Automated Deployment and Scaling of Named Data Networks in Cloud Environments

Sean Liao
Supervisor: Zhiming Zhao
Next 20 mins...

Some academic(s) arrive to tell us that (once again) they have Fixed the Internet, and (once again) it runs on top of the current actually-working internet, and (once again) if you sign up you can communicate with as many as twelve other computers.

n-gate.com in reference to SCION
Introduction
ENVRI-FAIR

ENVIronmental Research Infrastructures (ENVRI)

connection to

European Open Science Cloud (EOSC)

https://envri.eu/about-envri-fair/
Data distribution
Content Distribution Network
Federated Clouds
Named Data Networking (NDN)
Named Data Networking

NSF Future Internet Architecture Program

Information distribution network

Potential benefits:

- content caching
- network level security of data
1.2.3.4/a/b/c/d

/a/a/b/b
  hello
  /a/b/c/d
world

/a/a/b/c
  foo
  /a/b/c/d
bar
Route directly to data

IPv4: 1.2.3.4
NDN: /arbitrary/strings/infinite/address/space
1-1 Request-Response (Interest-Data)
In network caching
Run it
In the Cloud

Overlay over IP
Simplify deployment
Scalable

Existing Tools

Router: NFD
Link State Routing: NLSR
Dynamic route updates on static network
Connectivity: FCH
Find closest hub/gateway
Overlay over TCP / UDP

Roundtrip Time

Each point is average over 10K requests

Time (milliseconds)

Packet Content Size (bytes)
Processing Overhead

Hop Processing Time

Subtract network roundtrip time: 0.174ms
Testing notes

Performance severely degrades with 50000+ cached / in-flight requests
Scaling Up
Problem: Growing a node
Replace

TCP/UDP Load Balancer / Broadcast / Multicast
New Node in Front

TCP/UDP Load Balancer / Broadcast / Multicast
New Node Behind (Chain)
New Node Behind (Load Balancing)
Growing the Network
Problem: New Node
Broadcast / Multicast
Bootstrap Gossip
Central Discovery Server
Network Architecture
3 Layer Cache
Mesh Network
Automation
In-band Management
Out of Band Management
Proof of Concept
Necessary Configuration

Discovery Server
- Write down its address / Give it a preconfigured address

Load Balancer
- Address of Discovery Server
- Write down its address / Give it a preconfigured address

Caching Server
- Address of Load Balancer
Proof of Concept Performance

Scraping state of router through CLI (on a timer)
TCP connections propagate routes and updates
Coarse grained partitioning of routes
Conclusion
Does it work?

Reuse existing router in a load balancing configuration

Minimal configuration, self connecting network
Lessons Learned

Now

Rapidly evolving research testbed for new ideas

Things break or are not optimized

Future

Some new network based on these ideas

Or maybe not (IPv6…)

Demo: add load balancer (40s)
Demo: remove load balancer (16s)
Demo: add Cache (44s)
Demo: remove Cache (17s)