Predicate Encryption in decentralized Architecture For Security Enhancement in current DOSN

Mr.GaneshPotle,Mr.PremGaykar,Mr.ImranMujawar

Prof.Patil S.S.(Guide)

Abstract:

Centralized online social networks which we are currently using are suffers from various privacy issues like data mining from huge repository of data and distribution of data to a third party for business primitives.This is usually done by cryptographic means. In case of decentralized system the data is distributed to each user that means data storage takes place at each user's own database. In a DOSN, there are privacy-preserving variants of cryptographic primitives that do not reveal access policies. Therefore it is not suitable for usage in the decentralized architecture because of performance or storage restrictions. In a DOSN both privacy and performance are very important factors. We have used predicate encryption in this decentralized system. Aunivariate polynomial construction for access policies in PE that increases performance of the scheme but leaks some part of the access policy to users with access rights. We uses Bloom filters to decrease decryption time and indicate objects that can be decrypted by a particular user. We evaluate the performance of the decentralized scheme in the concrete scenario of a news feed.

Our decentralized architecture is best suited for a corporate professionals and organization and for small set of separate identities.

Keywords: Bloomfilters,DOSN,PredicateEncryption.

I.INTRODUCTION

Centralized networks operate with the help of central administrator who stores,manages,control and administer users data and has a sole right on whole profile data of users.So the data go through the processing of various intelligent data mining operations for advertisement purposes and exposure of data to a third party for business purposes.

In last years, decentralize networks has emerged to overcome drawbacks of centralized network.In decentralized systems many security policies have used to provide better security. New primitives like Attribute-Based Encryption (ABE) and PredicateEncryption (PE) that enable cryptographic access control have been developed in the cryptographic community. Using these concepts, access controls lists can now be constructed that do not rely on a trusted reference monitor to enforce access rules. Instead, the information is encrypted in a way that allows decryption only by parties that are eligible to decrypt them.

Generally used cryptographic primitives in DOSN's are *attributebased encryption (ABE), symmetric encryption, broadcast encryption (BE)*This method provides confidentiality of data but privacy is usually less addressed.

An access control mechanism should be such that it will preserve privacy.Decentralized network who preserves privacy should encrypt data by access policy and user satisfying this policy should only decrypt it.Size, data type and quantity of data is unknown to user unless he can decrypt it. Choosing one of the cryptographic primitive for your network does not guarantee performance and privacy of system. In previous cryptographic method (ABE), security key is associated to each attribute and encryption and decryption is possible only by using this keys.The second method BE, encrypted cipher texts are broadcasted to number of recipients.Main disadvantages regarding ABE and BE is that it encrypt data but revealing the access policy to the user.Here,policy is not hidden from users who are using this network.

Current decentralize networks provides encryption mechanism which increases the no of cipher texts in a cryptography and slows down

ISSN: 2278 – 1323

performance of the system by increasing decryption time.

In our Predicate Encryption (PE) method, cipher text size, encryption and decryption time is according to the size of the security keys used.Univariate polynomial construction in PE drastically increases performance of the system.Also access policies are hidden in private keys so that legitimate users are not aware of which data she can able to decrypt.We proposed mechanism of bloom filters in current PE scheme which notifies user about decryptable data by her so ultimately speeds up system.

Other work introducing concepts related to predicate encryption includes access policies management. In reverse to the current work, the threat model in those works does not consider *collusion* among users holding different secret keys. The problem becomes significantly more difficult when security against collusion is required. Much recent work aimed at constructing different types of fine-grained encryption schemes can be cast in the framework of predicate encryption(PE). Identity-based encryption (IBE) can be viewed as predicate encryption for the class of equality tests.Such schemes achieves some basic level of security and guarantees, informally, that a ciphertext associated with attribute *I* hides all information about the underlying message unless one holds a secret key giving the explicit ability to decrypt.

In our paper we have decentralized architecture which stores each user's data independently.Access control mechanism used in current DOSN is not dependent on storage but dependent on encryption.PE scheme used here is sets access policies on a data under encryption process.To share data to friends profile user creates personal decryption keys for each of recipient.*Keyfile* stores keys *of* each friends on the profile of the profile owner encrypted under the per-friendship
symmetric key shared by the profile owner and the friend.A digital signature has used for message authentication purposes.Each user is assumed to have a secret key which is useful for signing the messages which are available for him to decrypt. This is the whole idea of predicate encryption which implements encryption access policies in DOSN.

