A new Digital Signature Algorithm based on Factorization and Discrete Logarithm problem

Sushila Vishnoi[1], Vishal Shrivastava[2]
Department of computer science
Arya College of Engineering and IT, India

Abstract. **Generally, digital signature algorithms are based on a single hard problem like prime factorization problem, discrete logarithm problem, elliptic curve problem. If one finds solution of this single hard problem then these digital signature algorithms will no longer be secured and due to large computational power, this may be possible in future. There are many other algorithms which are based on the hybrid combination of prime factorization and discrete logarithms problem but different weaknesses and attacks have been developed against those algorithms. This paper also presents a new variant of digital signature algorithm which is based on two hard problems, prime factorization and discrete logarithm.**

Keywords: **Digital Signature; Discrete logarithm; Factorization; Cryptanalysis**

I. INTRODUCTION

In modern cryptography [5], the security of digital signature algorithms are based on the difficulty of solving some hard number theoretical problems. These algorithms stay secure as long as the problem, on which the algorithm is based, stays unsolvable. The most used hard problems for designing a signature algorithm are prime factorization (FAC) [27] and Discrete Logarithm (DL) [6] problems. For improving the security, the algorithms may be designed based on multiple hard problems. Undoubtedly, the security of such algorithms is longer than algorithms based on a single problem. This is due to the need of solving both the problems simultaneously. Many digital signature algorithm have been designed based on both FAC and DL [8, 11, 12, 14, 17, 19, 26, 28, 30, 31] but to design such algorithms is not an easy task since many of them have been shown to be insecure [9, 18, 19, 20, 21, 29, 30, 31].

In 1994, He and Kiesler [11] proposed digital signature algorithms based on two hard problems-the prime factorization problem and the discrete logarithm problem. In 1995, Harn [9] showed that one can break the He-Kiesler algorithm if one has the ability to solve the prime factorization. Lee and Hwang [18] showed that if one has the ability to solve the discrete logarithms, one can break the He-Kiesler algorithm. Shimin Wei [31] showed that any attacker can forge the signature of He-Kiesler algorithm without solving any hard problem. In 2002, Z. Shao [28] presents an algorithm based on factoring and discrete logarithms. But later Tzeng[30] showed that Shao digital signature algorithm is not secure and there are many weaknesses. He then proposed a new signature algorithm [30] to overcome the weaknesses inherent in Shaos signature algorithm. In 2005, Shao [29] proved that Tzeng signature algorithm is not secure as if attackers can solve discrete logarithm problems, they can easily forge the signature for any message by using a probabilistic algorithm proposed by Pollard and Schnorr [24] and if attacker can factor the composite number, he can recover the private keys of legal signers. Therefore the security of Tzeng digital signature algorithm depends only one of the problem, prime factorization or discrete logarithm.

A signature scheme cannot be unconditionally secure, since Adv can test all possible signature for a given message m. So, given sufficient time, Adv can always forge Sender's signature on any message. Thus, our goal is to find signature schemes that are computationally or provable secure. In this paper, a new variant of digital signature algorithm (DSA) is proposed which is based on the combined difficulties of integer factorization problem and discrete logarithm problem. Rest of the paper is organized as follows. Section 2 describes security threats against DL and FAC problem based algorithms. The proposed algorithm is described in section 3. In section 4, security analysis is carried out for the proposed algorithm. Performance analysis of the proposed algorithm is discussed in section 5. Finally, in section 6, paper is concluded.

II. SECURITY THREATS AGAINST DL AND FAC PROBLEM BASED ALGORITHMS

The ElGamal signature algorithm [6] is a digital signature algorithm which is based on the difficulty of computing discrete logarithms. The main threat against the ElGamal algorithm is that the strength of the algorithm solely depends on the discrete logarithm problem. If the discrete logarithm problem can be solved then it is possible to obtain the secret x from the public value g^x, and then one could sign messages as a genuine sender. In 1993 Daniel M. Gordon presented an algorithm [7] that could solve discrete logarithms for small numbers in a finite field of prime order p, $GF(p)$, using the Number Field Sieve. Takuya

