Security on the Internet, summer 2007

MICHAEL NÜSKEN, DANIEL LOEBENBERGER

11. Exercise sheet
Hand in solutions until Thursday, 28 June 2007.

Exercise 11.1 (Tradeoffs). (8 points)

(i) What are the possible trade offs between performance and security in IPSec? In particular, consider the protocols AH, ESP and IKE v2 as well as the use of different primitives used (Diffie-Hellman groups; encryption algorithms 3DES-CBC, AES-CBC, AES-CTR, DES-CBC; authentication algorithms HMAC-SHA1, AES-XCBC-MAC). [6]

(ii) Now assume that your computing power is infinitely higher than your transmission power. Which trade offs are still there? Discuss carefully. [2]

Exercise 11.2 (1999 IPsec criticism). (8 points)

(i) At `http://www.schneier.com/paper-ipsec.html` you find the IPsec and IKE v1 criticism by Bruce Schneier and Niels Ferguson. Read and summarize it. (What are their recommendations? What are their major reasons? Do they say whether IPsec/IKE is secure or how to make it secure?) [4]

(ii) Reconsider their arguments in the presence of IKE version 2 (that we discussed in the course). [4]

Exercise 11.3 (AtE and died: confidentially poisoned). (12+2 points)

The course raised the paradigm that the integrity of the plain text shall be ensured and offered as one solution to first authenticate and then encrypt (AtE). Though the paradigm is clearly correct and the conclusion grants integrity as desired, we overlooked a different issue here. This exercise shall prove it.

Suppose we use some encryption function ENC_{K_e} and any message authentication function MAC_{K_a}. For a message m we compute $a := \mathrm{MAC}_{K_a}(m)$ and send $c := \mathrm{ENC}_{K_e}(m|a)$. (Here, the vertical line '|' denotes concatenation.)

Assume both are as secure as you like. In particular, the encryption function shall guarantee that even to a *c*hosen *p*laintext *a*ttacker the encryptions of two known plain texts are *ind*istinguishable. In other words, there is no (ie. no probabilistic polynomial time) so-called IND-CPA attacker: the attacker may ask for encryptions of chosen plain texts and he fixes two further plain texts m_0, m_1 for which he never inquired the encryption. Finally, the attacker is given the encryption of m_0 or of m_1 and shall tell which of the two plain texts was used. One possible encryption function under these constraints is the one-time pad (assuming that the encryption procedure keeps track of the already used parts of the key).

Now, suppose additionally that the encryption XORs something on the cipher text (like AES-CTR), and define a variant $\mathrm{ENC}^*_{K_e}$ of this encryption function as follows: first replace every 0-bit by two bits 00 and every 1-bit by two bits 01 or 10, choose randomly each time, next encrypt with ENC_{K_e}. For the decryption we translate 00 back to 0, 01 and 10 to 1, and 11 is considered as a transmission error. So we send $\mathrm{ENC}^*_{K_e}(m|\,\mathrm{MAC}_{K_a}(m))$.

2 (i) Prove (at least, argue) that $\mathrm{ENC}^*_{K_e}$ is still secure in the previous sense.

+2 4 (ii) Suppose that malicious Michael (or hoeing Hugo) has overheard the messages of your login to some server which was done by sending the password. Of course, your password was authenticated and encrypted, as all messages. Now, malicious Michael takes the transmission of your password and resends it with a bit pair in the cipher text inverted.

(a) How does the recipient react if the original bit was 0?

(b) How does the recipient react if the original bit was 1?

Conclude that Michael learns the bit from the reaction of the server (and thus your passwords after enough trials).

2 (iii) Estimate the effect of this observation.

2 (iv) In SSH we transmit $\mathrm{ENC}_{K_e}(m)|\,\mathrm{MAC}_{K_a}(m)$, so we authenticate and encrypt (rather than first authenticating and second encrypting). Is that better? [Try to use $\mathrm{ENC}^*_{K_e}$ here.]

2 (v) In IPsec we transmit $\mathrm{ENC}_{K_e}(m)|\,\mathrm{MAC}_{K_a}(\mathrm{ENC}_{K_e}(m))$. Is the previous attack here also successful? What about the paradigm?