ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Identity-Based Proxy-Oriented Data Uploading and Remote Data Integrity Checking in Public Cloud

Gondesi Srikar Reddy
Student
Department of Computer Science & Systems Engineering
AU College of Engineering (A), Andhra University, Visakhapatnam.

S Jhansi Rani
Assistant Professor
Department of Computer Science & Systems Engineering
AU College of Engineering (A), Andhra University, Visakhapatnam.

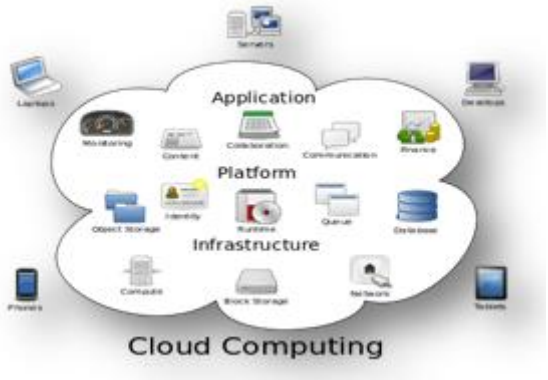
ABSTRACT:

More and more clients would like to store their data to public cloud servers (PCSs) along with the rapid development of cloud computing. New security problems have to be solved in order to help more clients process their data in public cloud. When the client is restricted to access PCS, he will delegate its proxy to process his data and upload them. On the other hand, remote data integrity checking is also an important security problem in public cloud storage. It makes the clients check whether their outsourced data are kept intact without downloading the whole data. From the security problems, we propose a novel proxy-oriented data uploading and remote data integrity checking model in identity-based public key cryptography: identity-based proxy-oriented data uploading and remote data integrity checking in public cloud (ID-PUIC). We give the formal definition, system model, and security model. Then, a concrete ID-PUIC protocol is designed using the bilinear pairings. The proposed ID-PUIC protocol is provably secure based on the hardness of computational Diffie–Hellman problem. Our ID-PUIC protocol is also efficient and flexible. Based on the original client's authorization, the proposed ID-PUIC protocol can realize private remote data integrity checking, delegated remote data integrity checking, and public remote data integrity checking.

INTRODUCTION

What is cloud computing?

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.



Structure of cloud computing

How Cloud Computing Works?

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games.The cloud computing uses networks of large groups of servers typically running low-cost consumer PC

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing.

Characteristics and Services Models:

The salient characteristics of cloud computing based on the definitions provided by the National Institute of Standards and Terminology (NIST) are outlined below:

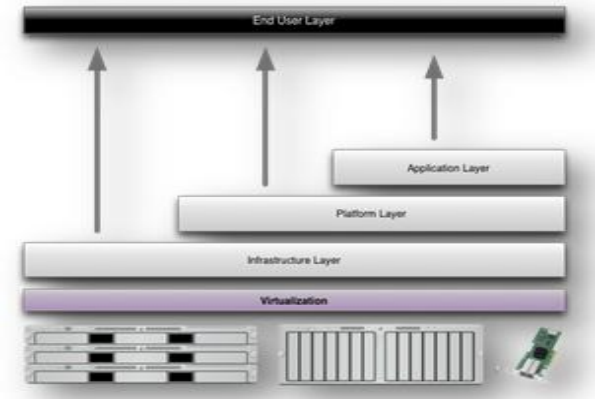
- **On-demand self-service:** A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.
- **Broad network access:** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).
- **Resource pooling:** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location-independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center). Examples of resources include storage, processing, memory, network bandwidth, and virtual machines.
- **Rapid elasticity:** Capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.
- **Measured service:** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be managed, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

Characteristics of cloud computing

Services Models:

Cloud Computing comprises three different service models, namely Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The three service models or layer are completed by an end user layer that encapsulates the end user perspective on cloud services.

The model is shown in figure below. If a cloud user accesses services on the infrastructure layer, for instance, she can run her own applications on the resources of a cloud infrastructure and remain responsible for the support, maintenance, and security of these applications herself. If she accesses a service on the application layer, these tasks are normally taken care of by the cloud service provider.