Hayashi [10] presented an algorithm that can solve a 676-bit Discrete Logarithm Problem in $GF(3^{6n})$ for n is any positive integer. It is clear from the work of Gordan and Hayashi that, in near future, it could be feasible to solve the discrete logarithms problem for large numbers in a polynomial time. RSA Digital Signature algorithm (RSADSA) [27] proposed by Rivest, Shamir and Adleman, is a popular and well known digital signature algorithm. RSADSA is an asymmetric digital signature algorithm as it uses a pair of keys, one of which is used to sign the data in such a way that it can only be verified with the other key. Security of RSADSA algorithm is based on difficulty of solving the prime factorization problem. Many efforts have been made in past to solve the prime factorization problem [13, 23, 22, 25]. In 2002, Weger [4] described a new attack for solving prime factorization problem as if there is small difference between the prime factors of modulus then a polynomial time cryptanalysis for factoring modulus is possible. In 2003, Boneh and Brumley [1] demonstrated a more practical attack capable of recovering RSA factorizations over a network connection. This attack takes advantage of information leaked by the Chinese remainder theorem optimization used by many RSA implementations. RSADSA is not only vulnerable to the prime factorization attacks but also to the pri- vate key d. Paul Kocher [16] described that if an Adversary Eve knows Alice's hardware in sufficient detail and is able to measure the decryption times for several known cipher texts, she can deduce the decryption key d quickly. Next, there are many threats if the RSA private exponent is chosen small. The first significant attack on small private exponent RSA was Wieners continued fraction attack [32]. Given only the public key (e, n), the attack factors the modulus using information obtained from one of the convergent in the continued fraction expansion of e/n. It was shown by Coppersmith [13], that an RSA modulus with balanced primes could be factored given only $1/2$ of the most significant bits of one of the primes. It was later shown by Boneh, Durfee and Frankel [2] that $1/2$ of the least significant bits of one of the primes was also sufficient. A theoretical hardware device named TWIRL designed by Shamir and Tromer in 2003 [15], questioned the security of 1024 bit keys. Nowadays due to the availability of high end resources of computation the chances of the various types of attacks have increased. It is quite possible that an organization with sufficiently deep pockets can build a large scale version of his circuits and effectively crack an RSA 1024 bit message in a relatively short period of time. The RSADSA algorithm is also forgeable for chosen-message attack, since RSA is multiplicative, the signature of a product is the product of the signatures.

III. The Proposed signature algorithm

This section proposes a new variant of digital signature algorithm based on the two NP-Complete problems named prime factorization and discrete logarithm. The algorithm is as follows:

A. *Key Generation*

- Choose a large prime p such that computing discrete logarithms modulo p is difficult and two large prime numbers p_1 and q_1 such that $p < n$ where $n = p_1 \times q_1$.
- Calculate $\varphi(n) = (p_1 - 1) \times (q_1 - 1)$
- Choose random numbers k and v such that $1 < k, v < p - 1$.
- Choose random numbers x, r and b such that $1 < x, r, b < n - 1$. x should be relative prime to $\varphi(n)$ (i.e. $gcd(x, \varphi(n)) = 1$)
- Choose a primitive root g in $Z^*{}_n$
- Calculate c such that

$$b^x \times c\ (mod) n = 1$$

- Calculate u, w, t and y as follows:

$$u = g^x \ mod\ p,$$
$$w = g^v \ mod\ p,$$
$$t = u^x \ mod\ p,$$
$$y = r^x \ mod\ n.$$

- Public key is (x, c, g) and private key is (k, v, u, w, b, r).

B. *Signature Generation*

Step-1:

Choose an integer z such that $1 < z < (p - 1)$ and it is relative prime to $(p - 1)$ i.e. $(gcd\ (z, p - 1) = 1)$. z should be different for every message m and is not public. Here $H(.)$ is a one way hash function.

Step-2**:** Calculate

$$h = g^z \ mod\ p,$$
$$\gamma = t \times w^h \ mod\ p,$$
$$f = \left(r \times b^{H(m)}\right) mod\ n,$$
$$s = \left(\left((H(m) - kw - hv + yz) \times z^{-1}\right)\right) mod\ (p - 1)$$

If $t = 0$ and/or $f = 0$ and/or $s = 0$ then repeat step 1 and 2 else tuple (γ, h, f, s) is the signature of m.

