Implementation of Honeypot using Kerberos Authentication

[1]Aishwarya.S, [2]SurabhiKinariwala, [3]Sujatha.G
SRM University, B.Tech-IT
Asst. Professor-SRM University

Abstract:
Implementation of Honeypot with Kerberos authentication provides a 3-step authentication process to classify white hat and black hat users. It uses an Authentication System, Ticket Granting System and A Server with symmetric key Cryptography to authenticate a client. It uses double encryption in the exchange of tokens and requires every client to authenticate with each of the three systems before accessing the service it requested. Any user who fails to authenticate will be identified as a hacker but with a negative token without the knowledge of the Client. At decryption only the three Systems can identify a user with a positive or negative token. And hence identify an external or internal hacker.

Once authenticated with a negative token the hacker's interactions are monitored and logged. In gathering this information the network administration can devise efficient warning systems to notify when the system is breached by an attacker.

In using Kerberos with honeypot, we overcome the restricted identification provided by IP Address and comparisons of network Services. The development of the system can be done with a platform-independent technology and can accommodate large and small traffic supporting systems.

Keywords: Kerberos, Honeypot, 3-Step Authentication, Positive and Negative Tokens, Hacker, Warning System, Platform-independent.

I. Introduction

In the advanced network of communication around the world, the good co-exists with the bad. The flow of information among the right, triggers the threat of theft by the wrong. How do you prevent it? In a network of systems, how do you identify the single untrustworthy coexisting negative?

Today's world is increasingly relying on computer networks. The increase in the use of network resources is followed by a rising volume of security problems. New threats and vulnerabilities are discovered every day and affect users and companies at critical levels, from privacy issues to financial losses. Monitoring network activity is a mandatory step for researchers and security analysts to understand these threats and to build better protections.[3]

Honeypots are closely monitored network decoys serving several purposes: they can distract adversaries from more valuable machines on a network, they can provide early warning about new attack and exploitation trends and they allow in-depth examination of adversaries during and after exploitation of a Honeypot.

Honeypots are a highly flexible security tool with different applications for security. They don't fix a single problem. Instead they have multiple uses, such as prevention, detection, or information gathering.

Honeypots all share the same concept: a security resource that should not have any production or authorized activity. In other words, deployment of honeypots in a network should not affect critical network services and applications. Ahoneypot is a security resource who's value lies in being probed, attacked, or compromised.[4]

This paper provides a simple cognition of two advanced technologies to thwart attackers and prevent any harm to a communication system

II. Related Work

The existing technologies on honeypot focus on gathering of data and their analysis to form an effective intrusion detection system. However in the ideal implementation of honeypot they employ techniques not much secure for authentication.[1]

The latest employed methods include classifying users from hackers based on authenticated or unauthenticated IP Spaces—this authentication may allow those with compelling IP-Spoofing attacks in a honeypot employed on a very restrictive network. [2]

One of the other typical techniques in honeypot authentication involves the comparison of incoming requests with patterns of existing viruses or worms. The discordance in these systems feature the need for a large system of aggregated information on Viruses and Worm patterns. With the increase in opportunities for mutant and polymorphic viruses, the domain of effectiveness of this system is dimunitive.

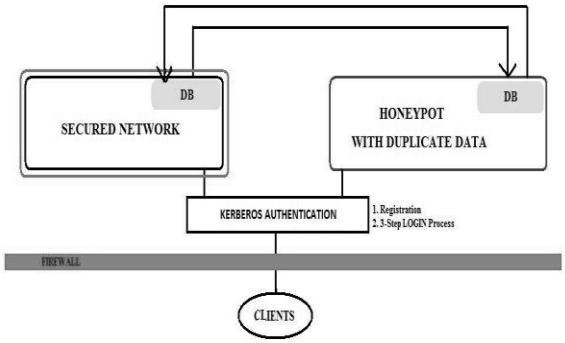
And they depend on kernel and Operating Systems in a convoluted loop making them less portable and function to a reduced potential.

A major concern is that if a honeypot is compromised, the attacker may attempt to use this as a stepping stone to damage or take over other systems. Ideally the honeypot should use several mechanisms to prevent this, and the operator should pay close attention so no harm comes to innocent third-parties. [1]

III. Proposed Technology

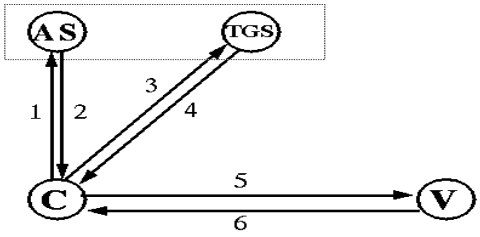
This paper proposes the integration of a 3step-Kerberos Authentication system with the implementation of honeypot to ably differentiate hackers and users.

Without knowledge of the identity of an individual requesting an operation, it is often difficult to decide whether the operation should be allowed. Traditional authentication methods are not suitable for use in computer networks where attackers can monitor network traffic and intercept passwords. The use of strong authentication methods that do not disclose passwords is imperative. The Kerberos authentication system supports strong authentication on suchnetworks.[5] The system suggests that the authentication system be placed within the firewall to avert outbound attacks.



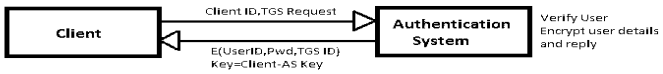
The Kerberos Authentication System uses a series of encrypted messages to prove to a verifier that a client is running on behalf of a particular user. The Kerberos protocol is based in part on the Needham and Schroeder authentication protocol, but with changes to support the needs of the environment for which it was developed. Among these changes are the use of timestamps to reduce the number of messages needed for basic authentication, the addition of a ``ticket-granting" service to support subsequent authentication[6]

The Kerberos authentication consists of Registration and a 3-step login process with 3 distinct systems in communication with each other using asymmetric cryptography techniques. The three systems involved in Kerberos are Authentication System, Ticket Granting System and Server. Each Server availing multiple services is listed with a Ticket Granting System and each Ticket Granting System with an Authentication System. Each of the subordinate will allow access to its inferiors only if authenticated by the system above it.



