Content-Centric Networking

Ihsan Ayyub Qazi

LUMS SSE

Slides use info from Srini Seshan, Van Jacobson
Why CCN?

- Problem in 1960s and 1970s – resource sharing
  - Lead to the host-to-host communication model
- Current use focuses on content
  - Think of Google
  - But communication is still in terms of ‘where’
    - DNS: Names → location
- Why this mismatch between the usage and communication models raises issues?
Example: Watching a Live Football Match
What should it look like?
Some Issues

• Availability
  – Fast, reliable access requires CDNs and P2P networks, and/or imposes bandwidth costs

• Security
  – Based on location and connection information rather than the content itself

• Location-dependence
  – Mapping content to host location complicates configuration (e.g., DNS)
CCN Model

• Name **content** rather than **hosts**
  – No source and destination addresses
• Routing based on (hierarchical) names
CCN Model

- Two packet types: **Interest** and **Data**
- Hierarchical content naming scheme
  - Allows dynamic content generation
  - CCN node has 3 components
  - **Forwarding Information Based (FIB)**
    - Allows multiple output faces
- **Content Store**
  - Buffer, also caches Data packets
- **Pending Interest Table (PIT)**
CCN Node

• Processing an Interest (In order of priority):
  – Matching Data is found in the Content Store
    => send it and consume Interest
  – Pending Interest in PIT
    => add this face to RequestingFaces list
  – Use FIB to forward Interest on outgoing faces, add to PIT
Processing of data packets

• What about data processing?
  – Is data routed in CCN?
    • Nope, follows the chain of PIT entries
  – How about route pinning?
    • What if the reverse path is not the best one?

• When a data packet arrives
  – Longest-match lookup is done on its content name
    • A match means?
      – data is a duplicate so it is discarded
    • What does a FIB match mean?
      – No matching PIT entries (ordering)
        – Unsolicited data so it is discarded – Attacks?
    • If PIT match, it is added to the Content Store and sent to all request faces
CCN Node Model
CCN Node Model
Transport

- **One** Interest packet $\rightarrow$ **one** data packet
- Interests serve the role of window advertisements
- Each packet is independent $\Rightarrow$ TCP SACK is implicit
- Flow balance is maintained at each hop, not end-to-end like TCP
  - Since all transfers are done hop-by-hop – so no need for congestion control
CCN Names

- **Hierarchical** content names with a flexible format
- Individual name consists of a number of components
- Names can be relative to some known name, e.g. next/previous
  - Same content can have multiple names! Problems with caching?
- A source of data performs a Register operation for a prefix
CCN Names

• Like IP, CCN imposes no semantics on names

• ‘Meaning’ comes from application, institution and global conventions:

  /parc.com/people/van/presentations/CCN
  /parc.com/people/van/calendar/freeTimeForMeeting
  /thisRoom/projector
  /thisMeeting/documents
  /nearBy/available/parking
  /thisHouse/demandReduction/2KW
Names Route Interests

• FIB lookups are longest match (like IP prefix lookups) which helps guarantee log(n) state scaling for globally accessible data

• Although CCN names are longer than IP identifiers, their explicit structure allows lookups as efficient as IP’s

• Since nothing can loop, state can be approximate (e.g., bloom filters)
Routing

• Routing between CCN nodes can occur over unmodified OSPF
  – Incremental deployment of CCN nodes is possible
• Integration with BGP is also possible
• Routers do not construct spanning trees
  – Loops are not possible anyway
  – Multiple paths can be used
Content-based Security

- Security travels with the content, it is not a property of the connection
- CCN authenticates name-content bindings by signing the name and content in each data packet
- Arbitrary key management schemes can be used over CCN
- Keys can be sent over CCN since they are just another piece of data
- If we trust some public keys, we can infer more
Network Security

• Sending a malicious packet to a host is difficult because CCN talks only about content, not to hosts

• Data based DoS attacks are impossible because only one Data packet is forwarded per Interest

• Interest flooding:
  – Multiple Interests for the same content are combined
  – Limit the forwarding of unsuccessful interests

• What if sender and receiver collude?
Evaluation

Transfer time vs Number of Sinks
What about VoIP?

• Key challenge - rendezvous

• Need to support requesting ability to request content that has not yet been published

• E.g., route request to potential publishers, and have them create the desired content in response
Evaluation

Failover
Questions

• Do we need QoS in CCN?
  – How do we select best paths and sources?
  – Prioritizing interest packets?
• Is supporting host-to-host apps hard?
  – How do other usage patterns fit into this picture?
• What are the implications of path pinning?
• How do you discover higher-level names and not necessarily content?
• The role of congestion control needs more investigation
• How ISPs do charging? How do advertisers place content?
• Is the overhead of interest packet excessive?
• Is the overhead of CCN significant?
• How about integrating processing in addition to storage?
Thank you!