

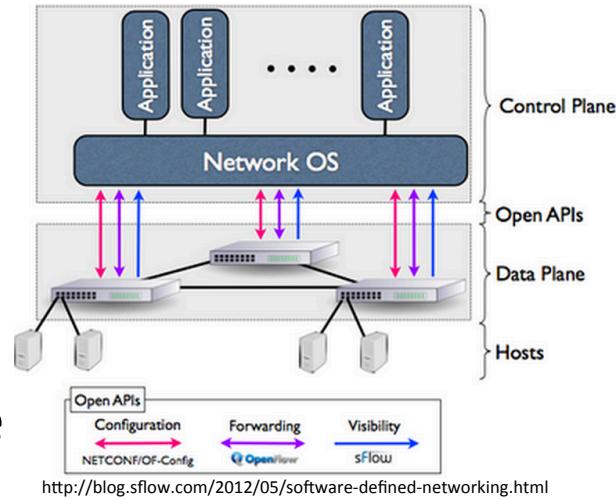
A ReMOOCable Experience: Teaching Computer Networking to the Masses

Nick Feamster
Georgia Tech

Co-Conspirators:
Vrushali Moghe (Course Designer),
Muhammad Shahbaz (TA)

My MOOC: Software Defined Networking

- Extremely hot, new topic
 - No existing “real world” courses on the topic (yet)
 - A chance to develop an archetype course (and material) that others might use in their own courses
 - I wanted to be copied. This seemed like a good way.
- Boutique topic means that there were no “set in stone” ways of teaching the course
 - Easier to think outside the box, since there was no existing material anywhere





Software Defined Networking

by Nick Feamster



Syllabus

Week 1

Week 2

Week 3

Week 4

Week 5

Week 6

Video Lectures

Quizzes

Programming Assignments

Discussions

Surveys

About Us

FAQ

Course Wiki

Join a Meetup

Help Articles

Flagged Content

Instructor Help Articles

I18N Editor

Announcements

Welcome to Week 4

Welcome to Week Four of the course, where we will cover the nuts and bolts of software defined networking. We wanted to draw a few specific points to your attention:

- **This week has the heaviest workload of any of the weeks in the course.** This week is the heaviest week of the course, as it has two modules: Module 4 on the control plane, and Module 5 on the data plane. It has about 150% of the amount of work of other weeks in the course, including a programming assignment in Pox. As a result, you may fall slightly behind this week. Keep in mind that, although we set weekly deadlines for the course, we have extended all hard deadlines for the course to August 5. Weeks 4 and 5 of the course have extensive programming, which will please people who thought earlier modules were too high level and hopefully challenge everyone. Do not worry: If you fall slightly behind, there will be some time to catch up in the coming weeks.
- **Second programming assignment: Pox.** This week, you will gain experience with programming with the Pox controller. The lectures have extensive demonstrations that walk you through the Pox code to build a layer-two learning switch. The lectures also extend the learning switch to build a simple firewall. In the programming assignment, you will extend the example from the lectures to build a more fully featured network firewall.
- **Official 32-bit VM Image.** By popular demand, we have built a virtual machine image for Mininet that works on 32-bit processors. You can download the image [here](#).
- **Grading of first programming assignment.** The automated grading of the first programming assignment tests the output of your running code against a regular-expression matcher, using Coursera's regular expression engine. Unfortunately, that engine is not perfect, and we have been ironing out kinks with the grading of the output of that first assignment. It is possible that your test may fail if the latency values of the links in your topology are not within the range we are looking for. *Please submit your code regardless.* We are working on refining the grading script so that everyone receives the proper grade for the assignment, and we will send an update on this in the forums (and in next week's announcement).

Finally, we hope everyone is enjoying the course. We are finding it to be tremendously fun to teach this course. We are learning about SDNs and about MOOCs as we go! We appreciate all of the suggestions we are receiving in the forums, and we will incorporate as many as we can to keep improving the course. We are putting a lot of effort into the course to make sure that everyone can get something out of it. There are a broad range of people taking the course, and it is a pleasure to be teaching you. Thank you for all of your positive enthusiasm and helpful feedback.

Enjoy the nuts and bolts!

