June 2013
Geoff Huston

The Company You Keep

This story started earlier this year, with a posting to the Australian network operators' mailing list, asking if anyone had more information about why the website that was operated by an outfit called "Melbourne Free University" was inaccessible through a number of major Australian ISPs. When they asked their local ISP if there was some issue, they were informed that, "this was due to an Australian government request, and could say no more about it."[1] This was unusual, as it was very hard to see that this site would fall under the gamut of Australian Internet censorship efforts, or fall foul of various law enforcement or security investigations. As the name suggests, their web site was all about a community-based educational initiative. To quote from their website: "The Melbourne Free University provides a platform for learning, discussion and debate which is open to everyone […] and aims to offer space for independent engagement with important contemporary ideas and issues."[2] What dastardly crime had the good folk at the Melbourne Free University committed to attract such a significant response?

One Australian technology newsletter, The Delimiter, (delimiter.com.au) subsequently reported that their investigation revealed that the Australian Securities and Investments Commission (ASIC) had used their powers under Section 313 of the Australian Telecommunications Act (1997) to demand that a network block be applied by local Internet Service Providers.[3] That section of the Act falls under Part 14, the part that is described as "National Security Matters." The mechanics of the network block was to demand that all Australian ISPs block IP level address to the IP address 198.136.54.104.

As it turned out in subsequent ASIC announcements, it wasn't Melbourne Free University that had attracted ASIC's interest. What had lead to this block was an investment scam operation that had absolutely nothing in common with the Melbourne Free University. Well almost nothing. They happened to use a web hosting company for their website, and that web hosting company used name-based virtual hosting, allowing multiple websites to be served from a common IP address. Some financial scammers attracted ASIC's interest, and when ASIC formed the view that the scammers had breached provisions of Australian corporate and / or financial legislation, they used their powers under Section 313 to require Australia ISPs to block the website of this finance operation. However, the critical aspect here was that the block was implemented as a routing block in the network, operating at the level of the IP address. No packets could get to that IP address from all customers of ISPs that implemented the requested network level block. The result was that financial scammers were blocked, but so were Melbourne Free University and more than a thousand other web sites.

At this point the story could head in many different directions. There is the predominately Australian issue of agency accountability on their use of Section 313 of the Australian Telecommunications Act (1997) to call for the imposition of network-level blocks by Australian carriers, or concerns over the more general ability under this section for Australian government agencies to initiate such blocking of content without clear accountability and evidently without consistent reporting mechanisms. However, what specifically interests me here is not the issues about agency behaviors and matters of the application of national security interests and criminal investigations. What interests me here is that this story illustrates another aspect of the collateral damage that appears to have arisen from IPv4 address exhaustion.

How do we make too few IP addresses span an ever-growing Internet? Yes, you can renumber all the network's internal infrastructure into private addresses, and reclaim the public addresses for use by customers and services. Yes, you can limit the addresses assigned to each customer to a single address, and require the customer to run a NAT to share this address across all the devices in their local network. And these days you may well be forced to run up larger Carrier Grade NATs (CGNs) in the interior of your network so that each public IP address can be shared across multiple customers.

What about at the server side of the client/server network? If you can get multiple clients to share an address with a CGN, can you share a single public address across multiple services?

For the web service at least, the answer is a clear "yes". The reason why this can be done so readily in the web is because the HTTP 1.1 protocol specification includes a mandatory Host request-header field.[4] This field is the DNS name part of the URL being referenced, and must be provided by the client upon every GET request. When multiple DNS names share a common IP address, a web server can distinguish between the various DNS names and select the correct server context by examining the Host part of the request. This allows the server to establish the appropriate virtual context for every request.

This form of virtual hosting appears to be very common. It allows a large number of small-scale web servers to co-exist on a single platform without directly interfering with each other, and allows service providers to offer relatively low-priced web service hosting. And it makes highly efficient use of IP addresses by servers, which these days is surely a good thing. Right?

Well, as usual, it may be good, but it’s not all good.

