An Efficient Privacy-Preserving Ranked Keyword Search Method

A.M.Anil, N.Srinivas, V. Sridhar Reddy

sai.meenugaanil@gmail.com, Srinivas.bhaskar3@gmail.com, vsridharreddy@vbithyd.ac.in

[1] PG Scholar, Dept of CSE, VBIT College of engineering, Aushapur (v), Ghatkasar (m), Medchal Dist, Telangana, India,

[2] Associate Professor, Dept of CSE, VBIT College of engineering, Aushapur (v), Ghatkasar (m), Medchal Dist, Telangana, India,

[3]Associate Professor, Dept of CSE, VBIT College of engineering, Aushapur (v), Ghatkasar (m), Medchal Dist, Telangana, India,

ABSTRACT— *Cloud computing economically enables the paradigm of data service outsourcing. However, to protect data privacy, sensitive cloud data have to be encrypted before outsourced to the commercial public cloud, which makes effective data utilization service a very challenging task. Although traditional searchable encryption techniques allow users to securely search over encrypted data through keywords, they support only Boolean search and are not yet sufficient to meet the effective data utilization need that is inherently demanded by large number of users and huge amount of data files in cloud*

In this paper, a hierarchical clustering method is proposed to support more search semantics and also to meet the demand for fast ciphertext search within a big data environment. The proposed hierarchical approach clusters the documents based on the minimum relevance threshold, and then partitions the resulting clusters into sub-clusters until the constraint on the maximum size of cluster is reached. Ranked search greatly enhances system usability by enabling search result relevance ranking instead of sending undifferentiated results, and further ensures the file retrieval accuracy.

1. INTRODUCTION

In the late 1960's the idea of "Utility computing" that was coined by MIT computer scientist and Turing award winner John McCarthy was preferably known as the concept of cloud computing over a network. Industries were looking for some sort of major solution, since utility computing ended up becoming something of a big business for companies such as IBM. Indeed, Martin Greenberger pointed out the concept that "advanced arithmetical machines of the future" were now being used not only institutionally for scientific calculation and research but also for business functions such as accounting and inventory. Further, he anticipated his piece of work in which computers would be universal almost like the major power companies running wires everywhere in due time. As the technology enhances, the question was immediately raised whether "Information utility" would become a regulate like the powe r industry or be a private entity in and of itself. Later on IBM saw the potential for enormous profit to be made in this type of business and took into consideration by providing computing services to companies for top dollar. The technical limitations on bandwidth as well as disk sp ace were a huge constraint on what could have been developed. The paradigm for this type of knowledge was simply not in place to evaluate yet for cloud computation to take into consideration, though the use of mainframe processing still proved to be profitable for quite some time. The companies such as Sun Microsystems began outing the concept in the market that "the network is the computer" successfully. Further, Larry Ellison implemented an idea (who invest ed in Salesforce.com) that had for terminal machines that would cost less than $300. These ideas were really appr eciable accordingly, but they never implemented as consumers were looking for more complete personal computer solutions that can be affordable, like some storage device that are available. Within a changeover in past decade in indexing the internet that had given rise to first Yahoo, and then Google, has shown us how working in to a vast network area of knowledge was likely the ancestor of the interactivity that we can enjoy today with cloud computing. Indexing an internet may be thought as of fun, but these search engines were composing of vast amount of information over network of server present around the world. Due to a revolutionary change in the field of industries over past decade, there has been increase in demand of outsourcing of data over a wide range of network. In order to manipulate this huge amount of data in cost effective manner enterprise has adapted a prevalent technology called cloud computing that remove the burden of data management. In this data driven environment enterprise tend to store their data onto cloud that compromise of valuable asset of customer data like emails, personal health data etc. Cloud computing is turning out to be most essential paradigm in the development of information technology which offer flex ible access , ubiquitous, on demand access and capital expenditure saving . Despite its technical advantage in business, enterprise should always keep concern of its privacy from the prying eyes over a network. Privacy preserving is one of the major hurdles in cloud for user, especi ally when the user data that reside in local storage is ou tsourced and computed onto cloud. The sensitive data that a cloud service provider is holding could be secure by firewalls ,intrusion detection system also CSP has full contro l over the infrastructure of cloud including lower level of system stack and system hardware. Although mitigate concern are taken still privacy breaches is likely to occur in the paradigm. In few cases the service provider is not fully trusted, but still we need the service. Therefore, some method should be empowered to protect the user data and user queries from unauthorized person in the cloud environment. Thus, before sending data onto the cloud, data must be encrypted to protect from data privacy and unsolicited access.

