# Socket Science(*)
# TAPS Bof, London, 5.3.2014

Jon.Crowcroft@cl.cam.ac.uk

http://www.cl.cam.ac.uk/~jac22

(*) Sun Engineer's job title (anyone remember who?)

# Problem Statement

- Sockets have a lingering legacy
- But lots of stuff changed
- Multi-core, Multi-path, Mobile(Migration), Multi-media, etc
- [lets not mention multicast]
- Better Security just got coerced on us
- App Semantics got complicated
- Users don't get any more patient

# Msg not byte, srcs, not src

- Byte stream TCP i/f pretty obsolete
  - Often have compiled (serialized) object so know size a priori
- Simplest client app now gets data from multiple places
  - Pipeline from nic to render doesn't have to be serial -
    - mux
    - i.e. ordering no longer needed either - latency
  - Not even (always) meaningful for stuff from different servers

# Multipath coming along

- Whether mptcp siri, or more mainstream, or sctp
- Again dispense with ordering may give speedup
- Striping to different cores
- Migrating “end point” (e.g. vm moves)
- Etc etc

# Multimedia transport reliability

- Delivery service reliability requirement probablistic
- Yes, still want TFRC or whatever
- But want to deliver segments with gaps
- Need to know their place (pace, RDMA☺) coz of dim codec design
- May need to carry timing data twixt APP and NIC too

