A Retrospective View of Network Address Translation

Lixia Zhang, University of California, Los Angeles

Abstract

Today, network address translators, or NATs, are everywhere. Their ubiquitous adoption was not promoted by design or planning but by the continued growth of the Internet, which places an ever-increasing demand not only on IP address space but also on other functional requirements that network address translation is perceived to facilitate. This article presents a personal perspective on the history of NATs, their pros and cons in a retrospective light, and the lessons we can learn from the NAT experience.

A network address translator (NAT) commonly refers to a box that interconnects a local network to the public Internet, where the local network runs on a block of private IPv4 addresses as specified in RFC 1918 [1]. In the original design of the Internet architecture, each IP address was defined to be globally unique and globally reachable. In contrast, a private IPv4 address is meaningful only within the scope of the local network behind a NAT and, as such, the same private address block can be reused in multiple local networks, as long as those networks do not directly talk to each other. Instead, they communicate with each other and with the rest of Internet through NAT boxes.

Like most unexpected successes, the ubiquitous adoption of NATs was not foreseen when the idea first emerged more than 15 years ago [2, 3]. Had anyone foreseen where NAT would be today, it is possible that NAT deployment might have followed a different path, one that was better planned and standardized. The set of Internet protocols that were developed over the past 15 years also might have evolved differently by taking into account the existence of NATs, and we might have seen less overall complexity in the Internet compared to what we have today.

Although the clock cannot be turned back, I believe it is a worthwhile exercise to revisit the history of network address translation to learn some useful lessons. It also can be worthwhile to assess, or reassess, the pros and cons of NATs, as well as to take a look at where we are today in our understanding of NATs and how best to proceed in the future.

It is worth pointing out that in recent years many efforts were devoted to the development and deployment of NAT traversal solutions, such as simple traversal of UDP through NAT (STUN) [4], traversal using relay NAT (TURN) [5], and Teredo [6], to name a few. These solutions remove obstacles introduced by NATs to enable an increasing number of new application deployments. However, as the title suggested, this article focuses on examining the lessons that we can learn from the NAT deployment experience; a comprehensive survey of NAT traversal solutions must be reserved for a separate article.

I also emphasize that this writing represents a personal view, and my recall of history is likely to be incomplete and to contain errors. My personal view on this subject has also changed over time, and it may continue to evolve, as we are all in a continuing process of understanding the fascinating and dynamically changing Internet.

How a NAT Works

As mentioned previously, IP addresses originally were designed to be globally unique and globally reachable. This property of the IP address is a fundamental building block in supporting the end-to-end architecture of the Internet. Until recently, almost all of the Internet protocol designs, especially those below the application layer, were based on the aforementioned IP address model. However, the explosive growth of the Internet during the 1990s not only signaled the danger of IP address space exhaustion, but also created an instant demand on IP addresses: suddenly, connecting large numbers of user networks and home computers demanded IP addresses instantly and in large quantities. Such demand could not possibly be met by going through the regular IP address allocation process. Network address translation came into play to meet this instant high demand, and NAT products were quickly developed to meet the market demand.

However, because NATs were not standardized before their wide deployment, a number of different NAT products exist today, each with somewhat different functionality and different technical details. Because this article is about the history of NAT deployment — and not an examination of how to traverse various different NAT boxes — I briefly describe a popular NAT implementation as an illustrative example. Interested readers can visit Wikipedia to find out more about existing types of NAT products.

A NAT box **N** has a public IP address for its interface connecting to the global Internet and a private address facing the internal network. **N** serves as the default router for all of the destinations that are outside the local NAT address block. When an internal host **H** sends an IP packet P to a

0890-8044/08/$25.00 © 2008 IEEE

public IP destination address **D** located in the global Internet, the packet is routed to **N**. **N** translates the private source IP address in P's header to **N**'s public IP address and adds an entry to its internal table that keeps track of the mapping between the internal host and the outgoing packet. This entry represents a piece of *state*, which enables subsequent packet exchanges between H and D. For example, when **D** sends a packet **P**' in response to P, P' arrives at **N**, and **N** can find the corresponding entry from its mapping table and replace the destination IP address — which is its own public IP address — with the real destination address **H**, so that P' will be delivered to **H**. The mapping entry times out after a certain period of idleness that is typically set to a vendor-specific value. In the process of changing the IP address carried in the IP header of each passing packet, a NAT box also must recalculate the IP header checksum, as well as the checksum of the transport protocol if it is calculated based on the IP address, as is the case for Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) checksums.