1. as_req: c, tgs, $time_{exp}$, n
2. as_rep: $\{K_{c,tgs}, tgs, time_{exp}, n, \ldots\}K_c$, $\{T_{c,tgs}\}K_{tgs}$
3. tgs_req: $\{ts, \ldots\}K_{c,tgs}$ $\{T_{c,tgs}\}K_{tgs}$, v, $time_{exp}$, n
4. tgs_rep: $\{K_{c,v}, v, time_{exp}, n, \ldots\}K_{c,tgs}$, $\{T_{c,v}\}K_v$
5. ap_req: $\{ts, ck, K_{subsession}, \ldots\}K_{c,v}$ $\{T_{c,v}\}K_v$
6. ap_rep: $\{ts\}K_{c,v}$ (optional)

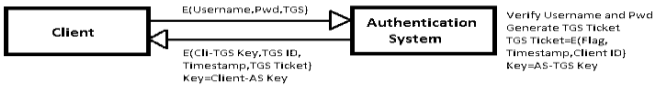
If a Client wishes to use a particular service, he has to register himself with the corresponding Authentication System after proving that he is an authentic and trustworthy user. Upon successful registration the client receives an encrypted token containing the login information of itself from the authentication system. The key used in this encryption is a function of the password that the client provided to authenticate itself. Using the same the client decrypts and retrieves the information.Though conceptually, Kerberos authentication proves that a client is running on behalf of a particular user, a more precise statement is that the client has knowledge of an encryption key that is known by only the user and the authentication server.
Because the ticket is encrypted in the server key, known only by the authentication server and intended verifier, it is not possible for the client to modify the ticket without detection.



REGISTRATION

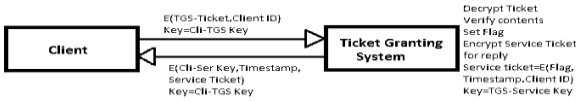
After successful registration the client has to provide these credentials to authenticate itself with the system. The base concept of Kerberos with honeypot is to be able to generate tokens to provide to hackers. However the tokens generated should substantially differentiate him in the eye of the system while being non-deceptive to the hacker. Another fulcrum of this concept is to provide double encryption of the tokens from the system to the client.

In the first step of login the client accesses the Authentication system with its credentials. The authentication system verifies the credentials and generates a positive token. This token consists of the one time key between client and TGS, the TGS ID and the secret key(between AS-TGS) encrypted TGS Tickets consisting of the client information. The authentic client is able to decrypt this information and the hacker is not.



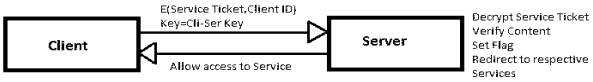
LOGIN -Step 1

However after successful decryption of the token the information from it is further again provided at the TGS. The TGS verifies the information submitted with the information , it secrets decrypts from the TGS ticket, if they match it generates and provides a similar positive token to the client if not it provides a negative token that only the server can differentiate. This token contains a service ticket encrypted with the shared key between the TGS and the Server. The token contains the one time secret key to be used between the client and the Server, the server ID and the Service ticket encrypted.



LOGIN-Step 2

The same process of decryption with client and comparison of information with submitted information takes place at Server, if this fails or if any of the previous authentication provided by the other two systems indicate that the client is a hacker then the server redirects the client to the honeypot otherwise the respective secure system.



LOGIN-Step 3

Each entry to the honeypot is registered in a separate database to be used for further development of an efficient Intrusion detection system. The data kept at the honeypot are replicas of those in the database. However highly secure files like the password files are duplicated with false passwords to be presented as an authentic system to the hacker. Each of these files is also stored in a database different from that of the secure system.

Any of the information the hacker tries to access is recorded along with an efficient timestamp and stores in a separate database, if this database exists on the honeypot system then for security purposes they need to be updated regularly with the secure system in order to prevent any changes to the information collected.

This category of honeypot comes under the low interaction honeypot that can be used for production as well as research purposes. The information collected from them can be used by the administrator to develop an efficient alarm system to notify in case of breach by hackers.

IV. Conclusion

With the effect of a 3-step authentication process, we can help prevent the defects of existing systems. It efficiently camouflages the existence of honeypot

The presence of this authentication also provides classification among true-users and internal hackers. Those that misuse their available access to that of a level higher than them.

It can be used to develop an intrusion detection system that can identify hackers from the details collected from the low interaction honeypot

This system facilitates its development in platform-independent technologies like ASP.Net and JAVA to support webserver and closed networks.[7]

V. References

1. Jamie Riden and Christian Seifert, 2002,http://www.symantec.com/connect/articles/guide-different-kinds-honeypots
2. Niels Provos, 2004, http://niels.xtdnet.nl/papers/honeyd.pdf
3. Robin G Berthier,Univeristy Of MaryLand, 2011Advanced Honeypot Architecture for Network Threats Quantification (Computer/Networking Project)
4. Article in http://www.honeypots.net/
5. Article in http://www.kerberos.info/
6. B Clifford Newman and Theodore T, http://gost.isi.edu/publications/kerberos-neuman-tso.html
7. V Maheshwari, PE Sankaranaryanan, Chennai, Capturing and Analysing the Intruder attacks using a platform independent Honeypot