www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 7 Issue 6 June 2018, Page No. 24071-24075
Index Copernicus Value (2015): 58.10, 76.25 (2016) DOI: 10.18535/ijecs/v7i6.09

NTRU based Security in Cloud Computing

Arshpreet Kaur[1], Miss Lofty Sahi[1]

[1]Chandigarh Engineering College Landran Mohali

Abstract

The users of Cloud Computing keep their facts onto the third party owners and experience the online demand programs, services and garage from a shared pool of configurable computing resources, without the load of facts garage and protection and costs. The furnished security must guarantee no longer best on authentication but also on files over the cloud for the outsourced records, that is now maintained by way of third parties consisting of cloud carriers. Unluckily, the infrastructure of cloud computing is constructed on internet, and will stumble upon all of the risk of net. To lower the risks, an authentication mechanism is the viable answer. However, conventional alphanumeric password authentication mechanism is not always relaxed sufficient. A cozy authentication mechanism, the use of totally public-key encryption based password is proposed in this paper for enhancing traditional authentication mechanism and let users get entry to cloud services securely.

Keywords – Cloud computing

1 Introduction

This research focuses on mechanism of generating new public key using NTRU algorithm (nth degree truncated polynomial) for the purpose of security and it gets encrypted according to the parameters. By means of doing this, no unauthorized users can recognize the credentials. Hence the software is stored from numerous community threats e.g. hacking. NTRU is a patented and open supply public key cryptosystem that makes use of lattice-based cryptography completely to encrypt and decrypt statistics. It includes algorithms: NTRUEncrypt, that is used for encryption, and NTRUSign, that is used for digital signatures. The NTRU Encrypt is a public-key cryptosystem which is based on the shortest vector hassle. Its most important characteristics are the low key reminiscence and computational requirements as presenting a high protection stage. In contrast to other popular public-key cryptosystems, it is proof against attacks, the use of Shor's set of rules and its overall performance is notably higher.

2 Literature Survey

M. FahimFerdous Khan [1] focuses on the access control issued in healthcare, with the goals of designing and developing access control mechanisms contingent upon various environmental and application-dependent contexts with provision for secure delegation of access-control rights.

Raghavendra Mishra [2] presents a cloud user-server anonymous mutual authentication framework in which user and server authenticate each other and establish a session key without disclosing users' original identity over the public channels. Moreover, user can change its private key periodically to avoid key compromise impersonation attack, which also enhances anonymity during communication. Establishment of secure session between the user and server without losing user anonymity has been made.

Brijesh Kumar Chaurasia et al.[3] In his research for uniform strong authentication and obviate the need for password registration, two factor authentication (T-FA) has been proposed. The data owner provides one of the credentials and is a two tier mechanism. T-FA needs two identities based on what the user knows and what he possesses which is implemented through Software as a Service (SaaS) in the cloud computing environment.

Jiaqing Mo et al.[4]In this research he puts forward a user identity authentication scheme based on trusted platform for Cloud Computing. In this scheme, the cloud user registers in the trusted certificate authority (CA), and obtains the certificate issued by CA. Afterwards, the

certificate is sent to the cloud server, and the cloud server verifies the validity of the remote user identity according to the certificate. At the same time, this scheme provides mutual authentication while it establishes communication key between the remote user and cloud server. The analysis shows that this scheme is secure against insider attack, replay attack, backward/forward attack, and forgery attack. Compared with the related work, the scheme has higher computing efficiency and less interaction rounds.

Saurabh Deyet al. [5] presented a novel authentication scheme, Message Digest-based Authentication (MDA). MDA strategically incorporates hashing, in addition to traditional user ID and passwords, to achieve mutual authentication. The effectiveness of MDA is validated with Scyther, a widely-used security protocol analyzer. Results indicate that MDA is capable of withstanding a variety of different security attacks, such as man-in-the-middle, replay attacks etc.

Qi Jiang et al.[6] proposed a two-factor authentication protocol based on elliptic curve cryptosystem which enables cloud users to access their outsourced data. However, their scheme suffers from the problem of wrong password login and is prone to denial of service attack in the password-changing phase. It also fails to provide user revocation when the smart card is lost or stolen. To remedy these flaws, an improved two-factor authentication and key agreement protocol has been introduced.

