VOL. 12, NO. 11, JUNE 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

OPTIMIZED SECURE CONFIRMATIONS USING SMART CARD EVALUATION IN MULTI CLOUD STORAGE

U. Susmitha and D. Rajeswara Rao
Department of Computer Science and Engineering, K. L. University, Guntur, India
E-Mail: susmithauddaraju@gmail.com

ABSTRACT

Distributed computing contains a collection of storage space web servers, providing a fantasy of endless storage space and obtaining. Security is one of the critical elements of such a process. Saving details at a remote third party's cloud product is always causing serious concern over details privacy and survivability. Many security techniques protect details reliability, but they limit the performance of the details owner especially with respect to cancellation because one key centered protection techniques are employed for secure details. So we recommend another cryptosystem that can create resolved approximated details obtaining important factors such that a data consultation occasion requires assigning an offer of infrequent secrets of irrelevant customers as understanding rights for particular agreement of realized material. An amazing element is that one can total numerous agreements of secret important factors from individual secret solidarity and at once make them as reduced as could be allowed simply like their protector individual solidarity, yet at the same time be pushing the force of the considerable number of important factors being gathered that can remarkably invest in a client. The JSON Web Methods (JWM) detail signs up cryptographic computations are used for WEB encryption with preferable key specifications in both JSON web framework and JSON web key presentation. It characterizes a few IANA registries for these identifiers. Every one of these details uses JavaScript Object Notation (JSON) centered details components. This is used to produce similar script picture era for managing effective capacity in distributed computing.

Keywords: key-aggregate encryption, patient-controlled encryption, JSON, web trademark, JSON web encryption, JavaScript object notation.

INTRODUCTION

Cloud storage area space is becoming more popular lately. In business configurations, we see the development of requirement for details freelancing, which helps in the ideal control of corporate details. It can be used as an important and first technology behind so many intermediate web services with real time applications. Present day's implementation of free hosting for maintenance of records, memory availability based on storage with space on remote availability. Access control between clients achieves efficient in data sharing with all the users. Procedure for the generation of relevant data presentation with different cloud services as shown in following Figure-1.

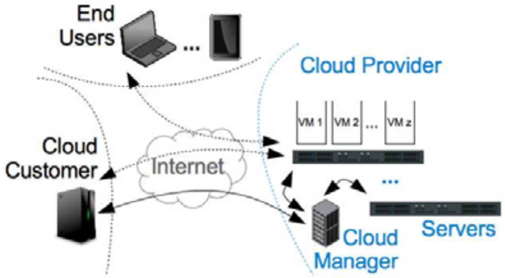

Figure-1. Cloud infrastructure to manage clients with security via virtual machines.

Based on conformational details of who have served and maintenance between server and registered users to apply understand in availability management with verification with all the verified details. In a shared-tenancy cloud handling environment, things become even more extreme. In virtual machines (VMs), different users maintain structural and individual data assurance on sharing in out sourced cloud. VM data resource management instantiating into concentration on different users with respect to operations of each client in outsourcing cloud.

For example, blog writers can let their buddy's viewpoint a part of their individual pictures; a business may allow her workers availability a part of sensitive details. The challenging problem how to efficiently discuss properly secured details. Of course, clients can obtain the properly secured details from the storage area space, decrypt them, then deliver them to others for discussing, but it falls the value of reasoning storage area environment. Because of increasing outsourced users in real time cloud environment, users directly contact with server and then access their data from server using their predefined available data from data service provider. Then an effective and properly secure approach is require for share users data with the cloud storage space, it is simple for sharing to more peoples, here we observe Drop Box as an example for simple file sharing and storage with accessible operations.

For additional privacy key components for encryption in two representations, i.e. sequential key or unequal key. For more privacy in data distribution, Alice require distinct key privacy values from other person (who is sending data to her), by observing this procedure traditional convergent encryption approach is not applicable for legitimately services with an organization to improve privacy considerations.

Therefore, the best remedy for the above problem that Alice encrypts details with unique public keys, but only delivers Bob only one (constant-size) decryption key. Since the decryption key should be sent via a properly secured route and kept the key, small key sizing is always appropriate. For example, we cannot anticipate large storage area space for decryption key elements in the resource-constraint devices like smart phones, brilliant bank cards, or wireless signal nodes.

