A Demonstration of Virtual Machine Mobility in an OpenFlow Network

David Erickson,
Brandon Heller,
David Underhill,
Jad Naous,
Guido Appenzeller,
Guru Parulkar,
Nick McKeown,
Mendel Rosenblum
Stanford University

Sailesh Kumar,
Valentina Alaria,
Pere Monclus,
Flavio Bonomi
Cisco Systems

Jean Tourrilhes,
Praveen Yalagandula,
Sujata Banerjee,
Charles Clark,
Rick McGeer
Hewlett Packard

ABSTRACT

OpenFlow is a new architecture that was recently proposed as a way for researchers to run experimental protocols in the campus networks we use every day. OpenFlow is added as a *feature* to commercial Ethernet switches, routers and wireless access points - and provides a standardized hook to allow researchers to run experiments, without vendors having to expose the internal workings of their switches. During the summer of 2008, the network in two buildings at Stanford has been upgraded to run on OpenFlow-enabled commercial Ethernet switches and routers.

In our demonstration, we illustrate how an OpenFlow network allows mobility across layer-2 and layer-3 networks. Consider the scenario where several mobile clients are moving in a wireless network; the mobile clients are running software that is connected to a central server running inside a virtual machine. Figure 1 shows an example network topology for such a scenario. The application in our demonstration is an interactive multi-player game. For this class of applications, the user experience is highly latency sensitive. In order to maintain the lowest-latency connection between client and server, the server's virtual machine is migrated between virtual machine hosts located in different parts of the building, to always be as close to the mobile clients as possible. This move must be seamless: as the VM moves, applications on the VM and the mobile client should continue to communicate, without requiring changes to their IP or MAC addresses.

Usually, when a device moves to a new subnet it needs a new IP address and all outstanding connections are lost. This could be handled by a rendezvous point (e.g. Mobile IP) and additional infrastructure, however such a configuration introduces undesired delay to the client, harming the user experience. With OpenFlow, devices and VMs are allowed keep their original IP addresses, maintaining all existing connections. Further, OpenFlow enables ultimate control over packet paths; in our demonstration we route packets along the least number of physical links between any two points in the network, and examine the network for shorter paths from clients to other VM hosts. Our goal with this

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.
ACM 978-1-60558-175-0/08/08.

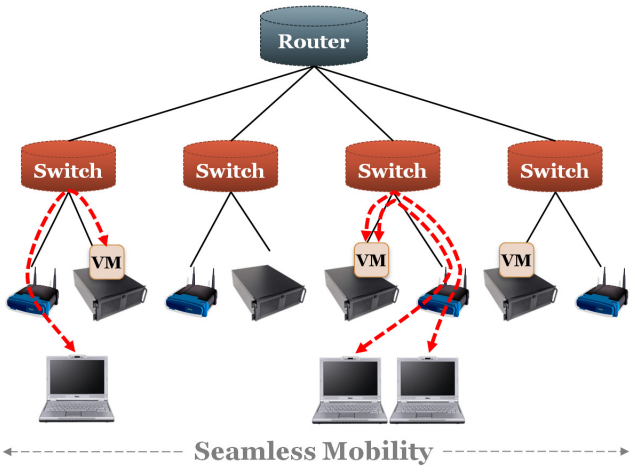

Figure 1: Example demonstration topology.

demo is not to promote one particular re-routing and mobility mechanism. Rather, we want to show how OpenFlow allows new re-routing and mobility managers to be quickly prototyped and deployed by researchers.

This demonstration also seeks to show that while OpenFlow is a research project, it can run on hardware that is in many campus wiring closets today, operating right alongside the production network traffic. Stanford is currently collaborating with Cisco Systems and Hewlett Packard to enable this functionality in their equipment.

Categories and Subject Descriptors

C.2.6 [**Computer Communication Networks**]: Internetworking; C.2.1 [**Computer Communication Networks**]: Network Architecture and Design; D.4.4 [**Operating Systems**]: Communications Management

General Terms

Management, Design, Experimentation

Keywords

Internet, architecture, virtual machine, mobility