EXTENDED NYMBLE: METHOD FOR TRACKING MISBEHAVING USERS ANONYMOSLY WHILE BLOCKING

M.Durga Prasad[1], Dr P.Chenna Reddy[2], Banoth Samya[3]

[1]Asst Professor, Department of CSE, Vijay Institute of Tech & Sciences, AP, India,
durgamamidyala@gmail.com
[2] Assoc Professor, Department of CSE, JNTUCE, Pulivendula, Y.S.R Dist, AP, India,
pcreddy1@rediffmail.com
[3]Assoc Professor, Department of CSE, Vijay college of Engineering for women, AP, India,
samyabanoth@gmail.com

ABSTRACT

We have many of anonymizing networks which provide users to access services of server anonymously. Anonymity has received increasing attention in the literature due to user awareness of their privacy. Nowadays, anonymity provides protection to users to enjoy network services without being traced. Anonymizing networks such as Tor allow users to access Internet services privately by using a series of routers to hide the client's IP address from the server. . Web site administrators routinely rely on IP-address blocking for disabling access to misbehaving users, but blocking IP addresses is not practical if the abuser routes through an anonymizing network. As a result, administrators block all known exit nodes of anonymizing networks, denying anonymous access to misbehaving and behaving users alike. To address this problem Nymble is developed, a system in which servers can "blacklist" misbehaving users. We present extensions to nymble framework for anonymizing blacklisting schemes. First, we provide a mechanism to nymble manager to track blacklisting of user in multiple linking windows while preserving anonymity of the users. Some users always look to misbehave with servers; there major intention is to make the server down. The problem with nymble is nymble manager blacklist a user for one likability window (i.e. 1 day), on the other day again he can misbehave with same server or other server. He can continue it as his everyday activity as Nymble manager doesn't have any mechanism to identify such type of users while preserving anonymity. To address this problem, we present a method which can identify such users, while preserving anonymity and nymble manager with identified information can decide upon how much time to blacklist a misbehaving user. We also suggest some algorithms which are changed in comparison to existing nymble.

KEYWORDS

Anonymous blacklisting, Anonymous Tracking, privacy, Nymble, Pseudo Tracker

1. INTRODUCTION

Anonymizing networks (such as Tor or I2P) provide a way to anonymize Internet communications, so as to make it hard to link communication parties (e.g., a user and the web server he/she is visiting). Those anonymizing networks often rely on a distributed overlay network and on onion routing to anonymize TCP-based applications like WEB browsing or P2P. Real-world deployments of anonymizing networks, however, have had limited success because of

DOI : 10.5121/ijcsit.2013.5107

their misuse .Websites Administrators are unable to blacklist malicious users' IP addresses because of their anonymity. Left with no other choice, these administrators opt to blacklist the entire anonymizing network. This approach eliminates malicious activity through such networks, but at the cost of the anonymity. With this approach we are unable to differentiate between behaving and non-behaving users.

To address this problem Nymble is developed where we can block misbehaving users anonymously while allowing behaving users to use the services of server. We have identified drawbacks in Nymble system and proposed Extended Nymble system. Nymble manager can blacklist a misbehaving user by collecting seed for a particular nymble and linking linkability window. This seed can be used to link future connections of this misbehaving user. Nymble manager makes misbehaving users linkable for one Linkability window (i.e. 1 day).After this Misbehaving users become unlikable. On the other day if the same user again misbehaves again he will be blacklisted, this Misbehaving can be a regular activity of certain users.

In Existing Nymble we don't have any technique to track such users because of backward Unlinkability. We have proposed a model which can track users with anonymity and backward Unlinkability.

In the same way Nymble Manager generates Nymble and gives it to a user by given pseudonym–server pair, so a nymble changes when user connects to different server. If a user misbehaves with different servers we don't have any mechanism to blacklist misbehaving users, as nymble changes. Our proposed model deals to solve this problem. We also presented changed data structures when compared to existing nymble.

2. EXTENDED NYMBLE APPROACH

We present our extended nymble approach and in the subsequent sections we explain the changed data structures in our extended nymble system.

