large organisations, for instance by multinational companies. In addition to this, the set of services that a generic VPN can offer is limited and, in normal cases, it is not possible to control the connectivity or the topology once the network has become operational. Multicast control is also an issue yet to be addressed and solved in the context of VPNs. Most of all, VPNs are not sufficiently flexible for the purposes of most groups, especially because of their not being created and modified by network clients, but by network providers.

As the Internet and VPN alternatives are considered to be insufficient, we propose a new model, that is, the supranet model. Supranets are "soft" networks, that can be built on top of existing physical networks like the Internet itself. The design of supranets takes into account the needs for low-cost, flexible, private and controlled group-communication environments.

3 The Supranet Approach

The supranet solution to the problem described in Section 2 consists of creating, by using a toolkit available to all network clients, a virtual network on top of a physical one that will have the properties desired for the group for which it is built, i.e., that will satisfy the requirements listed in Section 2.2. To allow such a virtual network with such properties to be constructed, the underlying physical network may have to be modified; for example, this is the case of the Internet, which has been seen in Section 2.3 not to be able to provide the services that are necessary to satisfy the requirements. What makes the supranet solution useful and appealing is its flexibility, richer set of services, and low cost compared with that of a private network designed to satisfy the same requirements. Our idea is that the creation of a supranet can be made sufficiently easy to handle through the use of appropriate tools (cf. Section 3.2), so that any group of users can build, manage, and tear down its own supranet at low cost and in a very short time. As the structure of the group changes (e.g., new members are allowed to join, new rules or security policies are defined, and so on), the correspondent supranet can be adapted to reflect the new requirements.

The first three types of requirements listed in Section 2.2 correspond to the specifications that are needed to design any network: the members of the group correspond to the hosts[1], whose set is restricted to those explicitly listed; and the locations and sizes of the virtual links and routers are determined by the topology and resource capacities assigned by the network designer[2]. Of course, when the description provided for the virtual network to be built is to be mapped onto a physical network, the set of members and their interactions must be protected from outside encroachments. Thus, security from external intruders must be guaranteed. These protections are implicit and compulsory requirements; they are not dependent on the preferences of the supranet's creator. Once the creator has expressed his wishes with respect to (1), (2), and (3), and the implicit protections have been implemented, a virtual network has been fully specified.

---

[1] A "supranet host" is a supranet node with a "member", e.g., a supranet user, on it. If there are no users associated with it, a node is called "supranet router". In this paper, for the sake of simplicity, we assume that each supranet host corresponds exactly to one "member".

[2] The network designer of a supranet is called the "supranet creator". The creator determines the characteristics of the specific supranet to be created and is responsible for its actual construction, regulation and activation. The creator may be an individual as well as a team or committee of more individuals.

The three requirement types (4), (5), and (6) are those that make the virtual network a true supranet. Since the null requirement is acceptable for any of these three types (while it is not for (1), (2) or (3)), there may be supranets that are just simple virtual networks without any special properties. We consider these as degenerate supranets, and those for which at least one of the types (4) through (6) has a non-null requirement as true supranets. Thus, supranets are virtual networks, but not all virtual networks are supranets (for example, the Mbone is a virtual network but is not a supranet).

3.1 Supranet Design

Supranets are exploited by single groups of users connected to a physical network (e.g., the Internet). These users can build, manage, and delete their own supranets by means of a supranet toolkit, for instance by following the steps described in Section 3.2. The supranet toolkit allows the users to specify, at network construction time, the main characteristics of the supranet including its size (in terms of number of hosts attached to the network), its topology, and the desired level of security. The toolkit is able to automatically execute many of the processes required by the construction of the network, so that the user is only asked to specify his preferences on a number of options.

As we have already said, it is in general necessary to modify the structure of the underlying physical network to embed support for supranet services. In the Internet architecture, this can be achieved by inserting a "supranet layer" between the traditional transport and network layers, as sketched in Figure 2 below.

