A Security Web Gateway Based on HTTP Reverse Proxy

Fen Yan* & Ye Wang
College of Information Engineering, Yangzhou University, Yangzhou, Jiangsu, China

ABSTRACT: In this paper, we try to solve the problems of traditional security protection technology, such as the structure coupling, the difficulty of module expansion and the performance bottleneck. A new web security gateway solution based on HTTP reverse proxy server is proposed and the related system design and implements of key technologies are introduced. We provide a new concepts that called Security Detection & Validate Chain (SDVC), based on SDVC and its hierarchical tree structure, and we can prevent network from attacking on the extensible way. This architecture can effectively resolve the difficult extending problem of the traditional web security protection problems. The web security gateway also can filter users by black & white list, and can restrict the resource access which needs to be protected, as well as matching & preventing many kinds of network attacking based on rules.

Keywords: web security gateway; reverse proxy; access control; security detection & validate chain

1 INTRODUCTION

As the age of Internet plus is coming, more and more Internet applications are constantly emerging. The design flaws of Internet protocol can cause many security problems. By the complex environment of real network, the application developers have to take too much time and energy to solve such security problems, or even though most security problems can be solved on web application layer by our efforts, most of them do not focus on application itself.

Most of the existing solutions are proposed without practical application. For example, Atul S. Choudhary et al. designed the malicious code injection detection tool [1], and San-Tsai Sun et al. designed a solution to detect and prevent SQL injection attacks independently of the application [2]. The researches of web security protection conducted by Dong Shen et al. is based on NDIS [3], Deng Liwu et al. designed protection system for particular database application to prevent SQL injection attacks [4]. Although these researches have different emphases, but they still have some common issues under the effective protection, e.g. the research in literature [1] and literature [4] try to separate the pattern-driven detection module from specific application to anti specific attack efficiently, but its detection module couples with web server software easily. The researches about detection and protection module in literature [2] and literature [3] separates from web application further and makes it as a part of operation system, but it couples with web server operation system. It may make the whole server become unstable if it is configured improperly or the deficiency involves in the detection and protection module. In addition, with the development of Internet, the type of the attacking from network becomes more complex penetration attack with a variety of methods from simple specific attack, which also requests the protection system to offer good expansibility for defense.

HTTP agent technology is usually a means for users to take the hidden identity, or need to access the Web application hosts which proxy servers can access but their hosts cannot reach. In commonly, the proxy server is constructed in client, but following the practical applications, and the architecture in server called reverse proxy server has appeared [5]. The reverse proxy server is transparent for users, so they cannot feel the existence of it. But all the accesses to the real web resource must pass through the reverse proxy server. Scholars both at home and abroad obtain some achievements on research in the aspect of network performance. One of the mature and high performance products which based on reverse proxy is Content...
Delivery Network (CDN). Its design is a kind of architecture solution which can improve the performance and throughput, so based on reverse proxy technology, the stable and efficiency of Web security system can be enhanced [6]. The research on the web security protection which uses reverse proxy technology is less currently. Guangyong Zhen et al. try to use reverse proxy software to protect Web server [7], but it’s difficult to take action of correspondence steps for different attack methods because it is implemented by the rules’ configuration on Squid software. The security gateway technology can offer functions like identification and filtration of traffic on application layer. It can implement firewall isolation between web application and incoming requests via configuration. Through flexible configuration, the security gateway can recognize various attack types from network.

The research is based on previous research on the web defense and puts right the weaknesses in real application which are discussed above. We design a low coupling, expansibility, reliable and high performance web security gateway systems based on HTTP reverse proxy. The major functions of this system are filtering and intercepting unauthorized or malicious requests from the users’ traffic, and do not affect the access of web application. In this ways the separation architecture based on reverse proxy solves the coupling with web server problem, as well as the difficulty of deployment and the performance bottleneck and weakness in disaster recovery problems. In order to defense various attacks, we propose Security Detection & Validate Chain (SDVC), like tree hierarchical structure for analysis to anti-attack in extensible way. And it provides a solution to solve the difficult problems about expansibility in traditional web security protection.

2 SYSTEM OPERATION ARCHITECTURE

The existing web security protection technologies can be classified into integrated architecture and separated architecture [8].

