Teaching network softwarization with SDN Cockpit

Robert Bauer, Hauke Heseding, Addis Dittebrandt, Martina Zitterbart
Motivation

- **Focus:** Software Defined Networking (SDN)
  - Basic concepts are simple
  - Enables rapid prototyping
  - Easy to “get your hands dirty”

- “Use networks to teach about networks”
  - Learning-by-doing approach
  - Provide an authentic experience

- What is **SDN Cockpit**?
  - An open ecosystem for teaching network softwarization
  - Simplifies working with SDN
Traditional Networking

- **Distributed control plane**
  - Control elements in each network device
    - High implementation cost
    - Vendor-dependency
  - Complex protocols
    - Auto-configuration
    - Monitoring
    - ...

![Diagram showing control elements and complex protocols in traditional networking](image)
Software Defined Networking

- **Simplifies networking**
  - Control plane: centralized control functionality
  - Data plane: simple forwarding components
  - Apps: software-programmable network

**Focus:** core logic
- Routing
- Load-balancing
- ...

- SDN controller
  - Consistent global network view
  - Standardized interface

- Apps
  - API
Virtualization

- **Network infrastructure**
  - Virtual switches replace hardware (*Open vSwitch*)
  - Easily deployed in emulated topology (*Mininet*)

→ (Most) features remain available

- **Drawbacks**
  - Configuration
  - Specific API
  - Maintenance
  - ...
Learning Objectives

- **Scenario-based approach**
  1. Define learning objective
     - Demonstrate the effect of a networking mechanism
  2. Specify suitable topology
  3. Specify suitable traffic
  4. Specify expected behavior
  5. Implement and audit solution

  Requires suitable scenarios

- **Practical example**: forwarding
  1. Program flow rules
  2. Initialize *packet generator* and perform *packet capture*
  3. Evaluate packet trace

  Requires a sound technical understanding
Design Goals

- **Ease of use**
  - Provide virtual network
  - Automatic configuration
  - Automated maintenance

- **Integrated traffic scenarios**
  - Tailored to learning objectives
  - Cover a variety of scenarios

- **Automatic evaluation**
  - No prior training required
  - Consistent, immediate feedback
  - Does the application run properly?
Architecture of SDN Cockpit

- **Open ecosystem**
  - Utilizes standalone networking tools
  - Integrated by SDN Cockpit modules
Architectures of SDN Cockpit

Open ecosystem
- Utilizes standalone networking tools
- Integrated by SDN Cockpit modules
- Provides structured interfaces

SDN controller
Virtual network
Traffic generator
User interface
Feedback
Traffic analyzer
Monitor
Solutions

Backend

Topoogy
Traffic
Specify
Specify
Instructor

Integrated scenario

Frontend

Candidates
Edit
Feedback
Watch

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Candidate View

1. Program the App
2. Detect changes
3. Evaluate solution
4. Observe the effects

Editor
User interface

View into the controller, the virtualized network and the traffic generator

Teaching network softwarization with SDN Cockpit

Institute of Telematics
User Interface

Traffic generator

Mininet

Watchdog

Assignment

Task: Forwarding
Scenario: forwarding
Description: Network N1 is connected to the network N2 via switch S1. N1 sends IP packets to N2 with subnet 22.0.0.0/8. N2 sends packets in the reverse direction. Forward the correct packets to N1 and N2.
Result: SUCCESS

mininet> n1 ping -c 1 n2
PING 22.0.0.1 (22.0.0.1) 56(84) bytes of data.

--- 22.0.0.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

mininet> sudo

+ stop currently running controller
+ delete all flows
+ restart controller
+ waiting for controller to start
+ stop currently running scenario
+ restart scenario

Institute of Telematics

Teaching network softwarization with SDN Cockpit
Look and Feel

- **Objective**: flow programming
  - **Scenario**
    - Host A emits two IP flows
    - Directed at two networks
    - Connected by switch S
  - **Assignment**
    - Program flow rules for S
Automatic Traffic Evaluation

- **Input**: integrated scenario
  - Traffic profiles: number of packets, start time, duration, …
  - Topology: nodes, links, sender/receiver association, …
  → Split into individual flows

- **Evaluation**
  1. Generate one traffic generator configuration per flow
  2. Count expected packets per receiver over all flows (tally)
  3. Capture expected packets (filter expression) and compare to tally
Assignments

Core SDN
- Matches
- Actions
- Priorities
- Timeouts

Advanced SDN
- Flow table pipeline
- Group tables

Complex scenarios
- Monitoring
- Load balancing
- Policy-based routing
- Service functions
- DDoS mitigation

Teaching network softwarization with SDN Cockpit
Institute of Telematics

13
Our Experience

- SDN-Cockpit introduced at two occasions
  - Voluntary assignments during an advanced networking lecture
    - Winter semester 2017/18
    - 10+ candidates
  - Obligatory assignments in a practical course
    - Summer semester 2018
    - 14 candidates
Feedback – 2018 Practical Course

- **Easy to use**
  - 1 = Strongly disagree
  - 7 = Strongly agree

- **Simplified the learning process**
  - 1 = Strongly disagree
  - 7 = Strongly agree

- **Should be used in the future**
  - 1 = Strongly disagree
  - 7 = Strongly agree

- **Preferable to “bare” components**
  - 1 = Strongly disagree
  - 7 = Strongly agree

Teaching network softwarization with SDN Cockpit

Institute of Telematics
Future Work

- Extend to **additional audiences**
  - Network administrators
  - Cooperation: bwNET100G+ project

- Improve **accessibility**
  - Currently vagrant
  - Web-based interface

- Additional **assignments**
  - More advanced learning objectives
  - Wider range of scenarios
Summary

- **SDN Cockpit** is an open software ecosystem
  - Simplifies the learning process of SDN
  - Provides a variety of integrated scenarios
  - Provides automated feedback
  - Provides convenience for instructors and candidates

- Prototype is freely available
  - [https://github.com/kit-tm/sdn-cockpit](https://github.com/kit-tm/sdn-cockpit)
  - Use it, play with it, give us feedback …

Any questions?