Cloud storage promises high data availability, easy access to data, and reduced infrastructure costs by storing data with remote third-party providers. But availability is often not enough, as clients need privacy guaran- tees for many kinds of sensitive data that is outsourced to un trusted providers. For example, privacy is clearly important for medical data, enterprise data and secret government documents. The standard approach to achieving privacy in storage systems is to encrypt data using symmetric encryption. Storage systems based on this approach provide end-to-end privacy in the sense that data is pro- tected as soon as it leaves the client's possession. While such a solution provides strong security guarantees, it induces a high cost in terms of functionality and is therefore inadequate for storage systems that handle data at large scales. This is because after the data leaves the client's machine in encrypted form, the server cannot perform any meaningful computation on it. To address this, one can either use general-purpose solutions (e.g., fully-homomorphic encryption or oblivious RAMs) or special-purpose solutions (e.g., searchable encryption). Although general-purpose solutions have advantages, including generality and stronger security properties, they are mostly of theoret- ical interest (e.g., recent work has shown that ORAM can be relatively practical). On the other hand, special-purpose solutions like searchable encryption are practical and aim to provide a reasonable trade-off between efficiency, functionality and security.

Searching on Encrypted Data

A Search in Encrypted Data (SED) scheme allows third-party server(s) to search on behalf of a client without the need to recover the plaintext data while pre- venting the server(s) from learning any plaintext informat ion . SED has become a very active research area in cryptography in recent years. Two seminal SED schemes are the one by Song, Wagner, and Perrig and the on e by Boneh et al.. The scheme from allows a client to encrypt its database and store the encrypted database at a remote server. Later on, the clie nt can instruct the server to search in the encrypted database and return the relevant data. The scheme from is often referred to as PEKS, namely public key encryption with keyword search. With a PEKS scheme, a client publishes h is public key so that any entity can encrypt messages for him. Later on, the client can allow a third-party server to search in the encrypted messages by as signing a token to it. Following a lot of variants have been proposed to extend the concepts in many aspects. For instance, Yang et al. proposed the concept of public key encryption supporting equality test . In contrast to the scheme from the scheme from allows a third-party server to search on the ciphertexts which are encrypted with public keys from multiple different clients. With the wide adoption of cloud computing applications, SED schemes have been regarded by many to be an important technology in securing outsourcing databases while preserving data utility and confidentiality

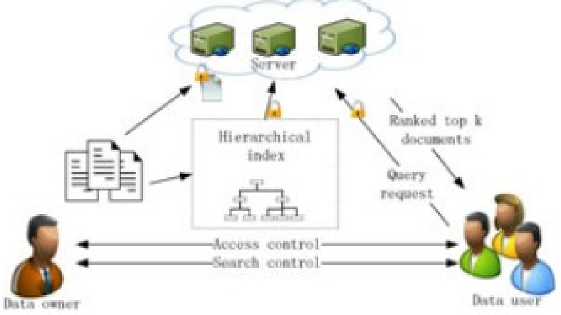


Fig:1 Architecture of ciphertext search.

Single Keyword Searchable Encryption

A single keyword searchable encryption schemes usually build an encrypted searchable index such that its content is hidden to the server unless it is given appropriate trapdoors generated via secret key(s). Our early work solves secure ranked keyword search which utilizes keyword frequency to rank results instead of returning undifferentiated results. However, it only supports single keyword search. Where anyone with public key can write to the data stored on server but only authorized users with private key can search. Public key solutions are usually very computationally expensive however.