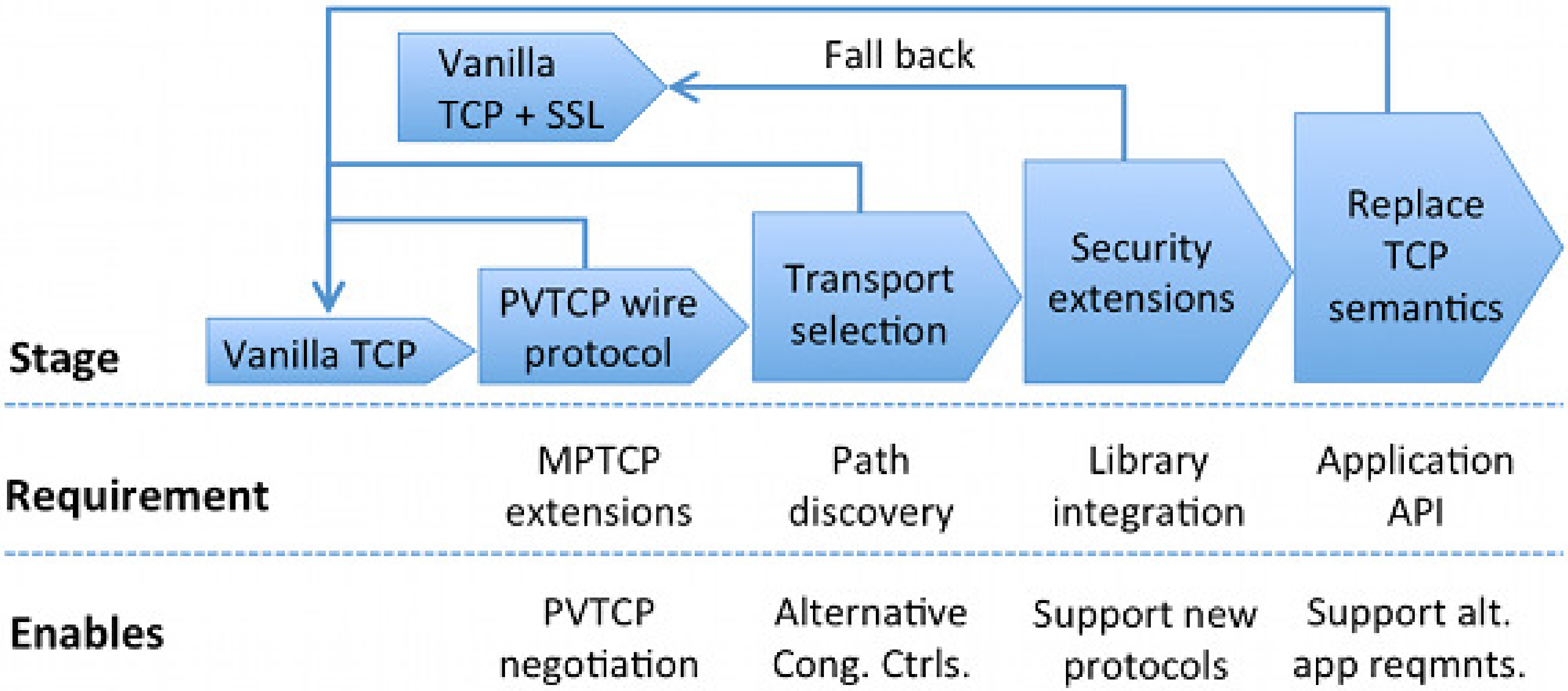
# Security needs delta

- TLS perhaps insufficient (4-7 RTT exchanges!)
    - “end” isn’t an end (multicore, multiparty)
        - So n*7 is **even worse**
    - Parts of the end move too (vm migrate)
        - So MITM is with us from get go…
- No service protection…
- Maybe re-visit kerberos . .. …

# Our strawman: polyversal tcp

- This is just our take on things - for exploring the space
- Plenty of other takes exist (see next talk for a list of many fine alts)
- Note we seem to agree that you need
  - API to select and be told what transport you are getting
  - Fall back to baseline TCP
    - Middlebox constraint
  - Probably some 3rd party hint&key server

