Volume 4, Issue 1, January 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Providing Security to Web Applications in Anonymizing Networks Using Nymble

Momi Maity, Neha Verma, Rupali Wadikar, Sayali Shevkar, Prof. V.K. Bhusari

Department of Computer Engineering ,

JSPM's BSIOTR(w),Pune, India

Abstract : With Web applications remaining a popular target for attackers,ensuring security in web application has become very crucial specially when the attacker is trying to access the applications through anonymizing network. Tor is a well known and most commonly used anonymizing network. An anonymizing network is a network in which the users involved in this network have hidden identity. This means the user's IP addresses are hidden. Tor stands for 'The onion router'. Unfortunately, there is a vital disadvantage of tor. If a user involved in tor misbehaves then the entire network is blocked. This entire network involves innocent users too. Hence along with the guilty user even the faithful users are blocked. This happens because the server is unaware of the address of the guilty user hence it blocks the entire network. To overcome this disadvantage Nymble system can be used. Nymble's main goal is to protect privacy of the user with respect to the server they connect. Nymble system allows web servers to selectively blacklist users of anonymous network such as TOR without the knowing identity of user and these blacklisted users are not allowed for future connection for some duration (which can be changed). Our system thus focuses on different server's definition of misbehaviours, detecting those misbehaviours, blocking selectively certain users instead of whole network and allowing all other honest users to stay anonymous.

Keyword : web application, anonymizing network, selectively blacklist, misbehavior

I. INTRODUCTION

As the world is developing, numerous technologies are emerging day by day. And as the technology evolves and changes, simultaneously so do the hacking attack methods. Most of the cyber attacks are done at web application level. Hence ensuring security in web application is crucial. Web application[1] is an application that the user access over the internet. Generally, any software which is accessed using a web browser could be called a web application.

Using web applications, there is no need to worry about installing and maintaining the application. And furthermore, it provides support for different platforms and is easy to use. Due to these reasons, web applications have gained a vast popularity. If no care is taken for web application's security or we may say that if web application vulnerability is allowed to happen, then not only the entire database is at serious risk, but the website can become the launch site of criminal activities or used to transfer illegal content. Due to this lack of web application security, some hackers take advantage. When you hear about web application security, the immediate thought that comes to mind is the attacker defacing web sites, stealing personal data, bombarding websites with denial of service attacks or uploading some virus infected database. Thus, for ensuring security for web application in anonymizing networks like Tor[2] several systems had been developed in past as anonymous credential system, blind signature, basic group signature[3], traceable signature[4], ring signature Etc for blocking the misbehaving users and the users may access the server based on the credentials provided to them and if they are not authenticated each won't be able to access the websites however all these methods .These methods hold true for few definitions of misbehavior and it is exhaustive to map more complex definitions of misbehavior with the approaches. Verifier-local revocation (VLR) overcomes this by requiring the verifier to perform only local updates during revocation. But VLR needs heavy computations at the .In contrast, our scheme takes the server about one millisecond per authentication, which is faster than VLR.

II. OUR SOLUTION

We designed a Nymble system, which provides the properties like anonymous authentication, backward unlinkability, subjective blacklisting, and fast authentication speeds, rate limited anonymous connections and revocation auditability (where users can verify whether they have been blacklisted)**.** Nymble's main goal is to protect privacy of the user with respect to the server they connect. Nymble system allows websites to selectively blacklist users of anonymous network such as TOR[5] without the knowing IP-address of user and these blacklisted users are not allowed for future connection for some duration (which can be changed). Our Nymble system blacklist only the misbehaving users and the behaving users are allowed to access the different web application. Our system ensures that the user is aware of their status (whether they are blocked or unblocked) .If the user is blacklisted and tries to login then they are disconnected immediately for a particular duration of time. Nymble architecture involves pseudonym manager and Nymble manager. Pseudonym manager is responsible for user's anonymity. Initially the user interacts with pseudonym manager and hence

obtains a pseudo name for its access. Now through the pseudonym the user accesses data. Consequently pseudonym provides identity resulting for allowance of individual blocking instead of entire network. Nymble manager is responsible for user registration and communication with the server. Only the server has authority to block the user. Misbehavior act includes password attack, Data modification attack, DOS attack etc.

