Second attempt of Cryptology I tests

January 11th, 2010

Exercises 1–4 correspond to the retry of the mid-term test. Exercises 5–8 correspond to the retry of the end-of-the-term test (a.k.a. final exam). You can attempt either of them or both of them. Please indicate on your submission whether you have retried the mid-term test, the final exam, or both.

1. Break the following ciphertext created from an English plaintext using the affine cipher:

   `f wlsv ehjym l rhgjafhy`

2. Show that if a cryptosystem is unconditionally secure, then $H(\mathbf{K}|\mathbf{P},\mathbf{C}) = H(\mathbf{K}) - H(\mathbf{C})$.

3. Let $\mathbf{R}_f$ be an LFSR with $t \geqslant 2$ registers and maximum possible period. Let f be the feedback function of $\mathbf{R}_f$, i.e. the bits $z_0, z_1, z_2, \ldots$ generated by $\mathbf{R}_f$ satisfy the equality $z_i = f(z_{i-1}, z_{i-2}, \ldots, z_{i-t})$ for $i \geqslant t$. Consider the following boolean function g of t arguments:

   $$g(z_{i-1}, \ldots, z_{i-t}) = f(z_{i-1}, \ldots, z_{i-t}) \oplus (z_{i-1} \wedge z_{i-2} \wedge \cdots \wedge z_{i-t+1}) \enspace .$$

   Consider the non-linear feedback shift register $\mathbf{R}_g$ whose feedback function is g. What is the period of $\mathbf{R}_g$?

4. Consider an encryption scheme that has been obtained from the substitution cipher in the output feedback mode. I.e. given an encryption function σ (a permutation of $\mathbb{Z}_{26}$), a the encryption of some string $x_1 \cdots x_n \in \mathbb{Z}_{26}^n$ is the string $c_0 \cdots c_n$, where $c_0 \in \mathbb{Z}_{26}$ has been randomly generated and $c_i = (x_i + t_i) \bmod 26$, where $t_0 = c_0$ and $t_i = \sigma(t_{i-1})$.

   How would you perform a ciphertext-only attack against this encryption scheme?

5. Let (n, e) be the public key of the RSA encryption scheme. Consider the function $s : \mathbb{Z}_n \to \{0, 1\}$ defined as follows:

   $$s(m) = \lfloor 4m/n \rfloor \bmod 2 \enspace .$$

   Show that if we have access to an efficient procedure that given (n, e) and the ciphertext $c = m^e \bmod n$ returns us $s(m)$, then we can efficiently decrypt.

6. Consider the ElGamal signature scheme in some cyclic group G (where the discrete logarithm problem is hard). Let $m = |G|$ and let g be a generator of G. Let the verification key χ be known to us, but the signing key α be unknown. We also know that in the implementation of the signing functionality, the linear congruential generator has been used to generate the random numbers. I.e. if the random number r was used to generate a signature, then $(ar + b) \bmod m$ will be used as the randomness in the next signature. Let a and b be known to us.

   Describe how we can forge a signature for any message m of our choice. The forging algorithm is allowed to invoke the method *random_sig* that returns a randomly chosen message m' and the signature corresponding to it.

7. The block cipher TEA (*Tiny encryption algorithm*) has 64-bit blocks and 128-bit keys. The "key strength" is at most 126 bits, though, because for each key $K \in \{0,1\}^{128}$ there are three other keys that define exactly the same encryption function. Those keys K', K'', K''' are obtained from K as follows:

   - $K' = K \oplus 10^{31}10^{95}$;
     - $10^{31}10^{95}$ denotes a bit-string where a single bit "1" is followed by 31 bits "0", then by a single bit "1" and finally by 95 bits "0".
   - $K'' = K \oplus 0^{64}10^{31}10^{31}$;
   - $K''' = K \oplus 10^{31}10^{31}10^{31}10^{31}$.

   Find a collision to the compression function

   $$H(x_1, x_2, x_3) = \mathrm{TEA}_{x_1 \| x_2}(x_1 \oplus x_3) \oplus x_2 \oplus x_3,$$

   where x_1, x_2, x_3 and the output of H are bit-strings of length 64.

8. Explain zero-knowledge proofs. What is the purpose of those protocols, what are the security requirements and what is the particular aspect of the security definitions that justify the name "zero-knowledge"?