ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 147

International Conference on Network, Communication, Computer Engineering (NCCE 2018)

A Two-characters Experiment Design for IPv6 Routing&Forwarding Based on Modularization

Xin Yu, Bingliang Xu, Sheng Cai, Zhiyang Jin, Yanping Dong, Ming Gao [a)]

Zhejiang Gongshang University, Hangzhou, 310018, China

[a)] Corresponding Author: gaoming@mail.zjgsu.edu.cn

Abstract. This paper brings forward an experiment design for IPv6 routing&forwarding which based on the special IETF ForCES technology, with the goal of more flexible using of IP routing&switch theory, better acquainted about inner architecture of network device, more familiar with the new technology of network industry for those students with the major of network engineering. First, this paper analyzes the IETF ForCES and its character, then a method of IPv6 routing&forwarding based on ForCES is proposed. The mechanism of IPv6 forwarding and modeling are included within the method above, and the two-character is great showing as well. Finally, we have an evaluation over the experiment design, the result turns out to prove the applicable design.

Key words: Routing&Forwarding, Two-characters experiment, IETF ForCES, Modularization.

BACKGROUND

Although IPv4 is the basic form of the current network, IPv6 is considered to be the foundation of the next generation Internet. Routing&Forwarding is the most important function of routers in an IPv6 network. It solves the problem of how IPv6 packets reach the destination host from the original host. At present, the practical teaching for IPv6 is divided into two levels [1] [2]. Most schools use the method of networking engineering to construct a small scale IPv6 data transmission network based on the existing IPv6 router products. Focus on the students' equipment operation and debugging ability, basically do not need to program, so it is not very difficult, but the insufficiency is that the “Two-characters” of the experiment cannot be reflected; For some schools with strong student abilities, they will use methods based on quadratic code development to allow students to write code to implement route lookups for IPv6 packets based on existing network function libraries. The method generally does not emphasize the concept of IPv6 networking. More attention is paid to the processing of IPv6 data packets within a single router. Focusing on students' basic theoretical knowledge and programming skills is difficult and not suitable for general promotion. In order to change the "ossification" of traditional IPv4 network, many researchers have tried to redesign the network architecture in recent years, and a batch of novel network design has sprung up, such as Active Network, Open Programmable Network, Flexible Reconfigurable Network, Software Defined Network, etc. [3]. As a student of network engineering, it is undoubtedly necessary to understand these new developments in the field of network. After a comprehensive review of various new network designs, some interesting research results have been found that can be used in normal network teaching. Among them, ForCES(Forwarding and Control Element Separation) is a typical example. ForCES, of IETF (Internet Engineering Task Force), is a working group dedicated to the research of open programmable IP routers, and creatively proposes that the CE (Control Element) and FE(Forwarding Element) in network devices should be physically separated. The communication protocols between each other should be standardized and made public [4] so that the CE and FE of any manufacturer can connect seamlessly and reduce the threshold and cost of research and development of network equipment. CE, which is traditionally the "brain" of network equipment, controls the "trunk" FE and cannot be said to be "confidential". It is not open to the outside world. In addition, ForCES also creatively decomposes FE into a series of Logical Function Blocks (LFB) combinations.

Each LFB is an action on packet processing within a network device, and the behavior of each LFB is controlled by CE [5]. The CE can flexibly configure the LFB combination method as needed to implement different processing flow for data packets, and then it can be embodied as a network device with different functions, such as IPv4 route forwarding, IPv6 route forwarding, firewall, intrusion detection, etc. The idea of ForCES is forward looking, especially its concept of LFB dynamic topology [5] and the current research hotspot micro service [6] and service function chain (SFC7) are transmitted from one pulse to the other, which is of great significance for learning the internal composition and principle of network devices.

The process of constructing network devices with different functions using LFB dynamic topology is similar to building blocks. Each LFB is like a building block. The builder needs to understand the functions, attributes, and input and output constraints of each LFB. Completing the entire construction process does not require a large amount of code but requires the operator to have a solid theoretical knowledge of route exchange and is familiar with the internal composition and principles of the network equipment. It is interesting, not boring and available. It is suitable for the development of "Two-characters" experiments in the field of network communication.

