2. Плавинский С.Л. Руководство по мета-анализу: учебное пособие. СПб.: СПбМАПО, 2014. 75 с.
3. Сергиенко В.И., Бондарева И.Б. Математическая статистика в клинических исследованиях. М.: ГЭОТАР-МЕД, 2015. 256 с.
4. Аксель, Е.М. Злокачественные новообразования молочной-железы: состояние онкологическое помощи; заболеваемость и смертность / Е.М. Аксель/ /Маммология. - 2006.
5. Ананина, О.А. Ранней диагностика рака молочной железы на основе информационной системы. / Диссертация/ - 2008
6. Очиқ маълумотлар портали – http://www.data.gov.uz
7. Ўзбекистон Республикаси Давлат комитети сайти – http://www.stat.uz

Зайнутдинова Мастура Баходировна
т.ф.н., Муҳаммад ал-Хоразмий номидаги Тошкент ахборот ткнологиялари университети (ТАТУ) Ахборот технологиялари (АТ) кафедраси доценти.
Сайфуллаев Шерзод Бахтиёр ўғли
Муҳаммад ал-Хоразмий номидаги ТАТУ АТ кафедраси магистранти
Эл. почта: shezodsay@gmail.com
Ахмедова Нозима Фарҳод қизи
Муҳаммад ал-Хоразмий номидаги ТАТУ АТ кафедраси магистранти
Эл. почта: Rachel_051294@mail.ru

Zaynutdinova M.B., Sayfullaev Sh. B., Akhmedova N.F.

Detection of breast cancer using information system which process big data

The article presents the results of the development of an information system for the analysis and identification of individual risk factors and the formation of groups of increased risk of breast cancer. The first results were obtained, which allowed to establish the most significant risk factors for breast cancer.

Keywords: *breast cancer, information system, hazardous risk group, risk factors, forecasting table, questionnaire statistics, oncology.*

М.Б. Зайнутдинова, Ш.Б. Сайфуллаев, Н.Ф. Ахмедова

Обнаружение рака молочной железы с использованием информационной системы, которая обрабатывает большие данные

В статье представлены результаты разработки информационной системы по анализу и выявлению факторов индивидуального риска и формированию групп повышенного риска рака молочной железы. Получены первые результаты, позволившие установить наиболее значимые факторы риска рака молочной железы.

Ключевые слова: *рак молочной железы, информационная система, группа риска, факторы риска, таблица прогнозирования, статистика вопросников, онкология.*

УДК 004.056.53

Рахманов А.Т., Керимов К.Ф., Камалов Ш.К.

АЛГОРИТМ АВТОМАТИЧЕСКОГО ОБНАРУЖЕНИЯ УЯЗВИМОСТИ ВИДА SQL ИНЪЕКЦИИ

В данной статье разрабатывается алгоритм обнаружения атаки по инъекции SQL с помощью функции извлечения одного символа, и оценку эффективности предложенного алгоритма с помощью искусственных данных.

Ключевые слова: SQL инъекция, уязвимость, обнаружение атак, определение угроз, использование специальных символов и строк, алгоритм.

Математическое моделирование и идентификация информационных объектов играют важную роль при решении задач распознования образов. Одной из таких задач является обнаружение атак или нормальных запросов на веб приложения. Исследования, посвящённые изучению обнаружения атак или нормальных запросов на веб приложения, начались сравнительно недавно. Но, тем не менее, существует много исследований в этом направлении [1-11]. Применение математических методов в решении таких задач в основном велись японскими учеными [1,2,6]. Например, в работе [1] предложено 2 способа обнаружения атак SQL инъекций, основанных на свойстве распределения символов при построении атак SQL инъекций. В работе [2]

предложены моделирование атак и нормальных запросов и их идентификация с помощью некоторой функции, нижняя граница которой зависит от длины входной строки и, вообще говоря, неограниченна снизу. В нашей работе предлагается математический способ идентификации атак SQL инъекций с помощью ограниченной снизу функции, которая зависит от входной строки. Для построения такой функции мы использовали специальные знаки и ключевые слова, которые часто встречаются в построении атак злоумышленников.

Часто в построении атак SQL инъекций используются специальные символы и специальные ключевые слова, которые приведены в следующих таблицах.