As Melbourne Free University can attest, the problem is that we have yet to come to terms with these address sharing practices, and as a result we are still all too ready to assign reputation to IP addresses, and filter or otherwise block these IP addresses when we believe that they are used for nefarious purposes. So when one of the tenants on a shared IP address is believed to be misbehaving, then it’s the common IP address that often attracts the bad reputation, and it’s the IP address that often gets blocked. And a whole lot of otherwise uninvolved folk are then dragged into the problem space. It seems that in such scenarios the ability of clients to access your service does depend on all your online neighbors who share your IP address also acting in a way that does not attract unwelcome attention. And while you might be able to vet your potential online neighbors before you move your service into a shared server, such diligence might not be enough, in so far as it could just as easily be the neighbor who moves in after you that triggers the problem of a bad reputation.

Exactly how widespread is address sharing on the server side?

I haven't looked at that question myself, but others have. There is a website that will let you know about the folk who share the same address. When I enter the address 198.136.54.104 into http://sameid.net then I see that more than a thousand various domain names are mapped to this particular IP address. So in the case of Melbourne Free University, they were relying on an assumption that none of these 1,000 unrelated online services were attracting unwelcome attention.

Does this sharing apply to all forms of web services? Do secure websites that use Transport Layer Security (TLS) have their own IP address all the time, or are we seeing sharing there as well? By default, sharing of secure web services requires that all the secure web service names that coexist on the same service IP address need to be named in the public key certificate used within the startup of the TLS session. This means that the transport key security is therefore shared across all the services that are located at the same IP address, which, unless the set of services are actually operated by a single entity, represents an unacceptable level of compromise. For this reason, there is a general perception that if you want to use a TLS-enabled secure channel for your service then you need your own dedicated IP address.

But that's not exactly true. Back in 2006 the IETF published RFC 4366, which describes extensions to TLS that allows each service on a shared service platform to use its own keys for TLS sessions.[5] (This has subsequently been obsoleted by a revised technical specification, RFC 6066.[6]) The way this is done is that the client can include the server name of the service connection when it starts the TLS session, allowing the server to then respond with the desired service context and allows the session to start up using keys associated uniquely with the named service. So if the server and the client support this Server Name Indication (SNI) extension in TLS, then it is possible to use name-based server sharing and also support secured sessions for service access. Now, if you are running a recent software platform as either a server or a client then its likely that SNI will work for you. But if the client does not support SNI, such as the still relatively ubiquitous Windows XP platform, or version 2 or earlier of Android platforms, then the service client does not recognize this TLS extension and will encounter certificate warnings, and be unable to use the appropriate secure channel. At this stage SNI still appears to be relatively uncommon, so while it is feasible to use a shared server platform for a secure service, most secure services tend to avoid that and use a dedicated IP address and not require specific extension functionality from TLS.

But back to the issue of shared service platforms and IP-level imposed filtering.

Why are IP addresses being targeted here? Why can't a set of distinct services share a common platform, yet lead entirely separate online lives? If there are diverse unrelated services that are located on a common IP address then maybe a filter could be constructed at the DNS phase rather than traffic blocking by IP address. Certainly the DNS has been used as a blocking measure in some regimes. In the world of imposed filters we see filter efforts directed at both the DNS name and at the IP address. Both have their weaknesses.

The DNS filters attempt to deny access to the resource by requiring all DNS resolvers in a particular regime not to return an IP address for a particular DNS name. The problem here is that circumvention is possible for those who are determined to circumvent such imposed DNS filters. There are a range of counter-measures, include using resolvers located in another regime that does not block the DNS name, running your own DNS resolver, or patching the local host by adding the blocked DNS entry into the local hosts.txt file. The general ease of circumvention of this approach supports the view that the DNS filter approach is akin to what Bruce Schneier refers to as "security theatre."[7] In this case the desired outcome is to be seen to be doing something that gives the appearance of improved security, as distinct to actually doing something that is truly effective in improving the security of the system.

