ISSN No. 0976-5697

Volume 2, No. 1, Jan-Feb 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

NTRU Digital Signature Scheme - A Matrix Approach

Rakesh Nayak*
Associate Professor: Department of IT
Sri Vasavi Engineering College, Tadepalligudem.
Andhra Pradesh, India
nayakrakesh8@gmail.com

Dr. Jayaram Pradhan
Professor: Department of Computer Science
Behrampur University, Behrapmpur
Odisha, India
jayarampradhan@hotmail.com

Dr. C.V.Sastry
Professor: School of Computer Science and informatics
Sreenidhi Institute of Science and Technology
Hyderabad, Andhra Pradesh, India
cvsastry40@yahoo.co.in

Abstract: Procedures used for digital signature development rely on the algorithms provided for data confidentiality. The decryption algorithms used for data confidentiality are generally inverse of the encryption algorithms and hence can be used for authentication purposes. Existing Algorithms based on truncated polynomials offer many advantages compared to RSA. At the same time these truncated polynomial algorithms suffer from the drawback that the encryption and decryption algorithms are not inverses of each other as in the case of RSA, and hence require separate computation for generation of digital signature.

In this paper we propose an algorithm based on matrix approach of NTRU[3] that combines the advantages of both RSA and the algorithms based on truncated polynomials and at the same time overcoming the difficulties faced in each. This proposed algorithm does not require either large prime number to be generated or extra computational effort for digital signature generation.

Keywords: Encryption, Decryption, Digital Signature, Message Digest

I. INTRODUCTION

With the fast growing of network and media techniques, there has been growing interest in developing effective techniques to discourage the unauthorized duplication of digital data like audio, image and video. In traditional method, cryptology is often used to protect them, but when the cryptograph has been decoded, copying and republishing of the digital data would be out of control.

The purpose of the digital signature is to establish the identity of the document's signer. If the secret key of a signer is compromised, then all of the past signatures become worthless. It is possible for a signer to deny ever signing the message by claiming that the private key has been compromised. The ordinary digital signatures have this limitation.

A digital signature serves the same purpose as a handwritten signature. A handwritten signature is easy to counterfeit while a digital signature is superior to a handwritten signature in that it is very difficult to counterfeit. At the same time digital signatures are used to authenticate information: that is, to securely tie the contents of an electronic document to a signer. Only the true signer should be able to produce valid signatures, and anyone should be able to verify them in order to convince oneself that the signer indeed signed the document.

II. GENERAL METHOD

The digital signature for a message is generated in two Steps:

A. Generation of Message Digest: A message digest [2] can be treated as the 'summary' of the message we are going to transmit, i.e it will be a number unique to every message such that even the slightest change in the message produces a different digest. The message digest is generated using a set of hashing algorithms.

B. Encryption: The message digest is encrypted using the sender's private key. The resulting encrypted message digest is the digital signature. The digital signature generated by the above method is then attached to the message, and sent to the receiver.

At the receiving end, the receiver then does the following:

A. Using the sender's public key, the receiver decrypts the digital signature to obtain the message digest generated by the sender.

B. The same message digest algorithm used by the sender is used to generate a message digest of the received message by the receiver also.

C. Compares both messages digests. If they are same, the message is authenticated.

We can be sure that the digital signature was sent by the sender, because only the sender's public key can decrypt the digital signature. If decrypting through the public key renders a faulty message digest, this means that either the message or the message digest is not exactly what the sender sent.

III. PROPOSED METHOD

In matrix approach of NTRU Cryptosystems[5], matrices are used instead of truncated polynomials. The message is represented in terms of a N x N Square matrix. After encryption and addition of the message digest, (N+2) x (N+2) matrix is sent.

The digital signature for a message is generated in the following steps:

[a] Generation of First Message Digest: The message digest is generated using a set of hashing algorithms for each row. The message digest can also be generated by calculating Eigen values of the matrix. These Eigen values are represented as a column matrix and added to the encrypted message as two additional columns before sending. Two columns are added as the Eigen values may be complex numbers. The first column is used for the real part and the second column is used for imaginary part. At the same time take integer values of irrational numbers.

[b] Encryption: The message is encrypted using the sender's private key.

[c] Generation of Second Message Digest: The second message digest is generated using a set of hashing algorithms for each column. The message digest can also be generated by calculating Eigen values of the encrypted matrix. These Eigen values are represented as row matrix and are added to the encrypted message as additional rows. Similar to the first message digest, here also two rows are added, one for real part and one for imaginary part and integer values of irrational numbers.

After adding the two message digests to the encrypted message, a (N+2) x (N+2) message is formed, with the last two element of (N+1)th and (N+2)th row/column as 0. This (N+2) x (N+2) matrix is sent. At the receiving end, the receiver then does the following:

[a] The receiver separates the encrypted message and the message digest from the received message by taking the first N X N matrix. The (N+1)th and (N+2)th row/column are the message digest.

