BEST: International Journal of Management, Information
Technology and Engineering (BEST: IJMITE)
ISSN 2348-0513
Vol. 2, Issue 10, Oct 2014, 1-10

DOUBLE GUARD: DETECTING AND PREVENTING INTRUSIONS IN MULTI-TIER WEB APPLICATIONS

EKTA NAIK[1] & NILAV M[2]

[1]PG Scholar, Dr. D Y Patil School of Engineering & Technology, Kolhapur, Maharashtra, India

[2]Assistant Professor, Dr. D Y Patil School of Engineering & Technology, Kolhapur, Maharashtra, India

ABSTRACT

Internet services and applications are very useful in our day to day life like communication and the management of personal information as well as social information from anywhere and anytime. An advanced multi-tiered design is used wherein the web server runs the application front-end logic and data is outsourced to a database or file server which is used by the web services. An independent IDS would not be able to identify. This drawback has been overcome in Double guard system which utilises an IDS system that models the network behaviour of user sessions across both the front-end web server and the back-end database. This is achieved by monitoring both web and subsequent database requests. So it is possible to ferret out attacks completely. Furthermore, it quantify the limitations of any multitier IDS in terms of training sessions and functionality coverage. I have implemented Double Guard using an IIS(internet information and services) web server with MySQL. Double Guard can handle both types of attack also i.e. on Front End (HTTP) and Back end (SQL SERVER).

KEYWORDS: IIS, DDOS, Intrusion Detection System, Anomaly Detection, Web Server

Attacks

Paper submitted on January 24, 2014 for review in Computer Post graduate Conference cPGCON at Matoshri College of Engineering, Eklahere, Nasik March 28th to 29th, 2014.

Ekta Naik is a PG scholar pursuing Masters degree in Computer Engineering in Dr. D Y Patil School of Engineering & Technology.

Nilav M is Assistant Professor in Dr. D Y Patil School of Engineering & Technology. His area of interest includes data mining, information retrieval and information & network security

INTRODUCTION

To protect multi-tiered web services, Intrusion detection systems (IDS) have been widely used to detect known attacks by matching misused traffic patterns or signatures [1]. In the existing system we require different IDS one for web server and another for database server. 2 IDSes required so we need to create 2 IDSes with different prevention measure first IDS that contains prevention measure related to web server so attack should not happen on web server but some time attack happen on database server bypassing web server so for that reason need to create another IDS with prevention measure related to database server attack. We want to avoid creating 2 IDS so we are creating one Double Guard system that act as IDS and prevent both side of attack. Attack may be on web server or database server. Most of the IDS examine the attack individually on web server and database server. In order to protect multi-tiered web services an efficient system call Intrusion Detection System is needed to detect attacks by mapping web request and SQL query[7][8].

In this paper, I present Double Guard, a system used to identify attacks in multi-tiered web services. My approach can create routine models of isolated user sessions that include both the web front-end (HTTP) and back-end (File or SQL) network transactions. To achieve this, I employ a virtualization technique to assign each users web session to a dedicated container, an isolated virtual computing environment.

In the proposed system we are implementing Double Guard that handle both sides of attack. Attack may be from static web site or dynamic web site. No need to create two different IDSes for two different web site. Double Guard can handle both types of attack also we are performing encryption algorithm and DDOS attack []. We are finding IP Address of intruder.

In addition to this static website case, there are web services that permit persistent backend data modifications. These services, which we call dynamic, allow HTTP requests to include parameters that are variable and depend on user input. Therefore, our ability to model the causal relationship between the front-end and back-end is not always deterministic and depends primarily upon the application logic. For instance, we observed that the back-end queries can vary based on the value of the parameters passed in the HTTP requests and the previous application state. Sometimes, the same applications primitive functionality (i.e., accessing a table) can be triggered by many different web pages. Therefore, the resulting mapping between web and database requests can range from one to many, depending on the value of the parameters passed in the web request.

