The Virtually Invisible Internet

Suresh Singh
Department of Computer Science
Portland State university
Portland, OR 97207
Email: singh@cs.pdx.edu

Jim Binkley
Department of Computer Science
Portland State university
Portland, OR 97207
Email: jrb@cs.pdx.edu

***Abstract*—We consider the question ”What if we could redesign the Internet, how would we do it?” The preliminary proposal discussed in this paper is based on two observations – today, and in the future, wireless access by users will be the predominant mode of communication and communication typically follows a group-pattern where groups of people and/or entities engage in a specific communication activity. Using these two observations, we propose a novel group-based architecture with the goal of making the Internet more usable as well as invisible to users.**

I. Introduction

In the networking community today, there is an ongoing discussion around the question: “if we had to design the Internet again, how would we do it?”. Numerous researchers have weighed in with different requirements that such a design should satisfy and there have also been many proposals for designs (or components). Our goal in this paper is not to extensively survey the various designs, but rather to add our $0.02 to the discussion. We note that the proposal here is modest and should not be thought of as a complete solution but rather as a piece of one.

The starting point for our discussion is by asking the question: “what kind of an Internet do people really want?” It appears that people don’t really want to know about the “Internet” in the sense that we, in the networking community, do. For instance, the mechanical aspects of today’s so-called cool activities like downloading music to a laptop, sharing, gaming, and so forth may well be viewed a decade or two later as primitive. Rather, in our view, what people want is to have the Internet be there just like air is – you use it constantly without being aware of doing so. An example of a similar trend in another domain is the replacement of film cameras with digital – the point always was to take pictures, the technology just got in the way. Likewise, people want to participate in all the social and information-based activities made possible by the Internet, they probably would just as well not know how it works.

In the remainder of this paper we provide a partial design of some aspects of such an Internet. In section II we argue that the fundamental unit of communication should be a group and consider routing in this context. Next, in section III we describe our view of the Internet architecture that can enable our vision of an invisible Internet.

II. Premise 1: Group-based networking

In order to motivate the design, a useful question to ask is, who are the power users of the Internet, today and in the future? The obvious answer, the people, is only partially correct . Indeed, other power users include autonomous systems like national defense or global banking systems with their automated trading widgets, and so forth. In all of these cases, we note that the mode of operation involves communication within well-defined groups[1]. In the case of people, examples of groups include online gaming communities and social networks, or members of a family sharing photos, or even a person surfing the web. In the latter case, the group is trivial and consists of two entities whose composition changes constantly. In other cases, like the banking systems or sensor networks, groups are persistent. Another characterstic of groups is the barrier to entry into a group. For banking, this barrier is quite high and typically uses out-of-band mechanisms to change the group’s composition. However, for online gaming (and possibly for commercial sensor networks that provide specific services), the barrier to entry is very small and may at most include a fee for service. A third charactersitic we note is the composition of a group itself – the entities may not always be people, they can be widgets or even groups. Based on these observations, we can write:

[1] We are not referring to multicast groups but to a set of communicating entities.

Group	$\leftarrow$	$\{$Collection of Entities $\mid$ Groups$\}$ $+$ $\{$Rules for Communicating$\}$ $+$ $\{$Group Attributes$\}$
Group Attributes	$\leftarrow$	$\{$Security, Location Database, etc.$\}$
Entity	$\leftarrow$	$\{$id, Location, Interface, etc.$\}$

Security is a group attribute because each group is free to define security among its members to suit the underlying application needs. A good example to keep in mind is a group consisting of a set of banks[2]. The location attribute will be discussed later in section III. An entity's interface attribute refers to the means for communicating with that entity and will depend on its implementation (for instance, we can think of a set of "methods" for communicating with an entity). Figure 1 illustrates a simple group structure. We see that group g_1 contains five members – e_1, e_3, e_7, g_4, g_5. Group g_5 in turn contains e_4 and g_6, and so forth. Observe that e_3 belongs to g_1 as well as g_4 (this is indicated by replicating e_3 in the figure). However, it has different rules of communication and privacy in each group.

