OpenFlow-based Content-Centric Networking Architecture and Router Implementation

Atsushi Ooka
Osaka University
Japan

Outline

- Research area
  - Content-Centric Networking (CCN)
- Objectives
  - Resolve Implementation and Deployment Issues
- Research Approach
  - OpenFlow-based CCN
- Major Outcomes
- Conclusion

Objective and Approach

- Objective: Realization of CCN
  - Implementation of CCN forwarding
  - Deployment of CCN
- Approach: Implement OpenFlow-based CCN
  - Allows to easily implement new protocols
  - Allows multiple systems to share one network

Content-Centric Networking (CCN)

- Content-based addressing with "name"
- Transmission using Interest and Data packets
- Natively supported mechanisms to disseminate contents
  - Interest aggregation, data multicasting, In-network caching

Realization of CCN is a big challenge

Design of OpenFlow-based CCN

- Fast Processing based on name address
  - Convert name to hash value (that is hierarchically structured)
  - Write the hash in IPv4 address field
- Multicast using multiple actions
  - DST IPv4 Addr = 18.52.0.0/16
  - Packet-out port 2
  - Packet-out port 3

Major Outcomes

- Hierarchically structured hash value
  - Support active name
- Analysis of hash collision probability
  - Explore the trade-off
- Implementation of the program for OpenFlow controller
  - Show the demonstration
Hierarchically Hashed Name

- Assign 4 bits to each component
  - Ex. /text/A.txt/v1/s1
    - "text" = 2, "A.txt" = 1, "v1" = 4, "s1" = 2
    - IP address: 33.66.0.0/16 (= 0x21420000)

- Trade-off between B and C
  - B: number of bits assigned to component
  - C: number of components that can be hashed

<table>
<thead>
<tr>
<th>Protocol</th>
<th>B = 4</th>
<th>B = 8</th>
<th>B = 16</th>
<th>B = 32</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 (32 bit)</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAC (48 bit)</td>
<td>12</td>
<td>6</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>IPv6 (128 bit)</td>
<td>32</td>
<td>16</td>
<td>8</td>
<td>4</td>
</tr>
</tbody>
</table>

Support Active Name

- Resolve a mismatch caused by flatly hashing\[1\]

**Program Demonstration**

**Conclusion and Future Work**

- **Conclusion**
  - Feasibility of OpenFlow-based CCN
    - Provides the implementation of CCN forwarding on OpenFlow
    - Support active name
    - Analyze hash collision probability

- **Future work**
  - Implement lacking mechanisms
    - Routing, hardware cashing
  - Explore scalability
    - Hash collision probability, flow entries
  - Evaluate performance using hardware