The ISP Column
A monthly column on all things Internet

Should the Network know what we're up to?

March 2003

Geoff Huston

The Internet has brought many changes to our concepts of networking and networking services. Sometimes these changes have been readily accepted and feel quite natural to both users and regulators. In other cases the changes have been more uncomfortable for some, and there has been considerable tension in coming to terms with the altered service model that is associated with the Internet.

In what way is this service model altered?

In a network-centric environment, it is the network itself that defines the services that can operate across the network. The network's edge devices act more in the role of a transducer, altering to the network signal to one that makes sense to the network's users.

The telephone network is a very good example of network adaptation to a single service model. Over the years it has been very precisely adapted to the role of supporting people speaking to other people. It's basic mission is to carry a pair of analog signals in real time. The bandwidth and dynamic range of the signals match very closely the human voice. The engineering of the capacity of the network is precisely attuned to the human behaviours and social networks that underpin telephony. Where compression techniques are used, they are tuned to compression that preserves the intelligibility of speech. Like many forms of biological evolution that leads to quite astonishing levels of specialization to the local environment, the telephone network is a highly specialized service environment. So closely is the service and the network intertwined, there is little that can be used to distinguish between the service and the network. Any additional services that are imposed upon the telephone network have to first be encoded such that it resembles the characteristics of a voice conversation.

The Internet, on the other hand, is not a single service platform. It is not a web access network, although it can be used as such. It is not a mail network, although it can be used as such. It is not an online chat network, although it can be used as such. Almost any service that can be translated into an exchange of binary data can be mapped into the Internet. The reason for this flexibility is that a robustly constructed Internet is one where the network itself makes no particular assumptions about the services that use it. Within the basic architectural model of the Internet, services are defined by the interaction of computers at the edge of the network, not by any interaction between the network and the devices at the edge.

In this model the network has no requirement to identify the services that may use the network, nor any requirement to react in different ways to various services. In this model, if the service uses a form of encryption that scrambles the interior payload of the IP packet, then the service traffic will be handled identically to an equivalent service instance that does not use encrypted payloads.

But of course, as is often quoted when discussion turns to the practical application of architectures, in theory there should be no difference between theory in practice, while in practice there often is.

Internet networks are increasingly becoming service-aware.

Before looking at the application, and mis-application, of this capability, lets first see how a network can be aware of the service. Or, in other words, how can a network know what services I, as a user, may be using?

Within IP, the service identifiers in the packet header are the protocol number in the IP packet header, and the port numbers in the TCP and UDP packet headers. These fields are designed to identify services to the end points of a network transaction.

> The Internet Assighned Number Authority (IANA) registry of assigned protocol numbers can be viewed at http://www.iana.org/assignments/protocol-numbers

In particular, the TCP or UDP port numbers are intended to identify the source service location and the destination service entity to the destination end point. By convention, certain port numbers are used as rendezvous points for particular services and are considered "well known" on the source or destination side of the communication.

> The IANA registry of such rendezvous points in terms of their port address can be found at http://www.iana.org/assignments/port-numbers

If the network 'eavesdrops' on these fields within the packet, then the network is in a position to make some guesses about the service that is associated with each packet.

For example, a packet with an IP protocol field of 6 and a TCP destination port field of 80 is part of a request from a client to a web server, using the HTTP protocol. A packet with a source of port 80 would be a response from a web server to a client, again part of the HTTP protocol. E-mail is delivered within a session that uses TCP port 25 as its destination port, and so on.

So why is this end-point service identification information useful to a network?

The most common use today is in network firewalls. A firewall can be thought of as a selective packet discard network element. The rules by which the packet discard is performed are based on the assumption that the packet headers identify a service. The firewall is often configured to discard all packets where the service is not recognised or is recognised, but regarded as 'hostile'.

Another example is in the area of auto-configuration of network-based differentiated services. In this model the active network element, such as a router, continually inspects the headers of network traffic, and dynamically adjusts its Quality of Service response to various forms of traffic based on the assumption of the demands of the implied service. A good example of this form of active response is one that attempts to preserve voice quality in a mixed voice and data IP network. In this model the network elements may monitor the relative load levels within the network and the amount of jitter and loss that is being imposed on packets that appear to be Voice Over IP (VOIP) packets. At some threshold the router may exchange signals with a control unit and then dynamically install packet filters that redirect VOIP traffic into high priority, low loss, low jitter queues

This aspect of network-based responses to the implied service identification with packets has been taken in various directions within the Internet. One direction is charging for network access using a charge schedule where the charges vary according to the service beinbg used. An early example is the charging scheme adopted by the New Zealand Academic and Research network in the early 1990's. The system used a number of active network elements that generated running counts of the volume of data associated with each of the member networks, broken down by service type and by time of day. This allowed the network's administrators to, for example, charge a higher rate for file transfer during daytime hours than at night. The intended result was using charging as a mechanism to alter the use profile of the network. The intention here was to provide some incentive for daytime use for services that were associated with interactive use, and nighttime use for automated batch transfers and other large volume background service tasks.

