IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-4 E-ISSN: 2347-2693

Achieving Security in Ad hoc Networks Using Identity and Trust with Key Management

[1]*Pranav Nair, [2]Swapnaja Gunjal, [3]Archana Jadhav and [4]Ruchita Gadsing
[1,2,3,4]*Department of Information Technology,University of Pune, India*

www.ijcseonline.org

Received: Mar/26/2015 Revised: Apr/06//2015 Accepted: Apr/19/2015 Published: Apr/30/ 2015

Abstract: A Network user has to provide susceptible personal information (e.g. name, residence address, credit/debit card number, contact number, driver's license number and date of birth, etc.) when they are requested by some Web page. This exclusive Personal Identity Information may be used to exclusively 2identify, contact and/or locate a particular user. This information may be exploited and abused if not properly protected. An Identity Management (IDM) system is therefore proposed to overcome this problem and helps to decide the access to this information in a secure manner. The concept of key management has been implemented to achieve the goal of trusted communication. The group public key management scheme, trust of a node.

Keywords:—Personal Identity Information (PII), Identity Management (IDM), Service Provider(SP), Trusted Third Party(TTP)

I. INTRODUCTION

The growing popularity, continuing development and maturation of Web computing services is an undeniable reality. Information stored locally on a computer can be stored in the Web, including word processing documents, spreadsheets, presentations, audio, photos, videos, records, financial information, appointment calendars, etc. The information about another entity is maintained by a third party i.e. A Web service provider.

(SP). Trusting a third party requires taking the risk of physical identity helps in secure data transfer over wireless channels through the concept of composite identity and trust based model(CIDT). To validate a node in the network Trust Factor of a node along with its key pair and identity is used. Proposed method works well for self certification scheme of a node in the network.

Assuming that the trusted third party will act as it is expected (which may not be true all the time). Privacy or confidentiality questions may arise whenever some entity stores or processes information in the cloud.
Privacy in Network computing can be defined as "the ability of an entity to control what information it reveals about itself to the cloud (or to the cloud SP), and the ability to control who can access that information".
Communication in Mobile network is done over a shared wireless channel. Responsibility of maintaining the veracity and privacy of data and nodes in the network are held responsible. A lot of approaches using key management has been implemented to attain the goal of trusted communication. The group public key management scheme, physical identity and trust of a node helps in secure data transfer over wireless channels and proposes a composite identity and trust based model (CIDT) which depends on it. Trust Factor of a node along with its key pair and identity is used to authenticate a node in the network. A valid certificate is generated for authentic node to carry out the communication in the network. Self certification scheme of a node in the network works well by this proposed method.

The use of Internet based services in network computing is used to support business processes and rental of IT-services on a value-like basis .It offers an awareness of resources but it can also pose a high risk for data privacy. A single violation can cause significant loss. The heterogeneity of "users" represents a danger of multiple, collaborative threats. In Identity Management computing, entities may have multiple accounts associated with a single or multiple service providers (SPs).

II. LITERATURE SURVEY

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). Remote services are entrusted by cloud computing with a user's data, software and computation.
The potential to eliminate the requirements for setting up

of high cost computing infrastructure for the IT-based solutions and services that the industry uses is held by Cloud computing. A flexible IT architecture, accessible through internet for lightweight portable devices is provided by it.

A many-fold increase in the capacity or capabilities of the existing and new software is allowed by it. The entire data reside over a set of networked resources, enabling the data which can be accessed through virtual machines in cloud computing environment. As these data centers may lie in any corner of the world beyond the reach as well as control of users, there are multifarious security, privacy challenges which are to be understood and taken care of. The possibility of a server breakdown that has been witnessed, rather quite often in the recent times cannot be denied . There are various issues with respect to security and privacy in a cloud computing scenario need to be dealt with.