From this brief description, it is easy to see the major benefit of a NAT: one can connect a large number of hosts to the global Internet by using a single public IP address. A number of other benefits of NATs also became clear over time, which I will discuss in more detail later.

At the same time, a number of drawbacks to NATs also can be identified immediately. First and foremost, the NAT changed the end-to-end communication model of the Internet architecture in a fundamental way: instead of allowing any host to talk *directly* to any other host on the Internet, the hosts behind a NAT must go through the NAT to reach others, and all communications through a NAT box must be initiated by an internal host to set up the mapping entries on the NAT. In addition, because ongoing data exchange depends on the mapping entry kept at the NAT box, the box represents a single point of failure: if the NAT box crashes, it could lose all the existing state, and the data exchange between all of the internal and external hosts must be restarted. This is in contrast to the original goal of IP of delivering packets to their destinations, as long as *any* physical connectivity exists between the source and destination hosts. Furthermore, because a NAT alters the IP addresses carried in a packet, all protocols that are dependent on IP addresses are affected. In certain cases, such as TCP checksum, which includes IP addresses in the calculation, the NAT box can hide the address change by recalculating the TCP checksum when forwarding a packet. For some of the other protocols that make direct use of IP addresses, such as IPSec [7], the protocols can no longer operate on the end-to-end basis as originally designed; for some application protocols, for example, File Transfer Protocol (FTP) [8], that embed IP addresses in the application data, application-level gateways are required to handle the IP address rewrite. As discussed later, NAT also introduced other drawbacks that surfaced only recently.

A Recall of the History of NATs

I started my Ph.D. studies in the networking area at the Massachusetts Institute of Technology at the same time as RFC 791 [9], the Internet Protocol Specification, was published in September 1981. Thus I was fortunate to witness the fascinating unfolding of this new system called the Internet. During the next ten years, the Internet grew rapidly. RFC 1287 [2], *Towards the Future Internet Architecture*, was published in 1991 and was probably the first RFC that raised a concern about IP address space exhaustion in the foreseeable future.

RFC 1287 also discussed three possible directions to extend IP address space. The first one pointed to a direction similar to current NATs:

> *Replace the 32-bit field with a field of the same size but with a different meaning. Instead of being globally unique, it would be unique only within some smaller region. Gateways on the boundary would rewrite the address as the packet crossed the boundary.*

RFC 1335 [3], published shortly after RFC 1287, provided a more elaborate description of the use of internal IP addresses (i.e., private IP addresses) as a solution to IP address exhaustion. The first article describing the NAT idea, "*Extending the IP Internet through Address Reuse*" [10], appeared in the January 1993 issue of *ACM Computer Communication Review* and was published a year later as RFC 1631 [11]. Although these RFCs can be considered forerunners in the development of NAT, as explained later, for various reasons the IETF did not take action to standardize NAT.

The invention of the Web further accelerated Internet growth in the early 1990s. The explosive growth underlined the urgency to take action toward solving both the routing scalability and the address shortage problems. The IETF took several follow-up steps, which eventually led to the launch of the IPng development effort. I believe that the expectation at the time was to develop a new IP within a few years, followed by a quick deployment. However, the actual deployment during the next ten years took a rather unexpected path.

The Planned Solution

As pointed out in RFC 1287, the continued growth of the Internet exposed strains on the original design of the Internet architecture, the two most urgent of which were routing system scalability and the exhaustion of IP address space. Because long-term solutions require a long lead time to develop and deploy, efforts began to develop both a short term and a long-term solution to those problems.

Classless inter-domain routing, or CIDR, was proposed as a short term solution. CIDR removed the class boundaries embedded in the IP address structure, thus enabling more efficient address allocation, which helped extend the lifetime of IP address space. CIDR also facilitated routing aggregation, which slowed down the growth of the routing table size. However, as stated in RFC 1481 [12], *IAB Recommendation for an Intermediate Strategy to Address the Issue of Scaling*: "This strategy (CIDR) presumes that a suitable long-term solution is being addressed within the Internet technical community." Indeed, a number of new IETF working groups started in late 1992 and aimed at developing a new IP as a long-term solution; the Internet Engineering Steering Group (IESG) set up a new IPng area in 1993 to coordinate the efforts, and the IPng Working Group (later renamed to IPv6) was established in the fall of 1994 to develop a new version of IP [13].

