

ゼータ分布を用いた SQL インジェクション攻撃を検知する方法について

氏 名 前田 すみれ^I, 松田 健^{II}, 園田 道夫^{III}, 趙 晋輝^{IV}

所 属 中央大学 理工学部 情報工学科

静岡理工科大学 総合情報学部 コンピュータシステム学科

サイバー大学 IT 総合学部

1. はじめに

インターネットの普及によって様々な情報の管理・運用が容易になったことに伴い、システムの脆弱性を狙って情報を盗み取るといった攻撃が増加している。そのような攻撃方法の一種である SQL インジェクション攻撃は、web 上に展開されているアプリケーションを狙った攻撃の一つで、web 上の入力欄に不正な文字列を入力することでデータベースに対して不正な操作を行う攻撃技術として知られている。

SQL インジェクションへの対策として最も効果的であるとされているのが、プリペアドステートメントを利用して web サイトを制作する方法である。しかし、この方法は比較的最近普及し始めた対策であるため、適切に使われていない場合もあり、この対策方法が世の中に広く浸透しているわけでもない。また、SQL インジェクション攻撃の検知方法として現在主流のブラックリスト方式では、攻撃文の多様化や、攻撃文が偽装された場合に対処しきれないといったデメリットがあり、対策が充分であるとは言いきれないため、SQL インジェクション攻撃に対する有効かつ本質的な防御方法は今なお確立されていないのが実情である(文献^[1])。

そこで本研究では、web サイト上の入力欄に入力された文字列が攻撃文字列か攻撃文字列でないかを判定する方法として、既存研究であるゼータ分布と入力文の記号の分布を近似する方法を参考に、SQL インジェクション攻撃を検出する方法を提案した。提案手法を用いた結果、入力文の記号の分布を特徴ごとに分類することで検出の精度を高くできることを実証した。

2. 事前知識

この章では、本研究で提案する攻撃検出法に

必要な知識について簡潔にまとめることにする。

2.1 SQL インジェクション攻撃の例

SQL インジェクション攻撃は、web サイトの入力フォーム等に悪意のある SQL 文を挿入することで実行される。以下の文字列は典型的な SQL インジェクション攻撃である。

```
; DROP members--
```

このタイプの攻撃が成立すると、データベースに登録されている情報が攻撃者によって削除されることになる。上述の例のように、SQL インジェクション攻撃にはスペースやセミコロンなどの区切り文字が多く現れる傾向があり、この性質は、文献^[2]で SQL インジェクション攻撃に含まれる文字の分布がゼータ分布で近似されることと関わりを持っていると考えられる。つまり、SQL インジェクション攻撃を成功させるためには何らかの記号を用いる必要があるが、その中でもいくつかの特定の記号が重要な働きをしている可能性があることを文献^[2]の研究成果は示していると考えられる。

そこで本研究では、このような SQL インジェクション攻撃に含まれる記号の分布（以下、攻撃の記号分布という）の性質を応用し、攻撃を検出する方法を提案する。

2.2 ゼータ分布

ゼータ分布は

$$p(x|a) = \frac{1}{\zeta(a)x^a}$$

で定義される確率分布である。ここで、 x は自然数であり、 a は 1 以上の実数とする。 $\zeta(a)$ は

$$\zeta(a) = \sum_{n=1}^{\infty} \frac{1}{n^a}$$

On the detection method of SQL Injection Attacks using the property of zeta distribution

I Maeda Sumire, Chuo University

II Matsuda Takeshi, Shizuoka Institute of Science and Technology

III Sonoda Michio, Cyber University

IV Chao Jinhui, Chuo University

[2] Matsuda Takeshi, Feature of SQL Injection Attacks and Zeta Distribution, FIT2013

で定義されるリーマンのゼータ関数である．本研究では，ゼータ分布のパラメータ a の値を文献^[2]の手法を参考にし，攻撃の記号分布とゼータ分布とのカルバック情報量を小さくするパラメータ a' の値を計算することでゼータ分布のパラメータを推定し，攻撃の記号分布を $p(x|a')$ と近似し， $p(x|a')$ の性質を用いて攻撃検出を行う．

ここで，攻撃の記号分布を以下のように定義する．

$$T(s_j) = \sum_{i=1}^I \frac{x_i(s_j)}{|l_i| \cdot I}$$

ここで， s_j は記号， I はデータ数， $|l_i|$ は文字列の長さ， $x_i(s_j)$ は文字列 l_i における記号 s_j の出現頻度を表す．

3 提案手法

本研究では，入力文の記号の分布をゼータ分布で近似し，ゼータ分布のパラメータ a を求めることにより入力文が攻撃文かそうでないかを判定する方法を提案する．入力文の記号の分布をゼータ分布で近似するために，以下のカルバック情報量

$$KLD(a) = \sum_{x=1}^X T(x) \log \frac{T(x)}{p(x|a)}$$

を考え， $KLD(a)$ を最小にするパラメータ a を計算する．本来， X は無限大であるべきであるが，入力文の記号の分布もゼータ分布も自然数 x が大きくなるとほとんど0になるため， X は有限な値とする．本研究で扱う入力文の記号は20個であるため， $X=20$ として計算することとする．

以上の準備のもとで，以下の手順を用いて攻撃かそうでないかの判定を行う．

- (1) 訓練データの $T(x)$ を $p(x|a_{Tr})$ で近似
- (2) テストデータの分布を $p(x|a')$ で近似
- (3) $|a' - a_{Tr}| < M$ ならそのデータを攻撃と判別

上記の手順(3)について，次章で準備したデータを用いて検討したところ， $0.3 < M < 1.0$ の値の範囲で上手く検出できることを確認したため，本研究では $M=1$ として攻撃検出を行った．

4 結果

文献^[3] ^[4] ^[5]の情報をもとに作成した攻撃ジェネレーターによってランダムに生成される攻撃文と通常文に対して提案手法による判定を行い，その検知率を検証した結果，以下の表のように

なった．

この表での攻撃文セットとは，ジェネレーターによりランダムに生成された1000個の攻撃文であり，検知率とはそれを攻撃文と判定出来た確率を表している．通常文セットとは，攻撃分と同様にランダムに生成された800個の，攻撃文ではない通常文のことであり，それを通常文と判定出来た確率を表している．

ランダム生成された 攻撃文または通常文	検知率
攻撃文セット 1	93.35%
攻撃文セット 2	93.62%
攻撃文セット 3	93.45%
攻撃文セット 4	94.89%
攻撃文セット 5	94.92%
攻撃文セット 6	94.40%
攻撃文セット 7	94.97%
攻撃文セット 8	94.21%
攻撃文セット 9	95.31%
攻撃文セット 10	94.57%
通常文セット 1	89.73%

このような結果から，本研究の提案手法を用いることで精度の高い検出ができると同時に，様々な攻撃方法に対しての攻撃検出が有効であると言える．

参考文献

[1]Kumar, P and Pateriya, R.K. "A survey on SQL injection attacks, detection and prevention techniques" International Conference on Computing Communication & Networking Technologies, pp.1-5, 2012

[2]Takeshi Matsuda, Feature of SQL Injection Attacks and Zeta Distribution, FIT2013.

[3]SQL Injection Cheat Sheet,
<http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/>.

[4]SQL Injection Cheat Sheet,
http://websec.ca/kb/sql_injection/.

[5]SQL Injection Cheat Sheet,
<http://www.byakuya-shobo.co.jp/hj/moh/sqlinjectioncheatsheet.html>.