Blossom: A Decentralized Approach to Overcoming Systemic Internet Fragmentation

Geoffrey Goodell
Scott Bradner
and
Mema Roussopoulos

TR-25-04

Computer Science Group
Harvard University
Cambridge, Massachusetts

Blossom: A Decentralized Approach to Overcoming Systemic Internet Fragmentation

Geoffrey Goodell
goodell@eecs.harvard.edu
Harvard University, Cambridge, MA

Scott Bradner
sob@harvard.edu
Harvard University, Cambridge, MA

Mema Roussopoulos
mema@eecs.harvard.edu
Harvard University, Cambridge, MA

November 1, 2004

Abstract

The Internet is systemically fragmented. In this paper, we examine the causes of fragmentation, including both technical concerns, such as middleboxes and routing failure, as well as political concerns, such as incomplete peering and the structure of Internet governance. While fragmentation may be desirable in certain circumstances and for various reasons, it can also be problematic, violating central Internet design principles and rendering routine tasks difficult. We motivate the need for a system designed to facilitate connectivity throughout the Internet, providing the benefits of locality, universal access, and distributed management, while interoperating with the existing infrastructure. We describe the research challenges in building such a system and outline our research agenda.

1 Introduction

The Internet consists of a network of peer nodes, arranged such that each peer node has access to a certain set of *resources*. The essence of Internet design is to provide a system in which each peer node can access the same set of resources and each resource has a globally unique identifier. The modern Internet does not allow for such an arrangement of peer nodes and resources. Instead, the set of resources to which a given node has access is a function of its location within the network topology. That is to say that the network itself acts to limit access to particular resources in such a manner that access to certain resources is restricted to peer nodes in particular locations. In this sense, we say that the Internet is *fragmented*.

There are many causes for fragmentation, ranging from accidental (routing failures, misconfigured policies, unreliable network elements) to deliberate (content filtering, network address translation, firewalls, malicious service providers). We assert that fragmentation is inevitable and we describe how prevailing Internet architecture fails to avoid fragmentation despite various efforts to provide consistency.

In this paper, we describe our vision of an architecture in which not all participants have the same idea of the set of resources provided by the Internet, what region of the Internet constitutes the "core", or which set of real-world organizations are responsible for Internet governance. We seek to promote the idea of an Internet whose management reflects the management of the physical world rather than imposing organizational structure where such structure need not exist. We also seek to provide a means by which the names used to identify resources in one area of the Internet do not unnecessarily restrict the names used to identify resources in other areas.

We propose Blossom, a peer-to-peer system of *forwarders* aimed at realizing this vision. Blossom forwarders act as intermediaries between nodes that cannot communicate directly, using an unstructured overlay to provide peer-to-peer communication across middleboxes. We show how Blossom can be used to defeat both purposeful and accidental fragmentation, and we show how it is possible to use the same system to present the Internet from different points of view.

One of our primary goals is to make it easier for providers to design reasonable policies that are less tightly integrated with routing mechanism. We hope to achieve this goal by providing an infrastructure that allows connections between networks with locally meaningful identifiers that does not draw a distinction between "internal" and "external" networks. We thus wish to devise a useful tool for allowing individual internet peer nodes to provide policy-compliant access to services without requiring special configuration of the network infrastructure. One might argue that this technology can be used to provide open access to networks protected by firewalls, but the reality is that technology that can create tunnels through firewalls exists today. What our system provides is a means by which otherwise-inaccessible resources can be accessed in a general way, without preordained arrangement between nodes on both sides of the obstacle. Ultimately, Blossom is a technology to enable a transition to better policy; because forwarders are not part of the underlying network-layer routing infrastructure, they can be deployed in a manner that allows for policy settings that are both more flexible and closer to the endpoints.

The rest of the paper proceeds as follows. In Section 2, we examine the causes of fragmentation and its natural inevitability. In Section 3, we describe the requirements a system should satisfy to overcome fragmentation. In Section 4, we describe Blossom, the solution we envision and the research challenges towards achieving this vision. In Section 5, we describe previous work addressing problems related to network fragmentation. Finally, in Section 6, we conclude with a brief description of current status of this work.

