#### IETF 90 – RADIUS Extensions WG Meeting Toronto, 23 Jul 2014 (remote)



## draft-winter-radext-populatingeapidentity

# **Document Overview**



- So you've configured multiple EAP types on a client.
- And you think EAP type negotiation will figure out which one works for any given authentication attempt.
- You're wrong.
- draft-winter-radext-populating-eapidentity lists conditions in which EAP behaviour may not be how you expect it to be ; and suggests workarounds.
- In particular, the EAP-Response/Identity which gets copied into a RADIUS or Diameter User-Name needs special care.

### Example setup



| SUPPLICANT                        |  | EAP SERVER               |
|-----------------------------------|--|--------------------------|
| EAP-AKA'<br>User = 1234@3gpp.org  | (+ possibly PROXY)<br>RADIUS Access-Request<br>User-Name = | EAP-AKA'<br>(supported?) |
| EAP-TTLS<br>User = anon@rälm1     |  | EAP-TTLS<br>(supported?) |
| TEAP<br>User = anon@realm2        | RADIUS Access-Request<br>User-Name =                       | TEAP<br>(supported?)     |
| EAP-PWD<br>User = realname@realm3 |  | EAP-PWD<br>(supported?)  |

« Pick any of those usernames for EAP-Response/Identity, EAP type negotiation will figure out which client-side identity matches which supported EAP type on the server » ? FAIL !

# FAIL, Pt. 1



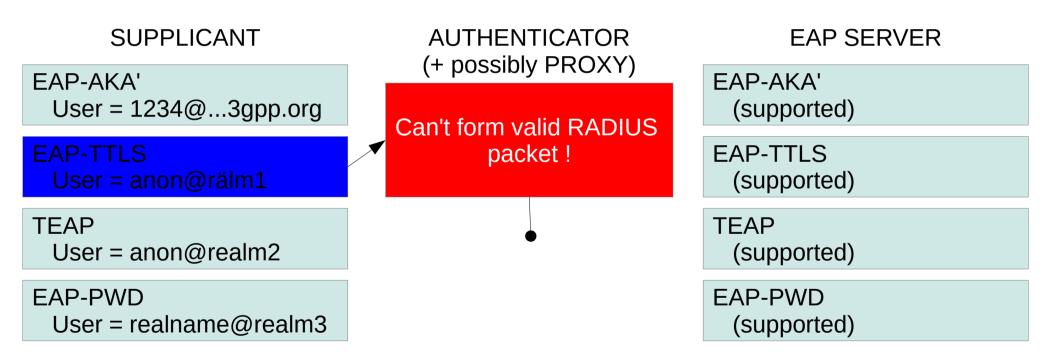
| SUPPLICANT                        | AUTHENTICATOR<br>(+ possibly PROXY)                                    | EAP SERVER              |
|-----------------------------------|--|-------------------------|
| EAP-AKA'<br>User = 1234@3gpp.org  | RADIUS Access-Request  |                         |
| EAP-TTLS<br>User = anon@rälm1     | User-Name =<br>1234@3gpp.org   | EAP-TTLS<br>(supported) |
| TEAP<br>User = anon@realm2        | RADIUS Access-Reject<br>(sorry, we don't do<br>business with 3gpp.org) | TEAP<br>(supported)     |
| EAP-PWD<br>User = realname@realm3 |  | EAP-PWD<br>(supported)  |
|                                   |  |                         |

Supplicant connects to an authenticator which would get him authenticated via TTLS, PEAP, or PWD.

If supplicant chooses to send the « wrong » username, no EAP type negotiation will ever take place  $\rightarrow$  DoS for the user, in spite of having valid credentials.

# FAIL, Pt. 2





Supplicant chooses username from one EAP-Type ; not UTF-8 encoded (and doesn't have to be). Sends it in EAP-Response/Identity Authenticator drops – malformed. Username might have worked for EAP server – inside TTLS tunnel. Never gets this far though  $\rightarrow$  DoS for user, inspite of having valid credentials.

# Solution



- Twofold :
- When EAP terminates with failure, check if more usernames from other configured EAP types are availab and not tried yet.
  - If yes, re-start EAP state machine and try that username silently.
  - If no, failure is final, inform user.
- When using username from an EAP type, convert to UT 8 if necessary when populating EAP-Response/Identity.