ECE 646 – Fall 2021

CrypTool – Public-Key Cryptosystems Lab Instruction

Part 1

Implementation and Analysis of RSA in CrypTool 1

Install CrypTool 1 on your laptop or desktop at home. You can download CrypTool 1 from

https://www.cryptool.org/en/ct1

The recommended version of software is CrypTool 1 (CT1) ver. 1.4.41 - English.

1. RSA DEMONSTRATION

Please go through all steps of the RSA Demonstration (available under **Indiv. Procedures => RSA Cryptosystem => RSA Demonstration…**).

Perform this procedure for at least the following sizes of the RSA keys (understood as the size of N in bits):

a) 16 bits (CrypTool default)
b) 2048 bits
c) 4096 bits.

Assume the equal sizes of P and Q.

For each case, record values of

A) all components of a public key
B) all components of a private key
C) message
D) ciphertext.

Try to encrypt 1, 2·Q, N-2·P, and N-1 and see if the results match your expectations. Record and discuss your findings.

Hint: In order to do that you will need to set your Input as numbers (as opposed to text) Please see the corresponding setting of CrypTool in the RSA Demonstration window.

2. FACTORING

Using the beginning of the RSA Demonstration generate values of N for at least the following initial sizes:

a) 120 bits
b) 135 bits
c) 150 bits.

For each size of N, double click on N, and then copy the entire value of N to the Input field in the window Factorization of a Number, obtained by choosing
Indiv. Procedures => RSA Cryptosystem => Factorization of a Number…

For each case, include in the report:

A. value of N
B. values of P and Q obtained after factoring N
C. sizes of N, P, and Q in decimal digits
D. factoring time
E. method used for factoring (listed after clicking on "Details", and then choosing "Save list into main window")

Find experimentally the size of N for which the factoring time is consistently greater than

A. one minute (required)
B. five minutes (bonus).

Then, generate the same size number(s) randomly (e.g., by typing arbitrary digits until the required size is reached). Factor these random numbers and record the same information as in case of N obtained using RSA Key Generation.

Hint: Please keep clicking on "Continue" until the number is fully factored.

Have you noticed any changes in the execution time or method(s) used?

Please explain your findings.

3. GENERATION AND CERTIFICATION OF KEYS

Generate a pair of RSA keys, with the secure size of the key, 3072 bits, using option **Digital Signatures/PKI => PKI => Generate/Import Keys…**

Provide user data required for the generation of a certificate and protecting access to the private key.

Press on "Show key pair", and then on "Show public parameters…" and "Show certificate". Record all generated data.

Repeat the same steps for

a) DSA with the bit length of DSA prime number equal to 3072 bits, and

b) Elliptic curve cryptosystem with the identifier (bit length and curve parameter) prime256v1.

Tabulate and compare the sizes of all system parameters, public keys, and private keys in all three aforementioned cryptosystems.

Explain the meaning of the abbreviation PSE used in CrypTool (Hint: Use Help Index Search).

4. SIGNATURE DEMONSTRATION

Open an arbitrary text document you would like to sign.

Choose option
Indiv. Procedures => RSA Cryptosystem => Signature Demonstration (Signature Generation)…

Follow the demonstration by clicking on active icons.

Use the "Provide certificate" icon to import the RSA certificate and the key.

Record all generated data.

Then, verify certificate using **Digital Signatures/PKI => Verify Signature**

5. MESSAGES NOT CONCEALED BY RSA

A. For a 16-bit RSA key, calculate values of 9 messages that are not concealed by RSA. Verify this special property of these inputs using RSA Demonstration. Document your findings.
B. Generate an RSA key that does not conceal any message. Verify the properties of this key using RSA Demonstration. Document your findings.
C.

6. CLASSICAL vs. PQC SCHEMES

Investigate, tabulate, and explain the differences in terms of

a) Public key sizes
b) Private key sizes
c) Ciphertext sizes

among

1) RSA, 2) ECC, 3) Saber, and 4) Classic McEliece

for the security level equivalent to AES-256.

Part 2 (Bonus)

Implementation and Analysis of RSA in CrypTool 2

INSTALLATION

Install CrypTool 2 (version 2.1, Stable Build) by going to
https://www.cryptool.org/en/ct2/downloads
Get familiar with the options of the program and its On-Line Help.

RSA DEMONSTRATION

Using visual programming available in CrypTool 2, prepare a demonstration of the operation of a hybrid system based on the use of RSA and AES.

AES should be used for the secret-key encryption of messages, and RSA for the exchange of AES session keys.

The demonstration should visualize all major operations performed on the sender's side and the receiver's side, and should allow exchange of medium size messages in English.

The users are assumed to know each other's public keys.

Do your best to support various security levels recommended by NIST.

As a part of your solution, please submit your CrypTool 2 project in an electronic form, and write a short report including screenshots illustrating your project operation on the sender's side and on the receiver's side.

RSA BREAKING

Investigate components available in CrypTool 2 for breaking RSA using factorization. Determine the sizes of keys that can be broken using these tools within a predetermined time limit, e.g. 5 minutes. Document your findings and submit your report, as well as an electronic version of your CrypTool 2 project.