2.1. Resource-based blocking

To limit the number of identities a user can obtain, the nymble system binds nymbles to resources that are sufficiently difficult to obtain in great numbers. For example, we can use IP addresses as the resource in our implementation, but our scheme generalizes to other resources such as email addresses, identity certificates, and trusted hardware. Here, Pseudonym Manager maintains identity information of users such that chosen resource or combination of resources uniquely identifies the user.

2.2. Pseudo tracker based tracking

Some users always look to misbehave with servers; there major intention is to make the server down. The problem with nymble is nymble manager blacklist a user for one likability window (i.e. 1 day), on the other day again he can misbehave with same server or other server. He can continue it as his everyday activity as Nymble manager doesn't have any mechanism to identify such type of users while preserving anonymity.

To address this problem, Pseudo Tracker is developed (as shown in Fig 1) as part of Pseudonym Manager in our Extended Nymble System. Pseudo tracker contains identity information of the user and Rating. A user registered newly is highly rated. This rating is used to track the users. If a user misbehaves with a server then the server complaints to Nymble Manager (NM). NM Complaints the particular Pseudonym to Pseudonym Manager (NM complaints only Pseudonym

of misbehaving user but not the server with which he misbehaved to preserve anonymity of user).Pseudonym Manager sends this information to Pseudo Tracker, where the rating of misbehaving user deteriorate depending on no of times he misbehaved. NM uses rating to blacklist a user for many linkability windows.

2.3. Pseudonym manager

The user must first contact the Pseudonym Manager (PM) and demonstrate control over a resource; for IP-address blocking, the user must connect to the PM directly (i.e., not through a known anonymizing network), as shown in Fig. 1.We assume the PM has knowledge about Tor routers, for example, and can ensure that users are communicating with it directly. Pseudonyms are deterministically chosen based on the controlled resource, ensuring that the same pseudonym is always issued for the same resource.

Note that the user does not disclose what server he or she intends to connect to and the PM's duties are not limited to mapping IP addresses (or other resources) to pseudonyms. Whenever a pseudonym is given for a particular user the PM enrolls the details of the user into pseudo tracker. Pseudo tracker contains Identity information and Rating. Identity information is provided by the user which is unique and used for tracking users. Whenever a new user registers with pseudonym manager by giving identity information PM maintains the identity details of the user and rating in pseudo tracker. For Newely registered user the rating will be high (For ex-10). The user as we will explain, the user contacts the PM only once per linkability window (e.g., once a day).On the other day as the registered users provide same identity, Pseudo tracker can be used to maintain identity and rating details of a user.

2.4. Nymble manager

After obtaining a pseudonym from the PM, the user connects to the Nymble Manager (NM) through the anonymizing network, and requests nymbles for access to a particular server (such as Wikipedia). A user's requests to the NM are therefore pseudonymous, and nymbles are generated using the user's pseudonym and the server's identity. These nymbles are thus specific to a particular user-server pair. Nevertheless, as long as the PM and the NM do not collude, the Nymble system cannot identify which user is connecting to what server; the NM knows only the **pseudonym-server** pair, and the PM knows only the user **identity-pseudonym** pair.