FE MODELING

The FE model was proposed by the IETF ForCES working group to describe the processing of data packets in the FE from input to output. Each individual packet processing function was modeled as an LFB, and the LFB processed the data packets according to the previously agreed operations. A complete LFB topology can implement network services such as IPv4 forwarding. The CE modifies the behavior of the FE by controlling the LFB, such as changing the attributes of each LFB within the FE. The typical FE model is shown in Figure 1.

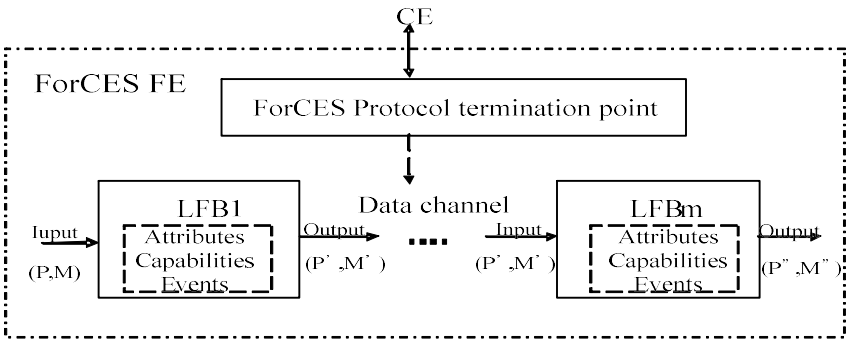

FIGURE 1. FE model

FE consists of several LFBs. Each LFB contains input, output, attributes, and its function definitions [5]. The data channel describes the middle process of the data packet from input to output. The output of the previous LFB is the input of the next LFB. The context between LFBs is constrained by the definition of the FE model. This constraint is shared between the CE and the FE in the form of an LFB library. The user needs to grasp this information before configuration.

Each common network service (such as IPv6 route forwarding) can be defined as a set of LFB collections and topologies. The key to implementing the ForCES-based IPv6 routing and forwarding service shown in Figure 2 lies in analyzing and designing the IPv6-related LFBs and constructing a FE model with a reasonable topology.

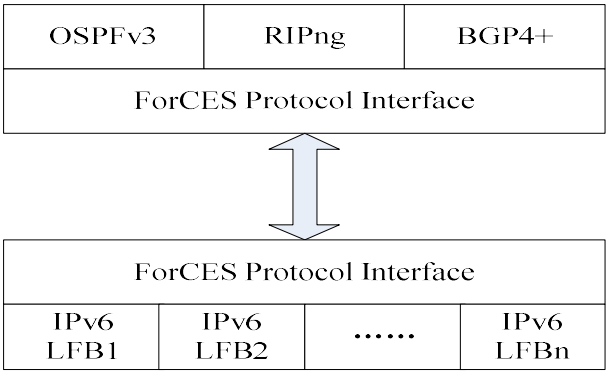

FIGURE 2. FE model supporting For IPv6 based on ForCES

LFB-BASED IPV6 ROUTING AND FORWARDING DESIGN

IPv6 packet forwarding consists of the following three phases [8]:

Verification phase: At this stage, firstly, the IPv6 packet header check that the packet does not have any identification error, and the wrong packet is sent to the CE for further processing or is deleted by the FE.

Forwarding phase: From the purpose of extracting the destination IPv6 address, query the forwarding table. After the query operation is successful, the next hop IPv6 address is obtained and the interface is output. When the query operation fails, the data packet is discarded or sent to the CE for further processing. The most important data structure in this stage is the forwarding table. Different implementations may have different organizations on the forwarding table. There are usually two schemes: a unified table and a separate table. This experiment uses a separate table design to represent IPv6 unicast forwarding information, as shown in Figure 3. These two separate data entities are called a prefix list and a next hop table. The prefix list consists of a Nexthop Index that includes the prefix (IPv6 Prefix) and entries in the next hop table.

The sending phase: After the data packet has obtained the correct routing information, it needs to encapsulate the link layer before sending, such as adding the Ethernet header. This stage maintains an important data structure Layer 2 address resolution table, which changes the lookup of the routed packet to the correct Layer 2 address.

According to the discussion of the above three forwarding stages, it can be seen that routing forwarding table modeling is the key to true FE modeling. Figure 3 describes the relationship between each table entry and the routing forwarding table and 2 address resolution tables. This separation table design has the following advantages over the unified table: 1) When a group of routes change, some high-performance network nodes need to update the entire FIB, and the workload is large, but the separation table can change a subset of table entries in next hop to effectively update forwarding information.