Multiple Keyword Searchable Encryption

a cryptographic scheme that allows a client to provide a single search token to the server, but still allows the server to search for that token's word in documents encrypted with different keys . We call such a scheme multi-key search . Intuitively, the scheme hides the content of the document and the words one searches for, and the only information the server learns is whether some word being searched for matches a word in a document. We formalize the security guarantees with cryptographic security definitions and prove the security of our scheme under variants of the Bilinear Decisional Diffie-Hellman and External Diffie-Hellman assumptions, as well as in the random oracle model. The scheme is practical and was designed to be included in a new system for protecting data confidentiality against attacks on the server. The most challenging aspect when coming up with such a scheme is that there is no single trusted user; for example, in many web applications, anyone, including an adversary, can create an account and become a user. As a result, users cannot agree on a secret, and each document must be encrypted under different keys that are generated independently, rather than generated from a common secret key. Another challenge is that the scheme must be practical because our goal is to use it in a real system.

Clustering Algorithm

Clustering is an important application area for many fields including data mining, statistical data analysis, compression, vector quantization, and other business applications. Clustering has been formulated in various ways in the machine learning, pattern recognition, optimization and statistics literature. The fundamental clustering problem is grouping together (clustering) similar data items. During the search process, the user has always desired to input multiple related keywords of his interest rather than a single keyword. Basically any document deal with single concept in brief and the interrelated sub-topics. Grouping the related topics together and forming cluster helps customers to get the desired document of their interest. The most general approach is to view clustering as a density estimation problem. We assume that in addition to the observed variables for each data item, there is a hidden, unobserved variable indicating the "cluster membership". The data are assumed to arrive from a mixture model with hidden cluster identifiers. In general, a mixture model M having K clusters Ci, i=1,...,K, assigns a probability to a data point x: where Wi are the mixture weights. The problem is estimating the parameters of the individual Ci, assuming that the number of clusters K is known. The clustering optimization problem is that of finding parameters of the individual Ci which maximize the likelihood of the database given the mixture model. For general assumptions about the distributions for each of the K clusters, the EM algorithm is a popular technique for estimating the parameters.

2 RELATED WORK

D. X. D. Song, D. Wagner, and A. Perrig explained It is desirable to store data on data storage servers such as mail servers and file servers in encrypted form to reduce security and privacy risks. But this usually implies that one has to sacrifice functionality for security. For example, if a client wishes to retrieve only documents containing certain words, it was not previously known how to let the data storage server perform the search and answer the query, without loss of data confidentiality. We describe our cryptographic schemes for the problem of searching on encrypted data and provide proofs of security for the resulting crypto systems. Our techniques have a number of crucial advantages. They are provably secure: they provide provable secrecy for encryption, in the sense that the un trusted server cannot learn anything about the plaintext when only given the ciphertext; they provide query isolation for searches, meaning that the un trusted server cannot learn anything more about the plaintext than the search result; they provide controlled searching, so that the un trusted server cannot search for an arbitrary word without the user's authorization; they also support hidden queries, so that the user may ask the un trusted server to search for a secret word without revealing the word to the server. The algorithms presented are simple, fast (for a document of length n, the encryption and search algorithms only need O(n) stream cipher and block cipher operations), and introduce almost no space and communication overhead, and hence are practical to use today.

We have described new techniques for remote searching on encrypted data using an un trusted server and provided proofs of security for the resulting crypto systems. Our techniques have a number of crucial advantages: the y are provably secure; the y support controlled and hidden search and query isolation; the y are simple and fast (More specifically , for a document of length , the encryption and search algorithms only need stream cipher and block cipher operations); and the y introduce almost no space and communication overhead. Our scheme is also very flexible, and it can easily be extended to support more advanced search queries. We conclude that this pro vides a powerful new building block for the construction of secure services in the un trusted infrastructure.

D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano study the problem of searching on data that is encrypted using a public key system. Consider user Bob who sends email to user Alice encrypted under Alice's public key. An email gateway wants to test whether the email contains the keyword \urgen t" so that it could route the email accordingly . Alice, on the other hand does not wish to give the gateway the ability to decrypt all her messages. We de ne and construct a mechanism that enables Alice to provide a key to the gateway that enables the gateway to test whether the word \urgent" is a keyword in the email without learning anything else about the email. We refer to this mechanism as Public Key Encryption with keyword Search . As another example, consider a mail server that stores various messages publicly encrypted for Alice by others. Using our mechanism Alice can send the mail server a key that will enable the server to identify all messages containing some specic keyword, but learn nothing else. We de ne the concept of public key encryption with keyword search and give several constructions.

the concept of a public key encryption with keyword search(PEKS) and gave two constructions. Constructing a PEKS is related to Identity Based Encryption (IBE), though PEKS seems to be harder to construct. We showed that PEKS implies Identity Based Encryption, but the con verse is currently an open problem. Our constructions for PEKS are based on recent IBE constructions. They are able to prove security by exploiting extra properties of these schemes.

