IPsec APIs

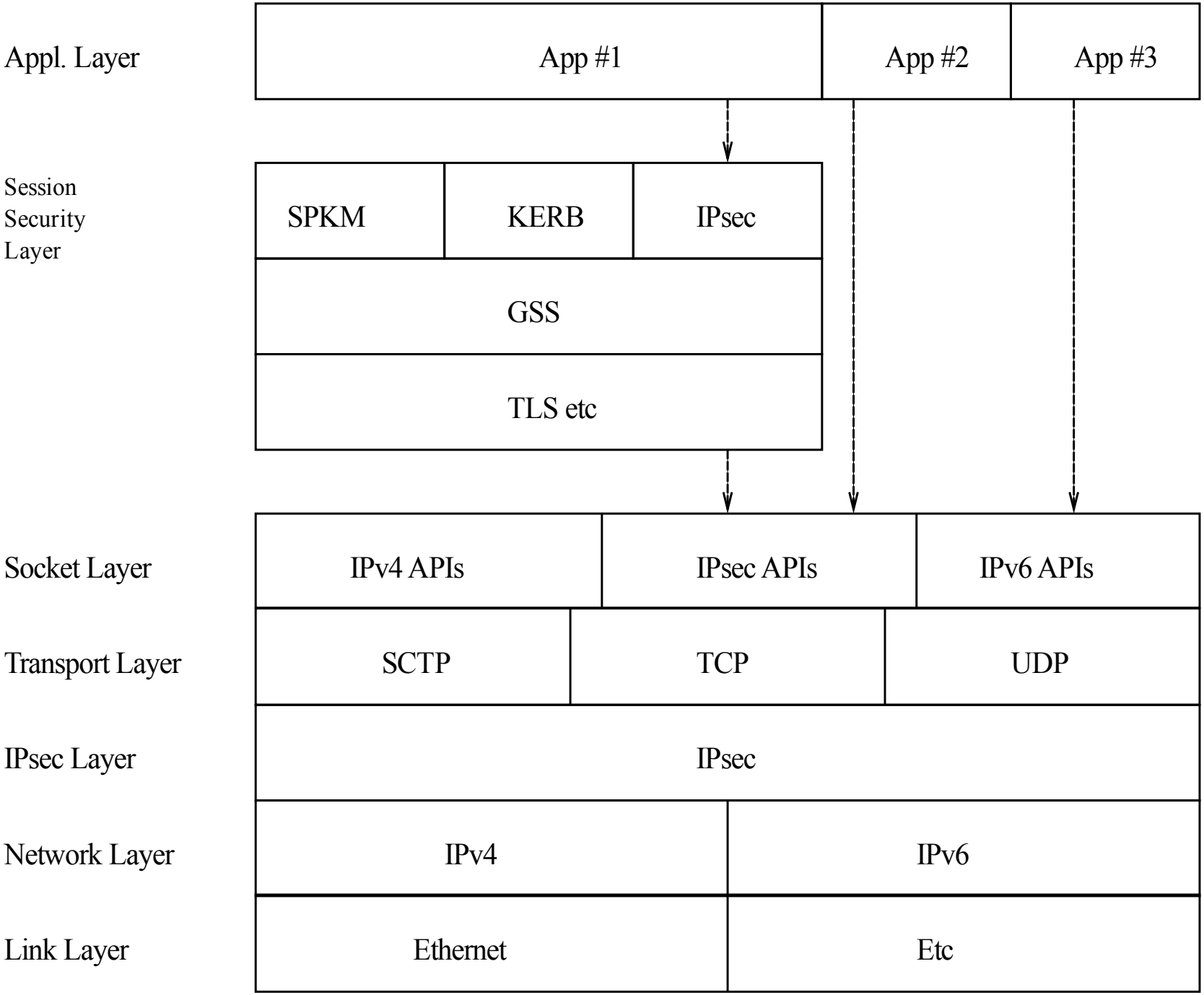
Miika Komu <miika@iki.fi>
6.11.2006
Helsinki Institute for Information Technology

Motivation

- How does application know that its connection is secured using IPsec?
  - → Increase IPsec visibility for applications
- Multiple levels of security (TLS+IPsec)
  - – Good for security critical environments
  - – Bad for performance (double authentication)
  - – Bad for management (double ACLs)
  - → Make IPsec authentication optional

BTNS Use Cases

- The “complete” API is currently spread with HIP and BTNS working groups
  - SHIM6 and HIP share also a multihoming API
- Use cases specific for BTNS:
  - Application #1: uses both TLS and IPsec explicitly (based on GSS)
  - Application #2: uses only IPsec explicitly (based on sockets API extensions)
  - Application #3: uses only IPsec implicitly (based on current sockets API)


Appl. Layer
App #1
App #2
App #3
Session
Security
Layer
SPKM
KERB
IPsec
GSS
TLS etc
Socket Layer
IPv4 APIs
IPsec APIs
IPv6 APIs
Transport Layer
SCTP
TCP
UDP
IPsec Layer
IPsec
Network Layer
IPv4
IPv6
Link Layer
Ethernet
Etc

API Design Details

- Native IPsec APIs
  - New protocol independent PF_SHIM family
  - New abstraction mechanism called end-point descriptor for future extensions
    - Used in place of addresses
  - get/set interface: getsockopt()/setsockopt()
  - Event API: use ancillary data of sendmsg() and recvmsg() similarly as in SCTP

API Design Details

- TLS+IPsec API based on GSS (mostly TBD)
  - The GSS APIs are built on top of the native IPsec APIs
  - Interested in co-authoring?

Questions

- Accept as an official wg draft?
- Move common part for HIP and BTNS from draft-shim-native-api to this draft?
- **Please use the mailing list, not the mike! Thank you.**