Introduction to Named Function Networling

Christopher Scherb

University of Basel
Department for Mathematics and Computer Science
Computer Networking Group

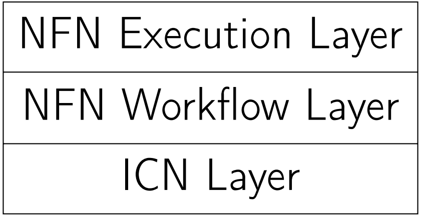
Internet of Things Hackathon 2017

SCHLOSS DAGSTUHL
Leibniz-Zentrum für Informatik

Introduction

- Information Centric Networking (ICN) connects to static data
- Content and Location are independent
- Special case of dynamic data/results
- Computation and Location independent?
- Related work: EU-Project RIFE

Layer Model



NFN Execution Layer

NFN Workflow Layer

ICN Layer

NFN-Workflow Layer

From Data To Computation

- In ICN:
  - Names in Interest Message
  - Names in the FIB
  - Forwarding: Longest Prefix Matching
- In NFN:
  - Computation in Interest Message
  - Names in the FIB
  - Forwarding: ?

λ-calculus

- Encode computations as names: λ-calculus
- Formal language system to describe computations
- three operators:
  - variables: n
  - abstraction: $(\lambda x.x)n$
  - application: $m\ n$
- Touring complete
- Can express any functional computation

Workflow Definitions

- How to translate this to NFN?
  - Store functions inside of content objects
  - Introduce "call" extension to invoke execution layer
  - build a workflow by combining call expressions
  - `call /funcname /dataname + call /funcname ...`
- How to route this expression?
  - use λ-abstraction
  - (λ`x.call /func x) /dataname`
  - prepend the data name:
  - `/dataname` (λ`x.call /func x)`

Routing

- encode prepended name in components
- encode computation in one component
- add NFN marker
- use longest prefix matching

```
data|name|comps |(λx.call /func/name x)|NFN
```

Name Manipulation

- Krivine abstract machine
  - Resolves λ expressions
  - Call By Need semantics
  - Request data when required
- Which name to prepend?
  - Hadoop: Move function to data
  - NFN: flexible scheme

Game of Names

- functions can be pinned to nodes
  - routing to data not always sufficient
  - e.g. for security reasons, private key required etc.
- `call /f /d`

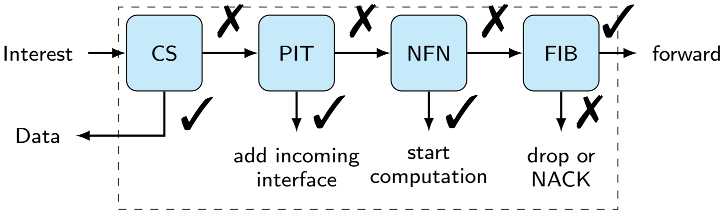
1. route to data: `/d` λ`x.call /f x`
   - prepend data name
   - try all available data names
2. route to function : `/f` λ`x.call x /d1`
   - prepend function name

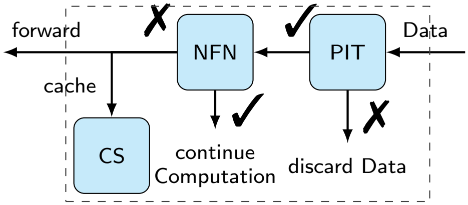
NFN-Execution Layer

Named Function

- high level functions
- manipulate data
- side effect free!
- executed on a single node
- can issue further computations
- cache and reuse results

Entire Workflow





Extensions

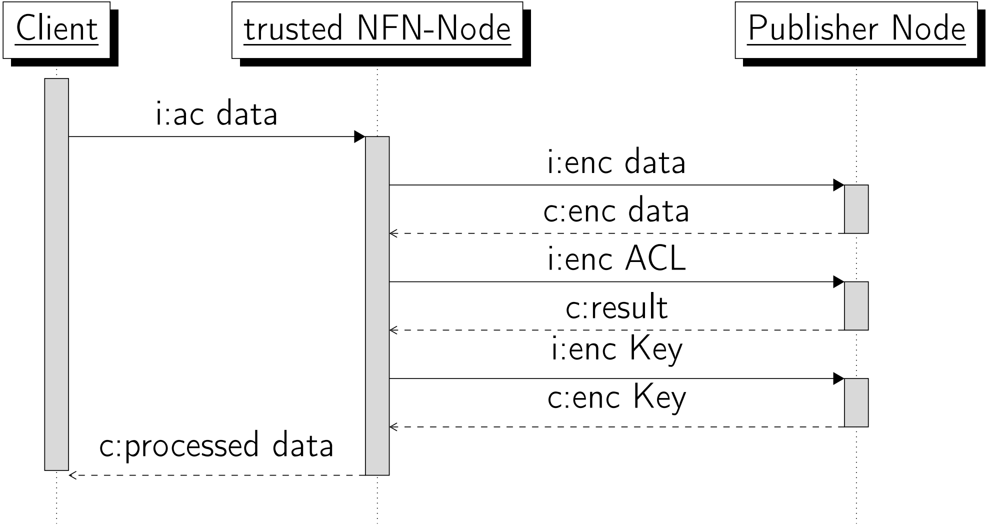
Secure Distribution of Computations

- Input Data are not public available
- Idea: Reduce information according to access level of requesting users
- NDNHealth Project: GPS Tracking data for a Study
  - Insurance level: can only access distance
  - Study level: access GPS track, but move it to (0,0)
  - Personal level: access all data

