An intrusion detection method based on behavior characteristics for business logic

LI Ru¹, ZHAI Shu-ying¹, ZHI Yong-feng²

¹ Northwestern Polytechnical University Ming De College, xi’an, shaanxi,710124, China;
² Department of Automatic Control, Northwestern Polytechnical University, xi’an, shaanxi, 710072, China

Corresponding author’s e-mail: liru0319@163.com

Abstract: In order to improve the detection efficiency and accuracy of business logic vulnerability, a test method based on behavior characteristics was proposed. It uses the test accounts to crawl url of business system, analyzes the characteristics of request sequence, parameter attributes, request parameters, response parameters, relationship between parameters, and digs out the vulnerability, and then uses the test elements to verify whether there is a vulnerability. Experimental and analytical results show that this method which is instead of manual method can effectively detect the vulnerability of invalid identity authentication, invalid access control and sensitive information leakage.

1. Introduction

FreeBuf security institute released “2018 annual report on financial industry application security situation”¹, it pointed out that based on the business design flaw attack the vulnerability of the highest proportion is 28%. The vulnerability is different from conventional security vulnerabilities such as SQL injection², XSS cross-site scripting³, the commands⁴, and so on. It has no obvious character characteristics on the use of code, and it is difficult to use the WEB application security protection equipment testing⁵, even unable to trigger the alarm, but it will directly affect the safety of users money and accounts, financial institutions asset loss and reputation. At present, the famous WEB vulnerability detection tool⁶-⁹ can detect conventional security vulnerabilities according to character characteristics, but it does not support the detection of business logic vulnerabilities. The mining of business class vulnerabilities still requires manual testing, so the batch discovery of vulnerabilities cannot be realized and the testing efficiency is low. At the same time, there is little research on the testing technology of business logic vulnerability¹⁰-¹², and the detection effect is not satisfactory due to the irregular characteristics of business logic.

Based on the above research, a test method is proposed to analyze the behavior characteristics of request sequence, parameter attribute, request parameter, response parameter and inter-parameter relationship. It builds the original stateless and connectionless HTTP/HTTPS protocol¹³-¹⁴ into a context-related and regular key behavior link (vulnerability test point), and then selects the behavior characteristics of vulnerability test point as the detection element to detect the intrusion behavior of business logic vulnerability.
2. Construct behavior characteristic model

2.1. Relevant concepts

define 1 Action: \( ac: \{ f(\bar{x}) = \bar{y} | \bar{x} \in [x_0,\ldots, x_n], \bar{y} \in [y_0,\ldots, y_n] \} \) represents the operation of the subject on the object under certain conditions. Where, \( \bar{x} \) represents the operation instruction input by the subject to the object, \( \bar{y} \) represents the execution of the return of the subject after the completion of the operation, and \( f \) represents the relevant operation of the subject to the object. For example, if the request information is GET/icap-ibank/acctDetailQuery?ACCT_NO=6230270100021776396, \( f \) is icap-ibank.acctDetailQuery, \( \bar{x} \) is [6230270100021776396], \( \bar{y} \) is [Li, 18392387896, 111.11].

define 2 Parameter-Attribute: \( pa: \{IDN_NO, NAME, PH_NO, \ldots \} \) parameter properties represent the characteristics of input and output parameters. If the parameter contains sensitive information such as name, id number, mobile phone number, card number, home address, amount, etc., and its characteristics can be correctly expressed by using regular expressions, it needs to be included in the critical path. For example, the regular expression for the phone number is \( ^{1}[1[(3)[0-9]][4][5-9]][5][0-3, 5-9]][6][5, 6][7][0-8][8][0-9]][9][1, 8, 9][0-9][8]$.

define 3 Parameter-Relationship: \( pr: \{ equal | ac_i, x_j = ac_i, x_j \} \)

\( \{ dependency | ac_i, x_j = ac_i, y_j \} \) contains both passing and dependent parameters. The pass-through relationship indicates that the input parameters are the same between the two actions. A dependent parameter relationship indicates that the input operation of one action is derived from the output parameter of another action.

define 4 Task: \( task: \{ ac_i, \ldots, ac_n \} \) represents a series of actions to accomplish a purpose. For example, reset password business, registration business, and so on.

define 5 Behavior Characteristics: \( bc: \{ task, pr, pa \} \) In the context of the action sequence, the key behavior path (vulnerability test point) is extracted by representing the action relevance characteristics according to parameter attributes and binary relations among parameters. The action sequence, parameter attribute and binary relationship among parameters of vulnerability test point are called behavior characteristics.

