

A Unified IPv6 Scheme

Open IPv6

China Telecom : Qiong Sun, Chongfeng Xie

Telefonica I+D : Diego Lopez

Tsinghua University: Jun Bi, Guang Yao

Huawei : Cathy Zhou, Will Liu, Tina Tsou

Viagénie : Guillaume Leclanche

IETF 89 May 2-7, 2014. London, UK

Will Liu liushcuheng@huawei.com

Agenda

- Background & Motivation
- Use Cases and Problem Space
- Architecture
- Work to Be Possibly Done in IETF
- Openv6 Side Meeting @IETF 89

Background & Motivation

- IPv6 Current Status
 - Lack of motivation and high cost for carriers / service providers / end users to transition to IPv6
- Design a **low-cost, unified** approach to IPv6
 - Enable DevOps for IPv6
 - Extensibility via programmability
 - Implementation, deployment and operation
 - Streamline the OSS
 - Simplify the IPv6 transition, reduce deployment and operational costs
 - Unified mechanism and data plane device
 - Decoupling the data plane and control plane
 - Reduce the operational complexity and risk
 - Unified data plane adapt to different transition scenarios
 - Nurture native IPv6 services
 - Via the open Interface of the openv6 devices

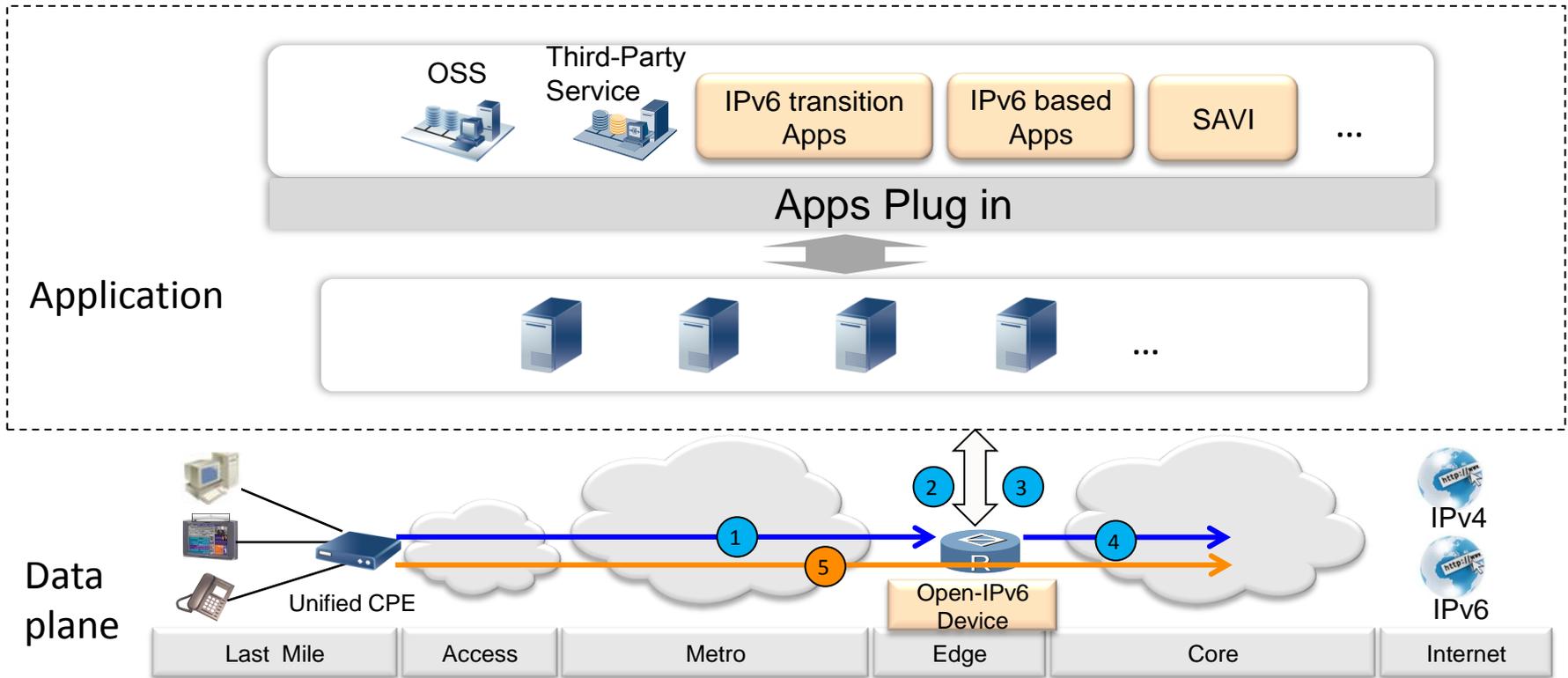
Use Case and Problem Space: IPv6 transition

- Problems:
 - Various transition technologies co-exist
 - How to manipulate Transition Data Plane with different modes?
 - How to identify the capabilities of different transition devices ?
- China Telecom requirements :
 - DS-Lite /CGN for *Broadband Customers*
 - CGN/NAT64 for *Mobile Customers*
 - DualStack /NAT64 / NAT46 for *Data Centers*
- Openv6 Benefits:
 - V6 transition as an APP
 - Identify different users for different scenarios
 - Have a centralized management of address pools

Use Case and Problem Space: SAVI

- Problems:
 - The difficulty of designing and implementing a all-in-one SAVI solution for all the scenarios.
 - The complexity of rule generation: various possible errors
- Openv6 benefits:
 - SAVI as an APP
 - Dynamically enable the SAVI solution specified for the deployment scenario
 - SAVI-DHCP, SAVI-FCFS, SAVI-SEND, SAVI-MIX...
 - Offloading the complexity of rule generation to controller

Openv6 Architecture



First packet of a subscriber (or a flow)



Subsequent packets

- Two layers: Application and Data Plane (Openv6 device)*
- The configuration protocols between the application and the openv6 devices are open, netconf/ restconf?