CIDR was rolled out quickly, which effectively slowed the growth of the global Internet routing table. Because it is a quick fix, CIDR did not address emerging issues in routing scalability, in particular the issue of site multihoming. A multihomed site should be reachable through any of its multiple provider networks. In the existing routing architecture, this requirement translates to having the prefix, or prefixes, of the site listed in the global routing table, thereby rendering provider-based prefix aggregation ineffective. Interested readers are referred to [14] for a more detailed description on multihoming and its impact on routing scalability.

The new IP development effort, on the other hand, took much longer than anyone expected when the effort first began. The IPv6 working group finally completed all of the protocol development effort in 2007, 13 years after its establishment. The IPv6 deployment also is slow in coming. Until recently, there were relatively few IPv6 trial deployments; there is no known commercial user site that uses IPv6 as the primary protocol for its Internet connectivity.

If one day someone writes an Internet protocol development history, it would be very interesting to look back and understand the major reasons for the slow development and adoption of IPv6. But even without doing any research, one could say with confidence that NATs played a major role in meeting the IP address requirement that arose out of the Internet growth and at least deferred the demand for a new IP to provide the much needed address space to enable the continued growth of the Internet.

The Unplanned Reality

Although largely unexpected, NATs have played a major role in facilitating the explosive growth of Internet access. Nowadays, it is common to see multiple computers, or even multiple LANs, in a single home. It would be unthinkable for every home to obtain an IP address block, however small it may be, from its network service provider. Instead, a common implementation for home networking is to install a NAT box that connects one home network or multiple home networks to a local provider. Similarly, most enterprise networks deploy NATs as well. It also is well known that countries with large populations, such as India and China, have most of their hosts behind NAT boxes; the same is true for countries that connected to the Internet only recently. Without NATs, the IPv4 address space would have been exhausted a long time ago.

For reasons discussed later, the IETF did not standardize NAT implementation or operations. However, despite the lack of standards, NATs were implemented by multiple vendors, and the deployment spread like wildfire. This is because NATs have several attractions, as we describe next.

Why NATs Succeeded

NATs started as a short term solution while waiting for a new IP to be developed as the long-term solution. The first recognized NAT advantages were stated in RFC 1918 [1]:

> *With the described scheme many large enterprises will need only a relatively small block of addresses from the globally unique IP address space. The Internet at large benefits through conservation of globally unique address space, which will effectively lengthen the lifetime of the IP address space. The enterprises benefit from the increased flexibility provided by a relatively large private address space.*

The last point deserves special emphasis. Indeed, anyone can use a large block of private IP addresses — up to 16 million without asking for permission — and then connect to the rest of the Internet by using only a single public IP address. A big block of private IP addresses provides the much needed room for future growth. On the other hand, for most if not all user sites, it is often difficult to obtain an IP address block that is beyond their immediate requirements.

Today, NAT is believed to offer advantages well beyond the above. Essentially, the mapping table of a NAT provides one level of indirection between hosts behind the NAT and the global Internet. As the popular saying goes, "Any problem in computer science can be solved with another layer of indirection." This one level of indirection means that one never need worry about renumbering the internal network when changing providers, other than renumbering the public IP address of the NAT box.

Similarly, a NAT box also makes multihoming easy. One NAT box can be connected to multiple providers and use one IP address from each provider. Not only does the NAT box shelter the connectivity to multiple ISPs from all the internal hosts, but it also does not require any of its providers to "punch a hole" in the routing announcement (i.e., make an ISP de-aggregate its address block). Such a hole punch would be required if the multihomed site takes an IP address block from one of its providers and asks the other providers to announce the prefix.

Furthermore, this one level of indirection also is perceived as one level of protection because external hosts cannot directly initiate communication with hosts behind a NAT, nor can they easily figure out the internal topology.

Besides all of the above, two additional factors also contributed greatly to the quick adoption of NATs. First, NATs can be unilaterally deployed by any end site without any coordination by anybody else. Second, the major gains from deploying a NAT were realized on day one, whereas its potential drawbacks were revealed only slowly and recently.

The Other Side of the NAT

A NAT disallows the hosts behind it from being reachable by an external host and hence disables it from being a server. However, in the early days of NAT deployment, many people believed that they would have no need to run servers behind a NAT. Thus, this architectural constraint was viewed as a security feature and believed to have little impact on users or network usage. As an example, the following four justifications for the use of private addresses are quoted directly from RFC 1335 [3].