2.2. Algorithm of building behavior characteristics

Input:A1,B1, tasks: \( \{ task_0, \ldots, task_i, \ldots, task_{n-1} \} \) /tasks include registration, password reset, query and other tasks, A1 users have higher privileges than B1 and A1 has corresponding sensitive information (id card, mobile phone number, card number, amount, name, location, etc.) */

Output: behavior characteristics

01. for \( i \leftarrow 0 \) to \( n-1 \) do /*i for task*/
02. \( j \leftarrow 0, k \leftarrow 0; /* J for action, k for rule */
03. While(\( \text{GetAction}(\text{task}, \text{ac}) \)) do
04. if(\( \text{Analyse Parameter-Attribute}(\text{task}, \text{ac}) \))
05. \( rule, ac_k, pa = task, ac_j, pa; \)
06. endif/* Get parameter properties using regular expressions */
07. \( k++; \)
08. for \( z \leftarrow j \) to 0 do
09. if(\( \text{Analyse Parameter-Relationship}(\text{task}, \text{ac}_j, \text{task}, \text{ac}_z) \))
10. \( rule, ac_k, pr = task, ac_j, pr; /* Dependencies and transients, and dependencies are stronger than
transients*/
11. break;
12. endif
13. z--; 
14. endif/* Get the parameter binary relationship */
15. j++;
16. endwhile
17.endfor

The general permission user and the administrator permission user are used for the business operation, the context-related and regular key behavior links are extracted, and the behavior characteristics of the key behavior links are formed into intrusion detection rules.

Take the payment transaction as an example, the mobile payment process is shown in Table 1. The detection rules of mobile payment payment task are shown in Table 2.

Table 1. mobile phone payment process

| URL                        | In parameter                                                | Out parameter                                      |
|----------------------------|-------------------------------------------------------------|----------------------------------------------------|
| 1  chinaUnitedRecharge.do   | [SVC_NO: “1520297329”, TRANS_AMT: “100.00”, PAY_NO:“6230270100021776503”, FLAG:“MOBILE”] | {SVC_NO: “1520297329”, MSG: “Successful trading”} |
|                            | PAY_NO:“6230270100021776503”, FLAG:“MOBILE”                |                                                    |
| 2  transfer.do             | PAY_NO:“6230270100021776503”, RECV_ACCT_NO:“6230270100021776396”,RECV_NAME:“Li” | {MSG: “Successful trading”}                       |
|                            | [MSG: “Successful trading”]                                 |                                                    |
| 3  acctDetailQueryNotCheck.do | {ACCT_NO:“6230270100021776503”}                           |                                                    |
|                            | [SVC_NO: “1520297329”, TRANS_AMT: “100.00”, PAY_NO:“6230270100021776503”, FLAG:“MOBILE”,RECV_ACCT_NO:“6230270100021776396”,AUTH_TYPE: “01”,CERT:”mctype_key:[MTIzNA==”U2FsdGVkX19b8PCSQJCB3vWSZj5x5w==”} | {AVAL_BAL: “111.4”, IDN_NO:“130435198710130958”,PRESV_MOBILE: “18392387896”,NAME:”Yang XX” } |
| 4  transferWithMobile.do   | [SVC_NO: “1520297329”, TRANS_AMT: “100.00”, PAY_NO:“6230270100021776503”, FLAG:“MOBILE”,RECV_ACCT_NO:“6230270100021776396”,AUTH_TYPE: “01”,CERT:”mctype_key:[MTIzNA==”U2FsdGVkX19b8PCSQJCB3vWSZj5x5w==”} | {MSG: “Successful trading”}                       |
|                            | {AVAL_BAL: “11.4”}                                          |                                                    |
| 5  custInfoQuery.do        | {LOGIN_ID:“18392387896”}                                    | {AVAL_BAL: “11.4”}                                 |
Table 2. mobile phone payment detection rules

| function name                  | action | parameter 1                      | parameter 2                      |
|-------------------------------|--------|----------------------------------|----------------------------------|
| chinaUnitedRecharge.do(01)     | ac     | {SVC_NO: 03, TRANS_AMT: 06,     | PAY_NO: 04} /* Use the regular   |
|                               |        | PAY_NO: 04} /* Use the regular   | expression to determine the name, |
|                               |        | name, id number, mobile          | id number, mobile                 |
|                               |        | phone number, card number, home  | address, amount and other        |
|                               |        | address, amount and other        | sensitive information, at         |
|                               |        | the same time with 01,02...said */ |
| transfer.do(02)                | pa     | {TRANS_AMT: 06, PAY_NO: 04}      | eq{01:TRANS_AMT, 01:PAY_NO}      |
|                               |        | eq{01:TRANS_AMT, 01:PAY_NO}      |                                  |
| acctDetailQueryNotCheck.do(03) |        | {ACCT_NO: 04} Respone=1          | eq{ACCT_NO: 02:PAY_NAME}          |
| transferWithMobile.do(04)      |        | {SVC_NO: 03, TRANS_AMT: 06,     | eq{PAY_NO: 03:ACCT_NAME:04,       |
|                               |        | PAY_NO: 04, RECV_ACCT_NO: 04,    | RECV_NAME:01}                     |
| custInfoQuery.do(05)           |        | {LOGIN_ID: 03}                   | de{LOGIN_ID: 03:PRESV_MOBILE}     |

