The ISP Column
An occasional column on things Internet

February 2005

Securing Routing – An ISP's Perspective

Security, it seems, is a very relevant topic in today's Internet. How can you be sure that you are communicating with the party you intended to be talking with over the Internet? How can you be sure the conversation is private? How can you be sure its content has not been modified?Security comes in many forms, from looking at the integrity of applications that support communications, the integrity of translation of service identifiers into protocol transactions across the network, through to the integrity of the operation of the network itself.

In this article I'd like to explore one aspect of this activity, that of some "good housekeeping" measures to secure one of the basic elements of the network infrastructure, that of the Internet's routing system. Configuring routing within the Internet is often seen as being as much an art as an engineering activity or an applied science, and there is certainly a significant element of folklore in the operator environment through the relatively widespread application of a set of common principles in designing and operating a network's routing system.

The Threat Model

When we talk about securing large scale distributed systems, its typically the case that complete and absolute security is not a feasible outcome. The objective is often focused on measures of risk mitigation, as a trade-off between cost, complexity, feasibility, flexibility and achievable outcomes. Securing a system, from an operational perspective, is often a case of making a reasoned judgment to spend a certain amount of resources in order to achieve an acceptable risk outcome.

A distributed system also involves consideration of the level of mutual trust each operator must place in the actions of others. Here the judgment is to what extent one network operator can implicitly trust the actions of others, and to what extent the network operator should explicitly re-validate the information provided by peer networks before accepting the information and including it into the network's routing state.

The first step in such an exercise is to understand the threat model. This analysis can generally be described as a set of questions:

- What might happen?
- What are the likely consequences?
- Can the event be prevented, or its consequences mitigated?
- What is the cost of such measures of prevention or mitigation?
- Does the risk of the threat, and the extent of its consequences, justify the cost of implementation of specific security responses?

It is often the case that in this exercise the law of diminishing returns applies. A number of basic actions can achieve a reasonable basic level of integrity of the routing system, and further measures tend to consume

increasing amounts of resources to administer, while providing only marginal improvements in the cumulative outcome.

When considering the routing system, the common element of specific threats directed at the routing system is the intent to corrupt the routers' forwarding tables. In forcing a router to make incorrect forwarding decisions, the consequences are the misdirection of traffic, where traffic addressed to a set of destinations is forwarded along some other network path. This can result in the traffic being passed through a location where third party traffic inspection is undertaken, or where the traffic may be altered in flight, where the traffic is discarded, or even being directed to an unauthorized and corrupted clone of the original service. It may also be the case that the corruption of the routing tables may result in additional false addresses being added to the network's routing system, which, in turn, may be used to disrupt the routing system. Corruption of the router's forwarding table may even result in the router discarding all traffic passed to it, including all forms of control traffic, effectively isolating the router from the network completely.

Corrupting the routing system can result in:

- Misdirecting traffic (subversion, denial of service, third party inspection, passing off)
- Dropping traffic (denial of service, compound attacks)
- Adding false addresses into the routing system (denial of service attacks)
- Isolating or removing the router from the network

A very common cause of routing incidents stem from configuration errors, where errors in the manual entry of routes into configurations and databases then lead to an erroneous route entry being entered into the routing system. Of course not all incidents stem from such benign sources, and it is also the case that malicious attacks have been directed to the routing system in the past, and doubtless will occur in the future.

The threat model should also take into account that at any point in time in the Internet there is a set of compromised routers, so that there is a constant capability to inject corrputed routing information into the Internet. Its also the case that circuits may be compromized. This is perhaps most obvious in the context to wireless links where it is possible both to eavesdrop on wireless traffic, and, under certain circumstances, inject false data into the conversation.

This threat model should be combined with the observation that the Internet, among its numerous other roles, an extensively deployed platform for eCommerce applications. Subverting the network allows a variety of attacks to be performed at the application level, and this poses a considerable financial risk, not only to the application service provider but to the network operator as a potential contributory party. Given the critical nature of such threats, protecting the integrity of the routing system is a very necessary security objective.

The Objectives of Routing Security

When considering routing security there is one primary objective, namely ensuring that the network is capable of delivering data packets to their correct destination. In other words, the primary objective is to ensure the integrity of operation of the data plane of the network. In looking in more detail as to how routing protocols operate in order to fulfil this function, are two somewhat different areas of activity:

- protecting the operation of the routing protocols, and
- protecting the integrity of the payload of the routing protocols.