Structure of service models

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Benefits of cloud computing:

1. **Achieve economies of scale** – increase volume output or productivity with fewer people. Your cost per unit, project or product plummets.
2. **Reduce spending on technology infrastructure** Maintain easy access to your information with minimal upfront spending. Pay as you go (weekly, quarterly or yearly), based on demand.
3. **Globalize your workforce on the cheap.** People worldwide can access the cloud, provided they have an Internet connection.
4. **Streamline processes.** Get more work done in less time with less people.
5. **Reduce capital costs.** There's no need to spend big money on hardware, software or licensing fees.
6. **Improve accessibility.** You have access anytime, anywhere, making your life so much easier!
7. **Monitor projects more effectively.** Stay within budget and ahead of completion cycle times.
8. **Less personnel training is needed.** It takes fewer people to do more work on a cloud, with a minimal learning curve on hardware and software issues.
9. **Minimize licensing new software.** Stretch and grow without the need to buy expensive software licenses or programs.
10. **Improve flexibility.** You can change direction without serious "people" or "financial" issues at stake.

Advantages:

1. **Price:** Pay for only the resources used.
2. **Security**: Cloud instances are isolated in the network from other instances for improved security.
3. **Performance:** Instances can be added instantly for improved performance. Clients have access to the total resources of the Cloud's core hardware.
4. **Scalability:** Auto-deploy cloud instances when needed.
5. **Uptime:** Uses multiple servers for maximum redundancies. In case of server failure, instances can be automatically created on another server.
6. **Control:** Able to login from any location. Server snapshot and a software library lets you deploy custom instances.
7. **Traffic:** Deals with spike in traffic with quick deployment of additional instances to handle the load.

EXISTING SYSTEM:

- In public cloud environment, most clients upload their data to PCS and check their remote data's integrity by Internet. When the client is an individual manager, some practical problems will happen. If the manager is suspected of being involved into the commercial fraud, he will be taken away by the police. During the period of investigation, the manager will be restricted to access the network in order to guard against collusion. But, the manager's legal business will go on during the period of investigation. When a large of data is generated, who can help him process these data? If these data cannot be processed just in time, the manager will face the loss of economic interest. In order to prevent the case happening, the manager has to delegate the proxy to process its data, for example, his secretary. But, the manager will not hope others have the ability to perform the remote data integrity checking.
- Chen et al. proposed a proxy signature scheme and a threshold proxy signature scheme from the Weil pairing.
- By combining the proxy cryptography with encryption technique, some proxy re-encryption schemes are proposed. Liu et al. formalize and construct the attribute-based proxy signature.
- Guo et al. presented a non-interactive CPA (chosen-plaintext attack)-secure proxy re-encryption scheme, which is resistant to collusion attacks in forging re-encryption keys.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

DISADVANTAGES OF EXISTING SYSTEM:

- Public checking will incur some danger of leaking the privacy.
- Less Efficiency.
- Security level is low

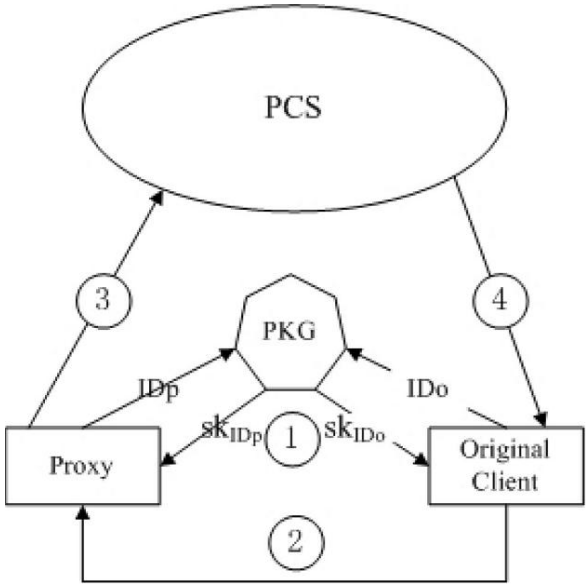
PROPOSED SYSTEM:

- This paper is based on the research results of proxy cryptography, identity-based public key cryptography and remote data integrity checking in public cloud.
- In public cloud, this paper focuses on the identity-based proxy-oriented data uploading and remote data integrity checking.
- By using identity-based public key cryptology, our proposed ID-PUIC protocol is efficient since the certificate management is eliminated. ID-PUIC is a novel proxy-oriented data uploading and remote data integrity checking model in public cloud. We give the formal system model and security model for ID-PUIC protocol. Then, based on the bilinear pairings, we designed the first concrete ID-PUIC protocol.
- In the random oracle model, our designed ID-PUIC protocol is provably secure. Based on the original client's authorization, our protocol can realize private checking, delegated checking and public checking.
- We propose an efficient ID-PUIC protocol for secure data uploading and storage service in public clouds.
- Bilinear pairings technique makes identity-based cryptography practical. Our protocol is built on the bilinear pairings. We first review the bilinear pairings.