Your fearless instructor,
Nick

Upcoming Deadlines

Quizzes

Module 4 Assessment Quiz

Mon 22 Jul 2013 3:00 AM EDT (UTC -0400)

Module 5 Assessment Quiz

Mon 22 Jul 2013 3:00 AM EDT (UTC -0400)

Programming Assignments

Module 4 Programming Assignment

Mon 22 Jul 2013 3:00 AM EDT (UTC -0400)

New Lectures

Module 3.1 Virtualization [14:18]

Module 3.2: Applications of Virtual Networking [10:53]

Module 3.3: Network Virtualization with Mininet [10:53]

Module 3.4: Testing Mininet Setup [16:45]

Teemu Koponen Interview

Module 4.1: The Control Plane [15:37]

Module 4.2: SDN Controllers [9:59]

Module 4.3: Customizing SDN Control (Part 1: Switching) [16:20]

Module 4.4: Customizing SDN Control (Part 2: Firewalls) [10:28]

Guido Appenzeller Interview

Module 5.1: Programmable Data Planes [8:31]

Module 5.2: Scaling Programmable Data Planes:

Making Software Faster [13:15]

Help



Software Defined Networking

by Nick Feamster



Syllabus

Week 1

Week 2

Week 3

Week 4

Week 5

Week 6

Video Lectures

Quizzes

Programming Assignments

Discussions

Surveys

About Us

FAQ

Course Wiki

Join a Meetup

Help Articles

Flagged Content

Instructor Help Articles

I18N Editor

Video Lectures

Having trouble viewing lectures? Try changing players. Your current player format is html5. [Change to flash.](#)

Week 1

Overview (Part 1)



Overview (Part 2)



Module 1.1: Central Control [6:25]



Module 1.2: Active Networks [11:55]



Module 1.3: Network Virtualization [12:44]



Module 1.4: From FORCES to Ethane: Control Plane Evolution [9:19]



Martin Casado Interview



Week 2

Module 2.1: Control/Data Separation [6:15]



Module 2.2: Opportunities in Various Domains [13:27]



Module 2.3: Challenges in Separating the Data and Control Planes [13:27]



VirtualBox/Mininet Installation Tutorial



David Clark Interview



Week 3

Module 3.1 Virtualization [14:18]



Module 3.2: Applications of Virtual Networking [10:53]



Module 3.3: Network Virtualization with Mininet [10:53]



Module 3.4: Testing Mininet Setup [16:45]



Teemu Koponen Interview



Week 4

Module 4.1: The Control Plane [15:37]



Module 4.2: SDN Controllers [9:59]



Module 4.3: Customizing SDN Control (Part 1: Switching) [16:20]



What the Instructor Sees

coursera | Content Messaging Setup Grading Data Advanced Nick Feamster

 Software Defined Networking
by Nick Feamster

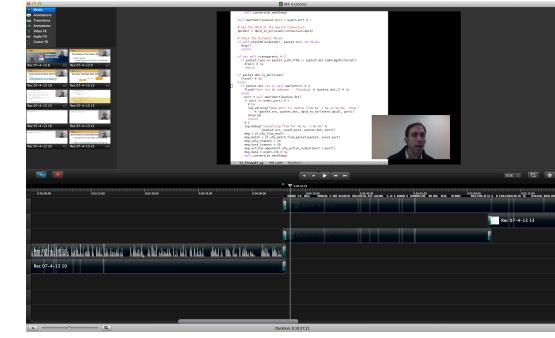
Create Section + Add ▾

		Status	Visible (UTC -4:00)	Due (UTC -4:00)	
Week1					
Overview (Part 1)		published	2 months ago	--	>
Pre-Assessment Quiz		published	2 months ago	16 days ago	>
Overview (Part 2)		published	2 months ago	--	>
Module 1.1: Central Control [6:25]		published	a month ago	--	>
Module 1.2: Active Networks [11:55]		published	a month ago	--	>
Module 1.3: Network Virtualization [12:44]		published	a month ago	--	>
Module 1.4: From FORCES to Ethane: Control Plane Evolution		published	a month ago	--	>
Module 1 Assessment Quiz		published	22 days ago	15 days ago	>
Martin Casado Interview		published	22 days ago	--	>
Week 2					
Module 2.1: Control/Data Separation [6:15]		published	15 days ago	--	>
Module 2.2: Opportunities in Various Domains [13:27]		published	15 days ago	--	>
Module 2.3: Challenges in Separating the Data and Control Plan		published	15 days ago	--	>
VirtualBox/Mininet Installation Tutorial		published	15 days ago	--	>
Module 2 Assessment Quiz		published	15 days ago	8 days ago	>
Module 2 Mininet Quiz		published	15 days ago	8 days ago	>
David Clark Interview		published	15 days ago	--	>
Week 3					
Module 3.1 Virtualization [14:18]		published	8 days ago	--	>
Module 3.2: Applications of Virtual Networking [10:53]		published	8 days ago	--	>
Module 3.3: Network Virtualization with Mininet [10:53]		published	8 days ago	--	>
Module 3.4: Testing Mininet Setup [16:45]		published	8 days ago	--	>

How Was the Course Developed?