Here $-kw, -hv$ are additive inverse of kw and hv respectively and z^{-1} is the multiplicative inverse of z with respect to $mod(p - 1)$.

C. *Signature Verification*

- Calculates $H(m)$ using the received message m at receiver's end.
- If $g^{H(m)} \times h^{(f^x \times c^{H(m)}\ mod\ n)} \equiv \gamma \times h^s mod\ p$ then the signature is valid else reject the signature.

D. *Proof of correctness*

R.H.S.

$$\begin{aligned}
&= \gamma \times h^{s} \, mod \, p \\
&= \gamma \times h^{\left((H(m)-kw-hv+yz)\times z^{-1} \, mod(p-1)\right)} mod \, p \\
&= \gamma \times g^{\left((H(m)-kw-hv+yz) \; mod(p-1)\right)} mod \, p \\
&= t \times w^{h} \times g^{(H(m)-kw-hv+yz) \, mod(p-1))} mod \, p \\
&= u^{w} \times w^{h} \times g^{(H(m)-kw-hv+yz) \, mod(p-1))} mod \, p \\
&= g^{H(m)} \times h^{y \, mod(p-1)} \; mod \, p \\
&= g^{H(m)} \times h^{y} \, mod \, p
\end{aligned}$$

And from L.H.S.

$$\begin{aligned}
& f^{x} \times c^{H(m)} mod \, n \\
&= \left(r \times b^{H(m)}\right)^{x} mod \, n \times c^{H(m)} \, mod \, n \\
&= r^{x} \times b^{H(m)\times x} \times c^{H(m)} \, mod \, n \\
&= r^{x} \times (b^{x} \times c)^{H(m)} \, mod \, n \\
&= r^{x} \; mod \, n \\
&= y
\end{aligned}$$

Therefore, L.H.S. = $g^{H(m)} \times h^{y} \, mod \, p$ is equal to R.H.S.

IV. Security Analysis

In this section, security analysis of the proposed algorithm is carried out. We shall show that the security of proposed algorithm is based on solving both the problem; prime factorization and discrete logarithm, simultaneously. We say that an Oracle O breaks the proposed signature scheme, if given the public key of the scheme and a message m_{adv}.

Theorem 1: If there is an ORACLE that can solve the prime factorization and Discrete logarithm problem, then it can also break the proposed algorithm.

Proof.: Lets the oracle O gives values of prime factor (p_1, q_1) of n and $(k, v, z, w)$from solving DL and FAC using (t, h). We know that $n = p_1 \times q_1$, and $\varphi(n)$ is the Euler's totient function. Consider the equation

$$b^{x} \times c = mod \, n \tag{1}$$

where b and $x \in z_n$. Now from Diophantine equation for x and $\varphi(n)$; ∃ u and v such that $xu - \varphi(n)v = f$, where $f \in Z_n$. Now as in the proposed algorithm $gcd\big(x, \varphi(n)\big) = 1$, so it is easy to solve equation (1) and the computation $b \equiv \left(\frac{1}{c}\right)^{u} mod \, n$ gives the required value of b, since

$$\begin{aligned}
b &= \left(\frac{1}{c}\right)^{u} mod \, n \\
&= \left(\frac{1}{c}\right)^{\frac{1+v\varphi(n)}{x}} mod \, n, \\
&= \left(\frac{1}{c}\right)^{\frac{1}{x}} mod \, n.
\end{aligned}$$

Further consider the equation

$$y = r^{x} \, mod \, n \tag{2}$$

where $r, x \in Z_n$. Now from Diophantine equation $xu - \varphi(n)v = f$, one can easily calculate the value of u and the computation $r \equiv y^{u}(mod)\, n$ gives the required value of r, since

$$\begin{aligned}
r &= y^{u} \, mod \; n \\
&= (y)^{\frac{1+v\varphi(n)}{x}} \; mod \, n, \\
&= y^{\frac{1}{x}} \; mod \, n,
\end{aligned}$$

Hence by factoring n, one can easily calculate $\varphi(n)$ and by solving Diophantine equation $xu - \varphi(n)v = f$, he can get the value of u and subsequently value of b and r.