2 Causes of Fragmentation

Systemic Internet fragmentation has many causes, including but not limited to the following:

Interdomain routing misconfiguration. Misconfiguration of routers that participate in the Border Gateway Protocol [16, 19] is a signficant cause of observed routing failures including unreachability and suboptimal routes [15, 11, 13, 9].

Interdomain routing instability. Interactions between BGP policies often lead to persistent interdomain routing oscillation [21, 10] and interdomain routing oscillation in turn leads to degraded network performance [11].

Interdomain routing policy. Accidental misconfiguration of interdomain routing policies, as well as purposeful configuration of these policies due to financial or political reasons, can lead to an incomplete set of available routes.

Firewalls. Firewalls deployed at routers block incoming traffic from entering the network behind the router. A firewall can thwart attacks that involve random automated probes for services with vulnerabilities over an address range, but this comes at the cost of of effectively enacting a policy, even when such policy is not required by organizational goals.

Network Address Translation. Organizations often deploy NAT for the same "security reasons" used to justify the deployment of firewalls. NATs violate the end-to-end principle [17], translating endpoint identifiers so that peer nodes do not know with whom they are communicating and making systems at the ends of the network dependent upon the reliability of systems in the interior of the network.

Content Filtering. Some firewalls and other devices are configured to filter packets based upon application-layer content (for example, filtering HTTP requests for particular URLs); this technique can be used for large-scale censorship of sensitive content [4]. The scale at which governments and providers are considering deployment of such technologies indicates the potential for misuse.

Explicit Address Filtering. Internet service providers may also configure their routers to explicitly block packets based on source or destination addresses. Deploying such filtering technology in the middle of the network suggests greater distance between those subject to the policy and those enacting it.

Transparent Proxies and Caches. Transparent web proxies often cache replies to improve performance, but may fail to issue cache-control directives to specify that a user does not want a cached copy [1]. Moreover, proxies pose a threat to network transparency by injecting intelligence into the center of the network [3].

Anycast. In anycast, the network chooses which server to forwarder a request to on a client's behalf. While this might have certain efficiency benefits, the client is unable to specify the exact server it desires to provides the service, and the end-to-end principle suggests that this should be possible [17]. Also, peers have no means of determining to which server it is conversing in case of problems.

DNS hacks. DNS can be used to provide different IP addresses for the same hostname based upon the location of the requester in the network. This makes it hard for the requestor to discover and access a resource at an IP address that is in a remote location.

3 Design Goals

We intend to thoroughly explore the problem space associated with fragmentation. We submit that the amorphous nature of the Internet facilitates its growth, that fragmentation is part of this amorphous nature, and that designing an architecture that acknowledges fragmentation as a fundamental characteristic of the underlying network would have a number of benefits. We will design our system around four central objectives:

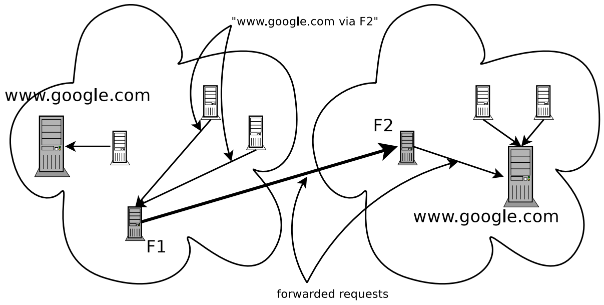

Figure 1: Locality. *Multiple services with the same name may coexist within different local namespaces.* (Meaningful names within a local space.)

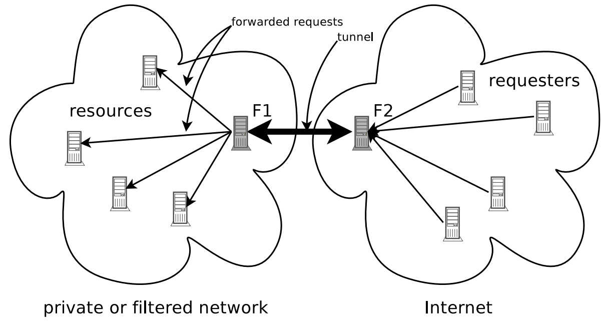

Figure 2: Universal Access. *If two hosts can both access forwarders within the same forwarding infrastructure, then those two hosts can use the infrastructure to communicate.* (Circumvent technical barriers.)

