Report on SEVECOM Architecture Status

Frank Kargl – Ulm University (frank.kargl@uni-ulm.de)
Antonio Kung – Trialog (antonio.kung@trialog.com)

ulm university universität ulm

TRIALOG

Safety

Information Society and Media

SEVECOM is Technology Driven

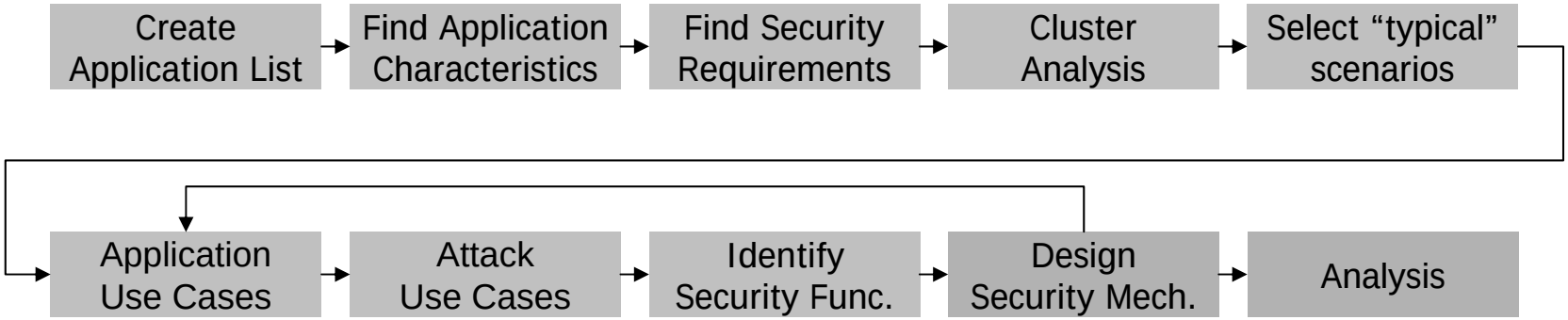
- Addresses the following research topic

	Topic	Scope of work
A1	Key and identity management	Fully addressed
A2	Secure communication protocols (inc. secure routing)	Fully addressed
A3	Tamper proof device and decision on cryptosystem	Fully addressed
A4	Intrusion Detection	Investigation work
A5	Data consistency	Investigation work
A6	Privacy	Fully addressed
A7	Secure positioning	Investigation work
A8	Secure user interface	Investigation work

SEVECOM Architecture Work

- Four versions of deliverable D2.1
  - v1 December 06, v2 June 07, v3 December 07, v4 June 08
- Content of v1 (Available to COMeSafety)
  - SEVECOM architecture design process
  - SEVECOM understanding
    - Relationship with Frame
    - Relationship with GST SEC security architecture
  - Baseline approach

Architecture Design Process



- Starting with applications and general characteristics
  - Analyzed > 50 different applications
- Identified security requirements based on this understanding
- Cluster Analysis 8 application clusters, selected 10 example applications
- Detailed application and attack use cases
- Identified 26 security functions that need to be
  - designed
  - implemented
  - integrated into overall system

Example applications

- SOS Services
- Stolen Vehicles Tracking
- Map Download/update
- Intersection Collision Warning
- Vehicle-based Road Condition Warning
- Electronic License Plate
- Road Surface Conditions to Traffic Operation Centre
- Software Update/Flashing
- Emergency Vehicle Signal Preemption
- Work Zone Warning

- Analysis showed that these match the C2C-CC application list quite well

Security Functions

- Identification & Authentication Concepts
  - ***Identification***
  - ***Authentication of sender***
  - ***Authentication of receiver***
  - Attribute authentication
  - Authentication of intermediate nodes
- Privacy Concepts
  - ***Resolvable anonymity***
  - ***Total anonymity***
  - Location obfuscation
- Integrity Concepts
  - ***Integrity protection***
  - ***Encryption***
  - Detection of protocol violation
  - Consistency/context checking
  - Attestation of sensor data
  - Location verification
  - Tamper-resistant communication system
  - DRM
  - Replay protection
  - Jamming protection
- Access Control/Authorization Concepts
  - Access control
  - Closed user groups
  - Firewall/Checkpoint
  - Sandbox
  - Filtering (e.g. at intermediate nodes)

