Blacklisting Misbehaving Users for Enhancing Security in Anonymizing Networks

Mrs Umama Tahera[1], Mrs MD Asma[2], Mr M.S Qaseem[3]
1. II-M.Tech Student, Nizam Institute of Engineering & Tech.,India.
2. Associate Professor, CSE Dept., Nizam Institute of Engineering & Tech., India.
3. Principal, Nizam Institute of Engineering & Tech., India.

Abstract: *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. The success of such networks, however, has been limited by users employing this anonymity for abusive purposes such as defacing popular Web sites. 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, we present Nymble, a system in which servers can "blacklist" misbehaving users, thereby blocking users without compromising their anonymity. Our system is thus agnostic to different servers' definitions of misbehavior—servers can blacklist users for whatever reason, and the privacy of blacklisted users is maintained.*

Nymble is a system in which servers can "blacklist" misbehaving users, thereby blocking users without compromising their anonymity and the privacy of blacklisted users is maintained. 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.

Keywords: *Anonymous Blacklisting, Anonymizing Networks, Backward Unlinkability, Privacy, Revocation, Realibility and Security.*

I. Introduction

Anonimizing networks such as Tor route traffic through independent nodes in separate administrative domains to hide a client's IP address. Unfortunately, some users have misused such networks—under the cover of anonymity, users have repeatedly defaced popular Web sites such as Wikipedia. Since Web site administrators cannot blacklist individual malicious users' IP addresses, they blacklist the entire anonymizing network.Such measures eliminate malicious activity through anonymizing networks at the cost of denying anonymous access to behaving users.

Nymble is a secure system which provides all the following properties: anonymous authentication, backward unlinkability, subjective blacklisting, fast authentication speeds, rate-limited anonymous connections, revocation auditability (where users can verify whether they have been blacklisted), and also addresses the Sybil attack to make its deployment practical.

.Blacklisting anonymous users.This system provide a means by which servers can blacklist users of an anonymizing network while maintaining their privacy.

.Practical performance.Nymble protocol makes use of inexpensive symmetric cryptographic operations to significantly outperform the alternatives.

.Open-source implementation.With the goal of contributing a workable system,an open-source implementation of Nymble is built, which is publicly available.We provide performance statistics to show that our system is indeed practical.

In Nymble, users acquire an ordered collection of nymbles, a special type of pseudonym, to connect to Web sites.Without additional information, these nymbles are computationally hard to link, and hence, using the stream of nymbles simulates anonymous access to services. Web sites, however, can blacklist users by obtaining a seed for a particular nymble, allowing them to link future nymbles from the same user—those used before the complaint remain unlinkable. Servers can therefore blacklist anonymous users without knowledge of their IP addresses while allowing behaving users to connect anonymously. This system ensures that users are aware of their blacklist status before they present a nymble, and disconnect immediately if they are blacklisted.

1.1 Motivation

The motivation of the present work is to provide a means by which servers can blacklist users of an anonymizing network while maintaining their privacy. With the goal of contributing a workable system, an open-source implementation of Nymble is built, which is publicly available. Nymble protocol makes use of inexpensive symmetric cryptographic operations to significantly outperform the alternatives.

1.2 Organization

This paper is organized as follows – Section 2 deals with related work, Section 3 presents the architecture model and methodology and Section 4 is about the problem definition. Section 5 gives the implementation of the proposed algorithm and performance analysis. Section 6 contains the conclusion. Nymble have been proposed to address the problem where administrators deny anonymous access to misbehaving and behaving users alike. In this system servers can blacklist misbehaving users, thereby blocking users without compromising their anonymity.

II. Related Work

To address the problem of blocking exit nodes which denies anonymous access to honest and dishonest users alike, Johnson et.al[1] proposed a system in which honest users remain anonymous and their requests unlinkable; a server can complain about a particular anonymous user and gain the ability to blacklist the user for future connections. This blacklisted user's accesses before the complaint remain anonymous.

Many security mechanisms are based on specific assumptions of identity and are vulnerable to attacks when these assumptions are violated. Levine et.al[2] presented the impact of the Sybil attack, an attack against identity in which an individual entity masquerades as multiple simultaneous identities. The Sybil attack is a fundamental problem in many systems, and it has so far resisted a universally applicable solution.

Boneh et.al [3] constructed a short group signature scheme that supports Verifier-Local Revocation (VLR). In this model, revocation messages are only sent to signature verifiers (as opposed to both signers and verifiers). Consequently there is no need to contact individual signers when some user is revoked. This model is appealing for systems providing attestation capabilities.