The use of Internet-based services to support business processes and rental of IT-services on a utility-like basis is allowed by cloud computing. A concentration of resources but also poses risks for data privacy is offered by it. A single breach can cause significant loss. A danger of multiple, collaborative threats are represented by heterogeneity of "users". In cloud computing, entities may have multiple accounts related with a single or multiple service providers (SPs). Sharing sensitive identity information (that is, Personally Identifiable information or PII) along with associated attributes of the same entity across services can lead to mapping of the identities to the entity, tantamount to loss of privacy. Identity management (IDM) is one of the core components in cloud privacy , security and can help alleviate some of the problems associated with cloud computing. Solutions using trusted third party (TTP) in identifying entities to SPs are available. The usage of solutions on untrusted hosts are not recommended by solution providers. We propose an approach for IDM, which is not dependent on TTP and has the ability to use identity data on untrusted hosts. The approach is dependent on the use of predicates over encrypted data and multi-party computing for negotiating the use of cloud service. It uses active bundle—which the middleware agent that has PII data, privacy policies, a virtual machine that will enforce the policies, and has a set of protection technique to protect itself. An active bundle will interact on behalf of a user to authenticate to cloud services using user's privacy policies.

SELECTED SEARCH PROBLEM

The research dilemmas that we approach in this paper are:

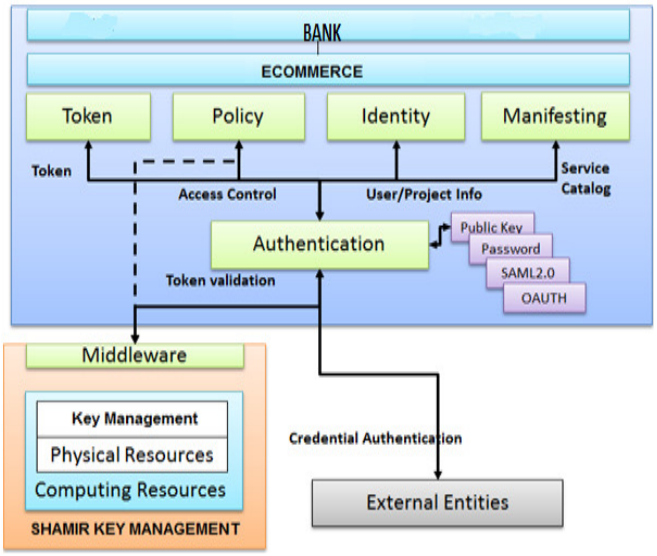
1) Authenticating without disclosing PII: Whenever the user forwards PII to authenticate for a service, the user might encrypt it. Nevertheless, PII is decrypted prior an SP uses it. As soon as PII is decrypted, it becomes prone to attack.

2) Using benefits on untrusted hosts: The available IDM solutions require user to execute IDM from a trusted host. They do not recommend using IDM on untrusted hosts, such as public hosts. Therefore, in cloud computing data may reside anywhere in cloud or in host, this issue needs to be addressed. E.g. User herself might be on a cloud Virtual Machine.

IV. PROPOSED MODEL

Identity management (IDM) is one of the core components in cloud privacy and security and can help alleviate some of the problems associated with cloud computing. Cloud computing has promoted the hosting and delivery of services over the Internet and the movement of computation and data from terminal devices and local servers to core data centre due to advantages in flexibility, scalability, and economics of savings. Most services have been supported by massive-scale distant datacenters located at sites. While the benefits of cloud computing is clear, security is a severe concern in these infrastructures. Privileged user access ensures only authorized users have access to an organization data and resources. Therefore, identity and access management is considered as a security concern in cloud computing. Various models have been proposed to address identity management in clouds, such as central IAM, trusted third party, federation solutions, etc. Most of solutions are mainly focused on federation of cloud providers, and pay less or no attention to access management.

In this project, we propose a new architecture to manage identity and control access to resources in a cloud infrastructure. First, we discuss system requirements, and then we propose architecture to address these requirements. Our architecture comprises two major components: middleware and central IAM (Identity Access Management) to manage user and infrastructure related data. Middleware sits in front of a resource provider and handles time-consuming decision making such as authentication and authorization, while the repository handles data manipulation without disclosing the user identity.



This section describes the proposed IDM approach, which is based on IDM using the active bundle scheme, computing predicates over encrypted data and multi-party computing.

