An Open Network Infrastructure

Xiaoliang Zhao
Verizon Communications Inc.
Ashburn, Virginia, USA
xiaoliang.zhao@verizonbusiness.com

Abstract— **The current closed network infrastructure creates a gap between industry and academy. This paper discusses a framework of a completely open network infrastructure, including open routing devices and open operation of those devices. By such an open network, academia researchers and industry practitioners can jointly work on the network design, optimization and innovation, which provide another dimension of openness for next generation networks.**[*]

Keywords-open system; network operation; router

I. INTRODUCTION

Almost all current Internet Service Providers (ISPs) are operating their networks in a closed world. Their routing devices are proprietary products; their network topology and architecture are one of company's top secrets; and their traffic volume and profile, well, it even should not be talked about. Though all of these have a valid business reason, it creates a gap between industry and academy. Academia researchers are actively searching real network data for their studies, while they often find it is very difficult, if not impossible, to obtain information from the industry. One example is the research in the Internet routing area which turns out even gaining a better understanding about current routing dynamics is hard [1, 2, 3, 4]. One of the major reasons is researchers have no visibility to the internal details of the network infrastructure managed by ISPs. Recently, there is an increased interest to evolve the routing architecture into its next generation [5, 6]. A number of new approaches have been proposed, and if an open infrastructure were available, it could provide a powerful toolset for people to experiment and verify the new proposals.

Open systems have had proven records to inspire new innovations and new business models. Examples are from operating system (Linux), database applications (MySQL), to recent mobile platforms (Android), just name a few. With the emergence of mobile computing, the openness in this field is also highly desired. However, most attentions are focusing on the edge access and mobile devices, fewer has been paid to the infrastructure, while we argue it is actually an integral part of the future mobile Internet.

The idea of an open network infrastructure to provide collaborative opportunities is not new. Many government funded projects, such as Internet2 in US, GÉANT in Europe, CERNET in China, TEIN2 in Asia-Pacific, and many others are all built for this purpose. However, most of those networks, if not all, are using commercial products, which has no or very limited interface for people to introduce new protocols or algorithms. Some pioneer work towards a completely open network infrastructure has been attempted [7, 8] but with a limited scope.

Once the network infrastructure is open, the reliability and security of the network are big concerns. Experimental ideas are immature and the implementations are often broken, how can we handle them with live user traffic? Does it open to attackers for malicious use? Admittedly, they are very difficult issues for business networks but probably applicable to research and education networks. In fact, CERNET2 architectural principles explicitly require the "visibility" of network operation to end users [9]. To address reliability concern, we propose to make redundancy and make-before-break concept as one of the design standards. By redundancy, we mean the infrastructure has more than one data forwarding paths between any sources and destinations. Experiments can be arranged to put on one path with a minimal impact to other paths. Make-before-break concept dictates only when a new path proven working, the major traffic then can shift to it from an old path. Such principles help keeping the impacts minimal, but may not completely avoid the disruption due to the nature of experiments. To address security concern, we can use similar security countermeasures such as tightly controlled jump hosts, TACACS and Kerberos systems etc., and also hide all security related configurations from the public.

Thanks to the Moore's Law, we now have capable hardware whose performance matches or exceeds their earlier counterparts, but with a dramatically reduced cost. For example, a Gigabits Ethernet card with 1Gbps transmission rate now is available for less than 100 USD, but an OC12 card with lower rate of 655Mbps cost about tens of thousand US dollars ten years ago. With cheap commercial off-the-shelf (COTS) hardware, plus availability of open source software, we are able to build routing devices with reasonable performance at low cost, which makes open infrastructure very affordable. As an initial step toward this direction, this paper will discuss the design sketch and some considerations on an open routing platform and open operation.

II. OPEN ROUTING PLATFORM

A routing platform is an essential device for a network, which populates routing information, finds the best routes and forwards data traffic at high speed. Commercial

[*] Disclaimer: the opinions expressed in this paper are authors' own personal opinions and do not represent their employer's view in any way.

