Software Defined Networking (SDN)

Wireless application
Overview

SDN Background
SWDN
Mobile Application
Traditional Networks

Components:
- Routers
- Switches
- Firewalls, NAT, firewall, load balancers, etc
- Manual configuration, decentralized management

Challenges:
- Inertia: difficult to change and evolve network infrastructure and protocols due to the internet being society’s critical infrastructure
- Devices using proprietary software (being “closed”)
- Programmable networks
- Simplified network management
- Decoupling data and control
SDN - Not a New Concept

- **OPENSIG (1995)**
  - “making ATM, Internet and mobile networks more open, extensible, and programmable”
  - **Creation of General Switch Management IETF (GSMP)**
    - label switch
- **Active Networking (mid-1990s)**
  - Separation of control and data flows
  - Capsules - configuration messages
- **DCAN (mid-1990s)**
  - Management function of many device should be decoupled from the devices
  - Minimalist protocol for communication between manager and network
- **4D Project (2004)**
  - Clean Slate Design
  - Separation of routing decisions
- **NETCONF (2006)**
- **Ethane (2006)**
  - ethane switch for flow table and secure channel to the controller
- Hardware Forwarding decoupled from (routing decisions)
  - Centralized Network intelligence (control plane)
  - Network device becomes simple (data plane)
- Architectures
  - ForCES and Openflow
OpenFlow

- Standardized Message exchange between two planes
- **Centralized control**
- Eliminating middle boxes and third-party “apps”
- Commoditizing network hardware
Forwarding Decision

1) Upon a packet arrival at an OpenFlow switch, packet header is check
2) If it fits an existing rule, execute the corresponding action
3) If it is a table-miss, consult the controller
4) Routes are then pushed out to the switch and added to table

- Granularity of these routing decisions are set at per flow basis. Why?
Forwarding rules

- Ternary Content- Addressable Memory (TCAM)
  - expensive and power-hungry
  - few thousand up to tens of thousands forwarding rules

- Devoflow
  - mice flow at switch, elephant flows handle by controller
  - 10 to 53 times less flow table space.

- DIFANE
  - Authority switch with full forwarding rules
  - Ingress Switch - caches rules

- Palette & One Big Switch
  - Using other policies and network metrics to partition forwarding table across network
Controller

- FloodLight, OpenDaylight, Ryu, etc.
- Scalable?
  - Handles 1.6 million new flow requests per second with an average response time of 2 milliseconds on 8 cores
  - Multiple Controllers for fault tolerance
    - Control Model
      - Onix and HyperFlow - Logically centralized by distributed control
      - Kandoo can utilize local controllers for local applications and redirect to a global for other traffic
Policy updates:
- **Reactive**
  - First packet delay
- **Proactive**
  - DIFANE

Control Plane
- Northbound - Controller Service
  - extract information about the network
  - no standards for interaction
- Southbound - Controller-Switch
  - Security (1.3.0 - Transport Layer Security TLS)
SDN APPLICATION

- **Enterprise**
  - Enforcing policy and allocate resource dynamically
  - Monitor and tune network on the fly

- **Data Centers**
  - Increase efficiency, power saving

- **Homes and Small Business**

- **Optical networks**
  - Packet/Circuit switch via Wavelength Selective Switching (WSS)

- **Infrastructure-based wireless access networks**
  - Performance handoffs
SDWN - Unbridling SDNs

Application of SDN on Infrastructure-less networks
- 802.15.4-based
- low rate, wireless personal area networks

Remember:
“There is nothing you can do with SDN which cannot be done without.”
Heterogeneity

Higher level 802.15.4 players
- ZigBee
- 6LOWPAN

Problem: The two are compatible with each other.

