Software Defined Networking

Dr. Nick Feamster
Professor

In this course, you will learn about software defined networking and how it is changing the way communications networks are managed, maintained, and secured.

Module 1: History of SDN

- *This lesson:* Active Networks
- What are active networks?
  - Motivation for active networks
  - Technologies behind active networks
- How do active networks relate to SDN?
- The legacy of active networks

Evolution of Supporting Technologies (Three Lessons)

- **Central network control:** Dates back (at least) to AT&T's network control point (1980s)
- **Programmability in networks:** Active networks (1990s)
- **Network virtualization:** Switchlets, XEN, VINI (1990s)

What are Active Networks?

- Networks where switches perform custom computations on packets
- Examples (and motivation)
  - Trace program running at each router
  - Middleboxes: firewalls, proxies, application services

Origins of Active Networks

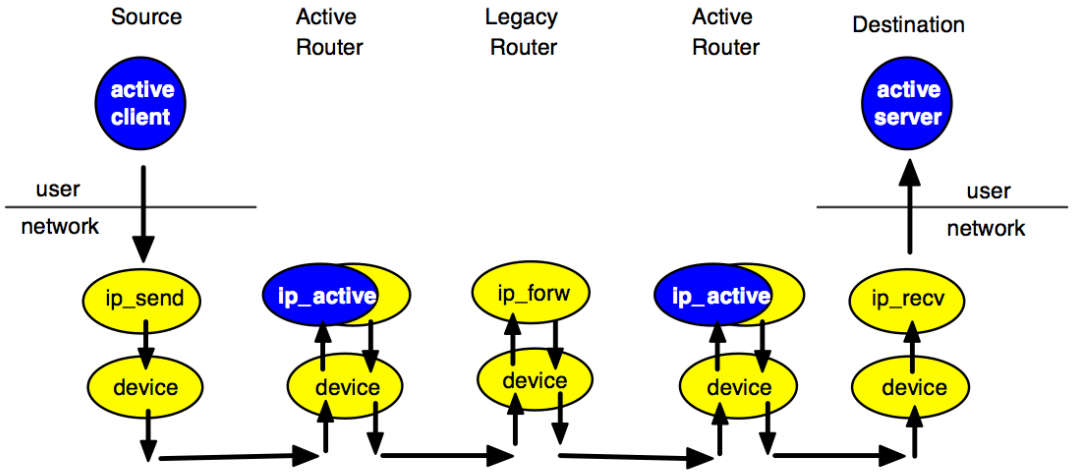
- DARPA research community (1994-1995)
- Identified problems with today's networks
  - Difficulty of integrating new technology
  - Poor performance due to redundant operations at several protocol layers
  - Difficulty accommodating new services

Motivation for Active Networks

- Accelerating innovation
  - Internet innovation relies on consensus
  - Takes ten years from prototype to deployment (standardization, procurement, deployment)

- **Active nodes** allow routers to download new services into the infrastructure
  - User-driven innovation

Idea: Messages Carry Procedures & Data



- Active routers coexist with legacy routers
- Each programmable switch can perform additional processing

Source: Tennenhouse, David L., et al. "A survey of active network research."*Communications Magazine, IEEE* 35.1 (1997): 80-86.

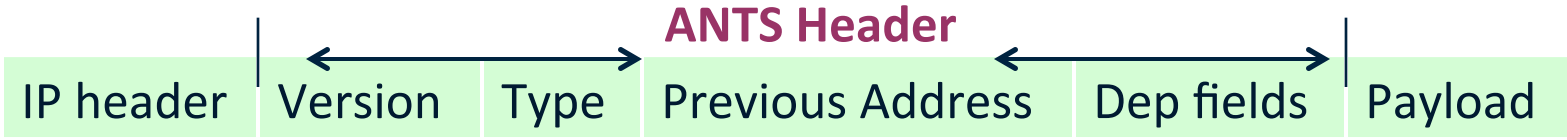
User "Pulls" and Technology "Push"

- User Pull (demand)
  - Proliferation of firewalls, proxies, transcoders, etc.
  - Goal: Replace ad hoc approaches
- Technology Push (enablers)
  - Safe execution of mobile code,Java applets
  - OS support
    - Scout: real-time communications
    - Exokernel: safe access to low-level resources
    - SPIN: trustworthy code generation

Two Different Approaches

- Capsules (“integrated”)
  - Every message is a program. Active nodes evaluate content carried in packets.
  - Code dispatched to execution environment
- Programmable Switches (“discrete”)
  - Custom processing functions run on the routers
  - Packets are routed through programmable nodes
  - Program depends on the packet header

Capsules (example)



- **Type**
  - Forwarding routine to be executed (carries code by reference)
- **Previous address**
  - Where to get the forwarding routine from if it is not available in the present node
- **Dependent Fields**
  - Parameters for the forwarding code
- **Payload**
  - Header + data of higher layers

Some Previous Notable Projects

- **ANTS (MIT):** Packet capsules (Java programs)
  - Some limitations for QoS guarantees. Arizona implemented Joust JVM to provide better real-time performance.
- **SwitchWare (Penn):** Programmable switch, scripting language to support invocation of switchlets
- **Smart Packets (BBN):** Network management
- **Open Signaling (Columbia):** NetScript, a language to provide programmable processing of packet streams.
- **Tempest (Cambridge):** Switchlets (more next time)

What happened?

- Timing was off
  - No clear application (pre-data center/cloud)
  - Hardware support wasn't cheap -- everyone was using ASICs, whereas now TCAMs, FPGAs, NPUs.
- Some missteps
  - Security, special languages for safe code, packets carrying code
  - End user as programmer (vs. network operator)
  - Interoperability
- **In contrast:** OpenFlow did a good job grappling with backwards compatible with switch hardware.
  - Simple firmware upgrade.
  - Switch hardware already supported the basics.

The Legacy of Active Networks for SDN

- Programmable functions in network to enable innovation
- Demultiplexing programs on packet headers
  - Planetlab, Flowvisor, GENI, etc. all use this
- Paying attention to middleboxes and how these functions are composed