Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 79 (2016) 845 – 851
Procedia Computer Science

7th International Conference on Communication, Computing and Virtualization 2016

Confidentiality-Conserving Multi-Keyword Ranked Search above Encrypted Cloud Data

Amol A. Dhumal, Dr.Sanjay Jadhav

ME Student, Saraswati College of Engineering, Kharghar ,Navi Mumbai, India.
Associate Professor, Saraswati College of Engineering, Kharghar, Navi Mumbai, India.

Abstract

In cloud computing, data owners outsource their complex data management systems from local sites to the public cloud for great flexibility and cost-effective. But before outsourcing, sensitive data have to be encrypted for protecting data privacy. This obsoletes traditional data utilization based on plaintext keyword search. Thus, allowing an encrypted cloud data search service is of vital importance. As large number of data users and documents in the cloud, it is necessary to allow multiple keywords in the search request and return documents in the order of their relevance to these keywords. In literature of searchable encryption single keyword search or Boolean keyword search techniques are discussed. In this paper, we propose the problem of privacy-preserving multi-keyword ranked search over encrypted data in cloud computing (MRSE). In which the queries are transferred to the server. Server searches the relevant content by using the coordinate matching and sends the results to the user. Then the user decrypts the data.

Peer-review under responsibility of the Organizing Committee of ICCCV 2016

1. Introduction

Cloud computing is an effective approach to deal with huge amount of data, by providing on-demand high quality services from powerful and configurable computing resources. The services provided by cloud computing are divided into three categories: 1. Infrastructure as a service, 2. Platform as a service, and 3. Software as a service.

1. **Software as a service (SaaS):** The SaaS enables the customers to use CSP's applications running on the cloud infrastructure through internet. It does not provide the facility to create an application or software. The customers pay for the usage and do not own the software.[2]

Amol A. Dhumal. Tel.: +91-8108274799
*E-mail address:*amol.dhumal@rediffmail.com

Peer-review under responsibility of the Organizing Committee of ICCCV 2016
doi:10.1016/j.procs.2016.03.105

2. **Platform as a service (PaaS):** The customers need a framework which includes integrated development environment (IDE), operating systems and runtime engines that executes the application, to execute and manage their applications. These services are provided by PaaS.[2]
3. **Infrastructure as a service (IaaS)**: The IaaS refers to the hardware infrastructure provided by CSP including network, storage, memory processor and various other computing resources [2].

In one of the application of cloud computing, the data owner outsources complex database management systems into the cloud server. To protect data privacy, confidential data must be encrypted before being uploaded to the cloud server which makes it difficult to perform the traditional query processing operations. The petty solution of downloading all the data and decrypting locally is clearly impractical, due to increase in bandwidth cost. Cloud eliminates local storage management but it serves no purpose until they can be easily searched and utilized. Thus, exploring privacy preserving and effective search service over encrypted cloud data is of vital importance. Considering the large number of on-demand data users and huge amount of outsourced data documents in the cloud, this problem is particularly challenging as it is extremely difficult to meet also the requirements of performance, system usability, and scalability.

On the one hand, to meet the effective data retrieval need, the cloud server needs to perform relevance ranking on the large amount of documents instead of returning undifferentiated results. Such ranked search system enables data users to find the most relevant information quickly, rather than sorting through every match in the content collection. Ranked search can also eliminate unnecessary network traffic by sending back only the most relevant data. For privacy protection, such ranking operation, however, should not leak any keyword related information.

On the other hand, to improve the search result accuracy as well as to enhance the user searching experience, it is also necessary to support multiple keywords search for such ranking system, as single keyword search often yields far too coarse results. In todays web search engines (e.g., Google search), data users may tend to provide a set of keywords instead of only one as the indicator of their search interest to retrieve the most relevant data. And each keyword in the search request is able to help narrow down the search result further.

"Coordinate matching", i.e., as many matches as possible, is an efficient similarity measure among such multi-keyword semantics to refine the result relevance, and has been widely used in the plaintext information retrieval (IR) community. However, how to apply it in the encrypted cloud data search system remains a very challenging task because of inherent security and privacy obstacles, including various strict requirements like the data privacy, the index privacy, the keyword privacy, and many others.