In the first case, protecting the integrity of operation of the routing protocol itself, the associated objectives include protecting the routing protocol from efforts intended to compromise the function of the routing protocol to maintain a correct network topology, destination reachability and the maintenance of "best" paths to each destination, and to protect the operation of the protocol itself.

In the second case the threat is the insertion of incorrect destination information into the routing system, and the resultant corruption of forwarding tables with incorrect information.

Routing Architectures

In order to look at methods associated with each of these objectives it is appropriate to briefly look at network architectures, from the perspective of an individual network manager. A network is a collection of switching elements and associated connectivity paths. "Edge" routers are switching elements that connect the network to external networks. If the network is an ISP these external networks include customers, upstream transit providers and various peers. "Interior" routers are internal to the network, and have no external connections.

Across the interior of the network an interior routing protocol is deployed. This interior routing protocol is responsible for topology maintenance, providing a complete view of the best path between any two points within the network. The interior routing protocol spans the collection of network routers, but no further. The interior routing protocol does not extend past the edges of the network. Typically an interior routing protocol only carries internal network addresses (interface and loopback addresses) that are used to identify BGP speakers and internal forwarding paths. Addresses associated with external networks are not normally carried in an interior routing protocol. ISP networks can be quite large, and one of the ways to ensure that the IGP can operate reliably in such cases of spanning a large network with dense internal interconnectivity is to limit the number of route objects being carried by the IGP.

At its edges a network uses an exterior routing protocol (EGP). These days the choice of an EGP is limited to a single candidate, the Border Gateway Protocol, or BGP. The exterior part of the BGP protocol, eBGP, is used to connect the edges of two distinct networks. The information carried in eBGP is concerned with maintaining a current view of which destinations are reachable via the respective networks. With eBGP there is no fine level of topology information about the "best" hop-by-hop path to reach a given destination, but instead the protocol supports a more abstract view of a "best" network-by-network path to each destination. At this level of iteration each network may have a set of local routing policies, such a preferred transit provider, or a preference for customer-advertised routers over upstream-advertised routes. Such policies are expressed as local preference rules in the network's BGP configuration. BGP also has an interior component, iBGP, which is used to synchronize the information between all BGP speakers, such that reachability information learned at one point is propagated to all other BGP speakers.

While particular routing designs are unique to each network, and vary according to the size and role of the network, the typical ISP network typically has a set of external relationships which can be classified to client networks, where the ISP is an upstream transit service provider, its own upstream transit providers and a set of peer networks which are used to for the mutual exchange of client traffic. Such networks typically use a routing configuration which holds all routing information associated with external networks in its BGP routing state, and use an IGP to maintain a current view of the internal topology of the network, carrying only routes associated with internal next hop addresses. The two protocol families are linked together by binding a BGP next hop attribute with an IGP-maintained route. One way of looking at this relationship between the two protocols is observe that each externally learned route is associated with the internal address of the exit point for that route. The IGP carries the information that provides the network with the best path to the exit point, and BGP carries the set of destination addresses that are associated with each exit point.

Its therefore the case that securing a network's routing system includes consideration of securing both the interior and exterior routing protocols, and examining both the configuration environment as well as the operational environment.

Basic Network Design Principles

It's a common principle that good network design is the foundation of any security framework, and routing security is no exception.

The basic principles of network design in this case is built upon effectively isolating the network's interior routing domain and using eBGP exclusively as the means of propagation of routing information across network boundaries.

Accordingly, it is highly inadvisable to operate the interior routing protocol across a network's boundary, in effect sharing the IGP state with an external network. Not only does this give the external network the ability to directly inject reachability information into the host network without any form of validation or filtering, it also has the potential for the external network to disrupt the operation of the local network's IGP. This could take the form of injection of a very large number of LSA's in the case of a link-state IGP as a part of a Denial of Service (DOS) attack, or the injection of a very large number of updates in the case of a dfistance-vector IGP. eBGP, on the other hand, has some basic protection mechanisms in the form of route filters on received and advertised information, the capability to protect against flooding of spurious route objects , and uses a reliable transport protocol to ensure that the traffic rate never exceeds the network's capacity and that each component of the BGP communication is not discarded through transient network congestion.