Nakanishi et.al [4] proposed a novel group signature satisfying the regular requirements. Furthermore, it also achieves the following advantages: (1) the size of signature is independent of the number of group members; (2) the group public key is constant; (3) Addition and Revocation of group members are convenient; (4) it enjoys forward security; (5) The total computation cost of signature and verification requires only 8 modular exponentiations.

Ateniese et.al [5] proposed a new group signature scheme that is well suited for large groups, i.e., the length of the group's public key and of signatures do not depend on the size of the group. This scheme is based on a variation of the RSA problem called strong RSA assumption. It is also more efficient than previous ones satisfying these requirements.

III. Methodology

Each server may register at most once in any likability window. Logins may be used to provide credentials when creating a client connection. Whether or not logins are required depends on the method calls used to start the server or create the connection. 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. After obtaining a pseudonym from the PM, the user connects to the Nymble Manager (NM) through the anonymizing network. Servers update their blacklists for the current time period. The architecture and modeling of the proposed algorithm is shown in Fig.1.

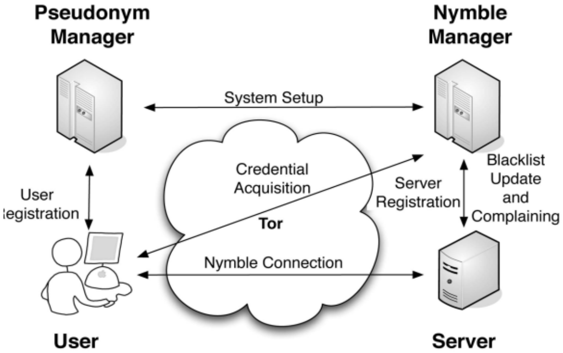

Figure.1.The Nymble system architecture showing the various modes of interaction.

3 .1 Server Registeration

To participate in the Nymble system, a server with identity Sid initiates a type-Auth channel to the NM, and registers with the NM according to the Server Registration protocol below. Each server may register at most once in any likability window. Logins may be used to provide credentials when creating a client connection. Whether or not logins are required depends on the method calls used to start the server or create the connection. For example, you might need logins for pooling. If you do not use logins, you must track and specify the user credentials manually.

Servers update their blacklists for the current time period for two purposes. First, as mentioned earlier, the server needs to provide the user with its blacklist (and blacklist certificate) for the current time period during a Nymble connection establishment. Second, the server needs to be able to blacklist the misbehaving users by processing the newly filed complaints (since last update).

3.1.1 Blacklist Update. Servers update their blacklists for the current time period for two purposes. First, as mentioned earlier, the server needs to provide the user with its blacklist (and blacklist certificate) for the current time period during a Nymble connection establishment. Second, the server needs to be able to blacklist the misbehaving users by processing the newly filed complaints (since last update).

3.2 User Registeration

A user with identity uid must register with the PM once in each likability window. To do so, the user initiates a type- Basic channel to the PM, followed by the User Registration protocol escribed below. A login generally requires the user to enter two pieces of information, first a user name and then a password. This information is entered into a login window on a GUI (graphical user interface). A user name, also referred to as an account name,is a string that uniquely identifies a user. User names can be the same as or related to the real names of users, or they can be completely arbitrary. A password is likewise a string, but it differs from a user name in that it is intended to be kept a secret that is known only to its use.

3.3 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.It is assumed that the PM has knowledge about Tor routers, and can ensure that users are communicating with it directly. Pseudonyms are deterministically chosen based on the controlled resource, ensuring that the same pseudonyms are always issued for the same resource.

The user does not disclose what server he or she intends to connect to, and the PM's duties are limited to mapping IP addresses (or other resources) to pseudonyms. As we will explain, the user contacts the PM only once per linkability window (e.g., once a day).

3.4 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.

To provide the requisite cryptographic protection and security properties, the NM encapsulates nymbles within nymble tickets. Servers wrap seeds into linking tokens, and therefore,the linking tokens are being used to link future nymble tickets.

IV. Algorithm

A. Problem Definition: Nymble aims for four security goals: Blacklistability, Nonframeability, Rate-limiting and Anonimity.

B. Assumptions: The servers and the users are allowed to be corrupt and controlled by an attacker.

C. Algorithm: Nymble uses the following building blocks:

- Secure cryptographic hash functions. These are one- way and collision-resistant functions that resemble random oracles. Denote the range of the hash functions by H.
- Secure message authentication (MA). These consist of the key generation (MA.KeyGen), and the message authentication code (MAC) computation (MA.Mac) algorithms. Denote the domain of MACs by M.
- Secure symmetric-key encryption (Enc). These consist of the key generation (Enc.KeyGen), encryp- tion (Enc.Encrypt), and decryption (Enc.Decrypt) algorithms. Denote the domain of

ciphertexts by Γ.