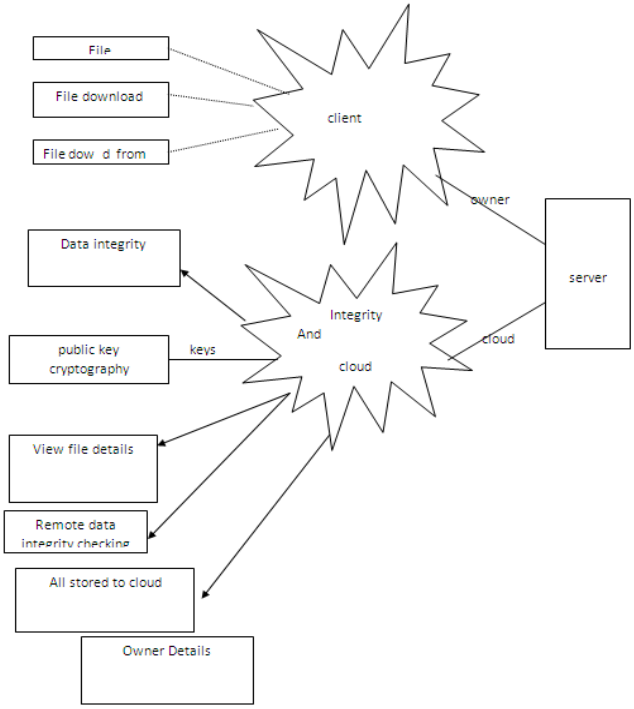
ADVANTAGES OF PROPOSED SYSTEM:

- High Efficiency.
- Improved Security.
- The concrete ID-PUIC protocol is provably secure and efficient by using the formal security proof and efficiency analysis.
- On the other hand, the proposed ID-PUIC protocol can also realize private remote data integrity checking, delegated remote data integrity checking and public remote data integrity checking based on the original client's authorization.

SYSTEM ARCHITECTURE:



BLOCK DIAGRAM:



ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

IMPLEMENTATION

MODULES:

- Achieving full anonymity
- Fully Anonymous Multi-Authority CP-ABE
- Security Model
- Security Analysis

MODULES DESCRIPTION

Achieving full anonymity

We have assumed semi-honest authorities in AnonyControl and we assumed that they will not collude with each other. This is a necessary assumption in AnonyControl because each authority is in charge of a subset of the whole attributes set, and for the attributes that it is in charge of, it knows the exact information of the key requester. If the information from all authorities is gathered altogether, the complete attribute set of the key requester is recovered and thus his identity is disclosed to the authorities. In this sense, AnonyControl is semianonymous since partial identity information (represented as some attributes) is disclosed to each authority, but we can achieve a full-anonymity and also allow the collusion of the authorities.

Fully Anonymous Multi-Authority CP-ABE

The KeyGenerate algorithm is the only part which leaks identity information to each attribute authority. Upon receiving the attribute key request with the attribute value, the attribute authority will generate H(att (i))ri and sends it to the requester where att (i) is the attribute value and ri is a random number for that attribute. The attribute value is disclosed to the authority in this step. We can introduce the above 1-out-of-n OT to prevent this leakage. We let each authority be in charge of all attributes belonging to the same category. For each attribute category c (e.g., University), suppose there are k possible attribute values (e.g., IIT, NYU, CMU ...), then one requester has at most one attribute value in one category.