The integrated architecture mainly represents the security module and the web server stay in same host. The structure is shown in Figure 1. It can be deployed simply in single point and require no additional investments for hardware, which can process small traffic high-efficiently because the security module and web server are integrated together. However, in this structure, different security module should be developed for different web server, which is especially specified for specified operation system because the development is affected by environment. For these reasons, it’s difficult to deploy in a large scale and even requires extra privileges for access the web server. Additionally, in high request and high concurrency, the processing efficiency of the security module is easy to be affected. And the crash of the security module will also affect the normal operation of the entire Web server.

![Figure 1. Integrated web security system.](image1)

The separated architecture mainly represents the security module independent of web server and become new security system. The structure is the same as shown in Figure 2. The new system is also independent of original web server and change configuration is optional, which is flexible because there is no need to change the web server’s configuration. For multiple web server clusters, the separated system can be deployed easily and multiple security systems can be established in distributed cluster. It can improve the whole throughput rate and ability of disaster recovery without considering the factors of web server operation systems. In addition, it is easy for related software development. The disadvantage is that most separated systems need more hardware facilities, so the cost of the investment may increase.

![Figure 2. Separated web security system.](image2)

By comparing these two architectures, the separated architecture not only have good general feature but also can implement distributed load balance or disaster recovery deployment easily. The web security gateway is a new server protection mechanism in separated web security systems and gains a promising expectation [9]. In the next section we use the separated architecture model to design and implement the web security gateway based on reverse proxy server. The new Web application security protection system is put forward, which mainly focuses on solving the problems of high coupling degree, extending difficulties and poor stability of traditional security protection.

3 SYSTEM DESIGN

3.1 System model and mechanism of implementation

The HTTP reverse proxy server plays as man-in-the-middle between users and web server side, it acts dual roles as web server and user. The reverse proxy server acts web server role for user and also acts user role for web server. All of the incoming requests from
user in the client side are first received by reverse proxy server, which are not sent to server directly. The main job of reverse proxy server is to forward these requests to the right web server, receive the responses and forward them to user. To implement the security functions for reverse proxy server, we design the Requests Analysis & Filter Module. The module is located at the processor’s position which processes the incoming requests. The integration and implementation of the web security gateway is based on reverse proxy server, as shown in dotted box Figure 3.

As shown in Figure 3, the users send requests and get resources from target web server via client (e.g., Web Browser) perform as follows:

1. Translate the request domain name to corresponding IP address of reverse proxy server (RPS) by query DNS.
2. The client establishes a TCP connection with reverse proxy server by IP address that is resolved through step (1).
3. After a connection is established, the client sends HTTP requests to reverse proxy server to get resources that want.
4. The HTTP requests pass through the inspection of Request Analysis & Filter (RAF) module and decide to forward to reverse proxy server or not.
5. According to the request message, the reverse proxy server gets the web server IP address by query configuration database.
6. The reverse proxy server establishes a TCP connection with target web server.
7. Forward the HTTP request message to the web server after the connection is established.
8. The Web server sends the response (also includes resources that request) to the reverse proxy server.
9. The reverse proxy server forwards this response to the final user.

3.2 Reverse proxy server

The new security system is independent of web server because the system uses a separate architecture. By the deployment of reverse proxy server, it gets the advantage of physical isolation from original web application and decreases the unreliable by application coupling to the largest extent. The practical applications prove that reverse proxy server still keeps higher reliable and processes when it faces the high traffic and concurrence by using load balance technology [10]. The web security protection system which is built on the basis of reverse proxy system inherits the features like low coupling, reliable and high performance.

To implement functions of reverse proxy server, the dual roles of it must be clear. To the request client, the reverse proxy server is web server side, but to the web server, the reverse proxy server is client. It requires the reverse proxy server can transform identity in different scheme and implements features about the dual requests receive and send to finish the whole business transaction. The implement model we designed is shown in Figure 4.

This architecture belongs to classic C/S pattern and the major functions include:
1. Receive user (client) requests as server side.
2. Forward user requests to web server side as client.
3. Receive the responses from web server as client.
4. Forward responses from web server to the final user as server side again.

3.3 Request Analysis & Filter Module

The RAF module is an important component for processing incoming requests of reverse proxy server. This module works as filter incoming requests and intercepts illegal requests in order to avoid threats to web host. The design is shown in Figure 5. Following the work flow of the reverse proxy server, the best position of RAF module is at stage of receiving client requests. This module includes sub modules like Black & White List, Access Control List and the SDVC. Among them the sub module of Black & White List filters the request users, intercepts user to
access web server in black list, and accepts the users in white list. This sub module is the first stage that RAF module processes the requests from users. The sub module of Access Control List reviews the user permissions of resource that request and intercept the unauthorized access. This sub module is the second stage of processing users’ requests. At last the HTTP attack analysis and filter is finished by the SDVC.