*reference: draft-liu-openv6-architecture-00

Work to Be Possibly Done in IETF

- Immediate protocol action
 - Multiple applications' concurrent message process
 - Multi-APP control, process multiple APP concurrent messages to avoid disorder, see RFC3588
 - Peer node messages
 - The openv6 device and applications should support both client-initiated and server-initiated messages. RFC3588 or to extend netconf/restconf.
 - Policy distribution:
 - The applications distribute the policy (table) to the openv6 device to make the APP have the network resource. Yang model [RFC6020] may be needed to provide the configuration information, [draft-zhou-netmod-openv6-transition-cfg-00]
 - Security/authorization mechanism Bidirectional authentication
 - Different applications should have different authority to operate the network resources
 - Bidirectional authentication between APP and OPENV6 device to guarantee security, new Diameter [RFC3588] AVPs
- Additional work
 - Packet -in events
 - The APP needs to analyze specific packet to configure the flow, which requires protocols (e.g., netconf/restconf extension) to support the packet report
 - High-Throughput for bidirectional messages
 - Service triggered packet from the openv6 device will bring additional configuration from the applications, which requires more high throughput than the existing protocols, e.g., netconf

Openv6 side meeting@IETF89

Openv6 mailing list: openv6@ietf.org

Openv6 Side Meeting *

Date: March 5, 2013 (Wednesday)

Time: 18:40~19:40

Location: Victoria (East Wing, Mezzanine Meeting Rooms), behind the Park Suite
Hilton London Metropole

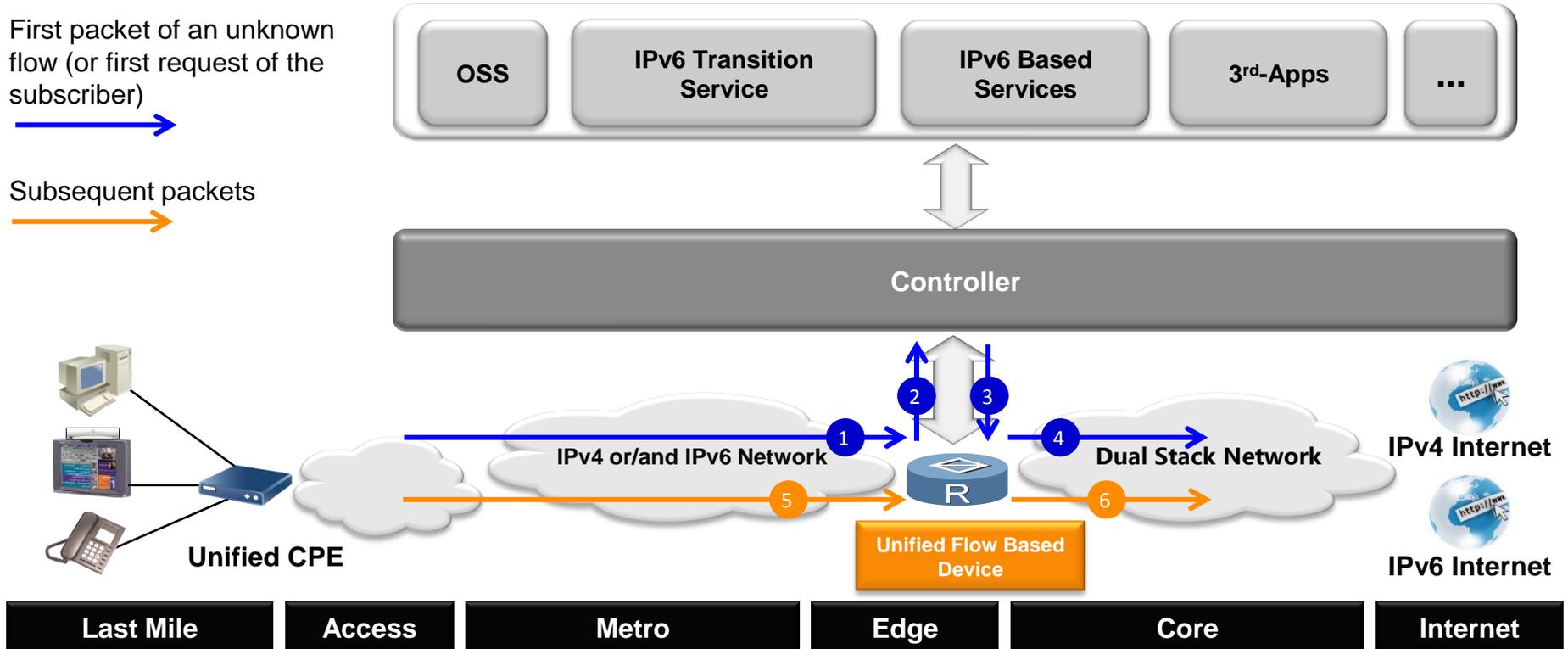
Please Join Us!

* Info about Openv6 Side Meeting can be found in the mailing list

Questions?

- Backup pages

Openv6 Architecture (2)



- Controller and unified flow based device are separated
 - Decoupling the data plane and control plane
 - Openflow/Diameter/I2rs/Netconf may be candidate protocols between controller and the devices
 - Providing Northbound Interface for services manipulate the traffics, Netconf/Restconf/CLI may be candidate protocols.