II.BACKGROUND AND RELATED WORK

Decentralized online social networks have been gone through much research work. We have gone through all this work to enhance our scheme of cryptography.

Safebook [3] is decentralized architecture which serves for confidentiality in both symmetric and asymmetric mechanism. Distributed Hash Table (DHT) is used to look up any data on a website but multi-hop routing is used to protect data from external observers.

Another example of decentralized network is *peerson*[4][5]follows p2p architecture and also uses DHT to look up for data on a website.But in this website security is less concerned as the public keys are stored with encrypted data.

DECENT, an architecture for OSNs that uses a distributed hash table (dht) to store user data, and enhancescryptographic primitives for confidentiality, integrity and privacy as well as support for flexible attribute policies and fast revocation. DECENT guarantee's that none of the data is visible to external observers and provides availability through replication and authentication of updates. DECENT is able to replicate the main functionality of current centralized OSNs with manageable
Overhead.

Cachet [9]is the updated version of DECENT In cachet base architecture; privacy is provided through a combination of design features including a DHT for decentralization, cryptography to enforce attribute-based policies, and data representation in terms of objects. Users can define Relationships of various types asymmetrically. User profile supports basic settings and wall features includingwall posts on social timeline,status updates, commentson posts, and a basic newsfeed in a profile.

Suscha Muller describes the concept of *policy anonymity* in cryptographic primitives. Theaccess policies are sent in clear along with the ciphertexts. To generalize the idea of policy hiding in cryptography "policy anonymity" is used where – similar to the well-understood concept of k-anonymity – the attacker is able to see the number of attributes used for encryption but could not recognize one which is used for current data encryption. A concept of graph theory is used by him to extend a known ABE construction to achieve the desired privacy property.

A. Centralized Computing

Centralized computing is computing done at a central service provider, using terminal computers attached to a central server. The

ISSN: 2278 – 1323

computer itself may control all the peripherals directly (if they are physically connected to the central computer), or it can control through intermediate terminal server. Alternatively, node terminals and central servers are connected to each other via a network link.The terminals might be textterminals or terminal clients.

In a centralized architecture central server handles all the control procedure. In accidental case, if a client computer breaks down, the person who log on can go to another client computer and can resume her session. Depending on the system architecture, they may start their new session again without concerning their previous state of system crash.

This type of a system suffers from some drawbacks also . All the computer processing is to be done at central server and it controls remote clients as well . This system is totally depends on Central computer. If the central computer fails to operate, entire system will become unavailable.

Another disadvantage is that central computing relies heavily on the quality of administration and resources provided to its clients. If the central server be loaded (e.g. size of home directories, problems regarding administration), then accessibility for clients will affect greatly. In the reverse condition, however, (i.e., a system supported better than our needs) is one of the key advantages to centralized computing.

B.Decentralized Architecture (p2p network):

Existing social networking services are centralized and the companies providing the services have the sole authority to control all the data of their users. It is not a simple task for any profile user to transfer their profile data like their posts, messages to friends into another application, as there are not many good channels to port all the data from one platform to another.Mostly, users don'thave control over how and what information about they are presented to their friends online.The type of social networking service decides how should be the presentation of profile data takes place.

Likewise, the users have to agree to the policies ofthe social networking sites when creating profile on the social website even if user's data goes for advertisement purposes. In addition, very often users need to explicitly get rid from certain advertisements if they are more aware about the security of their profile data. For example, social website Facebook, uses advertisement system for business intention, whichupset many users because it shows advertisements in a news feed about their friend's profile activities in external web sites. However with decentralized architectures administrator unable to control data and the capability to enforce arbitrary decisions like that.

In a decentralized social networking architecture, a user need not to join any particular social networking service such asFacebook or MySpace. Instead, the user chooses a his FOAF (Friend-Of-AFriend) file, his activity log and his photo posts. We can refer to these files by using their URI's which may located on different servers.

