International Journal of Computer Architecture and Mobility
(ISSN 2319-9229) Volume 1-Issue 6, April 2013

OBSCURE SURVEILLANCE AND SERVER MONITORING

Neelam Singh Parihar
Neelam.14parihar@yahoo.com
MIT Bhopal M.P

Deepa Verma
deepaverma123@gmail.com
MIT Bhopal MP

Parekh Ankita Arvindbhai
hardiksharma01@gmail.com
MIT Bhopal MP

ABSTRACT

Obscure Surveillance and server monitoring is a technique to administrate a remote server and monitor the logs of client's work. It's a browseable method through which we can administrate the server from any platform. It contains a two level encryption one at the browser side and another at the backend side. In this server maintains all the details about their client's i.e. userid, password, encryption key's and log's. A firewall at the backend detects the packets and patterns and block phishing attacks. This paper proposed a technique to monitor and administrate the server from remote side through browser and from any platform.

General Terms

Authentication Scheme, Security, encryption mode

Keywords

Phishing attacks, authentication, encryption.

1. INTRODUCTION

In this technique the server has all the services provided to it and with the help of browser the server is monitored and check whether it is not suffering from any authentication break attacks which gives the intruders to enter through backside and destroy the proper functioning of the server and misuse that data through this technique the server is capable of maintaining and monitor the attacks and generate an alarm when it find something suspicious.[1]

This discusses in detail the common monitoring technique in used for 'SQL injection' technique, as it applies to the popular Microsoft Internet Information Server/Active Server Pages/SQL Server platform. It discusses the various ways in which SQL can be 'injected' into the application and addresses some of the data validation and database lockdown issues that are related to this class of attack. The paper is intended to be read by both developers of web applications which communicate with databases and by security professionals whose role includes auditing these web applications.[3]

Monitor what remote users do on your Windows Terminal Server. Audit Windows Remote Desktop sessions and Citrix shared desktop and virtual applications. Terminal Server sessions recorder that captures every user action. Monitor your employees who tele-work from home or remote-in to the office during business trips via RDP[2]. Monitor what users do on thin clients in your network without installing any software on them. Document server configuration changes by recording remote and local administrative sessions. Secure your corporate data by preventing information theft by insiders. Increase staff productivity and improve security by using this unique software from Soft Activity.[5] The software is completely invisible for monitored users. Record activity in multiple user accounts in terminal and local sessions on the server:

any platform

The software records remote terminal sessions on the Terminal Server. Multiple sessions that may be running simultaneously on the server are recorded into a single database. The software can also record local user sessions on the server, such as an administrative session Select what user accounts to record. Prevent it from recording a sys admin &rsquo;s or manager &rsquo;s account if needed View Activity Reports: Managers/Administrators can view activity reports on the server :via remote terminal session from any computer; orvia shared folders from any computer; orin a local administrative session on the server Top Programs report shows most used applications Top Websites report shows most visited websites and total time spent by users on each website Review slide show of screenshots for each user, like a recording of a security camera Text typed in all documents, web forums, chats, social networking or email sites can be seen on a keystrokes log See a history when users log in and log off from the server Search the reports

2. RELATED WORK

Many techniques has been proposed to handle the server from remote side so that the administration of any organization cannot be suffered if the administrator is not present at the place where actual server is present so it can remotely login from any system through internet and browse his server on that system so that it can monitor, analysis and reconfigure the server according to the situation. In this proposed technique administrator enters its userid and password server check the database and authenticate it, after that the administrator monitor and manage, the server contain two login system in the server the web based login and another when it enter in the administration part it again prompt for another password at the backend after the web based login to provide a secure shell, as in the case of user also the server perform the same authentication scheme to authenticate the user at the server.[7] It block the unauthorized user from accessing the server services and maintains a log of that user as an suspicious user, and another is thee firewall that detects the packet going through server and client and block the infected packet to enter in the server it recognizes, the packets and check for their validity the server generates a appropriate alarm when it find any suspicious process and block that from accessing or entering into the server for the security.

3. PROPOSED TECHNIQUE

The proposed technique consists of three phases

1. Verification Phase.
2. Encryption Phase.
3. Monitor, administration phase.

3.1 Verification Phase:

In this phase the server verify the user and his work by checking the userid and password provided by the user in the database.

3.2 Encryption Phase:

In this phase when the user has successfully logged from the web based login after for secure channel the server again asked for password from the user and authenticate the user after that the password is encrypt by the SSH(secure shell) and a secured channel is provided by the server to communicate with it.

3.3 Monitoring and administration phase:

In this phase the administrator can analysis, monitor the server processing and can manage it remotely from any system.

Resource Sharing:

In this the system can share its resources if they want any extra resource for load balancing so that the server can work properly, it removes many problems ex resource conflicts, etc.

File Transferring:

In this the system can transfer or receive the files from client to server or vice-versa, during the transferring the security of the file is maintained at the transferring channel.

Disk Management:

In this the quota is managed for each user so that the disk can be fully utilized.

