Security Level of Cryptography: HIME-2

1 Cryptographic Primitive

Name: HIME-2 (High Performance Modular Squaring Based Public Key Encryption)

Category: Asymmetric Cryptographic Schemes

Security Function: Confidentiality

2 Evaluation

2.1 The Underlying Number Theoretic Problem

The scheme is based on the factorization of a composite number which is a product of d=4,5,6 primes. The suggested size of the composite n is 1024 bits. The choice of the number of primes implies that their size is 256 bits, for d=4 and 171 bits for d=6.

While I agree with the self assessment about the current state of the art of factoring (the elliptic curve method in particular since it applies here), I do not agree with the suggested size. The size of the prime factors is too small and does not leave room for possible improvement of the current state of the art of factoring, or increase in computing power. Such possible extrapolation of the power of computing and the necessary growth of length of keys is outlined by Lenstra and Verhul in the proceedings of PKC 2000.

The small size of the primes also forces the key generation to consider the p-1 and the p+1 methods, which may make key generation slower.

If one is to use Rabin based system on a national level, the performance justification to go to more than 2 prime factors is questionable. By the way, it increases the size of the private key to enable fast decryption. The main reason for Rabin over RSA is the fast encryption which is modular squaring. The choice of small primes in key generation is not a problem and should not imply the change.

2.2 Semantic security evaluation

Remark 2.2.4 when the user may submit plaintext information on the clear violates "semantic security" and since the potential applications are unknown may be problematic.

If semantic security, uses other methods to break ties in the square root procedure, and no extra information is sent, like random oracle preprocessing, then semantic security is achievable.

2.3 Complexity Theory and Security against active attacks

The OAEP method was shown to have some problem (when considering adaptive chosen ciphertext attack (CCA2), e.g. Shoup's recent work, and new proof for RSA which is not as good as far as security reduction between the security of RSA as a one-way function and the immunity to CCA2). It is recommended to replace it with the best "random oracle preprocessing" method available. Namely the one where the reduction in security is smaller. A complete "complexity theoretic" proof should follow, justifying the concrete security bounds, which will imply the period for key replacement.

GENERAL REMARK QQ: It is recommended that a general methodology of "random oracle preprocessing" will be developed for as many as possible algorithms, based on a scheme that applies to any partial-trapdoor function. The best one is the one that applies generally and has smallest reduction in security. Okamoto-Pointcheval scheme may be the best choice.

2.4 Other issuea and problems

Key Generation: The algorithm does not distinguish system wide parameters from choice of the key generation algorithm. It is suggested that the size of key, the number of primes, and the split into two sides of plaintext k0 and k-k0 be defined for the system and it is not up to the key generator. Of course I assume this is done but description is missing and k0 is assumed part of the public key, so may change between users. Also, the choice of k1 is in the user's hand which may be problematic again. (The implementation description fixes sizes, though).

3.2.6 the choice of G2 and H2– it is suggested to follow general remark QQ above, and not to follow the ad hoc method suggested here, which does not look as a good pseudorandom function for certain hash functions (dependency on the constants C1,C2 etc. is after the hashing of the unknown portion x, so certain correlations are possible).

2.5 General remarks

The fixes suggested above may make the system best possible under random oracle assumption ,and a break can then be reduced to factoring. However, why choose such small factors?

Given a cryptosystem it is always good to compare it to relatively similar available systems (in its family). It seems that here RSA and Rabin should be considered:

- CLOSET ALTERNATIVES AND COMPARISON: Using Rabin with two primes is better (if we insist on equivalence of decryption to factoring). It may be a hardware optimization which dictates the small prime Rabin, but is it necessary in general.
- If we do not insist and believe that the RSA assumption is sufficient then RSA with public exponent being 3 is quite competitive in encryption, while reducing much of the complications.

I believe that indeed, plain RSA or Rabin with 2-prime composites and a method of preprocessing (convert) better than OAEP (using more recent techniques like Okamoto-Pointcheval's), and standard pseudorandom and hash functions, and fixed k0,k1,k (partitions to message, randomness, redundancy) will make a better system security wise and one which is more easily deployed.

Rabin has certain complications, special structure of primes, Jacobi symbol calculations etc.

The choice of special and small primes is problematic for national level security.

Given the high possibility of insecurity, the performance advantages and exact hardware and software performance may not be a decisive factor in the evaluation of this cipher. As submitted it is behind the state of the art of what is considered secure system.