An IP block is the other readily available approach used as a service filter mechanism. Its implementation can be as simple as an eBGP feed of the offending (or offensive) IP addresses where the BGP next hop address is unreachable. It has been argued that such network filter mechanisms can be harder to circumvent than a DNS-based filter, in that you need to perform some form of tunneling to pass your packets over the network filter point. But in these days of TOR, VPNs and a raft of related IP tunnel mechanisms, that’s hardly a forbidding hurdle. It may require a little more thought and configuration than simply using an open DNS service to circumvent a DNS block, which may make it a little more credible as an effective block. So it should not be surprising to learn that many regulatory regimes use this form of network filtering using IP addresses as a means of implementing blocks. However, this approach of blocking IP addresses assumes that IP addresses are not shared, and that blocking an IP address is synonymous with blocking the particular service that is located at that address. These days, that's not a very good universal assumption. While many services do exist that are uniquely bound to a dedicated IP address, many others exist on shared platforms, where the IP address is no longer unique to just that service.

It’s not just the "official" IP blocks that can cause collateral damage in the context of shared service platforms. In the longstanding efforts to counter the barrage of spam in the email world, the same responses of maintaining blocking filters based on domain name and IP address "reputation" are used. Once an IP address gains a poor reputation as a spam originator and its value is placed on these lists as a spam originator, it can be a challenging exercise to "cleanse" the address, as such lists of spamming IP addresses exist in many forms and are maintained in many different ways.

In sharing IP addresses, it's not just the collection of formal and informal IP filters that pose potential problems for your service. In the underworld of Denial of Service (DOS) attacks, the packet level saturation attack is also based on victimizing an IP address. So even if your online neighbor has not attracted some form of official attention, and it has not been brought to the attention of the various spam list maintainers, there is still the risk that your neighbor has managed to invite the unwelcome attentions of a DOS attack. Again, here your online service is then part of the collateral damage, as when the attack overwhelms the common service platform all the hosted services inevitably fall victim to the attack.

For those who can afford it, including all those who have invested what is for them significant sums of money and effort in their online service, then using a dedicated service platform, and a dedicated IP address, is perhaps an easy decision to make. When you share a service platform, your online presence is always going to be vulnerable to the vagaries of your neighbors' behavior. But there are costs involved in such a decision, and if you cannot afford it, or do not place such a premium value on your online service, then using a shared service platform often represents an acceptable compromise of price and service integrity. Yes, the risks are a little higher of your neighbors attracting unwelcome attention, but that may well be an acceptable risk for your service.

And in those cases when your service is located on a shared service platform, if the worst case does happen, and you subsequently find that your service falls foul of a sustained DDOS attack that was launched at the common service platform, or your service address becomes the subject of some government agency's IP filter list, or is listed on some spam filter, you may take some small comfort in the knowledge that it's probably not personal. It's not about you. But there may well be a problem with the online company you keep.

Postscript:
What about www.melbournefreeuniversity.org? They moved hosts. Today they can be found at 103.15.178.29, on a server rack facility operated within Australia. Are they still sharing? Well, sameid.net reports that they share this IP address with www.vantagefreight.com.au. I sure hope that they've picked better online company this time around!

[1] http://lists.ausnog.net/pipermail/ausnog/2013-April/017911.html
[2] http://melbournefreeuniversity.org/?page_id=218
[3] http://delimiter.com.au/2013/05/15/interpol-filter-scope-creep-asic-ordering-unilateral-website-blocks/
[4] "Hypertext Transfer Protocol -- HTTP/1.1", R. Fielding et. al, RFC2616, June 1999. http://tools.ietf.org/html/rfc2616
[5] "Transport Layer Security (TLS) Extensions", S. Blake-Wilson at al, RFC 4366, April 2006 http://tools.ietf.org/html/rfc4366
[6] "Transport Layer Security (TLS) Extensions: Extension Definitions", D. Eastlake, RFC 6066, January 2011, http://tools.ietf.org/html/rfc6066
[7] http://en.wikipedia.org/wiki/Security_theater

Disclaimer

The views expressed are the authors' and not those of APNIC, unless APNIC is specifically identified as the author of the communication. APNIC will not be legally responsible in contract, tort or otherwise for any statement made in this publication.

About the Author

Geoff Huston B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region. He has been closely involved with the development of the Internet for many years, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. He is author of a number of Internet-related books, and was a member of the Internet Architecture Board from 1999 until 2005, and served on the Board of Trustees of the Internet Society from 1992 until 2001.

www.potaroo.net