Some care should be taken in avoiding infrastructure tunnels. Infrastructure tunnels often represent an immediate solutions to various forms of product requirements, where the intent to connect one part of the network into a different realm, such as is found in engineering various forms of regional transit services, where the customer is only serviced with a select subset of the full reachability domain. The problem here is that the tunnel can be a source of operational complexity and insecurity. In the worst case the tunnel path could automatically switch to one which traverses a third party network, opening up the network traffic to thrid part inspection and deliberate disruption.

This principle of network isolation extends all the way down the protocol stack to the media access layer, so that the use of shared LANs at the network's edge to connect client networks is inadvisable. Such shared LANs run the risk of fate sharing through the use of a single spanning tree state in the shared access subsystem, and its certainly a common occurrence for an operator to experience outages in the access network due to spanning tree malfunctions in external networks which disrupt the operation of access LANs. Such shared LANs also run the risk of inadvertent joining of the respective network's IGPs. Having a client network's router become the designated OSPF LAN router for the network's edge LAN is far from a desirable situation. Shared LANs as a demarcation between a network and its clients and peers also may not effectively isolate the networks from each other. Even the use of VLANs to simulate point-to-point circuits across the shared runs the risk of disruption through the configuration of a client switch into the supervisor VLAN.

A useful additional principle is that of clear role demarcation with routers. One possible design approach is to ensure that edge routers that interface with external networks do not also take on interior roles of supporting the internal network 'core'. This is often a feature of network POP designs where a clear distinction is maintained between edge routers and core routers, both physically an in terms of the IGP and BGP roles. Peering connections should be terminated on separate routers from nternal backbone routing, and again from customer connections.

In summary, the guidelines at the design level are:

- Isolate your network at the edge:
  - Route all traffic at the edge
  - No shared LANs with external networks
  - No shared IGP with external networks
  - No infrastructure tunnels

- Isolate your customers from each other:
  - No shared access LANs
- Isolate routing roles within the network:
  - Exterior-facing interface routers
  - Internal core routers

Protecting Access to the Router

Of course the most direct way to compromise a routing system is to gain access to a router's configuration and alter it. Securing the means of entering configuration information into routers is a basic security task. The basic checklist of actions is listed below, and should be considered as a bare minimum of necessary tasks. One aspect to note is that not only is it necessary to protect against unauthorized access to a network's routers, but the copies of the routers' configurations should be handled with care. Knowledge of the type of router, as well as its configuration often represents an effective springboard for various forms of attack. Obviously there is much more that could be said here about common techniques for securing access to routers, but within the scope of looking at securing BGP, this may be getting away from the topic of this article, and it's a topic already well covered in the literature.

Configuration Tasks - Access

- Protecting routing configuration access
- ssh access to the routers
- filter lists for interactive access and snmp
- user account management
- access log maintenance
- snmp read / write access control lists
- protect all copies of configurations
- monitor configuration changes

IGP Configuration

The interior gateway protocol also requires some consideration in terms of configuration and security. The IGP is the basic tool used by the network to maintain an accurate current view of the network's internal topology, and whether the IGP is OSPF, IS-IS, EIGRP or RIPv2 there are a number of common tasks associated with securing this environment.

The first major caveat is to never share an IGP with an external network. While it may seem a convenient and simple way to set up an routing connection to a neighboring network, the shared IGP essentially adds the external network into the local network's security domain. Through actions in the external network it is possible for the IGP state to be compromised either in terms of the IGP's view of the network topology, or by deliberately corrupt routing information into the IGP. All of the threats listed above in terms of traffic misdirection apply to a compromised IGP environment. This extends to a further cautionary note of never permitting a third-party managed piece of equipment to be an active party to the network's IGP.

There is also the consideration about the use of shared LANs and the operation of the interior routing protocol. Shared LANs represent a critical point of vulnerability in terms of the operation of shared spanning tree protocol domains, VLAN integrity and the vulnerability to attack through manipulation of MAC addresses on the LAN, and care should be taken to avoid running an IGP across such network segments.

IGP Configuration Tasks

- Protecting the IGP
  - No shared IGP configurations
- Don't permit third party managed equipment to participate in IGP routing
- No IGP across shared LANs!
  - shared LANs represent a point of vulnerability

BGP Configuration

The exterior routing protocol is in effect the workhorse of the Internet. Routing entries passed into the exterior routing domain will, in the absence of any specific filters, be propagated across the entire Internet. Indeed if this were not the case there would be no Internet at all! Of course BGP is not inherently equipped with reality filters, and BGP supports a routing realm that is equally efficient at propagating incorrect routing information as it is in propagating correct information.

