International Journal of Cryptology Research 9(2): 26 - 30 (2019)

Forging the TNC Digital Signature

Muhammad Rezal Kamel Ariffin[1,2]

[1]Al-Kindi Laboratory of Cryptography, Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia
[2]Department of Mathematics, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia

E-mail: rezal@upm.edu.my
**Corresponding author*

ABSTRACT

In this write up we put forward forgery scenarios for the TNC digital signature scheme for legitimate messages. The process is done by observing previous verification parameters from valid TNC digital signatures. While the analysis does not provide a mechanism that enables one to forge the TNC digital signature for any message, the result does suggest that users of TNC digital signature will need extra storage space and also need to compute extra operations in order to avoid adversaries to be able to create such forged signatures. To this end, users will need to store all previous signatures and make comparison with them. This is in contrast with the popular RSA signature scheme in the sense that the RSA digital signature scheme does not have such requirements to check with previous signatures in order to ensure forgery will not occur in the future.

KEYWORDS: TNC digital signature scheme, RSA digital signature scheme, digital signature forgery

1 INTRODUCTION

The cryptanalysis exercise upon the TNC digital signature is an attempt to obtain all or some of the following objectives:

1. To obstruct or negate the message integrity checking capabilities;
2. To obstruct or negate the message authenticity checking capabilities;
3. To obstruct or negate the non-repudiation capabilities;

Results provided in this write up will provide new insights on the objectives listed above.

2 FIRST CRYPTANALYSIS RESULT

We observe the first result via the following proposition.

Proposition 2.1. *Suppose we have the signature pair* (c_i, s_i) *and* (c_j, s_j) *from previous digital signatures such that* $c' = H(A_iA_j, B_iB_j, m_0)$ *where* $A_k = g^{s_k}y_1^{-c_k} \pmod p$*,* $B_k = y_1^{s_k}y_2^{-c_k} \pmod p$*,* $c' = c_i + c_j \pmod q$ *for our message choice* m_0*. Let* $s' = s_i + s_j \pmod q$*. Then* (c', s') *is the forged signature for the selected message* m_0 *for user with the parameters* (y_1, y_2, g, p, q, H) *for the corresponding private exponent* a*.*

Proof. Observe that the values (A_i, A_j, B_i, B_j) are obtained from the i-th and j-th verification procedures previously. Furthermore $s' = s_i + s_j = a(c_i + c_j) + r_i + r_j = ac' + r_i + r_j \pmod q$.

Compute

$$A' = g^{s'}y_1^{-c'} = g^{ac'+r_i+r_j}g^{-ac'} = g^{r_i+r_j} = A_iA_j \pmod p \qquad (1)$$

$$B' = y_1^{s'} y_2^{-c'} = g^{a(ac'+r_i+r_j)} g^{-a^2c'} = g^{ar_i} g^{ar_j} = B_i B_j \pmod p \tag{2}$$

Finally, $c' = H(A', B', m_0)$. ■

□

3 SECOND CRYPTANALYSIS RESULT

The next proposition provides our second cryptanalysis result.

Proposition 3.1. *Suppose we have the signature pair* (c_1, s_1) *from a previous digital signature such that* $c' = H(A_1^{c^*}, B_1^{c^*}, m_0)$ *where* $A_k = g^{s_k} y_1^{-c_k} \pmod p$*,* $B_k = y_1^{s_k} y_2^{-c_k} \pmod p$*,* $c' = c_1 c^* \pmod q$ *for some random number* c^* *and our message choice* m_0*. Let* $s' = s_1 c^* \pmod q$*. Then* (c', s') *is the forged signature for the selected message* m_0 *for user with the parameters* (y_1, y_2, g, p, q, H) *with the corresponding private exponent* a*.*

Proof. Observe that the values (A_1, B_1) are obtained from earlier verification procedures previously. Furthermore $s' = s_1 c^* = ac_1 c^* + rc^* = ac' + rc^* \pmod q$.

Compute

$$A' = g^{s'} y_1^{-c'} = g^{ac'+rc^*} g^{-ac'} = g^{rc^*} = A_1^{c^*} \pmod p \tag{3}$$

$$B' = y_1^{s'} y_2^{-c'} = g^{a(ac'+rc^*)} g^{-a^2c'} = g^{arc^*} = B_1^{c^*} \pmod p \tag{4}$$

Finally, $c' = H(A', B', m_0)$. ■

□

4 DISCUSSION -1

In order to avoid Proposition 1, all previous signatures need to be stored by signer in order to conduct a checking procedure upon future signatures to verify that there were no previous parameters satisfying the relations mentioned available for adversary to conduct such forgery attempts.

5 DISCUSSION -2

In order to avoid Proposition 2, signer has to ensure that for some random c^*, there will be no corresponding m_0 that produces c' which satisfies the relations mentioned.

6 DISCUSSION -3

Assume that the RSA signature given by $S = H(m)^s \pmod N$ can be used to forge our message choice m_0. That is,

$$S' = H(m_0)^s \pmod N \tag{5}$$

where $m_0 \neq m$ and $H(\cdot)$ is a suitable hash function. In order for (5) to be successful, the adversary needs to obtain the signing key s from the public verification key (v, N) where

$$sv \equiv 1 \pmod{\phi(N)} \tag{6}$$

It is known that the difficulty governing (6) is:

1. Integer Factorization Problem on $N = pq$;

2. The RSA Diophantine key equation

where both are not known to be able to be solved in polynomial time via classical computers. Thus, previous RSA signature parameters S are not able to be utilized for forgery of future messages.

7 CONCLUSION

We conclude that one of the considerations needed in order to utilize the TNC digital signature securely is for each user to setup a database of all previous signings and to conduct checking mechanisms upon the past signatures for future signatures. We also note here that this mechanism does not exist within the RSA signature scheme and thus might be inefficient for resource constraint devices.