XBurner: A XENebula-Based Native Traffic-Generation Platform

Toshiyuki Miyachi[1], Shinsuke Miwa[1], and Yoichi Shinoda[2]

[1] National Institute of Information and Communications Technology,
Asahidai 2-12, Nomi, Ishikawa, Japan
[2] Japan Advanced Institute of Science and Technology
Asahidai 1-1, Nomi, Ishikawa, Japan
{miyachi,danna}@nict.go.jp, shinoda@jaist.ac.jp

Abstract. There are two types of network traffic in experimental environments. One is traffic which is derived from target elements and the other one is background traffic which is derived from surrounding elements. There is very little knowledge on the relationships between new elements and existing elements; therefore, surrounding elements that seem to have no apparent relationship with the target elements should also introduced into the environmental environments. Therefore emulating background traffic is important to take realistic experimental results.

We propose XBurner—a platform that can be used to generate mass background traffic using a number of actual and native application software on virtual PCs. Driving actual application software is important to introduce real behavior of elements on real environment into experimental environments. The environment in this platform is developed using AnyBed and XENebula and is controlled by SpringOS.

1 Motivations

Background traffic has different characteristics from traffic for target elements. In most cases, one background connection doesn't share wide bandwidth, but many tiny connections totally occupy much bandwidth. Therefore, application software which send out background traffic don't require high performance machines and many of these connections with different source-destination pairs are required to make experimental environment more realistic.

Our approach is to build virtual network consists of many nodes for running many actual network software. Each virtual nodes with its own IP addresses can make many source-destination IP address pairs by their communication using actual network software. By using actual software, experimenters can introduce their own traffic pattern, their own implementations, and several implementations for one specification which have different behaviors.

2 Design and Implementation of XBurner

We use some existing proposals to generate native traffic from remotely-managed actual application software.

T. Magedanz et al. (Eds.): TridentCom 2010, LNICST 46, pp. 629–631, 2011.

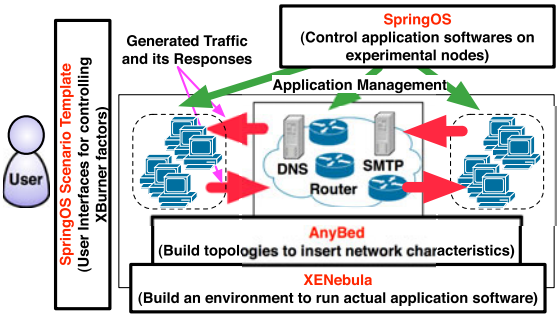

Fig. 1. Traffic Generation with XBurner

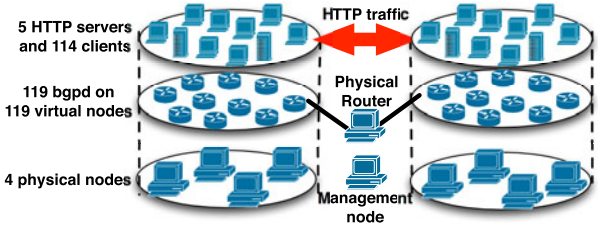

Fig. 2. Experimental Environment

XENebula[1] is a Xen-based platform for building large-scale network experimental environment. It has capability to setup several hundred of virtual machines on a dozen of physical nodes.

AnyBed[2] is a tool that create configurations for the quagga-routing daemon according to topology databases such as CAIDA AS Relationships[3] dataset.

SpringOS[4] is a software suite for controlling experimental nodes to implement user requirements on network testbeds.

Using AnyBed and XENebula, users can run many virtual nodes and emulate a large-scale L3 network that is actually routed by dynamic-routing protocols. SpringOS can trigger application software on the environment to generate native traffic. Its experimental scenarios can be used as templates for other experimenters.

Figure 1 shows the overall schematic representation of XBurner. Common services such as DNS and SMTP which are implicitly used by many applications, may be implemented on the environment.

To evaluate our proposal, we implemented XBurner for simple traffic generation. We built a large dumbbell topology using 238 virtual nodes on 8 physical nodes which consist of Pentium E2200 2.2 GHz CPU, 4 GByte Memory, and 2 NICs; this topology is illustrated in Figure 2. The nodes are connected to a intelligent switch. We emulate two Inter-AS networks on 4 nodes and they were connected via a physical PC router.

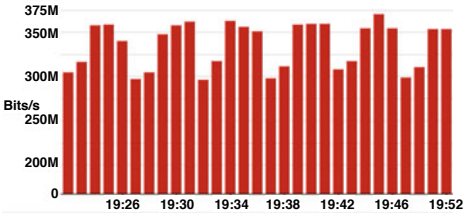

Fig. 3. Measured traffic on the beside of the physical router

To build this environment, we extend XENebula and AnyBed because current implementations of them are focused only to generate one BGP networks. Our extensions cover to run much more application software on virtual nodes and specifying IP address range of emulated network, and so on.

On the building the topology, we generate traffic using our SpringOS scenarios. The server nodes generate "`apache`" configuration and startup it then clients run "`wget`" HTTP clients to get 1M-size file to its servers. After getting the file, clients sleep 1 second and get the file again.

There are many source and destination IP address pairs and the throughput observed beside of the physical router by "`sflow`" is shown in figure 3. This experiment shows that XBurner can run as a platform for traffic generation.

3 Future Works

XBurner is merely a platform for generating traffic using native application software, and the most important point in future is consideration of traffic patterns and packet forms composing them for evaluating typical services and implementations. We'll generate many kinds of traffic and provide them to users as SpringOS scenarios to generate the same traffic patterns. There are still several open issues about XBurner; accuracy of triggering actual software, coverage investigation, preparing kinds of template, and so on.

References

1. Miwa, S., Suzuki, M., Hazeyama, H., Uda, S., Miyachi, T., Kadobayashi, Y., Shinoda, Y.: Experiences in Emulating 10K AS Topology with Massive VM Multiplexing. In: VISA 2009 (August. 2009)
2. Suzuki, M., Hazeyama, H., Kadobayashi, Y.: Expediting experiments across testbeds with AnyBed: a testbed-independent topology configuration tool. In: TridentCom 2006 (March 2006)
3. CAIDA. AS Relationships, `http://www.caida.org/data/active/as-relationships/`.
4. Miyachi, T., Chinen, K., Shinoda, Y.: StarBED and SpringOS: Large-scale General Purpose Network Testbed and Supporting Software. In: Valuetools 2006 (October 2006)