The salient features of the approach are:

1) Ability to authenticate without disclosing unencrypted data. This is achieved by using predicate over encrypted data.

2) Ability to use identity data on untrusted hosts. This is achieved through the use of the active bundle scheme.

An active bundle has a self-integrity check mechanism, which triggers apoptosis (a complete self-destruction) or evaporation (a partial self-destruction) when the check fails.

3) Independence of TTPs. This is achieved through the use of multi-party computing, in which secrets are split into shares distributed to different hosts.

A. Use of Predicates with Encrypted Data and Multiparty Computing

We use a predicate encryption scheme and multi-party computing for giving answer to predicates about PII.
Shamir proposes threshold secret sharing. First, a secret data item D is divided into n shares D1,..., Dn. Then, a threshold k is chosen so that: (a) to recover D, k or more of arbitrary Di's are required; (b) using any k-1 or fewer Di's leaves D completely undetermined.
Ben-Or, Goldwasser and Wigderson define a protocol for multi-party computing of a function f using secret input from all the parties. The protocol involves n "regular" parties, which calculate only partial function outputs. In, one of the players is selected as the dealer (denoted by us as DLR), and is provided the partial function outputs to find out the full results of function computation.

1) Let f be a linear function of degree n known to each of the n parties, and t be an arbitrary threshold value. Let Pi denote Party i, and xi denote the secret input of Pi for f. Dealer DLR will receive from the n parties the partial outputs of f calculated by the n parties using their respective secret inputs x1 , x2, …, xn. Let α 1, α 2, …, α n be distinct non-zero elements in the domain of f. Player Pi is assigned the point αi.

2) Each party Pi generates a polynomial hi of degree t(where t is the above threshold value) such that hi(0) = xi.

3) Each Pi sends to each Pj (from the subset of the other n-1parties) one share si,j = hi(αj) of Pj's input. Then, each Pi computes a portion of function f using shares si,j of the input that it has (its own) or received from n–1 other parties.

A predicate encryption scheme allows evaluating predicates with encrypted data. For example, Alice can compute the predicate "(email sender = 'Bob') and (date in [2006, 2007])" using encrypted data. Fig shows a sample predicate encryption scheme that has this property. Alice uses a Setup algorithm to generate a public key PK and a secret key MSK. Next, Alice uses PK to encrypt (with algorithm Encrypt) her PII and gets ciphertext CT. Then, she can store CT (the encrypted PII) on an untrusted host (e.g., in a cloud). She may also publish PK, so that it can be used to encrypt data that she can access. Alice has the function p representing a predicate that she wishes to evaluate for her encrypted PII. She uses the KeyGen algorithm, PK, MSK and p to output the token TKp (encoding p). Then, she gives TKp to the host that evaluates the token (with p included in the token) for CT (the encrypted PII), and returns the result p(PII) to Alice. Note that KeyGen uses the secret key MSK as input.

Hence, Alice can use KeyGen to generate TKp for p. Alice can give TKp to an untrusted host while protecting PII.

(Observe that if Alice gave KeyGen and MSK to the host, the scheme would not be secure—it would not protect PII.)

1.	Setup	PK, MSK
2.	Encrypt(PK, PII)	CT
3.	KeyGen(PK, MSK, p)	TK^p
4.	Query(PK, CT, TK^p)	p(PII)

Figure 3. The Public-key Predicate Encryption Scheme

For negotiating use of a cloud service, we combine computing predicate over encrypted data with secure multiparty computing. The secret key MSK is split between n parties 1 using the above-mentioned

Shamir's technique.
Then, the algorithm KeyGen is provided to n parties, and computed by them collaboratively using their shares of the secret key, function p representing a predicate, PK, and TK. This is done as specified in the above-mentioned protocol of Ben-Or, Goldwasser and Wigderson for multi-party computing.

In our algorithm, an owner O encrypts PII using algorithm Encrypt and O's public key PK. Encrypt outputs CT—the encrypted PII. SP transforms his request for PII to a predicate represented by function p. Then, SP sends shares of p to the n parties who hold the shares of MSK. The n parties execute together KeyGen using PK, MSK, and p, and return TKp to SP. Next, SP calls the algorithm Query that takes as input PK, CT, TKp and produces p(PII) which is the evaluation of the predicate. The owner O is allowed to use the service only when the predicate evaluates to "true".