Security Model

Setup→**PK**,**MK**k : This algorithm takes nothing as input except implicit inputs such as security parameters. Attributes authorities execute this algorithm to jointly compute a system-wide public parameter **PK** as well as an authority-wide public parameter yk , and to individually compute a master key **MK**k .KeyGenerate(**PK**, **MKk**, Au) → **SK**u: This algorithm enables a user to interact with every attribute authority, and obtains a private key **SK**u corresponding to the input attribute set Au. Encrypt(**PK**, M, {Tp}p∈ {0,...,r−1}) → (**CT**,**VR**): This algorithm takes as input the public key **PK**, a message M, and a set of privilege trees {Tp}p∈ {0,...,r−1}, where r is determined by the encrypter. It will encrypt the message M and returns a ciphertext **CT** and a verification set **VR** so that a user canexecute specific operation on the ciphertext if and only if his attributes satisfy the corresponding privilege tree Tp. As we defined, T0 stands for the privilege to read the file. Decrypt(**PK**, **SK**u , **CT**) → M or verification parameter: This algorithm will be used at file controlling (e.g. reading, modification, deletion). It takes as input the public key **PK**, a ciphertext **CT**, and a private key **SK**u, which has a set of attributes Au and corresponds to its holder's **GID**u.

Security Analysis

In the proposed scheme, an authority generates a set of random secret parameters and shares it with other authorities via secure channel, and is computed based on this parameters. It is believed that DDH problem is intractable in the group G0 of prime order p, therefore does not leak any statistical information about . This implies even if an adversary is able to compromise up to (N − 2) authorities, there are still two parameters kept unknown to the adversary.

SCREEN SHOTS

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

CONCLUSION

Motivated by the application needs, this paper proposes the novel security concept of ID-PUIC in public cloud. The paper formalizes ID-PUIC's system model and security model. Then, the first concrete ID-PUIC protocol is designed by using the bilinear pairings technique. The concrete ID-PUIC protocol is provably secure and efficient by using the formal security proof and efficiency analysis. On the other hand, the proposed ID-PUIC protocol can also realize private remote data integrity checking, delegated remote data integrity checking and public remote data integrity checking based on the original client's authorization.

REFERENCES:

[1] Z. Fu, X. Sun, Q. Liu, L. Zhou, and J. Shu, "Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing," IEICE Trans. Commun., vol. E98-B, no. 1, pp. 190–200, 2015.

[2] Y. Ren, J. Shen, J. Wang, J. Han, and S. Lee, "Mutual verifiable provable data auditing in public cloud storage," J. Internet Technol., vol. 16, no. 2, pp. 317–323, 2015.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[3] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in Proc. CCS, 1996, pp. 48–57.

[4] E.-J. Yoon, Y. Choi, and C. Kim, "New ID-based proxy signature scheme with message recovery," in Grid and Pervasive Computing (Lecture Notes in Computer Science), vol. 7861. Berlin, Germany: Springer- Verlag, 2013, pp. 945–951.

[5] B.-C. Chen and H.-T. Yeh, "Secure proxy signature schemes from the weil pairing," J. Supercomput., vol. 65, no. 2, pp. 496–506, 2013.

[6] X. Liu, J. Ma, J. Xiong, T. Zhang, and Q. Li, "Personal health records integrity verification using attribute based proxy signature in cloud computing," in Internet and Distributed Computing Systems (Lecture Notes in Computer Science), vol. 8223. Berlin, Germany: Springer-Verlag, 2013, pp. 238–251.

[7] H. Guo, Z. Zhang, and J. Zhang, "Proxy re-encryption with unforgeable re-encryption keys," in Cryptology and Network Security (Lecture Notes in Computer Science), vol. 8813. Berlin, Germany: Springer-Verlag, 2014, pp. 20–33.

[8] E. Kirshanova, "Proxy re-encryption from lattices," in Public-Key Cryptography (Lecture Notes in Computer Science), vol. 8383. Berlin, Germany: Springer-Verlag, 2014, pp. 77–94.

[9] P. Xu, H. Chen, D. Zou, and H. Jin, "Fine-grained and heterogeneous proxy re-encryption for secure cloud storage," Chin. Sci. Bull., vol. 59, no. 32, pp. 4201–4209, 2014.

[10] S. Ohata, Y. Kawai, T. Matsuda, G. Hanaoka, and K. Matsuura, "Re-encryption verifiability: How to detect malicious activities of a proxy in proxy re-encryption," in Proc. CT-RSA Conf., vol. 9048. 2015, pp. 410–428.

[11] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. CCS, 2007, pp. 598–609.

[12] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in Proc. SecureComm, 2008, Art. ID 9.

[13] C. C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in Proc. CCS, 2009, pp. 213–222.