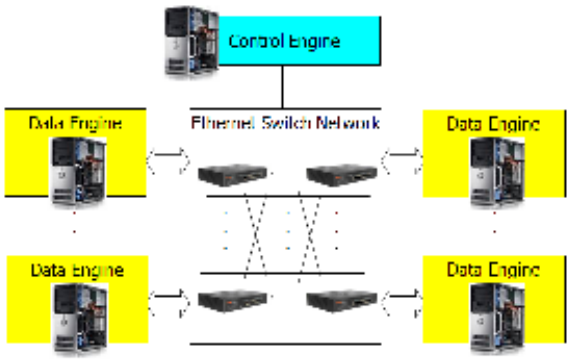

Figure 1: An Example of Open Routing Platform

routing platforms are built in a closed development environment with an aim on the performance and the feature set. Openness is usually not the goal, so commercial products have no or very limited interface for outsiders to modify the software or add new functionalities. Very recently, at least one vendor started opening its software to third party developers [10], but only available through formal business channels and the SDK is quite pricy.

Generally speaking, a routing platform is composed of three major components: control engine, data engine, and switch fabric. Control engine is mainly running routing protocols and algorithms to populate routing information and to select the best routes. Data engine forwards the data packets from one port to another port as fast as possible. A router usually has one (or two for redundancy purpose) control engine(s) but multiple data engines, and a switch fabric connects them together in a non-blocking fashion. We propose to build an open routing platform with commodity hardware and open source routing software.

- Control Engine: A decent COTS desktop PC should be capable to run routing protocols and algorithms in many cases. Open source routing software is also readily available. XORP and Quagga are popular ones which have been adopted by quite a few research projects and commercial products. However, neither of them provides the implementation to interact with data engines.

- Data Engine: There are several options to build a data engine, depending on the desired data forwarding rate. 1Gbps Ethernet card is quite common for many COTS desktop PCs, which can be a very affordable option but relatively low performance. By using high-end hardware, experiment demonstrated 10Gbps transmission rate [11]. Furthermore, by shifting packet processing from host PC to network interface card, NetFPGA project [12] promises line rate forwarding capacity. [13], [14] examined the different hardware architectures of the data engine (or *forwarder*, *line card* as referred in those work) and their performances.

- Switch Fabric: Multiple Ethernet switches with VLAN support can be used to build a multi-layer non-blocking switch network. The switch network will work at Layer 2 to provide point to point connections and non-blocking switching functionality between data engines.

Figure 1 illustrates an example of open routing platform. Multiple PCs are connected to an Ethernet switch network, one of them takes Routing Engine role, while others are Data Engines. The Ethernet switch network forms the Switch Fabric.

To build a low cost but reliable open routing platform, we also propose the following design choices:

- It will be an Ethernet centric platform. Ethernet will be the primary technology to connect different components inside the platform, as well as the primary or only connection type provided for external use. The choice is made because of its low cost, simplicity in design, and wide spectrum of transmission speed.

- Modular design will be used for reliability and extensibility.

- To enhance the reliability, a minimal set of routing protocols and features will be identified and implemented. New feature will be added into the system as a new module.

III. OPEN OPERATION

Commonly the network operation includes several tightly coupled tasks: network design, device configuration, and monitoring / maintenance. Network design takes customer requirements or research objective as inputs, outlines desired network properties then research on different design choices to produce a design document. Device configuration takes the design document as the input and maps the design into actual configurations. Network monitoring keeps collecting traffic statistics and monitors hardware or software failures. Once failure occurs, a maintenance ticket will be issued for people to troubleshoot the issue until a resolution is found. The configuration or design will also be re-examined when necessary. Figure 2 illustrates the process flow between above tasks.