- Secure digital signatures (Sig). These consist of the key generation (Sig.KeyGen), signing (Sig.Sign), and verification (Sig.Verify) algorithms. Denote the domain of signatures by Σ.

4.1 Algorithm Used

```
Algorithm 1. PMCreatePseudonym
Input: ðuid;wÞ 2 H_NN
Persistent state: pmState 2 SP
Output: pnym 2 P
1: Extract nymKeyP ; macKeyNP from pmState
2: nym :¼ MA:Macðuidkw;nymKeyP Þ
3: mac :¼ MA:Macðnymkw;macKeyNPÞ
4: return pnym :¼ ðnym; macÞ
Algorithm 2. NMVerifyPseudonym
Input: ðpnym;wÞ 2 P _NN
Persistent state: nmState 2 SN
Output: b 2 ftrue; falseg
1: Extract macKeyNP from nmState
2: ðnym; macÞ :¼ pnym
3: return mac ¼ ? MA:Macðnymkw;macKeyNP Þ
```

V. Implementation And Performance Analysis

5.1 Implementation and Experimental Setup

Nymble is implemented as a C++ library along with Ruby and JavaScript bindings.However, users can easily compile bindings for any of the languages (such as Python, PHP, and Perl) supported by the Simplified Wrapper and Interface Generator(SWIG), for example. OpenSSL for all the cryptographic primitives are utilized.

Here Fig.8(a) shows the amount of time it takes the NM to perform various protocols.For blacklist updates,the initial jump in the graph corresponds to the fixed overhead associated with signing a blacklist. Fig.8(b) shows the amount of time it takes the server and user to perform various protocols. These protocols are relatively inexpensive by design.

To check the performance of the Nymble, it is evaluated on a 2.2 GHz Intel Core 2 Duo

Macbook Pro with 4 GB of RAM. The PM, the NM, and the server were implemented as Mongrel (Ruby's version of Apache) servers. The user portion was implemented as a Firefox 3 extension in JavaScript with XPCOM bindings to the Nymble C++ library. For each experiment relating to protocol performance, the average of 10 runs were reported. The evaluation of data structure sizes is the byte count of the marshaled data structures that would be sent over the network.

5.2 Experimental Results

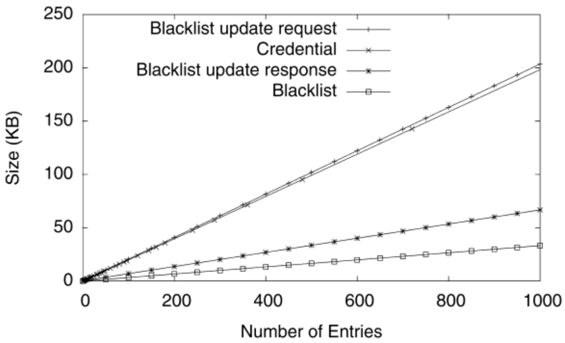

Fig.2 The marshaled size of various Nymble data structures. The X-axis refers to the number of entries—complaints in the blacklist update request,tickets in the credential, tokens and seeds in the blacklist update response, and nymbles in the blacklist.

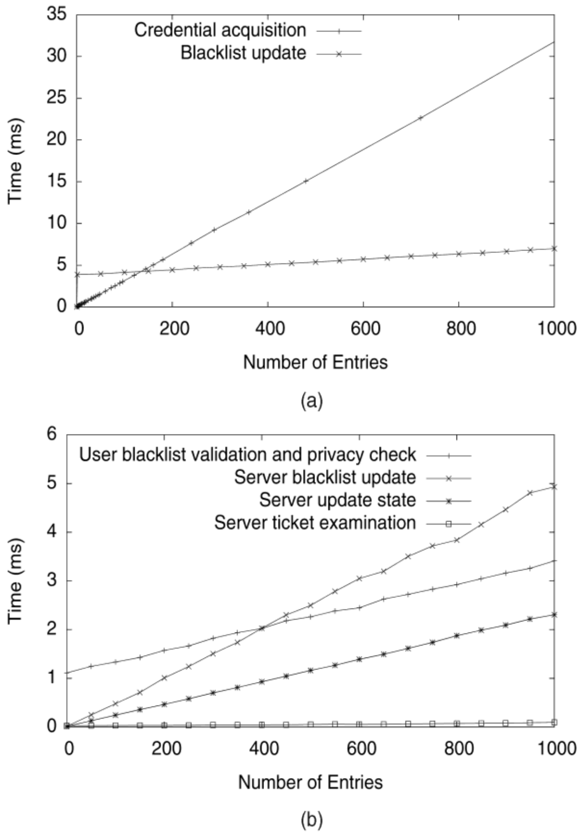

Fig. 3: Nymble's performance at (a) the NM and (b) the user and the server when performing various protocols. (a) Blacklist updates take several milliseconds and credentials can be generated in 9 ms for the suggested parameter of L ¼ 288. (b) The bottleneck operation of server ticket examination is less than 1 ms and validating the blacklist takes the user only a few ms.