[b] Before decrypting the message, he tries to generate the message digest of the received encrypted message. If the message digest of receive message matches with the message digest calculated, he proceeds further to decrypt it. Otherwise, he rejects the message.

[c] Using the sender's public key, the receiver decrypts the message and generates the message digest for the decrypted message.

[d] Compares the message digest with the first messages digest. If they are same, the message is authenticated.

When the message digest of encrypted message found to be same that of the calculated message digest of encrypted message, it ensures that no change in message during transmission. It means no intruder has tried to change the message. When the message digest of original message is found same that of the calculated message, it ensures the sender is well aware of what he has sent.

A. Terminology Used

Let the message M be a N x N matrix. The sender needs to generate a matrix F of degree N. Fq is the inverse of the matrix F mod q (i.e. [Inverse[F]](mod q)) [5], where q is a relatively prime to p. Similar to NTRU , F needs to be carefully chosen such that Fq exists. G is another matrix of order N x N. R is a random matrix of order same as F. The values of p and q may be made public.

B. The Algorithm for Signature Generation

The public key h is calculated as H= Fq + p * G. The private key is F, and the values of Fq need to be maintained in secrecy. Find the Eigen values of the message M. This Nx2 matrix of Eigen values (taking care that the Eigen values may be complex) serves as a message digest to be added to the encrypted message as two additional columns. Encrypt the message M as E= F*(M +R*q), where R is a random matrix. Again find the Eigen values of the encrypted message. This Nx2 matrix is added as two extra rows to the encrypted text padding it with zeroes to make it an (N + 2) x (N+2) matrix. Let us call this E1.

The receiver receives E1. He separates (N+1)th and (N+2)th row/column to get the N x N matrix encrypted message. The receiver finds the Eigen values of the encrypted message E and compares with the (N+1)th and (N+2)th row matrix. If both are same, further decryption is necessary.

The method for decryption is as follows:

Let the receiver calculate A = (H * E) mod q

Then Computes B = A mod p

C. Analysis of the Signature Generation Algorithm

In the encryption process since a random polynomial R is added to the message digest value and then multiplied by A, it is exponentially complex to find the private key A, even if the message digest value is known by the receiver after decryption. Hence this proposed algorithm is secure even if the values of N, p, q and H are known to everybody.

At the receiving end the receiver calculates

A = (H * E) (mod q)

= (Fq + p * G) *(F*(M+R*q))(mod q)

=(Fq*F(M+R*q) + (p*G)*F*(M + R*q))(mod q)

= (M+ p*G*F*M)(mod q)

[As F*Fq = I, (R*q) (mod q) = 0,(p*G*F*R*q)(mod q) =0]

B = (A)(mod p)

= (M+ p*G*F*M)(mod p)

= M [(p*G*F*M)(mod p)=0]

D. Algorithm for Providing Data Confidentiality

As described in earlier the public key is H and its corresponding private key (F, Fq). Let the message be M.

E, which is the cipher text generated by E = F*(M + R*q)(mod q). This cipher text that has been generated is decrypted at the receiving end using the receiver's public key By calculating

A = (H * E) (mod q) and B = A (mod p), which is the actual message M

Example: Let p = 3 and q=32

Let F = {{1,0,1},{1,1,1},{0,1,1}}, Its corresponding Inverse Modulo q i.e., Fq= {{0,1,31},{31,1,0},{1,31,1}}

Let R= {{1,0,1},{1,0,1},{0,1,0}} and

G = {{1,0,1},{1,1,0},{0,1,1}}

The public key H= (Fq + p*G)(mod q) =

{{3,1,34},{34,4,0},{1,34,4}} (mod q)=

{{3,1,2},{2,4,0},{1,2,4}}

Consider the message m={{1,0,1},{1,1,1},{1,1,1}}, its Eigen values are {2,0,0}. So represent it as {{2, 0},{0,0},{0,0}}

The cipher text E = F*(M + q*R)(mod q)=

{{34,33,34},{67,34,67},{34,34,34}} (mod q)=

{{2,1,2},{3,2,3},{2,2,2}}

The Eigen values of the cipher text are {6, 0, 0}. So represent it as transpose {{6, 0},{0, 0},{0, 0}}

The Eigen values of the cipher text and the original message are used as message digest, which is added as additional row and column of the encrypted message. With last element of the additional row/column is 0.

The encrypted message including the message digest, E1= {{2,1,2,2,0},{3,2,3,0,0},{2,2,2,0,0},{6,0,0,0,0},{0,0,0,0,0}}

When the receiver receives the message E1, he first takes out first N X N matrix as encrypted message and the remaining (N+1)th and (N+2)th row and (N+1)th and (N+2)th column as message digest before encryption and after encryption.

So the encrypted message he gets as

{{2,1,2},{3,2,3},{2,2,2}}

and the Eigen values is calculated as {{6,0},{0,0},{0,0}} which matches with the message digest of received message. It ensures that non interference of intruders.