Some examples of groups were provided earlier. However, the following example relates to our original motivation of making the Internet useful but invisible from an individual's point of view. Today, individuals possess multiple communication devices as well as email addresses and other remote widgets with which they communicate. Thus, we may now view the individual as a group rather than a person (or an IP address). The characteristics of the group are defined in some way including restrictions on membership, etc. The communication between members of this group is part of the group specification and thus, if a message needs to be delivered to the individual, then the group appropriately forwards it to the device the person has at that time. If the person buys a new device or replaces an old one, then the appropriate changes are made to the group membership. We note that this specific example of a group is not new – one of the goals of Bluetooth was to integrate multiple devices via a PAN (Personal Area Network). However, the group-based design we discuss is fundamentally different and can certainly use the PAN for communication.

As another example, consider today's Intranets (corporate, organizational, universities, etc.). These form a large component of today's Internet and their existence appears to be a side-effect of how the Internet evolved. Indeed, there appears to be no real reason to preserve this structure. We believe that groups ought to be the fundamental defining unit of the Internet. Thus, a university can be thought of as a group (all people, their various devices, all the databases that may be located somewhere else, etc.) with well-defined ways of communicating within the group. In this model, the group members are physically present anywhere but interact with one another as if they were co-located in an Intranet.

A. *Information Security*

The group structure we propose provides a natural way for implementing information dissemination boundaries and hence security. Thus, when a group is formed, we assume that the group members agree to a set of rules on information sharing. In other words, some information may only be shared within the group while other information may be shared with outside groups. Consider a few examples that explain the implications of this idea better. Consider a group g_1 that consists of all Banks and a group g_2 which consists only of branches of a specific bank (g_2 is fully contained within g_1). Information about an individual's credit report may be shared in g_1 but account information may be restricted to g_2 only. As another example consider two disjoint groups g_1 and g_2 such that $g_1 \cap g_2 \neq \phi$. In this case, we have the situation that information may *leak* between the two groups. This problem has no general solution except to ensure that the entities which lie in both groups self-manage so that information does not leak.

B. *Routing*

In order to explain how routing works within this group structure, it is useful to first discuss different routing scenarios. Using the notation of e for entities and g for groups, we have four possible cases:

1) *e2e:* The simplest example is when we have a group that corresponds to an individual and a message from an entity in the group needs to be delivered to the device (entity) that the individual has with her at a given time instant. Another example corresponds to a phone call between two individuals – here the two entities are the two phones but they belong to different groups (since each individual has their own group).

[2] We note that VPNs today can be thought of as an example of secure communication between members. However, the functionality is far less than in the case of group-based communication.

2) *e2g:* Say A wants to send a message to B. In this case, A uses some device (say a cellphone) and generates a message that gets delivered to B's group. After some entity in B's group receives this message, it then sends it on to the device that B currently has handy – this is a case of e2e routing.
3) *g2e:* An example of this may be automatic software updates of devices. Thus, if some large Seattle-based company needs to update software on devices of its manufacture, it will use g2e routing where the origin of the message can be any one of several entities belonging to the company.
4) *g2g:* When one bank exchanges information with another, that is an example of g2g routing. How this information then gets dispersed within a bank's various branches will correspond to a case of multiple e2e routing (perhaps implemented via multicast routing) where the origin is the entity that first receives the information on behalf of the bank's group.

For any of these forms of routing, we next need to understand how a route is created and implemented. For this, let us consider various cases from the example illustrated in Figure 1.