Relationship with FRAME/GST-SEC

- Frame architecture design process not appropriate
  - user-driven
- SEVECOM architecture design process defined
  - threat/attack driven
- Resulting security features are part of Frame specification
  - general performance, quality requirements and constraints specification.
- Security involves some functional aspects (e.g. privacy) that should be included in Frame
- Add to GST architecture specific aspects for secure communication
  - does not address C2C communication
  - does not address privacy
- SEVECOM to consider GST SEC as a starting point

Security Baseline Architecture

- Should we develop a solid and easy to implement security system or a more fancy version with lot of academic features?
  - Baseline vs. extended security system
- Objectives of baseline approach
  - Focus on communication
  - Well-understood security mechanisms
  - Future dynamic deployment of stronger security mechanisms
- Baseline solution design approach
  - Standardized cryptographic primitives
  - Easy-to-implement
  - Low overhead
  - Adaptable protection

Open Questions

1. How should we determine what security applications are needed by which application?

➔ Proposed solution: Security architecture which is

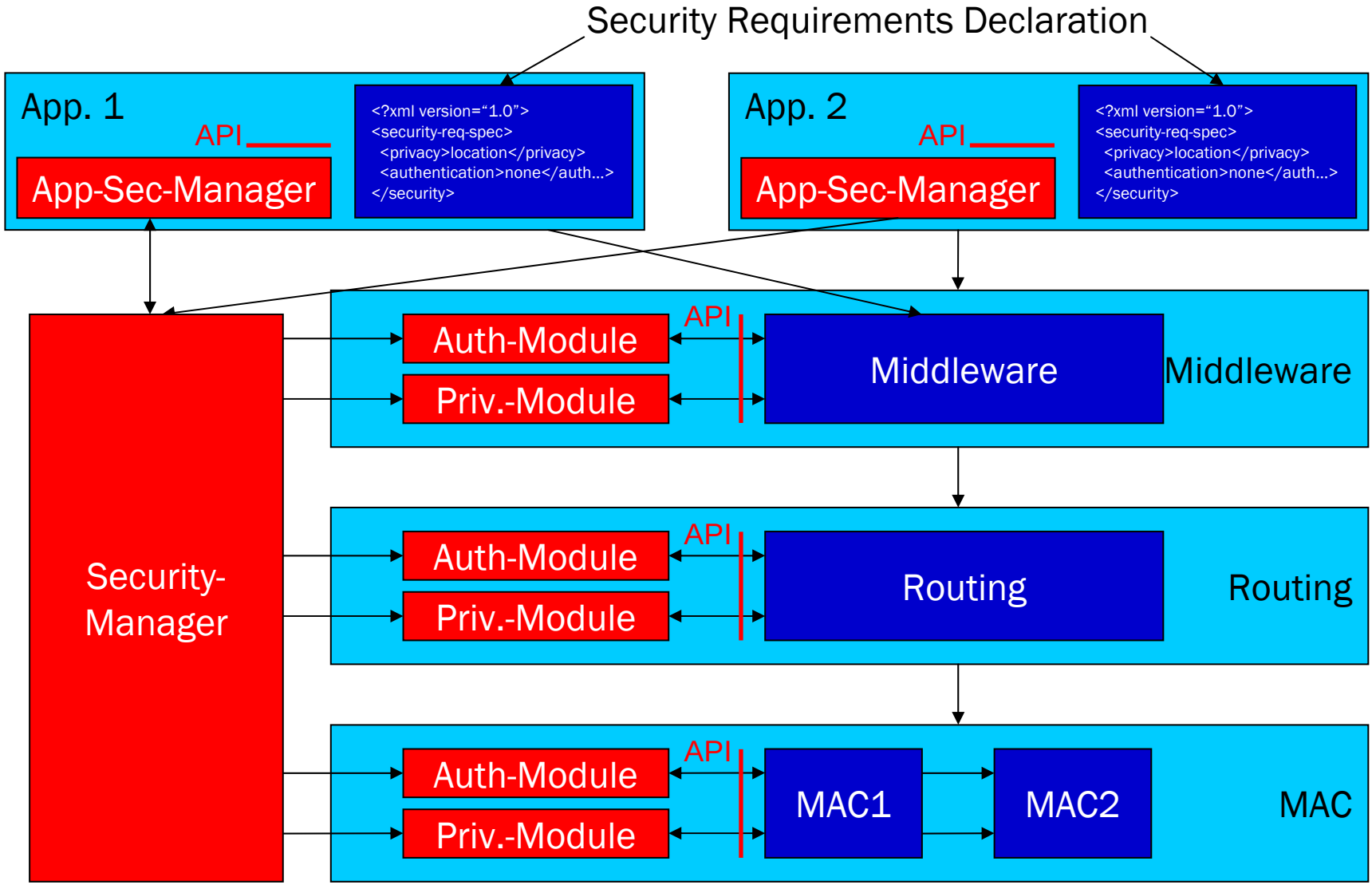
- Modular
- Extensible
- Dynamically configurable at runtime
- Security should degrade slowly when components are not present