The solution in Figure 2 requires the addition of a supranet layer on top of IP, and the installation of its modified or supplemented version on supranet hosts and routers; it does not require any changes to the applications, to the socket code or to the transport protocols, except for the simple indication that a message is destined to a given supranet rather than to the Internet. A specific format for the structure of the supranet layer header and data is defined. The header may include such information as supranet identifier, upper protocol, supranet destination address and so on. The data may be partially or totally encrypted based on the security level that was selected for the supranet. We illustrate the main functions that are to be included in the supranet layer in Section 3.3.

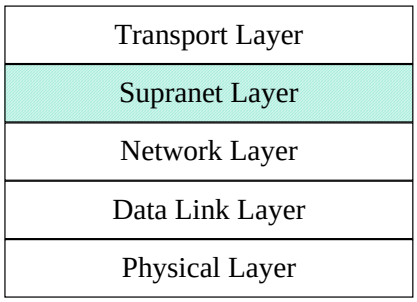

Figure 2: The Supranet Layer.

In a supranet it is possible to run all the existing applications that could be executed over the physical network on top of which the supranet was built. However, in order to fully exploit the functions provided by the supranet layer, it may be necessary to build new applications or to modify the source code of the existing ones.

3.2 The Supranet Toolkit

The supranet toolkit allows the "supranet creator" to establish, manage, and tear down a supranet. The toolkit is designed so that it can be used at different levels, thus accommodating the needs of beginner, intermediate, and expert creators. In the following, we distinguish between *creation/tear down* functions and *management* functions.

The toolkit facilitates the creation phase by providing help in several steps. In a preliminary phase of the construction, it collects general information on the supranet to be built. This knowledge will be used later on either to suggest alternatives to the creator when appropriate or to automatically take decisions in those cases in which the creator wishes to delegate the design of some aspects of the supranet to the system. Such information may include, for instance, type and number of users (both the

initial and the maximum number are to be specified), their class and the corresponding privileges, and such other information as user admission criteria.

In the second step, the toolkit facilitates the definition of the supranet topology. The topology may be entirely defined by the creator or automatically generated by the toolkit, or, again, the creator may want to define the main structure (e.g., the backbone of the virtual network) and let the toolkit complete the design. In this phase, all feasible paths that connect supranet nodes are established. The toolkit automatically generates appropriate routing tables that reflect the creator's decisions about the topology. Note that, within a supranet, routes are fixed, i.e., they never change during the whole supranet lifetime.

The third step is dedicated to the construction of multicast groups. The creator has to provide information on the type and number of these groups. The toolkit will then generate the supranet address space based on the maximum number of users and the maximum number of multicast groups that the creator has requested. Each supranet may have an address space of different size. Supranet addresses are intended to be meaningful only when used in the context of the supranet. Appropriate functions are created that map supranet addresses to the addresses of the underlying physical network (e.g., to IP addresses).

At this point, the creator is requested to express his or her requirements in terms of overall security of the supranet. Several different options are available: the creator will choose whether to restrict security measures to some specific cases or to define a common security level for all the communications that take place within the environment. In those cases where the creator leaves freedom, the users are allowed to decide if and when to use additional security mechanisms.

Finally, the toolkit allows the creator to express the rules that will govern the communications over the supranet. Such rules correspond to a behavior code that must be observed by all users. This is not strictly enforced by the system, though, and it is in general possible for the users to break the law. However, supranets provide the means to detect users misbehavior, and the creator may apply appropriate sanctions whenever such events are detected.

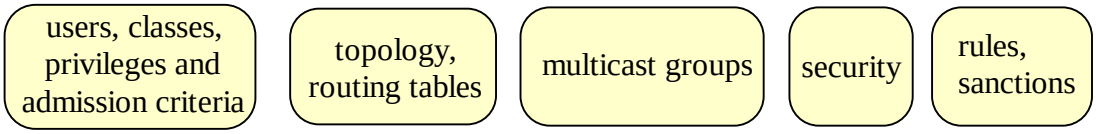

Figure 3: Supranet Construction Phases.

The management functions of the toolkit allow for modifications during the supranet lifetime, like adding or removing nodes, updating the routing tables, detecting rule violations, and so on.

3.3 The Supranet Layer

In this section, we anticipate the main issues to be considered when designing supranets. Please note that the goal at this stage is to discuss these issues and the design decisions that have been taken so far, rather than to provide details to be used as the basis of an implementation of the system.

