INM/WREN '10: 2010 Internet Network Management Workshop/Workshop on Research on Enterprise Networking

April 27, 2010
San Jose, CA

Introduction by Alva Couch (couch@eecs.tufts.edu)

INM/WREN '10 brought together academic and industry researchers to focus on the common goals of effectively managing the Internet and setting the future course of enterprise networking. This year's meeting involved many themes, including the rapidly evolving role of programmable networks and the OpenFlow interface to routing hardware, new monitoring and troubleshooting techniques, innovative uses of virtualization, and management challenges that arise from cloud computing.

OpenFlow is an open standard by which applications can interact directly with switches and routers via a portable interface. This exposes a usable set of switching controls at the application layer and works around the traditional roadblock of having to modify internal switch/router software in order to control switching. INM/WREN papers studied several uses of OpenFlow, including replacing traditional monitoring, engineering enterprise traffic, and distributing management.

Cloud computing was also a major theme. Papers not only studied how to manage clouds but also proposed several innovative techniques that show promise in enabling future cloud architectures. A multi-vendor panel exposed some of the more subtle challenges of cloud computing.

PROGRAMMABLE NETWORKS AND THEIR APPLICATIONS

Summarized by Alva Couch (couch@eecs.tufts.edu)

- ***Automated and Scalable QoS Control for Network Convergence***
  Wonho Kim, Princeton University; Puneet Sharma, Jeongkeun Lee, Sujata Banerjee, Jean Tourrilhes, Sungju Lee, and Praveen Yalagandula, HP Labs

Wonho Kim proposed a tiered system for QoS control using the OpenFlow architecture. Flows match flow specifications, which are grouped into slice specifications that have QoS objectives. Slices are mapped to hardware queues to determine priority. This mapping is performed independently for each switch. Multi-flow performance is managed by use of a Shortest-Span-First heuristic, in which the switch that is most performance-constrained is situated first in a required path. Small-scale experiments demonstrate substantive performance improvements.

- ***The Case for Fine-Grained Traffic Engineering in Data Centers***
  Theophilus Benson, Ashok Anand, and Aditya Akella, University of Wisconsin—Madison; Ming Zhang, Microsoft Research

Theo Benson proposed a new approach to datacenter traffic engineering called MicroTE. Datacenter traffic engineering involves balancing traffic within the center to avoid congestion and delays. But current techniques for traffic engineering, including equal cost, multi-path (ECMP) and spanning trees (STP), utilize single paths and ignore redundant paths when balancing. The MicroTE traffic engineering approach uses all paths, exploits short-term predictability for quick adaptation, and coordinates scheduling with a global view of traffic. The strategy was simulated using a trace of a cloud data center with about 1500 servers and about 80 switches. According to this simulation, MicroTE results in traffic patterns that are much closer to optimal than traditional approaches. An audience member asked how one defines optimality. It is estimated via linear programming with complete information. Another audience concern was whether the centralized approach advocated by MicroTE is scalable to large data centers.

- ***HyperFlow: A Distributed Control Plane for OpenFlow***
  Amin Tootoonchian and Yashar Ganjali, University of Toronto

One concern about OpenFlow is that its most common use case, involving centralized control, may not scale. Amin Tootoonchian presented HyperFlow, a distributed mechanism for managing OpenFlow switches, in which multiple switch groups are each managed by an independent manager. Managers do not know about one another and assume that they are alone in managing the network. Managers coordinate by distributing information on flows in adjacent switch groups, using the WheelFS file system to cache event streams. The authors conclude from performance limits for WheelFS that distributed consistency is sufficient for effective management if there are under 1000 updates/sec to the

WheelFS. The audience wondered how this approach can be combined with traffic engineering.

VIRTUALIZATION AND BEYOND

Summarized by Andrew D. Ferguson (adf@cs.brown.edu)

- ***The "Platform as a Service" Model for Networking***
  Eric Keller and Jennifer Rexford, Princeton University

Network virtualization is currently provided using the Infrastructure as a Service (IaaS) model, which requires customers to configure their virtual network equipment as if it were the physical hardware. In order to ease this configuration burden, Eric Keller described efforts to present virtual networks using the Platform as a Service (PaaS) model. In this model, customers would maintain the freedom to reconfigure their network without being confronted with the complexities of the physical infrastructure. Example applications include customer-controlled routing within a cloud platform (such as Amazon EC2) or customer-created multicast groups to avoid the need for overlay networks. Instead of configuration files, customers would provide executable scripts which dynamically adjust router functionality in response to network events. This effort is a work in progress; some of the challenges remaining are to identify the appropriate router abstraction for end users and to determine how to resolve conflicting updates from customers.

