A Signcryption Scheme Based On Integer Factorization

Ron Steinfeld

ron.steinfeld@infotech.monash.edu.au

Yuliang Zheng

yuliang.zheng@infotech.monash.edu.au

Laboratory for Information and NetworK Security
School of Network Computing
Monash University
Frankston 3099, Australia
http://www.netcomp.monash.edu.au/links/

Overview

- Introduction
- Related Past Schemes
- The New Signcryption Scheme
- Efficiency of New Scheme
- Setup by Trusted Authority
- Security Analysis
- Conclusions

Introduction

- Many applications need both:
  - 1. Message Confidentiality
  - 2. Message Origin Non-Repudiatable Verifiability
- Two-step Conventional approach:
  - Sign: Message originator produces digital signature on message
  - Encrypt: Originator encrypts signed message

3

Signcryption

- Introduced by Y. Zheng in 1996
- Achieves both confidentiality and authentication goals more economically than separate 'sign-then-encrypt'
- Savings in Computation and Communication
- Based on Discrete logarithm problem DLP(p,q) in the subgroup of order q in the multiplicative group Zp* (p and q prime).

4

Related Past Schemes

- Schnorr Signature
  - Common Parameters
    - (g,p,q) - where p,q are prime, g has order q in Z_p^*.
  - User Key Generation
    - Secret key $s_A \in_R Z_q^*$, Public key $v_A = g^{-s_A} \bmod p$
  - Signature Generation on message m
    - Pick random $r \in Z_q$
    - Output (e,y) with $e = H(m, g^r \bmod p)$ and $y = r + es_A \bmod q$
  - Signature Verification for (m',e',y')
    - Accept m' as signed by user A iff $e' = H(m', g^{y'} v_A^{e'} \bmod p)$

5

Related Past Schemes

- Girault (1991) Variant of Schnorr:
  - Composite RSA Modulus N=pq
  - Base g of maximal order lcm(p-1,q-1)
  - Forgery provably as hard as factoring N
  - Proof of security (Poupard and Stern, 1998) valid only for inefficient variants with secret key space size S >> $\sqrt{N}$
- Pointcheval (2000) Variant of Girault:
  - Base g with order < S/2
  - Forgery provably as hard as factoring N even for efficient version with S << $\sqrt{N}$

6

New Signcryption Scheme

- Setup by a Trusted Authority (TA)
  - Generate and publish 3 parameters:

(1) Large RSA modulus N = pq (p,q primes)

(2) Large secret key bound S
(can be much less than $\sqrt{N}$ for high efficiency).

(3) *Asymmetric basis* g in Z_N*:
- $Ord_p(g)$ and $Ord_q(g)$ have unequal 2-multiplicities
- $Ord_N(g)$ is large but less than S/2

7

New Signcryption Scheme

- Setup by TA (cont)
  - Publish algorithms for:

(1) one-way hash function $H_1(.)$ with large output length $|H_1|$ bits
(or alternatively any one-to-one mapping over domain <g>).

(2) Keyed collision-resistant one-way hash function KH(.,.) of large output length |KH| bits.

(3) Secure Symmetric Encryption/Decryption algorithms (E, D) with D(key,E(key,m)) = m for all messages m.

8

New Signcryption Scheme

- Key-Pair Generation by User Alice:

(1) Pick random secret integer s_A in the interval {0,...,S-1}.

(2) Compute and publish public key

$$v_A = g^{-s_A} \bmod N$$

9

New Signcryption Scheme

- Signcryption Algorithm by Alice the sender:
  - Alice=(s_A, v_A) to Bob=(s_B, v_B)
  - on Message m

(1) Pick random r in {0,...,R-1} with $R/(2^{|KH|}S) >> 1$

(2) Get Bob's PK, compute $x = v_B^r \bmod N$, and split x into two large 'pieces' $(x_1, x_2) = H_1(x)$

(3) Use x_1 to encrypt m into c = E(x_1 ,m)

(4) Use x_2 and own secret s_A to compute the pair

e = KH(x_2 ,m,bind)

y = r + e . s_A

(5) **Output** 'signcryptext' triple (c,e,y) to be sent to Bob.