Locality. *Multiple services with the same name may coexist within different local namespaces.* In the physical world, the meaning of a name is dependent upon its context. However, the existing Internet paradigm intends for there to exist a global namespace in which centralized authorities allocate names hierarchically and uniquely. A system that facilitates communication across network fragments may also allow for the development of distinct local namespaces, in which names have local meaning, while also allowing access to objects in other namespaces that happen to bear the same name. This may afford businesses the opportunity to protect their trademarks, avert some Internet namespace arbitrage, and generally lead to relaxation of an unnatural constraint on naming.

Universal Access. *If two hosts can both access forwarders within the same forwarding infrastructure, then those two hosts can use the infrastructure to communicate.* Sometimes, communication between networks is compromised for architectural convenience rather than policy reasons. We would like to provide an architecture that facilitates the use of intermediaries to allow communication between entities that for whatever reason cannot communicate directly.

Distributed Management. *Adding a network and its abundance of resources to the system need not require specific allocation of names, addresses, or routing from centralized au-*

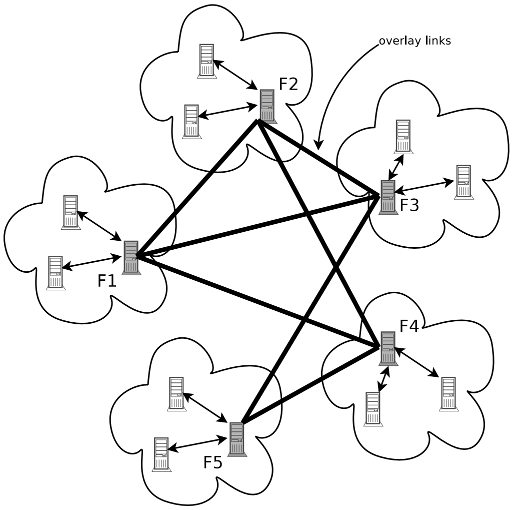

Figure 3: Distributed Management. *Adding a network and its abundance of resources to the system need not require specific allocation of names, addresses, or routing from centralized authorities.*

thorities. Contrary to popular belief, the Internet is not entirely a distributed network. While its management is somewhat decentralized, many aspects of its structure and governance are hierarchical in nature. Autonomous systems engage in peering relationships in a manner that promotes the set of "tiers" that characterize the organization of Internet service providers today. Both the addresses and the names used to identify resources are allocated by a collection of governance organizations, arranged hierarchically. Such an arrangement is contrary to the underlying relationships among organizations interested in using the Internet to communicate. In some sense we would like to provide a means by which the Internet can grow without requiring the consent of far-removed third parties.

Deployability. *Our system must be deployable within the current Internet infrastructure.* Any complex system of sufficiently large scale that cannot be deployed incrementally will never amass enough interest to overcome the economic hurdles to deployment. Our system must provide substantial benefit even if its rate of adoption is quite limited. So, we require that our system can coexist with existing Internet infrastructure. In particular, both clients and servers should be able to easily use both our system and the underlying Internet architecture.

We will demonstrate that one infrastructure can be used to provide all of these advantages. We will characterize ways in which systemic fragmentation occurs today, examine why existing architectures fail to avoid framgentation, and consider the design of existing systems created to mitigate aspects of the phenomenon. This analysis will provide a better understanding of what characteristics a general-purpose system for facilitating communication in a fragmented network will require, which will in turn allow us to argue in favor of particular design choices. We will provide a proof-of-concept implementation and show how the system can be used to promote the interests of locality while allowing access throughout the system. Finally, we will conclude by examining the impact and benefits of such a system.

Throughout our analysis, we onsider resources to be identified by names meaningful to humans (in this case, hostnames provided by DNS), rather then by their underlying network-layer addresses. This provides the advantage that the mapping of IP addresses to names can occur via local DNS.

Note that our system design achieves its seemingly conflicting goals of locality and universal access at the expense of universal naming. Indeed, one of the key features of the Internet today is that names used to identify resources are universal: they depend only upon the resource and are not defined by the name, physical location, or logical location of the entity requesting the resource. We argue that universal naming is not indispensable, and we believe that by relaxing this constraint we can achieve a considerably more flexible network.