Efforts to enhance the integrity of BGP start with each network operator, and the activity falls into the major areas of securing the operation of BGP through careful configuration of the BGP routing domain, and attempting to validate the information being passed across BGP to ensure that the propagated reachability information is, in some sense, 'correct' information.

BGP operates as a collection of point-to-point conversations, using a protocol layered above TCP. Each end of the conversation is identified using a neighbor IP address. The first step in BGP is to make sure that the neighbor address information is indeed that of the neighbor you intend to use for a BGP session. Basic protection of BGP sessions from intrusion attempts, including denial of service attacks in the form of SYN floods can include the use of filters on the router to restrict access to TCP port 179 (the port address used by BGP) to the address pairs configured in the local router's BGP state.

Even so, TCP sessions are liable to various forms of disruption through well known attack vectors, including attempts to guess the TCP sequence number and using source address spoofing to inject a reset TCP packet into the BGP conversation. BGP supports an authentication mechanism (described in RFC 2385), which allows each end of the conversation to be configured with a shared password. This shared password is used as a seed to generate a hashing algorithm that uses the TCP header and the BGP data to generate a MD5 message digest for the data. The receiver uses the same seed to generate an equivalent message digest, and only accepts the message if the two digest values match for each message. Literature on TCP reset attacks indicate that a BGP session may be vulnerable to an attack within a few hours of TCP sequence number guesses, while the use of MD5 authentication in BGP adds significantly greater complexity (and time) to the attacker's attempts to successfully guess both the sequence number and the MD5 message digest for a reset attack. Obviously, conventional password management recommendations come into play here, as third party knowledge of the password totally negates any protection that MD5 may offer, including the consideration not to use the same password for multiple BGP sessions to different parties. It is possible to go a further step and use IPSEC to encrypt and protect the TCP session, but IPSEC does imply an additional processing overhead for every BGP message. This additional processor overhead has some implications on the performance of the BGP protocol, particularly when initially constructing a session state that may include the transfer of hundreds of thousands of

routes. Additional delays to the operation of BGP route propagation has some further implications in terms of the dynamic load of BGP and the time taken for BGP to converge to a coherent routing state.

There are some further measures that should be considered here relating to the stability of the routing objects. BGP uses the concept of a 'nexthop' address as the target of a routing object. When accepting a route object from an external network, the nexthop address for the route object can be translated to the local loopback address of the eBGP speaker that received the update, in order to avoid the potential for transient failures in the routing system due to instability in the non-local nexthop address.

A further potential source of disruption to BGP is the flooding of a massive number of BGP route objects from an external peer. To date most of the sources of such route floods have been caused by inadvertent operation misconfiguration, typically from unfiltered redistribution of the unaggregated internal routes into the external BGP session. This may cause memory exhaustion in the route receiver, as well as generating a high transient traffic load in further propagation of such route object floods. In response to this BGP implementations typically include a configuration option to specify the maximum number of route objects a BGP speaker will accept from a peer. This is a useful configuration option to consider, but again a word of caution is in order. The typical followup action that the router will take is to drop the BGP session once the overload condition has been encountered, which may further compound the problem by effectively isolating the peer network. There is some discussion within the operator community of the use of a hold action in response to an overload condition, where no further updates are accepted once the session threshold is reached, but the peer session will remain up and the existing routing information will be maintained. There appears to be considerable merit in such an action if the overall objective is to maintain as much connectivity as possible while protecting the local network from such potential overload conditions.

In general BGP is used to connect immediately adjacent routers at the edges of the network. However there are cases where the eBGP peers may not be immediately adjacent. In this case BGP allows the use of the multi-hop configuration option, specifying that the BGP peer lies across a network path. In such cases the potential for third party injection of traffic into the BGP conversation is increased dramatically. One form of mitigation here is to use the so-called TTL hack, where the BGP speaker will only accept as valid packets those packets with a TTL of a given value. As IP packets have their TTL field decremented by 1 for each forwarding hop, limiting the minimum value of the incoming TTL limits the effective radius of packet injection into the BGP session to that part of the network that lies within the specified TTL radius. But it's a relatively weak form of mitigation, and , from the perspective of routing security, there is a strong body of thought that simply says "Don't use multi-hop".