- Course lesson plan, with learning objectives
- Top down design:
 - Figure out modules, continue subdividing until you get 10-minute “lectures”
- **Production**
 - **Lecture Filming and Production**
 - **Assignments**
 - Syllabus
- Operation
 - Forums

Filming: Camtasia



- Studio people will claim that quality suffers.
- I believe this is bogus, for several reasons.
 - People can take their time recording, take breaks, record when they are “in the zone”.
 - Screen captures from a laptop are easy.
 - Recording quality from a good laptop is quite fine.
 - Nobody has ever complained about the quality of the recordings. **Content is what matters most.**
- Takes a little getting used to at first (looking at the camera, etc.)
- You have to do your own editing.

Lectures: On Screen Demonstrations

- Camtasia: on-screen demonstration simple
- Walkthrough on slides + on-screen demo

```
File Edit Options Buffers Tools Python Help
#!/usr/bin/python

from mininet.topo import Topo
from mininet.net import Mininet

net = Mininet()

# Creating nodes in the network.

c0 = net.addController()
h0 = net.addHost('h0')
s0 = net.addSwitch('s0')
h1 = net.addHost('h1')

# Creating links between nodes in network (2-ways)
net.addLink(h0, s0)
net.addLink(h1, s0)

# Configuration of IP addresses in interfaces
h0.setIP('192.168.1.1', 24)
h1.setIP('192.168.1.2', 24)

net.start()
net.pingAll()
net.stop()
```

I



Examples of Things Shown in Demonstration

- Mininet and VM setup
- Mininet API
- Simple Pox application
- Pyretic applications
- Kinetic examples and applications

Assignments: Mininet

- 7 full assignments, almost all in Mininet
 - Mininet Python API
 - Simple controller applications (firewall, switch)
 - Virtualization (simple slicing in Pox)
 - Programming abstractions (Pyretic)
 - Verification (Kinetic)
 - Use cases (SDX)
- **Grading is automated**

Why Assignments Can Scale

- First of all, you do your best to work out the bugs, mostly out of fear 😊
 - We attempted everything that we assigned, sometimes on multiple platforms
- **More importantly: Self-selection!**
 - **Everyone doing the assignments wants to be there**
 - **This makes a huge difference**
 - **People not only fix issues themselves, they rewrite documentation**
- Caveat: Course difficulties vs. platform difficulties are currently hard to tease apart. Some of this is “user error”.

A student's blog post describing part of the assignment setup in detail...

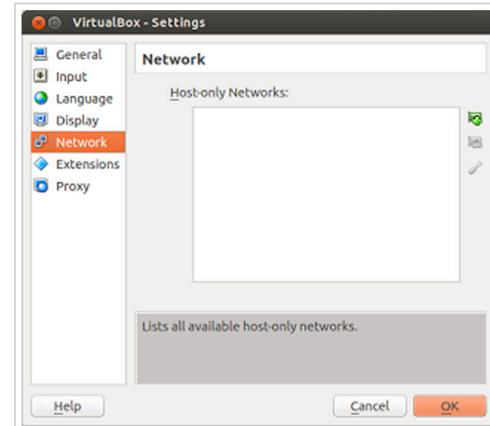
FRIDAY, JULY 5, 2013

VirtualBox Host-Only Networking

VirtualBox allows one to configure a VM with [host-only networking](#). This can be useful if you are connecting a number of VMs together and need to put them on the same switch/bridge.

However, it's darn frustrating to figure out how to enable it as all of the [googling](#) and [manuals](#) indicate you just enable it by selecting settings within the VM.

What they fail to mention though (but is covered in the built-in help in VirtualBox), is that you must first create a device for this host-only networking to use. From the main VirtualBox window, choose **Preferences** (below the **File**) menu. That will bring up the global preferences for the VirtualBox hypervisor. Click "Network" and then the "+" sign to add a host-only network device (typically vboxnet0).



Some Surprises

- The platform is “bare bones”
 - Wiki-style editing is not full-featured (e.g., no double-indented lists)
 - Regular expression matching does not work well
 - Responses almost always are “DIY”
 - UI is horrible... perhaps a good HCI project? ☺
- Copyright considerations appear to be different from normal classroom use (“fair use” may not apply)
 - People are making this up as they go along

Conclusions

- Mininet can scale to very large classes
- Auto-grading is critical at scale
- Some issues (performance of end hosts), but overall works pretty well
- Most issues arise when people do not use the course VM
- On-screen demonstrations are useful and highly watched