More recent examples are seen in association with VOIP services. VOIP traffic is commonly carried using a real time streaming protocol carried over UDP, and therefore normally carried over well known ports. It is possible to identify VOIP traffic by searching for IP traffic with a UDP protocol number and a Real Time Streaming Protocol UDP port address. iT is also possible to identify the session initiation traffic that sets up these VOIP flows by looking for traffic that uses the ports associated with the Session Initiation Protocol (SIP). In some applications this service 'signature" can be used to trigger low-latency higher priority queuing on the part of the network.

In some regulatory regimes VOIP traffic represents a significant regulatory and economic issue. Not only does VOIP have the potential to place service providers outside the scope of the prevailing regulated environment in such regimes, it also represents the potential for considerable amounts of revenue leakage due to bypassing the normal international call accounting settlement structures used in the public switched telephone network. It should come as no surprise to see such regimes react by insisting that Internet Service providers take steps to block VOIP traffic from their IP networks.

This seems like a somewhat extreme response to VOIP traffic. One relevant question arising from such actions is whether such blocking is feasible, or whether it is even in the interests of the users and the Internet itself.

As many a security firewall operator has found, it is very easy to alter the network 'signature' of a transaction so that the service is invisible to the network. It is also possible to alter the service fields in the other IP packets so that one service masquerades as another. Use of IP level encryption scrambles all the bits of the packet below the outer IP header, so that no TCP or UDP port information is available to the network. Another a approach is based on the observation that cooperating end points can use arbitrary port assignments. Ror example interactive telnet traffic can look like a secure web session by using TCP port 443 rather than the 'normal' port 23 address. Even simple tunnelling occludes the service identification field, and various forms of session rendezvous mechanisms, such as TCP MUX and SIP, can place services on arbitrary ports.

Increasingly common these days is a technique of embedding IP inside an HTTPS session. From the network firewall's perspective the session looks like a secure web session that a cache proxy cannot intercept and cache or filter. In reality the HTTPS layer is a thin camouflage shell, as the encrypted payload is commonly a secure shell session which in turn is used to tunnel various application sessions.

On a more general level, the invariable observed response to blocking access on the Internet is that of an inventive process of establishing ways to circumventing the blockage. In the case of blocking services by selective filtering on protocol and port addresses, the obvious response is to turn to dynamic port selection, or masquerading the offending service to look to the network as if it is an acceptable service, or various combinations of both responses. Its almost as if we are on a threshold here, where if we see wider use of various forms of service-selective 'blocks' being implemented into the Internet, then the response from the applications level will be to hide all service information from the network, and walk away from the concept of 'well-known' service ports and replace it with dynamic port assignation or overloading of one of two port addresses for all services and place the service selectors within the payload rather than in a network-visible location.

So whether its the corporate network administrator wanting to block all but 'safe' traffic, or a regulatory regime's desire to direct ISPs to block all VOIP traffic, or to block undesireable web content, for some value of undesireable, there appears to be a common outcome. The way that the network operator implements such directives is through active filtering elements within the network that inspect each packet looking for packet header values that relate to certain services, and let the packet past the inspection point based on a ruleset of allowed and denied services. And the common application and user response to such blockage mechanisms is to embed one service within another, so that the traffic masquerades as a service that is permitted through the filter. In other words, the inevitable response is that the service information is deliberately hidden from the network.

"So what!" you may say. Such port masquerading does not alter the Internet in any fundamental way, does it?

Maybe there is something here, however. To shift the model of the Internet from one where service information is available to the network, albeit in a limited form, to one where the majority of applications deliberately obscure their identity to the network and there is no remaining network-visible service identification information is a big step. Maybe we should think about this a bit more before enacting regulatory imposts that in effect force it to happen.

As the Internet Architecture Board (IAB) observes in a work-in-progress draft on this topic:

> "having stable and globally meaningful service identifiers visible at points other than the end systems can be useful for the purposes of determining network behavior and network loading on a macro level."

