IPv6 testing and deployment at Prague Tier 2

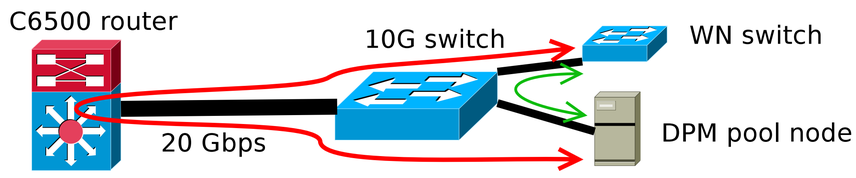
Tomáš Kouba, Jiří Chudoba, Marek Eliáš, Lukáš Fiala
Institute of Physics, Czech Academy of Sciences, Prague, Czech Republic

Introduction

The Tier2 Computing Center in Prague runs about 300 worker nodes and 100 storage and service nodes. Our site supports four big HEP experiments: ATLAS, ALICE, D0 and Pierre Auger Observatory. We have already exhausted our C-class IP range and we have to use NAT for several nodes which is not optimal from the performance point of view.

Motivation for transition to IPv6

With the lack of IPv4 addresses we are forced to place our worker nodes in the private address space. The DPM nodes have to be in public address space. Some SRM commands request the endpoint to always answer from the same IP address so we cannot use multihoming setup on our DPM nodes. This means that traffic between WNs and DPM have to be routed which results in performance penalties. IPv6 deployment would solve this problem. Our institute got assigned 256 IPv6 /64 subnets.



Our workflow in IPv4

When the computing center obtains new piece of hardware the path to production usually looks like this:

1. Obtain basic HW info and put it in our HW database application
2. Assign system and/or management IP address
3. Manually configure management interface (if present)
4. Install operating system via PXE + RHEL kickstart
5. Set the role of the new system in fabric management (CFengine2 and puppet in the future)
6. Run CFengine and check the result configuration
7. Add system to monitoring tools (nagios, munin, mrtg, ganglia)

Changes in workflow for IPv6

1. Adapt HW DB application for IPv6 addresses
2. Design the politics for IPv6 address assigning
3. Try to configure management interfaces with IPv6 addresses
4. Test PXE + kickstart installation over IPv6
5. Convert the CFengine configuration to Puppet (not exactly IPv6-related)
6. Test puppet over IPv6
7. Install monitoring tools with IPv6 support

All the workflow changes were tested on our IPv6 testbed. (see next column)

IPv6 related hardware problems

Firewall

- Does not support filtering IPv6 in transparent mode
- We had to switch to routing mode (carefully and without downtime)

Network and FC switches

- All switches pass IPv6 packets without problems
- Only 2 switch types (out of 8) support management over IPv6
- FC switches (IBM SAN16b, SAN24B) can have IPv6 address manually assigned but do not support SLAAC nor DHCPv6

Computing center facility

- None of the special hardware has IPv6 support
  - UPS, thermometer, air condition, water cooling, KVM on the net
- None of the disk array web managements have IPv6 support
  - NEXSAN Satabeast, Ultamus 1200, Ultamus 4800, Areca ARC-1680
  - Neo 8000 (tape robot)

Server management (BMC)

- The only server that offerred management accessible via IPv6 was IBM System x3550 M4 (borrowed for testing)
- Our production servers do not support management over IPv6

Currently this means that special HW is much less prepared for IPv6 than operating systems so we have to use private IPv4 network for our monitoring purposes.

Network services

DNS

- Running BIND 9.7.3 as a primary NS for ipv6 domain (ipv6-only node)
- Running BIND 9.3.6 as a secondary NS for ipv6.farm.particle.cz (dual stack node).
- We need dual stack because many IPv6 enabled services are in fact unresolvable using only IPv6 communication). Usually because of missing GLUE records for IPv6 addresses.

DHCP

- We use stateful DHCP to distribute fixed IP addresses, DNS servers, NTP servers
- Routing information still comes from RA
- We use DUID-LL formed as static prefix (00:03:00:01) plus MAC address - similar approach to IPv4 production setup

NTP server

- No special setup needed
- It is not easy to find IPv6 enabled server with low stratum
- Our server runs in dual stack mode and synchronizes with CESNET's stratum 1 NTP server