RELATED WORK

A network Intrusion Detection System (IDS) can be classified into two types: anomaly detection and misuse detection. Anomaly detection first requires the IDS[1][2][3] to define and characterize the correct and acceptable static form and dynamic behaviour of the system, which can then be used to detect abnormal changes or anomalous behaviour. Double Guard does not have such a limitation as it uses the container ID for each session to causally map the related events, permit content modification from users, there is a direct causal relationship between the requests received by the frontend web server and those generated for the database backend.

PROBLEM DEFINITION AND DESCRIPTION

Problem Definition

The intrusion detection system which detect intrusion attacks and prevent static and dynamic website from those types of attack is called a Double Guard system. This system does not require two different IDS rather it creates only one IDS and handle both the sides of attacks [12].

Also prevention techniques are included to prevent the intrusion attacks which were not included in previously defined system of Double Guard.

Problem Description

In multitier architectures, the back-end database server is often protected behind a firewall while the web servers are remotely accessible over the Internet. Unfortunately, though they are protected from direct remote attacks, the back-end systems are susceptible to attacks that use web requests as a means to exploit the back end. In order to protect multitier web services, an efficient system called as Intrusion detection systems is needed to detect known attacks by matching misused traffic patterns or signatures.

Algorithms

Static Model Building Algorithm

In this algorithm we are getting set of web request and generate SQL query according to web request. If user perform web request and for that web request SQL query is not generated then that web request mark as EQS (Empty Query Set) else generated SQL query and get result. If we got same result as expected up to threshold value then mapping is correct otherwise need more training sessions. In NMR (No match request) [1]SQL query generated without web request from user but according to SQL query action will be performed.

AES Algorithm

We are using AES encryption algorithm for encryption and decryption of data. We are already giving security to our application but not for data. By using encryption we can provide security for our data also. AES algorithm having 128 bit size. We are storing data in encrypted format. When user upload file, we get that file perform encryption by using encryption algorithm i.e. AES (Advance Encryption Standard)[4]. After that it give encrypted file and that file stored into database. When use upload file by using encryption method we encrypt that file and store into database in encrypted format but when user click on download file get that encrypted file from database perform decryption method on that file and convert it into original and readable format. Once data stored into database in encrypted format then even if database got hacked by hacker, hacker cannot understand what is that data due to encrypted format.

Explanation

Encryption

- **Input:** Attribute Value (Attr).
- Get Byte [] (B1) of that Attr.
- Generate Key ().
- Perform Encryption on B1.
- Convert B1 into string(EAttr).
- Decryption
- **Input:** Encrypted attribute value(EAttr)
- Convert EAttr into byte [](B2).
- Generate Key.
- Perform Decryption on B2.
- Convert B2 into string(DAttr).

Deliverables

- *Static Wiebsite*

  In static website we can allow user to upload and download files from web server.

- *Dynamic Website*

In dynamic website we can allow site visitors to read, post, and comment on articles. Site visitor view blogs by category wise and by calendar wise. Dynamic website like blogs requires regular updating of database. Admin can perform all activities of site visitor. Admin also perform add, edit, delete new category, blog and comment.

- How Attack Occur

In this we are showing how attacker can attack to our system.

- Prevent website from attack

In this we are showing how our system can be prevent from those types of attack.

Prevention Techniques

SQL injection

We prevent SQL injection by creating stored procedure. Store procedures are saved on database side and not on client side so even hacker trying to hack system via SQL injection he/she cannot hack the system.

SYSTEM ARCHITECTURE

The system architecture of the system is shown in figure 1. User has send his request according to which the response is generated by the system.

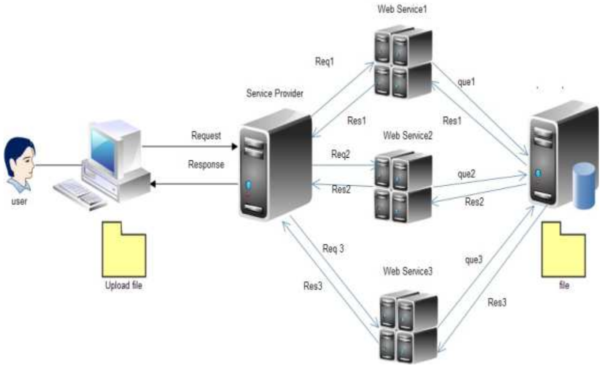


Figure 1: System Architecture

Here in dataflow diagrams the flow of data is shown form user to database. When user login with his/her correct credentials the login results are displayed. When he/she submits the request for required data then data will be displayed.