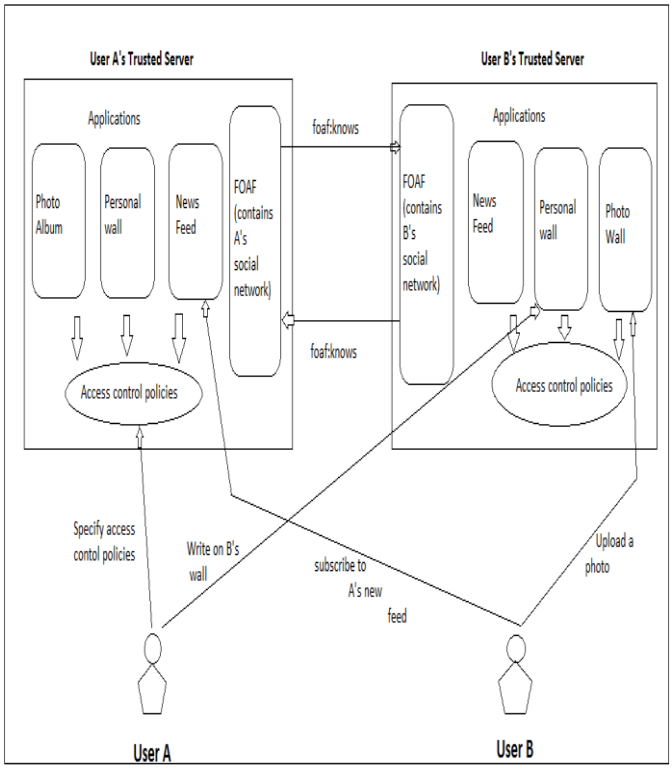

Figure : A framework of DOSN

II.PREDICATE ENCRYPTION:

A. Concept

In the predicate encryption the attribute based policies are used.In this kind of cryptographic primitives at the time of encryption access policies are defined.Then decryptor must have to satisfy the policies.In predicate encryption master secret keys are used.By using master secret key decryption keys can be generated for decryption purpose.

ISSN: 2278 – 1323

In the predicate encryption the secrete keys corresponds to predicate and ciphertext is associated with attributes.Predicate encryption is identity based encryption.The secret key *SKf*corresponding to a predicate *f* can used to decrypt a ciphertext associated with attribute *I* if and only if *f*(*I*) = 1.This type of schemes are nowadays constructed and they are known for relatively few classes of predicates.In PE the sender first creates the message of name M.Then he encrypts that message associated with public key PK.At the receiver's side the user who has secret key associated with public key PK can only decrypts the message.

PE is mostly used for categorise data, as well as to users permitted to read data associated with the particular keywords in question. Or, in a hospitals and clinics, a physician who has treated the patient in the past can able to access patient's clinical record.

As per the above application they require new cryptographic mechanism which should provide more fined control over access to encrypted data.

B. Definition:

Let us define the syntax of PE and the security properties discussed informally in the introduction.

A predicate encryption scheme for the class of predicates F over the set of attributes $\sum$ consists of four algorithms 1.Setup,2.GenKey,3.Enc,4.Decsuch that:

*1.Setup*takes as input the security parameter $1^n$and outputs a (master) public key PK and a (master) secret key SK.

*2.GenKey*takes as input the master secret key SK and a predicate f ϵ F. It outputs a key SK_f.

*3.Enc*takes as input the public key PK, an attribute I $\epsilon \sum$, and a message M in some associated message space. It returns a ciphertext C. We write this as $C \leftarrow Enc_{pk}(I,M)$.

*4.Dec*takes as input a secret key SK_f and a ciphertext C. It results into message denoted by M or symbol $\perp$.

For correctness, we require that for all n, all (PK,SK) generated by Setup(1^n), all f ϵ F, any key $SK_f \leftarrow GenKey_{sk}(f)$, and all I $\epsilon \sum$:

If f(I) = 1 then $Dec_{sk(f)}(Enc_{pk}(I,M)) = M$.

If f(I) = 0 then $Dec_{sk(f)}(Enc_{pk}(I,M)) = \perp$ with all but negligible probability.

C.Assumptions

We now state the assumptions we use to prove security of our construction. As remarked earlier, these assumptions are new but we justify them by proving that they hold in the generic group model under the assumption that finding a non-trivial factor of *N* (the group order) is hard. At a minimum, our construction can be viewed as secure in the group model architecture. Until, we state assumptions made in explicit manner and highlight that they are non-interactive and of fixed size.