Automatic installation

- PXE over IPv6 brings many problems
  - Poor support from HW vendors
  - Poor implementation in DHCP servers (ISC DHCP and Dibbler tested)
  - Open source gPXE works but freezes most of the time
- Anaconda and kickstart
  - Big difference between SL5 and SL6
  - DHCPv6 client in Anaconda does not support custom DUIDs
  - Anaconda has problems with timeouts (cannot download from IPv6 repository, timeout, second attempt works)
  - Anaconda in SL5 does not support DNS server (option `--nameserver`) specified in kickstart
  - Kickstart does not support using proxy server during package installation
- We set up unrouted IPv4 private network only for installation purposes. This network is disabled on the node when the installation is finished.
- Packages are downloaded either via proxy (see the picture) or directly from IPv6 enabled SL mirror.
- Automatic configuration - Puppet
  - Apart from IPv6 testing we use our testbed to test Puppet for fabric management
  - Puppet server communicates over IPv6 and client info provider (facter) collects IPv6 specific data
  - We did not test any Puppet-compatible dashboard

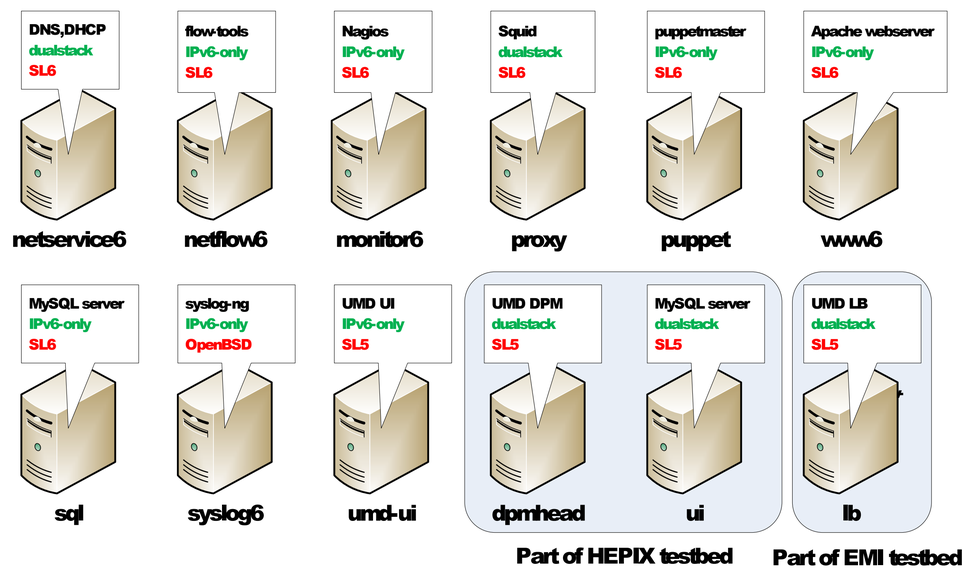
EMI middleware

Prerequisites

- CRLs - not all CAs in EUGridPMA offer CRLs over IPv6 ⇒ application to monitor status
- YAIM depends on working `hostname -f` which is broken on IPv6-only nodes ⇒ wrapper script
- Supported OS - We were not able to run DHCPv6 on SL5 successfully ⇒ static network configuration

CA Subject CA Crl url	IPv6 DNS	IPv6 GET
CN=AAA Certificate Services, O=Comodo CA Limited, L=Salford, S=Greater Manchester, C=GB		
http://crl.comodoca.com/AAACertificateServices.crl	YES	YES
CN=CERN Trusted Certification Authority, DC=cern, DC=ch		
http://ca.cern.ch/ca/CRL/CERN Trusted Certification Authority.crl	NO	
CN=CESNET CA 3, O=CESNET CA, DC=cesnet-ca, DC=cz		
http://crl.cesnet-ca.cz/CESNET_CA_3.crl	YES	YES
CN=CESNET CA Root, O=CESNET CA, DC=cesnet-ca, DC=cz		
http://crl.cesnet-ca.cz/CESNET_CA_Root.crl	YES	YES
CN=CESNET CA, DC=cesnet-ca, DC=cz		
http://www.cesnet.cz/pki/crl/cn=CESNET CA,dc=cesnet-ca,dc=cz.crl	YES	YES
CN=CILogon Silver CA 1, O=CILogon, C=US, DC=cilogon, DC=org		
http://crl.cilogon.org/cilogon-silver.crl http://crl.dosgrids.org/cilogon-silver.crl	NO	
CN=Root Certificate Authority at CNIC, DC=Grid, DC=CN		
http://ca.grid.cn/crl/cacrl.pem	NO	
CN=ESnet Root CA 1, OU=Certificate Authorities, O=ESnet, DC=ES, DC=net		
http://es.net/CA/d1b603c3/CRL/d1b603c3.r0	YES	ERR
CN=Kerberized CA HSM, OU=Certificate Authorities, O=Fermilab, DC=fnal, DC=gov		
http://security.fnal.gov/pki/99f9f5a3.r0	NO	

Schema of our testbed



Acknowledgement

This work is partially suppported by CESNET, project number 416R1/2011.