Sneha K. Khodke[7] presented the design for providing security and higher authentication scheme for executing secure data transaction in an Organization field over Internet. In case of Cloud computing, the whole authentication control lies toward the server side. So, it is very tough to trust the third party server in Cloud Computing. This work proposes a scheme in which authentication process is carried out in two levels or multi-levels. In this system, firstly activity happens at organization level. It reads the authentication password and checks to cloud access for organization and then it enters into a second level authentication which happens at team level. It reads the team login details and checks for authentication and then enters into a user level authentication where the authentication information is read to check for the user permission and privileges.

Ankit Dhamija [8] proposes a simple, convenient & secure hardware based two tier technique using Universal Serial Bus (USB). Proposed model provides solution to the limitations posed by the hardware based OTP scheme where a user is supposed to enter a pin or password, received on their mobile handset, on the web portal of the Cloud Service Provider. So this scheme defies the possibility of a phishing attack and brute force attack by any intruder of stealing that OTP or pin and misusing it

BinuSumitra et al.[9] in his research proposes a user authentication framework for Cloud which facilitates authentication by individual service providers as well as by a third party identity provider. The proposed two-factor authentication protocols use password as the first factor and a Smart card or Mobile Phone as the second factor. The protocols are resistant to various known security attacks.

Jeffrey Hoffstein, Jill Pipher, and Joseph Silverman[10] NTRU Cryptosystems develop and markets a Public-key cryptosystem. NTRU has targeted on the embedded markets (e.g., mobile phones and RFID chips) where processing electricity is scarce. Certainly one of NTRU's competitors is RSA. A few preliminary timing comparisons have been made between NTRU and RSA. The NTRU software is written in C and not optimized for speed. The principle makes use of to which PC's are carried out are the alternate of secret keys and short messages. Additionally, RSA, ECC and NTRU all work in gadgets of message blocks, and any message block in any of these structures is big enough to keep a secret key of very high safety, or a quick message. Numbers given for encryption and decryption are message blocks processed in line with second. In spite of the slower clock pace, NTRU comes out 16, 18 and 33 times quicker at encryption, decryption and key creation at moderate security and 2, 3 and 8 instances quicker at high security. NTRU's Cryptosystem affords for each encryption and digital signature referred to as NTRUEncrypt and NTRUSign respectively. NTRU is broadly famous because of its safety in opposition to quantum computer systems.

3. **Proposed Technique - Ntru Algorithm**

NTRU keys, parameters

N The polynomials in the ring R have degree N-1. (Non-secret)

q The large modulus to which each coefficient is reduced. (Non-secret) p The small modulus to which each coefficient is reduced. (Non-secret) df A polynomial that is the Private key space. Fixes the polynomial form defining the numbers of positive ones for the private key f, the negative ones are fixed by df - 1. dg Public key space. Fixes the polynomial form defining the number of positive and negative ones for the random polynomial g used to calculate the public key.dr Blinding value space. Fixes the polynomial form defining the number of positive and negative ones of the random polynomial r used in the encryption process. dm Plaintext space. NTRUEncrypt requires the message to be in a polynomial form, therefore the need of dm to defining the form of the message to be encrypted. NTRU Some Definitions

NTRU public-key algorithm is well described using the ring of polynomials $R=Z[X]/(X^N-1)$. The polynomials conforming R have integer coefficients: $a(X)= a_o + a_1X + a_2X^2 + ……..+a_{N-1}X^{N-1}$ that are multiplied together using the extra rule $X^N==1$ The product $C(X) = a(X)* b(X)$ is given by $c_k = a_0b_k + a_1b_{k-1} + a_2b_{k-2} ……..a_{N-1}b_{k+1}$

NTRU Key Generation

The key generation consists in the generation of the private key (f,f_p) and the public key h. Choose random polynomials f and g from R with small coefficients. Meaning small much Smaller than q, typically {-1,0,1} for p = 3. Then compute f_p, i.e. the inverse of f (mod p) defined by f*fp = 1 (mod P)

Compute f_q the inverse of f (mod q) that analogously satisfies the requirement: f*fp = 1 (mod q)

Compute the polynomial $h = g*p*f_q$ The public key is h and the private key is the set (f, f_p)The following is a numerical example for key generation operation. Domain parameters are as follows: N=11 q=32 p=3 We choose a polynomial f such that it is invertible in both modulus p and q . Also we choose the polynomial g which will be used in public key generation.