Further, we know the value of z, v, k, and v, hence the signature $(t, h, f, s)$of a message m_{adv},can be generated as follows:

$$\begin{aligned}
u &= g^{k} \, mod \, p, \\
w &= g^{v} \, mod \, p, \\
t &= u^{w} \, mod \, p, \\
h &= g^{z} \, mod \, p, \\
t &= t \times w^{h} \, mod \, p, \\
y &= r^{x} \, mod \, n, \\
f &= \left(r \times b^{H(m_{adv})}\right) mod \, n, \\
s &= \left(\left((H(m_{adv}) - kw - hv + yz\right) \times z^{-1})\right) mod(p-1)
\end{aligned}$$

Therefore, the tuple (t, h, f, s) is a valid signature of message m_{adv} using the proposed algorithm. There are some possible areas where an adversary (Adv) may try to attack on this new developed signature algorithm. Following are the possible attacks (not exhaustive) and the reasons why that would fail:

A. *Key-Only Attack*:

Adv wishes to obtain private key $(r, b, p_1, q_1, z, k, v)$ using all information that is available from the system. In this case, Adv needs to solve the prime factorization problem to find r and b from modulus $x = p_1 \times q_1$. Also he has to solve discrete logarithm problem to find z, k and v using t, h and g For finding b, Adv has to solve $b = c^{-1/x} \, mod \, n$ which is NP-Complete for large b because Adv has to find prime factorization of modulus n to calculate x^{th}root of c^{-1}. Again for finding r using y and x also a x^{th} root problem and this problem can be solved only when the factorization of modulus n is known. Therefore an Adv has to solve DL problem and FAC problem for finding the private key. This makes the proposed algorithm secure enough for this type of attacks.

B. *Chosen- message Attack:*

In this attack, Adv requires a sign on some messages of his choice by the authorized signatory. With the help of chosen-messages and corresponding signatures, Adv generates another message and can forge sender's signature on it. The RSADSA algorithm is forgeable for this attack. For attack on RSADSA, suppose, Adv asks signer to sign two legitimate messages m_1

and m_2 for him. Let us assume s_1 and s_2 are signatures of m_1 and m_2 respectively. Adv later creates a new message $m = m_1 \times m_2$ with signature $s = s_1 \times s_2$ Adv can then claim that signer has signed m. The chosen-message attack for the proposed algorithm is a matter of further research as there is no obvious method which shows that the proposed algorithm is vulnerable to this attack.

C. Known Partial Key Attack:

Let us assume that Adv is able to solve DL but not FAC problem. Suppose he also knows u and w then he will find z, k, v using DL. But for finding the value of b and r, Adv has to solve FAC problem. However, Adv can easily calculate the value of y, using $f^x \times c^{H(m)}$. But it is a function of original message digest and therefore the Adv's signature cannot match with the sender's signature. So using this y, he cannot forge the sender's signature. Now if Adv can solve FAC problem but not DL problem then he will not be able to find the value of z, k, v.

D. Known partial key and Message Attack:

Let us assume that Adv is able to solve FAC problem hence, he knows the secret key component b and r. Adv may also have i valid signatures $\left(t_j, h_j, m_j, f_j, s_j,\right)$ on message m_j where $j = 1, 2, \ldots \ldots \ldots \ldots i$ and public key (c, x, g) and he attempts to find secret keys $\left(k, v, u, w, r, z_j\right)$. Since y_i's, $j = 1, 2, \ldots \ldots \ldots \ldots i$, can be calculated by the Adv using $f^x \times c^{H(m_j)}$ so y can be treated as a known entity to Adv. Now, Adv has i equations as follows representing ${z_j}^{-1}$ as l_i

$$s_1 = \left((H(m_1)l_1 - kwl_1 - hvl_1 + y_1)\right)$$
$$s_2 = \left((H(m_2)l_2 - kwl_2 - hvl_2 + y_2)\right)$$
$$.$$
$$.$$
$$.$$
$$s_i = \left((H(m_i)l_i - kwl_i - hvl_i + y_i)\right)$$

In the above i equations, there are $(i + 3)$ variables namely k, w, v and l_j where $j = 1,2, \ldots \ldots \ldots \ldots, i$ which are not known by the Adv. Hence k, w, v and l_j stay hard to detect because for Adv, there are $i + 3$, unknowns to be found from i equations. In case, Adv is able to detect the variables k, l_j and v and tries to sign his message (say) m_{adv} using sender's signature. Hence, the Adv cannot sign its own message using sender's signature even if he knows the part of the secret key.

