GUAM – Generally Useful Authentication Mechanisms

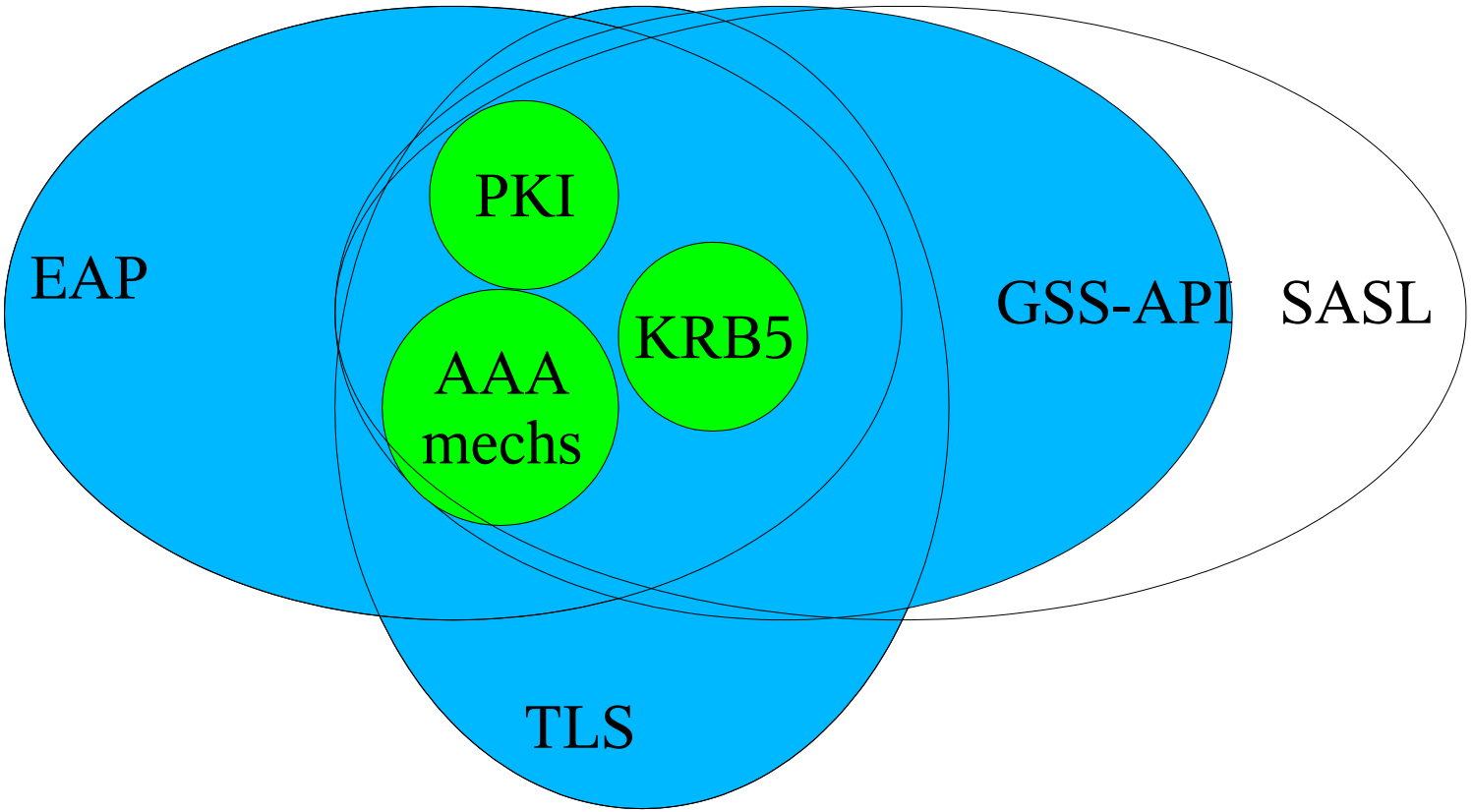
An argument for GUAM, outlines for proposals