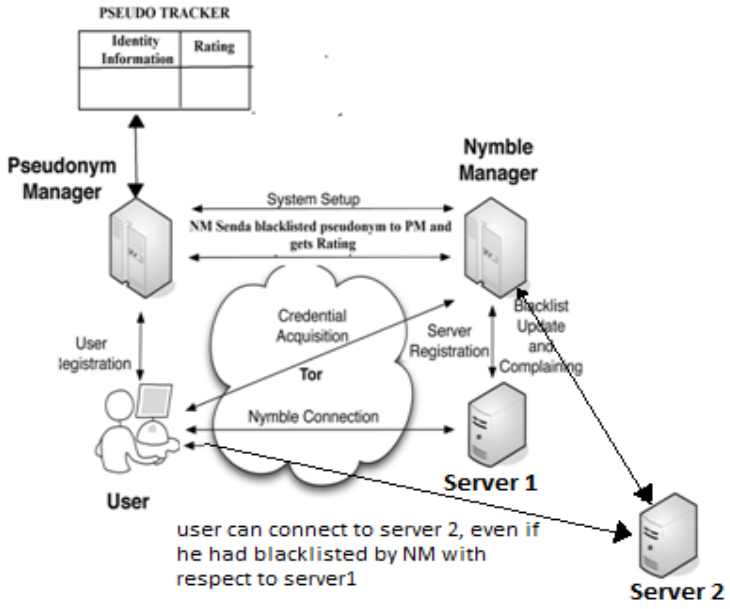

Figure 1: The Extended Nymble system architecture showing the various modes of Interaction. Note that users interact with the NM and servers though the anonymizing network

To provide the requisite cryptographic protection and security properties, the NM encapsulates nymbles within nymble tickets. Servers wrap seeds into linking tokens, and therefore, we will speak of linking tokens being used to link future nymble tickets. The importance of these constructs will become apparent as we proceed. Whenever a user is blacklisted the pseudonym of the particular user is send to PM(note that only pseudonym is send but not the name of server the user misbehaved, to preserve anonymity and backward Unlinkability).

2.5. PM to block frequently misbehaving users

In the above figure 1, user can connect to other server even if he blacklisted with server 1. This is because NM blacklists misbehaving users by collecting $seed_0$ of a particular nymble. $seed_0$ is obtained with server identity. So, if user misbehaves with one server had a chance to connect to other server anonymously. If it becomes his everyday activity to misbehave with different servers we don't have any mechanism in existing nymble. Extended nymble solve this by pseudo tracker where we can have rating which shows frequent behaviour of user.PM can have mechanism depending on rating to block users who misbehaved frequently(very low rated) to blacklist for some linkability windows

2.6. Time

Nymble tickets are bound to specific time periods. As illustrated in Fig. 2, time is divided into linkability windows of duration W, each of which is split into L time periods of duration T (i.e., W =L* T). We will refer to time periods and linkability windows chronologically as $t_1; t_2; \ldots ; t_L$ and $w_1; w_2; \ldots$, respectively. While a user's access within a time period is tied to a single nymble ticket, the use of different nymble tickets across time periods grants the user anonymity between time periods. Smaller time periods provide users with higher rates of anonymous authentication, while longer time periods allow servers to rate-limit the number of misbehaviors

from a particular user before he or she is blocked. For example, T could be set to five minutes, and W to one day (and thus, L = 288) or many days (L=n*288)

2.7. Blacklisting and Tracking Users

If a user misbehaves with a server then server complaints to Nymble Manager. The Nymble Manager before blacklisting a user gets the details of the user from Pseudonym manager. The pseudonym manager gets the details of the user from Pseudo tracker; Pseudo tracker maintains identity information and rating, if suppose a user misbehaved in past the rating of particular user moves down. Nymble manager gets the rating and if rating of particular user is high (For ex-10), it indicates that user misbehave for first or less frequent times. If rating is Low then Nymble manager can decides upon no of linkability windows the user should be blacklisted.

Nymble manager sends the misbehaving user pseudonym but not the details of the server with which he misbehaved. So our Extended Nymble maintains Anonymous authentication.

If a user misbehaves with a server, server may link any future connection from this user from the current linkability window. Consider Fig. 2 as an example: A user connects and misbehaves at a server during time period t* within linkability window w*.The server later detects this misbehavior and complains to the NM in time period t_c ($t^* < t_c <= t_L$) of the same linkability window w^*.