4 Architecture

The fundamental principle underlying Blossom is that it is possible and beneficial to design a system that sacrifices globally apportioned names in favor of a distributed method of assigning names. We achieve this goal by allowing for the existence of multiple independent Internet fragments, possibly overlapping, each with its own set of names. By considering each fragment to be its own namespace, we avoid requiring that all names are globally unique. However, in order to achieve universal access, we must provide a means by which all resources can be named. To this end, we stipulate that that names of forwarders are globally unique, and we identify some target resource R as a combination of the name of a forwarder that can reach R concatentated with the name of R as seen by that forwarder. Unlike Internet hosts, Blossom forwarders *choose their own names* by generating a self-certifying identifier and using that as a global name. In this sense, each resource accessible via Blossom is associated with at least one globally unique name, but the resources themselves are not responsible for guaranteeing global uniqueness—instead, all that is required is uniqueness within the local fragment observed by some particular forwarder.

Blossom itself consits of a peer-to-peer overlay network of *forwarders*, each of which has access to resources within its own local fragment. The overlay network that connects all of the forwarders to each other consists of a *data plane* that carries tunnelled DNS requests and TCP sessions, as well as a *control plane* that carries routing information.

4.1 System Overview

Suppose that the forwarders have organized themselves into an overlay that can route TCP traffic. Each forwarder independently generates a self-certifying identifier, and forwarders throughout the system refer to other forwarders using these identifiers. As long as the size of the identifier is sufficiently large and the sources of randomness are sufficiently effective, the chance of a namespace collision among these identifiers within the system will be negligible.

Next, we show how Blossom enables an Internet host to access resources outside its local fragment. See Figure 4.1. Suppose that the source (labeled `foo.source.net`) wants to communicate with a host known to forwarder $F4$ as

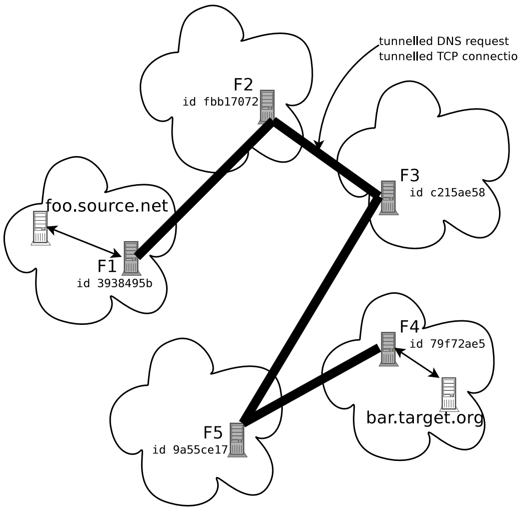

Figure 4: Accessing a Resource. *The source establishes a connection to* `bar.target.org.79f72ae5.blossom`*. DNS requests and TCP sessions are both tunneled through the infrastructure.*

`bar.target.org`. Suppose that the source knows how to talk to $F1$, and that the self-chosen ID of $F4$ is `79f72ae5`.[1] Then, the source will tell $F1$ to open a TCP session to `bar.target.org.79f72ae5.blossom` on its behalf. The control plane provides $F1$ with routing information indicating that $F2$ is the next hop en route to $F4$, so $F1$ knows how to forward packets through the overlay to $F4$. At this point, $F1$ forwards the request for `bar.target.org` through the overlay to $F4$, who uses DNS to resolves it to an IP address. At this point, $F1$ can tunnel the entire TCP session through the overlay to $F4$. Note that this involves segmenting the TCP session—the conversation between the source and $F1$ will have a different pair of source and destination addresses than the conversation between $F4$ and the target resource. This means that Blossom will not work with end-to-end address-based security systems such as IPSec.

Observe that the combined name `bar.target.org.79f72ae5.blossom` is globally unique, but the name was not apportioned by any authority of global scope. Also, there is no requirement that each resource is associated with exactly one forwarder; multiple forwarders may be able to reach the same resource, possibly using different names. See Figure 4.1. In this example, the source host may use any of `bar.target.org.79f72ae5.blossom`, `bar.target.org.c215ae58.blossom`, or `baz.other.com.9a55ce17.blossom` to refer to the target host. Another useful feature of this design is that if the network-layer address or name of a forwarder changes, its Blossom ID does not, thus promoting mobility.

