A Novel Centralized Control Implementation For Identity-Location-Separated Network Architecture

Chaoqi Yu, Hui Li, Huiling Liu, Kai Pan, Weiyang Liu

***Abstract*—We describe a generic polymorphic networking architecture with identity-based routing mechanism and centralized control fashion called Centralized Identity-based Network (CIN) that combine the ideas of separating control plane and forwarding plane and separating identity and location. In this networking architecture, host identity can be a globally unique name of terminal is a key idea to support mobility. When the terminal moves, database updated the identity and location mapping system in real time. Controllers issue the flow-tables dynamically according the analysis of mapping system. Switches' routing policy has been changed, which keeps the communication connected. At last, we introduce CIN based on Open vSwitch OpenFlow switches and RYU controllers. The result of experiment shows CIN is a scalable networking architecture.**

***Index Terms*—Centralized control, Software Defined Networking, Identity, Location, OpenFlow**

I. Introduction

TRADITIONAL network architecture is based on location dependent IP address. The hierarchical address space enables the use of longest prefix matching technique to perform route aggregation, thus reduces to the size of routing table and scales the Internet to the whole world. However, it brings difficulty in supporting mobility. Nowadays, mobility supported is through an address resolution scheme derived from RFC3344 [1]. The protocol provides for registering the care-of address with a home agent. The home agent sends datagrams destined for the mobile node through a tunnel to the care-of address. After arriving at the end of the tunnel, each datagram is then delivered to the mobile node. Although the method provides with the mobility feature, it also introduces many problems like triangular communication and poor QoS guaranteed.

An identifier is used on end-systems to identify a connection endpoint, while a locator refers to the attachment point in the Internet topology [2]. The proposals can be classified in two main categories: those associating locators directly to end-systems (e.g., HIP [3], SHIM6 [4]). And those associating locators to routers (e.g., LISP [5], MobilityFirst [6]). LISP splits the IP addressing space into two sub-spaces where addresses have one clear single semantic. MobilityFirst is a clean-slate project being conducted as part of the NSF Future Internet Architecture (FIA) program. The protocol also clean separation between identity and network location. It is intended to directly address the challenges ofwireless access and mobility at scale.

Network architectures such as Software Defined Networking (SDN) in which the control plane is decoupled from the data plane have been growing in popularity. The SDN controller typically has knowledge about the physical topology of the network either by discovery mechanisms or appropriate databases and can based upon this topology create paths that are programmed into the forwarding engines of network devices. OpenFlow[7] is considered as the enabler of SDN. It is a standard communications interface defined between the control and forwarding layers of an SDN architecture. OpenFlow allows direct access to and manipulation of the forwarding plane of network devices such as switches and routers, both physical and virtual (hypervisor-based). The path of packets through the network of OpenFlow enabled switches is determined by software running on a separate OpenFlow controller.

In this paper, we propose an identity based routing based on centralized network architecture and use the OpenFlow [7] network as the implementation. Compared with previous work, we combine the benefits of separation between identity and location, and flexibility of the SDN architecture. The key components of the CIN architecture are:

1) Separation of host identity and location address, implemented via a fast dynamic mapping system resolution service.

2) Self-certifying public key host identity to support strong authentication and security.

3) Hierarchical identity and location mapping system and resolution mechanism improve the efficiency and accuracy of analytical packets.

4) A separate network control plane that provides topology discovery and enhanced traffic management.

5) The centralized management of control plane and simplified function of data plane facilitate the integration of other heterogeneous networks.

Manuscript received January 8, 2015. This work is supported by National Basic Research Program of China (973 Program, No.2012CB315904), National Natural Science Foundation of China (No.NSFC61179028), Guangdong Natural Science Foundation (GDNSF, No.S2013020012822), Shenzhen Basic Research (No.JCYJ20140417144423192), and Shenzhen Basic Research (No.JCYJ20130331144502026).

Chaoqi Yu, Hui Li, Huiling Liu, Kai Pan, Weiyang Liu are with the Shenzhen Engineering Lab of Converged Networks Technology, Shenzhen Eng. Lab of Converged Networks Technology, Inst.of BigData Technology, Shenzhen Graduate School, Peking University, Shenzhen, Guangdong, 518055, China (email: yuchaoqi@foxmail.com, lih64@pkusz.edu.cn, liuhuilingstar@gmail.com, pankai@pku.edu.cn, wyliu@pku.edu.cn)

II. The Architecture of CIN

The remainder of the paper is organized as follows. In Section II, we introduce the architecture of Centralized Identity-based Network (CIN). Section III presentation the communication mechanism of CIN architecture. Finally, we describes how our Identity based routing is achieved on OpenFlow in Section IV and make a conclusion in Section V.

