Network Working Group
Internet-Draft
Intended status: Best Current Practice
Expires: August 14, 2008

J. Rosenberg
Cisco

February 11, 2008

UDP and TCP as the New Waist of the Internet Hourglass
draft-rosenberg-internet-waist-hourglass-00

Status of this Memo

Copyright Notice

Abstract

One of the fundamental design principles of the Internet is that IP represents a common intermediate protocol layer, linking together a variety of link layer technologies underneath with a large number of applications on top. When drawn graphically, this can be show as an hourglass with IP in the middle. The preponderence of NATs and firewalls in the Internet has changed this reality, such that UDP and TCP are now the waist of the hourglass. This document discusses this

change and describes its implications for protocol and application design.

Table of Contents

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Introduction

One of the fundamental design principles of the Internet is that IP represents a common intermediate protocol layer. This intermediate layer is designed to support a variety of link layers - ethernet, fiber, ATM, and so on. Furthermore, new link layer technologies can be added in the future, without affecting IP itself.

In addition, IP can support a broad set of applications on top of it, ranging from email to instant messaging to voice over IP. All of those applications do not require changes to IP itself. As with link layers, new applications can be added at any time.

This design principle is often described as, "Everything over IP, and IP over everything". Graphically, it is often shown as an hourglass, with IP in the middle:

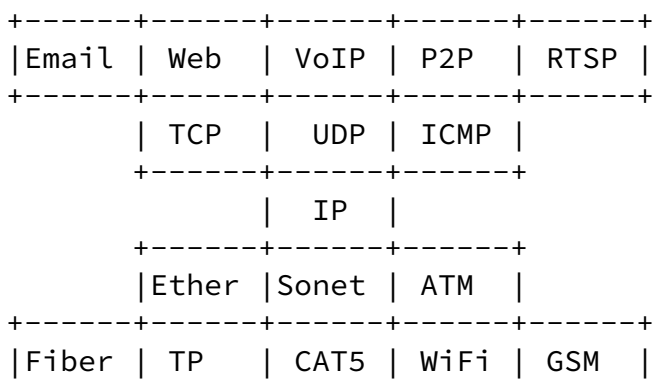

Figure 1: The Original IP Hourglass

This simple design is at the core of the success of the Internet, and is arguably the foundational principle on which it exists.

Unfortunately, there is a new reality in the Internet. Like it or not, this model is no longer true.

The preponderence of NATs and firewalls in the Internet has created a new model. In this new model, the waist of the hourglass is now UDP

and TCP, with ICMP being squeezed out. The new model looks like this:

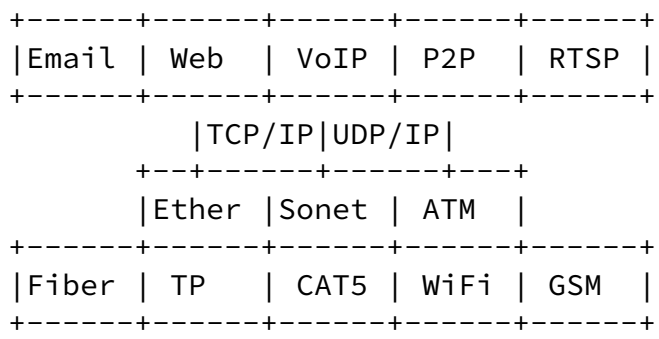

Figure 2: The Original IP Hourglass

This draft explains why this has happened and why it is now impossible to define new protocols natively on top of IP, discusses the implication for protocol design, and then considers the implications for IPv6.

3. What Caused the New Model?

Originally, NAT was designed to operate strictly at the IP layer - translating internal to external addresses 1-for-1. This was done primarily to avoid network renumbering when there was a change in provider. However, NAT-P - NAT with port translation - quickly

emerged. In NAT-P, a large number of hosts can utilize a single external IP address by using the UDP and TCP port numbers as a demultiplexing point.

In essence, the UDP or TCP port number became an extended version of the IP address - adding another 16 bits to the total amount of space, providing for 48 overall.

NAT-P in particular, often just called NAT, has seem extremely widespread deployment. Almost every residence with broadband Internet is running NAT (NAT-P to be specific). NAT is common in enterprises, and there are more and more cases of multilayer NAT within a single enterprise. Service providers have been known to NAT entire networks; for example some wireless networks give mobile devices net-10 addresses.

