Esecurity: secure internet & e-passports, summer 2014

Michael Nüsken

5. Exercise sheet
Hand in solutions until Monday, 12 May 2014, 13:00

Exercise 5.1 (IPsec and IKEv1 criticism). (8 points)

(i) At `http://www.schneier.com/paper-ipsec.html` you find the IPsec and IKEv1 criticism of Niels Ferguson and Bruce Schneier. 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?) $\boxed{4}$

(ii) Reconsider their arguments in the presence of IKE version 2 (that we discussed in the course). $\boxed{4}$

Exercise 5.2 (AtE and died: confidentially poisoned). (10+2 points)

Horton's principle says that one should always prove the integrity of the *message text*. One solution to ensure the integrity is to first authenticate and then encrypt (AtE). Though this 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 *m*essagetext *a*ttacker the encryptions of two known message texts are *ind*istinguishable. In other words, there is no (ie. no probabilistic polynomial time) so-called IND-CMA attacker: the attacker may ask for encryptions of chosen message texts and he fixes two further message 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 message 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 a one-time-pad), 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 $\mathrm{ENC}^*_{K_e}(m|\,\mathrm{MAC}_{K_a}(m))$.

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

$\boxed{4}$ (ii) Suppose that a ruthless person, called Rudiger, 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, Rudiger 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 Rudiger learns the bit from the reaction of the server (and thus your passwords after enough trials).

$\boxed{2}$ (iii) Estimate the effect of this observation.

$\boxed{2}$ (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.]