ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Information-Centric Networks

Section # 13.2: Alternatives

Instructor: George Xylomenos

Department: Informatics

European Union
European Social Fund

OPERATIONAL PROGRAMME
EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION, LIFELONG LEARNING AND RELIGIOUS AFFAIRS
MANAGING AUTHORITY

NSRF 2007-2013
programme for development
EUROPEAN SOCIAL FUND

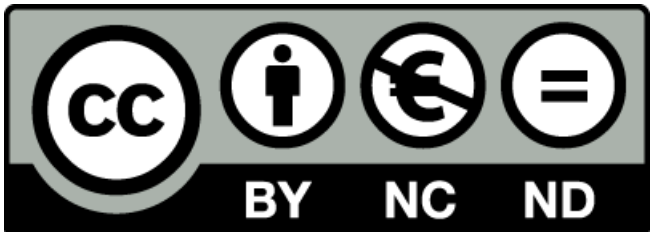
Co- financed by Greece and the European Union

Funding

- These educational materials have been developed as part of the instructors educational tasks.
- The **“Athens University of Economics and Business Open Courses”** project only funded the reformatting of these educational materials.
- The project is being implemented as part of the Operational Program “Instruction and Lifelong Learning” and is co-financed by the European Union (European Social Fund) and national funds.

Licencing

- These educational materials are subject to a Creative Commons License.



Week 13 / Paper 1

- OpenFlow: enabling innovation in campus networks
  - Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, Jonathan Turner
  - ACM SIGCOMM CCR, Volume 38, Issue 2, April 2008
- Main point
  - OpenFlow is a way to experiment with Ethernet switches
    - Adds a flow table to switches
    - Uses a standardized interface to manipulate that table
  - Allows cross-platform experimentation
  - Does not compromise vendor's intellectual property
  - Complements large-scale testbeds with switching intelligence

Need for programmable networks

- Networks have become to important to play with!
  - Vendors do not want to build “incompatible” switches
  - Operators do not want to experiment with production traffic
  - No way to try out new ideas in a realistic setting
  - Result: the infrastructure seems “ossified”
- Programmable testbeds
  - GENI in the US is a network built for experimentation
  - Virtualized switches allow many experiments to run
  - A very ambitious and costly plan that will take years to deploy
  - Can we at least run experiments in a campus network?
    - Need to persuade operators that they will not create problems
    - Need to isolate experimental traffic
    - Need to provide the right (which?) functionality

Using OpenFlow

- Assume that Amy invented Amy-OSPF to replace OSPF
  - Amy-OSPF runs in a controller connected to the switches
  - The traffic could start from a single machine
    - A wildcard action is used to capture packets from this machine
    - Each such packet is encapsulated and passed to the controller
  - Processing for each new flow
    - The controller creates a path through the switches
    - Flow table entries are inserted along the path
    - These entries are more specific than the generic one
- Approach tested in the Ethane prototype
  - A PC could process over 10.000 new flows per second
  - Dedicated hardware could do much better than that
  - A prototype controller called NOX is built as a follow-on

The OpenFlow switch

- OpenFlow-enabled switches
  - Normal switch with added OpenFlow capabilities
    - Flow table, protocol and secure channel
    - May reuse existing hardware for these
  - Must isolate production from experimental traffic
  - Two ways to isolate traffic
    - Use another action (forward packets normally)
    - Use VLANs to distinguish traffic
- Additional features
  - Type 0 switches are as described above
  - Extra features are possible
    - Header rewriting or matching non IP headers
  - Could be standardized in a Type 1 switch specification

Deploying OpenFlow switches

- OpenFlow consortium maintains specifications
  - Licensing is free even for commercial use
  - Products need to conform to Type 0 specifications
- Actual deployment
  - Many vendors are working on supporting OpenFlow
  - Stanford has deployed OpenFlow across some buildings
    - Different VLANs used to separate production traffic
    - Researchers can control their own traffic
- Reference platforms
  - Type 0 reference designs are available
  - Linux, NetFPGA and OpenWRT

Using OpenFlow

- Example 1: network management
  - The first packet of a flow triggers flow admission
- Example 2: VLANs
  - Packets are tagged with a VLAN id depending on flow entry
- Example 3: mobile wireless VoIP
  - The controller tracks clients and re-routes connections
- Example 4: non-IP networking
  - Identification via MAC type or IP version number
  - Ideally should use generic header matching
- Example 5: per packet processing
  - Process all packets at the controller
  - Divert packets to a NetFPGA device

Need for programmable networks

- Option 1: persuade vendors to offer programmability
  - Each vendor has a different, closed platform
  - No motivation to open their platform to competition
  - No assurance that regular operation will be unaffected
- Option 2: move to software platforms
  - Could be simple PCs with Linux, XORP, Click
  - Very low speed and port density compared to hardware
- Option 3: move to programmable hardware
  - Platforms with network processors exist but are too expensive
  - NetFPGA is cheap but too limited for production use
- Option 4: OpenFlow
  - Allow vendors to open a useful but limited part of their platforms

The OpenFlow switch

- Most switches contain flow-tables
  - Used for firewalls, NAT, QoS, statistics
  - Different per vendor but with a common set of functions
  - OpenFlow allows programming this common set
- OpenFlow switches
  - A flow table with an action per entry
    - At least a minimum set of actions must be supported
  - A secure channel to a controller to program the table
  - A protocol allowing the controller to talk to the switch
    - The only standardized part
  - Switches can be dedicated or just compatible with OpenFlow

The OpenFlow switch

- Dedicated OpenFlow switches
  - Simply follows actions from the flow table
  - A flow can be anything expressible in the protocol
    - Type 0 switches must understand 10 header fields
    - Port, VLAN, Ethernet SA/DA/Type, IP SA/DA/Protocol, TCP SP/DP
  - A basic set of actions
    - Forward packets to a port or ports
    - Encapsulate and forward packets to controller
    - Drop flow packets
  - Flows can be added dynamically by looking at their first packet
  - Flow table entries
    - Packet header describing the flow
    - Action to take on matching packets
    - Statistics counters

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

End of Section # 13.2

Course: Information-Centric Networks, **Section # 13.2: Alternatives**

Instructor: George Xylomenos, **Department:** Informatics

European Union
European Social Fund
OPERATIONAL PROGRAMME
EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION, LIFELONG LEARNING AND RELIGIOUS AFFAIRS
MANAGING AUTHORITY
NSRF 2007-2013
programme for development
EUROPEAN SOCIAL FUND
Co- financed by Greece and the European Union