2. Related Work

In the literature, searchable encryption is a helpful technique that treats encrypted data as documents and allows a user to securely search through a single keyword and retrieve documents of interest. However, direct application of these approaches to the secure large scale cloud data utilization system would not be necessarily suitable, as they are developed as crypto-primitives and cannot accommodate such high service-level requirements like system usability, user searching experience, and easy information discovery. Although some recent designs have been proposed to support Boolean keyword search as an attempt to enrich the search flexibility, they are still not adequate to provide users with acceptable result ranking functionality. Recent work provides a solution to the secure ranked search over encrypted data problem but only for queries consisting of a single keyword. How to design an efficient encrypted data search mechanism that supports multi-keyword semantics without privacy breaches still remains a challenging open problem.

In this paper, for the first time, we define and solve the problem of multi-keyword ranked search over encrypted cloud data (MRSE) while preserving strict system-wise privacy in the cloud computing paradigm. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching," i.e., as many matches as possible, to capture the relevance of data documents to the search query. Specifically, we use "inner product similarity" [6], i.e., the number of query keywords appearing in a document, to quantitatively evaluate such similarity measure of that document to the search query. During the index construction, each document is associated with a binary vector as a sub-index where each bit represents whether corresponding keyword is contained in the document. The search query is also described as a binary vector where each bit means whether corresponding keyword appears in this search request, so the similarity could be exactly measured by the inner product of the query vector with the data vector. However, directly outsourcing the data vector or the query vector will violate the index privacy or the search privacy. Our contributions are summarized as follows:

1. For the first time, we explore the problem of multi-keyword ranked search over encrypted cloud data, and establish a set of strict privacy requirements for such a secure cloud data utilization system.
2. We propose two MRSE schemes based on the similarity measure of "coordinate matching" while meeting different privacy requirements.
3. We investigate some further enhancements of our ranked search mechanism to support more search semantics and dynamic data operations.
4. Thorough analysis investigating privacy and efficiency guarantees of the proposed schemes is given, and experiments on the real-world data set further show the proposed schemes indeed introduce low overhead on computation and communication.

3. Implementation and Results

In our proposed work, Data owner outsource its own data to cloud server in encrypted form. The cloud server stores the encrypted data and performs operations on it based on the query send by the user. Data user sends a query to cloud server. When server receives this query it performs coordinate matching and send relevant results back to user. The results received by user are in encrypted form. The result is decrypted by data user after getting a key from data owner.

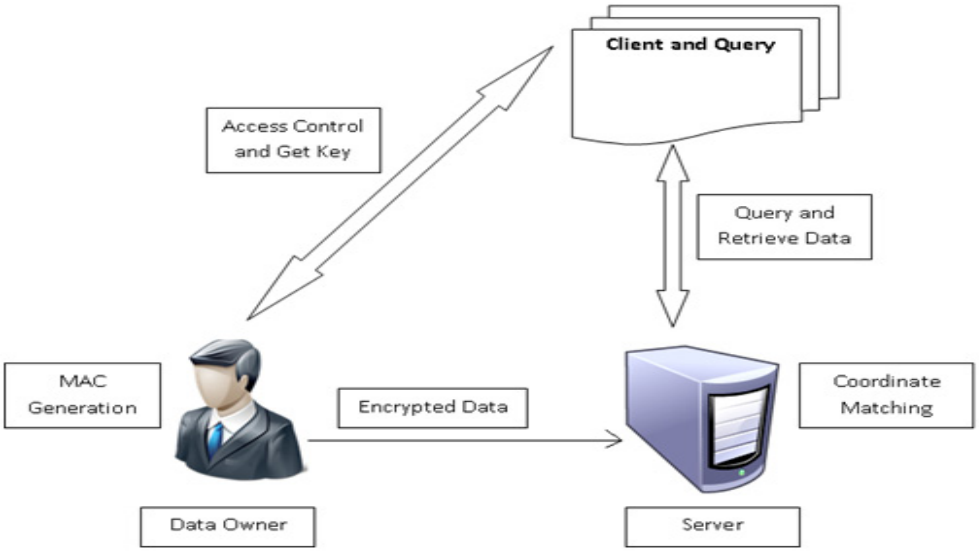

Fig 1. System Architecture

The whole procedure is divided into four modules:

- Binary data generation
- Data ciphering
- Data user access control
- Data user query

1. BINARY DATA GENERATION:

Data owner select the data and create the bit vector for that data. Using that bit vector of the data the binary data is generated. The binary data is the index for the data in the data owner. The bit vector is the bytes form of the data in the data owner. The bit vector is converted into the binary data. These bit vector and the binary data are ready for the data ciphering. Before that the message authentication code is generated for the data.

2. DATA CIPHERING:

Then the data owner have to encrypt the original data by blowfish algorihtm and send it to server. And then encrypt the binary data or the index and send it to server. Service provider did not know about the original content in the data owner. These index are used to refer the data in the service provider. It gives more security in the server side, so that the attackers can't use the data. Our system must prevent Server from learning any additional correspondence between plaintext values and ciphertext values except those obtained by prior knowledge. That is, we must protect the

plaintext values for any encrypted records or queries from being disclosed to Server.

3.DATA USER ACCESS CONTROL:

The user needs data from the server. The user have different choices and the user send the query to the server or service provider. Before that the user get the access from the data owner. For that the user send the details about him or her to the data owner. Then only the data owner receives the information from client and ready to send the decryption key. the access control mechanismis employed to manage decryption capabilities given to users This is a distributed setting where Server is on the remote side and not trusted.

DATA USER QUERY:

The data user query is processed by the service provider. The service provider generates the bit vector for the query from the client. Then the service provider converts the bit vector into binary data. Service provider finds the similar data from the index. And send the encrypted data to the client. Then the client decrypts the received data by the key from the data owner using blowfish algorithm**.** And checks the integrity of the data by using the message authentication code algorithm.

Mathematical Model for Proposed System

Input: Data Owner who have the data as book data.
Upload=(Data owner, input data, MAC)
Data owner generates the index for the data. It is binary index. Data owner upload the data and index to the server. Message authentication code is generated. It is used for the integrity checking process.

Process: Encryption and upload the data to server and query from user.
Process= (Encryption, User query)
Data owner encrypts the data by Blowfish algorithm and transfer to the server. Client send the query to server. Server searches for the relevant data based on the encrypted index and find the relevant data to the client query. And send it to the client.

Output: Retrieve relevant data from server and integrity verification process.
Output: (retrieve data, decryption, integrity checking)
Client retrieve the data and decrypts by using the blowfish algorithm. This retrieved data is relevant to the query of the client. And checks the integrity of the data by using the message authentication code.

BlowFish Algorithm

The data transformation process for PocketBrief uses the Blowfish Algorithm for Encryption and Decryption, respectively. Blowfish is a symmetric block cipher that can be effectively used for encryption and safeguarding of data. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for securing data.

The Blowfish Algorithm:

Blowfish uses a large number of subkeys. These keys must be precomputed before any data encryption or decryption. The P-array consists of 18 32-bit subkeys: P1, P2,..., P18.

Encryption

Blowfish has 16 rounds.
The input is a 64-bit data element, x.
Divide x into two 32-bit halves: xL, xR.

Then, for i = 1 to 16:
xL = xL XOR Pi
xR = F(xL) XOR xR
Swap xL and xR

After the sixteenth round, swap xL and xR again to undo the last swap.
Then, xR = xR XOR P17 and xL = xL XOR P18.

Finally, recombine xL and xR to get the ciphertext.
Decryption is exactly the same as encryption, except that P1, P2,..., P18 are used in the reverse order.

