McGill

DECEMBER 2008
Final Examination

FINAL EXAMINATION

Computer Science COMP-547A
Cryptography and Data Security

16 DECEMBER 2008, 9h00

Examiner:	Prof. Claude Crépeau	Assoc Examiner:	Prof. David Avis

INSTRUCTIONS:

- This examination is worth 50% of your final grade.
- The total of all questions is 100 points.
- Each question heading contains (in parenthesis) a list of values for each sub-questions.
- This is an **open book** exam. **All documentation is permitted.**
- Faculty standard calculator permitted only.
- The exam consists of 5 questions on 3 pages, title page included.

Suggestion:

read all the questions and their values before you start.

Question 1. Small private RSA exponent (5+5+5+5 points)

I mentioned in class that RSA public-keys **(N,e)** which correspond to small values of **d** ($||d|| < ||N||/4$) are easy to break using an algorithm developed by Wiener. This is unfortunate because it is useful to have small **d** for efficiency of decryption. On page 358 of your book, a small section is dedicated to a technique using the Chinese Remainder Theorem representation of **d** to speed up decryption. I summarize this idea here.

For a triplet of RSA keys **(N,e,d)**, where **N=pq** is a product of two large primes, the secret exponent **d** may be replaced by two much smaller exponents $d_p := d \bmod p\text{-}1$ and $d_q := d \bmod q\text{-}1$. The decryption algorithm $m := c^d \bmod N$ is then replaced by computing $m_p := c^{d_p} \bmod p$ and $m_q := c^{d_q} \bmod q$. The answer **m** is obtained by applying the Chinese Remainder Theorem to (m_p,p) and (m_q,q).

(a) Assuming exponentiation of an **n**-bit number (**c**) modulo an **n**-bit modulus (**N**) with an **n**-bit exponent (**d**) takes time n^3, compare the running time of the direct way to calculate $m := c^d \bmod N$ together with the alternate way to calculate **m** using the Chinese Remainder Theorem (assuming d_p and d_q were pre-calculated).

Assume **(N,e,d)** are carefully chosen so that the related pre-calculated d_p and d_q both have smaller size $k < n/2 = ||N||/2$.

(b) Express the size of exponent **d** related to d_p and d_q, both of size $k < n/2 = ||N||/2$.

(c) Assuming exponentiation of an **n**-bit number (**c**) modulo an **n**-bit modulus (**N**) with an **L**-bit exponent (**d**) takes time n^2L, compare the running time of the direct way to calculate $m := c^d \bmod N$ together with the alternate way to calculate **m** using the Chinese Remainder Theorem (assuming d_p and d_q, both have smaller size $k < n/2$, and were pre-calculated).

(d) If we use very small d_p and d_q, say both of size $k < n/4$, does it seem to reduce the security of the scheme. Explain your answer.

Question 2. $CNE_k(x) := ENC_x(k)$ (5+10 points)

Given a deterministic encryption scheme $ENC_k(x)$, where the key-size and message-size are the same, define another function family $CNE_k(x) := ENC_x(k)$.

(a) Explain why the new function family $CNE_k(x)$ might not even define a valid encryption scheme.

(b) Suppose that for a random half-size string r and arbitrary half-size message **m**, $ENC_k(r{:}m)$ is believed to be secure in the presence of an eavesdropper. What can be said about the security of $CNE_k(r{:}m)$ (assuming $CNE_k(x)$ is a valid encryption scheme) ? Explain your answer.

Question 3. CConlyA (8+5+6+6+5 points)

A cryptosystem is *secure against a Chosen Ciphertext-Only Attack* (**CConlyA**) if the adversary has access to a decryption oracle only (no encryption oracle).

• Define formally "*The CConlyA indistinguishability experiment*" and a security definition along the lines of **Definition 3.30**.

• For public-key cryptosystems argue that **CConlyA**-security is equivalent to **CCA**-security.

• For private-key cryptosystems argue that if **CCA**-security is achieved then **CPA**-security and **CConlyA**-security are both achieved.

• For private-key cryptosystems, if both **CPA**-security and **CConlyA**-security are achieved, can we conclude that **CCA**-security is necessarily achieved ? Explain.

• Why do you think **CConlyA**-security is not seriously considered as a useful notion ?

Question 4. Pretty-Strong Primes (10+10 points)

We have seen in class the notion of Strong primes that are such that **(p-1)/2 = q** is also a prime. We now define the notion of Pretty-Strong prime that are such that **(p-1)/2 = q r** is a product of two primes of the same size.

(A) If I give you a Pretty-Strong prime **p**, is it computationally easy to find a generator (primitive element) of the non-zero integers modulo **p** ? Explain.

(B) Give an efficient algorithm to generate (uniformly) any Pretty-Strong prime **p** of a certain (exact) size **k** and a random generator **g** of the non-zero integers modulo **p**. Explain how it works.

Question 5.  MACs (7+8 points)

In the class notes we have seen that if **F** is a strongly universal-two class of hash functions, the Wegman-Carter one-time authentication scheme $\mathbf{m \rightarrow (m,f_k(m))}$ is perfectly secure, when $\mathbf{f_k}$ is chosen uniformly from **F** for each authentication.

• Explain the relation between the security of this authentication scheme and Definition 4.2 of "*existential unforgeability under an adaptive chosen-message attack*".

• Explain how to combine Vernam's one-time-pad with Wegman-Carter one-time authentication to guarantee both confidentiality and integrity in a perfect way. Reduce as much as you can the amount of key bits necessary to accomplish both properties.