

PCP Proxy

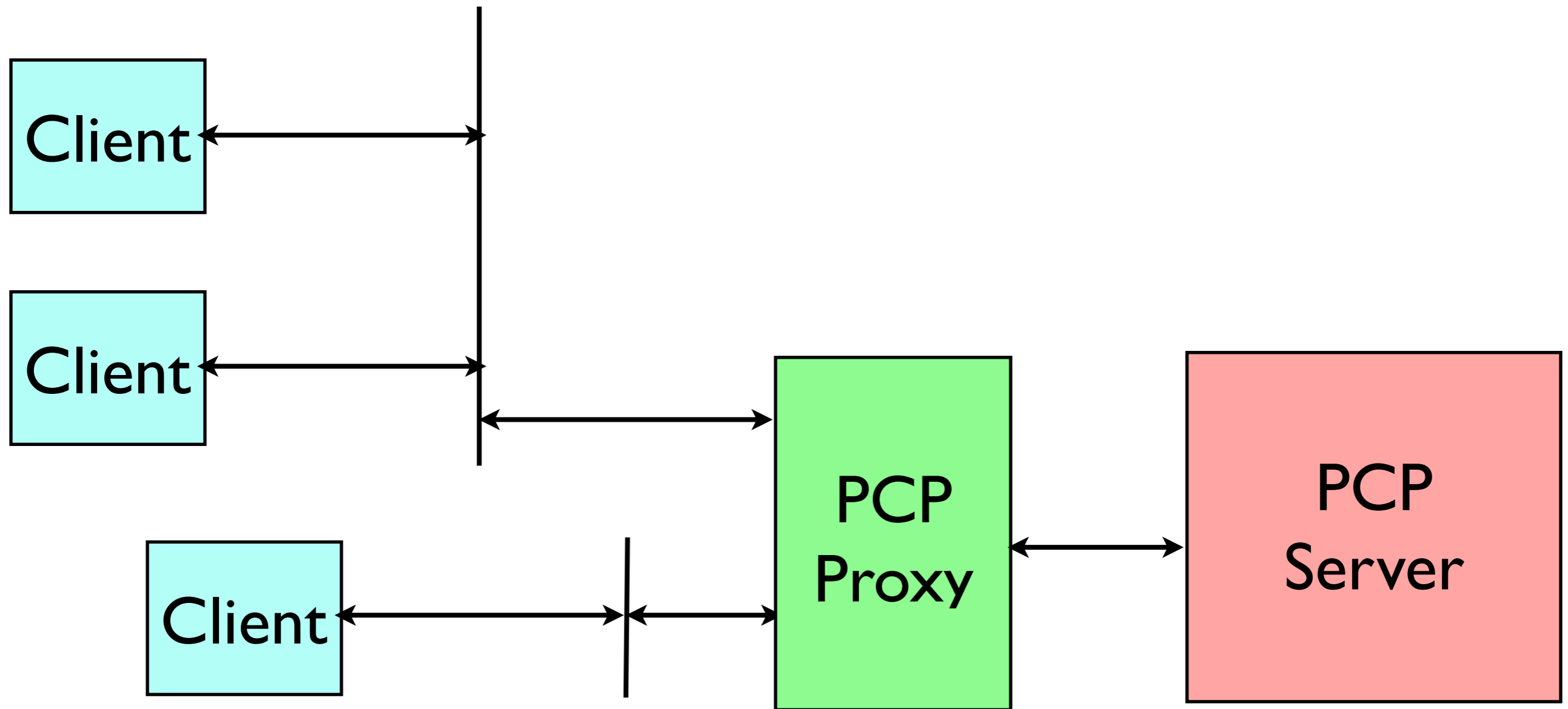
Mohammed Boucadair, Reinaldo Penno,
Dan Wing, Francis Dupont

