A Recursive Network Architecture

Joe Touch, Yu-Shun Wang, Venkata Pingali

USC/ISI
Motivation

- Layers of a stack becoming more similar
  - Security, soft-state, pacing, retransmission
- Desire to support interlayer cooperation
  - Message boundary, cong. control, compression vs. encryption interactions
- Desire to support overlay layers
  - Clearly needed, but don’t map to 1-7

Is layering more than a coding artifact?
Internet Architecture

Accused of ossification, but:

- Ossification = stability
- Flexibility is abundant:
  - Shim layers:
    - HIP, SHIM6, IPsec, TLS
  - Muxing layers:
    - SCTP, RDDP, BEEP
  - Connections:
    - MPLS, GRE, IKE, BEEP, SCTP
  - Virtualization:
    - L2VPN, L3VPN/X-Bone/RON/Detour, L7-DHTs
Challenges of Layering

- Which to add…
  - IPv4/IPv6, TCP/DCCP/SCTP
- When to add…
  - Security, muxing, cong. control
- Real vs. virtual
  - What’s the difference?
Observations

1. Services are relative

2. A template can avoid recapitulation

3. Composition requires coordination
The OSI 7-layer Model

- Layer indicates function

- But...
  - Functions are recapitulated:
    - Formatting at link and presentation
    - Muxing at transport and session
Scope defines a layer

- Its endpoints
  - A “hop” @layer N = E2E extent of layer N-1
- The layer above
  - What services this layer provides
- The layer below
  - What services this layer requires
- E.g.: Shared state at diff. layers for diff. services
  - Application binding
  - Transport delivery
  - Net security

The difference is scope
Adding Services is Hard

- Wedge between (IPsec, left) or replicate (virtualization, right)
Recapitulation

- Component services repeat:
  - handshake / state management
  - security
  - policy (admission control, filtering)
  - multiplexing and demultiplexing
  - retransmission
  - reordering
  - pacing / congestion control
  - switching / forwarding
- Compounded by virtualization
  - Layer on layer on layer
Composition Requires Coordination

- Many services integrate layers
  - Congestion control
  - Message boundaries
  - Security
  - State establishment
- Current interlayer interface is limited
  - Defined by each layer
  - No general security, state, etc. interface
RNA Stack

- Only needed layers
  - With only needed services
- One MP, many instances
  - Configurable like TP++
  - Retain layers to limit scope
  - Context-sensitive
Layer Context Sensitivity

- E.g., mp-1 morphs varies when over wireless vs. optical
  - Opportunity for auto-tuning
RNA Metaprotocol

- Template of basic protocol service:

  - Shared State
  - Security
  - Flow Control
  - Next Layer Resolution
MDCM from *Choices*

**Dynamic interlayer glue**

- One template for ARP, BGP, IP, DNS
  - Also for ‘BARP’
- Structured template
  - With plug-in funcs.

```plaintext
LAYER(DATA, SRC, DST)
    Process DATA, SRC, DST into MSG
    WHILE (Here <> DST)
        IF (exists(lower layer))
            Select a lower layer
            Resolve SRC/DST to next layer S’, D’
            LAYER(MSG, S’, D’)
        ELSE
            FAIL /* can’t find destination */
        ENDIF
    ENDWHILE
    /* message arrives here */
    RETURN {up the current stack}
```
Components of RNA MP

Instantiate MDCM’s “Process DATA”

- Establish / refresh state
- Encrypt / decrypt message
- Apply filtering
- Pace output via flow control
- Pace input to allow reordering
- Multiplex/demultiplex as indicated
  - includes switching/forwarding
Challenges

- MP design
  - Building a sensible, generic template
- Stack management
  - Supporting instantiation and composition
- Supporting interlayer coordination
  - Designing a sensible, recursive API
  - Makes it easier to interface (to yourself, e.g., LEGO)
- Supporting context sensitivity
  - Detecting environment and autotuning
Related Work

- Modular protocol environments
  - Click, x-Kernel, Netgraph, Flexible Protocol Stacks
  - RNA adds a constrained template

- Template protocol models
  - MDCM, RBA
  - RNA adds structured sequence of services

- Context-sensitive protocols
  - PEPs, Shims, intermediate overlay layers, etc.
  - RNA incorporates this into the stack directly

- Configurable protocols
  - XTP, TP++, SCTP
  - RNA makes every layer configurable, but keeps multiple layers.
Status

- Ongoing stack/protocol survey
  - Scope vs. layer structure
  - Intra-/Interlayer ‘feature creep’
  - Intra-/Interlayer bindings

- Observation: Inclusive Scoping Issues
  1. Layers have static, linear lineage
     - Works well for messages streams; poorly for connections
  2. Services are satisfied top-down
     - Consequence of ‘E2E argument’ + #1

www.isi.edu/rna