- $e_1 2 e_8$: The route here can be written as $e_1.g_2.g_3.e_8$. The routing between e_1 and g_2 corresponds to routing between e_1 and e_7 in this example.
- $e_1 2 e_6$: Again, we can write a route as $e_1.g_5.g_6.e_6$. Since e_1 and g_5 belong to the same group g_1, we assume the existence of some mechanism (described later) to route between them.
- $g_6 2 g_3$: A possible route is $g_6.e_4.e_3.e_7.e_2$. The route ends at e_2 since $e_2 \in g_3$ which is all that is required.

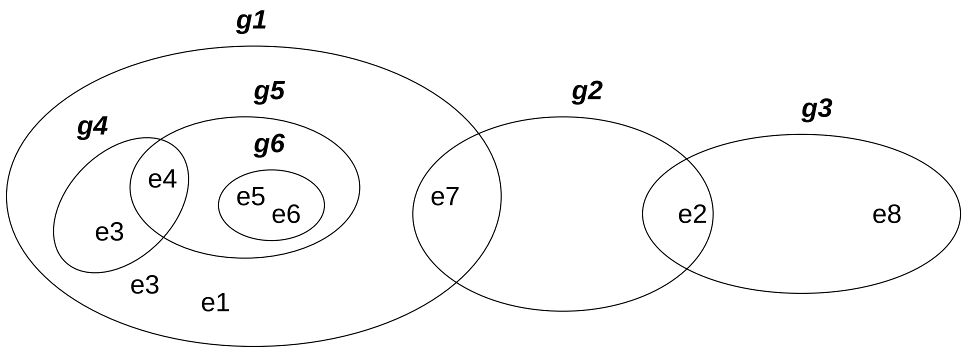
We note that routes in this model are not unique. Indeed, on the level of the Internet, we can easily imagine a combinatorially large number of routes between pairs because of the huge number of potential groups. However, we consider this to be a plus point of the design since it ensures that DOS attackers will have a hard time plugging up a single route to an organization.

The discussion above has ignored the problem of physically routing packets between end-points. Indeed, since information is created by entities and is consumed by entities, any routing scheme has to eventually perform $e2e$ routing of packets. The approach we propose is to define *infrastructure groups* where the entities are routers. Each group of these routers *knows* how to route packets between entities that belong to the group. In other words, each member of a group maintains a routing table for all members of the group. It is easy to see that this structure is similar to today's Internet structure with groups at some level of the hierarchy corresponding to ISPs. However, in this paper we are not constraining ourselves to the present-day Internet structure and therefore we have more flexibility in defining the infrastructure groups. We discuss this in more detail in the context of the Internet architecture in section III.

Given the existence of an infrastructure group structure routing can now be performed as follows. Note that every entity has to physically exist somewhere: software agents exist at some computer, physical devices (like cellphones) physically exist somewhere in the network, and so forth. The idea then is to identify the physical location of these entities with entities of the infrastructure groups. Once this is done, routing follows a simple algorithm as shown in the example of $e_1 2 e_8$ routing in Figure 1. Assume that the route provided (let us assume loose source routing) is $e_1.g_2.g_3.e_8$. Figure 2 shows seven routers $a - g$ with connections as shown. Each entity is associated with exactly one router. Each group g_i, on the other hand, is associated with all routers that members of g_i are associated with. Finally, we assume that there are four infrastructure groups $Ig_1 - Ig_4$ (in general we may have arbitrary nesting of infrastructure groups as in Figure 1, however for clarity we have not shown that case here). In Ig_1 each of the four routers a, b, c, d have a complete routing table which describes how to route between any e_i and g_j that are associated with these routers. Thus, the $e_1.g_2$ part of the path is translated into a route $a - b - c$. The $g_2.g_3$ path is translated into route $c - e - f$, and so forth. The final translation is:

$$e_1.g_2.g_3.e_8 \rightarrow e_1.a - b - c - e - f - g.e_8$$

We note that this route is not the shortest one possible, $a - b - d - f - g$ is one hop shorter. However, since we are constraining the route based on the path specified, we cannot go through router d in this example. An interesting optimization in the group definition and usage is to explore ways to optimize routes. A proposal we have is to artificially associate entities with more than one router. In other words, if we associate e_7 with d in addition to c, then g_2 gets associated with d as well and we can use the shorter route. However, two points need to be kept in mind: first, e_7 is physically associated with c and thus to get data to/from e_7 we need to go through c; and second, if we also associate e_7 with d then the network needs to maintain state which will allow packets



g1 = {e1, e3, e7, ***g4, g5***}; ***g4*** = {e3, e4}; ***g5*** = {e4, ***g6***}; ***g6*** = {e5, e6}
g2 = {e2, e7}; ***g3*** = {e2, e8}

Fig. 1. Routing examples.

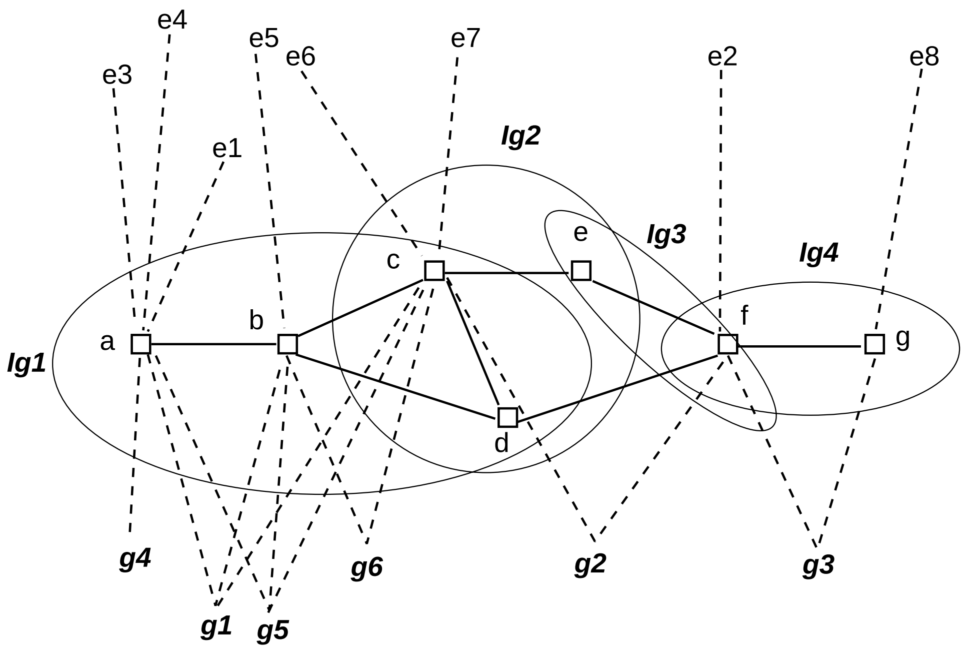


Fig. 2. Associating entities and groups with routers and infrastructure groups.

to e_7 to be forwarded from d to c. This is a network layer optimization which has associated overhead.

C. DOS-proofness

In today's Internet, organizations are vulnerable to DOS attacks because they have a very small number of points of connection to the Internet that can be easily overwhelmed. In the group-based model, on the other hand, an organization exists as a large group (consisting of multiple entities as well as groups) that by its very nature is physically distributed. A DOS attack to this organization results in packets being routed to any of the group members and thus gets harmlessly dispersed.

III. Premise 2: People-based Architecture

People form the largest community of users of the Internet and as such the Internet ought to be designed with their needs in mind. Some specific needs include very high data-rate connectivity and wireless rather than wired connectivity. Thus, the best architecture that meets these goals is to have high density wireless access points deployed to allow user's to remain connected at all times and a supporting high-speed wired infrastructure. If we think in terms of a graph, we can imagine a graph with high capacity links forming the backbone plus special access nodes that have a very large out-degree. These access nodes are the on-ramp to the wired portion of the network and they connect to a large number of access points. Figure 3 illustrates this architecture. As shown, the Internet can be viewed as a wired part surrounded by a large wireless cloud. At the level of a city, the more detailed structure consists of densely distributed access points all connected to a wired backbone. Other components (not shown) include wired computers, storage centers, etc.