When put together, the implication is that basic packet transport between point A and point B is, these days, frequently possible only if the transport is UDP or TCP. This is because NAT-P requires one

of these two in order to utilize the 16 bit ports as as a demultiplexing point. NATs, which are part of the network, have become not just TCP and UDP-aware, they have become TCP and UDP *dependent*. Almost every NAT in deployment today will simply discard a packet above IP that is not TCP or UDP. The primary exception is ICMP. Some NATs will pass ICMP packets, but it is filtered by many. Consequently, it only sometimes works on the Internet. This is also making it difficult to rely on. Indeed, it's success rate is sufficiently poor that new mechanisms for path MTU discovery have been designed which work without ICMP [RFC4821].

This means that the operation of the public Internet is dependent on the existence of UDP and TCP traffic. While it is true that, in some cases you can get other transport protocols to run between two hosts, if you want RELIABLE transport on the Internet - transport that works between ANY two points - you have but two choices - UDP or TCP. Nothing else works reliably, or even close to reliably.

Its not just NAT. Firewalls are also TCP and UDP aware, and are often configured to discard non-UDP or non-TCP traffic.

4. New Transport Protocols

Does this mean that it is impossible to define new transport protocols? Fortunately, the answer is no.

New transport protocols can be defined. However, if the intention is that these protocols ever actually work on the public Internet, they need to run on top of the only available packet transport on the Internet - UDP/IP. UDP has replaced IP as the 'raw' point to point packet transport on the Internet. If congestion control or signaling or other features are desired, they must be layered on top of UDP, rather than in a new protocol beside it.

It is tempting to design a protocol on top of IP in such a way that it is "NAT friendly". In this context, "NAT friendly" means that the protocol has been designed to make ALGs for that protocol easy to implement. However, there is a vicious cycle that prevents such ALG functionality from ever being built. New features get added to products, such as NATs, when the market demands them. The market demands them when there is enough usage to create such a demand. However, if the protocol will fail utterly to work in the face of existing NAT, there will never be enough usage to create such demand. Even if there was, the amount of time it would take to upgrade all of the NATs on the Internet to support this ALG functionality is astoundingly large. Thus, the protocol will continue to be unreliable on the Internet, since there is always a chance that it

hits a NAT which doesn't support the necessary ALG functionality.

Consequently, designing protocols to be "NAT friendly" in this way does not work in practice. New applications and protocols MUST be designed to run on top of either UDP or TCP. Full stop. Examples of where this has been done after the fact is [I-D.tuexen-sctp-udp-encaps] for SCTP and [RFC3948] for IPSec. However, it is far better to define this at the beginning, and furthermore, have it as the one and only mode of operation. This avoid protocol choices, and therefore simplifies design and improves interoperability.

5. What about IPv6?

Will the IPv6 Internet share the same fate as the IPv4 Internet, and work only with UDP and TCP? It seems likely that the answer will be yes.

The primary reason is that the IPv6 Internet is not something that will appear overnight to replace the IPv4 Internet. It will run alongside it for a very long time. Hosts that have connection to the IPv6 Internet will find themselves frequently using IPv4 (in a dual-stack deployment), because the target host is available only on IPv4, or will find themselves communicating with via IPv6 to a v4-only host through NAT. In both cases, any protocols except for UDP and TCP based protocols will not work. And thus, the v6 host will need to utilize protocols that do work in all cases - ones based on UDP and TCP - rather than ones that work only in a few cases. And so, when we eventually do cutover to IPv6 only, it will be with hosts which have, all along, only been running protocols that run on top of UDP or TCP.

Another reason is that the IPv6 Internet will certainly be filled with firewalls, and if history is any guide for the future, only TCP and UDP are likely to work through such firewalls.

Finally, the IPv6 Internet may be filled with NAT anyway, despite attempts to provide ample address space.

Consequently, for an application designers perspective, why build an application on top of a protocol which doesn't work on the IPv4 Internet, and won't work on anything but a pure IPv6 network, when an application running on top of UDP or TCP will work everywhere?

6. Security Considerations

This document does not introduce any additional security considerations for the Internet.

7. Acknowledgements

The author would like to thank Dan Wing, Cullen Jennings, Scott Brim, Paul Kyzivat, and Dave Oran for their comments.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

[I-D.tuexen-sctp-udp-encaps]
Tuexen, M. and R. Stewart, "UDP Encapsulation of SCTP Packets", draft-tuexen-sctp-udp-encaps-01 (work in progress), November 2006.

[RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, January 2005.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

Author's Address

Jonathan Rosenberg
Cisco
Edison, NJ
US

Phone: +1 973 952-5000
Email: jdrosen@cisco.com
URI: http://www.jdrosen.net

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).