Таблица 1

Специальные символы

Переменная	Символ
u_1	Пробел
u_2	Точка-запятая(;)
u_3	Апостроф(')
u_4	Правая скобка())
u_5	Левая скобка(()
u_6	Правая фигурная скобка (})
u_7	Левая фигурная скобка ({)
u_8	Правая квадратная скобка (])
u_9	Левая квадратная скобка([)
u_{10}	Диез(#)
u_{11}	Процент (%)
u_{12}	Кавычка (")
u_{13}	Амперсанд (&)
u_{14}	Обратная косая (\)
u_{15}	Вертикальная линия (|)
u_{16}	Знак равенства (=)
u_{17}	Больше чем (>)
u_{18}	Меньше чем (<)
u_{19}	Звездочка (*)
u_{20}	Косая черта (/)

Таблица 2

Специальные ключевые слова

Переменная	Ключевые слова
u_{21}	and
u_{22}	or
u_{23}	union
u_{24}	where
u_{25}	limit
u_{26}	group by
u_{27}	select
u_{28}	\'
u_{29}	hex
u_{30}	substr

Для определения SQL инъекции вводим характеристики атак SQL инъекций с помощью специальных символов из таблицы 1 и специальных ключевых слов из таблицы 2. Пусть наблюдается некоторая входная строка L, $x_1, x_2, ..., x_{20}$ - частота появления в L специальных знаков из таблицы 1, $x_{21}, x_{22}, ..., x_{30}$ являются частотой появления специальных ключевых слов из таблицы 2, x_{31} - частота появления всех остальных знаков и чисел 0,1,2,...,9 в строке L. С точки зрения определения атак SQL инъекций обычные символы $a, b, ..., z$ и числа 0, 1, … , 9 не играют важной роли. По этому в данной работе мы всегда считаем что частота появления всех этих символов и чисел в наблюдаемой строке L равно 1, т.е. $x_{31} = 1$. Таким образом, любую строку L можно определить с помощью характеристик следующим образом: $L = (x_1, x_2, ..., x_{20}, x_{21}, ..., x_{30}, x_{31})$, как элемент некоторого фазового пространства X.

Метод определения

Из определения L видно, что любой элемент L из построенного пространства X лежит на гиперплоскости

$$\Gamma = \{L = (x_1, x_2, ..., x_{20}, ..., x_{30}, x_{31}) : x_{31} = 1\}.$$

Используя данное уравнения гиперплоскости, можно предположить, что чем больше частота появления специальных знаков и ключевых слов во входной строке, тем очевиднее становится близость входной строки L к атакам SQL инъекций. Поэтому функция определения атаки должна быть возрастающей по переменным $x_1, x_2, ..., x_{20}, x_{21}, ..., x_{30}$ и убывающей по переменной x_{31}. Исходя из этого, предлагаем следующую возрастающую по $x_1, x_2, ..., x_{20}, x_{21}, ..., x_{30}$ функцию:

$$f(L) = f(x_1, x_2, ..., x_{20}, x_{21}, ..., x_{31}) = \frac{\sum_{i=1}^{30} x_i}{\sum_{i=1}^{30} x_i + x_{31}}$$

для определения атак SQL инъекций. Так как в данной работе мы всегда считаем, что частота появления всех остальных знаков и чисел 0,1, 2,..., 9 в строке L равно 1, то из последнего равенства получим:

$$f(L) = f(x_1, x_2, ..., x_{20}, x_{21}, ..., x_{31}) = \frac{\sum_{i=1}^{30} x_i}{\sum_{i=1}^{30} x_i + 1} \quad (1)$$

Данная функция имеет следующие свойства: 1) $0 \le f(L) < 1$ для всех $L \in \Gamma$;

2) для атак SQL инъекций минимальное значение функции снизу ограничено числом 1/2.

Таким образом, если входная строка L является атакой SQL инъекции, то это строка по крайней

мере должна содержать один специальный символ из таблицы №1 или одно ключевое слово из таблицы №2. Поэтому $\sum_{i=1}^{30} x_i \geq 1$, и так как функция $f(L)$ является возрастающей по каждому из переменных x_i, её минимум при $\sum_{i=1}^{30} x_i \geq 1$ достигается в точке L_0, для которого $\sum_{i=1}^{30} x_i = 1$.

Таким образом, если L произвольная строка и $f(L) \geq 1/2$, то L, возможно, является атакой SQL инъекции, или же $f(L) < 1/2$, то тогда входная строка, возможно, является нормальной, если при построении атак SQL инъекций используются специальные ключевые слова из таблицы 2. Поэтому функцию (1) можно использовать для распознавания нормальных строк и атак SQL инъекций, построенных с помощью специальных символов и ключевых слов.