E. Blinding:

In this attack, in case of RSADSA suppose Adv wants sender's signature on his message m. For this Adv try the following: he picks a random $r \in {Z^*}_n$ and calculates $m' = r^e \times m \, mod \, n$. He then asks sender to sign the message m'. Sender may provide his signature s' on the message m'. But we know that $s' = (m')^d \, mod \, n$. Adv now computes $s = s' / r \;\; mod \, n$ and obtains sender's signature s on the original m. This technique, called blinding, enables Adv to obtain a valid signature on a message of his choice by asking Sender to sign a random blinded message. Sender has no information as to what message he is actually signing. So, RSA is vulnerable to this attack. Again an intensive research is required to check whether the proposed algorithm is vulnerable to Blinding or not. Currently, best of authors efforts it seems not vulnerable for Blinding.

V. Performance Analysis

Using the criterion presented in [3], the complexity of each method is estimated as a function of number of bit operations required. The basic exponential operation here is $a^b \, mod \, n$ and time complexity of this operation is $O(\log b \times M(n))$, where $M(n)$ is the complexity of multiplying two n bit integers. In the proposed algorithm signature generation requires 3 modular exponentiation and signature verification requires 5 modular exponentiation which leads to the complexity of the algorithm to be $O(3 \times log^3 n)$ and $O(5 \times log^3 n)$ for signature generation and verification respectively as here $b = O(n)$ and time complexity of multiplying two n bit integers is $O\left(log^2 n\right)$. If the complexity of proposed DSA is compared with other DSA algorithms of same category (i.e. DSA algorithms that are based on multiple hard problems) then we see that the Dimitrios Poulakis signature algorithm [26] requires 6 modular exponentiation in signature generation and 2 modular exponentiation in signature verification. Ismail E. S signature algorithm [14] requires 5 modular exponentiation in signature generation and 5 modular exponentiation in signature verification. Shimin Wei signature algorithm [31] requires 5 modular exponentiation in signature generation and 8 modular exponentiation in signature verification. So it is clear that the complexity of the proposed algorithm is competitive equivalent to most of the digital signature algorithms which are based on prime factorization and discrete logarithm.

VI. Conclusions

In this paper, a new variant of digital signature algorithm is proposed which is based on the two hard problems called prime factorization and discrete logarithm. It is shown that one have to solve both the problems simultaneously for cryptanalysis of this algorithm. The performance of the proposed algorithm is found to be competitive to the most of the digital signature algorithms which are based on multiple hard problems.

References

[1]. D. Boneh and D. Brumley. Remote timing attacks are practical. Proceedings of 12th USENIX Security Symposium, 2003.

[2]. D. Boneh, G. Durfee, and Y. Frankel. Exposing an RSA private key given a small fraction of its bits. Full version of the work from Asiacrypt, 98, 1998.

[3]. D. Boneh and H. Shacham. Fast variants of RSA. CryptoBytes (RSA Laboratories), 5:1-9, 2002.

[4]. B. De Weger. Cryptanalysis of RSA with small prime difference. Applicable Algebra in Engineering, Communication and Computing, 13(1):17-28, 2002.

[5]. W. Di_e and M. Hellman. New directions in cryptography. Information Theory, IEEE Transactions on, 22(6):644-654, 2002.

[6]. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. Information Theory, IEEE Transactions on, 31(4):469- 472, 2002.

[7]. D. M. Gordon. Discrete Logarithms in GF(P) Using the Number Field Sieve. SIAM Journal on Discrete Mathematics, 6(1):124-138, 1993.

[8]. L. Harn. Public-key cryptosystem design based on factoring and discrete logarithms. In IEE Proc.-Compul. Digit. Tech, volume 141, pages 193-195. IET, 1994.