$f = -1 +x +x^2 -x^4 +x^6 +x^9 -x^{10}$

$g = -1 +x^2 +x^3 +x^5 +x^8 -x^{10}$

Next step is to calculate the inverse of in modulus and namely f_p and f_q

$f_p = 1+2x+2x^3+2x^4+x^5+2x^7+x^8+2x^9$ and

$f_q =5+9x+6x^2+16x^3+4x^4+15x^5+16x^6+22x^7+ 20x^8+18x^9+30x^{10}$

f is the private and the public key is calculated as follows:

$h = pf_q*g = 8+ 25x+ 22x^2+ 20x^3+ 12x^4+ 24x^5+ 15x^6+ 19x^7+ 12x^8+ 19x^9+ 16x^{10}$ (mod 32)

Encryption : The plaintext m is polynomial with coefficient taken mod p. note that convert the message m to a polynomial form is not part of NTRU public-key algorithm. Choose a blinding message r randomly from R with small coefficients. The Ciphertext is e = r * h + m (mod q)

The following is a numerical example representing how the encryption process works. In order to encrypt a message we need a public key h (we use the one which is calculated in the last section) and a random polynomial r besides the message polynomial m. so we chose the random r and the message m as : $r = -1+ x^2+ x^3+ x^4-x^5 – x^7$ $m = -1+ x^3 -x^4 -x^8 +x^9 +x^{10}$

The following operation calculates the encrypted message cipher text $e = r*h+m = 14+ 11x+ 26x^2+ 24x^3+ 14x^4+ 16x^5+ 30x^6+ 7x^7+ 25x^8+ 6x^9+ 19x^{10}$ (mod 32) Decryption: The decryption returns the message m from the encrypted message e using the private key (f,f_p).

Compute a = e*f (mod q)

Choosing the coefficients of a to satisfy

-q/2 <= ai <= q/2 Reduce a modulo p: b = a (mod p) Compute $c = b*f_q$ (mod p) Then c mod p is equal to the plaintext m. The following is a numerical example to demonstrate the decryption operation. One can compute the temporary polynomial a as follows: $a = f*e = 3- 7x- 10x^2- 11x^3+ 10x^4+ 7x^5+ 6x^6+ 7x^7+ 5x^8+ 3x^9- 7x^{10}$ (mod 32). The next step is to reduce the coefficients of a to modulo p. a results is polynomial b = a (mod p) $= - x- x^2+ x^3+ x^4+ x^5+ x^7- x^8- x^{10}$ (mod 3) Next we need to move the next step to calculate the plaintext $c = b*f_q$ (mod p) $= - 1+ x^3- x^4- x^8+ x^9+ x^{10}$ (mod 3)

Choosing Parameters The following is a numerical example to demonstrate the decryption operation. One can compute the temporary polynomial an as follows: Selecting p and q The parameters p and q must be relatively prime and q must be significantly larger than p since in the decryption process we need there to be no change when we reduce the coefficients of the polynomial p|o *g + f*m modulo q . We also want to choose these parameters so that we are able to find inverses of f modulo p and q . In general, recall that for f to have an inverse in a ring K = A[x]/x[N-1] where A is a field, we need that gcd (f, xN-1) = 1. In this case the extended 28 Euclidean algorithm finds as s,t€Ax such that

sf+t(xN-1) = gcd (f, xN-1) = 1. It follows that s is the inverse of in the ring . Now consider the problem of finding an inverse of modulo. That is we want to find the inverse of f in R/p. Assume that p is prime. Then R/p is isomorphic to A[x]/xN-1 where A = Z/pZ. Since Z/pZ is a field the extended Eculiden algorithm can be used to determine whether there exists an inverse of f in R/p and will find the inverse if it exists. If p is a power of a prime, say p=Qr with Q prime, then there is an algorithm to find the inverse modulo p=Qr assuming inverse modulo Q is known. Hence p and q are often chosen to be primes or prime powers. A couple of common combinations for p=2 and q= 2n. Taking to be a power of 2 allows for efficient reductions modulo q . Current NTRU parameter often take q = 2048. Using p = 3 implies that messages are trinary polynomials with coefficients in {-1,0,1}. Note that in general the NTRU encryption and decryption algorithms do not require p be an integer. It may be taken as a small polynomial as long as p and q are relatively prime in the ring R . For example current NTRU implementations often taken p = 2+x. Since this is relatively prime to 2, they take q as a power of 2 as in the trinary case and their messages are binary polynomials with coefficients 0 and 1. They also use a different algorithm for reducing a polynomial modulo p=2+x as described it is no longer simply a reduction of coefficients.