Y. C. Chang and M. Mitzenmache consider the following problem: a user U wants to store his files in an encrypted form on a remote file server S . Later the user U wants to efficiently retrieve some of the encrypted files containing (or indexed by) specific keywords, keeping the keywords themselves secret and not jeopardizing the security of the remotely stored files. For example, a user may want to store old e-mail messages encrypted on a server managed by Yahoo or another large vendor, and later retrieve certain messages while travelling with a mobile device. In this paper, we offer solutions for this problem under well-defined security requirements. Our schemes are efficient in the sense that no public-key cryptosystem is involved. Indeed, our approach is independent of the encryption method chosen for the remote files. They are also incremental, in that U can submit new files which are secure against previous queries but still searchable against future queries.

R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky studded Searchable symmetric encryption (SSE) allows a party to outsource the storage of his data to another party in a private manner, while maintaining the ability to selectively search over it. This problem has been the focus of active research and several security definitions and constructions have been proposed. In this paper we begin by reviewing existing notions of security and propose new and stronger security definitions. We then present two constructions that we show secure under our new definitions. Interestingly, in addition to satisfying stronger security guarantees, our constructions are more efficient than all previous constructions. Further, prior work on SSE only considered the setting where only the owner of the data is capable of submitting search queries. We consider the natural extension where an arbitrary group of parties other than the owner can submit search queries. We formally define SSE in this multi-user setting, and present an efficient construction.

3 OUR CONTRIBUTION

The problem of maintaining the close relationship between different plain documents over an encrypted domain has been investigate and propose a clustering method to solve this problem. We design a search strategy to enhance the rank privacy. This search strategy adopts the backtracking algorithm upon the above clustering method, by using backtracking algorithm we solve any query where it occurs. By applying the Merkle hash tree and cryptographic signature to authenticated tree structure, we provide a verification mechanism to assure the correctness and completeness of search results.

4 DEFINITIONS AND BACKGROUND

4.1 Threat Model

The adverse ability is concluded in two threat models. Known cipher text model: In this model, Cloud server can get encrypted document collection, encrypted query keywords and encrypted data index. Known background model: In this model, cloud server knows more information than that in known cipher text model. Statistical background information of dataset, such as the document frequency and term frequency information of a specific keyword, can be used by the cloud server to launch a statistical attack to infer or identify specific keyword in the query which further reveals the plain - text content of documents.

4.2 Design Goals

Search efficiency. The time complexity of search time of the MRSE - HCI is less where scheme needs to be logarithmic against the size of data collection in order to deal with the explosive growth of document size in big data scenario. Retrieval accuracy. Retrieval precision is related to two factors: the relevance between the query and the documents in result set, and the relevance of documents in the result set. Integrity of the search result the correctness, completeness and freshness of the document should be maintain