3. Intrusion detection for business logic

3.1. Algorithm of checking behavior characteristics

Input: tasks {task0, ..., taskn, taskn-1} /* Reset passwords, pay transactions, etc */

Output: action is True or False

01. z ← 0
02. while (GetAction(ac) do
03. Register(ac.session) /* Distinguish different users according to Cookie and session information */
04. for j ← 0 to total do /* total represents all actions */
05. if (ac == acj) /* Match AC action and parameter name */
06. Add(ac, actions[], z++) /* actions store intercepts */
07. if (acj.pa! = null)
08. if (isSen(acj.pa- respone) & & is NULL(ac.session)) /* If the session is NULL and the returned data has sensitive information, the unauthorized vulnerability is marked */
09. return False
10. endif
11. if (isRegExp(acj.pa, ac)) /* Sensitive information does not conform to the regular expression, for example -0.1 yuan does not conform to the transfer amount regular expression */
12. return False
13. endif
14. if (acj.pr! = null)
15. if (Check_Parameter-Relationship(ac, actions[], acj.pr, z)) /* Determine whether the intercept action binary relationship is consistent with the storage relationship */
16. return False
17. endif
18. break
19. endif
20. endif
21. endwhile
3.2. Detection capability analysis

The attacker exploits the business logic vulnerability, bypasses the business rules, and manipulates the business-related control flow and data flow to achieve malicious purposes. Because the data submitted by the client side is not properly limited by the validity checksum (frequency, frequency, execution order, etc.), there will be the cases where the key data is tampered, the control strategy is bypassed, the sensitive data is leaked, and the function is abused. Most of the exploitation of business logic loopholes occurs in the links such as identity authentication, authorization and payment transaction. According to the attack methods and vulnerabilities given in literature [10-12], the detection capability was analyzed item by item according to the intrusion detection model.

(1) unauthorized access vulnerability

Overreach vulnerability is a very common logic vulnerability. Overstepping the authority vulnerability is mainly because developers lack the judgment of the authority when they add, delete, change and query the data. Attackers modify the relevant parameters can have other accounts to add, delete, check, change the function.

The access control business usually consists of authentication, authorization, and operations. After the user enters the account number and password to complete the identity authentication, the server side returns the user identity ID, and the server determines the user authority and grants the relevant operation and query authority according to the identity ID. As long as the attacker obtains the identity of other users by modifying the ID, the dependency relationship of parameter ID will not be established, triggering an early warning.

(2) payment transaction vulnerability

Payment transaction vulnerability refers to the lack of completeness verification of the order, confirmation information and payment steps of goods purchased or transferred, resulting in payment vulnerability, such as tampering with payment amount, order number, quantity, etc. The attacker caused capital loss to the merchant by modifying the payment amount, order number and quantity.

Payment transaction business usually consists of order generation, payment order, and update account information. The user selects the quantity, item number and amount to generate the order number, and then makes the bank card payment according to the order number (determine the balance), completes the payment and updates the accounts of both parties or the third-party accounts. Item number and order number are passed as binary relation, and payment amount and bank card number are parameter attributes. If the attacker modifies the payment amount to be negative, the regular expression will not pass, thus causing the parameter property to be invalid and triggering the early warning. If an attacker modifies the item number and order number to obtain the item for a lower amount, the binary relationship will not be established to trigger the warning.

(3) unauthorized vulnerability

Unauthorized vulnerability means that there is no authorization verification, and any user who knows the access link can have the operation permission. The attacker only needs to get access to the link and can also operate on other users without logging into the system.

Access to sensitive information is usually as simple as logging in and reading the information. When there is an unauthorized vulnerability, the attacker only needs to delete the COOKIE information, can arbitrarily obtain all the user information. In the detection stage of the model, the returned information can be determined as an intrusion based on unauthorized access vulnerability by determining whether the returned information conforms to the parameter property requirements, and if the returned information conforms to the property requirements and the COOKIE information is null.