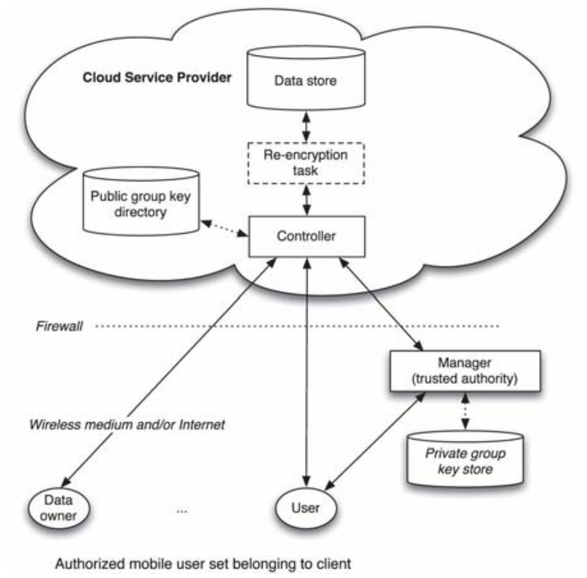
Cloud Service Provider
Data store
Re-encryption task
Public group key directory
Controller
Firewall
Manager (trusted authority)
Wireless medium and/or Internet
Private group key store
Data owner
...
User
Authorized mobile user set belonging to client

Figure-2. Cloud data storage with respect to cryptography.

Especially, these key elements are usually held in the tamper-proof storage area, which is relatively costly.

By observing the above data representation, it is a complex task to combination of different elements into common image, i.e. million data representation in outsourced data in distributed environment. Moving these secret important factors normally needs a properly secured course, and putting away these important elements needs rather luxurious assured space for storage space [6]. The expenses and muddling connected with usually improve with the mixed bag of the unscrambling important factors to be communicated. In a few words, it is extremely enormous and luxurious to do that. Protection important elements likewise go along with two preferences - Systematic key process or universal key process in data outsourcing. In systematic key approach, based on above procedure Alice need information of receiver to 3rd person, then she need to give her secret key to them; clearly, this is not reasonable process to support all the secure operations. By considering complexity of privacy data with unique verification based on key security assumptions. This procedure gives problem definition for our problem

Based on above data representation in outsourced cloud for data processing between alice and bob with privacy and access control image data sharing, we provide preferable Aggregate Key Approach for privacy based data sharing with operational events in distributed environment of out sourced data. Eventually our proposed approach gives better secure key operations to cloud outsourcing in distributed environment of data sharing. The present research activities generally focus on reducing the collaboration specifics. Our experimental results give better efficient results in access control based data sharing cloud with preferable operations.

RELATED WORK

There are different methods that announce to use the techniques of differential acquire, also known as delta stress, to boost the update of a certain details data file. The most efficient methods are described in this area.

Rsync needs a different way to differential acquire. It allows a customer to demand changes to a details data file from the server without challenging the server to maintain any old editions. The server decides the modifications on the fly. This is an obstacle since a longer time would be necessary when you compare with the LDDA. Besides, Rsync needs many of features on the customer part. Thus, it would present low performance if used by SDR devices, which are naturally limited and use a low details exchange utilization system.

The Xdelta requirements are according to the idea of preventing hand printing provided by Rsync. It also uses adler32 and MD5 checksums to generate hand marks, but different from Rsync, it needs that the server has all the available editions of the requested details data file. Thus, the modifications can be created off-line, a priori. A benefit of Xdelta is that it uses a divided development that differentiates the sequence of recommendations from the data result. The performance of Xdelta is also discouraging for limited SDR devices, since its thinking is based on the use PC extreme features used by a several of Linux system selections, such as glib and zlib.

BACKGROUND APPROACH

We first give the structure and centrality to key aggregate security. At that point we let you realize that to utilize KAC in a circumstance of its put in deduction storage room. Key-total securities arrange with 5 polynomial equations as takes after. The points of interest proprietor chooses the gathering framework parameter by means of Set up and produces an open/ace secret3 key couple through KeyGen. Based on reliable data representation in all the peoples in outsourcing data in cloud is configuration with cipher text with separate keys. The information proprietor can utilize the ace mystery to give an aggregate decoding key to an arrangement of figure composed content classes through Draw out. Outsourced data with separated keys with multiple file access control (by means of ensured messages or ensured gadgets). Finally, each client with an aggregate key can used to describe all access control files outsourcing data.