Techniques and Approaches

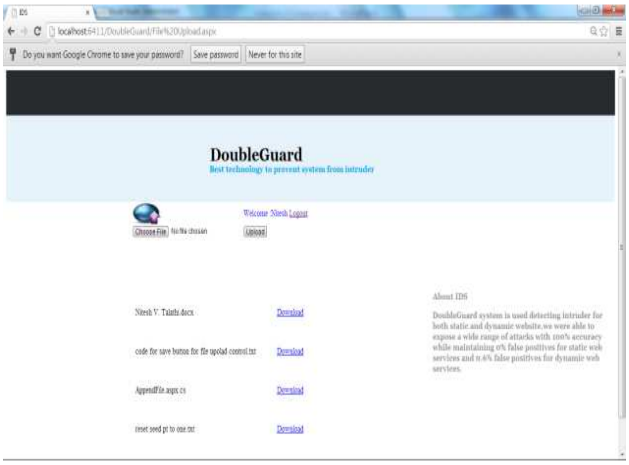
It is not possible to create 250 instances so I am creating only one instance. I am creating two applications one for static and one for dynamic website and I am considering only one pc as a web server.

WORK DONE

I representing the input data to static and dynamic web site and the environment used for implementation.

Input Data:

For static web site I am showing the static web page. this page contains the static environment like upload button, download button.



localhost:6411/DoubleGuard/File%20Upload.aspx

Do you want Google Chrome to save your password? Save password Never for this site

DoubleGuard
Best technology to prevent system from intruder

Choose File No file chosen

Welcome Nitesh Logout

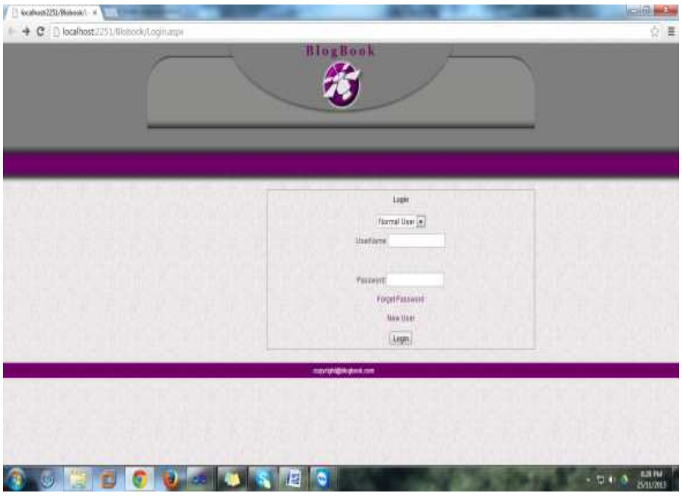
Upload

Nitesh V. Talathi.docx	Download
code for save button for file upolad control.txt	Download
AppendFile.aspx.cs	Download
reset seed pt to one.txt	Download

About IDS

DoubleGuard system is used detecting intruder for both static and dynamic website,we were able to expose a wide range of attacks with 100% accuracy while maintaining 0% false positives for static web services and 0.6% false positives for dynamic web services.