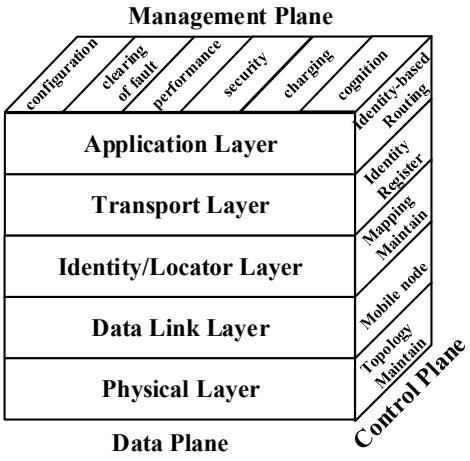

Fig. 1. Reference model of CIN.

A. Data Plane

The Centralized Identity-based Network (CIN) separates the identification and localization roles of IP ad- dresses by introducing two logical addressing spaces:

1) the Host IDentifier (HID) is a 128-bit (for IPv6) value used in the source and destination address fields of the first or last CIN header of a packet;

2) the Location IDentifier (LID) is an IPv6 address of an Edge Router (ER).

In data plane, the difference from TCP/IP stack is changing the network layer to Identity/Locator Layer in Fig.1. CIN provides this level of indirection through a particular routing mechanism over the core network, as shown in Fig.2. More specifically, any communicating host generates regular IP packets using HID (Host IDentifier) as source address and destination address. Forwarding towards the border router in the local domain (see the router which the number is ① in the left edge network of Fig.2). The border router, now called Edge Network Router (ENR), will rewrite the packets using the LID (Location IDentifier) addressing space, i.e., using its LID address as source address and the destination LID as destination address replacing the header of the original packet. The rewritten packets can now be forwarded over the core network (see the dashed line in Fig.2). The border router at the destination site will recovery the LID packets so that the original packet can be forwarded to its final HID destination (see the solid line in the right edge network of Fig.2).

B. Control Plane

In control plane, in order to perform the data plane operations, mainly related to rewrite and recovery as described in the previous section, controllers need to be able to associate HID to LID and vice versa. Control platform of CIN has five major functions.

1) Topology Discovery and Maintain

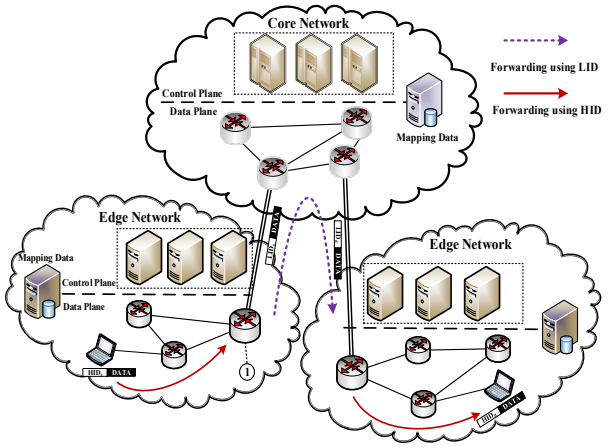

Fig. 2. The packets forwarding sequence of CIN architecture

To build the paths between each pair of hosts, global topology graph need to be maintained in the control plane. We have built applications that use DNS, DHCP, LLDP, and flow-initiations [8] to construct the network view, including both the network topology and the set of name-address bindings. When the topology changes, the controllers achieve the exchange of network topology detection and maintenance by periodical topology discovery methods.

2) Identity Register

When the host access to the network at the first time, flow-initiation would be forwarded to control platform through the OpenFlow switch. As we know the topology of network, MAC address and HID address of the host and the dpid of the OpenFlow switch which transfer the flow-initiation. The packet with identity information would be sent to the host ID register database to verify the legitimacy of HID. At the same time, the controller will send the information to the mapping system and establish a new HID-to-LID mapping entry in the database.

