Vol.2011-AL-134 No.14
2011/3/7

On an Insider Resistant Authentification Protocol and Its Security

Kimikazu Kato ,[†1] Naonobu Yatsukawa [†1]
and Tetsuya Shiroishi[†2]

Yatsukawa introduced the Insider-Resistant One-Time Password (IROTP) , which is an authentication protocol intended to be secure for attacks by insiders. In the IROTP, the RSA decryption function factors recursively on a seed, and its security depends on the period of the sequence generated by recursive factoring of the RSA decryption function.

We have found a condition which gives a sufficient long period for the RSA decryption function. In other words, we have proved that the IROTP, with certain parameters, is secure for attacks utilizing the period. We have also found an algorithm to find such parameters.

1. Introduction

The one time password (OTP) system is widely used for an authentication. The merit of the OTP compared to the static password is lower administration cost. Actually in the static password system, to keep it safe, an administrator has to 1) confirm the password used is strong enough, 2) make a user to change the password after a certain interval, and 3) care for the case that a user forgets a password. The OTP solves those problems.

Although there are such merits in the OTP, one of its week points is vulnerability for insiders. A password of the OTP is usually determined by either 1) a function value $f(t)$ of a certain shared number t (which is counter or time), or 2) recursive factoring of a one-way function f to a seed a, i.e. with $a_0 = a$, $a_i = f(a_{i-1})$ is used as a password[4]. In both cases, if an intruder sees the secret information stored in the server, say f and a_i, he/she can easily pretend to be an authenticated user.

To overcome that problem, Yatsukawa[10),11)] invented a novel protocol, which we call the Insider-Resistant One Time Password (IROTP) system. In the IROTP, just knowing the authentication information is useless for an eavesdropper. Actually, in the protocol, the client keeps a secret key of the RSA cryptography[9)] and the server authenticates the client with the corresponding public key.

In this paper, we present an algorithm to find good parameters so that the IROTP is secure enough. In the IROTP, in order to generate a sequence of passwords, the decrypt function of the RSA cryptography is repeatedly applied to a seed. If the period of the sequence is short, an eavesdropper can easily tell the password coming next and succeed in the attack. Thus we need to make the period long enough. We present a condition for a private key of RSA encryption function, i.e. a pair of prime numbers, to achieve a sufficiently long period. Furthermore, we give an algorithm to obtain a pair of prime numbers which satisfies the condition.

Our algorithm, of course, also uses the primality check for a given integer. The most practically effective primality check algorithm is by Miller[6)] and Rabin[8)]. Also refer to the textbook by Motwani and Raghavan[7)] for a comprehensive explanation.

The rest of this paper is organized as follows. In Sect. 2, we explain some basics of RSA, fundamental mathematical facts, and a specification of the IROTP. In Sect. 3, we show the main results: the main theorem and the algorithm to find a parameter to achieve a long period. Then we conclude in Sect. 4.

2. Preliminaries

2.1 RSA cryptography

We regard it is needless to explain the detailed protocol of the RSA cryptography, but just for some notations, we define the encryption and decryption function of the RSA as follows.

Definition 1. (RSA encryption and decryption function Let p and q be prime numbers, $n = pq$, and e be an integer coprime to $p-1$ and $q-1$. The RSA encryption function $f_{n,e}$ is defined as follows:

$$f_{n,e}(a) := a^e \mod n \tag{1}$$

†1 Nihon Unisys, Ltd.
†2 UNIADEX, Ltd.

For an integer d which satisfies $ed \equiv 1 \pmod{(p-1)(q-1)}$, the RSA decryption function $g_{p,q,d}$ is defined as

$$g_{p,q,d}(a) := a^d \mod pq \tag{2}$$

The tuple of the parameters (n, e) is called a *public key*, while the tuple (p, q, d) is called a *private key*.

It is known that if either p or q has a small prime factor, the encryption function is not secure enough (see the textbook by Koblitz[5] for example). In the actual implementation of RSA encryption function in OpenSSL[1], p and q are chosen so that they do not have a prime factor not bigger than 17863.

2.2 IROTP and related problem

2.2.1 IROTP

The Insider Resistant One-Time Password (IROTP) system is proposed by Yatsukawa[10),11)], and is intended to overcome the week point of existing one-time password (OTP) system. In the classical OTP system, a password or a password-generating function is shared between the server and the client. The main problem is that if an eavesdropper gets the shared information, he/she can easily impersonate to log in the server.

The idea of the IROTP is to utilize the RSA decryption function to generate a sequence of passwords and to use the encryption function to certify that a password is generated by the authenticated user. Only the decryption function is stored in the server and just obtaining it cannot make an impersonation attack possible. We explain the protocol of the IROTP as follows.

