A Native Content Discovery Mechanism for NDN

Onur Ascigil, Vasilis Sourlas, Ioannis Psaras, and George Pavlou
Department of Electronic and Electrical Engineering,
University College London, UK
Outline

• Content discovery
• Opportunistic Off-path Content Discovery
• Forwarding Strategies
• Results
• Future Work & Conclusions
Content Discovery
Content Discovery

• Goal: Retrieve a nearby (ideally the nearest!) copy of the content
  • Difficult to achieve without significant “overhead” in practice

• Why?
  • Placement of Data into the Content Stores happens frequently

• What does NDN/CCN do?
  • Route Interests to content origins
  • Search content opportunistically on-path (i.e., along the default path)

• Existing Solutions for Content Discovery:
  • Opportunistic on-path
  • Coordinated off-path
Content Discovery

• **Opportunistic on-path:** *limited gain, without overhead*
• **Coordinated off-path:** *coordination and communication overhead*
  • *Using control plane:* Advertise content names
  • *Using a function:* A Hash function determines the placement and routing
  • ...

• **What do we propose?**
  • Integrate an “opportunistic off-path content discovery mechanism” to the existing Interest/Data processing pipeline of NDN
    • With minimal changes to NDN packet processing
    • Without introducing excessive overhead
Opportunistic Off-path Content Discovery
Opportunistic off-path Content Discovery

- **Satisfied Interest Table (SIT):** Caches trails of Data packets

```
<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next-hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ndn</td>
<td>T</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Next-hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ndn/xyz</td>
<td>S</td>
</tr>
<tr>
<td>/ndn/xyz</td>
<td>T</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next-hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ndn</td>
<td>T</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next-hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ndn/xyz</td>
<td>U</td>
</tr>
</tbody>
</table>
```

Data /ndn/xyz
Forwarding Strategies with FIB & SIT
Forwarding Strategies with FIB & SIT

• Breadcrumb
Forwarding Strategies with FIB & SIT

- Breadcrumb
Forwarding Strategies with FIB & SIT

- **Multicast**
Forwarding Strategies with FIB & SIT

- **Multicast**: Once forwarded downstream, an Interest follows a single SIT trail
- **Forwarding Strategy**: Pick the freshest matching SIT entry
Forwarding Strategies with FIB & SIT

**Multicast**: Once forwarded downstream, an Interest follows a single SIT trail

**Forwarding Strategy**: Pick the freshest matching SIT entry

![Diagram of network with FIB and SIT entries](image-url)
Results - Multicast and Breadcrumb Strategies
Results- Settings

- **Topology**: Rocketfuel ISP topology
- **Content Providers**: Attached to 16 egress nodes are randomly chosen
- **Link Latencies**: Penalty for leaving ISP network: extra **50-100 msec**
- **Network Cache Capacity**: %80 of content can be cached in the network
- **Request Rate**: 100 Requests/sec (origins selected randomly)
- **Popularity Distribution**: Zipf Parameter 0.7
- **Experiment Duration**: 1 Hour warm-up and 3 hours of experiment
Results with Breadcrumb & Multicast

Overlap

Average Hit Rate

Average latency (msec)

Strategy

Latency

Breadcrumb
Multicast
NDN
• **Modified Multicast Strategy:**
  • Each Interest is associated with a *Forwarding Budget*
    • spend the budget on:
      • sending a copy upstream (following FIB)
      • sending a copy downstream (following SIT)
    • spend it on both

• **Cost** of sending Interest upstream/downstream
  • *Static*: Deduct one unit per each copy sent downstream.
  • *Dynamic*: Deduct variable amount per each copy sent downstream
Multicast with Forwarding Budget (Static)

<table>
<thead>
<tr>
<th>Name</th>
<th>Next-hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>/x/y/z</td>
<td>Y</td>
</tr>
</tbody>
</table>

Interest: /x/y/z
Budget = 6

Interest: /x/y/z
Budget = 5

OFF-PATH

Interest: /x/y/z
Results - Multicast with Forwarding Budget
Results - Multicast with Forwarding Budget

Satisfaction

Overlap

Average Hit Rate

Both
Cache Hit Only
Server Hit Only

Dynamic Cost
Extra Budget (for Off-Path)
Static Cost
Conclusions

• Opportunistic content discovery using **SIT table**
  • Significant increase in the percentage of requests satisfied from the caches

• Forwarding strategies using SIT/FIB are introduced
  • Possible to limit the overhead with a Forwarding Budget and Dynamic Costs
  • Requires minimal changes to the packet processing of NDN

• Security
  • End-user/host caches are not exploited by the mechanism
  • *Exclude* field can be used to ignore matching SIT entries
  • Our scheme inherits the existing cache poisoning problems with NDN
Thank you!

Questions?