- In most networks, the majority of the traffic is confined to its local area networks. This is due to the nature of networking applications and the bandwidth constraints on inter-network links.
- The number of machines that act as Internet servers, that is, run programs waiting to be called by machines in other networks, is often limited and certainly much smaller than the total number of machines.
- There are an increasingly large number of personal machines entering the Internet. The use of these machines is primarily limited to their local environment. They also can be used as clients such as ftp and telnet to access other machines.
- For security reasons, many large organizations, such as banks, government departments, military institutions, and some companies, allow only a very limited number of their machines to have access to the global Internet. The majority of their machines are purely for internal use.

As time goes on, however, the above reasoning has largely been proven wrong.

First, network bandwidth is no longer a fundamental constraint today. On the other hand, voice over IP (VoIP) has become a popular application over the past few years. VoIP changed the communication paradigm from client-server to a peer-to-peer model, meaning that any host may call any other host. Given the large number of Internet hosts that are behind NAT, several NAT traversal solutions have been developed to support VoIP. A number of other recent peer-to-peer applications, such as BitTorrent, also have become popular recently, and each must develop its own NAT traversal solutions.

In addition to the change of application patterns, a few other problems also arise due to the use of non-unique, private IP addresses with NATs. For instance, a number of business acquisitions and mergers have run into situations where two networks behind NATs were required to be interconnected, but unfortunately, they were running on the same private address block, resulting in address conflicts. Yet another problem emerged more recently. The largest allocated private address block is 10.0.0.0/8, commonly referred to as net-10. The business growth of some provider and enterprise networks is leading to, or already has resulted in, the net-10 address exhaustion. An open question facing these networks is what to do next. One provider network migrated to IPv6; a number of others simply decided on their own to use another unallocated IP address block [15].

It is also a common misperception that a NAT box makes an effective firewall. This may be due partly to the fact that in places where NAT is deployed, the firewall function often is implemented in the NAT box. A NAT box alone, however, does not make an effective firewall, as evidenced by the fact that numerous home computers behind NAT boxes have been compromised and have been used as launch pads for spam or distributed denial of service (DDoS) attacks. Firewalls establish control policies on both incoming and outgoing packets to minimize the chances of internal computers being compromised or abused. Making a firewall serve as a NAT box does not make it more effective in fencing off malicious attacks; good control polices do.

Why the Opportunity of Standardizing NAT Was Missed

During the decade following the deployment of NATs, a big debate arose in the IETF community regarding whether NAT should, or should not, be deployed. Due to its use of private addresses, NAT moved away from the basic IP model of providing end-to-end reachability between hosts, thus representing a fundamental departure from the original Internet architecture. This debate went on for years. As late as 2000, messages posted to the IETF mailing list by individual members still argued that NAT was architecturally unsound and that the IETF should in no way endorse its use or development. Such a position was shared by many people during that time.

These days most people would accept the position that the IETF should have standardized NAT early on. How did we miss the opportunity? A simple answer could be that the crystal ball was cloudy. I believe that a little digging would reveal a better understanding of the factors that clouded our eyes at the time. As I see it from my personal viewpoint, the following factors played a major role.

First, the feasibility of designing and deploying a brand new IP was misjudged, as were the time and effort required for such an undertaking. Those who were opposed to standardizing NAT had hoped to develop a new IP in time to meet the needs of a growing Internet. Unfortunately, the calculation was way off. While the development of a new IP was taking its time, Internet growth did not wait. Network address translation is simply an inevitable consequence that was not clearly recognized at the time.

Second, the community faced a difficult question regarding how strictly one should stick to architectural principles, and what can be acceptable engineering trade-offs. Architectural principles are guidelines for problem solving; they help guide us toward developing better overall solutions. However, when the direct end-to-end reachability model was interpreted as an absolute rule, it ruled out network address translation as a feasible means to meet the instant high demand for IP addresses at the time. Furthermore, sticking to the architectural model in an absolute way also contributed to the one-sided view of the drawbacks of NATs, hence the lack of a full appreciation of the advantages of NATs as we discussed earlier, let alone any effort to develop a NAT-traversal solution that can minimize the impact of NATs on end-to-end reachability.

Yet another factor was that given that network address translation could be deployed unilaterally by a single party alone, there was not an apparent need for standardization. This seemingly valid reasoning missed an important fact: a NAT box does not stand alone; rather it interacts both directly with surrounding IP devices, as well as indirectly with remote devices through IP packet handling. The need for standardizing network address translation behavior has since been well recognized, and a great effort has been devoted to developing NAT standards in recent years [16].

