IPv6 in Condor

MinJae Hwang

Condor Project
Computer Sciences Department
University of Wisconsin-Madison
Overview

1. IPv6?
2. What are required to run Condor in IPv6?
3. Issues in Porting Condor to IPv6
1. **IPv6**?

```bash
[m@submit ~]$ /sbin/ifconfig
eth0 Link encap:Ethernet  HWaddr 00:21:9B:8A:A5:2F
inet addr:128.104.55.9  Bcast:128.104.55.255  Mask:255.255.255.0
inet6 addr: 2607:f388:1086:0:221:9bff:fe8a:a52f/64 Scope:Global
inet6 addr: fe80::221:9bff:fe8a:a52f/64 Scope:Link
```

![IPv4 Assignments based on January 2010 IANA data](image)

---

**Figure 1:** IPv4 Assignments based on January 2010 IANA data

---

[www.cs.wisc.edu/Condor](www.cs.wisc.edu/Condor)
What is IPv6?

- Internet Protocol 6 by IETF
- Simply, extension of address space

You can even assign IP address to pebbles

$$6.67 \cdot 10^{27}$$ IPv6 addresses/m² on earth
Why IPv6?

- IPv4 addresses are expected to fully assigned by the end of 2011
- Large scale grid deployment especially non-US countries will suffer from IP address shortage
2. What are required to use Condor in IPv6?
Requirement for IPv6

› Every software and hardware stack should be rebuilt and tested
  • IPv6 has no ‘protocol-level’ compatibility to IPv4

› Most of current operating systems, switches, routers fully support IPv6
  • (Almost) Every OS that Condor supports provides IPv6
Condor Requirement for IPv6

- Most of external libraries that Condor use are IPv6 supported
  - Exception: Storage Resource Broker from SDSC (for Stork)
3. Porting Condor to IPv6
Issues in Porting Applications to IPv6

› Using Old BSD Socket Interface
  • gethostbyname(), inet_addr(), ...
› Address Parsing/Printing
  • printf("%u %u %u %u", addr >> 24, ...)
› Storing Address in Integers
  • unsigned int IP = ...
› IP address binding
  • 127.0.0.1, 255.255.255.255.0
› Buffer Allocation/Data Structure
  • char IPADDR[16] <- FALSE! IP address is now 46 letters long!
Issues in Porting Condor to IPv6

› Large Codebase
  • 680,000 LOC

› Scattered Source Code
  • Every daemon has networking code

› Implicit Use of IP Address
  • Non-networking code handles IP address
  • LOG( "%s: error ..", local_ip_buffer ),
    • local_ip_buffer may not have enough buffer

› Various OSes, various architectures
  • Condor supports more than 20+ OS, 6+ Architectures
Goals for deployment

- Backward-compatibility is our top priority
- Supporting IPv6 in incremental way
  - Old system continues to use IPv4-only binary
  - New system can use IPv6-enabled binary
Goals for users

- **User**: does not need to know whether it is IPv4 or IPv6
- **Admin**: minimal change to config file

<table>
<thead>
<tr>
<th></th>
<th>IPv4</th>
<th>IPv6</th>
</tr>
</thead>
</table>
My experiences

› Initially, tried to find a ‘automatic’ conversion
  • A set of classes that deals with networking and IP address
  • Use compiler tricks to detect ‘incompatible’ spots

› No automatic way
  • Implicit use of IP prevents automatic detection
  • No incremental progress
  • Hard to ensure backward-compatibility
Current Development Status

› Work on small milestone and do extensive test on each milestone
› Ensuring compatibility across OS, arch is still a problem
  • More than 20 operating systems, 6 architectures
  • Subtle difference in Socket API on each OS
› How to deal with heterogeneous network?
  • IPv4 clients, IPv6 servers?
  • IPv6 clients, IPv4 servers?
  • IPv4-to-IPv6 tunnel? Dual-stack?
Thank you

- Special thanks to Beihang Univ. in P. R. China
  - for providing IPv6 test-bed
IPv4-to-IPv6 convertor

Cheap-and-easy! IPv4-to-IPv6 hardware convertor (possibly using tunneling) by silex technology