4 Proposed Security Technique

In the research there is a gap that every time same key is used to encrypt the credentials before access to cloud and hacker can track the key via applying various combination to decrypt the credentials. To solve this we make use of NTRU algorithm which is as follows

1 User logins by entering the credentials. Username credentials gets encrypted using NTRU algorithm.

2 Encrypted text is sent to cloud for decryption.

3 Credentials are verified from database by cloud.

4 If the user matches, access is given to the user.

5. Results and Discussion

Now we discuss results. We have made the entire project by using java. The result is as follows.

Table 1

	Existing	Proposed
Execution Time level 0	35 sec	20 sec
Execution operation time	2 sec with Respect to 100 Request	1 sec with Respect to 100 Request
Total execution time for encryption and decryption	5 sec encryption, 5 sec decryption	3 sec encryption, 3 sec decryption

Figure 1

Figure 2

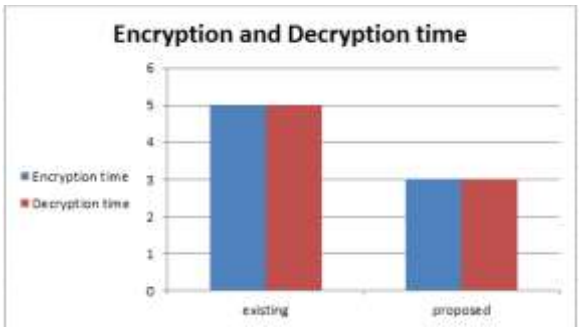

Figure 3

6 Conclusion

NTRU is used for the encryption because of the various advantages that it possess. They are that encryption and decryption is more efficient, in both hardware and software implementations. It has much faster key generation allowing the use

of "disposable" keys (because keys are computationally "cheap" to create). It's low memory use allows it to use in applications such as mobile devices and smart-cards. Parallel implementation lets in for benefits of NTRU on pinnacle of current crypto infrastructure with a negligible performance penalty. NTRU protects information that needs to remain secret for many years. It keeps encrypted traffic from being recorded and warehoused today and decrypted when quantum computers are available. It is ideal for systems that have long lifecycles or can't be updated easily.

References

[1] M. Fahim Ferdous "Context Aware Access Control for Clinical Information System". Innovations in Information Technology (IIT), Tokyo, Japan 123-128 Publication year 2012.

[2] Raghavendra Mishra "Anonymous Remote user Authentication and key Agreement for Cloud Computing" Proceedings of the Third International Conference on Soft Computing for Problem Solving 899-913 10.1007/978-81-322-1771-8_78 Publication year 2014.

[3] Brijesh Kumar Chaurasia "Authentication in Cloud computing Environment Using Two Factor Authentication" Proceedings of the Third International Conference on Soft Computing for Problem Solving 779-785 10.1007/978-81-322-1768-8_67 Publication date 9 March 2014.

[4] Jiaqing Mo, Zhongwang Hu, Yuhua Lin "A User Authentication Scheme Based on Trusted Platform for Cloud Computing" SpaCCS2016: 122-130 10.1007/978-3-319-49148-6_11 Publication year 2016.

[5] Saurabh Deyet "Message Digest Based Authentication for Mobile computing" 10.1186/s13677-016-0068-6 Publication year 2016.

[6] Qi Jiang, Bingyan Li, Jianfeng Ma, Youliang Tian, Yuanyuan Yang "Cryptanalysis and Improvement of a Smart Card Based Mutual Authentication Scheme in Cloud Computing" ICCS 311-321"10.1007/978-3-319-48671-0_28 Publication year 2016.

[7] SnehaK.Khodke "Multi Level Authentication Technique for Access Organization in Cloud Data" International Journal of Advance Foundation and Research in Computer (IJAFRC) Volume 1(2348 – 4853) Publication date 12 December 2014.

[8] Ankit Dhamija "A Two Tier Reliable User Authentication Scheme for Cloud Environment" International Journal of Computer Applications (0975 – 8887) Publication year 2015.

[9] BinuSumitra "Secure and Usable Authentication Framework For Cloud Environment " Advances in Intelligent Systems and Computing(springer) 183-202 10.1007/978-81-322-2650-5_12 Publication date 3 Nov 2015.

[10] Jeffrey Hoffstein, Jill Pipher, and Joseph Silverman "NTRU: A ring-based public key cryptosystem" Algorithmic number theory (1998) 267-288 Publication year 1998.