Within this architecture, we believe in providing zero-barrier wireless connectivity to users. Connecting to the Internet wirelessly today requires a fair degree of awareness of the process – hot-spots require some minimal user participation, possibly entering a password or otherwise navigating the available options. This model for providing connections is intimately related to the technology used at the access points as well as the ownership of the hot-spots. In our opinion, this form of connectivity is quite restrictive and thus will have a limited life. Consider an alternative model, which is an order of magnitude increase in scale as compared to various public WiFi networks being deployed today. Say access points are deployed with a very high density and are capable of supporting multiple radio protocols and frequencies (perhaps using software-defined radios). User devices remain connected automatically to one or more such hot-sopts all the time – there is no authentication or any other handshaking protocol involving the users.

This view of connectivity may appear to be highly insecure but is not. We believe that security should be provided at the level of groups where group members implement their own forms of authentication and encryption. Indeed, as has been noted by others, security between strangers (such as between an AP and a mobile device) is no security at all. Therefore, there is little point in providing that form of "security" since the cost of doing so is a significant amount of overhead as we see today.

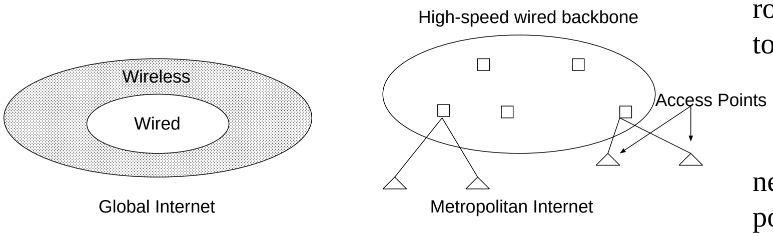


Fig. 3. High-level view of the Internet.

A. Location Database

We believe that a large fraction of data will originate and terminate at mobile end-points of the Internet because users will predominantly be mobile and will use multiple devices for connecting. Thus, we need to design for efficient routing in the face of user mobility. This problem has received a great deal of attention in the mobile computing community where solutions such as Mobile IP [7] were developed. However, we note that these solutions were developed assuming that mobility is the exception rather than the rule. In our architecture, on the other hand, mobility is the norm and we need to revisit routing from this point of view.

Two key architectural features in our design are the following: routing can specify a group as the destination (anycast) and entities are associated with routers but they may move. Since association of entities to routers is dynamic, we need to maintain location databases (as are maintained by cellphone providers). However, given the huge number of such entities (each person may have tens of entities at least), maintaining such a database is challenging. We believe that solving this problem is the key part of routing. An interesting subproblem we note is routing within groups. We assume that each group maintains its own location database for this purpose. Thus, as members of a group move, this location database tracks their movement and thus this database can be considered to be correct at all times. Other location databases (at the level of a city or region) lazily collect information from these group-level databases.

B. Naming

Unique naming is a fundamental requirement for any such network. In the spirit of not imposing any apriori structure on the Internet, we propose names of entities and groups be random strings. The only drawback with this scheme is that the name now provides no clues for routing. However, we believe that the combination of random names, location databases, and the loose source routing scheme described previously can work efficiently together.

IV. Related Work

[2] lists a set of requirements that a future Internet ought to meet. These include support for mobility, policy-driven auto-configuration of the backbone, support resources that change over time (e.g., point of connection), allocate capacity for fairness or other policies, and support long propagation delays for space-borne

nodes. It is interesting to note that while very insightful, this document still presents incremental changes to the present-day Internet. A reason is that in the six years since the document was written, previously unthought of applications have emerged. Indeed, as Web 2.0 now gets into its fastest growth spurt, we believe that many more disruptive applications will emerge to make us revisit the design question.