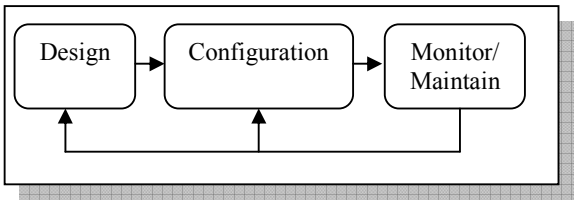

Figure 2: Network Operation Process

- Open Design: For an open network infrastructure, the network design should be open to broader participants. Open design encourages innovative

ideas and constructive discussions on engineering choices, such as performance versus cost, richness of service set versus simplicity and reliability, operational impacts, backward compatibility, and so on.. With the inputs from different perspectives, it is hopeful to make choices based on a more solid foundation, instead of empirical thinking.

- Open Configuration: The configuration for each routing device should be openly accessible, at least world readable. Configuration to a network is analogous to source code to a software system. It should be carefully managed via a configuration management system. We propose the following properties of an open configuration management system: 1) it should support annotations for documenting the design choices along with the configuration; 2) it should be able to keep track of configuration changes; 3) it should support discussion feature to allow free discussions on every configurable parameter. A wiki-based system seems promising in this regard.

- Open Monitoring: Data from network monitoring system should be publicly accessible. It includes user data traffic statistics, routing protocol traces, CPU and memory usage, performance measurements etc. Such data will provide a realistic data source for researchers to study new algorithms, network stability and security, and many other research problems.

IV. CHALLENGES

No doubt that there remain a number of challenges when building a completely open network infrastructure. How to ensure end users' privacy is one of them. When data statistics are made public, it's also made possible to trace a particular user's network activity. This issue can be at least partially alleviated by only publishing aggregated statistics instead of full details.

The coordination between different groups which may or may not share the same interest or same design view will be another challenge. A coordination committee may be necessary to make sure the resources are fairly shared.

Security is another important concern. It is still debatable if security can be enhanced by openness or not, but we feel at least an open infrastructure can provide a new tool for new ideas such as building the security countermeasures into the routing devices to protect end users from worm propagation or DDoS attacks.

REFERENCES

[1] Anja Feldmann, Olaf Maennel, Z. Morley Mao, Arthur Berger, Bruce Maggs, "Locating Internet Routing Instabilities", ACM SIGCOMM, 2004.

[2] Shih-Ming Tseng, Xiaoliang Zhao, Shyhtsun Felix Wu, Ke Zhang, "On Reverse Engineering the Management Actions from Observed BGP Data", 1st IEEE Workshop on Automated Network Management (ANM'08), 2008.

[3] Wang, F. and Gao, L., "On inferring and characterizing internet routing policies", ACM Internet Measurement Conference (IMC), 2003.

[4] Mahajan, R., Wetherall, D., and Anderson, T., "Understanding BGP misconfiguration", ACM SIGCOMM, 2002.

[5] Dan Jen, Michael Meisel, He Yan, Dan Massey, Lan Wang, Beichuan Zhang, Lixia Zhang, "Towards a New Internet Routing Architecture: Arguments for Separating Edges from Transit Core", HotNets-VII, October 2008.

[6] The Locator Identifier Separation Protocol (LISP), http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_11-1/111_lisp.html

[7] User Controlled LightPath (UCLP), http://www.canarie.ca/canet4/uclp/uclp_software.html

[8] The Hybrid Optical and Packet Infrastructure (HOPI) project, http://networks.internet2.edu/hopi/

[9] CERNET2 Architectural Principles (in Chinese), http://www.edu.cn/internet_2_1339/20080620/t20080620_303870_1.shtml

[10] Juniper Partner Solution Development Platform, http://www.juniper.net/us/en/products-services/nos/junos/psdp/

[11] Hagsand, Olsson, Görden, "Towards 10G/s open source routing", Linux kongress, 2008.

[12] NetFPGA Project, http://www.netfpga.org

[13] Khan, A.J. Birke, R. Manjunath, D. Sahoo, A. Bianco, A. , "Distributed PC based routers: Bottleneck analysis and architecture proposal", in Proceeding of High Performance Switching and Routing (HSPR), 2008.

[14] A. Bianco, J. M. Finochietto, G. Galante, M. Mellia, D. Mazzucchi, and F. Neri, "Scalable layer-2/layer-3 multistage switching architectures for software routers," *Globecom* 2006.