

Role-Based Architecture & Network Pointers

Jona Schoch

Seminar in Distributed Computing



From Protocol Stack to Protocol Heap - Role-Based Architecture

R. Braden, T. Faber, M. Handley

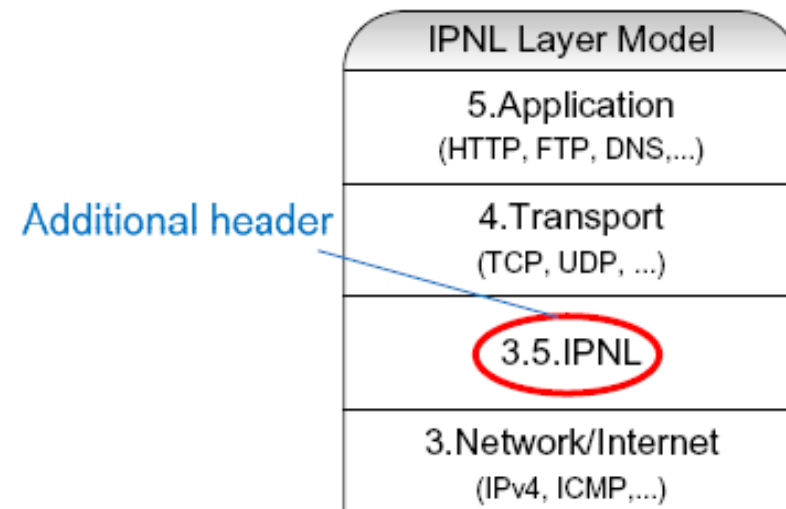
Protocol Stack model

- Consists of
 - Protocol layers
 - Protocol headers
- Created in the 70s
- Worked well

5 Application DNS, FTP, HTTP...
4 Transport TCP, UDP...
3 Network IP, ARP, OSPF...
2 Data Link Ethernet, Wi-Fi...
1 Physical Optical Fiber, Coaxial Cable...

Problems

- Internet has changed!
- Usual suspects: NAT, Firewalls, Proxies...
- „Last on – first off“
- Inflexibility



Source: M. Kaufmann

A new architecture

- No stack
- No Layers
- New functional unit: Role

→ Role-based Architecture: RBA

Role

- Communicating building block
- Performs function on packets
- Generally not hierarchical
- May be stateful

Properties of Roles

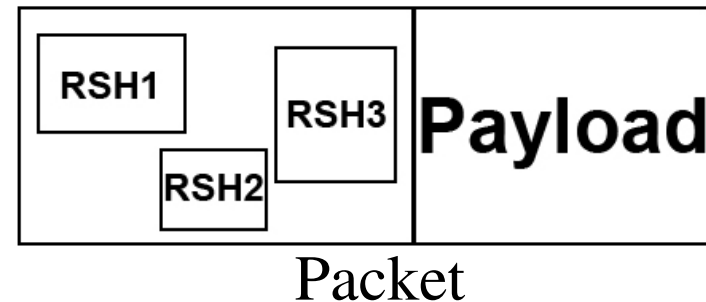
- Instantiated in Network-nodes
- Identified by a RoleID
- Role address: RoleID@NodeID
- Reflective roles
- Generic roles
- ...

Role Data

- Input / output of a role
 - Application data payload
 - Metadata
- Role Specific Headers (RSHs)

Role Specific Headers

- Divided metadata
- Format role specific
- Addressed to roles



- RSH (<RoleAddressList> ; <RSHBody>)

Datagram Delivery Example

- RSH (<RoleAddressList> ; <RSHBody>)

- RBA equivalent to a simple IP datagram

```
{ RSH(LinkLayer@NextHopAddr ; ),  
  RSH(HbHForward@* ; destNodeID),  
  RSH(HbHSource@* ; sourceNodeID),  
  RSH(DestApp@destNodeID ; AppID, payload) }
```

RBA Objectives

- Extensibility
- Portability
- Auditability
- Controlled access to metadata

Realization of RBA

- Variety of ways → further research required
 - Entirely role-based
 - Above particular layer
 - Only abstraction
- Packet format design
 - Generality vs. wasting bytes