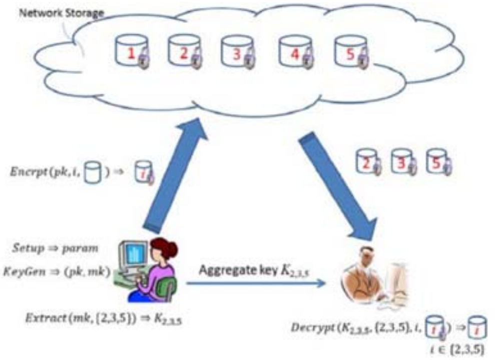
Network Storage
1
2
3
4
5
Encrpt(pk, i,) ⇒
i
2
3
5
Setup ⇒ param
KeyGen ⇒ (pk, mk)
Aggregate key $K_{2,3,5}$
Extract(mk, {2,3,5}) ⇒ $K_{2,3,5}$
Decrypt($K_{2,3,5}$, {2,3,5}, i,) ⇒
i ∈ {2,3,5}

Figure-3. Key aggregate system for out sourcing data in cloud.

Setup: Applied by the details owner to develop a for ex consideration on an untrusted server. On surveys an encryption organize parameter 1 and the scope of figure composed content sessions n (i.e., order gathering ought to be a whole number encompassed by 1 and n), it results the gathering program parameter, which is missed from the criticism of alternate strategies for curtness.

KeyGen: implemented by the details owner to arbitrarily have a public/master secret key several.

Encrypt: Connected by any individual who needs to secure points of interest. On surveys an open key pk, a gathering I indicating the figure composed content grouping, and a thought m, it results a figure composed content C

Extract; SÞ: executed by the subtle elements proprietor for allocating the decoding power for a specific arrangement of figure composed content sessions to a dole out. On surveys the ace mystery key msk and a set S of creepy crawlies relating to various sessions, it results the total key for set S signified by KS.

Decrypt; S; i; CÞ: actualized by an allocate who acquired a total key KS created by Remove. On audits KS, the set S, a gathering i indicating the figure composed content characterization the figure composed content C connected with, and C, it results the unscrambled result m in the event that i 2 S.

PROPOSED APPROACH

JSONP (or JSON with Padding) is a procedure utilized by web designers to defeat the cross-space repressions constrained by applications to let data to be recovered from structures other than the one the page was with. JSONP looks good exactly when utilized with a framework consider. For each new JSONP request, the framework must comprise of another <script> element or reusing a dynamic one [16]. The past decision-including another framework segment-is done through element DOM control, and is known as framework element implantation. The <script> figure is filled the HTML DOM, with the URL of the fancied JSONP endpoint set as the "src" highlight [17]. This effective framework considers mixture is regularly done by a JavaScript related gathering. Jquery and distinctive procedures have JSONP help limits; there are besides separate arrangements.

An type of the strongly filled system factor for a JSONP contact appears like this:

```
<script type="application/JavaScript"
src="http://server.example.com/Users/1234?callback=parseResponse"> </script>
```

After the factor is filled, the system analyzes the factor, and works a HTTP GET on the src URL, recuperating the substance.

Input: Uploaded files
Output: Aggregated Key with Embedded Script
Step-1: Users initialization with respect credentials of users in data sharing.
Step-2: Check for access control policies for uploaded data in cloud storage.
Step-3: Apply script code generation using JavaScript and HTML with JQuery.
Step-4: JSON Compression algorithms considerably reduce JSON file size.
Step-5: The compression on the server-side does make sense when the client doesn't know how to work with gripped content and it is important to keep the traffic value as low as possible (due to cost and time).
Step-6: It is important to unpack the JSON compression content on the server before consuming it.

Algorithm 1: Procedure for Compression of JSON in HTML code script.