Solution: Why not use SDN?
SDWN Requirements

**Traditional networks** - line rate is needed => limits flexibility

**LR-WPANs** - low rate, but low power is a **must**

What is required of SDN as a solution?
- support for duty cycles - reduce energy
- Support in-network data aggregation - reduce redundancy/overhead
- flexible definition of the rules
- Support for node mobility and the resulting topology changes
SDWN - Sinks

- network will have **sink(s)** where the network **Controller** is located
  - running embedded OS
  - micro-control unit
  - collects information to keep track of network topology
SDWN - Sinks

Elements:
- Network Operating System (NOS)
- Adaptation layer
- Virtualizer
- Controller

Sink periodically generates a beacon packet
- **RSSI** is the relative received signal strength is stored for the sender of the packet, allows to figure out most efficient path
  - choose the node that declares the lowest distance from the sink
  - the node with the longest battery life
  - alternatively, the node toward which the highest RSSI value was measured
Nodes

Network Operating System (NOS)
- Aggregation Layer
- Forwarding layer
  - maintains flow table

Task of these Nodes
- Forward beacons
  - Increment hops
  - Log Received Signal Strength Info.(RSSI)
  - replace Battery Level
- Neighbor table
- Report packets
Rules

Strict limitations in terms of available memory

**Exemplary Flow Table**

<table>
<thead>
<tr>
<th>Window Size</th>
<th>Window Op</th>
<th>Window Addr</th>
<th>Window Value</th>
<th>Window Size</th>
<th>Window Op</th>
<th>Window Addr</th>
<th>Window Value</th>
<th>Action Type</th>
<th>Action Value</th>
<th>Stats Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 = 2</td>
<td>170.24</td>
<td>2</td>
<td>4</td>
<td>170.11</td>
<td>..</td>
<td>Forward</td>
<td>170.23</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 = 2</td>
<td>170.16</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>..</td>
<td>Drop</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 ≠ 2</td>
<td>170.24</td>
<td>1</td>
<td>7</td>
<td>25</td>
<td>..</td>
<td>Modify</td>
<td>7/26</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 = 2</td>
<td>170.17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>..</td>
<td>Forward</td>
<td>170.21</td>
<td>11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Packet Types

5 types:
1. Data Packet
2. Beacon packets
3. Report Packets
4. Rule/Action Request
5. Rule/Action response
Extremely Dense Wireless Network

Increase in networked devices:
- backhaul capacity, the increased energy consumption, and the explosion of signalling

Need to:
- tame extreme density of wireless networks.
  - accounting for MAC control and Mobility Management has been proposed
What we need

- Interference and mobility management in wireless dense networks
- Rate distribution (user-perceived, i.e., accounting for load)
- Area spectral efficiency
- IP Mobility

Proposed:

- Connectivity management for eneRgy Optimised Wireless Dense networks (CROWD)
Push for

- flexibility and reconfigurability,
- an energy-efficient network infrastructure
  - radio access part
  - the backhaul

**SDN**

Provide controllers are:

- technology-agnostic
- vendor-independent
Architecture

CROWD Regional Controller (CRC)
- Centralized, long term optimization
- only requires aggregate data
- charge of dynamic deployment of CLC

CROWD Local Controller (CLC),
- short-term optimisation
- instantaneous data from network
- Covers districts
- Optimization at the edge
High Level Business

Provides ability to buy control application

Policies are passed into a policy translation then a dispatcher install configuration on CRCs

**CRC** dynamically partitions network into districts, 1 CLC per district

Fig. 2. High-level business process.
Alternatives - Existing Methods

Operations Support Systems (OSSs)
- Business Support System (BSS) handling billing data, near-real-time services for fault man. using Network Management System (NMS)

Mobile Cloud network (MCN)
- Support for efficient and elastic network resource, provision and optimized based on demand

Self Optimising Network (SON):
- Autonomic" execution of configuration/optimization procedures base station of wireless networks
Mobility - Joining

1) node connect
2) CLC check local binding cache
3a) if not found, contact CRC to check if register with different district
   - inter-district mobility may be required
3b) CLC can assign any available ipv6 prefixes on the district and DMM_GW
4) After attachment, SLAAC is used for IP assignment
Mobility - Handover

3a) CRC has information regarding the terminal, informing the CLC of the prior connection

4) with the information CLC2 decides the DMM-GW to use

5) CRC tells first CLC1 new location of terminal of the node

6) CLC1 sets up ip-in-ip tunnel to new gateway and changes the route to be routed through that tunnel

7) new CLC does the same on the other end to set up bidirectional communication
Questions?

Thank you