

Position statement – technology requirements (PAMPAS)

(Disclaimer: these thoughts are just some of my ideas for research into combined mobility/security management architecture/protocols and not really a "position statement")

Network layer mobility and security management in heterogeneous (IP based) network environments

One of the challenges one faces when designing security architectures for open heterogeneous systems is that one often wants to be able to execute the security procedures globally (end-to-end).

The general availability of IPv4 now means that one technically can perform all security services end-to-end at the network layer; it is often a poor choice to do so for systems where one can expect a high degree of "mobility".

The problem is partly associated with the fact that security protocols traditionally expects a stable environment where one optimizes for throughput at the expense of a costly set up procedure (multiple passes etc). The other main cause is the inability of the network layer mobility management protocols to take into account **a**) the characteristics of the physical (topology of the) network and **b**) the physical realities (as seen on layer three) of the radio environment (high bit error rates, variable delays, variable throughput ...).

If one accepts that it may be desirable to execute security procedures end-to-end and if one also accept that mobility management should be generic and transparent to the user, then it seems that a network layer solution is nevertheless the best alternative.

It is my firm belief that good performance and user transparency is absolutely essential to general acceptance and adoption of mobility management protocols. So, one clearly must optimize the mobility management protocol for performance. This would in all likelihood mean that the network layer would have to be able to take QoS-type of hints from the lower layers in order to react fast and appropriately to the varying conditions on the lower layers. These "hints" should be fairly generic yet they will have to be specific enough for the network layer to base its decision on the hints. One would also have to acknowledge that the network layer of an autonomous mobile node cannot really gain a good understanding of the current state of the local environment unless there is some assistance from the network being made visible to the network layer of the mobile node¹. One particularly important feature for a network layer mobility management mechanism would be to coexist peacefully with lower layer mobility management mechanism in order to get the best possible performance.

An open network layer architecture presents a challenging and hostile environment for the mobility management protocol. One must obviously be able to protect the integrity of the protocol (and accompanying architecture) and provide essential security services like

¹ Assuming here that it is desirable to let the network layer mobility management control function be located at the mobile node. While the assumption can be defended, it is by no means the only available option. One can easily envisage scenarios where the control function is located in the network. "network" in this context may mean a operator owned infrastructure (with subscription required for access, access by means of micro-payment, or free access), but it can also be an "anarchistic" ad hoc type of network or a hybrid of these.

authentication, authorization, accountability, user data confidentiality and integrity. One must also realize that there is often be a conflict between the requirements for a fast mobility management protocol and the requirements for security. It is therefore imperative to design the mobility management mechanisms and the basic security mechanism together. The preparation steps for a handover would then also include preparation for transfer of the security context to the target access point. This would enable the security context to be immediate available to the mobile node when it arrives at the new access point, and one may then have (almost) continuous service seen from the users perspective. *(This means that one will have to abandon the traditional “internet” way of doing things only when needed (just-too-late), and go for a more aggressive strategy (not unlike the strategy often chosen for cellular networks) were one actively prepares for new events.)*

Needless to say, to develop an integrated security/mobility management protocol/architecture for the network layer is an ambitious undertaking. There are numerous pitfalls and a lot of hard problems that is bound to surface for such design. It is therefore important that one takes into account the lessons learned over the years when it comes to both mobility management and security for mobile environments.

Protocols are hard to design. Protocols with (mobility management) real-time requirements are harder to design. Protocols with both real-time and security requirements are very hard to design.

To make sure that the design is sound one certainly needs good design tools. It is essential that one can feel fairly confident about the correctness of the protocols and the associated security procedures. Of particular interest here are temporal logic based model checking tools (SPIN/TLA) that can verify protocol correctness². One would also need to verify that the security procedures are robust against active and passive attacks. Here again one will likely benefit from using logic tools (TLA/BAN) to prove fundamental properties. While automated model checking tools and logic tools cannot guarantee correctness, the level of confidence attained for protocols that have been subjected to the rigueur of formal methods is comparatively higher than for protocols designed by less rigorous methods.

/Geir M. Kjøien, Telenor R&D

² The correctness proofs are almost always limited to a very small subset of the entire set of possible proposition that one ideally would have to verify to claim that the protocol is “correct”. However, this does not diminish the usefulness of model checkers as design tools.