Secure Distribution of Computations 2

- Symmetrically encrypted data
- Asymmetrically encrypted Keys
- Encrypted Access Control lists
- Trusted execution nodes
- Encrypted results
- Chaining of computations
- Caching vs. Perfect Forward Secrecy

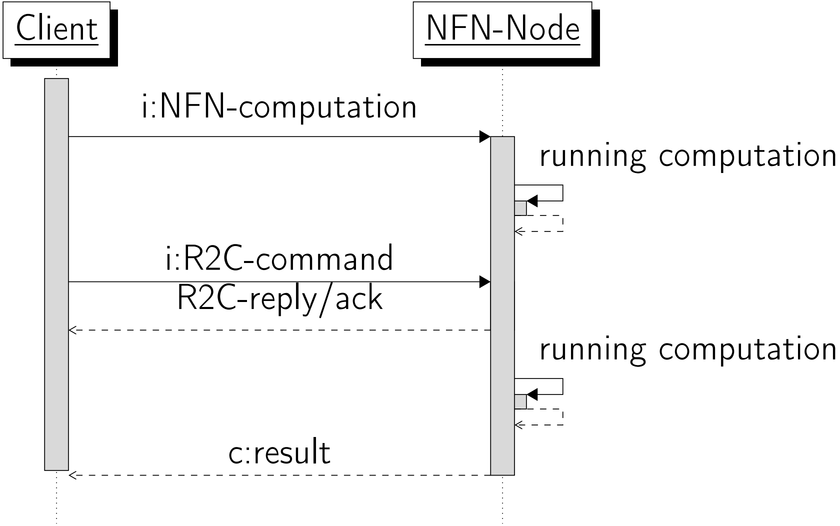
Secure Distribution - Workflow



Request-2-Computation

- instructions for a running computation
- stop, pause, resume or update a computation
- timeout prevention with keep-alive
- use name of computation + R2C Marker
- enable long running computations

Request-2-Computation - Workflow



Work in Process

NFN Push

- In ICN push requires callback
- NFN-workflow could contain push notification
- Reqirements:
    - Clients have a NFN-execution unit
    - Clients can be named in some way

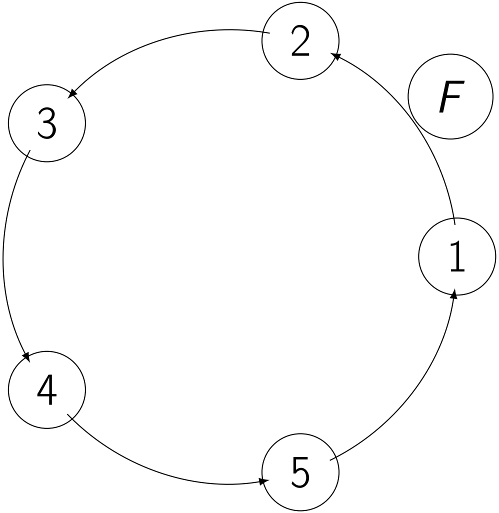
Optimizing Execution Location

- Focus: Data center and HPC
- Best execution location not always where data are stored
- Move data and results of (sub)computations to a location
  - where execution cost are minimal according to a metric
  - metric:
    - network load
    - power consumption
    - execution time

Traveling function / Function Migration

- function executed on several nodes.
- function carries execution state to next node
- collecting and processing data
- how to encode this scenario in a name?
- how to update the PIT state when computation moves?
- parallel execution –> dynamic map reduce
- function move parallel to a vehicle on road-side-units?

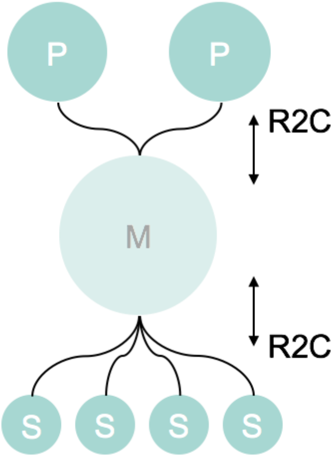
Traveling function



NFN and IoT

Universität Basel

- Computation on low power nodes?
- On Gateways?
- Add few computation nodes, collecting data?
- PubSub Model?



The End

Thank your for your attention!