[4], [6] describes an architecture based on a model called FARA (Forwarding directive, Association, and Rendezvous Architecture). The idea is that routing is no longer between hosts but between entities. An entity can be a process, a thread, a machine or even a cluster. Routing does not use IP addresses but rather FDs (Forwarding Directive) which are descriptors that contain enough information to get the packet to its destination. FDs are discovered through a directory service. We note on some similarities between our architecture and FARA. Specifically, the use of entities rather than host computers as end-points for communication and the idea of source routing. However, we go many steps further in (1) breaking from the ISP based architecture (which FARA still uses) and (2) in creating the general group-based communication model. [3] was an early attempt at considering the routing problem in a very large Internet. The idea is to use link state maps distributed through the Internet. Route generation and selection may be based entirely on user needs (QoS, pricing, etc.). To manage the scaling problem, Nimrod uses clustering which reduces the number of entities visible to routing while still providing flexibility in route selection. We note that our architecture is quite different from Nimrod since we focus on group-based communication and indeed place no restrictions on group formation (i.e., we are not explicitly requiring clustering).

[9] is not directly relevant to our paper but is interesting in that it proposes exposing routing to end users and giving them the ability to dictate inter-domain routing. The motivation is to spur innovation in the ISP architecture. [5] is an interesting approach for thinking about designing the future Internet. They propose we first define a set of invariants that must be satisfied and then derive an architecture from those invariants. Approaches such as SILO [1] approach the problem from the point of view of services. Thus they propose a vertical stack of data transformation methods to implement services which are provided by agents. In this view, the current Internet appears as one instantiation of the methods. [8] presents an opinion about the ongoing discussion on a future Internet and concludes by discussing emerging research issues. It is a good reference to motivate this area of research. One motivation that we have ignored but is discussed by most of the references is the economic incentive to ISPs to reinvent the Internet. While this is an important question, we ignore it so as to be unconstrained in our thinking.

V. Discussion

The overall architecture we propose here moves towards a human-centric view of communication – that of a group. Some of the challenges of realizing this structure are discussed in this paper. However, two important questions that we have ignored include scalability and evolvability. By evolvability we mean the ability of group-based communication to support future, hithero unknown, modes of communication. In terms of scalability, we believe that the primary challenge lies in designing very large location databases. The evolvability question, on the other hand, is not as easy to answer. An approach that may help us is to use language-based abstractions to define arbitrary modes of communication and then to see if they can be simply mapped to the group model presented in this paper. This is work that we are currently pursuing.

Acknowledgement

This work was supported by the NSF under award NeTS-NR: 0435328.

References

[1] Llia Baldine. Future internet architecture: Issues and non-issues, the silo perspective. Panel DIscussion, BROADNETS, 2006.

[2] R. Braden, D. Clark, S. Shenkar, and J. Wroclawski. Developing a next-generation internet architecture. DARPA White Paper, July 2000.

[3] I. Castineyra, N. Chiappa, and M. Steenstrup. The nimrod routing architecture. RFC1992, August 1996.

[4] D. Clark, K. Sollins, J. Wroclawski, and T. Faber. Addressing reality: An architectural response to demands on the evolving internet. In *Workshop on Future Directions in Network Architecture (FDNA'03)*, August 2003.

[5] B. Ahlgren et al. Invariants: A new design methodology for network architectures. In *Workshop on Future Directions in Network Architecture (FDNA'03)*, August 2003.

[6] D. Clark et al. Newarch: Future generation internet architecture. Technical report, USC/ISI, December 2003.

[7] C. Perkins. Ip mobility support, rfc 2002. Technical report, IETF, 1996.

[8] T. Roscoe. The end of internet architecture. In *ACM HotNets-V*, 2006.

[9] X. Yang. Nira: a new internet routing paradigm. In *Workshop on Future Directions in Network Architecture (FDNA'03)*, August 2003.