In other words knowing more about the characteristics of traffic on a network can assist in determining how to make improvements to the network. For example, Quality of Service (QoS) mechanisms are most appropriate to provide differentiated network responses to jitter and loss-sensitive traffic. If the proportion of such traffic in the network is insignificantly low then investing in a QoS deployment is probably unnecessary. On the other hand if the traffic level is over one third then the network operator may find the investment in QoS a productive one. While the Internet is sufficiently flexible to host almost any service, it is possible to tune its

operational characteristics to suit a particular service profile. Without network-visible service information, however, such efforts are little different to working in the dark.

Its not just the network engineers who are impacted by this move to hide service information from the network. Application developers and users are also impacts. As the IAB also observes in the same work:

> "application protocols that include dynamic port negotiation for both ends of a connection adds to the complexity of the application, and limits its applicability to those domains that do not include various forms of service filtering through port blocking mechanisms."

Various forms of encapsulation also add to the number of active intermediaries to support a session, and, as we've already learned from experience, the more the number of active intermediaries, the more vulnerable the system and the failures are more frequent and more complex.

It would be good if we felt that they were not being forced to head down this path of greater complexity within the network and within its applications. It would be good if we did not have to use an array of proxy agents, dynamic port negotiators, encapsulators and corresponding decapsulating agents just to be able to use the network. This path of invoking service-based network filters makes the network more complex, less robust and, ultimately, more expensive for us, the users.

It should be needless to say this, but maybe it is worth repeating: this is not what we had in mind for the Internet. The true leverage of the Internet relies on its simplicity and efficiency, not on how complex and cumbersome we can force it to be.

So perhaps the regulatory response to various economic, social and political pressures is not to blithely leap to a "solution' of advocating active service-blocking within the network. Such measures have cost but no benefit, either in terms of implementing the desire outcome or in preserving the robustness the Internet. Its often the case that the solutions to issues within economic, social and political domains are best phrased in terms of economic, social or political responses. In resorting to ad hoc technology solutions, we often find that they are accompanied by unintended consequences whose impact is far greater than the original issue.

As the IAB also observes:

> "From this perspective of network and application utility, it is preferable that no action or activity be undertaken by any agency, carrier, service provider or organization which would tend to cause end-users and protocol designers to generally obscure service identification information from the IP packet header."

There is no merit in adding complexity and irrelevant functionality to the Internet at the network level, and such network-level measures intended to block particular services are a classic example of carelessly creating irrelevant complexity. To quote Antoine de Saint-Exupéry:

> "A designer knows he has achieved perfection not when there is nothing left to add, but when there is nothing left to take away."

Antoine de Saint-Exupéry (1900-44) was an aviator pionieer of air-mail in the 1930s. He disappeared in 1944 in the Meditteranean on a military mission during World War II. He is the author of many works, imagined during his long solitary flights including "Night flight", "Ground of the men", and "Fighter Pilot". His most memorable work is "The Little Prince", a symbolic account of an encounter in the desert of a man with a young boy from an unknown planet.

Further Reading

The Internet Architecture Board work in progress document on the topic of Service Identification can be viewed at: http://www.iab.org/drafts/draft-iab-service-id-considerations.html

Disclaimer

The author is a member of the Internet Architecture Board (IAB). The opinions expressed in this article are entirely those of the author, and are not necessarily shared by the IAB as a whole.

The above views do not represent the views of the Internet Society, nor do they represent the views of the author's employer, the Telstra Corporation. They were possibly the opinions of the author at the time of writing this article, but things always change, including the author's opinions!

About the Author

GEOFF HUSTON holds a B.Sc. and a M.Sc. from the Australian National University. He has been closely involved with the development of the Internet for the past decade, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. Huston is currently the Chief Scientist in the Internet area for Telstra. He is also a member of the Internet Architecture Board, and is the Secretary of the APNIC Executive Committee. He was an inaugural Trustee of the Internet Society, and served as Secretary of the Board of Trustees from 1993 until 2001, with a term of service as chair of the Board of Trustees in 1999 – 2000. He is author of *The ISP Survival Guide*, ISBN 0-471-31499-4, *Internet Performance Survival Guide: QoS Strategies for Multiservice Networks*, ISBN 0471-378089, and coauthor of *Quality of Service: Delivering QoS on the Internet and in Corporate Networks*, ISBN 0-471-24358-2, a collaboration with Paul Ferguson. All three books are published by John Wiley & Sons.

E-mail: gih@telstra.net