(4) identity authentication vulnerability

Identity authentication vulnerability refers to the existence of a flaw in the identity authentication mechanism, which can bypass the specific steps of identity authentication and obtain the corresponding authority. Typical authentication vulnerabilities include batch registration, password reset, etc.

The authentication business usually consists of several two-factor processes to determine the user's sensitive information. To register and reset passwords, users need to authenticate sensitive information,
including name, identity account number, mobile phone number, dynamic SMS verification code. Batch registration vulnerability is usually completed by repeatedly submitting the last step of identity authentication (other steps are bypassed), but the last step of identity authentication information must have a binary relationship with other identity authentication steps and the authentication information conforms to the requirements of the relationship attribute. The attacker batch registration will cause the identity authentication information binary relationship is not established. To reset the password, the authentication of identity information needs to be completed under the premise of logging in the system, and then the password is reset according to the user ID or user name. At this time, the user ID or user name establishes a binary relationship with the user ID and user name in the login stage. When the attacker resets others' password, the parameter binary relation will be broken and the warning will be triggered.

4. Experimental results and evaluation

Three urban commercial Banks were selected as the test objects for the online banking system. The leakage sweep tool was AWVS. The manual test was conducted by a senior penetration test engineer with 6 years of work experience. The experimental results only recorded the high-risk business logic vulnerabilities, excluding SQL, XSS and other character-based vulnerabilities. The experimental results are shown in the following table:

| Target site               | Sweep leakage test | Manual testing | Model detection early warning | Note                                      |
|--------------------------|--------------------|----------------|-------------------------------|-------------------------------------------|
| 1  ibank.96262.com        | 0                  | 13             | 13                            | The bug has been fixed                    |
|                          |                    |                |                               | The bug has been fixed                    |
|                          |                    |                |                               | The two unwarned intrusion attempts are due to the failure of dynamic encryption information association and the uncollected part of AC in the training phase. |
| 2  ebank.xacbank.com      | 0                  | 9              | 7                             |                                           |
|                          |                    |                |                               |                                           |
| 3  ebank.ccabchina.com    | 0                  | 1              | 1                             | The bug has been fixed                    |

The completeness of the training stage of the business logic behavior characteristics has a great impact on the detection accuracy. To improve the detection accuracy, it is necessary to improve the training completeness, including at least the key tasks.

5. Conclusion

By proposing an instrumented intrusion detection method for business logic vulnerabilities, the original stateless and connectionless protocols are constructed into contextual and regular behavior characteristics. As long as the attacker changes the behavior characteristics of the business logic, the system will be alerted and prevent further intrusion.

References

[1] FreeBuf security research institute. 2018 annual report on applied security in the financial industry [EB/OL].https://www.freebuf.com/articles/paper/190442.html.

[2] Li, X., Zhang, W.W., Sui, Z.C., Zheng, L.X. (2016) Research and Analysis on the Novel SQL Injection and Defense Technique. Netinfo Security, 02:66-73.

[3] Gu, J.T. (2019) Research on XSS vulnerability detection method based on dynamic analysis. Beijing university of posts and telecommunications.
[4] Li, J.X. (2019) Research on WordPress 5.0.0 remote code execution vulnerability. Journal of Information Security Research., 04:352-360.
[5] Zhang, Q., Zhai, J.H. (2017) Research of WAF rules auto detection. Intelligent Computer and Applications, 06:103-107.
[6] Suo, L. (2010) The testing and analysis of vulnerability detection ability of mainstream scanning tools. Information Security and Technology, 06:120-128.
[7] He, X.G. (2017) Design and implementation of network security scanning system. Harbin Institute of Technology.
[8] Xiao, Z. (2014) Design and implementation of network vulnerability scanning detection system based on web crawler system. Jilin university.
[9] Yu, S.Y., Wang, Y.T., Liu, X. (2016) Application of the Burpsuite tool in vulnerability detection. Netinfo Security, 09:94-97.
[10] Xue, N.F. (2018) Penetration Testing based Logical Vulnerability Detection Technology. University of electronic science and technology.
[11] Jing, S. (2015) Web application vulnerability detection technology research based on penetration test. Beijing university of posts and telecommunications.
[12] Chen, J.J. (2019) Design and implementation of high performance Web application vulnerability scanning system. Beijing university of posts and telecommunications.
[13] Li, C.Z. (2019) A computer network forensics method based on HTTP message analysis. Network Security Technology & Application, 03:101-102.
[14] Zhang, B.Y. (2016) A brief analysis of the principle and application of HTTPS protocol. Network Security Technology & Application, 07:36-37.