Institut for Matematik og Datalogi
Syddansk Universitet

March 24, 2006
JFB

Computer Security – F06 – Lecture 9

Announcement

The discussion sections on April 3 and April 10 will be at 8:30.

Announcement

Check the homepage for the course to make sure that your project is scheduled properly. If no date is assigned, please contact Peter Kornerup or Joan Boyar.

Lecture, March 24

We covered up through section 11.5.2 of chapter 11, but didn't get as far as cipher feedback mode (CFB).

Lecture, March 31

We will finish chapter 11 and begin on chapter 12.

Lecture, April 7

We will finish chapter 12.

Discussion section: April 3 at 8:30

1. How is DSA similar to the El Gamal signature scheme?
2. What advantages and/or disadvantages does the El Gamal encryption system have over RSA?

3. Consider the table comparing RSA and DSA on page 208. How could verifications for RSA be so much faster than computing signatures? Does the table say anything about which one is better? Note that the table does not compare their relative security.

4. Given that for n bit numbers, the current best factoring algorithms take time $e^{O(\log^{1/3}(n)\cdot\log\log^{2/3}(n))}$, what effect on security do you get by doubling the number of bits in an RSA modulus?

5. Try using gpg.

   - Create a public and private key using `gpg --gen-key`. You should choose DSA and El Gamal, and size 1024. Go to the directory `.gnupg` using `cd .gnupg` and list its contents. Try the commands `gpg --list-keys` and `gpg --fingerprint` to list the keys you have, with the fingerprints, which make it easier for you to check that you have the correct key from someone. How would you use fingerprints?
   - You can save your public key in a file in a form that can seen on a screen using `gpg --export -a Your Name >filename`. You are "exporting" your key and specifying where the output should go. Then look at it using `more filename`; the `-a` made it possible to see it reasonably on your screen, since it changes it to ASCII.
   - Mail this file to someone else.
   - Try to figure out how to use `gpg` to "import" the public key you got from someone else. Check the fingerprint.
   - Create a little file and encrypt it. You can use `gpg -sea filename`. What does this do?
   - Mail your file to whoever has your public key. Read their file using the command `gpg -d inputfile >outputfile`. Then look at the output file you created.
   - You can also encrypt a file for your own use using a symmetric key system protected by a pass phrase. Try using `gpg --force-mdc -ca filename`. Then try decrypting as with the file you decrypted previously. Why might you want to do this?

6. Suppose that user A wants to send a message $s \in \{s_1, s_2, ..., s_k\}$ to user B, where $s_i < 1024$ for $1 \leq i \leq k$. Assume that RSA is secure (when

the modulus is large enough and is the product of two equal length prime factors).

- Why would you still advise user A not to use RSA directly?
- What would you recommend instead, if you still wanted to use RSA?

7. Do exercises 12.1 and 12.2.