And Dynamic web site contain the option for login, blog writting, commenting, reading etc.



localhost:2251/Blobook/Login.aspx

BlogBook

Login

Normal User

UserName

Password

Forget Password

New User

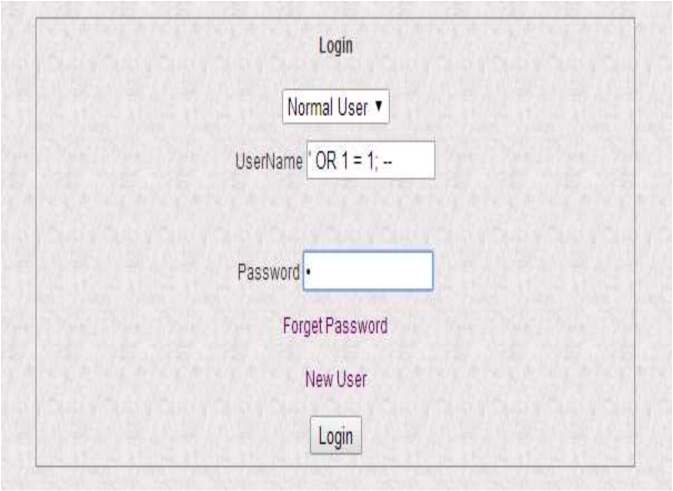
Login

copyright@blogbook.com

Detecting Intrusions Attacks

While detecting intrusions in multitier web applications following things may happen.

- Login.aspx



Login

Normal User

UserName ' OR 1 = 1; --

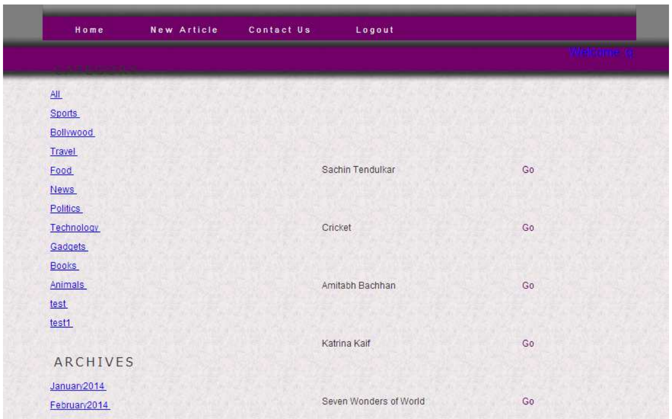
Password •

Forget Password

New User

Login

- Home.aspx



If we login with the user name " ' OR 1=1; -- " with any random password then we will get the previous user's account. It is called the SQL injection attack.

RELEVANT MATHEMATICS

Set Theory Analysis

A] Identify the double guard

A = fd1, d2g

Where D is main set of double guard like d1, d2

D1 = static web site

D2= dynamic web site

B] Identify the number of request

R= fr1, r2, r3.g

Where R is main set of number of request like r1, r2, r3

C] Identify the Numbers of query

Q= fq1, q2, q3.g

Where Q is main set of query q1, q2, q3

D] Identify the Number of user.

U= fu1, u2, u3.g

Where U is main set of requested all attribute in emergency u1, u2, u3

E] Identify Mapping.

M= fm1, m2, m3.g

Where M is main set of Mapping m1, m2, m3

F] Identify the processes as P.

P= fSet of processesg

P = fP1, P2, P3,P4g

P1 = fe1, e2g

P2= ff1, f2, f3, f4, f5, f6, f7g

P3= fg1, g2, g3, g4, g5, g6g

Where

Static Web site

fe1= upload data on serverg

fe2= download data from server

Dynamic web site

ff1 = User view blogs

ff2 = User upload new blogs

ff3= user give comment on blogs

ff4 =user view blogs by calendar and category wise

ff5 = admin add new category

ff6 = admin update comment, blog, category

ff7 = admin delete comment, blog, category

Mapping

fg1 = Get User Web request

fg2 = Query generate according to web request

fg3= Map web request and sql query

fg4 = Mapping perform up to threshold value

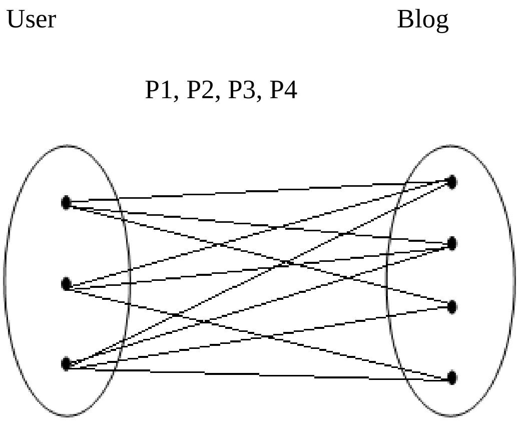
fg5= if mapping done as expected it is normal user

fg6= if mapping not done as expected it is abnormal user

Relationship

Static Website:

