Happy Eyeballs for Transport Selection

draft-grinnemo-taps-he-03

K.-J. Grinnemo, A. Brunstrom, P. Hurtig,
N. Khademi, Z. Bozakov

neat

Horizon 2020
European Union funding
for Research & Innovation

Introduction

- TAPS work item 3
  - "... explain how to select and engage an appropriate protocol and how to discover which protocols are available for the selected service between a given pair of end points"
- Calls for happy eyeballs mechanism for transport protocol selection
  - Try multiple protocols in parallel

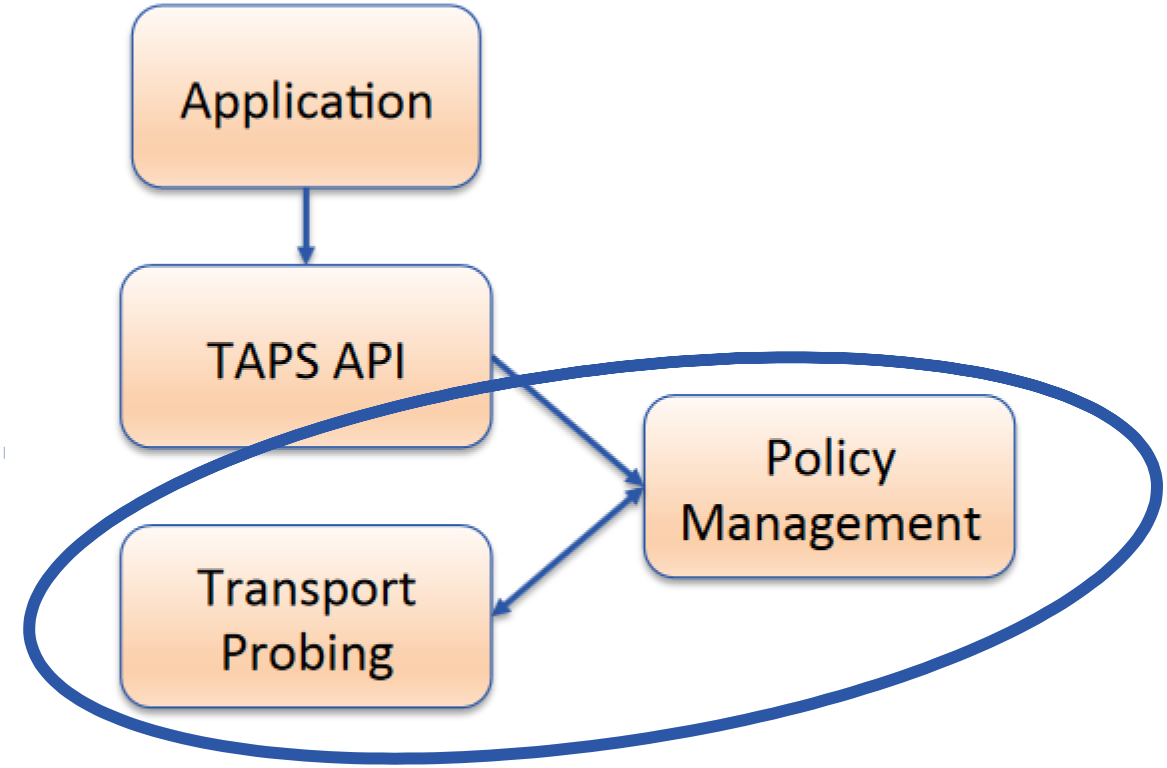
Introduction

- Generalizes previous work
  - D. Wing and A. Yourtchenko, “Happy Eyeballs: Success with Dual-Stack Hosts”, RFC 6555, April 2012.
- Selection of complete transports, not single protocols
- Dynamic selection on the basis of pre-set policies and estimated network characteristics