[9]. L. Harn. Comment: Enhancing the security of El Gamal's signature scheme. IEE Proceedings-Computers and Digital Techniques, 142:376, 1995.

[10]. T. Hayashi, N. Shinohara, L. Wang, S. Matsuo, M. Shirase, and T. Takagi. Solving a 676-Bit Discrete Logarithm Problem in GF (3 6n). Public Key Cryptography-PKC 2010, pages 351-367, 2010.

[11]. J. He and T. Kiesler. Enhancing the security of El Gamal's signature scheme. In Computers and Digital Techniques, IEE Proceedings-, volume 141, pages 249-252. IET, 1994.

[12]. W. H. He. Digital signature scheme based on factoring and discrete logarithms. Electronics Letters, 37(4):220-222, 2002.

[13]. M.J. Hinek. Cryptanalysis of RSA and its variants. Chapman & Hall/CRC, 2009.

[14]. ES Ismail, NMF Tahat, and RR Ahmad. A New Digital Signature Scheme Based on Factoring and Discrete Logarithms. Journal of Mathematics and Statistics, 4(4):222-225, 2008.

[15]. B.Kaliski. TWIRL and RSA Key Size. http://www.rsa.com/rsalabs/node.asp?id=2004, 2003, Accessed on Nov. 2010.

[16]. P. Kocher. Timing attacks on implementations of Di_e-Hellman, RSA, DSS, and other systems. In Advances in CryptologyCRYPTO96, pages 104-113. Springer, 1996.

[17]. C.S. Laih and W.C. Kuo. New signature schemes based on factoring and discrete logarithms. IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, 80(1):46-53, 1997.

[18]. NY Lee. Security of Shao's signature schemes based on factoring and discrete logarithms. In Computers and Digital Techniques, IEE Proceedings-,volume 146, pages 119-121. IET, 2002.

[19].] N.Y. Lee and T. Hwang. Modi_ed Harn signature scheme based on factorizing and discrete logarithms. In Computers and Digital Techniques, IEE Proceedings-, volume 143, pages 196-198. IET, 2002.

[20]. N.Y. Lee and T. Hwang. The security of He and Kiesler's signature schemes. In Computers and Digital Techniques, IEE Proceedings-, volume 142, pages 370-372. IET, 2002.

[21]. J. Li and G. Xiao. Remarks on new signature scheme based on two hard problems. Electronics Letters, 34(25):2401, 2002.

[22]. P.L. Montgomery. A survey of modern integer factorization algorithms. CWI quarterly, 7(4):337-365, 1994.

[23]. M.A. Morrison and J. Brillhart. A Method of Factoring and the Factorization of F 7. Mathematics of Computation, 29(129):183-205, 1975.

[24]. J.M. Pollard and C.P. Schnorr. An efficient solution of the congruence x2+ ky2= m (mod n). IEEE Transactions on Information Theory, 33(5):702- 709, 1987.

[25]. C. Pomerance. A tale of two sieves. Biscuits of Number Theory, page 85, 2008.

[26]. D. Poulakis. A variant of Digital Signature Algorithm. Designs, Codes and Cryptography, 51(1):99-104, 2009.

[27]. R.L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2):120-126, 1978.

[28]. Z. Shao. Signature schemes based on factoring and discrete logarithms. In Computers and Digital Techniques, IEE Proceedings-, volume 145, pages 33-36. IET, 2002.

[29]. Z. Shao. Security of a new digital signature scheme based on factoring and discrete logarithms. International Journal of Computer Mathematics, 82(10):1215-1219, 2005.

[30]. S.F. Tzeng, C.Y. Yang, and M.S. Hwang. A new digital signature scheme based on factoring and discrete logarithms. International Journal of Computer Mathematics, 81(1):9-14, 2004.

[31]. S. Wei. A New Digital Signature Scheme Based on Factoring and Discrete Logarithms. Progress on Cryptography, pages 107-111, 2004.

[32]. S. Wei. A New Digital Signature Scheme Based on Factoring and Discrete Logarithms. Progress on Cryptography, pages 107-111, 2004.