3GPP TSG SA WG3 Security — S3#22 **S3-020125**

25 - 28 February 2002

Bristol, UK

Source: **Qualcomm**

Title: **Comments on draft EAP/SIM**

Document for: **Discussion**

Agenda Item: **IMS (7.3)**

TDOC S3-010663, draft-haverinen-pppext-eap-sim-02.txt, describes a mechanism whereby multiple GSM-AKA authentication triplets can be combined to create authentication responses and encryption keys of greater strength than the individual GSM triplets. Qualcomm has reviewed this document and has the following comments. The mechanism as proposed appears **adequately secure** for the purposes described, none of the comments below are expected to be a problem in practice.

NONCE_MT

The draft states that NONCE_MT is „picked up by" the client. We find this wording confusing but assume that it means that it is "chosen by" the client. It is stated that NONCE_MT should be a random number. It is not stated that it should be chosen freshly for each EAP session; while people who understand the meaning of the term "nonce" (a number used only once) will assume this, it would appear that a compliant implementation could choose NONCE_MT *only once* and use it for all further interaction.

Devices such as wireless LAN cards and mobile phones are often lacking a good source of random numbers, which might make reuse of NONCE_MT appear attractive. Also, some threat models might consider that the MT could be hostile to the SIM, and could choose NONCE_MT to be constant to give an advantage to an outside attacker (although such a hostile MT already has direct access to the SIM so it is unclear why it would do this instead of more directly revealing the RES and Kc corresponding to particular RAND values).

We raise this issue because the draft states in section 13 that "using a one-way function to combine the keys, we are assured that even if an attacker manages to learn one of the EAP/SIM session keys, it doesn't help him in learning the original GSM Kc's". If NONCE_MT is held constant over a large number (more than 2^{32}) of EAP interactions, there is a birthday-paradox based attack that appears to lead to the ability to recover individual Kc's with 2^{64} time and space when n=2. Thus it is the case that **NONCE_MT should vary**, and so we feel that the wording of the draft should be made stronger in this area (Section 3).

Number of GSM triplets

The draft does not specify an upper limit to the number of GSM triplets to be used. The recommended lower limit is two. We feel that allowing *n* (the number of triplets) to be greater than 3 provides no extra security, but might lead to a false confidence. In fact, the only reason to even allow *n == 3* is that many fielded SIM cards only provide 54 bits of entropy in the output Kc, so combining two Kc's only gives 108 bits, not the desired 128 bits.

Size of derived keys

Section 13 of the draft does not specify the size of K_master, but since it is the output of a SHA-1 hash it is assumed to be 20 octets. Its actual entropy is only 128 bits though, since it is derived algorithmically from information disclosed to a hypothetical attacker, and the SIM key K. The "extra bits" are useful only to the extent that they depend on the NONCE_MT, and hence provide additional security against precomputation attacks, which are in practice infeasible at 2^{128} anyway.

A similar comment applies to K_randsres and K_int (and any EAP specific derived application keys). Note that the derivation of K_master already defeats any form of precomputation attack, however. We feel that setting the length of these output keys to 20 octets is:

- Misleading, in that they only have 128 bits of entropy
- Counterproductive, since in many cases they will be used with algorithms such as Rijndael which accept 128 bit keys, but not 160 bit keys.

We feel that the draft should be more “up front” in disclosing that the ultimate strength of any derived secret key material is no more than 128 bits. In particular, we feel that K_randsres and K_int should be specified to be 16 octets, and not 20 octets, for this reason.

Conclusion

The draft achieves its security objectives. We hope that the above comments are useful.