V.EXPERIMENTS, RESULTS, DISCUSSION

SHAMIR KEY ALGORITHM:

Shamir's Secret Sharing is an algorithm in cryptography created by Adi Shamir. It is a form of secret sharing, where a secret is divided into parts, giving each participant its own unique part, where some of the parts or all of them are needed in order to reconstruct the secret.

Counting on all participants to combine together the secret might be impractical, and therefore sometimes the threshold scheme is used where any k of the parts are sufficient to reconstruct the original secret.

MATHEMATICAL DEFINITION:

The goal is to divide secret S (e.g., a safe combination) into n pieces of data $S_1, \cdots, S_n$ in such a way that:

1. Knowledge of any k or more S_i pieces makes S easily computable.
2. Knowledge of any $k-1$ or fewer S_i pieces leaves S completely undetermined (in the sense that all its possible values are equally likely).

This scheme is called (k, n) threshold scheme. If $k = n$ then all participants are required to reconstruct the secret.

Shamir's secret-sharing scheme

The essential idea of Adi Shamir 's threshold scheme is that 2 points are sufficient to define a line, 3 points are sufficient to define a parabola, 4 points to define a cubic curve and so forth. That is, it takes k points to define a polynomial of degree $k-1$.

Suppose we want to use a (k, n) threshold scheme to share our secret S, without loss of generality assumed to be an element in a finite field F of size P where $0 < k \leq n < P; S < P$ and P is a prime number.

Choose at random $k-1$ positive integers $a_1, \cdots, a_{k-1}$ with $a_i < P$, and let $a_0 = S$. Build the polynomial $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots + a_{k-1} x^{k-1}$. Let us construct any n points out of it, for instance set $i = 1, \cdots, n$ to retrieve $(i, f(i))$. Every participant is given a point (an integer input to the polynomial, and the corresponding integer output). Given any subset of k of these pairs, we can find the coefficients of the polynomial using interpolation. The secret is the constant term a_0.

AES ALGORITHM:

The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data. AES is based on a design principle known as a substitution-permutation network, combination of both substitution and permutation, and is fast in both software and hardware. AES operates on a 4×4 column-major order matrix of bytes, termed the *state*, although some versions of Rijndael have a larger block size and have additional columns in the

state. Most AES calculations are done in a special finite-field.

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext. The number of cycles of repetition are as follows:

14 cycles of repetition for 256-bit keys.

Each round consists of several processing steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

The main feature of AES is as follows:
(i) Symmetric and parallel structure: This gives the implementers of the algorithm a lot of flexibility. It also stands up well against cryptanalysis attacks.

(ii) Adapted to modern processors: The algorithm works well with modern processors.

(iii) Suited to smart cards: The algorithm can work well with smart cards.

AES supports key lengths and plain text sizes from 128 bits to 256 bits, in the steps of 32 bits. The key length and the length of the plain text blocks need to be selected independently. Version of AES used is: 128 bit plain text block with 256 bit key block.

STEPS OF ALGORITHM:

1. Key Expansions—round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.
2. Initial Round
3. AddRoundKey—each byte of the state is combined with a block of the round key using bitwise xor.
4. Rounds
    1. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
    2. ShiftRows—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
    3. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
    4. AddRoundKey
5. Final Round (no Mix Columns)
    1. SubBytes
    2. ShiftRows
    3. AddRoundKey.

V.WORKING OF PROJECT:

ADMIN MENU

The e-commerce server side operator has the following options in the menu:

-Update Stock
-Add manufacturer
-Add Category
-Add Product
-New Orders
-Logout
Once the user logins he can firstly update the stock .The stock needs to be updated both on addition of new product or on selling of product.
There are number of manufacturer which can be added to list depending on the product.

The products can be categorized into various types such as electronics , clothing , cosmetics etc. This helps in classifying the products.

The particular product can then be added belonging to the specific manufacturer and category.