Let *G* be as above. We say that *G* satisfies if theadvantage of any algorithm *A* in the following experiment is negligible in the security parameter *n*:

1.G(1^n) is run to obtain (*p*,*q*, *r*,G,G_T ,ê). Set *N* =*pqr*, and letg_p,g_q,$g_r$becreator of G_p, G_q, and G_r, respectively.

2. *A* outputs a bit *º0*, and succeeds if *º0* = *º*.

A is very beneficial as the absolute value of the difference between its success probability and 1=2.

D.Access policies

Based on the primitive of vector inner products, onecan implement various more complex access formulas. Theconstruction of these go via polynomial evaluation, followinga construction by Katz et al. [15]. Here, the access policyis represented by a polynomial p(x) and decryption is possible if the key corresponds to one of the roots of the polynomial.

Based on polynomial evaluation, we can support Boolean CNF or DNF formulas. In caseof disjunction, the attribute "A *V*B" can be expressed using the following univariate polynomial:

$p(x) = (x - A)(x - B) = x^2 - (A + B)x + AB$

The resulting vector is thus

p = (1,−(A + B),AB) And keys representing either A or B can decrypt the corresponding ciphertext.

E.Construction for access policies

ISSN: 2278 – 1323

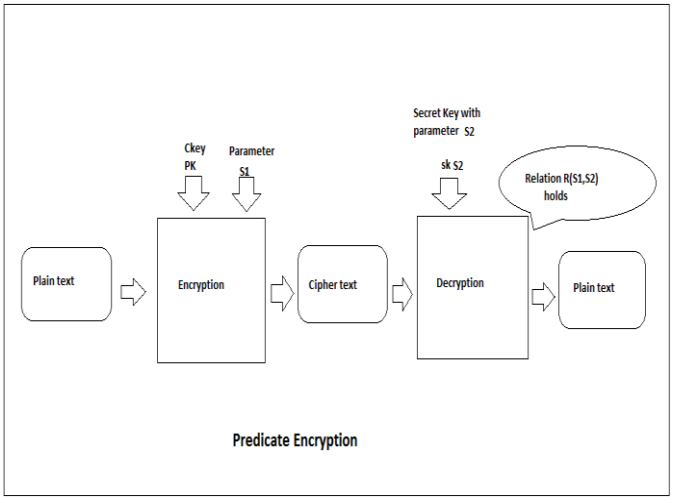

Fig.Predicate Encryption Scheme

Depending on the observation that univariatepolynomials result in more efficient schemes than multivariate ones, we propose to construct access policies solely on univariate polynomials.

We are usingPE scheme with our construction for access policies as "PE with modified access policies" (PEMAP) in the rest of the article.We have not found data on how many groups a single user is typically placed in, but given the low number of groups used, we speculate that a given user would very rarely be in a large fraction of the groups used, and that the scheme would thus work in practice, despite the exponential scaling.

Assume we want to encrypt for "Av(B^C)". This policy is expressed as the following univariate polynomial:

$p(x) = (x - A)(x - B \wedge C) = (x - A)(x - V)$

The given vector can be representedp = (1,−(A + V), AV).

III BLOOM FILTERS

Search engines have primarily focused on presenting the most relevant pages to the user rapidly. A concept which is not well addressed is to remove or group allnear-duplicate documents in the results presented to the user.Enterprise and web search has become a ubiquitous partof the web experience.The initial study by many researchers and recent surveys , shows that around 29.2% of data is common across pages in a sample of 150 million pages.

Similar data can be grouped or eliminated to improve the search experience.A profile in the DOSN contains various objects encrypted for various users. It is impossible for a particular user to determine if an object is encrypted for him without trying to decrypt it since the cipher texts do not reveal access policies.

The main purpose of bloom filter is to show only that data which can be decrypted to the user.so that the rendering speed of user increases and waste of time decreases.

We found that popular search engines, Google, Yahoo, MSN, even today have a significant fraction of near-duplicates in their top results. For example, consider the results of the query "emacs manual" using the Google search engine. we study the current state of popular search engines and evaluate the application of a Bloom filter based near-duplicate detection technique on search results.