As shown in the above procedure analyzes the appearance payload as JavaScript. The JavaScript same-source agreement typically keeps applications from delivering AJAX marketing to another area and getting a response (more current applications that enhance CORS can chill out this requirement). An overseeing intermediary server, in any case, does not have such controls and can exchange programming sales to a server in an alternate space, shop the outcome, and after that appearance that JSON payload when this technique makes the second need. The server would be advised inside of the initially need to shop the produce (POST returning JSON payload) besides into a group shop (for example me reserved or inside of a period variable), and the second need from this technique then would get the orchestrated react to the beginning inquiry.

PERFORMANCE EVALUATION

In this section, we describe the overall development procedure of our proposed approach. For developing our application effectively we are using latest technologies like JDK 1.8, Net beans 8.0 for cloud user interface construction to communicate each client with all the clients in the distributed environment for efficient data sharing, we need to use cryptographic web interface for data security. Security should consider all the favorable conditions to check whether original users or unauthorized client detection with height lighted content class label generation for generating efficient examination based on tree data representation. Table 1 shows time efficiency results of our proposed JSON (JavaScript Object Notation) approach with KAC (Key Aggregation Cryptography) to introduce depth of the key results processing with comparison to all approaches.

Table-1. Data processing with key structure with respect to time efficiency.

Privacy key with depth	JSON Time results
1	0.0425
2	0.0514
3	0.0684
4	0.0789
5	0.0892

By using PBC (Pairing Based Cryptography) in JOSN (JavaScript Object Notation) for generating bat file with preferable data representation in data sharing between users in out sourced data with access control policies in privacy oriented out sourced data. After generating and analyzing all the preferable data aspects with time efficient in key generation and maintenance for processing users in distributed environment.

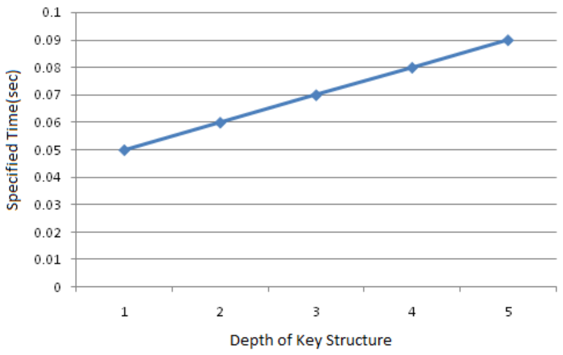

Figure-4. Aggregate key generation based on bit size and length of the structure information.

Above figure shows depth of the key structure with all required information stored in reliable data aggregation with processing of all the files with single aggregation. Performance of encryption and decryption in key maintenance and key data sharing may perform aggregate data generation in multi user access control in cloud computing.

Results w.r.t. Group Sharing: In this scenario, we compare both KAC and proposed JSON approach with respect to group sharing images to selected users with full of privacy in data sharing. JSON allows single .bat file with include all the uploaded images based on receiver user address which include embedded script representation. Table 2 shows group sharing performance values in distributed environment via users data address.

Table 2. Group sharing values in data values KAC and JSON in distributed environment.

No.of Users	KAC	JSON
5	15.52	12.65
10	13.56	11.24
15	12.54	10.24
20	11.58	9.54
25	10.84	8.65
30	9.54	7.25
40	8.52	6.14
50	6.24	5.14

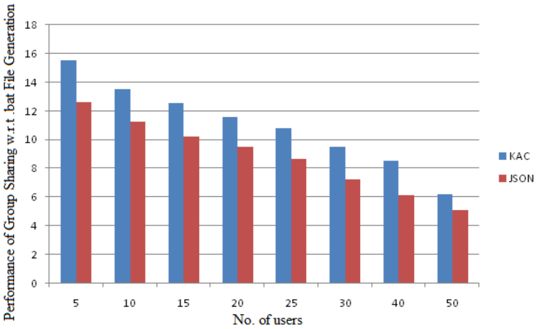

Figure-5. Group sharing performance based on increased group data sharing.

As shown in figure 5, JSON gives better group sharing and privacy representation in reliable data sharing presentation in cloud. JSON achieves excellent cryptographic data sharing perform aggregate data with .bat file security operation. Our minute results furthermore accept to what can be seen from the formula in Attract out and Decrypt - two combining capabilities take unimportant time; the important time frame of Decrypt is around a double of Attract out. Watch that our assessments handled up to 65536 combined bag of classes (which is furthermore the weight component), and ought to be completely tremendous for fine-grained information analyzing as a guide [12].

