Keyword-Based Mobile Application Sharing through Information-Centric Connectivity

Dr Ioannis Psaras
EPSRC Fellow / Lecturer
University College London
Email: i.psaras@ucl.ac.uk
Web: http://www.ee.ucl.ac.uk/~ipsaras/

Best Paper Award
ACM MobiArch 2016

ICNRG Interim Meeting
16th July 2017
Prague

I. Psaras, S. Rene, K.V. Katsaros, V. Sourlas, N. Bezirgiannidis, S. Diamantopoulos, I. Komnios, V. Tsaoussidis, G. Pavlou
The Cloud is not enough

<rant>

• Always trying to reach the cloud does not work
  – E.g. trains, airplanes, crowded areas

• 5G needs to integrate some edge-computing functionality

• The cloud is neither the only nor the best way

• There are enormous amounts of computation and storage available around us
  – 5G has to exploit the computation, storage and software resources of edge devices (smartphones, tablets, Raspberry PIs, WiFi APs)

• Connecting randomly to the nearest device does not work
  – Information-Centric Connectivity becomes necessity when we need to specify to which of the 100s of smartphone devices to connect to.
  – This need does not exist when we always connect to the main cell tower
</rant>
What is KEBAPP – Contribution

An application sharing and information-processing framework for smartphone apps

Route Finder App

Game or Video-Streaming Server

Super Boring!
What applications does KEBAPP deal with – Design Space

• By and large, smartphone apps target:
  – Static content, e.g., news updates
  – Personalised content, e.g., Facebook/Twitter updates
  – Processed information, e.g., route finder, gaming
    • Keep demand for local services, locally!

We envision a pool of application resources to provide D2D access to processed and non-personalised information
Where/When do we need KEBAPP (Target environments)

- Overcrowded areas
  - Airports, festivals, stadiums, IETF :)
- Fragmented networks
  - Natural disasters (floods, earthquakes)
- Not (or poorly) connected environments
  - Airplanes, trains, ferries, developing regions

In most of those cases, Internet connectivity is not even necessary!
How does KEBAPP work?

Applications act both as clients and as servers

Three Main Components

1) Application-centric naming
   - Applications share common name-spaces and support the use of keywords

2) Application-centric connectivity
   - Applications manage connectivity by defining and/or joining WiFi broadcast domains

3) Information-centric forwarding
   - Extending Named Data Networking primitives
Information-Centric Mobility

• Content is the addressable entity
  – Not the host!

• Content is the routing target
  – Not the host!

• Interface to the content is used
  – Not to a socket!

• Content is secured individually
  – Not the communication channel!

No need to keep references of moving nodes
Information Exposure through Names

- ICN can enable features not possible with IP
  - Exposure of information through names.

A network-layer naming scheme that enables fine-grained description of the desired processed information
Application-Centric Naming (App IDs)

- Needs to support fine-grained description of the desired processed information

  Hierarchical Part
  \[ /a/b/c/ \]
  App Market
  App Developer

  Hash Tags
  \[ \#tag1, \#tag2 \]
  App Developer

- **Fixed part:** *NDN hierarchical naming, longest prefix match*
  - Needs to guarantee compatibility between applications
  - Can define static content: /NewsApp/politics/
  - Or invoke computation: /myTravelAdvisor/Top10Restos
  - App GUI indicates naming, users do not have to be aware of naming

- **Hashtags:** *free keywords to assist application processing*
  - Enables *partial matching* of responses to requests
  - /myTravelAdvisor/Top10Restos #userRating; #London; #indian
  - /routeFinder/tube #euston; #waterloo
Application-Centric Connectivity

• Application-specific 802.11 broadcast domains, through Basic Service Set(s), BSSs
  – Need a “hook” between BSS and the corresponding application
  – Every KEBAPP application advertises its own SSID, through WiFi Direct Groups
  – WiFi Neighbour-Awareness Networking (NAN) can find applications behind BSSs – also optimised for energy efficiency

Information-Centric Forwarding

- We build on a modified version of NDN
- Forward messages to single-hop broadcasting (BSS) domains
- Single-hop operation

- Broadcast domains are considered as node interfaces
- FIB is populated with neighbouring BSSIDs

<table>
<thead>
<tr>
<th>Name Prefix</th>
<th>BSSID</th>
<th>if</th>
</tr>
</thead>
<tbody>
<tr>
<td>/travel/tripAdvisor</td>
<td>#x #y</td>
<td>tripAdvisor #1</td>
</tr>
<tr>
<td>/gaming/gameX</td>
<td>#z</td>
<td>gameX #2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name Prefix</th>
<th>BSSID</th>
<th>if</th>
</tr>
</thead>
<tbody>
<tr>
<td>/travel/routeFinder</td>
<td>#x</td>
<td>routeFinder #1</td>
</tr>
</tbody>
</table>
One PIT entry per request

WiFi Manager populates FIB with hierarchical name advertised by SSID

BSSIDs are the new interfaces

Server part of app internalFace entry links BSSID to specific app that listens to this SSID.
Feasibility – RouteFinder App

Setup
Mobility trace from 3300 users in a Stockholm subway station throughout one hour

All users: 3300
KEBAB Users (10%)

RouteFinder App Users

Graph 1: Number of overlapping KEBAPP users over time

Graph 2: Average response ratio (%) vs. KEBAPP users as percentage of overall nodes in the tracefile

Graph 3: Average first response time (s) vs. KEBAPP users as percentage of overall nodes in the tracefile

Graph 4: Comparison of 5% and 10% sharing rates
Route Finder App

Find the route

UCL Gower street  Victoria station

Public transport  Driving  Walking

SEND REQUEST

Enable Kebapp

DEVICES LIST

Find the route

UCL Gower street  Victoria station

Public transport  Driving  Walking

SEND REQUEST

Start request

Result received

Route from UCL Gower Street to Victoria station
- Walk to Warren Street Station (10 min)
- Take the Victoria Line to Brixton (6 min 3 stops)
- Get off at Victoria Station

TOTAL TIME 16 min

Enable Kebapp

DEVICES LIST

Start request

Result received
Vision: An Edge ICN IoT Platform based on Information-Centric Connectivity

• The long-term plan is to develop a platform for IoT applications
  – users can build applications or applets
  – API should be lightweight and easy to use, e.g., IFTTT-like

• Some applications already implemented in Raspberry PIs – plan to extend to WiFi APs through OpenWRT
How to implement KEBAPP?

Android implementation components

KEBAPP Application

- KEBAPP Application Model
- KEBAPP Application Activity
- KEBAPP Application Service

KEBAPP framework

- jndn (java libs)
- NFD
- jndn-manager
- WiFiDirect libs

KEBAPP middleware

Network devices

KEBAPP background service

KEBAPP APP software

KEBAPP Foreground app UI
Thanks!

Dr Ioannis Psaras
University College London
http://www.ee.ucl.ac.uk/~ipsaras/
i.psaras@ucl.ac.uk
BACKUP SLIDES
Taxi Share App / Carpooling

• Group commuters into taxis/vehicles locally
  – User 1 wants to travel from A – C
  – User 2 wants to travel from A – B, where B is along the route A – C
  – User 3 travels from A – D and so on
• Can’t think of many good reasons not to do this locally...
Online vs Offline Micropayment

- Central trusted authority issues certificates
- Certificates trusted by nodes who pay with vouchers
- Vouchers later validated when users get back online