Nymble is a system that ensures forgiveness which means that the bad behaviour is forgiven after a certain amount of time. Thus, our system allow behaving user's to enjoy anonymity while blacklisted users are not allowed for future access for particular time duration and their previous connection remain unlinkable. Thus these properties of our system enable websites to block selectively certain users instead of whole network and allowing all other honest (behaving) users to stay anonymous.

III. SYSTEM OVERVIEW

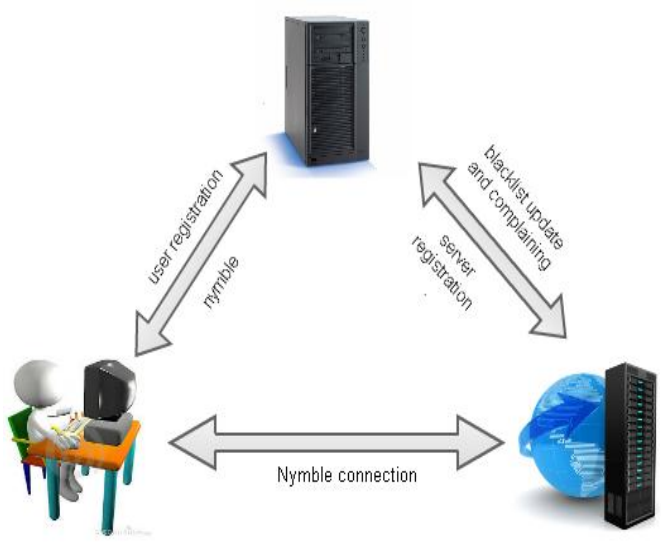

Fig 1: system architechture

A. Nickname Manager

Our Nickname Manager is comprised of Pseudonym manager (PM) and Nymble manager. The pseudonym's sole responsibility is to map IP address to pseudonyms. The user first connects to the pseudonym manager (PM) in order to receive a pseudonym and takes control over the resource i.e. the user's IP address. The user must connect to the pseudonym manager[6] (PM) directly that means not through a network which is known. Pseudonym manager (PM) has no knowledge of the user's destination, and the PM's duties are limited to mapping IP address.

The Nymble Manager (NM) gets connected to the user by presenting user's pseudonym and its target server. The Nymble Manager (NM) has no knowledge of the user's IP address but the pseudonym provided by the pseudonym manager (PM) guarantees that some unique IP address maps it to the pseudonym. The user receives a set of nymble tickets as credentials for the target server by the Nymble manager. These Nymble tickets are unlinkable, and therefore user can use these tickets (once each) to gain anonymous access at the destination server.

Equation:

$$(a)\ f(x) = \sum_{i=0}^{i=n} U_i$$

$$(b)\ P_s = P\big(f(x)\big)$$

Where

- f(x) is function to concatenate all string of the field from the user profile
- Ui -> each profile attributes
- Ps-> Pseudonym
- P(f(x)) -> Random Function to calculate Pseudonym

Algorithm:

Input : Set U ={ u1,u2,u3……un}
Output : pseudonym (Ps)
Step 0: Get the User Profile attribute set U
Step 1: Convert all the attributes to String type
Step 2: Concatenate all the String to get a single String
Step 3: Get the auto incremented User ID as I
Step 4: x=ID mod 7

Step 5: for i=0 to String length
Step 6: Fetch xth character from the String
Step 7: Continue till 7 characters are selected
Step 8: concatenate all the 7 characters
Step 9: return Pseudonym

B. Attacks To Be Implemented

We are implementing Denial of Service , Data Modification , Man in the Middle , Password based , Application layer and Compromised key attack in our nymble system.