3.4 The sub module of SDVC

The SDVC that we proposed is the core of RAF module. The attacks from network are various or in variant types. It requires the web security gateway to recognize the possible network attack flexibly, so extensibility is one of the most important requirements in the design of web security gateway. For the expansibility, the design of detection and validation is based on separated components. For different scene requires, the SDVC consists of a logical combination of configuration file. The requests from each user are verified by the SDVC, and the requests will be filtered if the validation fails. The design of SDVC refers to the assembly line mode of factory in our real life, and has realistic meaning. The modularization design avoids coupling, gains extensibility, and enhances configuration to a certain extent.

In accordance with the way of working of the SDVC, it is further divided to Incoming Validation Chain, Analysis & Processing Module, Business Logic Module, Filter & Intercept Module as shown in Figure 6.

(1) The work pattern of Incoming Validation Chain
The Incoming Validation Chain only dispatches the user requests to the right process module on the basis of the configured rules. It accepts not only the requests that user start, but also the requests that Filter & Intercept Module starts. It chooses the next node of the right Analysis & Processing Module by the decision of Filter & Intercept Module.

(2) The work pattern of Analysis & Processing Module
The Analysis & Processing Module peeks at the user requests from Incoming Validation Chain in real time and sends them to corresponding sub module for logic detection via self-analysis configuration. At last, the main Analysis & Processing Module tells the Filter & Intercept Module to take action through result of logic conjunction or disjunction.

(3) The determination of Business Logic Rules
The comprehensive judge of Analysis & Processing Module mainly depends on Business Logic Rules. The tree structure will be made by well configured Business Logic and Sub Analysis Module, the Business Logic is the trunk or branch, and the Sub Analysis Module is leaf node at the end of branch. The final result about incoming requests on the basis of the corresponding leaf nodes’ analysis and detection from detect different path of tree.

(4) The work pattern of Filter & Intercept Module
The Filter & Intercept Module processing requests from user in different ways on basis of the result that Analysis & Processing Module make. It will let requests pass to the next Analysis & Processing Module if legal, otherwise illegal result from the Analysis & Processing Module will put requests into intercept queue and show message to user finally to avoid long time wait.

4 THE IMPLEMENTS OF KEY TECHNOLOGY

This web security gateway was written by Python language and the program structure, as shown in Figure 7. The next sections introduce the implements of key technology.

4.1 The implements of reverse proxy server
To ensure the high performance and reliability to process user’s request for reverse proxy server, we choose
the multi-threads and multi processes mixed model in work pattern of reverse proxy server, because it may provide maximum excellent efficiency and is reliable. For implementation, it also needs to solve the connection of final user and the resources of real web application. In other words, how to map the URL of real web application to the HTTP service of reverse proxy server and make user to feel that reverse proxy server is real web server. It requires the reverse proxy server to get the resources from web application by user requests in real time and replaces the URL to the URL of reverse proxy server. This operation keeps users operating the resources under the management of reverse proxy server.

We implements reverse proxy server by using Twisted which event-driven networking engine is written in Python [11]. The Twisted framework owned high extensibility, event-driven based, high performance and cross platform features, and also can configure the work mode with multi-threads and multi-processes mixed to adapt the requirement of reverse proxy in system.

The source code below implements the reverse proxy application based on Twisted and it can provide the local reverse proxy service on basis of the web service at port 80 that host example.com open, the proxy service port is 8080.

```python
from twisted.internet import reactor
from twisted.web import proxy, server
site = server.Site(
    proxy.ReverseProxyResource('example.com', 80, ''))
reactor.listenTCP(8080, site)
reactor.run()
```

Implements analysis and forwarding of requests by extend class proxy.ReverseProxyResource:

```python
from twisted.web import proxy
from twisted.web.resource import Resource
class MyReverseProxyResource(Resource):
    isLeaf = False
    # Only accept HTTP method GET or POST
    allowedMethods = ('GET', 'POST')
    def getChild(self, name, request):
        # Detect specified page
        if name == "TargetPage":
            # To processing the user which start requests
            return proxy.ReverseProxyResource(
                'example.com', 80, """/+name)
        else:
            # To processing other users’ requests
            return proxy.ReverseProxyResource(
                'example.com', 80, """/+name)
```

4.2 The implements of RAF module

The RAF module mainly consists of sub modules like Black & White List, Access Control List and SDVC. The implements of each sub module are as below:

1. The sub module of Black & White List. It uses MySQL database to create rules for each user. At first get identification that is combined with IP, User-Agent and additional Cookie-ID and hash it, then query by the database, first query the white list table, let user pass if exist, otherwise query black list table blocked it if user exists. Pass the user to the detection of next sub module if Black & White list does not be found.