[14] E. Esiner, A. Küpçü, and Ö. Özkasap, "Analysis and optimization on FlexDPDP: A practical solution for dynamic provable data possession," Intelligent Cloud Computing (Lecture Notes in Computer Science), vol. 8993. Berlin, Germany: Springer-Verlag, 2014, pp. 65–83.

[15] E. Zhou and Z. Li, "An improved remote data possession checking protocol in cloud storage," in Algorithms and Architectures for Parallel Processing (Lecture Notes in Computer Science), vol. 8631. Berlin, Germany: Springer-Verlag, 2014, pp. 611–617.

[16] H. Wang, "Proxy provable data possession in public clouds," IEEE Trans. Services Comput., vol. 6, no. 4, pp. 551–559, Oct./Dec. 2013.

[17] H. Wang, "Identity-based distributed provable data possession in multicloud storage," IEEE Trans. Services Comput., vol. 8, no. 2, pp. 328–340, Mar./Apr. 2015.

[18] H. Wang, Q. Wu, B. Qin, and J. Domingo-Ferrer, "FRR: Fair remote retrieval of outsourced private medical records in electronic health networks," J. Biomed. Inform., vol. 50, pp. 226–233, Aug. 2014.

[19] H. Wang, "Anonymous multi-receiver remote data retrieval for pay-TV in public clouds," IET Inf. Secur., vol. 9, no. 2, pp. 108–118, Mar. 2015.

[20] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proc. ASIACRYPT, vol. 5350. 2008, pp. 90–107.

[21] Q. Zheng and S. Xu, "Fair and dynamic proofs of retrievability," in Proc. CODASPY, 2011, pp. 237–248.

[22] D. Cash, A. Küpçü, and D. Wichs, "Dynamic proofs of retrievability via oblivious RAM," in Proc. EUROCRYPT, vol. 7881. 2013, pp. 279–295.

[23] J. Zhang, W. Tang, and J. Mao, "Efficient public verification proof of retrievability scheme in cloud," Cluster Comput., vol. 17, no. 4, pp. 1401–1411, 2014.

[24] J. Shen, H. Tan, J. Wang, J. Wang, and S. Lee, "A novel routing protocol providing good transmission reliability in underwater sensor networks," J. Internet Technol., vol. 16, no. 1, pp. 171–178, 2015.

[25] T. Ma et al., "Social network and tag sources based augmenting collaborative recommender system," IEICE Trans. Inf. Syst., vol. E98-D, no. 4, pp. 902–910, 2015.

[26] K. Huang, J. Liu, M. Xian, H. Wang, and S. Fu, "Enabling dynamic proof of retrievability in regenerating-coding-based cloud storage," in Proc. IEEE ICC, Jun. 2014, pp. 712–717.

[27] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in Proc. IEEE INFOCOM, Mar. 2010, pp. 1–9.

[28] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 5, pp. 847–859, May 2011.

[29] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," IEEE Trans. Services Comput., vol. 5, no. 2, pp. 220–232, Apr./Jun. 2012.

[30] Y. Zhu, G.-J. Ahn, H. Hu, S. S. Yau, H. G. An, and C.-J. Hu, "Dynamic audit services for outsourced storages in clouds," IEEE Trans. Services Comput., vol. 6, no. 2, pp. 227–238, Apr./Jun. 2013.

[31] O. Goldreich, Foundations of Cryptography: Basic Tools. Beijing, China: Publishing House of Electronics Industry, 2003, pp. 194–195.

[32] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in Proc. ASIACRYPT, vol. 2248. 2001, pp. 514–532.

[33] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Proc. CRYPTO, vol. 2139. 2001, pp. 213–229.

[34] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," IEICE Trans. Fundam. Electron., Commun. Comput. Sci., vol. E84-A, no. 5, pp. 1234–1243, 2001.

[35] C. Research. SEC 2: Recommended Elliptic Curve Domain Parameters. [Online]. Available: http://www.secg.org/SEC2-Ver-1.0.pdf, accessed 2015.

[36] The GNU Multiple Precision Arithmetic Library (GMP). [Online]. Available: http://gmplib.org/, accessed 2015.

[37] The Pairing-Based Cryptography Library (PBC). [Online]. Available: http://crypto.stanford.edu/pbc/howto.html, accessed 2015.

[38] B. Lynn, "On the implementation of pairing-based cryptosystems," Ph.D. dissertation, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 2008. [Online]. Available: http://crypto.stanford.edu/pbc/thesis.pdf