An experiment with stateless NAT64

Carlos Martinez – LACNIC
@carlosm3011
NAT64

• We all know what it is...
  – Connect IPv6-only clouds with IPv4-only clouds
  – Translate protocol headers including addresses
  – Source IPv4 address for translated packet
    • Multiplexed using TCP / UDP port numbers: Stateful NAT64
    • One-to-one mapping: Stateless NAT64
  – Destination IPv4 address is embedded / encoded in IPv6 destination address
Stateful NAT64

IPv6 Packet
- SRC: 2001:db8::128
- DST: 64:ff9b::42dc:9e19
- <<other headers>>

TCP Frame
- SRC PORT: 32768
- DST PORT: 80

Algorithmic Address Mapping

IPv4 Packet
- SRC: 190.216.38.14
- DST: 69.63.190.18
- <<other headers>>

TCP Frame
- SRC PORT: 15547
- DST PORT: 80

NAT64 Router
Stateless NAT64

IPv6 Packet
SRC: 2001:db8::128
DST: 64:ff9b::42dc:9e19
<<other headers>>

Algorithmic Address Mapping

NAT64 Router
SRC address is mapped one-to-one to the IPv6 hosts

IPv4 Packet
SRC: 190.216.38.X
DST: 69.63.190.18
<<other headers>>
Introducing TAYGA

• TAYGA is a user-mode, stateless NAT64 implementation
  – Uses the TUN driver
• Mapping between IPv6 hosts and IPv4 hosts is one-to-one
  – You need to have as many IPv4 addresses as hosts you want to have
• [http://www.litech.org/tayga/](http://www.litech.org/tayga/)
Configuring TAYGA

• {Taken from TAYGA’s website}
• Compile:
  – The usual ./configure && make && make install
• Create /usr/local/etc/tayga.conf:
  tun-device nat64
  ipv4-addr 192.168.255.1
  prefix 64:ff9b::/96
  dynamic-pool 192.168.255.0/24
  data-dir /var/db/tayga

• Create tun device and configure static routes and addresses
The Experiment

- LACNIC 18, Montevideo, Uruguay in October 2012
- 350 attendees, ~550 devices
- Multiple SSIDs
  - Dual-Stack, IPv6-Only+NAT64
The Network
Network Setup

• NAT64 to a private IPv4 pool
• Stateful NAT44 configured in the Cisco boxes
  – Could have used the servers themselves with `iptables -j MASQUERADE`
Network Setup (ii)

- Server #1 runs TAYGA and performs DNS64
- Server #2 only performs DNS64
- Cisco box does NAT
**DNS64**

- Implemented with BIND 9.8
- DNS64 synthetic answers restricted to the IPv6-only VLAN
- Very simple configuration:
  ```
  dns64 64:ff9b::/96 {
    clients <lacnic18pfx>:b0b0::/64;
  }
  ```
Some Results

• Number of users
  – Not many, 10-12 devices peak
• Performance
  – No noticeable degradation compared to the dual stack SSID
• What works and what doesn’t
  – Skype
  – Dropbox
  – Some users reported Twitter not working, couldn’t confirm
Some results (ii)

• The good about it:
  – No kernel modules needed
  – No unnecessary or possibly conflicting** IOS upgrades to perform on the Cisco boxes
  – Easier troubleshooting / user tracking as every NAT64ed hosts has its own IPv4 address
    • Stateful NAT performed at network’s edge
THANKS !