1) ***Denial of Service attack :***

Denial of Service (DoS) is one of the most common attack on the web applications. It is an attack technique which restrict the web sites from allowing the normal users to access the service of that particular website for a particular duration of time. Randomization of internal information system, sending invalid data to applications or network services,floodinh network with traffic and blocking traffic leads to Denial of Service.

Dos attack can be applied to application as well as network layer. But it is most easily applied to network layer. Dos at the network layer requires a large amount of connection attempts whereas Dos at the application layer may target specific user, database server or web server individually.

Equation:

$$f(dos) \Rightarrow UP_{data} > lim$$

Where

- f(dos) is function to identify dos attack
- UP $_{data}$ -> Uploading data
- Lim -> Limits

Dos attack in the web application make the web site inaccessible when any of the available system resources like disk and memory space reaches to their full utilization. When a user login to a website in anonomizing network and try to upload a file or data , the size of the file or data is checked with the said size. If the file or data is oversized then the user is detected and warned not to upload the oversized file. And if the user again try to upload the same oversized data then nymble manager identifies him and block the user for said interval of time.

2) Data Modification Attack :

Data modification attack deals with any changes made in the existing file or the data. The changes in the existing file includes deletion , insertion and alteration of data in the file. The attacker can change or modify the data after reading it without the knowledge of the sender or receiver.

Equation:

$$f(dma) \Rightarrow U_{data} \in U_i$$

Where

- f(dma) is function to identify DMA attack
- U_{data} -> Users Uploaded data
- U_i ->Respective User

Data modification attack on web application allow the user to login to the website and upload file to public space. That user are allowed to see and modify the data of that particular file. However , if user try to modify others data then that is considered as misbehavior of user and warning is given. If the user again try to modify same file then nymble manager identifies him and block the user for said interval of time. Thus the file is allowed to be modified only by the user who uploaded the file to the public space.

3) Password Based Attack :

Password-based access control is the most common security plan that is provided to the network. In this the access right of the user to network is determined by its username and password . However, the attacker may have the same rights of real user after finding a valid user account. Hence the attacker can obtain lists of valid user, computer names, network information and modify server and network configurations and even modify,reroute or delete your data.

Equation:

$$f(pwd) \Rightarrow U_{pwd} \in U_i$$

Where

- f(pwd) is function to identify Password attack
- U_{pwd} -> Users Password
- U_i ->Respective User

Password attack on web application allow the user to login the website and if the user is trying to login with other users password continuously that is the user password does not belong to the user name then the warning is given 3 times. Even though the user is trying to login again and again then nymble manager identifies him and block the user for said interval of time. At the same time a mail is sent with an alternate password to the valid user and allow him to reset the password.

4) Application Layer Attack :

An Application-layer attack targets application servers, gains control of your application or system or network, introduces virus program **,** abnormally terminate your data applications or operating systems and disable other security controls. In this the data or operating system can be read, added, deleted or even modified. A sniffer program is also introduced which analyze and gain information about network in order to corrupt or crash your system.

Equation:

$$\textbf{Set } V = \{v1, v2, v3 \ldots\ldots v_n\}$$
$$\textbf{f(apl) => } UP_{data} \in V$$

Where

- Set V is Virus definition database
- f(apl) is function to identify application layer attack
- UP_{data} -> Uploading data

An Application-layer attack on web application allows the user to login and upload the file on the network. However, If user try to upload some malicious file(like virus) then the user is detected and warning is given. If user again try to upload same malicious file again then Nymble manger identifies him and block the user for said interval of time.

5) Unblocking User

The misbehaving user is unblocked after some threshold time for ensuring dynamism[6] since resources such as IP addresses can get reassigned and it is undesirable to blacklist such resources
indefinitely, and it ensures forgiveness of misbehavior after a certain period of time.