Таким образом, если L произвольная строка, содержащая минимум 2 специальных символов из таблицы 1, то $f(L) \geq 2/3$, и L, возможно, является атакой SQL инъекции, или же $f(L) < 2/3$, то тогда входная строка, возможно, является нормальной, если при построении атак SQL инъекций используются только специальные символы из таблицы 1. Поэтому функцию (1) можно использовать для распознавания нормальных строк и атак SQL инъекций, построенных с помощью специальных символов из таблицы 1 и специальных ключевых слов из таблицы 2.

В обоих случаях, используя функцию (1), мы имеем критерий качества для определения угроз. Похожая на (1) функция была построена и использована в работе [*], но там значение функции зависит от длины входной строки L и минимума такой функции для любой строки L не существует. Поэтому в работах [*] для определения границы разпознающей функции строятся дополнительные усредненные критерии качества, решая соответствующую оптимизационную задачу, используя дополнительные математические аппараты.

В нашем случае границей разпознающей функции (1) является рациональное число 1/2. Таким образом, если строка L содержит хотя бы один специальный знак или же одно ключевое слово, то условия $f(L) \geq 1/2$ достаточно для определения угрозы.

Таблица 3

Образцы строк по инъекции SQL

Номер	Строки атаки
1	id=1'
2	AlexanderPHP'
3	AlexanderPHP'%20--%20habrahabr
4	1 UNION SELECT 1,2
5	1 UNION SELECT 1,2,3
6	1 UNION SELECT 1,2,3,4,5
7	1 GROUP BY 2
8	1 GROUP BY 8
9	-1 UNION SELECT 1,2,3,4,5
10	-1 UNION SELECT 1,2,3,4,5 FROM users WHERE id=1
11	-1 UNION SELECT name,2,pass,4,5 FROM users WHERE id=1
12	-1' UNION SELECT name,2,pass,4,5 FROM users WHERE id=1 --%20
13	-1' UNION SELECT 1,'<?php eval($_GET[1]) ?>',3,4,5 INTO OUTFILE '1.php' --%20
14	-1' UNION SELECT 1,2,3,4,5 INTO OUTFILE '1.php' --%20
15	-1' UNION SELECT 1,LOAD_FILE('1.php'),3,4,5 --%20
16	4+OR+1
17	4+--
18	4+UNION+SELECT+*+FROM+news+WHERE +id=4
19	admin' --
20	admin' #
21	admin'/*
22	' or 1=1--
23	' or 1=1#
24	' or 1=1/*
25	') or '1'='1--
26	') or ('1'='1—
27	' HAVING 1=1 --
28	' GROUP BY table.columnfromerror1 HAVING 1=1 --
29	' GROUP BY table.columnfromerror1, columnfromerror2 HAVING 1=1 --
30	' GROUP BY table.columnfromerror1 columnfromerror2, columnfromerror3 HAVING
31	10 UNION SELECT TOP 1 password FROM admin_login where login_name='neo'--
32	10 UNION SELECT TOP 1 password FROM admin_login where login_name='trinity'--
33	10 UNION SELECT TOP 1 convert(int, password%2b'%20morpheus') FROM admin_login where login_name='trinity'--
34	10; UPDATE 'admin_login' SET 'password' = 'newpas5' WHERE login_name='neo--
35	10; INSERT INTO 'admin_login' ('login_id', 'login_name', 'password', 'details') VALUES (635,'neo2','newpas5','NA')--
36	hi' or 1=1—
37	food' or 1=1—
38	10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES--
39	10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME NOT IN ('table1')--

Образцы нормальных строк

Номер	Нормальные строки
1	test

2	password
3	kamil@
4	@kamil
5	{(1%2)+(3/4)}/5}
6	&temptest(URL){ width,height }

Определение степени важности специальных символов по данному алгоритму: результаты эксперимента

	степени важности для y
=	0.4872
%	0.2051
‘	0.6923
*	0.0769
/	0.0513
]	0.0256
[	0.0257
{	0
}	0
&	0
\	0
#	0.0513
“	0
!	0
<	0.0256
>	0.0255
(	0.1538
)	0.1795
;	0
пробел	0.7949

4. ЗАКЛЮЧЕНИЕ

В данной работе мы предложили алгоритм обнаружения атаки по инъекции SQL с помощью определенной функции и оценку эффективности предложенного алгоритма с помощью искусственных данных.