3) Mobile Management

The purpose of mobile management is locating the mobile nodes. Host tracking can be realized in various ways, for example, explicit host registration or ARP intercept. In the OpenFlow network, every switch has a unique ID named dpid. When the host connects to a new switch, the packet (contain the information of HID and dpid) must be forwarded to controller from the switch. The controller also needs to synchronize the location map with other controllers to maintain a global host map periodically. Meanwhile, the mapping database would be updated according to the information of dpid and HID from the control plane.

4) Mapping Maintain

Each edge network (Autonomous System) has a mapping system consists of several LID-to-HID mapping databases. The controllers of edge networks detect the altering mapping database effected by mobility nodes. Packets will be sent to the core network controllers across the border routers of edge networks which contain the altered mapping information. In the core network mapping database stores the mappings for HID-prefixes to LID and maintains the global mapping center. Issued the interrelated mapping information to living the edge network mapping database update, in the core network control plane.

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

5) Identity based routing

In order to perform the data plane operations, mainly related to rewrite and recovery addresses as described in the previous section, controllers and border routers need to be able to associate HIDs to LIDs. The function of rewriting and recovering addresses in the control plane and enforced in the data plane is the key to identity based routing. When the packets need to forwarding across the core network or other border routers. The controllers issue the flow-tables to change the header of packets from the terminals according to the HID-to-LID mapping database. At the destination site, the router will recovery the rewritten packets' header to HIDs from LIDs according the flow-tables. We will introduce the communication combining with mapping database detailed in the next section.

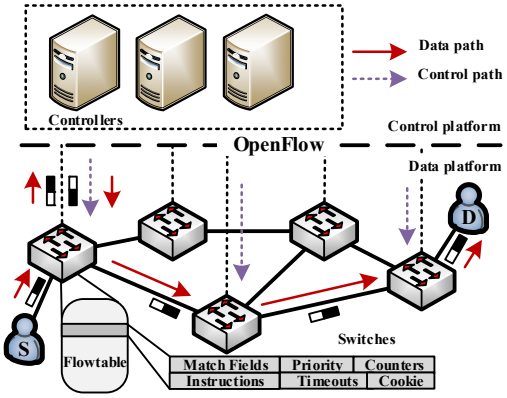

Fig.3. Illustration of Control plane and Data plane

As shown in Fig.3, Controllers instruct the switches as to what actions they should take via what is commonly called the southbound API. OpenFlow is rapidly becoming the dominant way for an SDN controller to communicate with switches. Every switch has a flowtable used to perform packet matching. When a packet matches an entry in the flowtable, associated actions are executed. If table mismatch occurs, packet is send to a logical centralized controller (like NOX [8][9], POX [10], RYU [11]) by OpenFlow protocol through a security channel. Since the controller has a global view of the whole network, when it receives a packet from the data plane, it could react appropriately in any software defined way. Flowtable is the most important component of an OpenFlow switch, and it evolves with the OpenFlow specification. An entry in the flowtable has six fields showed in the Fig.3. A controller adds and removes flow-entries from the flowtable to change the packets forwarding.

III. Communication Mechanism of CIN

Mapping system is one of the most important part of CIN has been mentioned before but hasn't expound yet. The mapping system in the edge network is constituted by Local ENMD and Visitor ENMD. The Local ENMD (Edge Network Mapping Database) stores the mappings for HIDs of local host for which the border routers' LIDs. In another aspect, the Visitor ENMD only keeps the mapping s for visitors' HIDs with LIDs. In particular, a mapping consists of an HID associated with a list of dpid (access router) , dpid (border router) , LID tuples. The CNMD (Core Network Mapping Database) stores the mappings of HID-prefixs with LIDs in the core network.

In the Fig.4, We show the process of communication between $host_s$ and $host_d$

1) The first packet has been forwarded from the hostsusing HID_S as source address and using HID_D as destination address. This packet does not match a flow entry and has been forwarded to a controller process as flow-initiation.

2) Controller process sends a map-request packet to the current ENMD depend on the source address of flow-initiation. In the example of Fig.4, The destination host is in the other edge network. Controller process forwarding a search request packet to core network.

3) The core network performs the <HID-prefix, dpid (border router), LID> lookup in its mapping database. The purpose is to know the LID of ENR in the destination edge network. The CNC issues the map-request to the border router according the information from the CNMD.