4.2 Design Challenges

Designing the control plane presents an assortment of research challenges related to establishing the overlay and routing traffic. The primary objective for routing is to create a means by which each forwarder can learn how to route a packet toward the destination specified by its Blossom ID, and two fundamental methods are potential candidates for serving this task. The first method is *advertisement*, in which each forwarder announces its availability to the entire overlay network using a path-vector flooding protocol similar to BGP. The primary constraints to this method are that all peer nodes must be informed of all changes to availability, and each peer node must contain entries corresponding to all of the currently available peer nodes in the system. Since all hosts need to know about all forwarders and self-certifying identifiers cannot be aggregated in any meaningful way, scalability becomes a substantial concern.

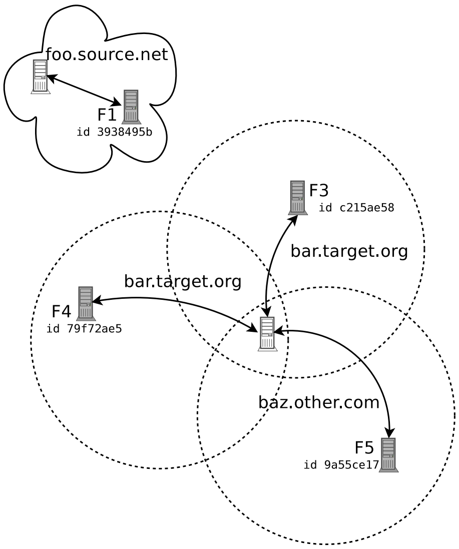

Figure 5: Multiple Names. *A Blossom DNS name uniquely identifiesa resource, but a resource need not have only one Blossom DNS name. In this example, the target host is known to $F3$ and $F4$ as* `bar.target.org` *and to $F5$ as* `baz.other.com`*.*

An on-demand alternative for propagating routing information is *query*, in which a forwarder who wants to find a route to another forwarder in the system issues a request that is propagated through the network. To reduce the cost associated with this operation, we may stipulate that each peer node maintain recently cached entries corresponding to the various forwarders about which it received information and that such entries. However, scalability may depend upon limiting the propagation of queries through the system, creating a tradeoff between scalability and universal access.

Another challenge lies in associating meaningful attributes with individual forwarders. For example, a user of the system may want a means of accessing a resource via a forwarder in the Netherlands, but it need not matter which forwarder specifically. Creating a means by which a requester can specify a set of attributes rather than a specific destination forwarder ID in identifying a resource can address this goal, but providing the infrastructure means adding complexity to the system.

[1] Four bytes of entropy is certainly insufficient for a self-certifying ID. We chose four bytes to create an illustrative example; actual IDs would be longer.

5 Related Work

A number of systems in the literature address problems related to network fragmentation and provide services similar to those of Blossom. Andersen et al. propose the use of Resilient Overlay Networks (RON) [2] to address several limitations of BGP. RON has three goals: (a) provide additional robustness in the event of localized network malfunction, specifically recover from malfunctions faster than BGP, (b) provide tighter integration with applications to allow them some control over the underlying routing, and (c) provide the ability to express more complex policies than those that can be expressed via BGP. RON provides an overlay infrastructure that participating nodes within the Internet can use to attain these additional benefits. Like Blossom, RON aims to overcome network obstructions. However, its purpose is essentially limited to finding alternate routes more effectively than BGP. Thus, it does not address our interest in locality or decentralized management.

Previous work addresses the problem of routing data based upon content, often employing overlays to organize content logically and providing suitable naming infrastructures to enable a means of accessing arbitrary resources (e.g., [5, 12]). TRIAD [6] characterizes the Internet as a set of regions with local addressing, arranged such that some peers have access to multiple regions. The authors justify this characterization by noting the preponderance of NAT boxes. Peers with access to multiple address spaces use a protocol called WRAP to relay content between different regions of the network in a stateless manner. A key difference between Blossom and TRIAD is that TRIAD uses globally unique hierarchical, DNS-style names to identify networks and relies on a modified BGP to route according to these names.