Unfortunately the early misjudgment on NAT already has cost us dearly. While the big debate went on through the late 1990s and early part of the first decade of this century, NAT deployment was widely rolled out, and the absence of a standard led to a number of different behaviors among various NAT products. A number of new Internet protocols also were developed or finalized during the same time period, such as IPSec, Session Announcement Protocol (SAP), and Session Initiation Protocol (SIP), to name a few. Their designs were based on the original model of IP architecture, wherein IP addresses are assumed to be globally unique and globally reachable. When those protocols became ready for deployment, they faced a world that was mismatched with their design. Not only were they required to solve the NAT traversal problem, but the solutions also were required to deal with a wide variety of NAT box behaviors.

Although NAT is accepted as a reality today, the lessons to learn from the past are yet to be clarified. One example is the recent debate over Class-E address block usage [17]. Class-E refers to the IP address block 240.0.0.0/4 that has been on reserve until now. As such, many existing router and host implementations block the use of Class-E addresses. Putting aside the issue of required router and host changes to enable Class-E usage, the fundamental debate has been about whether this Class-E address block should go into the public address allocation pool or into the collection of private address allocations. The latter would give those networks that face net-10 exhaustion a much bigger private address block to use. However, this gain is also one of the main arguments against it, as the size limitation of private addresses is considered a pressure to push those networks facing the limitation to migrate to IPv6, instead of staying with NAT. Such a desire sounds familiar; similar arguments were used against NAT standardization in the past. However if the past is any indication of the future, we know that pressures do not dictate new protocol deployment; rather, economical feasibility does. This statement does not imply that migrating to IPv6 brings no economical feasibility. On the contrary, it does, especially in the long run. New efforts are being organized both in protocol and tools development to smooth and ease the transition from IPv4 to IPv6 and in case studies and documentation to show clearly the short- and long-term gains from deploying IPv6.

Looking Back and Looking Forward

The IPv4 address space exhaustion predicted long ago is finally upon us today, yet the IPv6 deployment is barely visible on the horizon. What can and should be done now to enable the Internet to grow along the best path forward? I hope this review of NAT history helps shed some light on the answer.

First, we should recognize not only the fact that IPv4 network address translation is widely deployed today, but also recognize its perceived benefits to end users as we discussed in a previous section. We should have a full appraisal of the pros and cons of NAT boxes; the discussion in this article merely serves as a starting point.

Second, it is likely that some forms of network address translation boxes will be with us forever. Hopefully, a full appraisal of the pros and cons of network address translation would help correct the view that all network address translation approaches are a "bad thing" and must be avoided at all costs. Several years ago, an IPv4 to IPv6 transition scheme called Network Address Translation-Protocol Translation (NAT-PT; see [18]) was developed but later classified to historical status,[1] mainly due to the concerns that:

- NAT-PT works in much the same way as an IPv4 NAT box.
- NAT-PT does not handle all the transition cases.

However, in view of IPv4 NAT history, it seems worthwhile to revisit that decision. IPv4, together with IPv4 NAT, will be with us for years to come. NAT-PT seems to offer a unique value in bridging IPv4-only hosts and applications with IPv6-enabled hosts and networks. There also have been discussions of the desire to perform address translations between IPv6 networks as a means to achieve several goals, including insulating one's internal network from the outside. This question of "Whither IPv6 NAT?" deserves further attention. Instead of repeating the mistakes with IPv4 NAT, the Internet would be better off with well-engineered standards and operational guidelines for traversing IPv4 and IPv6 NATs that aim at maximizing interoperability.

Furthermore, accepting the existence of network address translation in today's architecture does not mean we simply take the existing NAT traversal solutions as given. Instead, we should fully explore the NAT traversal design space to steer the solution development toward restoring the end-to-end reachability model in the original Internet architecture. A new effort in this direction is the NAT traversal through tunneling (NATTT) project [19]. Contrary to most existing NAT traversal solutions that are server-based or protocol-specific, NATTT aims to restore end-to-end reachability among Internet hosts in the presence of NATs, by providing generic, incrementally deployable NAT-traversal support for all applications and protocols.

Last, but not least, I believe it is important to understand that successful network architectures can and *should* change over time. All new systems start small. Once successful, they grow larger, often by multiple orders of magnitude as is the case of the Internet. Such growth brings the system to an entirely new environment that the original designers may not have envisioned, together with a new set of requirements that must be met, hence the necessity for architectural adjustments.

