

## Homework 2

Lecturer: Sasha Boldyreva

Due: March 2, 2006

Recommended reading is Chapters 9-12 from the lecture notes of Bellare and Rogaway and the relevant sets of slides.

**Problem 2.1, 10 points.** A company suggests the following hash function. Let  $p \geq 3$  be a large prime with respect to which the discrete logarithm problem is intractable in  $\mathbb{Z}_p^*$ . Let  $g_1, g_2 \in \mathbb{Z}_p^*$  be two distinct generators of  $\mathbb{Z}_p^*$ . Define  $H: \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1} \rightarrow \mathbb{Z}_p^*$  as  $H(x_1, x_2) = g_1^{x_1} \cdot g_2^{x_2} \bmod p$ .

The company claims that this hash function is collision-resistant. Show that they are wrong. Namely, present an algorithm  $C$  that given any  $p, g_1, g_2$  that satisfy the above conditions, very easily (formally, in time polynomial in  $|p|$ ) outputs  $(x_1, x_2), (x'_1, x'_2)$  such that  $(x_1, x_2) \in \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1}, (x'_1, x'_2) \in \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1}$  and  $H(x_1, x_2) = H(x'_1, x'_2)$ , but  $(x_1, x_2) \neq (x'_1, x'_2)$ . Briefly justify your answer.

**Hint.** Use some facts from number theory.

**Problem 2.2, 10 points.** In class we saw that the ElGamal scheme is not IND-CCA secure. Consider the following attempt to make the ElGamal scheme IND-CCA secure. Fix a cyclic group  $G$  of order  $q$  and a generator  $g$ . Let  $H$  be a public function such that for any message in the message space  $M \| H(M)$  can be viewed as a group element. A proposed HEG =  $(\mathcal{K}, \mathcal{E}, \mathcal{D})$  scheme is as follows.

Algorithm $\mathcal{K}$	Algorithm $\mathcal{E}_X(M)$	Algorithm $\mathcal{D}_x(Y, W)$
$x \xleftarrow{\$} \mathbb{Z}_q$	$y \xleftarrow{\$} \mathbb{Z}_q$	$K \leftarrow Y^x$
$X \leftarrow g^x$	$Y \leftarrow g^y$	$M' \leftarrow WK^{-1}$
$pk \leftarrow X$	$K \leftarrow X^y$	Parse $M'$ as $M \  Z$
$sk \leftarrow x$	$W \leftarrow K \cdot (M \  H(M))$	If $Z = H(M)$ then return $M$
Return $(pk, sk)$	Return $(Y, W)$	else return $\perp$

Note that the attack presented in the slides does not work. Show, that nevertheless the scheme is still IND-CCA insecure even if DDH is hard for  $G, g$ . Assume that an adversary knows  $G, g, q$ .

**Problem 2.3, 10 points.** In class we studied (or will study the very next time) the ElGamal signature scheme (see the slides). In the original version of the ElGamal signature scheme the message space was  $\mathbb{Z}_q$ , there was no hash function, and instead of  $H(M \| Y)$ , the message  $M$  itself was used. Show that the original version is not uf-cma secure.

**Problem 2.4, 10 points.** Make yourself familiar with the cryptographic libraries Java Cryptography Extension (JCE) <http://java.sun.com/products/jce/> OR Open SSL <http://www.openssl.org/>. Use either one<sup>1</sup> to implement the following. Create a public-secret key pair for yourself, and show how an arbitrary file can be encrypted by anyone for you to receive and decrypt.

Your public-key encryption algorithm should be stateless, IND-CCA secure in the RO model under the common assumptions and very efficient even for large files. You are to implement and test the decryption algorithm as well.

I believe that if you search well, and perhaps explore the libraries' extensions, either library should contain every subroutine you need so that you don't have to perform any operations (such as xor or modular exponentiations) yourself. If you believe there is a subroutine that you require but is not provided by the library of choice then you are encouraged to talk to me and see if this is really the case. Actually, it is a very good idea to verify your design with me in any case.

You are to email me your solution to this question before the time of the class on the due day in a form that allows easy verification. You should include your source code, and also a clear pseudo-code description of your scheme, the generated keys, and a sample file and its ciphertext. For convenience you are encouraged to pick any of the pdf files on the class web page.

---

<sup>1</sup>If there is another library you prefer to use please clear it with me first.