

# **Extended Identifier Attribute for radius**

*draft-chen-radext-identifier-attr-01*

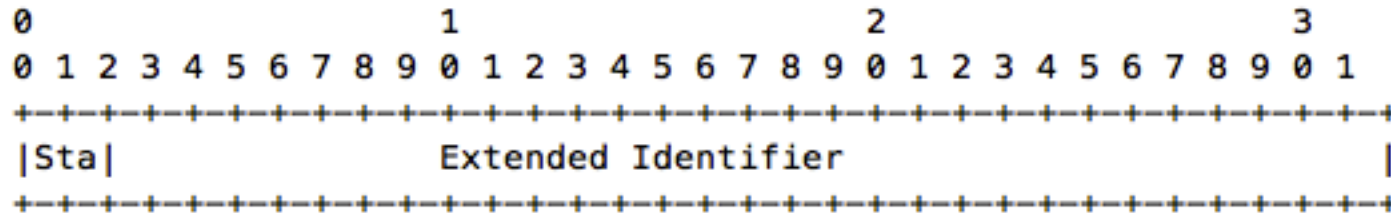
Enke Chen, Naiming Shen

IETF 99, Prague

# Summary of Changes (IETF 96 ☐)

- Defined a new attribute for the Extended ID
- Removed changes to the packet header
- Removed the optional “Extended Code”
- Relaxed the use of “Identifier” and “Extended ID” from “OR” to “AND” for flexibility (and even larger ID space, if ever needed)
- Renamed the file handle from draft-chen-radext-extended-header

# Extended Identifier Attribute



- For capability discovery in “Status-Server Msg”, and for the Extended-ID in other msgs
- Status field in “Status-Server Msg”
  - 1 (“request”): from client to server
  - 2 (“accept”) or 3 (“reject”): from server to client
- “Extended-ID” field: can be used alone, or with the “Identifier” field (left to implementation)

# Operations

- Capability discovery using Status-Server Msg
- “Extended-ID Attribute” to be encoded as the very first one in the attribute list
- Client or proxy server can use the “Extended ID” once the capability is confirmed
- PROXY server: remove the attribute and reconstruct if needed

# Implementation

- FreeRadius patch & testing report posted to [radext@ietf.org](mailto:radext@ietf.org) on April 28, 2017
  - <https://www.ietf.org/mail-archive/web/radext/current/msg09905.html>
- Complete implementation:
  - Capability discovery
  - Full client / server / proxy support
- About 190 lines of code in C

# What Does It Tell Us?

- Simple proposal & simple implementation
- Code changes are mechanical, localized and small
- ID management can be trivial – may simply use a monotonically increasing counter
- Large benefit – removed this long standing protocol limitation!

# Next Steps

- The spec is stable
  - Simple extension
  - Backward compatible
  - No change to packet header encoding
  - No change or overloading of existing fields
- Request RADEXT WG adoption