# PVTCP Progression

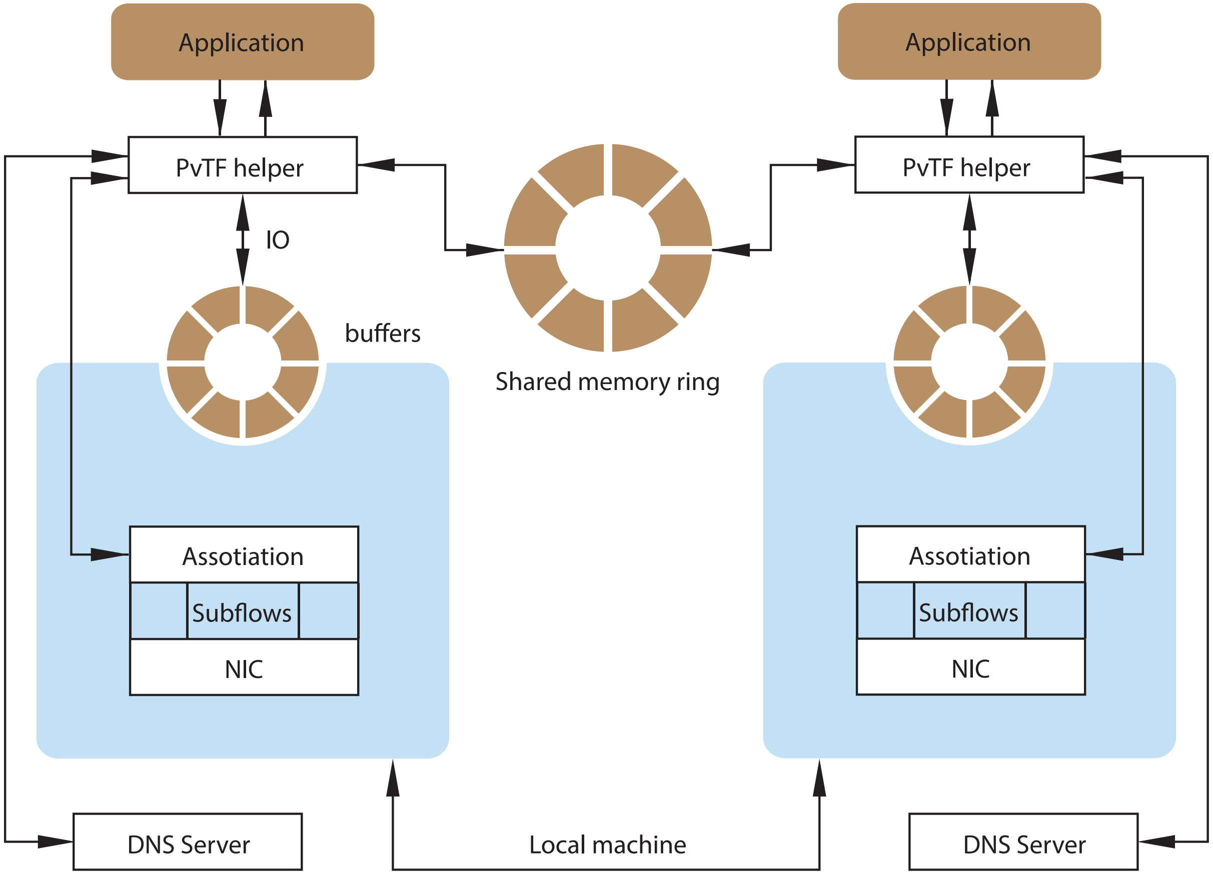


| Stage | Vanilla TCP | PVTCP wire protocol | Transport selection | Security extensions | Replace TCP semantics |
|---|---|---|---|---|---|
| Requirement | | MPTCP extensions | Path discovery | Library integration | Application API |
| Enables | | PVTCP negotiation | Alternative Cong. Ctrls. | Support new protocols | Support alt. app reqmnts. |

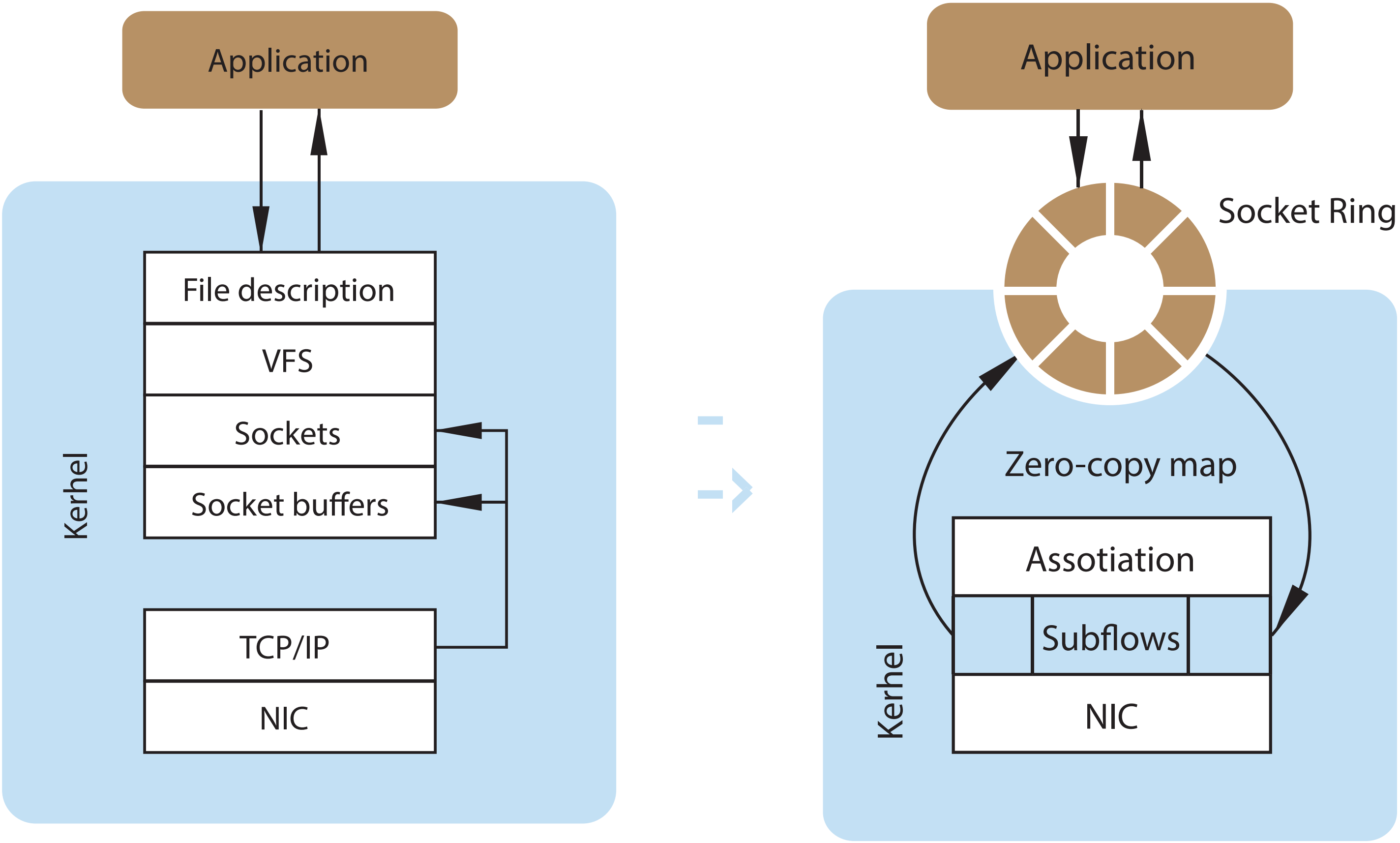
# Open Arches...