The results of our proposed system are shown below:

a b

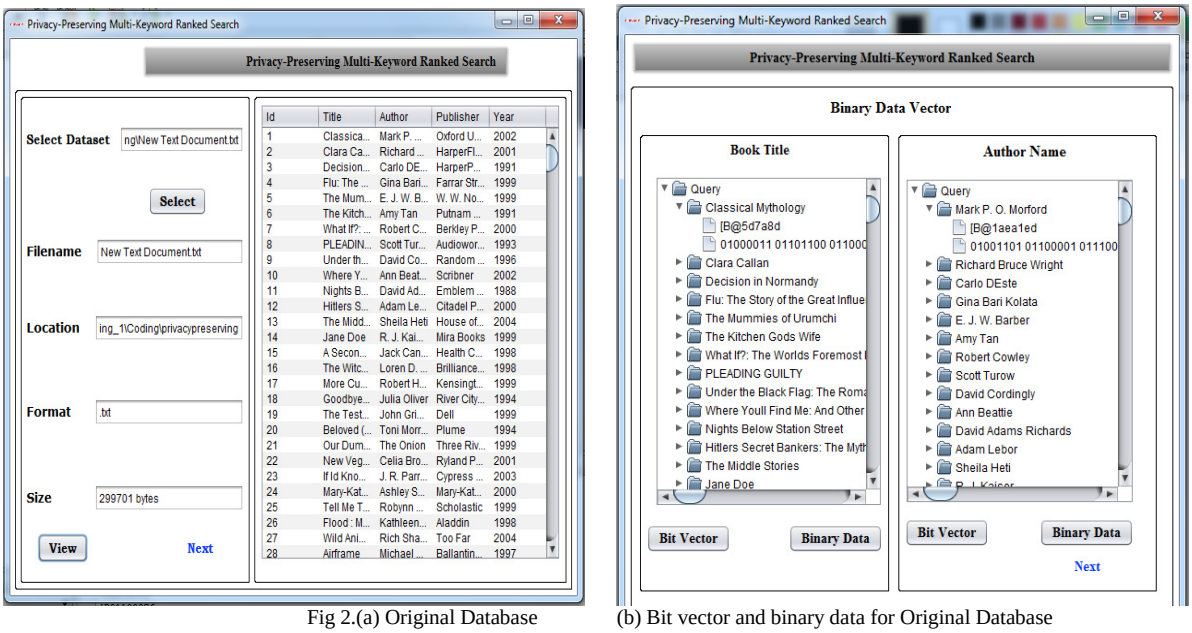

Fig 2.(a) Original Database (b) Bit vector and binary data for Original Database

a b

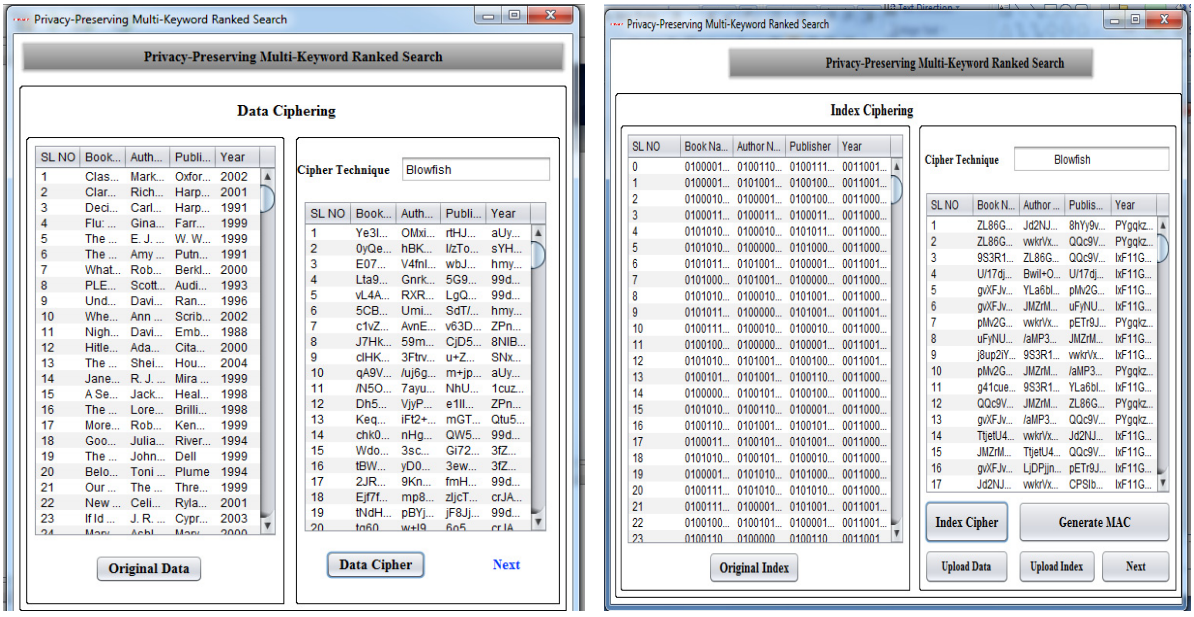