   Calculate the identification K(User) that determines unique identity for per user by the hash formula below.

   \[ K(User) = \text{Hash(Request-IP, User-Agent, Cookie-ID)} \]

   Create request_user table to store the users’ identification information, and create black & white list table bw_rules though 1-n type, as shown in Figure 8. In table bw_rules, the user_id is a foreign key point to the request_user table and represents build rules for some users, the status represents the rules type, number -1 represents in black list and number 1 represents in white list, and the prior represents priority of this rule. The rules with high priority can override the rules with low priority.

   ![Figure 8. The declare of request users table and black & white list table.](image)

   For example, if the request user's key_id is 9052e40b07aac0ca, the next statement will query this user is in the white list or not.

   ```sql
   SELECT COUNT(*) FROM `bw_rules` WHERE `status` = 1 AND `user_id` = '9052e40b07aac0ca' ORDER BY `prior` DESC
   ```

   If returned value is Number zero, it indicates it is not in white list, or if returned value is number non-zero, it indicates it is in white list. Then it compares blacklist and the priority value of black to white list to determine filter of this requests by module.

   (2) The sub module of Access Control List. To manage the resource of web application through database system, the web security gateway raises identity verification by proxy if the resources are restricted for users. The gateway will compare user provide credentials with database stored, passes if match, otherwise fails.

   The web security gateway can send HTTP 407 message to implements the authenticate credentials receive:

   HTTP/1.1 407 Proxy Authentication Required
   Proxy-Authenticate: Basic realm="Restrict Resource"
The Basic means the type of authentication, and the realm means the authenticate message which is shown to user. The final user is authenticated by responding this page and provides the required credentials.

Create two tables named auth_users and restrict_urls representing authenticated users and restricted URLs, and connect with many to many through table ausers_rurls. Thus the restricted resource can be granted access to many users and one user can own the permissions with many restricted resources, as shown in Figure 9. For the resources’ URL that user requests, query the table restrict_urls and query the permissions through connecting table ausers_rurls and table auth_users.

For a validation chain is based on tree type, the algorithm of inserting any node to anywhere is as follow:

Algorithm 1: Insert node to hierarchical data structure based on left and right value
Input: Tree, ParentNode, Node will insert
Output: Inserted Node with left and right value
InsertChildNode(Tree, ParentNode, Node)
Begin
(1) If Node and ParentNode is neighbor node Then
(2) Base Value := Max right weight of all parents
(3) Else If Node is ParentNode’s sub node Then
(4) Base Value := ParentNode’s left weight
(5) End If
(6) Nodek = Tree->Node.First
(7) While Nodek->Nodek.Next is not empty
(8) Leftk = Nodek’s left weight
(9) Rightk = Nodek’s right weight
(10) If Leftk > BaseValue Then
(11) Leftk = Leftk + 2
(12) End If
(13) If Rightk > BaseValue Then
(14) Rightk = Rightk + 2
(15) End If
(16) Update Nodek Left weight to Leftk
(17) Update Nodek Right weight to Rightk
(18) End While
(19) Update Node Left weight to BaseValue + 1
(20) Update Node Right weight to BaseValue + 2
(21) Insert Node to Tree
End

For any node or sub node with any rules in the validation chain tree can get from the SQL statement below:

SELECT * FROM process_rules
WHERE left_value BETWEEN left weight of rule
AND right weight of rule node
5 EXPERIMENTAL EVALUATION

To evaluate web gateway, we set up the experimental environment as illustrated in Figure 11. We set up two virtual hosts named A and B through VMWare software and establish the intranet. Among them, A is the web application server and serves web application which is built with PHP 5.5 and MySQL 5.5. Host B is web security gateway system based on reverse proxy technology and the database system using MySQL 5.5. In order to evaluate performance of the proposed web gateway, we open internet port from host B. The testing host also consists of an automatic testing script in Python, browser application for confirm the function implemented as expected and the performance tools for stress testing.