First the client and the server share an integer a_0 which is used as the seed of the sequence. The sequence of passwords is generated by $a_i = g(a_{i-1})$. In the i-th time authentication process, the client send $a_i = g(a_{i-1})$ to the server, and the server certifies $f(a_i) = a_{i-1}$ to authenticate the client.

2.2.2 Period of the IROTP

The problem which arises here is whether the period of the sequence $\{a_i\}$ is long enough. If the period π of $\{a_i\}$ is short, just seeing the segment of the sequence $a_k, \dots a_{k+\pi}$ makes an attack successful.

The period of (g, a) for a function g and an element a in the domain of g is defined as the smallest integer m such that $g^m(a) = a$. We denote the period of (g, a) by $\pi(g, a)$. Our problem can be stated as follows. Find the suitable parameters p, q, d so that the period of the RSA decryption function $g_{p,q,d}$ is long enough.

2.3 Mathematical facts

Here we state some mathematical facts. For the proofs of the following two theorems, refer to the textbook[3)] for example.

Theorem 1 (Fermat)**.** *If p is prime and $a \neq 0$, then*

$$a^{p-1} \equiv 1 \pmod{p} \tag{3}$$

Theorem 2 (Chinese Reminder)**.** *Let $n_1, \dots, n_m$ be integers with $\gcd(n_i, n_j) = 1$ for $i \neq j$. Let n be product $n = n_1 \cdots n_m$. Let $a_1, \dots, a_m$ integers. Consider the following system of equations:*

$$\begin{aligned} x &\equiv a_1 \pmod{n_1} \\ x &\equiv a_2 \pmod{n_2} \\ &\vdots \\ x &\equiv a_m \pmod{n_m}. \end{aligned} \tag{4}$$

Then there exists only one $x (0 \leq x < n)$ which satisfies this system.

For the proof of the following theorem, refer to the Erdös et al.[2)].

Theorem 3. *When an integer n is factored as*

$$n = 2^{t_0} p_1^{t_1} p_2^{t_2} \cdots p_m^{t_m}, \tag{5}$$

define $\lambda(n)$ as

$$\lambda(n) = \begin{cases} \operatorname{lcm}(2^{t_0-1}, p_1^{t_1-1}(p_1-1), \dots, p_m^{t_m-1}(p_m-1)) & (t_0 \leq 2) \\ \operatorname{lcm}(2^{t_0-2}, p_1^{t_1-1}(p_1-1), \dots, p_m^{t_m-1}(p_m-1)) & (t_0 \geq 3) \end{cases}. \tag{6}$$

Then,

$$d^{\lambda(n)} \equiv 1 \pmod{n} \tag{7}$$

for an integer d such that $\gcd(d, n) = 1$. Moreover there exist an integer d such that $\lambda(n)$ is smallest number s which satisfies $d^s \equiv 1 \pmod{n}$

3. Algorithm

The following theorem is essential to show the existence of a parameter set which achieves a sufficiently long period.

Theorem 4. *There exists an integer a such that the period $\pi(g_{p,q,d}, a)$ of the RSA decryption function $g_{p,q,d}$ with respect to a is*

$$\pi(g_{p,q,d}, a) = \lambda\big(\operatorname{lcm}(p-1, q-1)\big) \tag{8}$$

where $\lambda(n)$ is the function defined in Theorem 3.

Proof. The period is the smallest k which satisfies $a^{d^k} \equiv a$. This is equivalent to $a^{d^k-1} \equiv 1$. Because of Theorem 1,

$$a^{p-1} \equiv 1 (\mathrm{mod}\ p), \quad a^{q-1} \equiv 1 (\mathrm{mod}\ q). \tag{9}$$

Let $s = \mathrm{lcm}(p-1, q-1)$, then s is the smallest number that satisfies $a^s \equiv 1 (\mathrm{mod}\ p)$ and $a^s \equiv 1 (\mathrm{mod}\ q)$. Due to Theorem 2, s is the smallest integer that satisfies

$$a^s \equiv 1 \pmod{n}. \tag{10}$$

This means

$$d^k \equiv 1 \pmod{s}. \tag{11}$$

By applying Theorem 3, we complete the proof. □

In Theorem 4, it is only said that for a suitable d, the period becomes $\lambda(n)$. The following theorem includes a condition for d although the period stated is weaker than Theorem 4

Theorem 5. *For a prime number p, suppose that $p-1$ can be divided by a prime number $r \geq 3$. For an integer d, suppose that $d \mod r \in \mathbb{Z}/r\mathbb{Z}$ is a generator of $\left(\mathbb{Z}/r\mathbb{Z}\right)^\times$. Then the period is bigger than or equal to $r-1$.*