The Internet Indirection Infrastructure (I3) [20] provides a "rendezvous-based communication abstraction" in which providers of services advertise to a particular location in the network, and those peers requesting services communicate with that location rather than with the provider directly. Indeed, services like anycast, multicast, and mobility all require some measure of "indirection". I3 offers a standard substrate upon which all of these can be built and provides mechanisms for achieving composition of services, scalable multicast, etc., which have tangible benefit in the real world. The authors present how the functionality of various existing systems for providing these services can be achieved with I3. Finally, I3 provides useful delegation primitives that serve as inspiration for DOA, which we discuss next.

Walfish et al. [22] describe how to create a Delegation-Oriented Architecture that allows middleboxes to perform their functions without violating two fundamental principles of Internet design: that every Internet entity can be reached via the use of a unique network identifier, and that network elements should not violate the principles of layering. DOA allows nodes to express, using an endpoint identifier (EID), how they can be reached by others. Each EID resolves to another EID, a list of EIDs, or an IP address. DOA uses a DHT to perform this resolution. DOA provides the benefit that the functionality of a middlebox, such as a firewall can be orthogonal to topology: for example, a node could choose to use a firewall by using an EID that would address packets to the firewall, which could process them and pass them along to the ultimate destination.

There are a number of differences between Blossom and DOA. First, DOA requires modification to the IP stacks of both clients and servers. Blossom aims to provide universal access to remote resources without the need to modify protocol stacks. Second, DOA allows for an architecture of Network Extension Boxes (NEB), which would replace NATs. This architecture relies upon the existence of a well-known Internet "core." This raises questions of whether nodes in the core have to be specially configured, whether they know that they are in the core, and whether the Internet needs to have an inherent hierarchy in order for NEBs to function. In Blossom, we explicitly avoid making any core-based assumptions. Moreover, the DHT functionality central to DOA also seems to rely upon the existence of a well-known core.

The FARA proposal [7] specifies a general framework for decoupling identity from network location. FARA aims to provide associations between peer nodes without requiring that all entities share a common, global namespace; in this sense, its goals are similar to ours. FARA makes use of "forwarding directives" to establish rendezvous points through the infrastructure between a source and a destination; a shim protocol between IP and the transport layer used to support this functionality is reminiscent of TRIAD. FARA is structured so that discovery may be handled by higher layer services; the location of an entity is defined by the forwarding directive, which may be obtained via the rendezvous mechanism or a FARA directory service, for example. By not requiring all entities to share a common, global namespace, one can argue that FARA takes a step toward our goals of distributed management and locality. However, the authors do not seem to envision this possibility in their test implementation, M-FARA, which avoids the "complexity" associated with dealing with an unstructured Internet by relying upon a well-known Internet core.

Snoeren and Raghavan [18] argue that routing policy should be enforced on the forwarding plane rather than on the control plane, as it is done today with BGP4. The authors propose a new routing architecture, Platypus, which uses loose source routing (LSR) to allow fine-grained, policy-aware route selection by the sender. In Platypus, autonomous sytems advertise all available routes, irrespective of policy, along with "network capability" metadata. Loose source routing information would be included in each packet, allowing end users to take advantage of Platypus directly. Also, routers within the network could use the metadata to improve route selection. Most notably, this work presents a major paradigm shift: instead of requiring that local policies dictate all routing, propagate advertisements deeper into the network so that hosts and networks can make more informed decisions. Let those who need to use the route make the routing decisions, and rely upon filtering techniques to guarantee that routes incompatible with policy are not used.

A key difference between Blossom and Platypus is that Platypus depends upon ISPs being on board, willing to advertise routes that they themselves may prefer not to use. In Blossom, we assume that if an ISP does not want to forward traffic in a particular direction, it has no reason to do so and no reason to advertise such a possibility either.

Finally, anonmyizing networks such as Tor [8] and ANON [14] disguise the identity of clients and servers by identifying services as part of the infrastructure and making clients appear to servers as part of the infrastructure. The techniques used for forwarding packets through the infrastructure, as well as the corresponding arguments about efficacy, may prove useful to the design of Blossom.

6 Remarks

Our work offers three main contributions. First, we provide a set of design goals that afford a new way of considering Internet resources, specifically the idea of sacrificing globally apportioned names to allow locality in naming, distributed managment, and universal access to resources. Second, by specifying globally unique identifiers that are locally generated, we provide a means by which Internet resources can be identified in the absence of a hierarchical naming scheme. Third, we provide a means by which it is possible to communicate across multiple, independently managed Internetworks, despite each having its own independent naming and routing infrastructure.