Comparison results w.r.t Granted Keys: Based on demonstrated experimental evaluation with delegation ratio in key sharing and multi user access control data representation in distributed environment. Table 3 shows efficient delegation ratios with comparison of KAC.

Delegation ration will be calculated based on number of granted keys were sharing to all the peoples present in real time distributed data storage using JSON operations.

Table-3. Delegation ratio's in KAC and JSON values in group data sharing with granted keys presented in data representations**.**

Delegation Ratio Values	Tree-Based Approach	KAC	JSON
0.1	100	110	101
0.2	200	220	102
0.3	300	250	103
0.4	400	270	104
0.5	500	290	105
0.6	600	320	106
0.7	700	370	107
0.8	800	450	108
0.9	900	500	109
0.95	1000	550	110

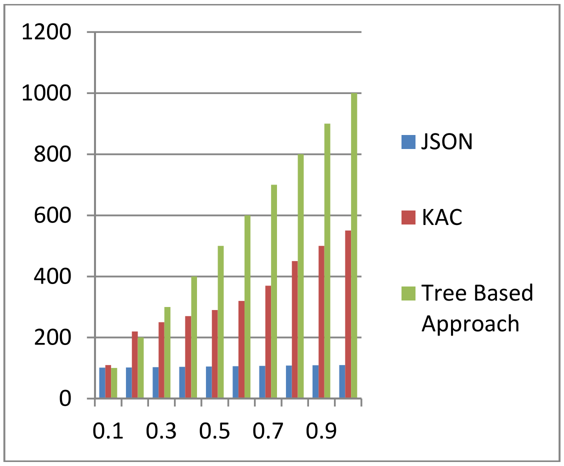

Figure-6. Number of granted keys (na) required for different approaches in the case of 1500 classes of data.

Numbering framework brands from inaccessible web servers permits the far off web servers to create any material into a site. In case the removed web servers have shortcomings that empower JavaScript implantation, the website page gave from the primary server is appeared to a broadened hazard. For the situation that an attacker can create any JavaScript into the main site page, then that program code can reestablish additional JavaScript from any zone, skipping Same-source methodology. The Material Protection Policy HTTP Headlines gives locales a chance to tell web applications which space that projects might be coordinated from. An exertion was acknowledged around 2011 to characterize a more secured tight part importance for JSONP that applications would can possibly keep up on framework prerequisites with a specific MIME sort, for instance, "application/jsonp" [14] [15]. For the situation that the reaction did not parse as tight JSONP, the framework could toss a slip-up or essentially disregard the entire reaction. Notwithstanding, this framework was deserted for CORS, and the correct MIME sort for JSONP keeps on being application/JavaScript.

Simple organizations of JSONP are capable to cross-site prerequisite assembling (CSRF or XSRF) assaults. Since the HTML <script> tag does not regard the same-root methodology in web framework usage, a dangerous site page can request and get JSON points of interest appropriate in with another site [17]. This will give the JSON-encoded points of interest to be assessed in the content of the poisonous site page, maybe uncovering security passwords or other delicate subtle elements if the client is starting at now settled into the other site.

This is a complex approach to provide smart card evaluation in reliable data efficiency with consistent privacy and the server depends on upon the same-beginning agreement of the system to prevent the conveyance of the details about account of an unapproved requirement. In summary, all the above figure shows data relevancy and privacy with JSON string operations in maintenance of our approach. Be that as it may, we put away excessive protected storage space without the anxiety of looking after a framework of task session.

CONCLUSIONS

In this paper, we consider how to "pack" enter components out in the open key cryptosystems which bolster assignment of key components for various figure composed content classes in speculation stockpiling territory room. Whatever one among the power set of classes, the relegate can simply get a total key of continuous estimating. Our technique more flexible than the asked for key process which can just safeguard places if every single key-holder discuss a comparative arrangement of benefits. Despite the fact that the parameter can be down-loadable with figure instant messages, it would be better if its measuring is a person of the vast majority of the figure composed content classes. The stress of JSON details is effective when comprehensive details components must be passed on from the web system to the server. In that keeping, it is incorrect to implement gzip pressure, on the reasons that it is incorrect for the system to know in advance whether the server increases gzip. The system must be a traditionalist, in light of the fact that the server may have modified capabilities between demands. Nowadays, we should deal with the most compressing issue: the need to always rehash key titles again and again. I will show a Coffee Program collection for compacting JSON post via normally getting a design from several contents. The collection can supply as a fall as a part of business for the workouts JSON. stringify () and JSON. parse (), aside from that it needs enhance for a reviver potential. In mix with Rison, the source resources could be huge.

