

Introduction to RSA

CS681 & CS392 Computer Security
Fall 2005
Due 09/29/2005

1 Objective

The object of this assignment is to develop a mechanism, using RSA and AES together, to extend the functionality of enc and dec commands you implemented in the previous assignment.

2 RSA

You should be familiar with the RSA algorithm. This was covered extensively in class this week.

- You can learn more about RSA from: <http://www.geometer.org/mathcircles/RSA.pdf>
- Download RSA code from: <http://www.funet.fi/pub/crypt/cryptography/asymmetric/rsa/rsaref2.tar.gz>
- Get familiarized with RSA Public/Private key generation, encryption, and decryption functions and the sample code. Execute the sample code and encrypt and decrypt files using it.

2.1 Warm Up

Answer the following questions:

1. What is the fundamental difference between RSA and AES?
2. In terms of computational complexity which algorithm is better RSA or AES? Explain your answer in detail.
3. Alice has selected the public key ($n = 33$; $e = 3$). Using this key, Bob sends plaintext x to Alice which encrypts as ciphertext 10. What is the plaintext x ?
4. Show that RSA is insecure against a chosen ciphertext attack. In particular, given a ciphertext y , describe how to choose a ciphertext y_1 ($y_1 \neq y$), such that knowledge of the plaintext $x_1 = d(y_1)$ allows $x = d(y)$ to be computed, where $d()$ is the RSA decryption function.

2.2 Extending File Encryption

In the last assignment you have developed commands to encrypt and decrypt files using individual AES key. Now you want to provide the ability in FEAU that a given file be decrypted by **multiple users** without passing the AES key around. The goal now is to develop a mechanism that will allow users to share encrypted files, such that only an authorized subset of users can decrypt a given file.

Here is a simple mechanism to solve the above problem.

- UK_{pub} = User's (with whom the file has to be shared) Public Key
- UK_{prv} = User's (with whom the file has to be shared) Private Key
- fk_s = AES file encryption Key

File encryption and decryption process:

1. Generate a n bit random number and use it as fk_s . n is equal to the size of the AES key in use (128, 192, or 256).
2. Encrypt the file F using AES: $E_{fk_s}(F)$

3. Encrypt fk_s using public key of the other user, with whom the file is to be shared, and concatenate it to the encrypted file using a constant string “mark” as a preamble and post-amble. If you want to store the key in a separate file, have a fool-proof way of co-relating between the file you are encrypting, file that stores the key and the user to whom the key belongs.

4. To decrypt a given file a user simply has to decrypt the corresponding $E_{UK_{pub}}(fk_s)$ to get the AES encryption key fk_s and decrypt the file: $D_{UK_{prv}}(E_{UK_{pub}}(fk_s)); D_{fk_s}(F)$.

Note that in this mechanism encryption and decryption of files are done with the efficiency of AES and not RSA.

2.2.1 For this section you should develop and integrate this extended file encryption mechanism into FEAU. You could use your own mechanism if like but discuss it before you do so. Remember however that it should be extensible in that it should be possible to share a file among a “group” of users. You should pre-compute private key and public key pairs for your users and groups. For now, you could test it to work for two users, one who encrypts and the other who decrypts and accesses and we can extend it later to groups.

3 What and how to Handin

Handin: You should submit assignment through my.poly drop box no later then 12 AM midnight on the due date. You must zip all files related to the assignments and use the following convention to name the zip file:
<First Name>_<Last Name>_<Lab#>.zip
Submit using the name of only one of the members in your group.
REMEMBER IF YOU DO NOT USE THIS NAMING CONVENTION YOUR ASSIGNMENT WILL NOT BE GRADED AND YOU WILL NOT RECEIVE ANY CREDIT.

PLEASE SUBMIT HOMEWORK ON TIME.