

IPv6: The Slow Innovation Speed of the Core Infrastructure

Paneldiskussion: Die Zukunft von IPv6

September 2012

Hannes Tschofenig <hannes.tschofenig@nsn.com>

Status: IPv6 Deployment Increasing

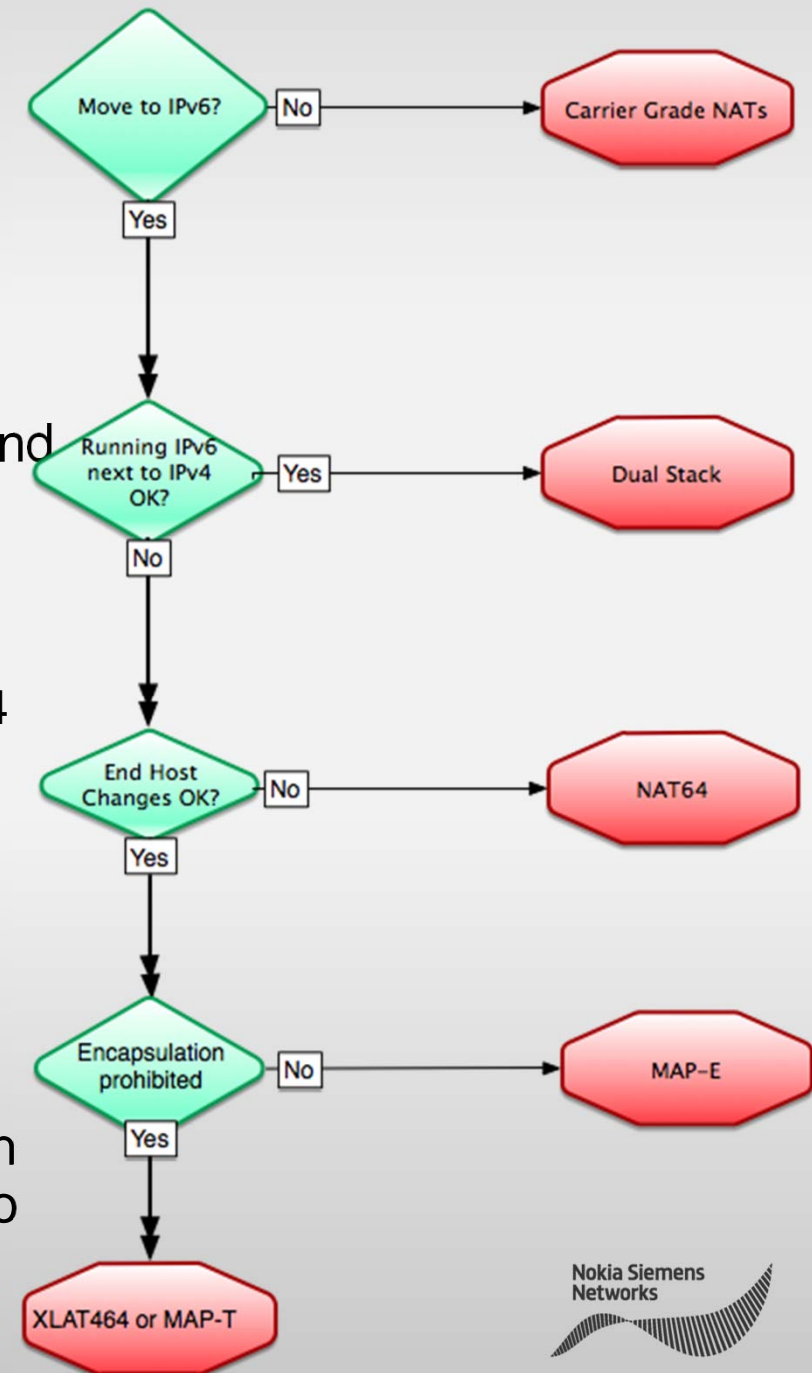
- Around 20 years of standardization work in the IETF and other SDOs.
- Addresses are (finally) running out
 - [31st January 2011](#): IANA allocates two blocks of IPv4 address space to APNIC
 - [14th September 2012](#): RIPE NCC Begins to Allocate IPv4 Address Space From the Last /8
- IPv6 Deployment finally increases (slowly)
 - World IPv6 Day (June 2011): IPv6 traffic growing from 0.06% to almost 0.15% of all Internet traffic in the weeks leading up to June 6 according to [Arbor Networks](#). Additional data available at <http://www.google.com/ipv6/statistics.html>
 - Alexa 1000 list shows that big web content providers & popular sites have turned IPv6 on since world IPv6 launch day.
- Lots of IPv6 deployment stories in the meanwhile. For example, DSL provider using a Dual Stack solution.
 - Sep. 2012 by : 24.92% RCS-RDS customers are using IPv6 now, see <https://ripe65.ripe.net/presentations/135-RDS-IPv6-ripe65.pdf>
- Support for IPv6 in equipment is getting better (phones, laptop/PC, servers, etc.)

Transition Path Differs

Example: Mobile Operator

Notes:

- Carrier Grade NATs (work done in IETF [V6OPS](#) and [BEHAVE](#)) cause various problems, see [RFC6269](#) and [draft-ietf-intarea-nat-reveal-analysis](#).
- Dual Stack only helps with address shortage in the long term.
 - May require two PDP contexts (one for IPv4 and another one for IPv6)
 - IPv6 requires software changes in the network (e.g., for charging and billing)
- NAT64 is a NAT - consequences for applications. Even apps that work with normal NATs break (e.g., [Skype](#))
- Changes to end hosts lead to less state in the network. But applications may need to be modified.



Application Area is changing around us...

- Main trends:
 - Application development focuses on Client-Server model (not p2p anymore)
 - HTTP/HTTPS is used as a foundation.
 - More aggressive multiplexing (e.g., HTTP 2.0 proposal SPDY <http://code.google.com/speed/>)
 - Update cycles get faster - JavaScript-based Web applications (e.g., HTML5) & application stores
- Applications today already need to deal with NAT and firewall traversal to work in a robust fashion.
- Keeping state alive consumes energy (end devices)
 - http://www.pasieronon.com/publications/haverinen_siren_eronen_vtc2007.pdf
- Keeping multiple connections alive through a NAT is more expensive
 - Connectivity tests needed and state has to be kept alive for reachability.
 - For connectivity tests [ICE](#) may be used.
- Would be great to tell NAT how long to keep bindings or to learn other NAT characteristics
 - Examples: UPNP, NSIS [RFC 5793](#), PCP <http://datatracker.ietf.org/wg/pcp/>

My Recommendation

The road to the future is bumpy.
Go for Dual Stack and get over it.