Proof. Because $d \mod r$ is a generator, if $l < r-1$, then $d^l \not\equiv 1 \pmod{r}$. Thus, because of 2, $d^l \not\equiv 1 \pmod{\mathrm{lcm}(p-1, q-1)}$. This means $a^{d^l-1} \not\equiv 1 \pmod{pq}$ □

We give a subroutine which computes the factoring of a large number if possible. Even though an arbitrary large number cannot be factored within a practical time, if an integer is expressed as (product of small primes)×(large prime), the explicit prime power factorization can be obtained by the following algorithm. Here p_i means i-th prime number with $p_0 = 2$.

```
function SmallNumberFactor (n,m, b)
    x ← n; F ← ∅
    while x > b and i ≤ m do
        if x is divisible by p_i
        then x ← x/p_i; F = F ∪ {p_i}
        else i ← i + 1
    end while
    if x is prime then
        return F ∪ {x}
    else
        return "not factored"
    end if
end function
```

This function factors n with first m primes. It returns prime power factorization of n if possible and "not factored" otherwise.

Now we consider a method to find suitable p and q which guarantee a long period. Theorem 5 means if p has a large prime factor, the period is proved to be long. We propose the following algorithm to determine only p, not for both p and q. The other prime q can be determined arbitrarily. Here, by $\{p_i\}_{i=0}^m$, we denote the first $m+1$ prime numbers with $p_0 = 2$. Here, l, m, π_0, and π_1 are parameters explained below.

```
n ← sufficiently large odd number
while true do
    if n is prime then
        for i := 1 to l do
            if n − 1 is divided by p_i then goto next
        end for
        Call SmallPrimeFactor(n − 1,m,π_0)
        if factored then
            r ←Largest prime factor of n − 1
            F ← SmallPrimeFactor(r − 1,m,π_1)
            if factored then return n and F
        end if
    end if
    n ← n + 2
end while
```

This function returns a prime number which should be used as p and the factors of (largest prime factor of $n-1$) -1. The factors is used to determine a and d.

Note that l is the parameter to keep the encryption safe, and as is mentioned, the value used in OpenSSL is $p_l = 17863$. The parameter $m(\geq l)$ is to determine

the balance between how many primes are targeted and judging time per number. For a smaller m, the judging time becomes shorter but fewer primes are considered as candidates. The algorithm works even for the case $m = l$, which means the case that $p-1$ is (power of 2)$\times$(prime), but suitably large m makes it easier to find an appropriate parameter set.

The parameter π_0 is to assure a long period. Let the largest prime divisor of $p-1$ be r, then $r > \pi_0$. Due to Theorem 5, the period becomes $\geq \pi_0 - 1$ for some d.

The rest is determine a suitable d. Since $r-1$ is factored, it is easy to find a generator of $\left(\mathbb{Z}/r\mathbb{Z}\right)^{\times}$, and let d be the generator. Actually, to say d is a generator, just check $d^{\frac{r-1}{s}} \not\equiv 1 \pmod{r}$ for all prime divisor s of r.

4. Conclusion

We found a condition to achieve a long period of the RSA decryption function, it is essential to assure the security of the authentication protocol of the IROTP. We have also given an algorithm to get suitable parameters of the IROTP which satisfies the condition. Our result directly leads to the implementation of the IROTP with sufficient security.

References

1) : OpenSSL, `http://www.openssl.org`.
2) Erdös, P., Pomerance, C. and Schmutz, E.: Carmichael's lambda function, *Acta Arithmetica*, Vol.58, pp.363–385 (1991).
3) Everest, G. and Ward, T.: *An Introduction to Number Theory*, Springer (2005).
4) Haller, N.: The S/Key One-Time Password System, RFC1760 (1995).
5) Koblitz, N.: *A Course in Number Theory and Cryptography*, Springer (1994).
6) Miller, G.L.: Riemann's Hypothesis and Tests for Primality, *Journal of Computer and System Sciences*, Vol.13, No.3, pp.300–317 (1976).
7) Motwani, R. and Raghavan, P.: *Randomized Algorithms*, Cambridge University Press (1995).
8) Rabin, M. O.: Probabilistic algorithm for testing primality, *Journal of Number Theory*, Vol.12, No.1, pp.128–138 (1980).
9) Rivest, R., Sharir, A. and Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, pp. 120–126 (1978).
10) Yatsukawa, N.: Authentication system using authentication information valid one-time, US Patent : US-6148404 (2000).
11) Yatsukawa, N.: One-Time Password Authentication System Using Public-Key Cryptography (2011). to appear.