Clarifying PROBE (rfc8335)

Bill Fenner
IETF 120 @intarea
Recall the motivation

Do we need clarifications?
A survey of existing responders and my opinion simply based on the spec

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>D</th>
<th>Me</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extended checksum</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>required?</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packet data echoed?</td>
<td>✓</td>
<td>✓</td>
<td>n/a</td>
<td>✓</td>
</tr>
<tr>
<td>Extra data allowed?</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Do we need clarifications? (Clients)
A survey of existing clients and my opinion based on the spec

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>C</th>
<th>Bill</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fill in extended checksum</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Include extra data</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Rely on responder echoing extra data</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Include padding in option length</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Do we need clarifications? (Users)
Is this a successful probe request?

```
user@Router> probe a.b.c.d by-name ael
PROBE a.b.c.d (a.b.c.d):
  27 bytes from a.b.c.d: icmp_seq=0 ttl=255 code=2 state=0
  active=0 IPv4=0 IPv6=0 time=6.981 ms
27 bytes from a.b.c.d: icmp_seq=1 ttl=255 code=2 state=0
  active=0 IPv4=0 IPv6=0 time=4.420 ms
27 bytes from a.b.c.d: icmp_seq=2 ttl=255 code=2 state=0
  active=0 IPv4=0 IPv6=0 time=2.530 ms
```
Changes since last meeting

Used brackets to emphasize that the “Data” field is optional

Github Issues:

#1 - use NUL as specified in RFC20

#2 - update interfaces to use more yang-y and less SNMP-y names
Current Status

Authors think it’s a valuable update to RFC8335, and ready to move forward.
Resources

https://github.com/aristanetworks/probe-tools contains an IPv4 client (to test a responder) and a synthetic response generator (to test a client)

Pull request #1131 at https://github.com/the-tcpdump-group/tcpdump/ implements PROBE printing for IPv6; IPv4 is in the top-of-tree (merged since last time)

https://github.com/aristanetworks/iputils contains:

- The list of issues from my review of the original code by Andreas Roeseler and Spencer Lang
- A rebase to the current iputils head of the original work, and fixes to most of the issues

https://fenner.github.io/probe-clarification/ for this document
What’s next?

Before publication as RFC, decide what to do about author count limitation

Interop testing?

WG adoption?