- ***vDC: Virtual Data Center Powered with AS Alliance for Enabling Cost-Effective Business Continuity and Coverage***
  Yuichiro Hei, KDDI R&D Laboratories, Inc.; Akihiro Nakao, The University of Tokyo; Tomohiko Ogishi and Toru Hasegawa, KDDI R&D Laboratories, Inc.; Shu Yamamoto, NICT

Yuichiro Hei presented a proposal for virtual data centers, defined as geographically distributed components presented as a single data center. Currently, providers must build multiple physical data centers in order to implement business continuity. However, fixed costs make this prohibitively expensive for small companies. By sharing physical data centers, small providers could maintain the scale and isolation of a single data center while achieving the reliability of multiple data centers. Cost-effective and reliable paths between the physical components of the virtual data center can be provided by an AS Alliance, a group of ASes which share information about unused routes and exchange traffic. In the event of link failure, an AS in the alliance can select an alternate route through the alliance in order to restore connectivity within the virtual data center.

- ***Europa: Efficient User Mode Packet Forwarding in Network Virtualization***
  Yong Liao, University of Massachusetts at Amherst; Dong Yin, Northwestern Polytech University, China; Lixin Gao, University of Massachusetts at Amherst

Lixin Gao presented a solution to the problem of achieving high performance in network virtualization platforms such as VINI. These platforms run in user mode and are slowed by the overhead of copying packets and handling system calls. Europa applies zero-copying schemes from the OS research community to platforms for virtual routing. The Europa kernel shares a packet buffer between the kernel and user space and avoids system calls by requiring that access to the buffer be asynchronous. An atomically updated state variable on each packet in the buffer serves as a mutex to mediate access. Experiments show that Europa-enabled virtual network platforms running in user mode are able to nearly match the performance of the Click virtual router when running in kernel mode.

MEASUREMENT AND MONITORING

Summarized by Alva Couch (couch@eecs.tufts.edu)

- ***A Preliminary Analysis of TCP Performance in an Enterprise Network***
  Boris Nechaev, Helsinki Institute for Information Technology; Mark Allman and Vern Paxson, International Computer Science Institute; Andrei Gurtov, Helsinki Institute for Information Technology

Surprisingly little study has been made of the actual performance of enterprise networks, perhaps because they are viewed as working "well enough." This study analyzes enterprise data from the Lawrence Berkeley National Lab from October 2005 to March 2006. Data was captured at switches for 351 hosts (4% of the network) and described 292 million intra-enterprise TCP packets. Bro 1.5.1 was used to associate packets with connections and analyze the nature of connections. The distribution of data within connections was exceedingly heavy-tailed: 90% of bytes appear in 160 connections out of about 363,000. Inter-subnet traffic was about 10 times intra-subnet traffic. Conversely, 57% of connections were short-lived and transmitted small amounts of data, while 37% of connections were long-lived and transmitted small amounts of data. The audience wondered whether there was any evidence of congestion or network problems, but in this data, there was no such evidence.

- ***Extensible and Scalable Network Monitoring Using OpenSAFE***
  Jeffrey R. Ballard, Ian Rae, and Aditya Akella, University of Wisconsin—Madison

Jeffrey Ballard presented OpenSAFE, a flow-monitoring layer based upon OpenFlow. OpenSAFE can selectively map flows to a designated port and direct exceptions to a software component. The configuration of OpenSAFE is specified via the ALARMS (A Language for Arbitrary Route Management for Security) language. Monitoring requests define input sources, data filters, and application processing. Line-speed monitoring is accomplished by tapping data from a port of interest to an otherwise unused dedicated monitoring port. OpenSAFE allows monitoring of waypoints, which are virtual points in a flow's path that may not correspond to hardware entities. This approach promises selective line-

speed monitoring, but there is a danger of data overrun for high traffic rates. The audience asked whether some of the software features could be implemented within OpenFlow itself rather than at the application layer. The authors responded that current OpenFlow implementations suffer from both implementation limitations and portability issues that might prevent implementation within OpenFlow.