Fig.2 shows the size of the various data structures. The X-axis represents the number of entries in each data structure—complaints in the blacklist update request, tickets in the credential (equal to L, the number of time periods in a linkability window),nymbles in the blacklist, tokens and seeds in the blacklist update response and nymbles in the blacklist.
In general,each structure grows linearly as the number of entries increases. Credentials and blacklist update requests grow a t the same rate because a credential is a collection of tickets which is more or less what is sent as a complaint list when the server wishes to update its blacklist.

Fig. 3a shows the amount of time it takes the NM to perform various protocols. This protocol occurs only once in every linkability window for each user wanting to connect to a particular server. For blacklist updates, the initial jump in the graph corresponds to the fixed overhead associated with signing a blacklist.

Fig.3b shows the amount of time it takes the server and user to perform various protocols. These protocols are relatively inexpensive by design, i.e., the amount of computation performed by the users and servers should be minimal.

VI. Conclusion

In this paper a comprehensive credential system called Nymble is proposed, that adds an additional layer of security to the anonymous networks. This system is used to block the misbehaving users in anonymzing networks without affecting their privacy and anonymity.It provides anonymous authentication, backward unlinkability, subjective blacklisting, fast authentication speeds, rate-limited anonymous connections, revocation auditability, that is the users can verify whether they have been blacklisted. The System ensures that users are aware of their blacklist status before they present a nymble, and disconnect immediately if they are blacklisted.
The experimental results obtained are satisfactory. Thus this method is practical, effective and efficient to the needs of both users and services.

In the existing methodologies the message frequency is very high, the NM forwards only the simple messages. Latency is more and hence the speed gets reduced.

In future this work will be enhanced to work on a remote machine. This can also be extended into a multiple rounds of pseudonym construction in which the PM participates in multiple rounds of communication with the user. This adds one more layer of security to the system. This may increase the speed and mainstream acceptance of anonymizing networks such as Tor, which has been completely blocked by several services because of users who used their anonymity.

Acknowledgement

I Thank Dr. Mohd. Sanaullah Qaseem, Principal Nizam Institute of Engineering and Technology, Dr. S.M.Afroz, Professor & Dean, Nizam Institute of Engineering and Technology and Mrs. MD Asma, Associate Professor, Nizam Institue of Engineering and Technology for their constant support and encouragement.

References

[1] P.C. Johnson, A. Kapadia, P.P. Tsang, and S.W. Smith, “Nymble: Anonymous IP-Address Blocking” in *Proc, Conf. Privacy Enhancing Technologies,* Springer, pp. 113-133, 2007.

[2] B.N. Levine, C. Shields, and N.B. Margolin, “A Survey of Solutions to the Sybil Attack”, *Technical Report 2006-052,* Univ. of Massachusetts, Oct. 2006.

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

[4] T. Nakanishi and N. Funabiki, “Verifier-Local Revocation Group Signature Schemes with Backward Unlinkability from Bilinear Maps”, *Proc. Int’l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT),* Springer, pp. 533-548, 2005.

[5] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik,” A Practical and Provably Secure Coalition-Resistant Group Signature, Scheme”, *Proc. Ann. Int’l Cryptology Conf. (CRYPTO),* Springer, pp. 255-270, 2000.

[6] R. Dingledine, N. Mathewson, and P. Syverson, “Tor: The Second- Generation Onion Router,” *Proc. Usenix Security Symp.,* pp. 303-320, Aug. 2004.

[7] C. Cornelius, A. Kapadia, P.P. Tsang, and S.W. Smith, “Nymble: Blocking Misbehaving Users in Anonymizing Networks,” *Tech- nical Report TR2008-637,* Dartmouth College, Computer Science, Dec. 2008.

[8] J.E. Holt and K.E. Seamons, “Nym: Practical Pseudonymity for Anonymous Networks,” *Internet Security Research Lab Technical Report 2006-4,* Brigham Young Univ., June 2006.

[9] A. Lysyanskaya, R.L. Rivest, A. Sahai, and S. Wolf, “Pseudonym Systems,” *Proc. Conf. Selected Areas in Cryptography,* Springer, pp. 184-199, 1999.

[10] J.R. Douceur, “The Sybil Attack,” *Proc. Int’l Workshop on Peer-to- Peer Systems (IPTPS),* Springer, pp. 251-260, 2002.