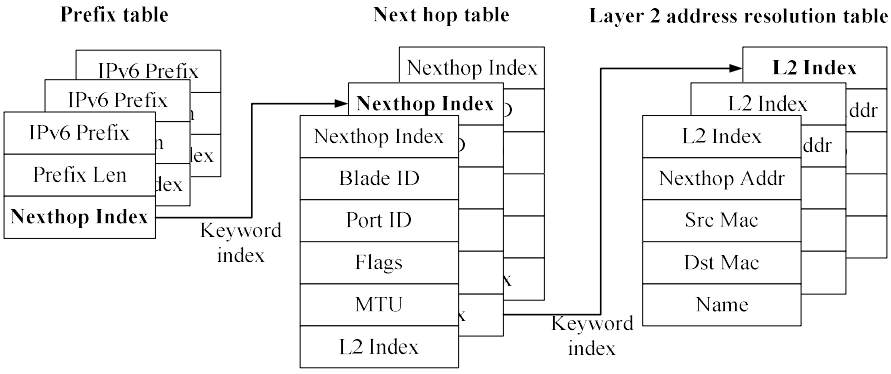

FIGURE 3. IPv6 routing table design

A LFB topology that satisfies IPv6 Routing&Forwarding is shown in Figure 4. The following LFBs are defined: IPv6 Validator LFB, IPv6 UcastLPM LFB, and IPv6 Nexthop Applicator LFB. The prefix table in FIG. 3 is stored in

the IPv6 Ucast LPM LFB, the next hop table is stored in the IPv6 Nexthop Applicator LFB, and the layer 2 address resolution table is stored in the EtherUcastEncap LFB.

EtherPort LFB (input): indicates the network interface, the user accepts data packets input from the outside;

Responsible for removing Ethernet frame headers and performing simple packet classification, mainly for offloading IPv4 and IPv6;

IPv6Validator LFB: verify the IPv6 data packet and input the correct data packet to the IPv6UcastLPM LFB;

IPv6UcastLPM LFB: Longest Prefix Matching (LPM) for IPv6 data packets based on the stored prefix table to determine the NextHop Index;

IPv6 Nexthop Applicator LFB: According to the NextHop Index determined by the previous LFB, the next hop table is queried to obtain the L2 Index;

L2 Index to find the 2-layer address resolution table, get the next hop Dst Mac and other information to complete the IPv6 packet Ethernet frame encapsulation;

EtherPort LFB (output): indicates the network interface used to send encapsulated Ethernet frame data.

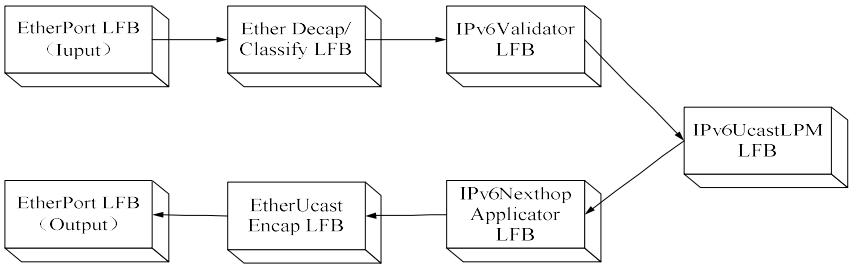

FIGURE 4. LFB topology implementing IPv6 Route&Forwarding

LFB is the basic component of FE. The implementation of each LFB includes model description, input and output, and data structure. Here we take the IPv6 Nexthop Applicator LFB as an example to introduce the design ideas of LFB in detail.

