QMUL experiences with enabling IPv6

Terry Froy, Daniel Traynor
QMUL, 11/09/2015

IPv6 @ QMUL
(a brief history)

2003

Assigned 2001:630:11::/48 by JANET

2007

IPv6 deployed on central IT Services network

/52 assignment routed to department of Electronic Engineering and Computer Science

Early 2012

QMUL undergoes major IT transformation

/48 no longer sufficient

IPv6 @ QMUL
(a brief history)

- Request larger assignment from JANET
  - JANET NOC wanted to know why.
- Require justification for allocation larger than /48 (as per RIPE policy)
  - Who are RIPE ?
- 'Further developments' require multi-homing of QM network
  - What is multi-homing ?
- More cost effective to join RIPE...
  - Why join RIPE ?

IPv6 @ QMUL
(a brief history)

- Autonomous System Number (ASN) and /32 IPv6 allocation obtained directly from RIPE
  - What is an ASN ?
  - Why did we get a /32 ?
- QMUL Tier-2 operates 2a01:56c0:4033::/48
  - Why did we assign a /48 to a Tier-2 ?

IPv6 @ QMUL (a brief history)

- Mid 2012
  - QMUL joins RIPE
  - /32 allocated to QMUL
  - /48 (from the /32) assigned to Tier-2
- 2013 – 2015
  - Migration of hosts from IPv4-only to dual-stack + MTU 9000 network
  - Participation in IPv6 testbed and production services

Jumbo Frames on JANET

- JANET confirm they can support jumbo frames.
  - What does this mean for Tier-2 ?
- Two possible deployment strategies:
  - 'flag day' (involves co-ordination with Central IT)
  - Controlled migration from existing to new (MTU 9000) network.
- Guess which one we picked ?

IPv6 and Jumbo Frames

- New network provisioned (v6: 2a01:56c0:4033::/64 – v4: 194.36.11.0/24) with MTU of 9000
- Migration process (roughly):
  - Change IPv4 and IPv6 configuration
  - Add MTU=9000 to ifcfg-ethX
  - 'service network restart'
  - Change VLAN tag on associated switchports
- Remember to drop TTLs on your DNS records!

Issues seen 'in the wild'

- pMTU discovery and/or jumbo frames broken in parts of the JANET network (not IPv6 related).
- IPv6 routing occasionally not optimal.
- IPv4 routing occasionally not optimal.
- Broken firewalls (not ours!)
- Throughput concerns.
  - A lot of routing hardware out there will forward IPv4 in ASIC (hardware) but will only forward IPv6 in CPU (software).

IPV6ification of Cluster

- Basic setup of world facing nodes with public and private ip address + compute and storage nodes behind nat with private address only.
- Vlan with no ipv6 route, ignored ipv6 settings.
- Move servers one by one onto new IPV6 enabled vlan with dual stack addresses. Care needed!
- Non production services moved first. IPV6 addresses (AAAA) added to DNS record from start

Disable IPV6

- Automatic ip address allocation – race condition can mean that you get an IPV6 address based on MAC address.
- First thing to do is to learn howto disable IPV6!
- /etc/sysctl.conf

```
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
```

-
- Ifcfg-XXX

```
IPV6INIT="no"
IPV6_AUTOCONF="no"
```

IPV6 with nat

- Nodes in the nat need to talk to each other using IPV4. Will prefer IPV4 to IPV6. /etc/gai.conf (get address info).
-

```
#
#    For sites which prefer IPv4 connections change the last line to
#
precedence ::ffff:0:0/96  100
```

Typical IPV6 host setup

- DNS entry

```
[root@se02 ~]# host ce04.esc.qmul.ac.uk
ce04.esc.qmul.ac.uk has address 194.36.11.23
ce04.esc.qmul.ac.uk has IPv6 address 2a01:56c0:4033::17
```

-
- Ifcfg-XXX

```
[root@se02 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE="eth1"
NAME="eth1"
BOOTPROTO="static"
HOSTNAME=se02.esc.qmul.ac.uk
IPADDR=194.36.11.19
NETMASK=255.255.255.0
GATEWAY=194.36.11.1
IPV6ADDR="2a01:56c0:4033::13/64"
IPV6INIT="yes"
IPV6_AUTOCONF="no"
IPV6_DEFAULTGW="2a01:56c0:4033::1"
ONBOOT="yes"
USERCTL="no"
NM_CONTROLLED="no"
MTU=9000
```

VMhost setup

- VMhosts do not have external IP address but do need bridge to external vlan as VMguests do have external IP address.
- Issue with automatic Ip address allocation!
- Disable IPV6 address but keep ipv6 stack enabled. Disable IGMP snooping.

```
# in /etc/sysctl.conf
# keep the IPv6 stack functional but not assign IPv6 addresses to any of your network devices
net.ipv6.conf.all.disable_ipv6 = 1
```

```
# in /etc/rc.local
# disable on IGMP snooping on the bridge interface
#
echo 0 > /sys/devices/virtual/net/br1/bridge/multicast_snooping
```

GGUS-Ticket-ID: # 115017

- Synopsis:
  - “We at QMUL recently added IPv6 dual stack address to our CEs (e.g ce05.esc.qmul.ac.uk) including a DNS entry. As a result we stopped getting pilot jobs from ATLAS.“
- Diagnosis:
  - “After investigation we discovered that the pilot factories at CERN (e.g. aipanda017, but expected to affect all pilot factories) have a default IPv6 route but no routeable IPv6 address on the server.”
- IPv6 connectivity from this ATLAS pilot factory was definitely broken but why was it not falling back to IPv4 ?

GGUS-Ticket-ID: # 115017

Do we have IPv6 connectivity ?

- If yes, can we obtain a AAAA record ?
  - Connect to IPv6 address returned in AAAA record
    - Did it work ?
      - If yes, we have **working** IPv6 connectivity and the endpoint accepted our IPv6 connection.
      - If no, we throw a 'Connection timed out' error and give up!
- If no, we fall back to IPv4...

Do we have IPv4 connectivity ?

- If yes, can we obtain a A record ?
  - Connect to IPv4 address returned in A record
    - Did it work ?
      - If yes, we have **working** IPv4 connectivity and the endpoint accepted our IPv4 connection.
      - If no, we throw a 'Connection timed out' error and give up!
- If no, throw a DNS resolution error and give up!

GGUS-Ticket-ID: # 115017

- CERN response:

  'Closing this and continue tests on the devel factory and test CEs, avoiding changes to production systems.'

- Current Situation:

  QMUL currently does not publish AAAA records for the affected nodes.

- Proposed Workaround:

  Implementation of DNS blacklisting on our DNS servers to ensure that CERN are **not** served with AAAA records for affected hosts.

- Correct Solution:

  CERN need to either remove IPv6 defaultroute from affected pilot factories **or** add an IPv6 global unicast address to the interfaces on the affected pilot factories.

Next steps

- Complete move
  - IPV6 capable squids, argus, bdii, xrootd.
- Get IPV6 addresses on compute nodes (inside nat).
  - Replace nat gateways with routers/firewalls?
- Run our own authoritative DNS server?
- Each compute job with own container and IPV6 address (contain and traceability)?