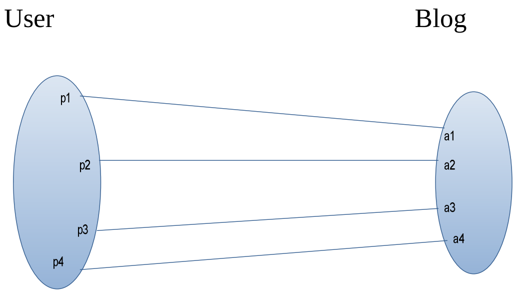
- User may upload multiple files
- One file uploads by multiple users



Many to Many Mapping

Dynamic Web Site

- User may upload blogs.

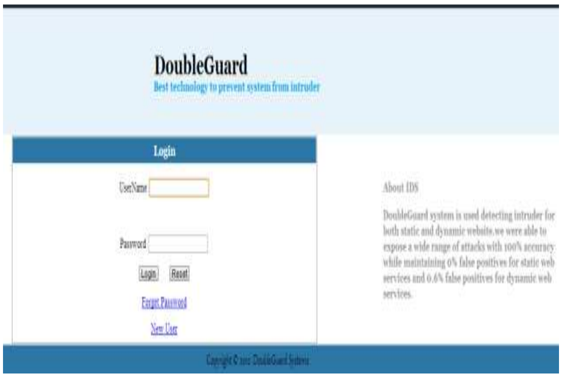


One to one Mapping

RESULTS OF THE PRACTICLE WORK

When the static web page is used,

We can upload or download the file.

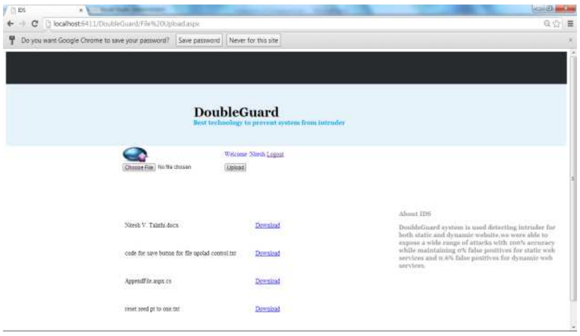


About IDS

DoubleGuard system is used detecting intruder for both static and dynamic website.we were able to expose a wide range of attacks with 100% accuracy while maintaining 0% false positives for static web services and 0.6% false positives for dynamic web services.

Copyright © 2012 DoubleGuard Systems

Loin page.



Do you want Google Chrome to save your password?

DoubleGuard
Best technology to prevent system from intruder

Nitesh V. Talathi.docx	Download
code for save button for file upolad control.txt	Download
AppendFile.aspx.cs	Download
reset seed pt to one.txt	Download

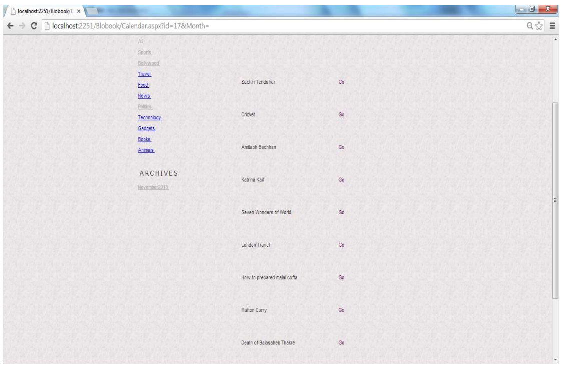
About IDS

DoubleGuard system is used detecting intruder for both static and dynamic website.we were able to expose a wide range of attacks with 100% accuracy while maintaining 0% false positives for static web services and 0.6% false positives for dynamic web services.