10

New Signcryption Scheme

- UnSigncryption Algorithm by Bob the receiver:
  - Bob receives (c',e',y') from "Alice"

(1) Get Alice's PK and use own secret s_B to compute

, $x' = (g^{y'} v_A^{e'})^{-s_B} \bmod N$

and split x' into two large 'pieces' $(x'_1, x'_2) = H_1(x')$

(2) Use x'_1 to decrypt c' into m' = D(x_1',c')

(3) Use x'_2 and decrypted message m' to test:

Is it true that e' = KH(x'_2,m',bind)?

If true, set b = 1 (accept authentic m' from Alice to Bob)

Else, set b= 0 and m' = *

(reject m' as erroneous/modified message).

(4) **Output** (m',b) - recovered message and verify bit.

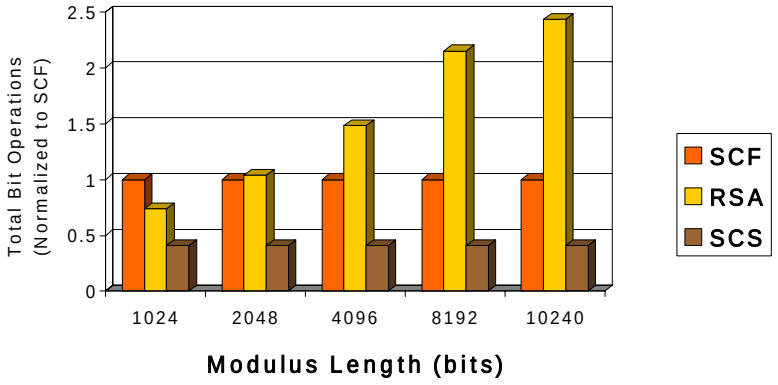
11

Efficiency of New Scheme

- Computation
  - By Sender (Signcryption)
    - On-Line: Very efficient - 1 Hash, 1 Encryption and 1 integer add & multiply (no modular reduction)
    - Off-Line: 1 modular exponentiation
  - By Receiver (UnSigncryption)
    - Product of 2 exponentials with common modulus
      (only ~17% more expensive than 1 modular exponential)
    - 1 Hash
    - 1 Decryption