nicolas.williams at sun.com
jaltman at secure-endpoints.com

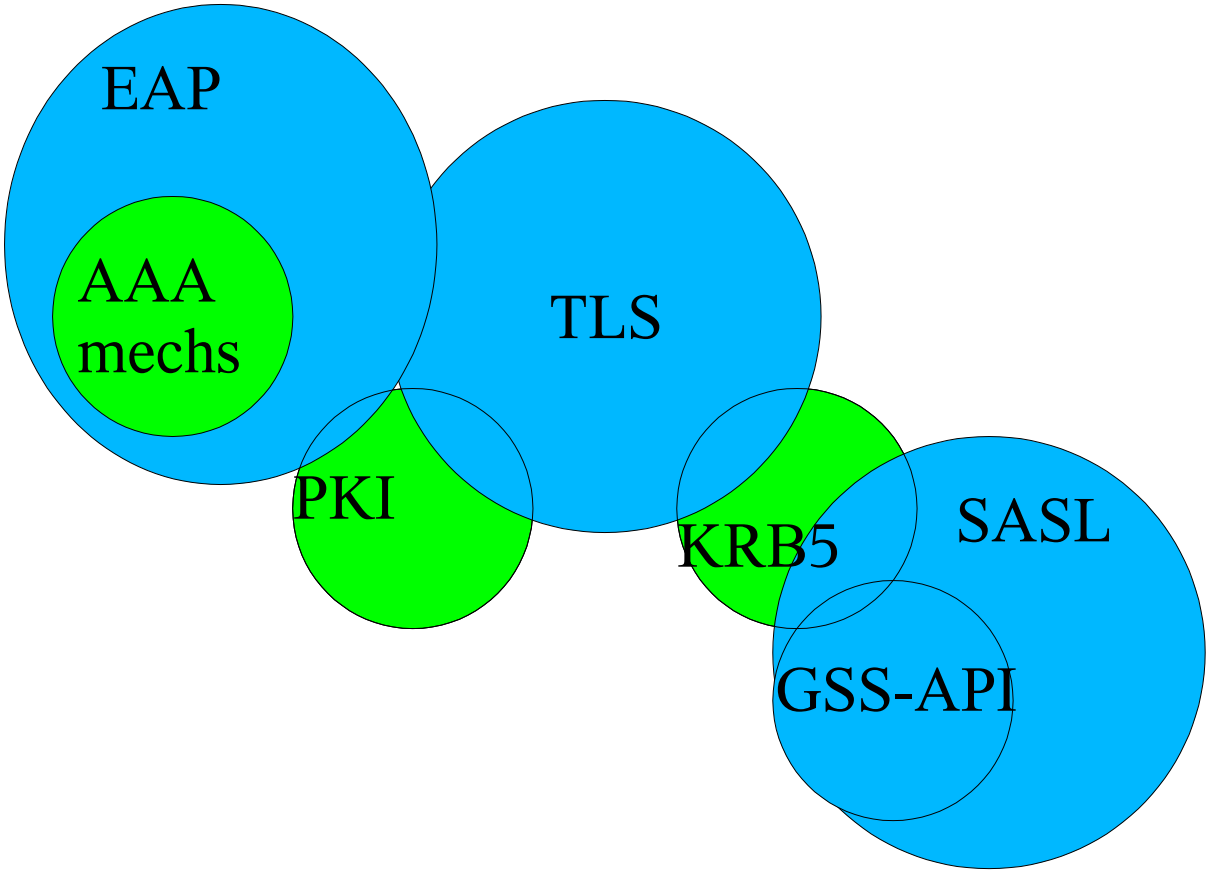
Who, What, When, Where

- SECMECH BoF is an attempt to address a serious IETF problem:
  - That Internet authentication/security frameworks and mechanisms form disjoint sets, and so application protocol developers tend to pick frameworks to use according to mechanism availability rather than framework applicability
- GUAM is the proposal, or set of proposals, rather, to fix this

What We'd Like to Have



But... Behold the Status Quo



A Bit of Terminology?

- Framework
- Mechanism
- Framework-specific mechanism
- Framework bindings of a mechanism
- Framework bridging

- GUAM is all about the last two items

Concrete Proposals

- A GSS-API mechanism based on DTLS
  - Replace SPKM
- GSS-API PLAIN/AAA stackable mechanisms
  - Stack above KRB5, DTLS – replace LIPKEY
- TLS v1.x (1.2?) – support multi-round-trip mechs
  - Then add TLS GSS-API ciphersuites
- IAKERB, EAP-IAKERB?
- *cont.*

Concrete Proposals for EAP/GSS

- Bridging in one or the other direction likely to cost a round-trip, sort of, due to EAP's server-initiated nature
- So requiring that all new mechanisms be native to one or the other of EAP or the GSS-API frameworks then bridged to the other may turn out to be a non-starter (but we don't know this yet)
- So...

Concrete Proposals for EAP/GSS

- ...revive and complete EAP-GSS (GSS->EAP bridge) allowing for use of GSS-API mechanisms as EAP mechanisms
- Allow new EAP-native mechanisms,
  - but require that corresponding GSS-API mechanisms be specified concurrently as well?
- Anyways, EAP conversations can be modelled as GSS-API conversations given some GSS extensions to deal with MSK/EMSK, naming and other differences

Concrete Proposals for EAP/GSS

- EAP/GSS convergence started several meetings back, when the need for EAP keying from any EAP method based on GSS came up
- Some KITTEN WG work is making it easier to build EAP methods out of GSS
  - GSS_Pseudo_random(), specifically
- So a bridge from GSS to EAP is starting to happen; let's accelerate the process!

Work Distribution

- Who should do what?
  - Base work in a SECMECH WG? or RG?
  - TLS WG should do TLS GSS cipher suites and GSS-DTLS mechanism
  - GSS extensions for EAP to KITTEN WG
  - GSS PLAIN/AAA stackable mechs to AAA WG
  - IAKERB to KRB WG; EAP-IAKERB to??
  - Mechanism standardization? Continue with individual submission approach? Or place in a WG? SECMECH?