

Protocol-Error

RADEXT - IETF 125

Or “silently discard” considered
harmful

RADIUS HAS A SIGNALLING ISSUE

- ▶ Many parts of the specifications say that servers will “silently discard” packets
 - ▶ Most notably Accounting-Request
- ▶ Many implementations silently discard packets in other situations, too
- ▶ Clients then do not know if the server is slow, down, or if the next hop is down
- ▶ The protocol allows for responses to be ACK or NAK, but only sometimes
 - ▶ But not “unable to perform action”

“SILENTLY DISCARD” CAN BE FOR MANY REASONS

- ▶ Unknown client
- ▶ Known client, invalid or unauthenticated packet
- ▶ Packet code not accepted on this port
- ▶ Cannot process the packet
 - ▶ too busy / proxying failed / processing failed
 - ▶ *Protocol-Error is really only for this situation*

THE PROPOSAL

- ▶ Allow Protocol-Error response to mean
 - ▶ “I received your packet, but I cannot process it. Please try somewhere else”
- ▶ Already defined in RFC7930
 - ▶ In 1-2 paragraphs, with minimal explanatory text
- ▶ This document goes into substantially more detail
 - ▶ When it can / cannot be used, etc.

ADDITIONAL CONSIDERATIONS

- ▶ No link-layer negotiation is proposed
- ▶ Admins can configure this new behavior manually
- ▶ It is always safe to send Protocol-Error responses, as old clients will ignore it
- ▶ Much of the document text is explaining what is wrong with implementations, and what should be done instead
- ▶ To address long-standing issues with implementations
- ▶ Perhaps this document could be “improved RADIUS signalling”

HACKATHON RESULTS

- ▶ Notes on the RADEXT Wiki - [Concerns with Protocol-Error](#)
- ▶ FreeRADIUS [implementation notes](#)
- ▶ Over all, the implementation is simple, and is interoperable
- ▶ The belief is that it will help with corner cases
 - ▶ Real-world numbers will help here.

QUESTIONS / COMMENTS?

- ▶ The document needs substantial updating, but the basic foundation is there