CSE 534
Software Defined Networking

Vyas Sekar
Stony Brook University
Assigned Reading

• Ethane
  – Casado et al

• OpenFlow white paper
  – McKeown et al

• Like/Dislike?
Network Management

Traffic Engineering
Performance
Security
Compliance
Resilience
Problem: Toolbox is bad!

Traffic Engineering
Performance
Security
Compliance
Resilience
Why: Toolbox is implicit in routers!

- Traffic Engineering
- Performance
- Security
- Compliance
- Resilience
Reachability Example

- Enterprise with two remote offices
- Only A&B should be able to talk to server C
• Network designers add two links for robustness
• Configure routing protocols to use new links in failure
Reachability Example

- Designers apply packet filters to new links
Reachability Example

A

B

C

Permit A->C

Permit A->C

Permit A->C

Permit B->C

Internet
• Packet from B->C dropped!
• Testing under normal conditions won’t find this error!
Management today

- Data plane
  - Packet forwarding mechanisms

- Control plane
  - Routing protocols
  - Distributed

- Management plane
  - Has to reverse engineer what the control plane
  - Work around rather than work with!
Driving principles:
4D → Ethane → SDN

• Network-level objectives
  – High-level, not after-the-fact

• Network-wide views
  – Measurement/monitoring/diagnosis

• Direct control
  – No more “reverse engineering” or “inversion”
  – Direct configuration
Ethane:
Taking Control of the Enterprise
Motivation

• Enterprise configuration
  – Error prone: 60% of failures
  – Expensive: 80% of IT budget

• Existing solutions
  – Place middleboxes at chokepoints
  – Retrofit via Ethernet/IP mechanisms
Specific Problem: Access Control
Three principles in Ethane

• Descriptive/declarative policies
  – Tie it to names not locations/addresses

• Packet paths determined explicitly by policy

• Binding between packet and origin
  – No spoofing
  – Accountability
How Ethane Works

First packet sent to Controller
Subsequent packets use FlowTable

Amazingly simple and easy to describe!
Advantages of Ethane

• Switches
  – Dumb
  – No complex protocol
  – Simpler memory architecture
  – Focus purely on forwarding
Comments on Design

• Common vs worst case design?

• Latency, scalability

• False drops/positives
Some optimizations/constraints

• Only support exact matches

• Controller has to reverse paths

• Controller reliability
  – Cold, warm, hot
Drawbacks

• Support for broadcast is limited

• Overlays could still circumvent policy

• Using port numbers/matches is unreliable
Ethane Network Management

Motivation: Management is complex, expensive, fragile
Need: Direct control, Expressive policy, Network-wide views
OpenFlow

Controller

OpenFlow

Config

Config
Other work: NOX, ONIX
OpenFlow: Motivation

• Internet is a “success disaster”
  – Many successful applications
  – Critical for economy as a whole
  – Too huge a vested infrastructure
  – Vendors are loathe to change anything

• Fear in community: “ossification”
  – New ideas cannot get deployed
Driving questions

• Get operators comfortable with running experimental

• Isolate experimental from production

• What is the functionality that can enable innovation?
Rejected alternatives

• Get vendors to support

• Use PC/Linux based network elements

• Existing research prototypes for programmable elements
Their path

• “Pragmatic compromise”

• Sacrifice generality for
  – Performance
  – Cost
  – Vendor “buy-in”
Three Basic Features in OpenFlow

Controller

Secure Channel

Open Protocol

Config

Flow Table

Does this start to remind you of Ethane?
FlowTable Actions

• Forward on specific port/interface

• Forward to controller (encapsulated)

• Drop

• Forward legacy

• Future support: counters, modifiers
What is nice

• Fits well with the TCAM abstraction

• Most vendors already have this

• Can just expose this without exposing internals
Example Apps

- Ethane
- Amy’s own OSPF
- VLAN
- VoIP for Mobile
- Support for non-IP
Driving questions: Did it achieve this?

- Get operators comfortable with running experimental?
- Isolate experimental from production?
- What is the functionality that can enable innovation?
Emergent themes

• Abstractions for network management
• Enable innovation
• Logical centralization
• Incremental deployment