Packet Structure

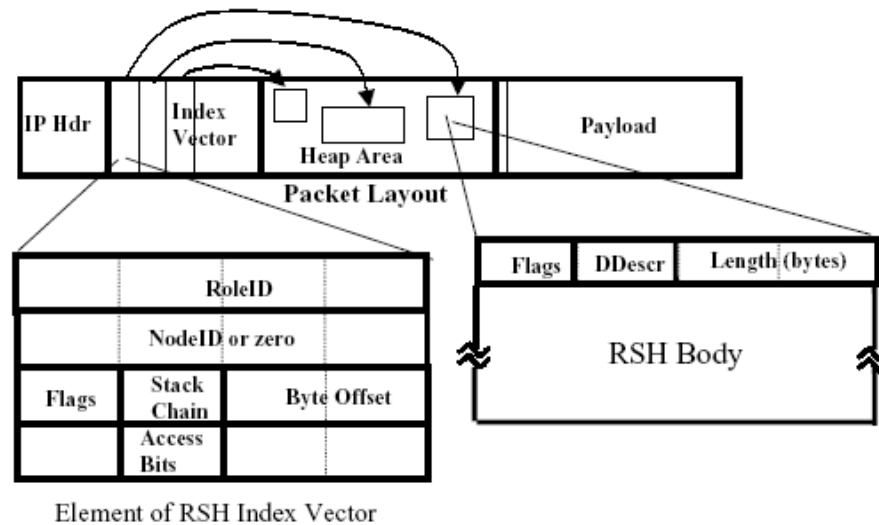


Figure 2: Possible RBA Packet Layout

Network Pointers

R. Gold, C. Tschudin

Goals

- Break the processing black box!
 - Control packet handling
- Empower the end-nodes!
 - Directable indirection

Network Pointers

Packet processing function

- Addressable
- Forwarding
- Header modification

Basic idea

- Keep IP
- Configure Data Link layer
- Pointer Space
 - Mapping from/to IP
 - Build new services

Network Pointers - Terminology

- Network Pointer
- Selector (local) = Pointer value
- Context

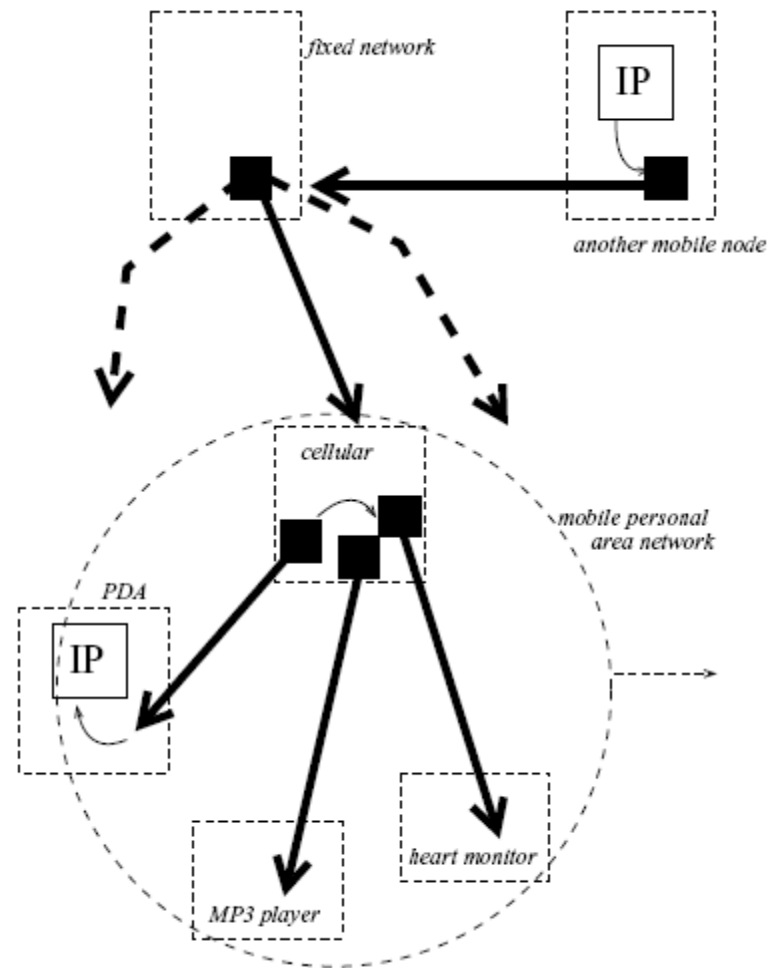
- Resolution function
 - On all network nodes
 - Well known selector

Implications

- Forwarding State
 - in network
 - in packet (parameter stack)

- From End- to Start-Addresses!

Example: Mobile Personal Area Network



Combining RBA & Network Pointers

- Get rid of implicit layer 2.5
- Network Pointers as back plane for RBA
 - Steering packets
 - Identifying processing instances

Remaining questions

- What is really new?
- How to change state of pointers?

References

- „From Protocol Stack to Protocol Heap – Role-Based Architecture“, R. Braden, T. Faber, M. Handley
- „New Internet Architectures“, M. Kaufmann

- „Network Pointers“, R. Gold, C. Tschudin
- „A Virtualized Link Layer with Support for Indirection“, R. Gold, P. Gunningberg, C. Tschudin
- „LUNAR: Lightweight Underlay Network Ad-hoc Routing“, R. Gold, C. Tschudin



Thank you!

Questions, Comments?