Also, in this laundry list of BGP configuration considerations, some considerable thought needs to be given to the architecture of BGP speakers within the network, considering the flow of eBGP information between all eBGP speakers, and the iBGP flow of information within the network along the default-free core of the network. Larger networks require the deployment of BGP route reflectors to reduce the impact of a full internal mesh peering of iBGP sessions between all BGP speakers. Not only should this topology consider various forms of redundancy in Route Reflector configuration to ensure improved availability of BGP information, but the transient traffic load generated by the establishment of external sessions that feed in a complete route set should be considered.

This transient traffic load is particularly high in the case of BGP session resets, where the entire BGP route set is removed, and then reinstated as a series of updates. The time taken to complete a full reset is proportional to the number of route objects. One way to alleviate this load is to use the so-called 'soft clear' mechanisms to perform session resets. Soft clearing of a session allows an active session state to be reset, but upon re-establishment the retained routing state is used as a starting point to rebuild the session, eliminated the extended update sequence normally associated with a session reset.

Of course running two routing systems concurrently, the interior and exterior routing systems, can be a source of endless operational amusement, if not a source of disruption and instability in the routing system. A good guideline is to attempt to delineate as cleanly as possible which routes are carried in which routing system. The IGP should be used to carry local router loopback addresses and interface addresses, and should be limited to just this route set. BGP should be used to carry all externally-learned routes. iBGP is a good place to carry routes for the network's data centres, singly-homed customer routes using sub-allocations of the network's address space, and similar internal address applications where the routes are not learned from external sources. Local addresses that are to be advertised into the external routing system should be generated as aggregate route objects, generated by static configuration commands at the network's edge. This implies that there should be no need to configure redistribution of BGP route objects into the IGP, and no direct redistribution of IGP routes into BGP.

BGP Configuration Objectives

- Protect the TCP session from intrusion and disruption
- Minimize the impact of session disruption on BGP.
- Minimize third party dependencies to a minimum (use local nexthop targets, for example)

BGP Configuration Tasks

- Basic BGP configuration tasks:
  - Use filter lists to protect TCP port 179
  - Use session passwords and MD5 checksums to protect all BGP sessions
  - For iBGP use the local loopback address as the nexthop
  - No redistribution from iBGP into the IGP
  - Use maximum prefix limiting (hold mode rather than session kill mode preferred)
  - Use eBGP multi-hop with care (and consider using TTL hack)
  - Align route reflectors with topology to avoid iBGP traffic floods
- Operating BGP:
  - Use soft clear to prevent complete route withdrawals
  - Use BGP session state and BGP update monitors and generate alarms on session instability and update floods

There's a lot of detail here, and much to consider. One way to check your configuration design is to compare it with some published configuration templates. There are a number of such templates that can be found. The author has found the work done at Team Cymru to be a very useful source for such a template, and Rob Thomas's configuration at http://www.cymru.com/Documents/secure-bgp-template.html merits some study in terms of comparing it to the approach you may be using for your network.

Securing the Route Payload

Of course protecting the integrity of the operation of the routing protocols is only one half of the task of protecting routing. The other half, and at times the more daunting task, is to protect the integrity of the payload of the protocol, or the integrity of the routing information itself. As a network operator you know which addresses are yours, and, hopefully, you know that they are valid addresses and your claim to be able to advertise them in the routing system is authentic. But what about the routes you learn from others? When you accept these routes into your network, and readvertise these routes to your routing peers, how can you apply a similar level of confidence in the routing information's validity? In other words, how can you ensure that what you learn from the routings system, and what you advertise into the routing system is authentic and accurate at all times?

In looking into this task at a finer level of detail for an ISP, there needs to be some consideration of the basic forms of interaction between networks in the Internet. In general an ISP operates three distinct forms of external route peering relationships: those that relate to customers, where the local network is acting as the external networks upstream transit provider, those that relate to ISP peers, where the local network exchanges customers' routing information with the external peer, and those that relate to the network's own upstream transit providers. The considerations relating to instilling some level of trust in the exchange of routing information differ to some extent across these three forms of external routing relationships.