2. How can the security mechanisms be integrated with the other functional components?

➔ Proposed solution: Hooking Approach

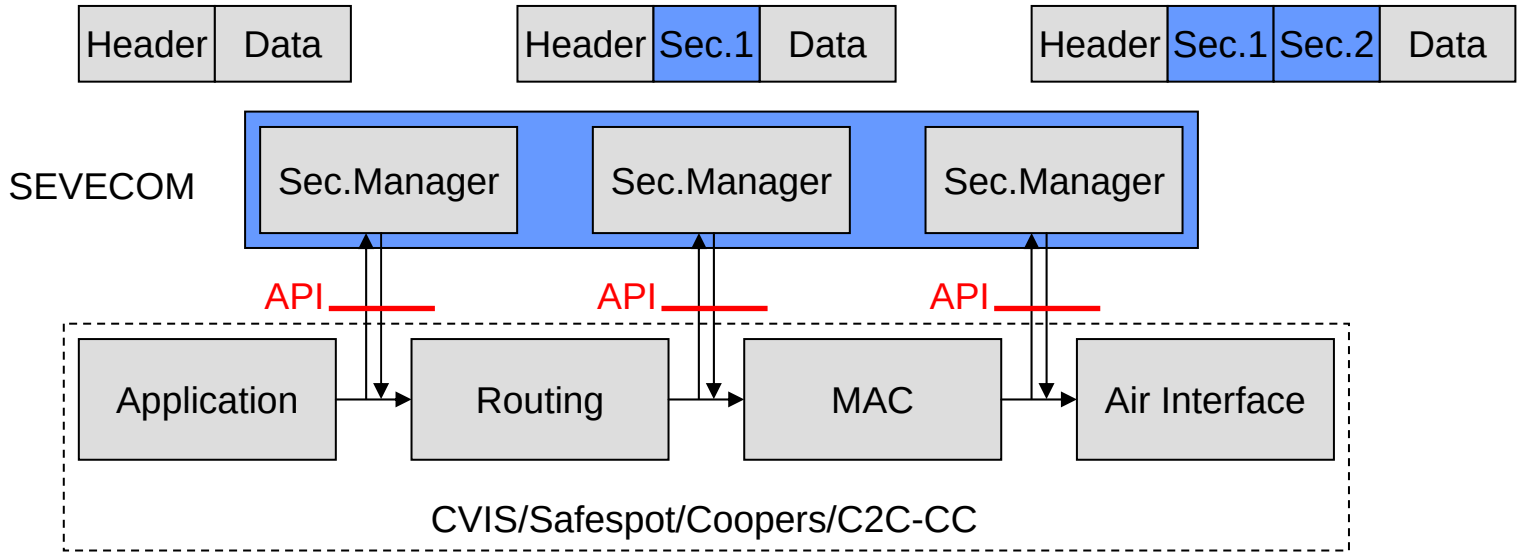
- Communication infrastructure allows registration of callbacks at specified hooks, security modules can analyze, modify, and even drop packets at defined hooks

SW Architecture Proposal



Integration with other components

- How to combine security modules and other functionality?
  - Communication infrastructure allows registration of callbacks at specified hooks, security modules can analyze, modify, and even drop packets at defined hooks
  - Security headers can be attached
  - Similar to Linux netfilter architecture



Thank You

http://www.sevecom.org/