4) When forwarding to the destination network, the map-request packet will be sent to ENMD through the controller. The result will be return to the core network.

5) Core Network issues the mapping information to the source and destination edge network. The controllers in both edge network store the mapping information as Mapping Cache. Meanwhile, controllers issue the different flow-tables to the ENRs.

6) In the source edge network, the header of packets have been replaced HID addresses to LID addressing space as source and destination addresses. Antithetically, the packets will be recovery as original packets in the ENRs of destination edge network.

At last, the packets have been forwarded to $host_d$. The communication established between $host_s$ and $host_d$.

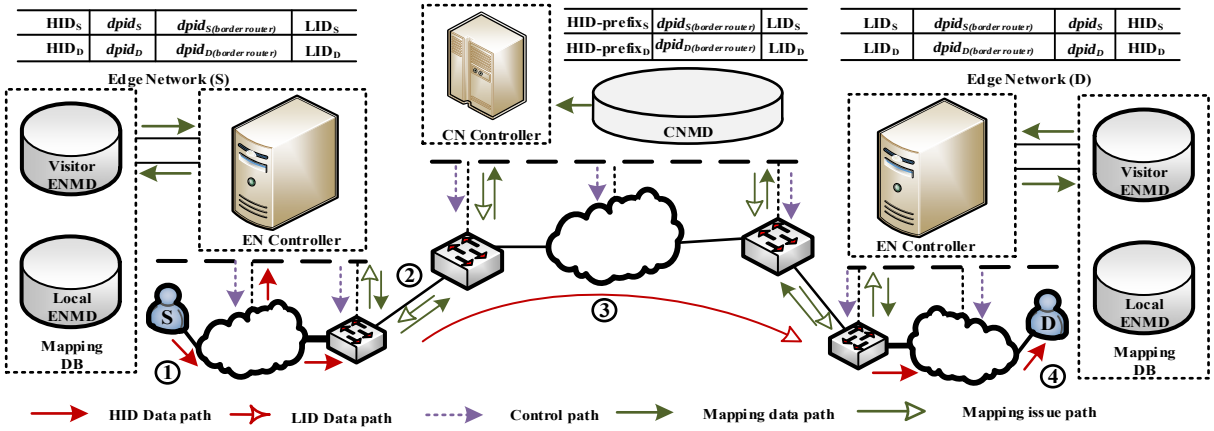

Fig.4. Illustration of communication mechanism of CIN

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

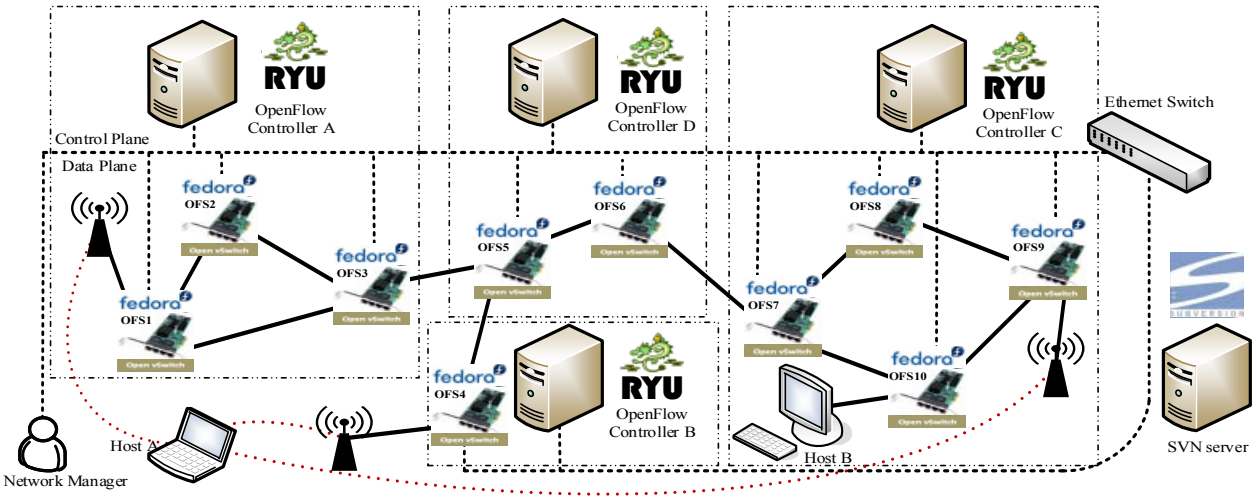

Fig.5. Illustration of Control plane and Data plane

IV. The Experiment and Analysis of CIN

In Fig.5, we build a prototype of CIN, where each OpenFlow switch is a Linux PC running Fedora14 Linux operating system deployed a virtual switch named Open vSwitch with a four ports Gigabit network interface card plugged in. There are total ten OpenFlow switches, three wireless routers and several hosts PC in the network. Figure.4 only draws two hosts. Host_B connected OFS10 which controlled by OpenFlow Controller C. Host_A connected wireless router.

The controller we used in the test named RYU, which is written in Python. Compared with other open-source controllers, RYU's Python interface provides with easy user module integration and rapid prototyping. Due to Python's interpreting nature, its performance is much lower than others. In this experiment, we only care about the feasibility of our method and pay less attention to the performance. We use 4 controllers to manage several switches showed by dashed boxes in the test bed. Deserve to be mentioned that we use the OpenFlow 1.3 version[12] as southbound API to support IPv6 header rewrite in the experiment.

The first test is carried out by make each host ping all other hosts in the network. The first packet goes through a delay about 50ms, which is due to flow table setup and subsequent packet pass the network without any significant delay. The second test is about maintain the mobile hosts' communication continually. The host_A is playing a movie smoothly coming from host_B which connected to OFS10. When host_A moves to connect other wireless router in the test bed, the communication between the host_A and B continually, and the movie after a short pause resumes again.