REFERENCES

[1] S.S.M. Chow, Y.J. He, L.C.K. Hui, and S.-M. Yiu. 2012. SPICE - Simple Privacy-Preserving Identity-

Management for Cloud Environment- Proc. 10th Int'l Conf. Applied Cryptography and Network Security (ACNS). 7341: 526-543.

[2] L. Hardesty. 2009. Secure Computers Aren't so Secure. MIT press, www.physorg.com/news176107396.html.

[3] C. Wang, S.S.M. Chow, Q. Wang, K. Ren and W. Lou. 2013. Privacy-Preserving Public Auditing for Secure Cloud Storage. IEEE Trans. Computers. 62(2): 362-375.

[4] B. Wang, S.S.M. Chow, M. Li and H. Li. 2013. Storing Shared Data on the Cloud via Security-Mediator. Proc. IEEE 33rd Int'l Conf. Distributed Computing Systems (ICDCS).

[5] S.S.M. Chow, C.-K. Chu, X. Huang, J. Zhou and R.H. Deng. 2012. Dynamic Secure Cloud Storage with Provenance. Cryptography and Security. pp. 442-464, Springer.

[6] D. Boneh, C. Gentry, B. Lynn and H. Shacham. 2003. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '03). pp. 416-432.

[7] M.J. Atallah, M. Blanton, N. Fazio and K.B. Frikken. 2009. Dynamic and Efficient Key Management for Access Hierarchies. ACM Trans. Information and System Security. 12(3): 18: 1-18: 43.

[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter. 2009. Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records. Proc. ACM Workshop Cloud Computing Security (CCSW '09). pp. 103-114.

[9] F. Guo, Y. Mu, Z. Chen and L. Xu. 2007. Multi-Identity Single-Key Decryption without Random Oracles. Proc. Information Security and Cryptology (Inscrypt '07). 4990: 384-398.

[10] V. Goyal, O. Pandey, A. Sahai and B. Waters. 2006. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data. Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06). pp. 89-98.

[11] Cheng-Kang Chu, Sherman S. M. Chow, Wen-Guey Tzeng, Jianying Zhou and Robert H. Deng. 2014. Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage. IEEE Transactions on Parallel and Distributed Systems. 25(2).

[12] G. Ateniese, A. D. Santis, A. L. Ferrara and B. Masucci. 2012. Provably-Secure Time-Bound Hierarchical Key Assignment Schemes. J. Cryptology. 25(2): 243-270.

[13] S. S. M. Chow, Y. Dodis, Y. Rouselakis and B. Waters. 2010. Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions. in ACM Conference on Computer and Communications Security. pp. 152-161.

[14] T. Okamoto and K. Takashima. 2011. Achieving Short Ciphertexts or Short Secret-Keys for Adaptively Secure General Inner-Product Encryption. in Cryptology and Network Security (CANS '11). pp. 138-159.

[15] T. H. Yuen, S. S. M. Chow, Y. Zhang and S. M. Yiu. 2012. Identity- Based Encryption Resilient to Continual Auxiliary Leakage. in Proceedings of Advances in Cryptology - EUROCRYPT '12, ser. LNCS. 7237: 117-134.

[16] D. Boneh, R. Canetti, S. Halevi, and J. Katz. 2007. Chosen-Cipher-text Security from Identity-Based Encryption. SIAM Journal on Computing (SIAMCOMP). 36(5): 1301-1328.

[17] Q. Zhang and Y. Wang. 2004. A Centralized Key Management Scheme for Hierarchical Access Control. in Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '04). IEEE. pp. 2067-2071.

[18] J. Benaloh. 2009. Key Compression and Its Application to Digital Fingerprinting. Microsoft Research, Tech. Rep.