В предлагаемом алгоритме мы определили примерный набор символов, который сочетается как с атакой, так и с нормальными обнаружениями, и с ранее известным порогом, используя примерные данные атакующих и нормальных строк. Согласно нашим экспериментам с искусственными данными, набор содержит пробел, точку с запятой, а правая скобка хорошо зарекомендовала себя для большего диапазона веса для атаки и нормальной строки. Важным моментом, однако, является гибкий выбор лучшего набора в зависимости от наблюдаемых данных.

Проблемой в предлагаемом алгоритме является сбор естественных атак из серверов веб-приложений в эксплуатации и нормальных строк и их генерация.

ЛИТЕРАТУРА

1. Michio Sonoda, Takeshi Matsuda, On Automatic Detection of SQL Injection Attacks by the Feature Extraction of the Single Character, Proceeding of 2011 IEEE International Conference on Systems.

2. Takeshi Matsuda, Daiki Koizumi, Michio Sonoda, and Shigeichi Hirasawa, “On Predictive Errors of SQL Injection Attack Detection by the Feature of the Single Character,” Proceeding of 2011 IEEE International Conference on Systems, Man, and Cybernetics (to appear).

3. G. T. Buehrer, B. W. Weide, P. A. G. Sivilotti, *Using Parse Tree Validation to Prevent SQL Injection Attacks*, SEM 2005

4. W. G. Halfond, A. Orso, *Combining Static Analysis & Runtime Monitoring to Counter SQL-Injection Attacks*, WODA 2005

5. W. G. Halfond, A. Orso, *AMNESIA: Analysis and Monitoring for NEutralizing SQL Injection Attacks*, ASE 2005

6. Tsunoda Naoki, Yasui Hiroyuki, and Matsuyama Minoru, Detection for SQL Injection with Anomaly Detection Method, The 71th National Convention of ISPJ collection of papers (3), pp.379–380, 2009.

7. William Robertson, Giovanni Vigna, Christopher Kruegel, and Richard A. Kemmerer, “Using Generalization and Characterization Techniques in the Anomaly-based Detection of Web Attacks,” in Proceeding of the Network and Distributed System Security (NDSS) Symposium, San Diego, CA, February, 2006.

8. Justin Clarke, SQL Injection Attacks And Defense, Syngress Publishing Inc., 2009.

9. Ke Wei, M. Muthuprasanna, S. Kothari, *Eliminating SQL Injection Attacks in Stored Procedures*, pp. 191-198, IEEE ASWEC, 2006

10. W. R. Cook, S. Rai, *Safe Query Objects: Statically Typed Objects as Remotely Executable Queries*, ICSE 2005

11. R. McClure, I. Kruger, *SQL DOM: Compile Time Checking of Dynamic SQL Statements*, ICSE 2005

Рахманов Аскар Таджибаевич
т.ф.н., Муҳаммад ал-Хоразмий номидаги ТАТУ ТАД кафедраси доценти.
Эл. почта: domack@gmail.com
Керимов Камиль Фикратович
т.ф.н., Муҳаммад ал-Хоразмий номидаги ТАТУ ТАД кафедраси мудири
Эл. почта: domack@gmail.com

Камалов Шухрат Камалович
Муҳаммад ал-Хоразмий номидаги ТАТУ ТАД кафедраси катта укитувчи
Эл. почта: kamalov.shukhrat@gmail.com

Rahmanov A.T., Kerimov K. F., Kamalov Sh.K.
Algorithm for automatic detection of vulnerability of the form of SQL injection

The article presents an algorithm for detecting attacks on SQL injection using the function of extracting one character, and evaluating the effectiveness of the proposed algorithm using artificial data.

Keywords: SQL injection, vulnerability, attack detection, threat detection, use of special characters and strings, algorithm.

Rahmanov A.T., Kerimov K. F., Kamalov Sh.K.
SQL in'ektsiyasi shaklining zaifligini avtomatik aniqlash algoritmi

Maqolada bir belgini chiqarish funktsiyasidan foydalangan holda SQL in'ektsiyalariga qarshi hujumlarni aniqlash bo'yicha algoritm ishlab chiqadi va sun'iy ma'lumotlardan taklif qilingan algoritmning samaradorligini baholaydi.
Kalit so'zlar: SQL in'ektsiyasi, zaiflik, hujumni aniqlash, tahdidlarni aniqlash, maxsus belgilar va satrlarni ishlatish, algoritm.