BGP ROUTING SECURITY OUTLINE

-- Most users have no idea how Internet traffic gets from one ISP to another. Network engineers, however, can tell you that BGP (the "Border Gateway Protocol") is the underlying magic (technically, the "exterior routing protocol") that helps packets get where they need to go. (Relevant BGP RFCs can be seen at http://www.bgp4.as/rfc). Given the size of the Internet, and the basic simplicity of BGP, the fact that BGP works and scales as well as it has is really quite impressive.

-- Unfortunately, BGP is subject to a variety of intentional (and accidental) attacks. One such attack (but not the only such attack) is known as "route injection" or "route hijacking." In a route injection attack, a site "injects" or "advertises" an unauthorized route via BGP for someone else's address space. When that happens, particularly if the injected route is "more specific" than the normally-present route, network traffic that *should* be going via an authorized route to its real destination instead gets misrouted ("hijacked") via the unauthorized/evil competing "more specific" route. This has been described by *Wired* as "The Internet's Biggest Security Hole," see http://www.wired.com/threatlevel/2008/08/revealed-the-in/

-- Route injection attacks have been observed many times on the Internet. One of the most famous route injection incidents occurred in 2008, when Pakistan, in an attempt to domestically limit access to the video sharing site YouTube, accidentally leaked routes for YouTube's address space worldwide. The route monitoring company Renesys has a nice summary of this incident, see "Pakistan Hijacks YouTube," http://www.renesys.com/2008/02/pakistan-hijacks-youtube-1/

-- Besides those sort of accidental incidents, as the Internet comes increasingly close to exhausting its supply of IPv4 address space, we expect that we may see more and more address space hijacking attacks by spammers and other miscreants. BGP route injection may also be exploited for espionage purposes by nation state intelligence services, temporarily rerouting selected traffic, eavesdropping upon it, and then silently reintroducing it for eventual delivery to its actual destination.

-- Multiple approaches have been tried over the years to prevent these sort of activities. A minimum standard of care entails checking whois to verify assignment of any provider-independent address block an ISP gets asked to route for a customer, and requiring customers to provide a letter of authorization if the provenance of a particular netblock is at all clouded. See, e.g., http://business.comcast.com/enterprise/services/internet/ethernet-dedicated-internet/edi_tech_specs at section 5.9.

-- Another approach is use of "routing registries" (see http://www.irr.net/) with "Routing Policy Specification Language" (RPSL). A nice introduction to "Using RPSL in Practice" can be seen in RFC2650, by Dave Meyer et. al. In a nutshell, routing registries allow ISPs to describe the routes they originate, and the ASs ("Autonomous Systems") that should be announcing them. If everyone was conscientious about documenting their routes in routing registries, and all network service providers built their operational routing filters directly from routing registry data, it would be difficult for a third party to accidentally hijack another site's address space. An example of an ISP that requires customers to use a routing registry can be seen at http://www.us.ntt.net/support/policy/routing.cfm#RR Regrettably, in practice, routing registries are not as widely used as they might be, and the data that's in them is often stale/out-of-date, or otherwise unacceptably accurate.

-- There has been growing community interest recently around RPKI. RPKI uses cryptographically-verifiable certificates, known as "Route Origin Authorizations," or "ROAs", to specify what ASNs are authorized to originate a particular prefix. ROAs are normally issued by the RIRs (ARIN, RIPE, APNIC, etc.), see for example https://www.arin.net/resources/rpki/ Unfortunately, as noted in http://tools.ietf.org/id/draft-ietf-sidr-origin-ops-22.txt at section 7, RPKI is not intended to deal with malicious/intentional route injection, but only inadvertent incidents (such as the Pakistan YouTube incident).

-- Another stream of work-in-progress involves BGPSEC, see http://tools.ietf.org/html/draft-ietf-sidr-bgpsec-overview-03 and http://tools.ietf.org/html/draft-ietf-sidr-bgpsec-protocol-07 (draft expired August 25th, 2013). BGPSEC builds on RPKI, but endeavors to secure the *chain* (or ASPath) of autonomous systems that should be originating each authorized prefix. A discussion of the threat model underlying BGPSEC can be seen in http://tools.ietf.org/html/draft-ietf-sidr-bgpsec-threats-07

-- While this all gets worked out, one alternative focuses on at least *detecting* route hijacking if/when it happens. Detecting route hijacking typically depends on the availability of routing data from ISPs all around the world, since "every routing table is different." (One such repository of routing data is Dave Meyer's Oregon Routeviews Project, http://www.routeviews.org/ ; another resource is http://www.ripe.net/data-tools/stats/ris/ris-raw-data). Some companies also offer productized route monitoring, see for example http://www.bgpmon.net/services/route-monitoring/ (free for up to five prefixes)

-- If the risk of route injection isn't intelligently managed via one of the preceding options, some ISPs may attempt to reduce their risk of experiencing route injection via deaggregation, or the announcement of multiple more specific netblocks (rather than using maximally-aggregated routes). When this happens, every border router on the Internet ends up getting penalized due to having to carry those gratuitous additional routes. See for example http://www.cidr-report.org/as6447/#Gains