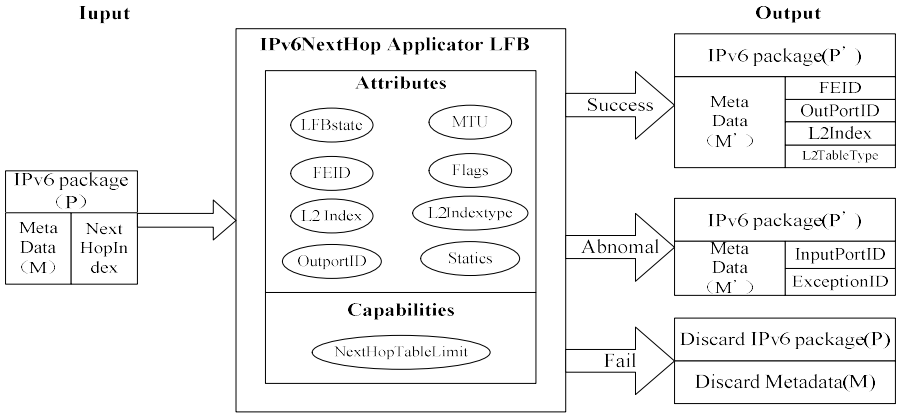

FIGURE 5. LFB design of IPv6 Nexthop Applicator

As shown in Figure 5, the IPv6 Nexthop Applicator LFB performs next-hop operations on IPv6 packets, such as jump limit increments and checksum recalculations. After the IPv6 prefix table is successfully searched, the next hop information needs to be looked up according to the Nexthop Index in the prefix table. Therefore, the IPv6 data packet and metadata retrieved from the IPv6 prefix table become the input of the LFB. . The output has three conditions: 1) It succeeds, and it satisfies the next-hop application's data packet. It generates a new IPv6 data packet modified by IPv6 Nexthop Applicator LFB and new metadata, including FIELD, output port ID, L21ndex, L2TableType, etc.; 2) Abnormal. The following packets are marked as abnormal: the limit is zero, the MTU of the interface is smaller than the packet size, the output port is the same as the input/output port of the received packet, and the packet destination address is used for the local interface. In this case IPv6Nexthop Applicator The LFB will generate modified IPv6

packets and new metadata, including the input port ID and the anomaly ID. 3) Fail to identify the packets that failed in the next hop operation. The IPv6 Nexthop Applicator LFB deletes the IPv6 packets and does not generate metadata.

EXPERIMENTAL DESIGN ASSESSMENT

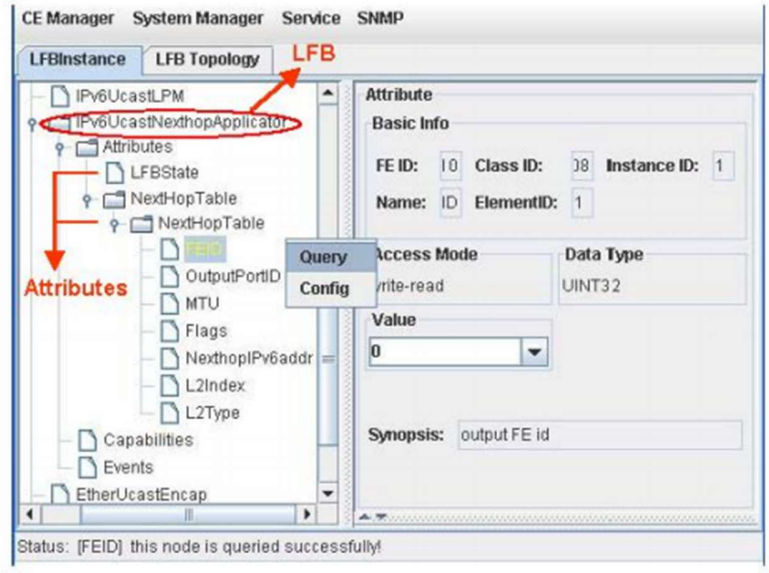

FIGURE 6. Graphical operation interface of the experiment

In the experiment of IPv6 Routing&Forwarding based on ForCES, CE is based on the Linux system, the hardware platform is a general-purpose processor, and the IPv6 routing protocol stack uses GNU Zebra. The FE hardware platform is Intel's network processor and each LFB is based on micro-block processing data packets. Microblocks are physical components that have a single function, typically processing data packets at wire speed, and LFBs are the logical abstraction and modeling of microblocks. Web-based GUIs (Graphical User Interfaces) can dynamically add and remove LFBs, as well as query and configure LFB attributes and capabilities, as shown in Figure 6. This process of defining IPv6 forwarding through the dynamic configuration of the LFB topology is as flexible as a module that is dynamically inserted or deleted in the Linux kernel.