Customer Routes

The basic handling procedure for managing customer-originated routes is to firstly to validate the customer's route requests, to provide a level of assurance that the address block is valid and the customer has a valid claim to originate a route to this space, and secondly to place an entry in the incoming route filter associated with the eBGP session with the customer that permits the customer to announce that route. Being at the edge of the network there is some support for the proposition to also apply some form of route flap dampening to the route, so that if the announcement becomes unstable the network will ignore further announcement flaps for a period of time. On the other hand, route flap dampening is not altogether a benign configuration to add to your BGP sessions. The BGP protocol is a noisy protocol, and a single network event at one location can cause a transient burst of updates and withdrawals at remote locations, based about the operation of the BGP protocol in searching for a new stable routing state. These transient bursts can trigger flap dampening actions, transforming a multi-second event with minimal impact on the data plane into an extended multi-hour event with potential complete denial of service. These days it is, perhaps, a matter of taste, but I see BGP flap dampening to cause more trouble than its worth, and I would be reluctant to deploy it.

The most critical aspect of managing routes presented to the network by its clients is authenticating the validity of the routing request. Where the service provider is operating its own address pool and passing out sub-allocations to customers the task is to associate the routing request with the customer allocation, and ensure that the requests match. Where the customer has obtained the address space from other source then the authentication task may become more convoluted. Now the network operator has to track down the delegation history of the address prefix, looking first for an original allocation record in the Regional Internet Registries' public databases for the address prefix, or a covering aggregate, in order to confirm that the address space is valid, rather than being held in existing reserved blocks. The operator, if it is being diligent needs to then establish whether the organization described in this database report is identical to the identity of the customer. If not then the operator really should confirm the sub-delegation sequence that has lead to the customer being delegated this address block, and confirming with each of the delegating entities in the chain that they do not have prevailing policies that deny the ability to separately announce the address space. In the worst case this task, if performed diligently, is a time consuming and complex task. This, in turn, equates to a considerable cost to the network operator in processing such routing requests. Customers have grown to expect immediate service activation turnaround, and taking extended periods of time to independently validate a customer's assertion of their ability to originate a particular route is often seen as being an unnecessary delay and a needless administrative overhead by the customer, and may also be perceived as a costly overhead by the ISP's product management. It is a seductive option to make the operational decision to accept the customer's assertion of their ability to originate the route, and for the ISP to accept the customer request immediately and activate the service. Obviously such a decision, although being seen as both a customer-friendly process and one which also reduces network administrative costs, is one with far reaching implications. The problem here is that the customer may be, deliberately or inadvertently, misrepresenting the viability of their claim to be able to route that particular address block. It may be an address block that has been allocated to a different entity, and, in the worst case, the customer is deliberately attempting to subvert the routing system to steal the traffic to be redirected to their systems. Even if the network operator opts to perform due diligence in independently validating routing requests, the address may be part of a long-standing allocation and there may be a legitimate

dispute over current control over the address space. When the ISP is being placed into the position to accept one party's claim to announce the route in preference to another then this may place the ISP in a less than impartial position with respect to the dispute. Also the information available to validate the routing request, and particularly with various forms of mergers and acquisitions the subsequent transfers of address across organizational entities may be very challenging to trace. Even with such a process of due diligence in tracing back the address routing request to the original allocation actions and subsequent transfers and sub-delegations, at times there may not be a definitive answer.

One potential solution is to use a Routing Registry (RR), where the customer can enter their address prefixes, and the associated policies, in the registry. The network operator can then use the registry to create route filter lists that can be loaded into their edge routers. This provides some level of authenticity to the customer route advertisements, in that inadvertent (or deliberate) routing misconfiguration then has to occur both when the entry is made into the route registry and again when the entry is placed into the customer's outbound route advertisement, but this is not a very high level of assurance. Some effort has been invested into the maintenance of RRs tied to the allocation path, where recipients of address resources can generate registry entries describing those resources and their associated advertisement policies, while transit providers can in effect describe their transit policies in terms of inter-AS connectivity and routing preferences. It's a step in a useful direction, as it has the potential to offload one part of the expensive activity of manually verifying the authenticity of a customer routing request and replace it with the task of loading up configuration filter lists derived from RR data. But Route Registries tend to suffer from a level of benign neglect in many cases, where the information in the registry is incomplete, and where old entries are not removed or updated, and it is still possible for false information to be injected into the registry. The implication is that manual validation of routing requests continue to be a part of an ISP's cost, and the maintenance of route registry entries becomes yet another cost element rather than a means of both cost reduction and instilling higher levels of authenticity in the routing system.