Equation:

$$f(unb) \Rightarrow (t_c - t_b) > T$$

Where

- f(unb) is function to identify unblocking user
- t_c -> Current time
- t_b-> blocked time
- T-> threshold time

Algorithm:

Input: Blocked time as t_b, Current time as t_c and Threshold time as T
Output: Unblocked Blocked State
Step 0: Get tb and t_c and T
Step 1: if ($t_c - t_b$) >T
Step 2: Get Pseudonym
Step 5: Add Pseudonym in unblocked list
Step 6: Update User's state
Step 7: return user state

IV. SECURITY ANALYSIS

A) Anonymous authentication

Nymble enables the property of providing security to the user by keeping the user anonymous, that means the IP address of the user is hidden. Hence the user is identified by pseudonym without revealing its IP address and other information to the server.

B) Backward unlikability

Nymble cannot determine whether two or more different tickets come from the same user or different users. From the nymble tickets the user address cannot be traced.

C) Subjective blacklisting

This property enables the honest server to block the misbehaving user exclusively. If a server complaints about the misbehaviour of the user, then the complaint is processed and the user is blocked for a certain period of time.

D) Fast authentication speed

It provides faster authentication and speeds up the security process.

E) Rate limited anonymous connection

This property enables any honest user to prevent any user from establishing connection with it, when the same user tries to connect to it more than once in same single time period.

F) Revocation auditability

This property enables the user to check their status before revealing any connection through nimble tickets. It allows user to get the status whether its blacklisted or not. Hence it keeps the user aware about its connection status.

V. CONCLUSION

Finally we have proposed and built a credential system called Nymble that adds an additional layer of security to any publicly known anonymizing network such as Tor. In nymble system servers blacklist misbehaving users but also maintain their anonymity and privacy i.e., the IP-address of the user is not known to the server. Behaving users enjoy anonymity while blacklisted users are not allowed future connections for a duration of time while their previous connections remain unlinkable. Nymble also provides websites the power to define their own definition of "misbehavior". Nymble is based on Pseudonym Manager (PM) and the Nymble Manager (NM) where the PM is responsible for pairing a user's IP address with a pseudonym generated based on the user's IP address and the NM pairs a user's pseudonym with the target server. If and when a user misbehaves, the server may not realize it for some amount of time and may not report it until a later time period. However, after receiving a linking token the server is able to block all future connections until the next linkability window. Nymble also provides anonymous authentication, fast authentication speeds, subjective blacklisting, backward anonymity and revocation auditability. This method is practical, effective and efficient to the needs of both users and services. Dynamism and Forgiveness is the important characteristic of the Nymble system.

We hope that our work will motivates the need for security in anonymous network and will increase the mainstream acceptance of anonymizing networks such as Tor, which has thus far been completely blocked by several services because of users who abuse their anonymity.

FUTURE WORK

The proposed system is dealing with Pseudonym Manager, Nymble Manager and server on the local anonomizing network. In future this work will be enhanced to work on a remote machine. This work can also be extended into a multiple rounds of pseudonym construction in which the Pseudonym Manager participates in multiple rounds of communication with the user. This extension may adds one more layer of security to the system.

ACKNOWLEGMENT

The authors are grateful for the suggestions and help from prof. V.K.BHUSARI and wish to thanks her for helping in the early stages of prototyping.

REFERENCES

[1] web application : http://en.wikipedia.org/wiki/Web_application [Online; accessed 12-jan-2014].

[2] Roger Dingledine, Nick Mathewson, and Paul Syverson. “Tor: The Second-Generation Onion Router”, *Usenix Security Symposium*, pages 303–320, August 2004.

[3] Giuseppe Ateniese, Jan Camenisch, Marc Joye, and Gene Tsudik. “A practical and provably secure coalition-resistant group signature scheme”, Mihir Bellare, editor, CRYPTO, volume 1880 of LNCS, pages 255–270. Springer, 2000.

[4] Aggelos Kiayias, Yiannis Tsiounis, and Moti Yung. “Traceable signatures”. In Christian Cachin and Jan Camenisch, editors, EUROCRYPT, volume 3027 of LNCS, pages 571–589. Springer, 2004.

[5] the onion router(tor): http://en.wikipedia.org/wiki/Tor_(anonymity_network) [Online; accessed 12-jan-2014]

[6] 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.