Updates to the Draft

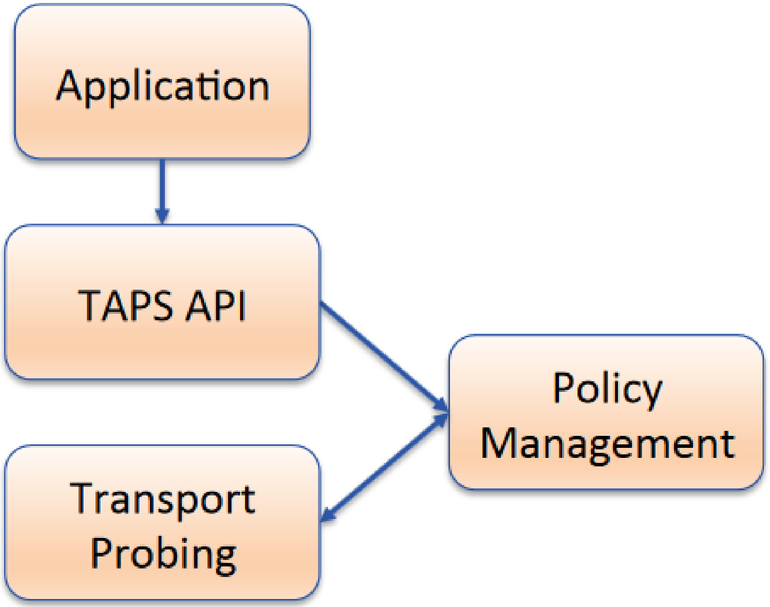
- Improved wording in relation to RFC 6555
- Clarified that the HE draft applies to connection-oriented transports
- Clarified that only one transport connection is returned to the client and that other connection attempts are used only to populate the cache (INTENDED)
- Cache lifetime indicated as system dependent
- Changed example to use IPv6
- Various text clarifications and other smaller improvements

Happy Eyeballs Framework



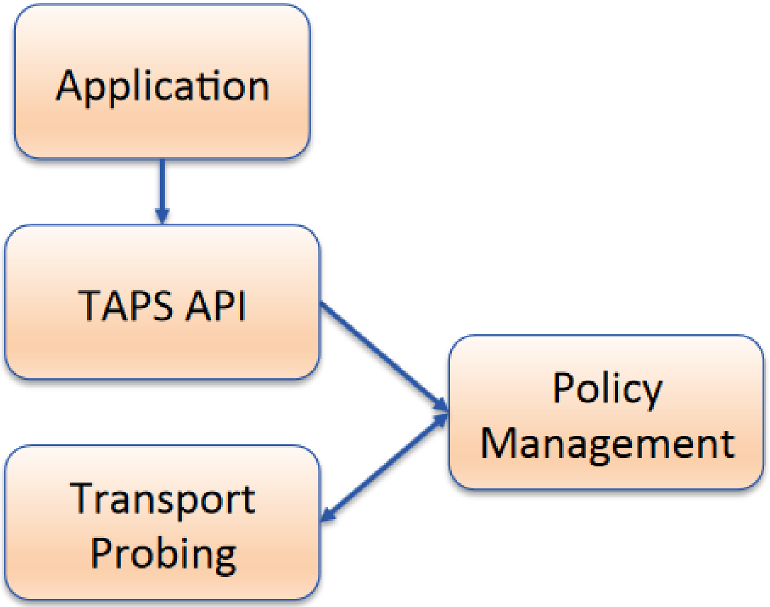
Policy Management

- Creates a list of candidate transport solutions
  - ordered by priority
- Created based on application requirements, available information about the network and configured policies
  - should use cached information
  - comply with RFC 6555



Transport Probing

- Initiate connection attempts for each candidate transport solution
  - initiated in priority order
  - difference in priority between two candidates is translated into a delay
  - first connection to be established is returned to the application
- Cache the outcome of the connection attempts
  - allow initiated connection attempts to complete



Discussion Items

- What should the interface between the HE algorithm and the policy management look like
  - More general what do the interfaces between the TAPS components look like?
  - Do we need some form of architecture document?
- What should be detailed in the specification of the HE algorithm and what should be left open for implementation?
  - Cache lifetime, time between connection attempts, candidate list generation