Whenever a server complaints NM about misbehaving user the NM identifies

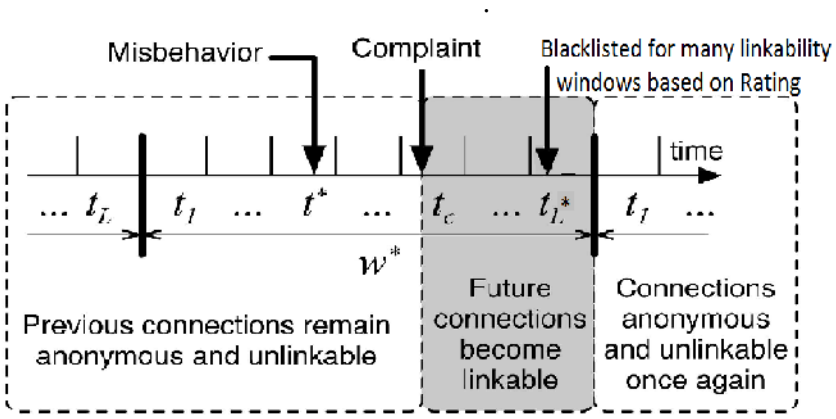

Figure 2: The life cycle of a misbehaving user. If the server complains in time period t_c about a user's connection in t^*, the user becomes linkable starting in t_c. The user is blacklisted to many linkability windows based on rating.

Pseudonym of particular nymble as part of the complaint, the server presents the nymble ticket of the misbehaving user and obtains the corresponding seed from the NM. The server is then able to link future connections by the user in time periods t_c;($t^* < t_c <= t_{L^*}$); t_{L^*} can be linkability window w^* Or many linkability windows. Therefore, once the server has complained about a user, that user is blacklisted for the rest of the day, or many days depending on rating. For example (the linkability window).Note that the user's connections in t_1; t_2; . . . ; t^*; t^*+1; . . . ; t_c remain unlinkable (i.e., including those since the misbehavior and until the time of complaint). Even though misbehaving users can be blocked from making connections in the future, the users' past connections remain unlinkable, thus providing backward unlinkability and subjective blacklisting.

2.7. Notifying the User of Blacklist Status

Users who make use of anonymizing networks expect their connections to be anonymous. If a server obtains a seed for that user, however, it can link that user's subsequent connections. It is of utmost importance then that user's be notified of their blacklist status before they present a nymble ticket to a server. In our system, the user can download the server's blacklist and verify her status. If blacklisted, the user disconnects immediately. Since the blacklist is cryptographically signed by the NM, the authenticity of the blacklist is easily verified if the blacklist was updated in the current time period (only one update to the blacklist per time period is allowed). If the blacklist has not been updated in the current time period, the NM provides servers with "daisies" every time period so that users can Verify the freshness of the blacklist ("blacklist from time period t_{old} is fresh as of time period t_{now}"). these daisies are elements of a hash chain, and provide a lightweight alternative to digital signatures. Using digital signatures and daisies, we thus ensure that race conditions are not possible in verifying the freshness of a blacklist. A user is guaranteed that he or she will not be linked if the user verifies the integrity and freshness of the blacklist before sending his or her nymble ticket.

3. CHANGED DATA STRUCTURES FOR EXTENDED NYMBLE

In extended nymble approach we have changed some data structures used for existing nymble. The changed data structures are related to pseudonyms and tracking and blacklisting.
It is the responsibility of pseudonym to issue pseudonyms to users. Whenever a new user registers with pseudonym manager his identification details are stored in has two components Nym and Mac. Nym is pseudo random mapping of user identity,the linkability window w for which the pseudonym is valid,and the PM secret key $Nymkey_p$, Mac is a MAC that the NM uses to verify the integrity of the pseudonym

Algorithm1: PM Create Pseudonym and Enroll in Pseudo_Tracker

Input: (Uid, w, rating) ϵ H×N
Persistent state: PM state ϵ s_p
Output: Pnym ϵ P & Pseudo tracker ϵ P

1. Check for Uid in Pseudo_Tracker
2. If found ='False'
   2.1 Collect Uid and rating (New user is rated high. For ex :10)
   2.2 Pseudo_Tracker :=(Uid, rating)
   2.3 If failed to Insert repeat steps 3 and 4
3. If (Found = 'True')
4. Extract $Nymkey_p$, $Mackey_{np}$ from PM state
5. Nym=MA.Mac(Uid || w, $Nymkey_p$)
6. Mac=MA.Mac(Nym || w, $Mackey_{np}$)
7. return Pnym = (Nym, Mac)

The above algorithm is used to obtain pseudonym for users.PM before issuing pseudonyms enters the details of user in pseudo tracker, which helps in tracking misbehaving users.