Also whatever orders are placed by the user will be updated in the new orders section. The server side operator has to provide acceptance to the order depending on the availability of the product in stock.

NEW CLIENT

I am a new client

At this stage of implementing project the user can either login using the registered email address and password. If the user is not registered he can create a new account . Once the user registers himself all the details are stored.

MY ACCOUNT INFORMATION:

The customer registers himself following up the details like personal details, company details, address details, contact information. The details will be saved for that particular customer. he can do shopping every next time just using login details.

welcome : pranav nair

When the customer successfully logins he will get a menu with following options.

-Change password
-Contact & Billing Info
-Purchase
-My Cart
-Order Status
-Add Question

In the change password option the customer can change his login password option the user can change his login password. The same would be updated in the record.

In this section the customer has to provide the contact details. This will include billing address where the invoice is to be sent. The customer can select the mode of payment at this stage which will include two options namely Credit/Debit card and cash on delivery. On clicking the continue the customer is forwarded towards confirming order.

Purchase section will contain details of the shopping by the customer in past.

My cart will contain the products which the user has chosen for purchase but the order is not confirmed. My cart can contain any number of products which customer wishes to buy.

When the order for any product is placed by the customer it needs to be confirmed by the server side operator. on acceptance of order only further procedure can be done. The status of the order can be checked in this section.

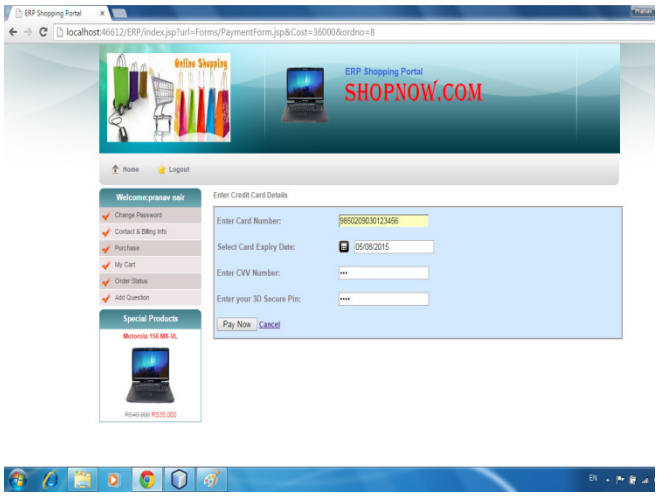
Add question is the part where customer can select a security question and provide answer to it. The same is used for generating the key in the encryption and decryption process.

CREDIT CARD DETAILS:

On entering the billing information the next part is the credit card details if the user chooses card as mode of payment. The details like card number , expiry date , CVV number , 3D secure pin , to be provided. On clicking Pay Now option the user is notified regarding the confirmation of processing.

SECURE LOGIN

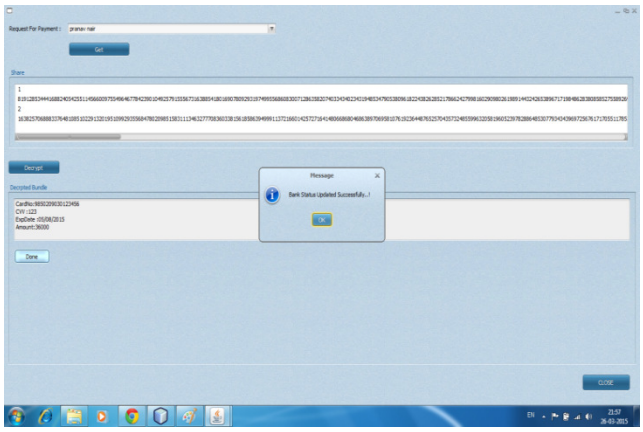
This is the bank side login page, where the card details will be made available for payment.



ENCRYPTED INFORMATION

Here the Shamir Key Algorithm is used. Once the administrator login he firstly needs to enter the name of customer for whom the payment is to be requested. When the "**GET**" button is clicked the card details and amount for payment appears in the encrypted form.

