Padding Schemes for RSA and their Security

Calvin Ryan D’Souza

Advisor:- Prof. Stanisław Radziszowski

CONTENTS

- Background - RSA Algorithm
- Motivation and Problem Statement
- Prior Work
- Milestones
- Experimental Plan

BACKGROUND – RSA ALGORITHM

- Rivest-Shamir-Adleman (RSA) Algorithm:
  - Choose 2 large prime numbers (p and q), and calculate ***n*** = p*q
  - Calculate Phi(***n***) = (p-1)(q-1)
  - Choose a number ***e***, such that GCD(e, Phi(***n***)) = 1
  - Choose d such that; *e*.*d* ≡ 1. mod Phi(***n***) i.e., d = e^{-1}

- Now, we can perform encryption and decryption as follows:
  - Encryption:- c = $(m)^{e}$ mod n
  - Decryption:- m = $(c)^{d}$ mod n

MOTIVATION AND PROBLEM STATEMENT

- Since RSA is a deterministic algorithm, it is susceptible to chosen plaintext attacks. There are other mathematical attacks possible on RSA too.

- Padding Schemes are used to help solve this problem. Some examples are:-
  - Public key Cryptography Standards (PKCS) #1 v1.5
  - Optimal Asymmetric Encryption Padding (OAEP)

- Analysis of how padding schemes affect the security of RSA and its performance. Is there another way of making RSA secure?

PRIOR WORK

- RFC for RSA-OAEP:- https://datatracker.ietf.org/doc/html/rfc8017.
- Eiichiro Fujisaki, Tatsuaki Okamoto, David Pointcheval, and Jacques Stern. “RSA-- OAEP is secure under the RSA assumption”.
- Nemec, Matus; Sys, Marek; Svenda, Petr; Klinec, Dusan; Matyas, Vashek (November 2017). “The Return of Coppersmith's Attack: Practical Factorization of Widely Used RSA Moduli”.
- Coron, Jean-Sébastien; Joye, Marc; Naccache, David; Paillier, Pascal (2000). Preneel, Bart (ed.). “New Attacks on PKCS#1 v1.5 Encryption”.

MILESTONES

- **Milestone 1:-**
  - Review literature on RSA and Padding schemes.
  - Implementation of RSA and parameter selection algorithm.
- **Milestone 2:-**
  - Review literature on malleability of RSA.
  - Review literature on probabilistic versions of RSA.
  - Implement padding schemes for RSA.
- **Milestone 3:-**
  - Perform experiments and comment on security and performance of RSA with/without Padding Schemes.

EXPERIMENTAL PLAN

- Implementation of Parameter Selection (in Java):-
  1. Randomly generate n-bit odd number ***p***.
  2. Perform primality tests to check if ***p*** prime.
  3. If prime, check if ***(p - 1) = 2 * (prime)***. Else go to Step 1.
  4. If yes, choose number for RSA. Else go to Step 1.

- Implementation and testing for RSA (with/without Padding schemes) will be done in Java.

ANY QUESTIONS?

THANK YOU