## Proposed architecture

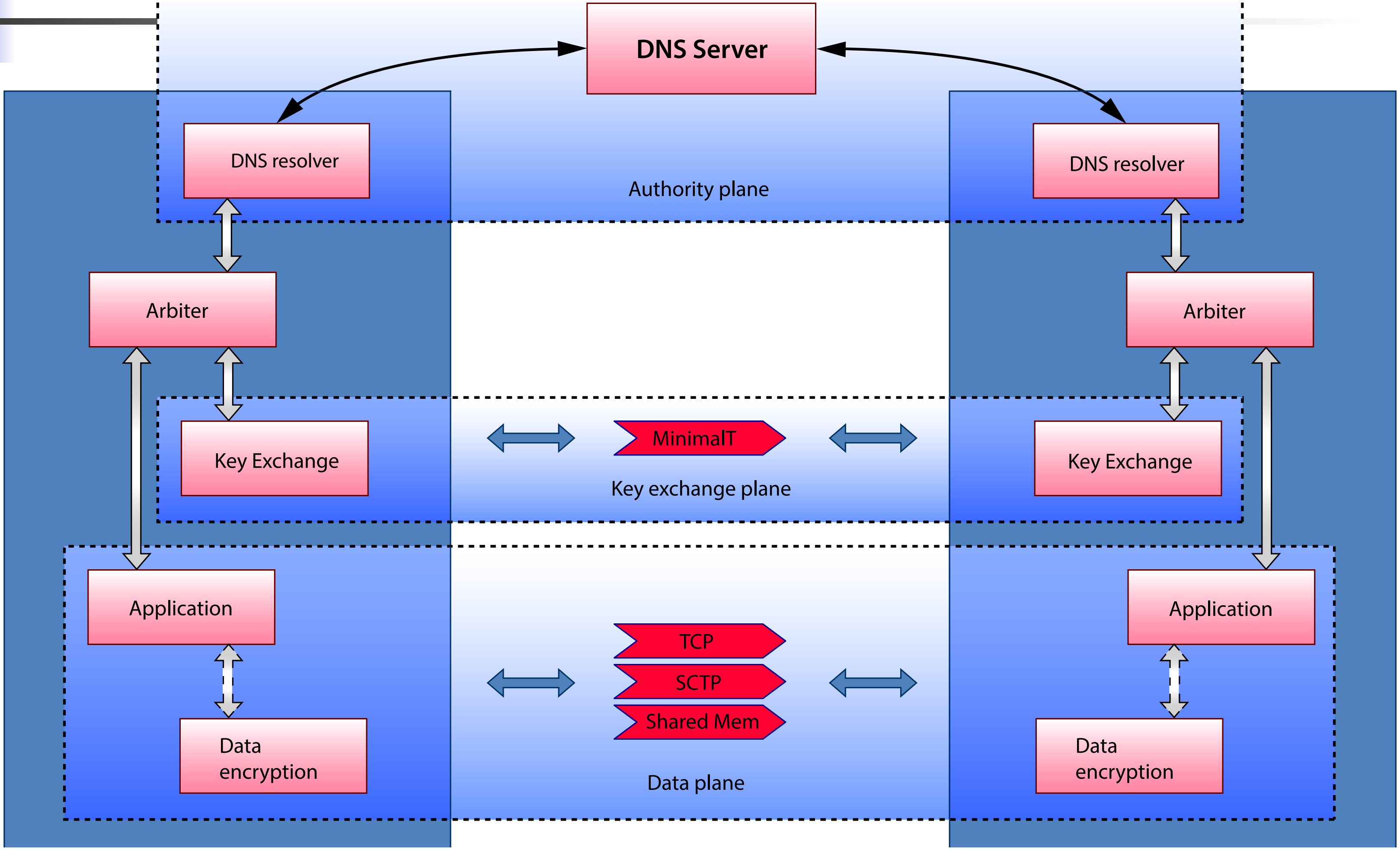
### Overall architecture



# Old Sock stack + New Strawman

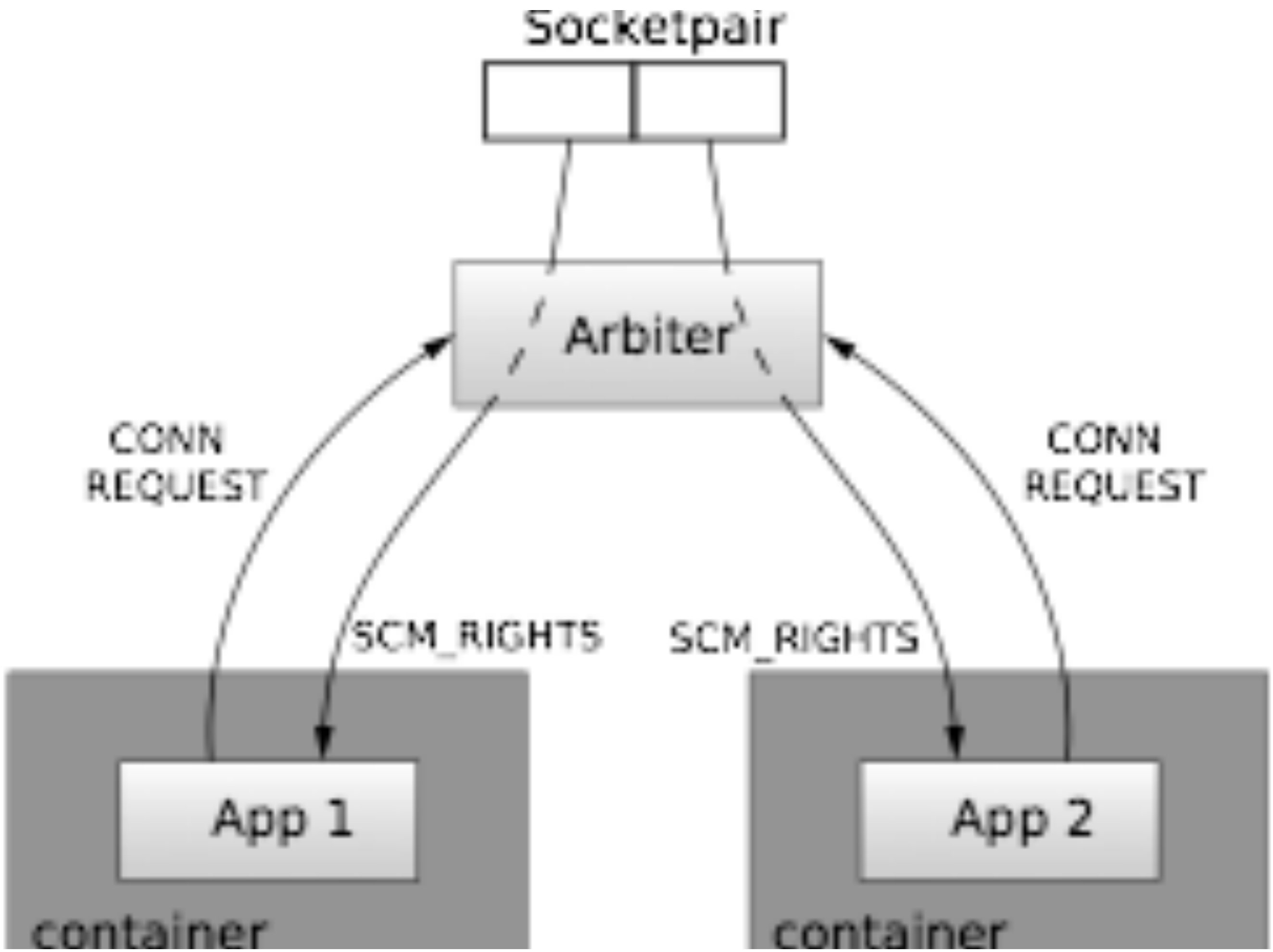


# Straw sec arch



- Note where keys served

# Setup - note “container” capsicum



# Its not optional

- Be careful what we wish for
- Security v. RTTs  is v. v. difficult
- Must do no harm, at min

# Who Am I?