[19] B. Alomair and R. Poovendran. 2009. Information Theoretically Secure Encryption with Almost Free Authentication. J. UCS. 15(15): 2937-2956.

[20] D. Boneh and M. K. Franklin. 2001. Identity-Based Encryption from the Weil Pairing. in Proceedings of Advances in Cryptology - CRYPTO '01, ser. LNCS, vol. 2139. Springer. pp. 213-229.

[21] A. Sahai and B. Waters. 2005. Fuzzy Identity-Based Encryption. in Proceedings of Advances in Cryptology - EUROCRYPT '05, ser. LNCS, vol. 3494. Springer. pp. 457-473.

[22] S. S. M. Chow, Y. Dodis, Y. Rouselakis, and B. Waters, "Practical Leakage-Resilient Identity-Based Encryption from Simple Assumptions," in ACM Conference on Computer and Communications Security, 2010, pp. 152–161.

[23] F. Guo, Y. Mu, and Z. Chen, "Identity-Based Encryption: How to Decrypt Multiple Ciphertexts Using a Single Decryption Key," in Proceedings of Pairing-Based Cryptography (Pairing '07), ser. LNCS, vol. 4575. Springer, 2007, pp. 392–406.

[24] M. Chase and S. S. M. Chow, "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption," in ACM Conference on Computer and Communications Security, 2009, pp. 121–130.

[25] T. Okamoto and K. Takashima, "Achieving Short Ciphertexts or Short Secret-Keys for Adaptively Secure General Inner-Product Encryption," in Cryptology and Network Security (CANS '11), 2011, pp. 138–159.

[26] R. Canetti and S. Hohenberger, "Chosen-Ciphertext Secure Proxy Re-Encryption," in Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07). ACM, 2007, pp. 185–194.

[27] C.-K. Chu and W.-G. Tzeng, "Identity-Based Proxy Re-encryption without Random Oracles," in Information Security Conference (ISC '07), ser. LNCS, vol. 4779. Springer, 2007, pp. 189–202.

[28] C.-K. Chu, J. Weng, S. S. M. Chow, J. Zhou, and R. H. Deng, "Conditional Proxy Broadcast Re-Encryption," in Australasian Conference on Information Security and Privacy (ACISP '09), ser. LNCS, vol. 5594. Springer, 2009, pp. 327–342.

[29] S. S. M. Chow, J. Weng, Y. Yang, and R. H. Deng, "Efficient Unidirectional Proxy Re-Encryption," in Progress in Cryptology - AFRICACRYPT 2010, ser. LNCS, vol. 6055. Springer, 2010, pp. 316–332.

[30] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Transactions on Information and System Security (TISSEC), vol. 9, no. 1, pp. 1–30, 2006.

[31] D. Boneh, C. Gentry, and B. Waters, "Collusion Resistant Broadcast Encryption with Short Ciphertexts and Private Keys," in Proceedings of Advances in Cryptology - CRYPTO '05, ser. LNCS, vol. 3621. Springer, 2005, pp. 258–275.

[32] L. B. Oliveira, D. Aranha, E. Morais, F. Daguano, J. Lopez, and R. Dahab, "TinyTate: Computing the Tate Pairing in ResourceConstrained Sensor Nodes," in Proceedings of 6th IEEE International Symposium on Network Computing and Applications (NCA '07). IEEE, 2007, pp. 318–323.

[33] D. Naor, M. Naor, and J. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers," in Proceedings of Advances in Cryptology - CRYPTO '01, ser. LNCS. Springer, 2001, pp. 41–62.

[34] T. H. Yuen, S. S. M. Chow, Y. Zhang, and S. M. Yiu, "IdentityBased Encryption Resilient to Continual Auxiliary Leakage," in Proceedings of Advances in Cryptology - EUROCRYPT '12, ser. LNCS, vol. 7237, 2012, pp. 117–134.

[35] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," in Proceedings of Advances in Cryptology - EUROCRYPT '05, ser. LNCS, vol. 3494. Springer, 2005, pp. 440–456.

[36] D. Boneh, R. Canetti, S. Halevi, and J. Katz, "Chosen-Ciphertext Security from Identity-Based Encryption," SIAM Journal on Computing (SIAMCOMP), vol. 36, no. 5, pp. 1301–1328, 2007.