Algorithm 2: PM tracks misbehaving user
Input: (Pnym, T, w)
Output: rating ϵ Misbehaving user

1. Whenever server complaints NM collects Pnym and sends to PM for rating related information.

2. Extract (Nym,Mac) = Pnym
3. MA.Mac (Uid|| w, $Nymkey_p$) = Nym
4. PM collects rating of misbehaving user rating := $rating_{uid}$
5. Updates rating by any choosen unique function $rating_{uid}$:= f($rating_{uid}$)
6. return rating

NM complaints pnym of misbehaving users, but not the server with which he misbehaved. NM doesn't send the details of user with which server he misbehaved, to preserve anonymity of users.

Algorithm 3: NM decides on blacklists based on rating
Input: rating
Output: (no of linkability windows user blacklisted) ϵ Misbehaving user

1. NM collects rating of misbehaving user
2. Decides upon no of days to blacklist(T_L^*) depend upon rating

NM collects the rating information of misbehaving user and decides upon no of days to blacklist misbehaving user. When blacklisted the user becomes linkable throughout linkability windows.

4. CONCLUSIONS

We have proposed extended nymble, which provides users to connect anonymously without giving his identity details to server. We have covered the drawbacks of existing nymble and strengthened the power of nymble. Extended nymble can block misbehaving users as well as track them anonymously. We have preserved anonymity which is the basic property of anonymizing networks.

5. REFERENCES

[1] Patrick P. Tsang, Apu Kapadia, Member, IEEE, Cory Cornelius, and Sean W. Smith "Nymble: Blocking Misbehaving Users in Anonymizing Networks" IEEE transactions on dependable and secure computing, vol. 8, no. 2, march-April 2011.

[2] G. Ateniese, D.X.Song, and G. Tsudik, "Quasi-Efficient Revocation in Group Signatures," Proc. Conf. Financial Cryptography, Springer, pp. 183-197, 2002.

[3] M. Bellare, R. Canetti, and H. Krawczyk, "Keying Hash Functions for Message Authentication," Proc. Ann. Int'l Cryptology Conf.(CRYPTO), Springer, pp. 1-15, 1996.

[4] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption," Proc. Ann. Symp. Foundations in Computer Science (FOCS), pp. 394-403, 1997.

[5] M. Bellare and P. Rogaway, "Random Oracles Are Practical: A Paradigm for Designing Efficient Protocols," Proc. First ACM Conf. Computer and Comm. Security, pp. 62-73, 1993.

[6] M. Bellare, H. Shi, and C. Zhang, "Foundations of Group Signatures: The Case of Dynamic Groups," Proc. Cryptographer's Track at RSA Conf. (CT-RSA), Springer, pp. 136-153, 2005.

[7] D. Boneh and H. Shacham, "Group Signatures with Verifier- Local Revocation," Proc. ACM Conf. Computer and Comm. Security, pp. 168-177, 2004.

[8] S. Brands, "Untraceable Off-Line Cash in Wallets with Observers (Extended Abstract)," Proc. Ann. Int'l Cryptology Conf. (CRYPTO), Springer, pp. 302-318, 1993.

[9] E. Bresson and J. Stern, "Efficient Revocation in Group Signatures," Proc. Conf. Public Key Cryptography, Springer, pp. 190-206,2001.

[10] J. Camenisch and A. Lysyanskaya, "An Efficient System for Non- Transferable Anonymous Credentials with Optional Anonymity Revocation," Proc. Int'l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT), Springer, pp. 93-118, 2001.

[11] J. Camenisch and A. Lysyanskaya, "Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials," Proc. Ann. Int'l Cryptology Conf. (CRYPTO), Springer, pp. 61-76, 2002.

[12] J. Camenisch and A. Lysyanskaya, "Signature Schemes and Anonymous Credentials from Bilinear Maps," Proc. Ann. Int'l Cryptology Conf. (CRYPTO), Springer, pp. 56-72, 2004.

[13] D. Chaum, "Blind Signatures for Untraceable Payments," Proc. Ann. Int'l Cryptology Conf. (CRYPTO), pp. 199-203, 1982. blishers.