As a result the decryption process of the received message starts.

A= (H.E)(Mod q)= {{13,9,13},{16,10,16},{16,13,16}} (mod q)= {{13,9,13},{16,10,16},{16,13,16}}

At last, B is calculated as

B= A (mod p) = {{13,9,13},{16,10,16},{16,13,16}} (mod p) ={{1,0,1},{1,1,1},{1,1,1}}, which is the Original message. The Eigen value of this is {{2, 0},{0,0},{0,0}}, which shows the originality of the message.

IV. COMPARISION

This algorithm can also be applicable by using polynomials [4] instead of matrix. The basic operations involved are addition, multiplication and modular operations. As represented in [5], the matrix operation takes less times as compared to polynomials. Apart from this, we need a N value in polynomial approach in order to have the polynomial truncated polynomial to it. So it needs a extra operation, which lead to some extra time added to each basic operations.

While encrypting/decryption using polynomial we need operations, like polynomial multiplication and polynomial addition. The algorithms that can be used are.

```
void polymul(int a[S],int b[S],int f[S], int n,int m)
{
int k,j,i;
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
k= i+j;
if (k>=n)
k=k-n;
f[k]=(f[k] + a[j]*b[i])%m;
}}}
```

The above function multiplies two polynomial of order n.

a[S] and b[S] are two polynomial of order n which are to be Multiplied and result is saved in another polynomial f [S]truncated to n. m gives the modular operation on the coefficients. The complexity of this operation is O(n2).

```
void polyadd(int a[S],int b[S],int f[S],int m)
{
int d[S],k,j,i;
for(i=0;i<S;i++)
f[i]=(a[i]+b[i])%m;
}
```

This function gives the sum of two polynomial modulus m. here a[S] and b[S] are the polynomial are to be added. And the result is saved in f[S]. The complexity of this operation is O(n). Similarly, we need multiplication and addition operation for encryption/decryption of data using matrix. The algorithms we use are.

```
void matmul(int a[S][S], int b[S][S], int c[S][S],int m)
{
int i,j,k,sum;
for(i=0;i<S;i++)
{
for(j=0;j<S;j++)
{
sum=0;
for(k=0;k<S;k++)
{
sum=(sum+(a[i][k]*b[k][j]))%m;
c[i][j]=sum;
} }}}
```

The above function adds two matrix of order NxN. a[S][S] and b[S][S] are two matrix of order NxN which are to be multiplied and result is saved in another polynomial c[S][S].

m gives the modular operation . The complexity of this operation is O((n½)3).

```
void matadd(int a[S][S],int b[S][S], int c[S][S] )
{
int i,j,k,sum;
for(i=0;i<S;i++)
{
for(j=0;j<S;j++)
{
c[i][j]=a[i][j]+b[i][j];
}
}}
```

The above function multiplies two matrix of order NxN. a[S][S] and b[S][S] are two matrix of order NxN which are to be multiplied and result is saved in another polynomial c[S][S] m gives the modular operation. The complexity of this operation is $O(((n½)^2)$.

V. CONCLUSION

In this paper we have proposed an authentication scheme based on Digital Signatures using a Matrix Approach. The parameters F, R and G need to be chosen for encryption with care. The mathematical process involved requires a lot of modular arithmetic on matrices, to be performed. For better security large values of q need to be chosen, which in turn leads to complex and time consuming mathematical operations. To simplify this, Montgomery method may be applied to reduce the overheads involved in arithmetic operations. The algorithm presented in this paper can be used for both signature generation and for providing data confidentiality unlike a few other algorithms working on polynomial arithmetic. As the Eigen values are unique and any change in the matrix reflects a difference in the Eigen values so it can be used as message digest

VI. ACKNOWLEDGMENT

The authors would like to express their cordial thanks to the
Management of Sri Vasavi Engineering College, Tadepalligudem, to support.

VII. REFERENCES

[1] R.L. Rivest. RFC 1321: The MD5 Message-Digest Algorithm. Internet Activities Board, April 1992.

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

[3] Rakesh Nayak, C.V.Sastry, Jayaram Pradhan "A Matrix Formulation for NTRU Cryptosystem". Proceedings 16th IEEE International Conference on Networks (ICON 2008), New Delhi.

[4] P.Prapoorna roja and P.S.Avadhani "Digital Signature Development using Truncated Polynomials". IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.7, July 2007 pp 130 to 133.

[5] Rakesh Nayak, C.V.Sastry, Jayaram Pradhan "Algorithmic Comparison between Polynomial Base and Matrix Base NTRU Cryptosystem", International Journal of Computer and Network Security, (IJCNS), Vol. 2, No. 7, July 2010 pp 58 to 63.

[6] NTRU Cryptosystem, Technical Reports 2002 available at http://www.ntru.com.