3.3.1 Address Space

A supranet has its own address space. Supranet addresses are assigned to supranet hosts and routers by the creator. These virtual network components are to be mapped onto physical components; a one-to-one mapping will usually be preferred for the sake of simplicity. Thus, a physical host that is also a supranet host has two addresses, a physical one and a virtual one. Furthermore, a supranet is endowed with its own name space and needs a centralised or distributed name server that translates supranet names (e.g., supranet URLs) into supranet addresses.

3.3.2 Routing

The degree of topological and connectivity control normally desired in a supranet makes it essential for all routes to be fixed. Having fixed routes in a supranet means that, for instance, to reach supranet host 'E' from supranet host 'A', supranet packets always visit supranet routers 'B', 'C' and 'D' in this order (see Figure 4).

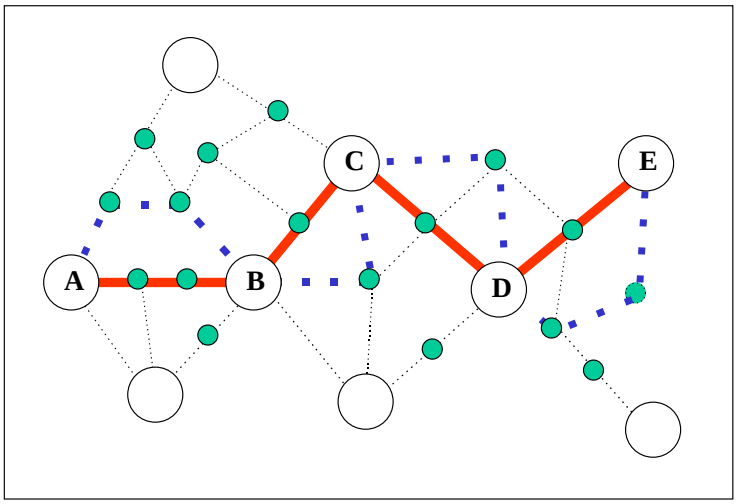

Figure 4: A supranet route connecting hosts 'A' and 'E' through routers 'B', 'C', and 'D'. Large circles represent supranet (and Internet) routers; small circles are Internet routers only. Note that paths AB, BC, CD and DE are supranet (i.e., virtual) links; to each link there correspond multiple physical paths.

When the network on top of which the supranet is created is, like the Internet, connectionless at the network layer, router 'B' may be reached from host 'A' through different physical routes (hence, different Internet routers) for different packets. When, on the other hand, the underlying network is connection-oriented at the network layer (or offers both types of service, as integrated-services networks ought to [4]), also the physical route will be fixed. In all cases, the supranet need not be connection-oriented, as the appropriate static routing tables generated by the creator and stored in supranet routers will be sufficient for the purposes of topology, connectivity and multicast control.

The supranet header, which, in the case in Figure 3, for transmissions from 'A' to 'E' is assembled by 'A' and examined by 'B', 'C', 'D' and 'E', must include the supranet addresses of the source (A) and the destination (E), as well as an "upper protocol" field. The sizes, in bits, of the address fields are dictated by the sum of the maximum number of supranet hosts, the maximum number of supranet routers, and the maximum number of supranet multicast groups. All of these maximum numbers are to be specified at the outset by the supranet's creator.

In the Internet, as shown in the architecture in Figure 2, supranet packets will travel between supranet hosts and supranet routers encapsulated into IP packets. Tunnelling is therefore an important technique in the construction of supranets.

3.3.3 Security

The security requirements of supranets will cover a wide spectrum, given the wide variety of applications of such networks and the dependency of security needs on specific applications. We believe that supranet creators should have access to several different security mechanisms satisfying the many possible requirements, and should be allowed to select those they need and combine them for their purposes. We also believe that supranet members should be able to always add, if they wish, extra security at the application level, and sometimes should be left free to choose the amount of security they want from supranet mechanisms on top of the base amount dictated by the creator. As in most networks, the main security requirements of supranets we expect to be:

a) authentication

b) confidentiality

c) integrity

Authentication mechanisms will prevent supranet members or outsiders from successfully impersonating a supranet member. Confidentiality and integrity must always be enforced with respect

to outsiders (except for messages that are totally non-sensitive) and sometimes even with respect to insiders. To simplify the discussion in this introductory paper, we shall not elaborate on what more precisely will be needed and how it can be obtained in a supranet. Other types of security requirements that are expected to be important in some supranets are non-repudiation, resistance to traffic analysis, and anonymity.

