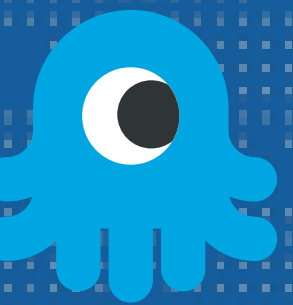


RAD-EXT-RA

IT JUST WON'T GO AWAY

ALAN DEKOK IETF 115



WHY RADIUS?

- ▶ If Diameter exists, why do people still use RADIUS?
 - ▶ RADIUS is “good enough” for most purposes
- ▶ Diameter equipment is \$\$\$, RADIUS is \$
- ▶ Diameter is used in 3G/4G/etc.
 - ▶ RADIUS is used in WiFi, enterprise, university, Eduroam, OpenRoaming, ISPs
- ▶ Diameter is simply *not a choice* for most situations



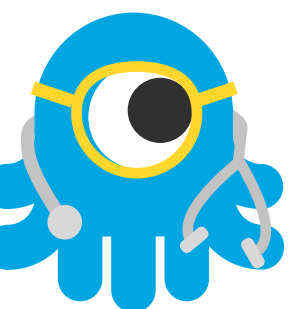
WHAT'S WRONG WITH RADIUS?

- ▶ Other than “almost everything”
- ▶ Security
- ▶ Scalability
- ▶ Features
- ▶ MD5. Enough said.
- ▶ 8-bit IDs are very 1993
- ▶ Credit control, kicking users



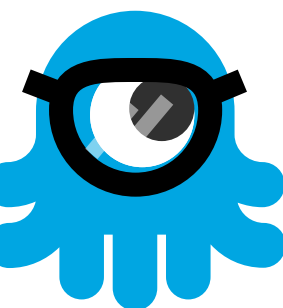
WHAT NOW?

- ▶ Patch it. No one wants a new protocol.
- ▶ Minor changes to code
 - ▶ (1K LoC, not 100K LoC)
- ▶ Work within existing operational models
- ▶ Fix security issues.
- ▶ Backwards compatible



CURRENT PROPOSALS

- ▶ Move RADIUS/TLS and RADIUS/DTLS to “Standards” track
- ▶ Deprecate RADIUS/UDP and RADIUS/TCP
- ▶ Help roaming operators (best practices, ping, traceroute, roam routing)
- ▶ RADIUS without MD5
- ▶ Extend the 8-bit ID space
- ▶ reverse CoA to work around NAT / FW issues



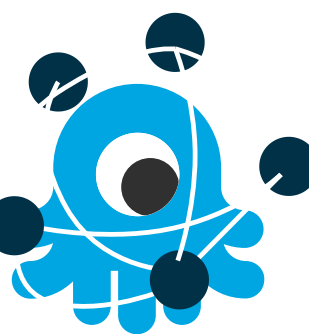
IMPLEMENTATION STATUS

- ▶ RADIUS/TLS and RADIUS/DTLS
- ▶ RADIUS without MD5
- ▶ Status-Realm
- ▶ Extended ID
- ▶ Reverse CoA
- ▶ Widely implemented and used
- ▶ on GitHub, ~2K patch
- ▶ on GitHub, ~1K patch
- ▶ nothing
- ▶ Shipping ~1yr in Aruba, Cisco, and FR

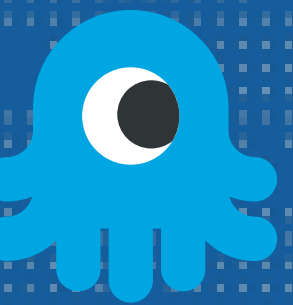


NEXT STEPS

- ▶ Questions?



Deprecate UDP

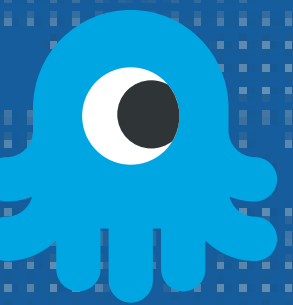


DEPRECATE RADIUS/UDP AND RADIUS/TCP

- ▶ MD5 has been cracked.
 - ▶ Given a RADIUS packet, a hobby attacker can crack all 8-character shared secrets in a short period of time.
- ▶ Sensitive data such as device information, personal location is sent in the clear
- ▶ Just use TLS.
 - ▶ Mandate TLS-PSK
 - ▶ Add text around TLS missing from RFC 6614.

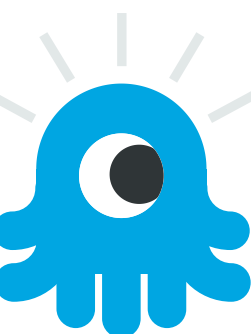


SRADIUS



SECURE RADIUS - SRADIUS

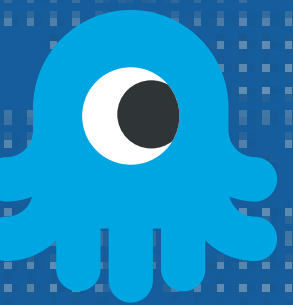
- ▶ A new transport protocol for RADIUS
- ▶ Requires TLS, and changes packet signing to not use MD5
 - ▶ User-Password etc. are encoded as strings, protected by TLS.
 - ▶ Message-Authenticator is ignored
 - ▶ CHAP, MS-CHAP, etc. can still be transported
- ▶ Mandates TLS 1.3 and TLS-PSK.



SRADIUS - REUSING AUTHENTICATOR

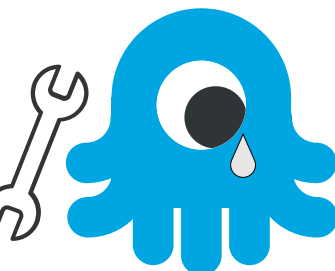
- ▶ 16-octet unused field in the packet header
- ▶ Add 64-bit request / reply token (extended ID)
- ▶ Add flag saying “Require secure transport for this packet”
- ▶ Implemented in GitHub branch. ~2K diff

Extended ID

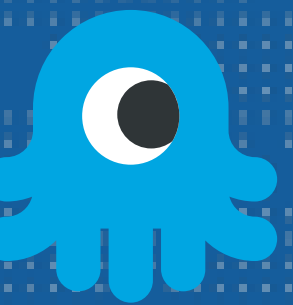


EXTENDED ID

- ▶ Just use Authenticator as unique ID for RADIUS packets
 - ▶ It's already globally / temporally unique!
- ▶ Needs replies to contain Original-Request-Authenticator attribute
- ▶ Lots of text around negotiation and signalling
- ▶ Not implemented
 - ▶ Maybe just use SRADIUS?



Reverse CoA



REVERSE COA

- ▶ NAS is unreachable due to FW / NAT, so sending CoA is impossible
- ▶ But... we have a RADIUS/TLS connection from NAS -> server!
 - ▶ Let's just use that
- ▶ Local network
 - ▶ Server magically "knows" what the NAS is based on TLS session information
 - ▶ Perhaps use NAS-Identifier, etc. from Status-Server to correlate with CoA



PROXYING REVERSE COA

- ▶ Just use Operator-Realm as per RFC 8559
 - ▶ Server magically “knows” what the realm is based on TLS session information
 - ▶ Or via static configuration
- ▶ Other than that, pretty much everything is just
 - ▶ “RFC 5176 and RFC 8559, but using inbound RADIUS/TLS connections”
- ▶ Implemented and shipping in Aruba, Cisco, FreeRADIUS