Besides a more scalable Internet architecture, CIN also provides an elegant solution to several networking problems not directly related to Internet scalability, representing important benefits for early adopters. The architecture makes networking functions available as programmable resource, via a logically centralized controller, which manages and operates a network (switches) from a global view of the network. Meanwhile, Decoupling identity with location information increased the flexible of data transmission and security. CIN offers new perspectives for known scenarios.

As previously described, the packets has been changed by rewritten. Consider that the Ipv4 is the major routing addressing in legacy network. The identity transformation mechanism will use the LID addresses as source and destination addresses in the tunnel header encapsulating the original packets. The map-and-encap of CIN makes it a flexible tool and a perfect candidate for supporting the transition to IPv6. Having two ways to process packets supports any combination of locator and identifier address families. It is the possible to bind IPv6 HIDs with IPv4 LIDs and vice versa. This allows for transporting IPv6 packets over an IPv4 network (or IPv4 packets over an IPv6 network), thus enabling the use of CIN as an IPv6 transition mechanism.

V. Conclusion

In this paper, we propose an identity-location-separated network architecture that is based on OpenFlow network. Preliminary prototype, though in a small scale, proves CIN is feasible under realistic environment.

References

[1] C. Perkins et al., "Rfc 3344: Ip mobility support for ipv4," Network Working Group, 2002.
[2] T. Li, "Recommendation for a routing architecture," 2011.
[3] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host identity protocol," RFC5201, April, 2008.
[4] E. Nordmark and M. Bagnulo, "Shim6: Level 3 multihoming shim protocol for ipv6," RFC 5533, June, Tech. Rep., 2009
[5] D. Farinacci, D. Lewis, D. Meyer, and V. Fuller, "The locator/id separation protocol (lisp)," 2013.
[6] Seskar, K. Nagaraja, S. Nelson, and D. Raychaudhuri, "Mobilityfirst future internet architecture project," in Proceedings of the 7th Asian Internet Engineering Conference. ACM, 2011, pp. 1–3.
[7] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 69–74, 2008.
[8] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker, "Nox: towards an operating system for networks," ACM SIGCOMM Computer Communication Review, vol. 38, no. 3, pp. 105–110, 2008.
[9] M. Fernandez, "Evaluating openflow controller paradigms," in ICN 2013, The Twelfth International Conference on Networks, 2013, pp. 151–157
[10] About POX in NOXRepo, http://www.noxrepo.org/pox/about-pox/
[11] Ryu SDN Framework, http://osrg.github.io/ryu/
[12] O. S. Consortium et al., "Openflow switch specification version 1.3.4," 2014.

ISBN: 978-988-19253-9-8
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)