Blossom introduces a number of issues for future study. At its core, our system is designed to heal fragmentation. This means that Internet peers may gain access to some resources to which they did not have access previously. There are various implications of routing around network-layer access restrictions, and we will have the task of showing that our system can coexist with reasonable policy frameworks. Another question is whether Blossom can be used to resolve namespace arbitrage; proper use of this system may lead to a reduced number of lawsuits related to trademark contention resulting from allocation of resource names. Finally, two important tasks for affirming the usefulness of Blossom would be (a) to determine how networks would have to arrange themselves to best take advantage of fragmentation, and (b) to show that it is possible to derive substantial benefits with a limited number of peer nodes.

There are both costs and benefits to considering a less rigid, more organic structure to the allocation of resource names. We believe that this work presents a new and useful way of considering the organization of Internet services and Internet connectivity.

References

[1] L. Abba, M. Buzzi, D. Pobric, and M. Ianigro. Introducing transparent web caching in a local area network. In *Intl. Computer Measurement Group Conference*, 2000.

[2] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient Overlay Networks. In *SOSP*, October 2001.

[3] S. Bhattacharjee, K. L. Calvert, and E. W. Zegura. Active Networking and End-to-End Argument. In *ICNP*, 1997.

[4] M. Bright. BT Puts Block on Child Porn Sites. *The Guardian, 6 June 2004*, 2004.

[5] R. Chand and P. Felber. A Scalable Protocol for Content-Based Routing in Overlay Networks. In *Intl Symposium on Network Computing and Applications*, 2003.

[6] D. R. Cheriton and M. Gritter. TRIAD: A New Next-Generation Internet Architecture. http://www-dsg.stanford.edu/triad/, 2000.

[7] D. Clark, R. Braden, A. Falk, and V. Pingali. FARA: Reorganizing the Addressing Architecture. *ACM SIGCOMM Computer Communication Review*, pages 313–321, 2003.

[8] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The Second-Generation Onion Router. In *USENIX Security Symposium*, 2004.

[9] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin. Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing. In *NDSS*, 2003.

[10] T. G. Griffin, F. B. Shepherd, and G. Wilfong. Policy Disputes in Path Vector Protocols. In *ICNP 1999*, 1999.

[11] T. G. Griffin and G. Wilfong. On the Correctness of IBGP Configuration. In *SIGCOMM*, 2002.

[12] M. Gritter and D. R. Cheriton. An Architecture for Content Routing Support in the Internet. In *USITS*, 2001.

[13] S. Kent, C. Lynn, and K. Seo. Secure Border Gateway Protocol (Secure-BGP). *IEEE Journal on Selected Areas in Communications*, 18(4):582–592, 2000.

[14] H. T. Kung, C.-M. Cheng, K.-S. Tan, and S. Bradner. Design and Analysis of an IP-Layer Anonymizing Infrastructure. In *DARPA Information Survivability Conference and Exposition (DISCEX 3)*, 2003.

[15] R. Mahajan, D. Wetherall, and T. Anderson. Understanding BGP Misconfiguration. In *SIGCOMM*, pages 3–16. ACM, 2002.

[16] Y. Rekhter and T. Li. A Border Gateway Protocol 4 (BGP 4). Internet Engineering Task Force: RFC 1771, 1995.

[17] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-End Arguments in System Design. *ACM TOCS*, 2(4):277–288, 1984.

[18] A. C. Snoeren and B. Raghavan. Decoupling Policy from Mechanism in Internet Routing. In *HotOS*, 2003.

[19] J. Stewart. Bgp4: Interdomain routing in the internet. Addison-Wesley, 1998.

[20] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet Indirection Infrastructure. In *SIGCOMM*, 2002.

[21] K. Varadhan, R. Govindan, and D. Estrin. Persistent Route Oscillations in Inter-Domain Routing. *Computer Networks*, 32(1):1–16, 2000.

[22] M. Walfish, J. Stribling, M. Krohn, H. Balakrishnan, R. Morris, and S. Shenker. Middleboxes no longer considered harmful. In *OSDI*, Dec. 2004.