In designing the prototype toolkit, we have made the tentative decision to guarantee integrity to all supranet messages, as we believe this to be a non-negotiable requirement of many important applications. A well-known technique to achieve integrity for a packet is to transmit with that packet its checksum encrypted in such a way as to be decryptable only by the intended recipient (either using a secret conversation key or its private key). In our case, the encrypted checksum will be included in the supranet header. Note that the same technique provides (at zero or very low cost) authentication of the sender to the receiver. Thus, our decision means that all supranet messages will also be authenticatable. Of the main requirement types, only confidentiality remains: our tentative decision has been not to allow the creator to impose confidentiality on all communication whenever this is felt desirable; the creator may only set rules, indicating what types of messages ought to be confidential, and sanctions against those who do not follow those rules (see Section 3.2.5 below). Confidentiality may be obtained by the user through application-level mechanisms, but also through supranet-level encryption of the payload, so that only the receiver will be able to decrypt it.

In the Internet, these mechanisms could in principle be replaced by those recently proposed at the IP level: the Authentication Header (AH) [5] and the Encapsulating Security Payload (ESP) [6] More precisely, AH can provide authentication and integrity for IP datagrams; with some cryptographic algorithms and keying techniques, it may provide also non-repudiation, i.e., the ability by a receiver to prove that the sender of a certain message did indeed send it even though the sender denies having sent it. The ESP mechanism provides confidentiality and integrity to IP datagrams; with some cryptographic algorithms and algorithm modes, it may also provide authentication. AH and ESP may be used alone or together.

Furthermore, access to supranet-owned information must be controlled. The creator is in charge of setting up the access control list for each repository of such information, taking into account the duties and privileges of the various classes (if any) of supranet members. In our current design, access control lists take the form of bitmaps, one for each repository and for each type of access. A packet's supranet header includes a bitmap indicating what classes the sender belongs in; to prevent a sender from altering it, this bitmap is given to them by the creator and is encrypted (together with the sender's supranet address) using the creator's private key. Coupled with authentication, which we have made (in the prototype toolkit) universal, access control lists should make unauthorised access practically impossible.

3.3.4 Multicasting

As supranets provide a controlled communications environment, it has to be possible to achieve control also over multicast forms of communications. To this purpose, the supranet creator is allowed to specify a number of multicast groups, and to assign a supranet multicast address to each of them. Since supranet membership is restricted, it is possible in general to build multicast trees that define the paths from each member of the multicast group to the others. When building such paths, appropriate mechanisms exist that can be used to minimize the overall group traffic. Every change in group membership requires a modification of all the trees pertaining to the multicast group. Additional security mechanisms, such as private keys for each group, will be used to provide security services at the multicast group level.

3.3.5 Rules and Sanctions

In human societies, groups and associations are governed by a number of rules. In a similar way, the operation of a supranet is governed by rules dictated by its creator. The construction of a supranet requires a number of choices concerning necessity and freedom to be made by the creator. Some actions or prohibitions will be made compulsory by the absence of alternatives. In other cases, the creator will choose to provide alternatives, set rules, and let users decide whether they wish to follow the rules or not. Those who do not will incur into well-publicized sanctions. Obviously, in order for sanctions to work, user behavior must be monitored; non-monitorable actions have therefore to be made compulsory. Rules are concerned with admission, access rights, use of supranet resources, relationships among members, etiquette, and so on. For instance, leaking protected information to the outside will generally be prohibited.