Fig. 3 (a) Data Ciphering using blowfish algorithm (b) Index Ciphering using blowfish algorithm

a b

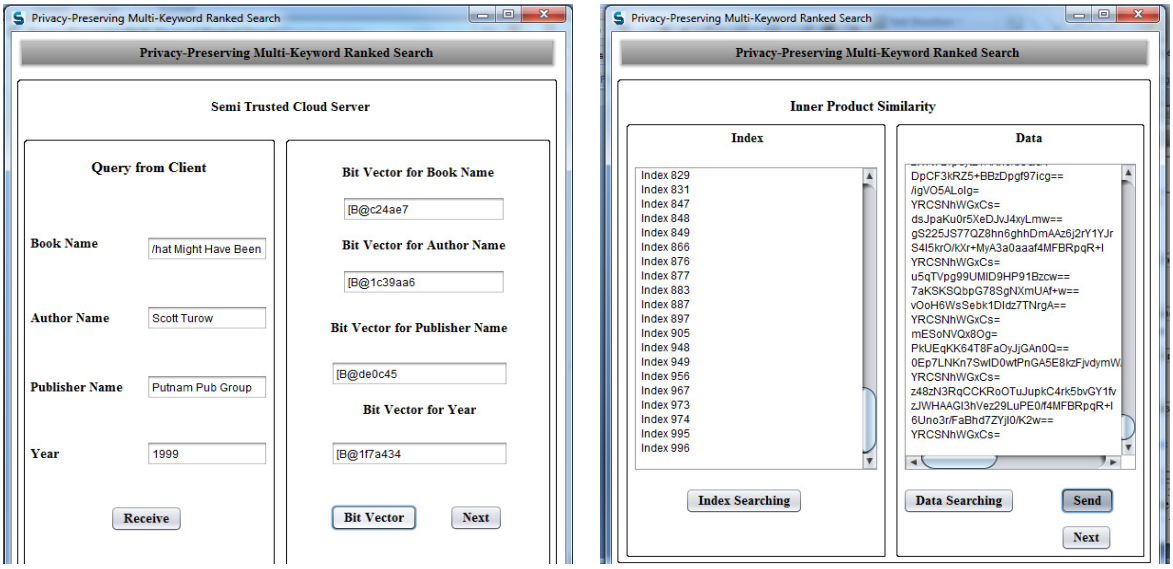

Fig. 4 (a) Multi-keyword query received by server (b) Server search for query and sends the result in encrypted form