IETF 80, Prague

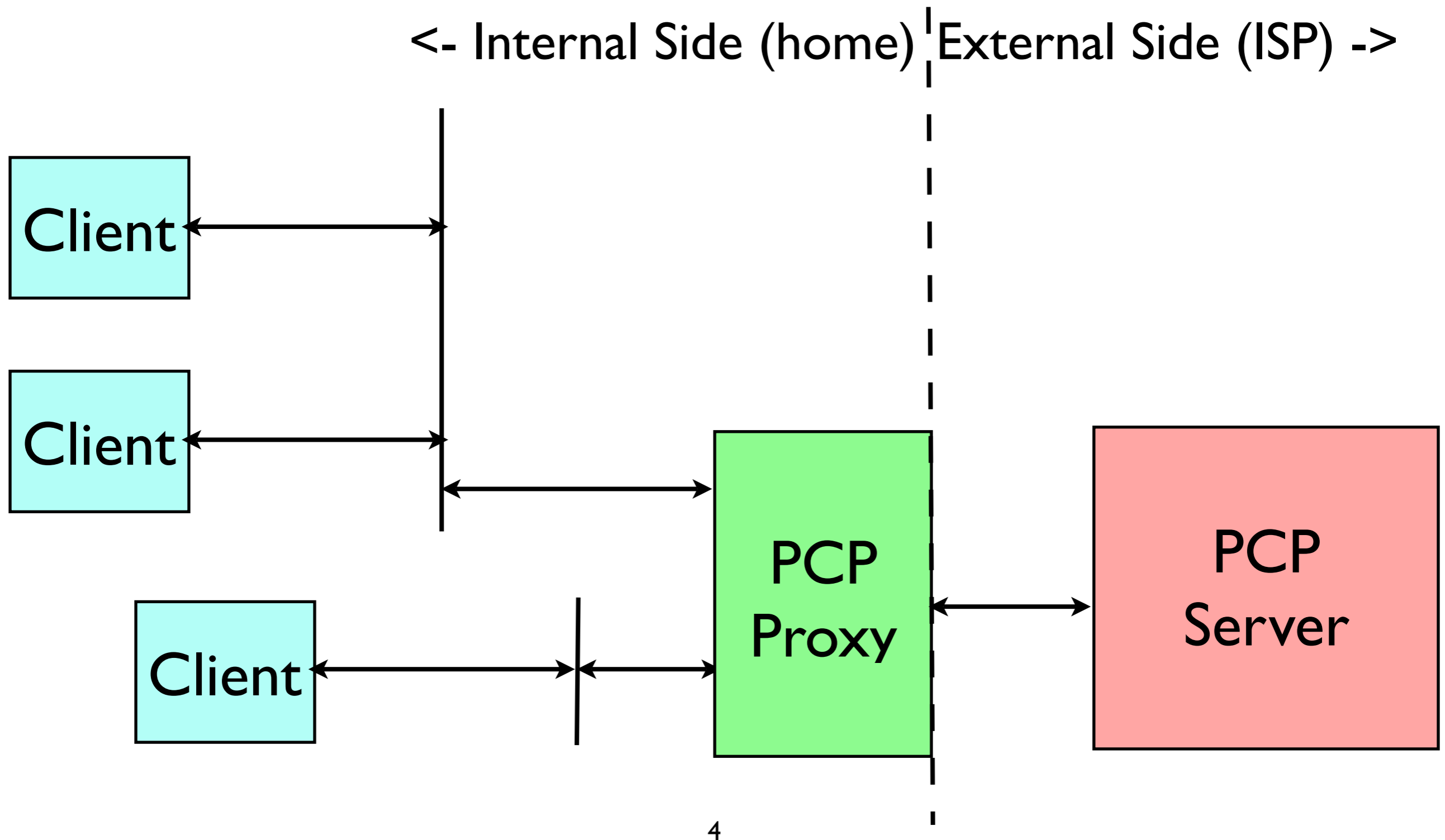
Plan

- Architecture
- Simple Proxy
- Embedded firewall and NAT
- Smart Proxy
- Security
- Open questions

Architecture (I)



Architecture (2)



Architecture (3)

- Simple or smart PCP Proxy
- Can be integrated inside an InterWorking Function (for UPnP IGD / NAT-PMP)
- Can enforce security controls

Simple Proxy (I)

- Minimal processing, on received requests:
 - check third party
 - apply security controls (if any)
 - build error response if reject
 - adjust request (e.g., add 3rd party option)
 - forward the updated request on a fresh socket connected to the PCP Server

Simple Proxy (2)

- Wait for response from the PCP server
- Build an ICMP error on hard send() error
- On response:
 - adjust it back (e.g., remove previously inserted 3rd party option)
 - send it back to the PCP Client

Embedded firewall

- Must open the corresponding hole on a MAP response
- Lifetime issue (similar but simpler than for embedded NAT)

Embedded NAT (I)

- Get or create the corresponding local explicit dynamic mapping on MAP requests
- Must translate internal address and port in request (part of the “adjust”)
- Must translate them back in response

Embedded NAT (2)

- Lifetime issue
- Easy case: local mappings have lifetime:
 - enforce compatible value in requests
 - copy assigned lifetimes from responses
- Hard case: no local lifetime:
 - must maintain full state for MAP messages
 - delete local mapping on expiry

Smart Proxy

- Extra functions:
 - handle multiple PCP Servers
 - handle Epoch value (needed for other functions)
 - request/response caching
 - handle timeouts (improvement of the previous)
 - manage full state for explicit dynamic mappings

Epoch value

- Smart PCP Proxy function example:
 - the Epoch value in responses forwarded to clients is taken from an internal timer
 - this timer is reset to zero when needed
 - please check the I-D in the case we forgot a condition for such a reset!

Security (I)

- Split-horizon anti-spoofing
- Third party policy (default is to not authorize)
- ACL based authorization
- Unknown OpCodes and/or mandatory to process Options

Security (2)

- Required security controls when the PCP Proxy is on a trust domain boundary:
 - split-horizon anti-spoofing (just two tests to add in standard proxy code)
 - a third party policy must be enforced
- These requirements themselves are compatible with a minimal implementation

Open Questions (I)

- What to put in a built response (Epoch, external address)?
- only on requests rejected by security controls
- Epoch handling solves this
- generic IVWF issue: the solution will be specified in a dedicated document

Open Questions (2)

- Adding a third party option in a request can make a valid request too large
- Corner case but no clean solution
- The current idea is to ignore this, in particular to never check the size of a request (of course this doesn't imply to overflow buffers :-)