- ***Beyond the Best: Real-Time Non-Invasive Collection of BGP Messages***
  Stefano Vissicchio, Luca Cittadini, Maurizio Pizzonia, Luca Vergantini, Valerio Mezzapesa, and Maria Luisa Papagni, Università degli Studi Roma Tre, Rome, Italy

In current BGP monitoring approaches, including BMP, data is gathered by polling BGP routers. Stefano Vissicchio proposed that data be collected by passive observation of traffic on BGP links. Links are tapped and the tap information is sent to a collector node. This is much less intrusive upon router performance than relying upon the router's management interface, and can be configured and extended without modifying BGP router software. Initial experimental results show promise for the technique.

NETWORK MANAGEMENT—EXPERIENCES

Summarized by Wonho Kim (wonhokim@princeton.edu)

- ***Experiences with Tracing Causality in Networked Services***
  Rodrigo Fonseca, Brown University; Michael J. Freedman, Princeton University; George Porter, University of California, San Diego

Rodrigo Fonseca presented their experience with integrating X-Trace into existing networked systems. As there is no standard way to associate software components in a networked system, it is difficult to find root causes and debug the systems via the existing device-centric approach. Rodrigo showed that the instrumentation can be easily done when X-Trace is integrated into 802.1X and that full "happen-before" events can be captured. When X-Trace is used in complicated large-scale distributed systems (e.g., CoralCDN), it gives previously unavailable information about existing performance problems such as long delays incurred for unknown reasons. X-Trace requires modifications to target systems, but it enables instrumentation with sampling and does not require support for time synchronization between multiple nodes. One interesting question was how kernels could support appropriate tracing. If a kernel can support recording system events (e.g., thread creation), this would be helpful for getting richer information about existing causality.

- ***Proactive Network Management of IPTV Networks***
  R.K. Sinha, K.K. Ramakrishnan, R. Doverspike, D. Xu, J. Pastor, A. Shaikh, S. Lee, and C. Chase, AT&T Labs—Research

Rakesh Sinha presented the architecture of the IPTV service network within AT&T. The IPTV service is inherently sensitive to delay and packet losses, and quality of service is directly visible to its 2 million-plus subscribers. Each channel has its own multicast tree, and to prevent service disruptions from link failures, Fast Reroute (FRR) is implemented using virtual links as backup links, so no OSPF convergence is required. However, multiple concurrent failures are common, and the failures can propagate, because FRR is not visible to IGP. To handle failures, one must provide network administrators with a more holistic view from multiple monitoring tools (e.g., NetDB, OFSPMon, MRTG). The authors developed Birdseye, a Web-based visualization tool, to provide a more comprehensive view of monitoring data from multiple sources and alert administrators to important events in the IPTV network.

PANEL

- ***What Do Clouds Mean for Network and Services Management?***
  Summarized by Eric Keller (eric.r.keller@gmail.com)

  Moderator: Dr. Anees Shaikh, IBM T.J. Watson Research Center
  Panelists: Adam Bechtel, Yahoo!; Tobias Ford, AT&T; Stephen Stuart, Google, Inc.; Chang Kim, Microsoft

Each of the panelists was given five minutes to summarize their position.

Chang Kim of Microsoft emphasized that there are several different management perspectives. Management may involve managing existing networks using the cloud, or building new networks and services on the cloud, or managing the cloud provider's network and services themselves. Each case presents great opportunity because there are many more affordable resources on demand. The downside, however, is that there is a lack of visibility, since cloud providers are less likely to expose individual components (failures, links, etc.), and lack of a standard management interface, especially because the interface is likely to be more service-centric than component-centric.

Tobias Ford from AT&T pointed out that the cloud is a convergence of several effects. Most importantly, the cloud should make the customers feel comfortable. SLAs explain availability to enterprises, while the provider must work closely with enterprises to explain the benefits, such as cost savings, of using the cloud. Ford believes the key to making the cloud work is automation. They are automating the creation of VPLS and VPN. The proper level of abstraction is crucial, and transparency makes customers more comfortable, so he advocates exposing the network elements to customers.

