On Evolvability, Architecture, Tussle, Layering and Signalling

Mark Handley
UCL
Change

Huge innovation in applications

Ossification of the core protocols

Relentless evolution of the underlying technology

- email
- WWW
- phone...

- SMTP
- HTTP
- RTP...

- TCP
- UDP...

- ethernet
- PPP...

- CSMA
- async
- sonet...

- copper
- fiber
- radio...
Tussle and the death of end-to-end.

Different parties want varying degrees of control over connections.

- End systems (obviously) - to enable applications.
- Firewalls - to enhance security.
- Deep packet inspection - to differentiate service.
- Link layers - to enhance transport performance.
- Transparent caches - to enhance application performance, reduce bandwidth costs.
- Security services - to be spooky.
Current Layered E2E Architecture
Current Sort-of-Layered Sort-of-E2E Architecture
Evolvability

- Any new architecture must permit tussles to play out within the architecture.
- Alternative is:
  - Difficult to evolve because of unintended feature interactions.
  - Eventual ossification and stagnation.
Some New Approaches

- Role-based Architecture.
- Connection Signalling.
Role-based Architecture

- Break packets into separable functionality.
  - Avoids unnecessary coupling as the architecture evolves.
- Address sub-packets to entities that perform specific roles.
  - Provides a way to talk to an entity (e.g., Firewall) other than the remote end system.
  - May not know its address (or it may not have an address).
- Allow entities along the path to add or remove sub-packets as required to perform their job.
  - Provides a place in the architecture for them.
Roles and Role-Specific Headers
Contrived Example

RSH( Forward.HbH@*; B, A)
RSH( AppMux@B; destPort, SrcPort )
RSH( Firewall@*; ```Disable = Cache```
RSH( Cache@*; )
RSH( DestApp@B; <payload>)

- Directive indicates data can be cached, but then indicates to the firewall to disable the Caching directive.
- Allows caching only within the firewall.
Connection Signalling

Use a signalling protocol (“CSP”) to initiate all transport connections.
- Not VCs though, connections can still be datagrams.
Not strictly layered under or over transport protocols.
- More like alongside, akin to how ICMP is to IP.
Provides a hook within the architecture for different entities to signal their needs.

HTTP, SMTP, RTP, ...

TCP CSP  
SCTP CSP  
UDP CSP  
IP ICMP
Firewall redirect to offpath proxy

\[
\text{Setup}(A, p_1 \leftrightarrow B, 80)
\]

\[
\text{Redirect}(A, p_1 \leftrightarrow P:B, 80)
\]

\[
\text{Setup}(A, p_1 \leftrightarrow P:B, 80)
\]

\[
\text{Setup}(P, p_1 \leftrightarrow B, 80)
\]

HTTP Connection

HTTP Proxy
Hidden Mobile Server

Setup \( (A, p_1 \leftrightarrow S, p_2) \)

Redirect \( (A, p_1 \leftrightarrow S:B, p_2) + \text{nonce+sig} \)

Set up \( (A, p_1 \leftrightarrow S:B, p_2) \)

Data Transfer \( (A, p_1 \leftrightarrow B, p_2) \)

Detach + Attach + Nonce + Sig

Data Transfer \( (A, p_1 \leftrightarrow C, p_2) \)

Register \( (S \text{ at } B) \)