a b

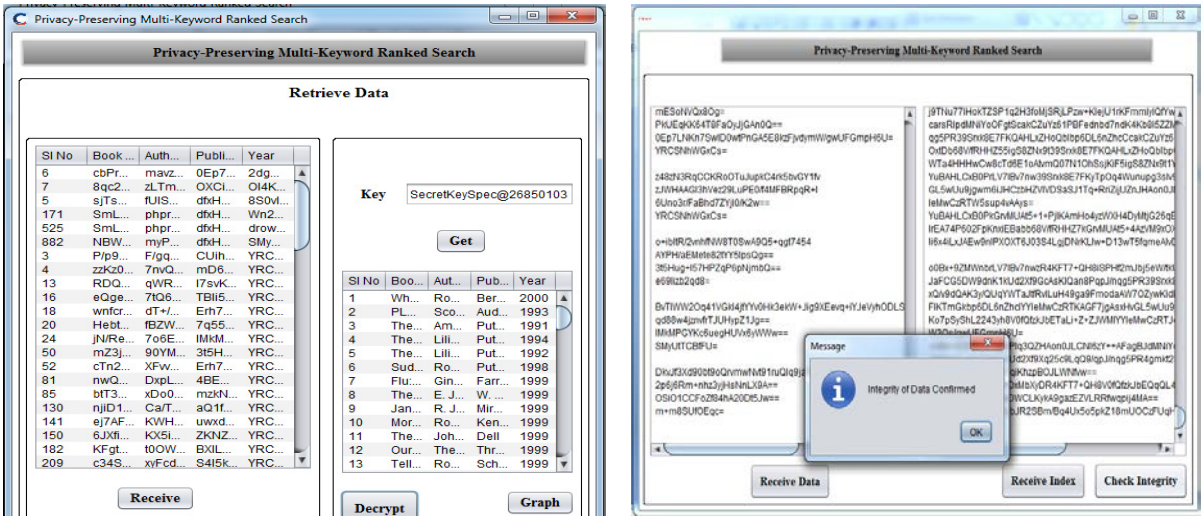

Fig. 5(a) *Client get key from privacy to decrypt the data (b) Data integrity is checked by privacy*

Conclusion

In this work, we define and solve the problem of multi-keyword ranked search over encrypted cloud data, and establish a variety of privacy requirements. The existing systems uses AES algorithm for encryption of data. Here we propose blowfish algorithm for the same. Among various multi-keyword semantics, we choose as many matches as possible; to effectively capture the relevance of outsourced documents to the query keywords, and use “inner product similarity” to quantitatively evaluate such similarity measure. It also interrogate some further improvement of our ranked search tool, contains supporting more search definition. There is a appeal to make advantage the powerful sources of the cloud server to produce services to clients.

References

1. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data," Proc. IEEE INFOCOM, pp. 829-837, Apr, 2011.
2. L.M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A Break in the Clouds: Towards a Cloud Definition," ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, pp. 50-55, 2009.
3. N. Cao, S. Yu, Z. Yang, W. Lou, and Y. Hou, "LT Codes-Based Secure and Reliable Cloud Storage Service," Proc. IEEE INFOCOM, pp. 693-701, 2012.
4. S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc. 14th Int'l Conf. Financial Cryptograpy and Data Security, Jan. 2010.
5. A. Singhal, "Modern Information Retrieval: A Brief Overview," IEEE Data Eng. Bull., vol. 24, no. 4, pp. 35-43, Mar. 2001.
6. I.H. Witten, A. Moffat, and T.C. Bell, Managing Gigabytes: Compressing and Indexing Documents and Images. Morgan Kaufmann Publishing, May 1999.
7. D. Song, D. Wagner, and A. Perrig, "Practical Techniques for Searches on Encrypted Data," Proc. IEEE Symp. Security and Privacy, 2000.
8. E.-J. Goh, "Secure Indexes," Cryptology ePrint Archive, http:// eprint.iacr.org/2003/216. 2003.
9. Y.-C. Chang and M. Mitzenmacher, "Privacy Preserving Keyword Searches on Remote Encrypted Data," Proc. Third Int'l Conf. Applied Cryptography and Network Security, 2005.
10. R. Curtmola, J.A. Garay, S. Kamara, and R. Ostrovsky, "Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions," Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06), 2006.
11. D. Boneh, G.D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), 2004.
12. M. Bellare, A. Boldyreva, and A. ONeill, "Deterministic and Efficiently Searchable Encryption," Proc. 27th Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.
13. M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi, "Searchable Encryption Revisited: Consistency Properties, Relation to Anonymous Ibe, and Extensions," J. Cryptology, vol. 21, no. 3, pp. 350-391, 2008.
14. J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy Keyword Search Over Encrypted Data in Cloud Computing," Proc. IEEE INFOCOM, Mar. 2010.
15. D. Boneh, E. Kushilevitz, R. Ostrovsky, and W.E.S. III, "Public Key Encryption That Allows PIR Queries," Proc. 27th Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO '07), 2007.