Figure 11. The architecture of experimental evaluation.

The web security gateway we proposed in this paper has Python language environment installed. For configuration and debugging, the system is displayed in UI as shown in Figure 12. The Working Host sets the listening host and port and the Proxy Site input sets the website URL that wants to protect. After configured, we press Start button to enable web security gateway. If the program runs in normal state the list box displays waiting for incoming requests now message. After this we can request the proxied site via port 8080 on web security gateway.

Figure 12. Secure web gateway UI.

5.1 Major functions testing

In our experiments, we first validate the function of black & white list and the access control list in our security web gateway. By configuring the utility script that called config.py to add the current user to the blacklist, the web security gateway will block the requests sent by this user. When we add an existing URL to ACLs, the user who requests this URL will be asked to enter the username and password. The related configuration console interface as shown in Figure 13 and 14.

Figure 13. Add the request user to blacklist

Figure 14. Add URL to ACLs for Restrict request.

We tested the major function in different ways and the testing results are shown in Table 1.

<table>
<thead>
<tr>
<th>ID</th>
<th>Black List</th>
<th>White List</th>
<th>ACL</th>
<th>Restrict</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Pass</td>
</tr>
<tr>
<td>2</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Pass</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Pass</td>
</tr>
<tr>
<td>4</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Pass</td>
</tr>
</tbody>
</table>

From Table 1, we have tested for 4 times, from ID 1 to 4, the test items were Black List, White List, ACL, Restrict which means restrict resource, Yes represent client in restrict option and No represent client not in restrict. Result which means testing result and pass represent the testing functions works otherwise Failure. All tests in our experiments passed which means requests will be blocked if user is in black list and is not in white list, user viewing the restrict web resource will require the username and password by popup dialog and etc.

For testing SDVC by using SQLMap: the automatic SQL injection and database takeover tool written in Python. The attack detection algorithm based on SDVC is N-gram features vector extract with Information gain theory, the BP algorithm for training and recognize the attack traffic. About this algorithm will be explained in other papers that provider by authors, this paper will not explain more. The testing result is shown in Table 2.
Table 2. Testing result by using automatic testing script.

<table>
<thead>
<tr>
<th>ID</th>
<th>Normal</th>
<th>Attacks</th>
<th>Reject</th>
<th>FP</th>
<th>FN</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4328</td>
<td>2368</td>
<td>2364</td>
<td>8</td>
<td>4</td>
<td>99.83%</td>
</tr>
<tr>
<td>2</td>
<td>18653</td>
<td>9675</td>
<td>9668</td>
<td>39</td>
<td>18</td>
<td>99.81%</td>
</tr>
<tr>
<td>3</td>
<td>25331</td>
<td>12335</td>
<td>12318</td>
<td>40</td>
<td>17</td>
<td>99.86%</td>
</tr>
<tr>
<td>4</td>
<td>39856</td>
<td>21034</td>
<td>20798</td>
<td>362</td>
<td>236</td>
<td>98.88%</td>
</tr>
</tbody>
</table>

As testing result shown in Table 2, we have tested for 4 times, from ID 1 to 4. Each testing contains normal requests (Normal) and attack requests (Attacks). We calculated the number of false positive (FP) and true positive from the number of rejected (Reject) of testing result. At last we got detection rate (Rate), from this testing represents the detection algorithm based on SDVC worked perfectly and have high detection rate with high requests.

6 CONCLUSION

This paper considers the web application and web security suffers from insufficient protection technology, and provides a solution based on separated architecture about web security gateway based on reverse proxy. This web security gateway has low coupling, extensibility, reliable and high performance features and uses SDVC to defense various types of network attacks by configuring different policy rules flexibly. The web security gateway in this paper also can establish distributed load balance network to implement high concurrency, large traffic processing ability. Moreover, the proposed web security gateway can implement line backup as well as keeping the whole protection network reliable through multi point redundancy of the web security gateway. The research in this paper can provide strong reference value for some products about web protection technology.

ACKNOWLEDGEMENT

This paper is sponsored by Postdoctoral Research Funding Program of Jiangsu PR of China (1501106C), and Lv Yang Jin Feng 2014 of Yangzhou, China.

REFERENCES