File upload or download

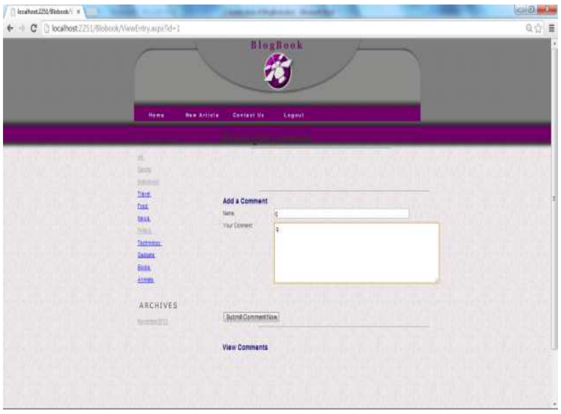
When the dynamic web page is used,

We can create our own account, can login with the respective credentials, can write a blog, comment on the blog, can read the comments.



Sachin Tendulkar	Go
Cricket	Go
Amitabh Bachhan	Go
Katrina Kaif	Go
Seven Wonders of World	Go
London Travel	Go
How to prepared malai cofta	Go
Mutton Curry	Go
Death of Balasaheb Thakre	Go

Blog Page



Add a Comment

Name: q

Your Comment: q

Comment page.

CONCLUSIONS AND FUTURE WORK

I presented an intrusion detection system that builds models of normal behaviour for multi- tiered web applications from both front-end web (HTTP) requests and back-end database (SQL) queries. Unlike previous approaches that correlated or summarized alerts generated by independent IDSes, Double Guard forms a container-based IDS with multiple input streams to produce alerts. Such correlation of different data streams provides a better characterization of the system for anomaly detection because the intrusion sensor has a more precise normality model that detects a wider range of threats. Rather it also can prevent the web applications from intrusions.

REFERENCES

1. Meixing Le, Angelos Stavrou, Brent ByungHoon Kang. ”DoubleGuard: Detecting Intrusions In Multi-tier Web Applications” IEEE transaction on dependable and secure computing vol.9 no.4 year 2012
2. Openvz. http://wiki.openvz.org.
3. Virtuozzo containers. http://www.parallels.com/products/pvc45/.
4. Linux-vserver. http://linux-vserver.org/.
5. B. Parno, J. M. McCune, D. Wendlandt, D. G. Andersen, and A. Perrig. CLAMP:
6. Practical prevention of large-scale data leaks. In IEEE Symposium on Security and Privacy. IEEE Computer Society, 2009.
7. http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2010-4332.
8. http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2010-4333.
9. Common vulnerabilities and exposures. http://www.cve.mitre.org/.
10. Five common web application vulnerabilities.http://www.symantec.com/connect/articles/five- common-web-application-vulnerabilities.
11. Udaya Kiran Tupakula Vijay Varadharaj an, A Practical Method to Counteract Denial of Service Attacks 2003.Stavrou, G. Cretu-Ciocarlie, M. Locasto, and S. Stolfo. Keep your friends close: the necessity for updating an anomaly sensor with legitimate environment changes. In Proceedings of the 2nd ACM Workshop on Security and Artificial Intelligence, 2009.
12. D. Wagner and D. Dean. Intrusion detection via static analysis. In Symposium on Security and Privacy (SSP 01), May 2001.
13. M. Christodorescu and S. Jha. Static analysis of executables to detect malicious patterns.
14. P. Vogt, F. Nentwich, N. Jovanovic, E. Kirda, C. Krugel, and G. Vigna. Cross site scripting prevention with dynamic data tainting and static analysis. In NDSS 2007.
15. R. Sekar. An efficient black-box technique for defeating web application attacks. In NDSS. The Internet Society, 2009
16. V. Felmetsger, L. Cavedon, C. Kruegel, and G. Vigna. Toward Automated Detection of Logic Vulnerabilities in Web Applications. In Proceedings of the USENIX Security