Can Virtual Machine Network Simulate Real Network?

Gang Peng, Anirban Sinha

shuaige@cs.ubc.ca, anirbans@cs.ubc.ca
Computer Science Department, University of British Columbia

Why do we care about this?

Although there exists some well known network simulation tools, such as NS2 [1] and Emulab [2], Network simulation with a decent number of nodes for experimental purposes is still an interesting area that needs further exploration primarily due to the scalability & simulation problems of existing tools.

NS2 is a representative of a large number of software network simulators that might be considered to be the most popular one among them. It can simulate all kinds of network devices, such as ordinary PCs, mobile devices and sensors, and can establish complicated network environments & topologies. It is also convenient to download and install NS2 on our personal computers. Despite its popularity & simplicity, it does have two major drawbacks. First, NS2 is a complete software simulator, which inevitably brings some bugs and limitations along with it [6]. Second, NS2 is a single process simulator, which means that it has to simulate all the networking components, including network devices and events, sequentially. However, network environment is essentially a parallel environment in which network devices change their states concurrently and events happen simultaneously.

Emulab is another important network simulation tool, which resorts to physical devices to simulate network devices. It might be considered better than NS2, but it still has some problems. Firstly, it is a shared resource and is simultaneously being used by many users. Although currently it has about 400 physical nodes, sharing among a large number of users makes the effective available number of nodes very small. Further, the cost to build such a resource by an individual institution might be very high.

What do we plan to do?

We are trying to find a tradeoff between these two kinds of simulation tools. We plan to use virtual machines (VM) to simulate real network devices and use virtual machine network (VMN) to simulate real network. However, it is not a new idea. In 1981, Richard T. Wang proposed a similar idea [3] and recently, Michael Vrable used a this idea in his paper on security [4]. However Richard's paper only provides a very simple experiment. Moreover network devices and structures have changed a lot since the time the paper was written. Michael's paper focuses on security, not network performance. We believe that it is the right time to revitalize this idea and do some investigation on it.

In our work, we will try to establish some network environments on VMN, collect performance data from it and compare these data with those from real network, NS2 and

Emulab. With the comparative figures, we can figure out whether VMN is a good simulation tool of real network, and if it is not a good choice, why.

The merits:

First, it might prove to be a cheap solution. What we need is a powerful PC and VMM software. Second, it is more extensible compared with Emulab. A single VMM can support thousands of simultaneous virtual machines [5], if the hardware platform is powerful enough, which means we can simulate a network composed of thousands of network devices. Third, it is based on real hardware as opposed to NS2. We use real applications and OSes, which include real protocol stack.

Possible Problems:

1. **Timing:**
   Network Performance is really time sensitive, and we know that VMMs' timer will not be so accurate. We worry this inaccuracy could "pollute" our performance data and make this method impractical. If we can guarantee that every VM gets almost the same amount of time slot, maybe we can handle this problem.
2. **Number of VMs:**
   Although VMM can support a large number of VMs, our hardware platform maybe a bottleneck. If we can only support limited number of VMs, this method wouldn't be very useful. Maybe we should recompile Linux kernel to make it as small as possible.
3. **Data collection and analysis:**
   In simulation, data collection and analysis is really a challenge. NS2 provides some tools to help users to collect and analyze data. Inefficient data collection and analysis methods could make our work difficult and vulnerable.

Milestones:

Mar. 3 ~ Mar. 12: Background review and Installing Xen
Mar. 13 ~ Mar. 26: Conducting experiment on VMN and comparing with other simulation tools
Mar. 17 ~ Mar. 31: Finishing report and further work

Future Work:

We can modify Xen to provide us with complicated network environment. For example, we can setup the network MMU, ratio of losing packages and some other physical features of network, by modifying Xen's virtual network part.

Bibliography & Reference:

[1] Network Simulator homepage, http://www.isi.edu/nsnam/ns/doc/index.html
[2] Emulab, http://www.emulab.net/
[3] Michael Vrable, Justin Ma, Jay Chen, David Moore, Erik Vandekieft, Alex C. Snoeren, Geoffrey M. Voelker, Stefan Savage. Scalability, fidelity, and containment in the potemkin virtual honeyfarm. In Proceedings of the twentieth ACM symposium on Operating systems principles 2005, Brighton, United Kingdom October 23 - 26, 2005
[4] A. Whitaker, M. Shaw, and S. D. Gribble. Scale and Performance in the Denali Isolation Kernel. In Proceedings of the 5th ACM/USENIX Symposium on Operating System Design and Implementation (OSDI), Boston, MA, Dec.2002.
[5] Richard T. Wang, J. C. Browne. Virtual machine-based simulation of distributed computing and network computing. ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1981 ACM SIGMETRICS conference on Measurement and modeling of computer systems SIGMETRICS '81, Volume 10 Issue 3
[6] NS2 Limitations, (http://www.isi.edu/nsnam/ns/ns-limitations.html)