To properly adjust a successful architecture, we must have a full understanding of the key building blocks of the architecture, as well as the potential impact of any changes to them. I believe the IP address is this kind of key building block that touches, directly or indirectly, all other major components in the Internet architecture. The impact of IPv4 NAT, which changed IP address semantics, provides ample evidence. During IPv6 development, much of the effort also involved a change in IP address semantics, such as the introduction of new concepts like that of the site-local address. The site-local address was later abolished and partially replaced by unique local IPv6 unicast addresses (ULA) [20], another new type of IP address. The debate over the exact meaning of ULA is still going on.

The original IP design clearly defined an IP address as being globally unique and globally reachable and as identifying an attachment point to the Internet. As the Internet continues to grow and evolve, recent years have witnessed an almost universal deployment of middleboxes of various types. NATs and firewalls are dominant among deployed middleboxes, though we also are seeing increasing numbers of SIP proxies and other proxies to enable peer-to-peer-based applications. At the same time, proposals to change the original IP address definition, or even redefine it entirely, continue to arise. What should be the definition, or definitions, of an IP address today, especially in the face of various middleboxes? I believe an overall examination of the role of the IP address in today's changing architecture deserves special attention at this critical time in the growth of the Internet.

Acknowledgments

I sincerely thank Mirjam Kuhne and Wendy Rickard for their help with an earlier version of this article that was posted in the online IETF Journal of October 2007. I also thank the coeditors and reviewers of this special issue for their invaluable comments.

References

[1] Y. Rekhter *et al.*, "Address Allocation for Private Internets," RFC 1918, 1996.
[2] D. Clark *et al.*, "Towards the Future Internet Architecture," RFC 1287, 1991.
[3] Z. Wang and J. Crowcroft, "A Two-Tier Address Structure for the Internet: A Solution to the Problem of Address Space Exhaustion," RFC 1335, 1992.
[4] J. Rosenberg *et al.*, "STUN: Simple Traversal of User Datagram Protocol (UDP) through Network Address Translators (NATs)," RFC 3489, 2003.
[5] J. Rosenberg, R. Mahy, and P. Matthews, "Traversal Using Relays around NAT (TURN)," draft-ietf-behave-turn-08, 2008.
[6] C. Huitema, "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)," RFC 4380, 2006.
[7] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol, RFC 2401, 1998.
[8] J. Postel and J. Reynolds, File Transfer Protocol (FTP), RFC 959, 1985.
[9] J. Postel, Internet Protocol Specification, RFC 791, 1981.
[10] P. Tsuchiya and T. Eng, "Extending the IP Internet through Address Reuse," *ACM SIGCOMM Computer Commun. Review*, Sept. 1993.
[11] K. Egevang and P. Francis, "The IP Network Address Translator (NAT)," RFC 1631, 1994.
[12] C. Huitema, "IAB Recommendation for an Intermediate Strategy to Address the Issue of Scaling," RFC 1481, 1993.
[13] R. M. Hinden, "IP Next Generation Overview," http://playground.sun.com/ipv6/INET-IPng-Paper.html, 1995.
[14] L. Zhang, "An Overview of Multihoming and Open Issues in GSE," *IETF J.*, Sept. 2006.
[15] L. Vegoda, "Used but Unallocated: Potentially Awkward /8 Assignments," *Internet Protocol J.*, Sept. 2007.
[16] http://www.ietf.org/html.charters/behave-charter.html; IETF BEHAVE Working Group develops requirements documents and best current practices to enable NATs to function in a deterministic way, as well as advises on how to develop applications that discover and reliably function in environments with the presence of NATs.
[17] http://www.ietf.org/mail-archive/web/int-area/current/msg01299.html; see the message dated 12/5/07 with subject line "240/4" and all the follow-up.
[18] G. Tsirtsis and P. Srisuresh, "Network Address Translation-Protocol Translation (NAT-PT)," RFC 2766, 2000.
[19] E. Osterweil *et al.*, "NAT Traversal through Tunneling (NATTT)," http://www.cs.arizona.edu/~bzhang/nat/
[20] R. M. Hinden and B. Haberman, "Unique Local IPv6 Unicast Addresses," RFC 4193, 2005.

Biography

LIXIA ZHANG (lixia@cs.ucla.edu) received her Ph.D. in computer science from the Massachusetts Institute of Technology. She was a member of research staff at the Xerox Palo Alto Research Center before joining the faculty of the UCLA Computer Science Department in 1995. In the past she served as vice chair of ACM SIGCOMM and co-chair of the IEEE ComSoC Internet Technical Committee. She is currently serving on the Internet Architecture Board.

[1] *Historical status means that a protocol is considered obsolete and is thus removed from the Internet standard protocol set.*