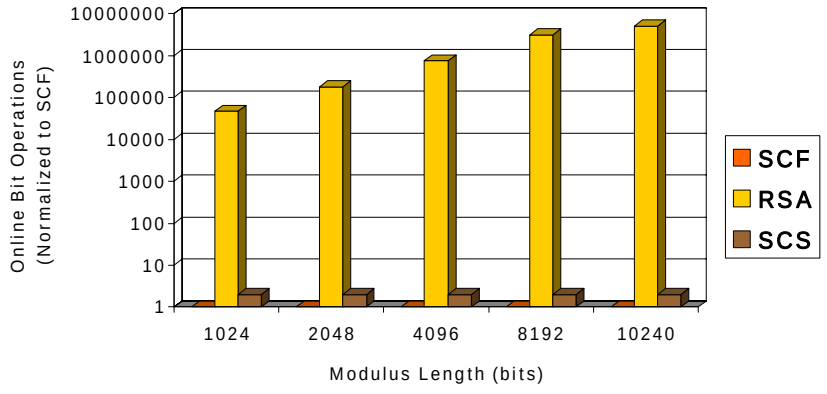
12

Total Computation: Comparison



13

Online Computation: Comparison



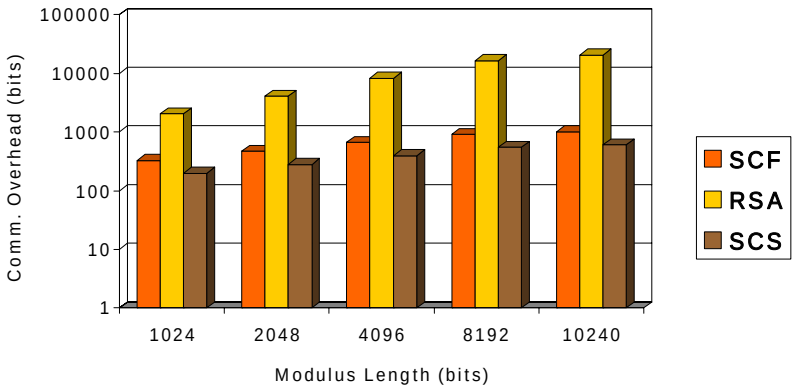
14

Efficiency of New Scheme

- Communication Overhead
  - Overhead larger than original Signcryption Scheme due to Lack of modular reduction in computation of $y = r + e \cdot s$
    - Unreduced product/sum is longer
    - To minimize leakage of the secret, the random commited integer r must be long compared with product $e \cdot s$
  - But still much more efficient than the 'sign-then-encrypt' approach.

15

Communication Overhead: Comparison



16

Setup by Trusted Authority

- Need to generate common parameters (g,N,S) to set up a users 'community'
  - N = pq is an RSA modulus
  - g has order << $\sqrt{N}$ and must be asymmetric basis in Z_N^*
- Problem: Knowledge of (p,q) and factors of (p-1) and (q-1) is needed for generation - Need for user trust in generator!
- TA not needed after (g,N,S) have been generated
  - TA need not manipulate user secret keys (users self-generation)
  - Much weaker constraint than factorization-based ID schemes (eg Fiat-Shamir).

17

Setup by Trusted Authority

- Possible Forms of Trusted Authority:
  - Sealed Black-Box Algorithm
    - Input: Secret key length |S| and Modulus size |N|
    - Output: (g,N) of desired properties
      - Generates primes p,q with (p-1) and (q-1) having known large prime divisors r_p and r_q
      - Uses Chinese Remainder Theorem (CRT) to compute random asymmetric basis g of order $2r_pr_q$
      - Destroys all traces in memory of (r_p, r_q, p, q)
  - Group of Users in private distributed computation - Problem of g generation.

18

Security Analysis

- Theorem: The proposed scheme is
  - existentially unforgeable,
  - under adaptive chosen message attacks,
  - in the random-oracle model,
  - subject to the assumption that:
    - factoring N, given (g,N,S), is 'hard'.
- Stated factoring problem does not appear to be easier than standard factorization problem.

19

Security Analysis

- Main Ideas of Proof:
  - Lemma 1: A non-zero multiple of Ord(g) in Z_N^* reveals factorization of N.
  - Using 'forking technique' (Pointcheval & Stern), an attacker can be used to efficiently extract a multiple L of Ord(g).
    - L is zero iff h(attacker view) = sender's secret key s_A
    - But what if L = 0? - It cannot be used to factor N!
  - Lemma 2 (WI): If $R/(2^{|KH|}S) >> 1$, it is impossible to distinguish which secret key mapping to the public one is used by sender.
  - So if s_A is chosen uniformly, then since attacker has almost no information on the choice, L is non-zero with high probability!

20

Conclusions

- New Signcryption Scheme
  - Very low on-line computation
  - Slightly worse off-line/overhead efficiency than original signcryption (but much better than 'sign-then-encrypt')
  - Forgeability provably as hard as factoring N (in random-oracle model) knowing an asymmetric basis of order much less than $\sqrt{N}$

21

Conclusions

- Open problems:
  - Proof of confidentiality with respect to:
    - Diffie-Hellman Problem in <g>

    or better, with respect to
    - Factorization
  - Private Distributed key generation protocol
    - Generate (g,N,S) of specified properties
    - No participant minority should learn factors of N
    - Reasonably Efficient

22

Conclusions

- Open problems (cont):
  - Efficient signcryption scheme which has at least one of:
    - Provably as hard to break as *standard* factorization problem
    - Each user has a personal modulus

23