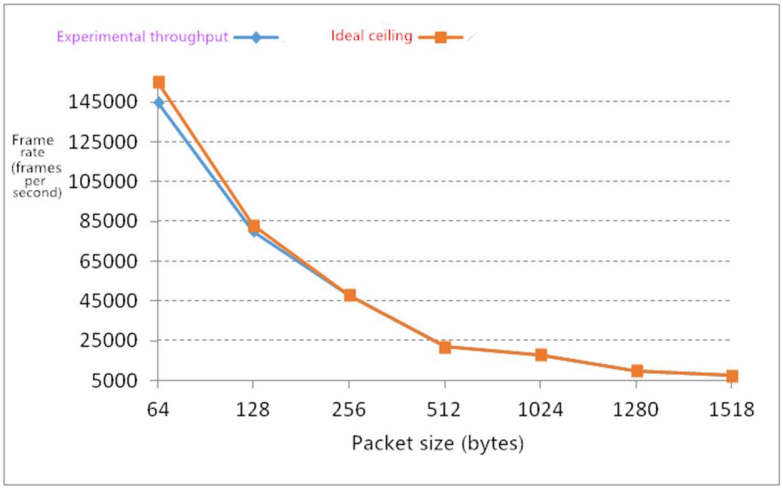

FIGURE 7. Packet Throughput

In the experiment, the FE has two 100M ports. The scenario is configured for IPv6 forwarding across two ports. Spirent SmartBits software is used to measure the zero-loss throughput at the line rates of the two 100M ports. The test generates seven sizes of data packets: 64, 128, 256, 512, 1024, 1280, and 1518 bytes. The result is shown in Figure 7.

It can be seen that the throughput is close to the theoretical limit. Without sacrificing any performance, the IPv6 Routing&Forwarding platform designed and implemented in this lab is highly flexible, configurable, and extensible.

CONCLUSION

A modular IPv6 Routing&Forwarding experiment design is essentially using the modular and abstract features of the ForCES LFB and building a LFB topology supporting the IPv6 packet forwarding function by building blocks, which of course can be constructed using this approach. The network function is far more than IPv6 packet forwarding. Others include firewalls, intrusion detection, and so on. The experimental design has obvious "Two-characters" experimental characteristics. Through experiments, the students have the opportunity to get a glimpse of the internal structure of the network equipment and lay a good foundation for future research and development in this field.

ACKNOWLEDGMENTS

Fund Project: Zhejiang Gongshang University Ministry of Education Innovative Entrepreneurship Training Program(GJ201711002). Zhejiang Gongshang University Higher Education Research Project (Xgy17047)

Corresponding Author: Ming Gao (1979-), Male, Network Professional, Associate Professor, Ph.D., E-mail: gaoming@mail.zjgsu.edu.cn.

REFERENCES

1. Yan Liu, Xiuhong Hou, Guoan Wang, Tingtang Ming. Windows XP based IPv6 experimental network formation and tunnel configuration [J], Journal of Henan University of Science and Technology: 2006,27 (2) :52-54.
2. IPv6 protocol forwarding experiment [OL]. https://qingfeng14.github.io/article/network/IPv6 protocol forwarding experiment.html.
3. Lian Liu, Shuai Han, Wengang Lv. Research on Software Defined Network Architecture and Development[J]. Information Technology and Standardization, 2015 (9).
4. Forwarding and Control Element Separation (ForCES) Framework[S]. IETF RFC 3746, 2004. https://datatracker.ietf.org/doc/rfc3746/.
5. Forwarding and Control Element Separation (ForCES) Forwarding Element Model[S]. IETF RFC 5812, 2010. https://datatracker.ietf.org/doc/rfc5812/.
6. Joe Stubbs, Walter Moreira, Rion Dooley. Distributed Systems of Microservices Using Docker and Serfnode[C]. 2015 7th International Workshop on Science Gateways, 2015:34-29.
7. Service Function Chaining (SFC) Architecture[S]. IETF RFC 7665, 2015. https://datatracker.ietf.org/doc/rfc7665/
8. Jingxiong Zhao, Xiaoju Zhao. IPv4 Protocol and IPv6 Protocol[J]. Software Guide, 2010, 09(2):109-110.
9. Peng Zhichao, Chen Daiwu, Xiao Ben. Research and Application of IPv6 Tunneling Technology[J]. Communications Technology, 2010, 43(6): 71-73.