A Bloom filter is a space-efficient data structure that represents membership of elements in the set. For finding similar documents, we compare the Bloom filter of one with that of the other. When large number of 1's (bit-wise AND) are shared by two documents then they are marked as similar. In this case, the bit-wise AND can also be perceived as the dot product of the two bit vectors. If the set bits in the Bloom filter of a document are a complete sub- set of that of another filter then it is highly probable thatthe document is included in the another. Web pages are typically made up of fragments, either static ones (e.g. html text pages), or dynamic fragments which changes dynamically as per request.

Let A and B be the two sets being compared for similarity. Let m denote the number of bits (size) in the Bloom filter. For easy approach, Let us assume that both sets have the same number of elements. Let n stands for the number of elements in both sets A and B which is, $|A| = |B| = n$. In that hash function is denoted by k. The probability that a hash function will set a bit for a function h is $1 \leq i \leq k$ is 1m.

A bit can beset by any of the k hash functions for each of the elements. Therefore, the probability that a bit is not set by any hash function for any element is $(1-1/m)^{nk}$.

Thus, the probabilityp, that a given bit is set in the Bloom filter of A is given by

$$P=(1-(1-1/m)^{nk})=1-e^{(-nk/m)}$$

For an element to be considered k bits corresponding to member set should be set. Hence, the false match probability,i.e., Set a is considered to have an outside element pk. Let C denote the

ISSN: 2278 – 1323

intersection of sets A and B and C denote the following formula,

$C = A \cap B$ and $|C| = c$.

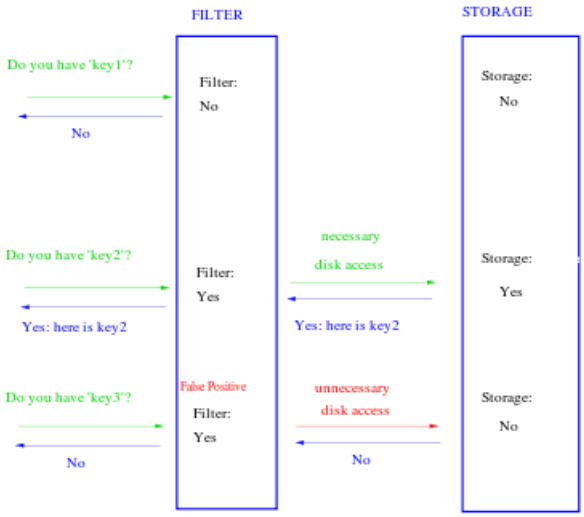

Fig: Bloom filter Storage system

The storage overhead imposed by Bloom filters should beas small as possible. The number of bits m needed to encodemembership information in the Bloom filter depends on the false-positive probability p and the number of elements n to be inserted.

IV CONCLUSION

We have proposed to apply a privacy preserving scheme to the DOSN context: which uses predicate encryption to support univariate polynomial construction.Our predicate scheme is depends on polynomial vectors so hides access policies and increases performance of the system drastically.

We have gone through the deep study of background work.Our main emphasis is on centralized architecture which stores and controls data by central service provider but decentralization of this networks made to change the way people using online social websites.Because control and storage of data is totally user dependent and aimed at removing central service provider.

We have implemented predicate encryption scheme in DOSN by using policy hiding mechanism so that profile user unable to understand the attribute of data.Also it drastically increases performance by using univariate polynomial construction in encryption and decryption process.

Bloom filter used to speed up answers in a key-value storage system. Mostly disk contains values that have slower access time. Bloom filter decisions are much rapid. When the filter gives a positive reports, some memory accesses are made that are not required (in order to weed out the false positives).Memory access speed given by Bloom filter is better than without the Bloom filter. But finally bloom filter operating in this way increases memory accesses.

In this paper we focused on policy hiding mechanism to evaluate performance of the predicate encryption in a decentralized online social network,starting from the security and privacy properties of the original system. Our future scope is to focuson security and privacy, as well as to improve access policies of our modifications.

REFERENCES

[1] G. Greenwald and E. MacAskill, "NSA prism program taps in touser data of apple, Google and others,"2013.[Online]Available:http://www.guardian.co.uk/world/2013/jun/06/us-tech-giants-nsa-data.

[2]S. Jahid, S. Nilizadeh, P. Mittal, N. Borisov, and A. Kapadia, "Decent: A decentralized architecture for enforcing privacy in online social networks," in PerCom Workshops, 2012, pp. 326–332.