When the "**DECRYPT**" button is clicked the details are decrypted and displayed. The new step is to click DONE button on which a dialog box will be displayed that **"BANK STATUS UPDATED SUCCESSFULLY".**



VI.CONCLUSION

In this project, we have introduced a novel architecture for a cloud-based IDM. Our architecture uses a central IDM and middleware to carry out Security and Ecommerce duties. The main features in this architecture are scalability, security in access control by implementing Key Management Protocol, and extensibility to future technologies. The Key Management Shamir's Protocol introduces great flexibility in selecting authentication method by designing a common Pluggable Authentication module. It also introduces an open platform for authorization and delegation that is an essential requirement for an application-centric infrastructure. Our experiment shows that the IDM improves the throughput of as the system stands more secure. In the future, we plan to study federation to enable bursting to other cloud providers. We are also interested in including new authentication technologies such as QR-Code and mobile device into IDM.

VII.REFERENCE:

[1] Pallavi Khatri, "Using Identity and Trust with Key Management for achieving security in Ad hoc Networks" INDIA, _c 2014 IEEE

[2] Lingfang Zeng , Shibin Chen , Qingsong Wei , and Dan Feng, " SeDas: A Self-Destructing Data

System Based on Active Storage Framework", China,2013 IEEE

[3] R. Geambasu, T. Kohno, A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self destructing data," in *Proc. USENIX Security Symp.*, Montreal, Canada, Aug. 2009, pp. 299–315

[4] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[5] S. Wolchok, O. S. Hofmann, N. Heninger, E. W. Felten, J. A. Halderman, C. J. Rossbach, B. Waters, and E. Witchel, "Defeating vanish with low-cost sybil attacks against large DHEs," in *Proc. Network and Distributed System Security Symp.*, 2010.

[6] L. Zeng, Z. Shi, S. Xu, and D. Feng, "Safevanish: An improved data self-destruction for protecting data privacy," in *Proc. Second Int. Conf. Cloud Computing Technology and Science (CloudCom)*, Indianapolis, IN, USA, Dec. 2010, pp. 521–528.

[7] J. R. Douceur, "The sybil attack," in *Proc. IPTPS '01: Revised Papers From the First Int. Workshop on Peer-to-Peer Systems*, 2002.

[8]. C.E. Perklins, "Ad Hoc Networking", 1st edition. Addison –Wesley Professional, 2001.

[9]. I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," Computer Networks, Volume 47, 2005, pp. 445-487.

[10]. H. Dahshan, and J. Irvine, "A trust based threshold cryptography key management for mobile ad hoc networks," *IEEE 70th Vehicular Technology Conf.*, Anchorage, AK, USA, pp. 1-5, Sept. 2009,.

[11]. S. Capkun, L. Buttya, and J.-P. Hubaux, "Selforganized public-key management for mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52-64, Jan. – Mar., 2003.

[12]. A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," *CRYPTO'84*, 1984, pp. 47–53. [14]. Y. G. Desmedt, "Threshold cryptography," European Transactions on Telecommunications, vol. 5, no. 4, pp.

[15]. J. Huang and D. Nicol, "A calculus of trust and its application to PKI and identity management," *ACM 8th Symposium on Identity and Trust on the Internet*, Gaithersburg, MD, USA, April 2009.

[16]. N. V. Vinh, M.-K. Kim, H. Jun, and N. Q. Tung, "Group-based public-key management for self-securing large mobile ad-hoc networks," *Int'l Forum on Strategic Technology*, pp. 250-253, Oct. 2007 .

[17]. B. Poettering, 2006, SSSS: Shamir's Secret Sharing Scheme [Online]. Available: http://point-at-infinity.org/ssss/

[18]. Khatri, P., Tapaswi, S. & Verma, U.P. (2012). Trust evaluation in wireless ad hoc networks using fuzzy system. In V. Potdar & D. Mukhopadhyay (eds.), pp. 779-783, CUBE 2012.

[19]. L. Qin and D. Feng, "Active storage framework for object-based storage device," in *Proc. IEEE 20th Int. Conf. Advanced Information Networking and Applications (AINA)*, 2006.

[20]. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for storage security in cloud computing," in *Proc. IEEE INFOCOM*, 2010.