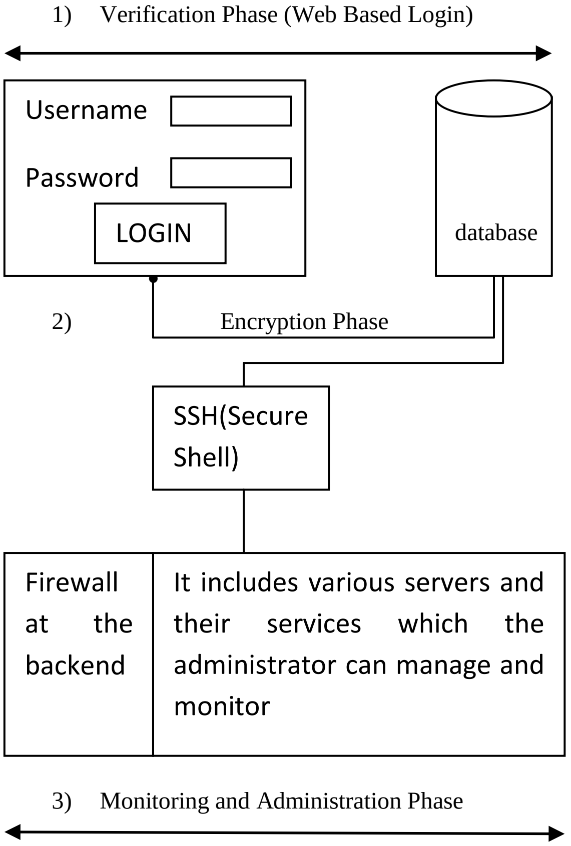
Log Management:

In this log is maintained by the server to check the user's work and monitor it's processing for security purpose.

Process Management:

In this multiple processes that are running on the server are handled so that the processes can synchronized with each other

Proposed Model


1) Verification Phase (Web Based Login)
Username
Password
LOGIN
database
2)
Encryption Phase
SSH(Secure Shell)
Firewall at the backend
It includes various servers and their services which the administrator can manage and monitor
3) Monitoring and Administration Phase

This model describes the proposed model, the first phase i.e. verification phase check the userid and password and authenticate at the web based login, it then goes to the secure shell(SSH).It again prompt for different password after that encrypt the password i.e. the client will be on any platform after that the administrator can analyze, monitor or make any changes according to the conditions, and at the third phase firewall is implemented to check the packets entering in the server and discards the infected packets.

The server maintains all the logs of the user so that the administrator can later check the malicious attacks that are done on the server and improves its security level of the server and improves the firewall settings so that new arrived patterns can be recognized next time and this entry is maintained in log field of the system if a infected packet is present among the different packets than the server first checks the allowable packet category stored in the database of the server if it follow the rule and did not contain any malicious content than it is allowed to enter in the server otherwise it is discarded immediately and type of malicious pattern is entered in the server log files.

Results

The proposed technique has taken a work on remote administration and is able to handle the server side attacks that are done by any intruders. We have made one system as a server and another as a client and provide different services for it and from client system tried to enter unauthorisely to check whether it is capable of blocking that attack or not and check the security level of the server, from client system remotely access the server services and work to manage them remotely. The idea is taken from two papers one is the authentication mechanism to prevent from SQL injection attacks and another is the self organizing method to check the packets that generate an alarm if it found anything wrong. The proposed technique is implemented on a Intel Pentium(R) Dual CPU E2180 @ 2.00 GHz, 1.00 GB of RAM using RHEL 5(Red Hat Enterprise Linux) on the server and access its services from client system that contains windows XP.

CONCLUSION

This paper proposed a technique to implement a server security and remote administration of the server to manage and monitor its services. It maintains the logs of all users with a platform independent feature, in this there is a two way security which is much effective and efficient in administration. The technique works well and reduce the maximum overhead of the administrator and provide a better management and monitoring scheme.

REFERENCES

1) C.Anley, 2002, “Advance SQL Injection In SQL Server Applications,” White paper, Next Generation Security Software Ltd.

2) Simon Hay kin, ―Neural Networks: A Comprehensive Foundation, Prentice Hall, 2nd edition, 1999

3) Kemalis,K. and T. Tzouramanis 2008.”SQL-IDS: a specification-based approach for SQLinjection detection, SAC08.Fortaleza, Ceara Brazil, ACM: pp.2153-2158.

4) C. Cowan, S. Beattie,J. Johansen, and P. Wagle 2003”point Guard: protecting pointers from Buffer Overflow vulnerabilities,” In proceedings of the 12th USENIX Security Symposiums, pages 91-104.

5) D. Larochelle and D. Evans, 2001.’Statically Detecting Likely Buffer Overflow Vulnerabilities,” In Proceedings of the 10th USENIX Security Symposium, Pages 177-190.

6) D.A Frincke, D. Tobin ,J.C. McConnell, J. Marconi and D. Polla. A framework for cooperative intrusion detection In Proc. 21st NIST-NCSC National Information Systems Security Conferences, Pages 361-373, 1998.

7) Simon Haykin-Neural Networks: A Comprehensive Foundation, Prentice Hall, 2nd edition, 1999.

8) Kohonen, T.-Self Organising Maps, Springer Series in Information Sciences, Berlin, Heidelberg: Spinger 1997.