3.3.6 Supranet Modifications

The creator's initial specifications for all user requirements cannot be assumed to be perfect or immutable. Changes will have to be made to them during the supranet lifetime due to the addition of new members, the departure of old members, the discovery of new needs or of mistakes in the previous specifications, and so on. The creator will have to be provided with tools facilitating all reasonable modifications of the requirements on which the supranet has been based.

In summary, functions to be included in the "supranet layer" include: definition of the address space for the supranet (both unicast and multicast addresses), creation and management of supranet-level routing tables; assembling supranet packets and forwarding them to the next supranet node on the way to the final destination; encryption of a portion of the packet (or all of it) as requested by the security level associated with the supranet; support for multicast communications; checkpoints, filters, and other forms of control to detect users misbehaviors.

4 Future Work

At *CRATOS*, we intend to further study the effectiveness of supranet-based solutions with respect to the needs discussed in Section 2. At this stage, we are working on the full design and specification of supranets that can be mapped on top of the Internet. Our current approach positions the supranet layer just above the IP layer, as shown in Figure 2, and considers some of the existing techniques to achieve secure communications for adoption within the supranet framework.

In parallel with this effort, we intend to design and build both a number of applications that can exploit supranet services and a rich set of tools that facilitate the tasks of the supranet creator during a supranet's construction at the beginning as well as its management and control once the network is fully operational. On a longer term, we would like to examine how well can supranets cope with complex communication issues such as members mobility, supranet hosts or routers failures, and how extensible the supranet model is when considering large scale supranets with a very large number of hosts and routers.

5 Conclusions

This paper introduced a number of requirements on group collaborative environments that are felt to be highly relevant and need to be addressed in the near future by network architects and designers. The primary requirement is for a fast way to satisfy the others and to adapt the system rapidly to the changes in group needs. The "service broker" scenario served the purpose to highlight and motivate some of these requirements by considering a potential real-life application. The discussion that followed demonstrated that today's systems are either inadequate to meet these requirements or too expensive for a wide adoption.

Hence, the supranet approach was proposed as a feasible and cost-effective solution and was presented at a high level. Supranet are "soft" networks, that are created and deleted using a toolkit available to all network users, and are mapped onto a physical network such as the Internet. We argued that such toolkit can be built and it can effectively be employed by most users to build supranets. Membership in a supranet is restricted, and the supranet environment, including connectivity, topology, resource capacities and the level of security, is controlled by the supranet creator. The creator defines a set of rules to govern the supranet communications, and is allowed to impose sanctions on those members who do not respect these rules. The creator may be an individual or a committee. We feel that supranets have the potential to respond well to many of the user needs and requirements for future group collaborative environments.

6 Acknowledgements

We thankfully acknowledge Andrea Mills, who brought to our attention a scenario very similar to the "service broker" scenario described in this paper, from which the idea of supranet was born, and Maurizio Galli and Barbara Rossi for the many productive discussions and criticisms to the approach, which have been essential in the verification of the ideas presented in this work.

7 References

[1] A. Bhimani: “Securing the commercial Internet”, Communications of the ACM, Vol. 39, No. 6, June 1996.

[2] J. Postel: “Internet Protocol”, RFC 791, DARPA, September 1981.

[3] M. R. Macedonia, D. P. Brutzman: “Mbone provides Audio and Video across the Internet”, IEEE Computer, Vol. 27, No. 4, April 1994.

[4] D. Ferrari: “Should an Integrated Services Internetwork be Connectionless or Connection-Oriented ?”, 6th International NOSSDAV Workshop, pp: 3-4, Zushi, Japan, April 1996.

[5] R. Atkinson: “IP Authentication Header”, Internet RFC 1826, Proposed Standard, August 1995.

[6] R. Atkinson: “IP Encapsulating Security Payload”, Internet RFC 1827, Proposed Standard, August 1995.

[7] S. Fotedar, M. Gerla, P. Crocetti, L. Fratta: “ATM Virtual Private Networks”, Communications of the ACM, Vol. 38, N. 2, pp: 102-109, February 1995.

[8] J. M. Schneider, T. Preuss, P. S. Nielsen: “Management of Virtual Private Networks for Integrated Broadband Communication”, Proceedings of the ACM SIGCOMM’93 Conference, pp: 224-237, September 1993.