Stephen Stuart from Google considers the cloud to be a collection of different applications competing for resources. In this environment, things break all the time, but the network works (perhaps too hard) to hide broken parts from applications (e.g., TCP is good at hiding a broken network). Instead, things that break should be exposed to applications, which could behave differently if given better data—

they would have "actionable intelligence." To achieve this, we need programmable networks. Today, most networks are configured via a command line interface with layers of software that have to translate intent into vendor-specific configuration languages. Because of this, configuration is an act of faith. Instead, we need an RPC interface into the network element which allows application developers to "program the network." For example, when a rack switch fails, it is preferable for applications to react by throttling input to the rack or by moving the application off to another location, rather than continuing to bang on the network and let TCP handle it. This, of course, can be solved with OpenFlow.

Adam Bechtel of Yahoo! noted that things are much more complex than they used to be. Web pages are now dynamically prepared via a collection of services rather than statically served by a single host. There are no monopolies among the Internet/Cloud, so interaction among companies is more important. If external providers can do something better, customers will use them (e.g., CDNs). In networking, we have the appropriate abstraction for federation, which is BGP. It remains unclear what the proper abstraction level is for outsourcing services. Providing and consuming outsourcing comes with new problems. When you do something yourself, you have a lot of visibility into it, but if you outsource it, you lose that visibility. When you pay for something (e.g., outsourcing some service), you have an incentive to manage how applications use resources. How does the outsourcing provider manage the service? How does one kill the customer's misbehaving MapReduce? These remain open problems.

The floor was then opened for questions.

Alva Couch asked whether the application programmer will continue to have to understand the underlying architecture of the cloud, or whether it will be eventually abstracted away. Stephen Stuart responded that this depends upon one's performance requirements. Ford and Kim pointed out the need to interface with the networking team on the customer's side. Customer networks are not going away, so ideas on exposing APIs are important. Bechtel pointed out that for certain applications, one does need to know internals (e.g., Hadoop), while for others (e.g., search), internals can remain hidden. Stephen Stuart then reversed the question: do you expect legacy applications, which use particular features, to still be supported (e.g., Ethernet multi-cast)? Ford said that for new applications, control over things like multi-cast is essential, so we need to figure out how to expose them meaningfully.

Nick Feamster then asked what kind of interface (e.g., command line, scripts, database configuration sent to routers, RPC, programmable hardware) administrators and software should have in order to manage network elements. Stephen Stuart pointed out that OpenFlow allows him to pull the control plane out of switches. The application will not know what "my load will be X" means, but a controller can and will be ahead of the game rather than reactive. Nick Feamster later followed up, pointing out that you can do that with routers today—pushing configurations rather than an RPC interface. Stuart clarified that OpenFlow gives more control than using intermediate software layer interpretations. Bechtel mentioned that AT&T is aiming for using configuration stored in databases and sent to switches. In his experience, managing ACLs took the most amount of time, so they built a system for automating configurations; the next target is VIPs (virtual IP) and sending that to load balancers.

Kobus van der Merwe then asked whether one should think about end-to-end or localized performance in the data center. Stephen Stuart said that in his wackier thoughts, he does not need a network. The applications do not mind having a network, as long as the network does what the applications want. Ford believes in simplifying APIs so they apply across many domains.

Someone then asked how the panelists are dealing with the multi-level security requirement of the government. Ford pointed out that this is consistent with their view that an enterprise wants to know where its data is. By exposing APIs which controls those policies (e.g., where data flows), this would bring together the security concerns.

Sanjay Rao brought up the cloud provider lock-in problem and asked whether there would be an API that enables switching providers. Kim believes that providers will come up with their different APIs, much as network management APIs are not well standardized today. Bechtel believes the APIs will be a source of differentiation. Ford disagrees. He instead feels that the providers must work with partners, have some form of federation, and use open protocols and APIs.

Jeff Mogul questioned what the incentive is for providers to have provider lock-in. Aditya Akella suggested that there is a role for a cloud broker that leases from multiple providers and handles the differences. Stuart pointed out the success of Gmail's IMAP interface. Because customers can download their email and escape the cloud features when needed, they feel more comfortable with adopting them. Chang noted that IaaS is working because customers are free to bring whatever software they want and can move it if they want.

Amin Tootoonchian asked about challenges facing the deployment of OpenFlow. Stuart stated that before deployment is implementation and before implementation is design. Right now Google is working hard on the design of OpenFlow.

Kobus van der Merwe asked how one deals with operational complexity. Dave Maltz answered that the nice thing about the cloud is it forces you to do things that can scale.