PCP

Two IESG DISCUSSes for
draft-ietf-pcp-base

August 2, 2012

IETF-84, Vancouver

Authors: Dan Wing, Stuart Cheshire, Mohamed Boucadair, Reinaldo Penno, Paul Selkirk

Agenda

- Document Status
- Robert Sparks DISCUSS
  - PCP server state change synchronization problem
  - New text
- Pete Resnick's DISCUSS
  - Broadcast on state loss

Document Status

- Nearly finished
- Two DISCUSSes to share with working group

Robert’s DISCUSS: retransmissions

- Robert Sparks pointed out PCP server could change state between retransmitted requests
- Problem only occurs if:
  - Request generates error (which is delayed)
  - PCP server state changes
  - Second (retransmitted) request generates success (which might be lost)
  - PCP client sees error response
- Always been a problem in PCP

Diagram of the problem

```
 PCP client                               PCP server
     |                                        |
     |----request, Nonce=1------------------->|
     |                                        |
     |                                 -------|
     |                                /       |
     |                               /    (state changes)
     |                              /         |
     |----request, Nonce=1---------/--------->|
     |                            /           |
     |<--error response, Nonce=1--            |
     |                                        |
(processed)                                   |
     |                                        |
     |  (lost) X<-success response, Nonce=1---|
     |                                        |
```

New text for Robert’s DISCUSS

Each of the retransmissions SHOULD use the same Mapping Nonce value. By using the same Mapping Nonce value, any of the responses with that Mapping Nonce are considered valid by the client, which allows PCP requests to be satisfied as quickly as possible even when there are network delays or PCP server processing delays. **However, using the same Mapping Nonce for each retransmission means the PCP client and server can be desynchronized. For example, a PCP server error response might be delayed (by the PCP server or by the network) but the retransmitted request (several seconds later) generates a success response which is lost by the network.**

Pete’s DISCUSS (1/3)

- Ensure PCP server communicates state loss to PCP clients
- Otherwise, PCP clients won’t trust PCP lifetime
  - They will aggressively re-query the PCP server

Pete’s DISCUSS (2/3)

- Suggestion 1: unicast, multicast, broadcast
- Authors unaware of scenario where multicast unavailable, but broadcast available

- Authors: “No”

Pete’s DISCUSS (2/3)

- “PCP servers ~~SHOULD~~MUST implement ~~both~~at least one [rapid recovery] mechanism.”
- Additional suggestion: server indicates support for unicast Rapid Recovery and multicast Rapid Recovery

End