5 SYSTEMARCHITECTUREAND ALGORITHM

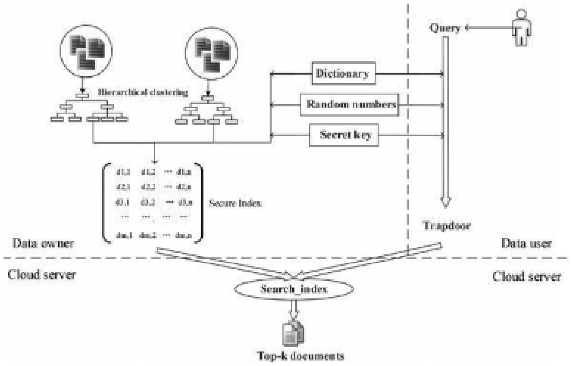


Fig. 2. MRSE-HCI architecture

The MRSE - HCI architecture is depicted by Fig.2.where the data owner is responsible for collecting documents, building document index and outsourcing them in an encrypted format to the cloud server. The cloud server provides a huge storage space, and the computation resources needed by cipher text search. Upon receiving a legal request from the data user, the cloud server searches the encrypted index, and sends back top - k documents that are most likely to match users query. The number k is properly chosen by the data user. Our system aims at protecting data from leaking information to the cloud server while improving the efficiency of cipher text search. In this model, both the data owner and the data user are trusted, while the cloud server is semi - trusted, which is consistent with the architecture in In other words, the cloud server will strictly follow the predicated order and try to get more information about the data and the index. The data user needs to get the authorization from the data owner before accessing to the data. Until now many hierarchical clustering methods have been proposed but all this method are not comparable to the partition clustering method , K - mean and K - Medoid are popular clustering algorithms but the size of k is fixed here. So we proposed a quality hierarchical clustering algorithm based on dynamic k - means. Algorithmic Dynamic k - means 1, input the initial set of k cluster C 2, set the threshold TH min 3, while k s not stable 4 , generate a new set of cluster center C 0 by k - means 5, for every cluster center C 0, i 6, get the minimum relevance score: min(S i) 7, if the min(S i)<TH min 8, add new cluster center: k=k+1 9, go to while 10, Until k is steady Every cluster is checked whether its size exceeds TH or not. If the size exceeds the cluster will split into child cluster which is formed by dynamic k - means this procedure will be repeated until all the cluster meet the requirement of maximum cluster size.]

```
Algorithm Quality Hierarchical Clustering (QHC)
1, input document and set the size threshold TH
2, build cluster ser C 0 in first level by dynamic k - means
3, while there new cluster set C i
4, for every cluster C i,j
5, if the size C i,j is bigger than TH
6, split this cluster into sub - cluster C i+1
7, until all clusters match the size constraint
```

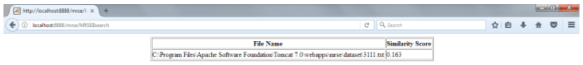
The retrieved document has possibility to be wrong because of unstable network and the data may damage due to hardware or software failure, so verifying the authenticity of search result is critical issue in cloud environment. Therefore, the minimum hash sub tree is designed to verify the correctness and freshness of search result

```
Algorithmbuildingminimumhashsubtree(MHST)

1,build hast tree based on hierarchical clustering result
2,for every leaf node I,
3,calculate its hash value:
4,while not tree root
5,for every non leaf node j,
6,calculate its hash value
7,construct node (idj)
8,goto the upper level
9,calculate tree root's hash value:
10,calculate the signature of hash value
```

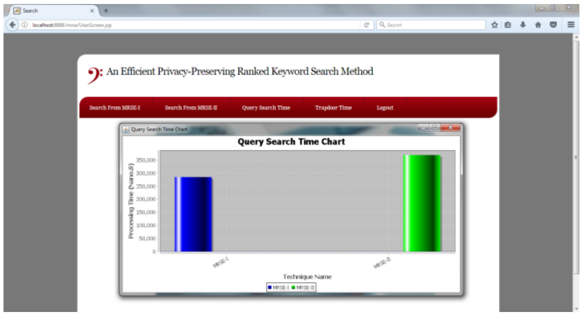
6. EXPERIMENTAL RESULTS

Query result:

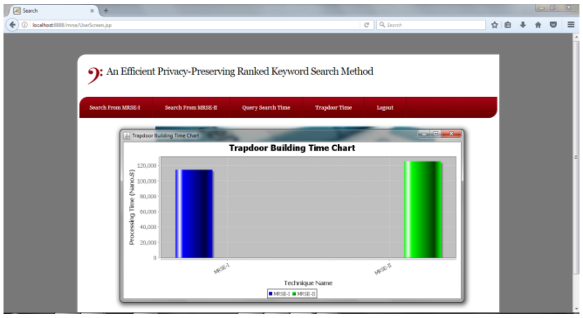


File Name	Similarity Score
C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\mrse\dataset\3111.txt	0.163

Query search time comparison chart for MRSE I and MRSE II :



Trapdoor time:



7. CONCLUSION

Theproblemofmaintainingthesemanticrelationshipbetweendifferentplaindocumentshasbeenexploredandgiventhedesignmethodtoenhancetheperformanceofthesemanticsearch.Fortheadaptationtotherequirementsofdataexplosion,onlineinformationretrievalandsemanticsearch,MRSEHCIarchitecturehasbeenproposed.Accordingly,averifiablemechanismisproposedfortheguaranteeofcorrectnessandcompletenessofsearchresults.Inaddition,thesearchefficiencyandsecurityundertwopopularthreatmodelshasbeenanalyzed.Anexperimentalplatformis builttoevaluatethesearchefficiency,accuracy,andranksecurity.Theproposedarchitecturenotonlyproperlysolvesthemultikeywordrankedsearchproblem,butalsobringsanimprovementinsearchefficiency,ranksecurity,andtherelevancebetweenretrieveddocuments.

8 REFERENCES:

[1] S. Grzonkowski, P. M. Corcoran, and T. Coughlin, "Security analysis of authentication protocols for next-generation mobile and CE cloud services," in Proc. IEEE Int. Conf. Consumer Electron.,2011, Berlin, Germany, 2011, pp. 83–87.

[2] D. X. D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Proc. IEEE Symp. Security Priv., BERKELEY, CA, 2000, pp. 44–55.

[3] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. EUROCRYPT, Interlaken, SWITZERLAND, 2004, pp. 506–522.

[4] Y. C. Chang and M. Mitzenmacher, "Privacy preserving keyword searches on remote encrypted data," in Proc. 3rd Int. Conf. Applied Cryptography Netw. Security, New York, NY, 2005, pp. 442–455.

[5] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proc. 13th ACM Conf. Comput. Commun. Security, Alexandria, Virginia, 2006, pp. 79–88.

[6] M. Bellare, A. Boldyreva, and A. O'Neill, "Deterministic and efficiently searchable encryption," in Proc. 27th Annu. Int. Cryptol. Conf. Adv. Cryptol., Santa Barbara, CA, 2007, pp. 535–552.

[7] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in Proc. 4th Conf. Theory Cryptography, Amsterdam, NETHERLANDS, 2007, pp. 535–554.

[8] E.-J. Goh, Secure Indexes, IACR Cryptology ePrint Archive, vol. 2003, pp. 216. 2003.

[9] C. Wang, N. Cao, K. Ren, and W. J. Lou, “Enabling secure and efficient ranked keyword search over outsourced cloud data,” IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 8, pp. 1467–1479, Aug. 2012.

[10] A. Swaminathan, Y. Mao, G. M. Su, H. Gou, A. Varna, S. He, M. Wu, and D. Oard, “Confidentiality-preserving rank-ordered search,” in Proc. ACM ACM Workshop Storage Security Survivability, Alexandria, VA, 2007, pp. 7–12.

[11] S. Zerr, D. Olmedilla, W. Nejdl, and W. Siberski, “Zerber+R: Topk retrieval from a confidential index,” in Proc. 12th Int. Conf. Extending Database Technol.: Adv. Database Technol., Saint Petersburg, Russia, 2009, pp. 439–449.

[12] C. Wang, N. Cao, J. Li, K. Ren, and W. J. Lou, “Secure ranked keyword search over encrypted cloud data,” in Proc. IEEE 30th Int. Conf. Distrib. Comput. Syst., Genova, ITALY, 2010, pp. 253–262.

[13] P. Golle, J. Staddon, and B. Waters, “Secure conjunctive keyword search over encrypted data,” in Proc. Proc. 2nd Int. Conf. Appl. Cryptography Netw. Security, Yellow Mt, China, 2004, pp. 31–45.

[14] L. Ballard, S. Kamara, and F. Monrose, “Achieving efficient conjunctive keyword searches over encrypted data,” in Proc. 7th Int. Conf. Inform. Commun. Security, Beijing, China, 2005, pp. 414–426.

[15] R. Brinkman, “Searching in encrypted data” in University of Twente, PhD thesis, 2007.

[16] Y. H. Hwang and P. J. Lee, “Public key encryption with conjunctive keyword search and its extension to a multi-user system,” in Proc. 1st Int. Conf. Pairing-Based Cryptography, Tokyo, JAPAN, 2007, pp. 2–22.