So the task of validating a customer's routing request, particularly for address space that was obtained through other sources is not a straightforward one, and probably represents the most critical point of weakness in the entire routing domain today.

Customer Routes

- Authenticate customer routing requests:
  - Check validity of the address
- Own space validate request against local route object registry
- Other space validate request against RIR route object database registered POC
- Adjust explicit neighbor eBGP route filters to accept route advertisements for the prefix

Inter-ISP Peer Routes

Once you move more than a step away from the point of origination of a route object that task of validation of a route becomes more challenging. The next class of route objects to examine are those which are accepted from peer ISPs, with the mutual exchange of customer-originated routes. Here the ISP typically takes the option of using a higher level of mutual trust and does not, and probably can not explicitly independently validate each route object that may be received from the peer ISP. The typical process used for peers is to accept the validity

of all peer-generated routes, subject to a filtering of routes. In many cases a Route Registry is again a useful mechanism to perform a sanity check of the advertised routes through the use of a generate routing filter.

But in security terms 'mutual trust' is not a reassuring term. If the peer network admits and readvertises corrupted routing information then the local network would also accept and readvertise this same corrupt information. In this case the integrity of your local routing information is based on that of your peers.

As well as the use of route registries to maintain peer filter lists there are a small number of additional basic measures that can be used to protect the local session. These include the use of maximum prefix thresholds for the peer sessions in order to prevent the peer from flooding the local network with prefixes and run the risk of disrupting the local routing system through memory exhaustion on the local routers is advisable> of course the same caveat applies here, that the preferable option when the threshold is reached is for the session to remain up and no further updates be accepted, in preference to tearing down the session and awaiting a manual reset.

However the basic issue here is that mutual trust in the integrity of the route objects being passed across the peering session is a major aspect of this form of routing relationship, and in terms of assurance of integrity of this information, there is a very low level of trust.

Peer Routes

- Higher level of mutual trust
- Accept peer routes - apply local policy preferences
- Filter outbound route advertisements according to local policy settings
- Use max prefix with "discard-over-limit" action (if available)

Upstream Routes

The relationship with an upstream transit provider can also be characterized as an asymmetric trust relationship, where the ISP is placed in the position of having to trust the upstream to advertise correct routing information, while the upstream may request the ISP to provide some level of assurance regarding the validity of the routes advertised to the upstream. In this case the ISP can, at most, adopt a defensive position in terms of the routes that should be accepted, and filter out the most obvious candidates for incorrect routes. A typical route filter for an upstream service provider would be to filter out the most obvious candidate incorrect routes, including the private addresses, your own local routes, and potentially also filtering out any routes that originate from address space that is yet to be allocated for use. The use of maximum prefix thresholds should be considered carefully in this context, in that the normal action of exceeding the threshold, that of tearing down the BGP session, results in isolating your network from your upstream service provider.

Upstream Routes

- Asymmetric trust relationship
- Apply basic route filters to incoming route advertisements
  - RFC 1918 routes
  - own routes

Securing Routing

The overall picture is not all that good is it? It appears that while an operator can do a relatively robust job in protecting their routing infrastructure in terms of protecting routers and the integrity of operation of the routing protocol, the task of protecting the integrity of the payload is a much more frustrating task. Even if the

ISP is prepared to devote considerable time and expense in being diligent about researching the authenticity of every routing request coming from their customer base, the ISP still is in the position of having to accept routes from their peers and upstreams, without any particular assurance of the quality and authenticity of that information. And its not clear that the ISP is strongly incented to perform thorough and exhaustive checks on the authenticy of routing requests, particularly when they relate to non-locally assigned address space. The business proposition here is that this activity is a net expense to the business, with no incremental revenue to offset the expense. So the ISP often performs cursory checks, but not to a level of detail that would detect a determined effort to inject incorrect information into the routing system. Unfortunately integrity of routing is one of the cornerstones of integrity of the operation of the Internet itself, and the picture today is that we are not in a strong position to defend the Internet against determined efforts to disrupt it at this level.

But we should not throw up our hands in frustration. There are approaches that the ISP industry can adopt that can add strong elements of authenticity into the payload of routing protocols. In the next article I'd like to look at such measures.

Geoff Huston

Disclaimer

About the Author

Geoff Huston B.Sc., M.Sc., 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 has been active in the Internet Engineering Task Force for many years.

www.potaroo.net