[3] L. Cutillo, R. Molva, and T. Strufe, "Safebook: A privacy-preserving online social network leveraging on real-life trust," Communications Magazine, IEEE, vol. 47, no. 12, pp. 94 –101, dec. 2009.

[4] Y. Afify, "Access control in a peer-to-peer social network," Master's thesis, EPFL, Lausanne, Switzerland, 2008.

[5] S. Buchegger, D. Schi¨oberg, L.-H. Vu, and A. Datta, "Peerson: P2P social networking: early experiences and insights," in Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, ser. SNS '09, 2009, pp. 46–52.

[6] S. M. A. Abbas, J. A.Pouwelse, D. H. J. Epema, and H. J. Sips. ,"A gossip-based distributed social networking system", In Proceedings of the 2009 18thIEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises, WETICE '09, pages 93–98, Washington, DC, USA, 2009.

[7] D. Adami, C. Callegari, S. Giordano, M. Pagano, and T. Pepe.," A real-time algorithm for

ISSN: 2278 – 1323

skype traffic detection and classification." In S. Balandin, D. Moltchanov, and Y. Koucheryavy, editors, Smart Spaces and Next Generation Wired/Wireless Networking, volume 5764 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2009.

[8] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in eurocrypt, ser. LNCS, N. Smart, Ed. Springer-Verlag, 2008, vol. 4965, pp. 146–162.

[9]ShirinNilizadeh,SoniaJahid,Prateek Mittal, Nikita Borisov, ApuKapadia,"Cachet: A Decentralized Architecture for Privacy Preserving Social Networking with Caching.",8th acm international conference emerging networking experiments and technologies.

[10]J. Park, "Inner-product encryption under standard assumptions," Designs,Codes and Cryptography, vol. 58, pp. 235–257, 2011.

[11] T. Okamoto and K. Takashima, "Efficient (hierarchical) inner-product encryption tightly reduced from the decisional linear assumption," IEICE Transactions, vol. 96-A, no. 1, pp. 42–52, 2013.

[12] E. Shen, E. Shi, and B. Waters, "Predicate privacy in encryption systems," in Proceedings of the 6th Theory of Cryptography Conference on Theory of Cryptography, ser. TCC '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 457–473.

[13] P. G. Kelley, R. Brewer, Y. Mayer, L. F. Cranor, and N. Sadeh, "An investigation into facebook friend grouping," in Proceedings of the 13thIFIP TC 13 international conference on Human-computer interaction- Volume Part III, ser. INTERACT'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 216–233.

[14] A. Broder, M. Mitzenmacher, and A. B. I. M. Mitzenmacher, "Network applications of bloom filters: A survey," in Internet Mathematics, 2002, pp. 636–646.

[15] S. Tarkoma, C. Rothenberg, and E. Lagerspetz, "Theory and practice of bloom filters for distributed systems," Communications Surveys Tutorials, IEEE, vol. 14, no. 1, pp. 131 –155, 2012.

[16] A. Kirsch and M. Mitzenmacher, "Less hashing, same performance:Building a better bloom filter," Random Struct. Algorithms, vol. 33, no. 2, pp. 187–218, Sep. 2008.

[17] K. N. Hampton, L. S. Goulet, C. Marlow, and L. Rainie, "Why most Facebook users get more than they give," Feb. 2012. [Online]. Available: http://www.pewinternet.org/Reports/2012/Facebook-users/Summary.aspx

[18] L. Backstrom, E. Bakshy, J. Kleinberg, T. M. Lento, and I. Rosenn, "Center of attention: How facebook users allocate attention across friends," in International Conference on Weblogs and Social Media,2011.

[19] R. Jim´enez, F. Osmani, and B. Knutsson, "Sub-second lookups on a large-scale Kademlia-based overlay," in IEEE P2P. IEEE Computer Society, 2011, pp. 82–91.

[20] (2013, Dec.) Alexa: Facebook analitics. [Online]. Available: http: //www.alexa.com/siteinfo/facebook.com

[21] J. Sobel. (2010, Feb.) Making Facebook 2x faster. [Online]. Available: https://www.facebook.com/note.php?note id=307069903919

[22] Z. Yang. (2009, Aug.) Every millisecond counts. [Online]. Available: http://www.facebook.com/note.php?note id=122869103919

ISSN: 2278 – 1323