Vol.2015-CSEC-71 No.13
2015/12/4

機械学習を用いたクロスサイトスクリプティング(XSS)攻撃の検知に関する考察

梅原 章宏[†1], 松田 健[†2], 園田 道夫[†1], 水野 信也[†2], 趙 晋輝[†3]

概要：クロスサイトスクリプティング(XSS)攻撃は,HTML の入力部分などの脆弱性に対するサイバー攻撃の一種である.XSS 攻撃に用いられる入力は,正常な入力との判別が難しく,機械的な検知が難しいとされている.本研究では,XSS 攻撃に用いられる入力と正常な入力に対して,ASCII 文字の出現頻度を特徴量とした特徴抽出を行い,機械学習のアルゴリズムを用いて,特徴空間上で入力がどのように分布しているか調べ,攻撃入力と正常入力の分離を試みた.

キーワード：クロスサイトスクリプティング(XSS),特徴抽出,機械学習, SVM, SCW, Random Forest

Consideration on the Cross-Site Scripting Attacks Detection Using Machine Learning

AKIHIRO UMEHARA, TAKESHI MATSUDA, MICHIO SONODA, SHINYA MIZUNO, JINHUI CHAO

Abstract: Cross-Site Scripting attack is a kind of cyber attacks against vulnerabilities such as HTML and javascript. Input of Attack is characteristically, but it is difficult to distinguish a normal from an attack, automatically. In this study, we extracted the feature of Cross-Site Scripting attack and normal focusing on ASCII code, and investigated the distribution of the feature space by using the machine learning algorithm. Moreover we tried to classify of an attack and normal.

Keywords: Cross-Site Scripting(XSS), feature extraction, machine learning, Support Vector Machine (SVM), Soft-Confidence Weighted Learning (SCW), Random Forest

1. はじめに

近年インターネットの普及により,ネットバンキングなど個人情報を含む取引をインターネット上で行うことが増加している.それに伴い,Web サイト上に入力された個人情報を狙うサイバー攻撃も増加している.

クロスサイトスクリプティング(XSS)攻撃は,HTML の入力部分などにおける脆弱性に対して攻撃を行う,サイバー攻撃の一種である.従来の対策として,構文解析によるフィルタが提案・実現されている[1]が,XSS 攻撃に用いられる入力は,正常な入力との区別が難しく,機械的な攻撃検知が容易でない.

先行研究[2]では,XSS 攻撃に用いられる入力と正常な入力に対して特徴抽出を行い,生成した特徴ベクトルを用いて攻撃検知を試みた.結果としてある程度の攻撃検知が可能であったが,より攻撃と正常の入力を特徴空間上で分離できるような特徴抽出法を用いることで,性能の向上が期待された.

本研究では,XSS 攻撃に含まれる ASCII 記号に着目し,攻撃と正常の入力に対して新たな特徴抽出の手法を適用した.また,特徴抽出により生成した特徴ベクトルを機械学習アルゴリズムである Support Vector Machine(SVM), Random Forest, Soft-Confidence Weight Learning(SCW) を用いて学習,分類を行い,攻撃検出への影響を考察した.

以下,2 章では XSS 攻撃,3 章では実験に使用した機械学習アルゴリズムについての概要を述べる.4 章では実験データの作成法や提案する特徴抽出法の紹介など,攻撃検知実験の準備について記述した.5 章では攻撃検知実験の結果をまとめ,6 章では実験結果に対する考察を行っている.

2. XSS 攻撃

本章では,XSS 攻撃とそれに対する既存の対策手法について,概要を述べる.

2.1 XSS 攻撃

XSS 攻撃は脆弱性の存在する HTML の入力部分などに対して,不正なスクリプト文を入力することにより,本来開発者が意図しない動作を誘発させるサイバー攻撃である.主な被害として,Cookie 値を盗まれることによる成りすまし被害や,Web ページの改ざんが挙げられる.

例として,通信販売サイトのログインページに XSS の脆弱性が存在する場合,スクリプトを用いてセッション ID を

†1 中央大学大学院理工学研究科情報工学専攻
Departments of Information and System Engineering, Graduate School of Science and Engineering, Chuo University.
†2 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
Department of Computer Science, Faculty of Comprehensive informatics, Shizuoka Institute of Science and Technology.
†3 中央大学理工学部情報工学科
Departments of Information and System Engineering, Faculty of Science and Engineering, Chuo University.

奪うことにより,第三者が不正にログインすることが可能となる.これにより,クレジットカードや連絡先などの個人情報が流出し,不正に利用される恐れがある.

2.2 既存の対策手法

既存の攻撃検知手法として,ブラックリスト方式,ホワイトリスト方式がある.ブラックリスト方式は,あらかじめ指定した文字列が入力に存在する場合,その入力を拒否する手法である.ホワイトリスト方式は逆に,あらかじめ指定した文字列の入力のみを許可し,それ以外を拒否する手法である.どちらの手法も事前に処理を行う入力を指定する必要があるため,未知の攻撃に対しては対応することが難しい.

他の対策手法として, '<' や '&' といった HTML における特別な記号を別の文字記号で置き換えるエスケープ処理がある.これは XSS 攻撃の根本的な対策として非常に有効であるが,例外的な処理を行う記号が増加した際にプログラムの記述が煩雑になり,エスケープ処理を行う部分の記述に漏れが生じる可能性が高くなってしまう.

3. 機械学習アルゴリズム

本章では,実験で使用した機械学習アルゴリズムについて,概要を述べる.

3.1 Support Vector Machine (SVM)

SVM について,N個の要素からなるデータの分類は以下の式で定義される[3].

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b = \sum_{n=1}^{N} a_n t_n k(\mathbf{x}, \mathbf{x}_n) + b$$

ここで$\mathbf{x}$は入力ベクトル,$\mathbf{w}$は重みベクトル,ϕは特徴空間変換関数,a_nはラグランジュ乗数,t_nは目標値,$k(\mathbf{x}, \mathbf{x}_n)$はカーネル関数,bはバイアスパラメータである.

SVM は分類境界と最も近いデータとの距離(マージン)を最大化することで,汎化誤差が最小になるような分類境界を求める.マージンを最適化する解は,以下の目的関数を最小化することで得られる.

$$C\sum_{n=1}^{N} \xi_n + \frac{1}{2}\|\mathbf{w}\|^2$$
$$\text{s.t.} \quad t_n\big(y(\mathbf{x}_n)\big) \geq 1 - \xi_n, \quad n = 1, \ldots, N$$
$$\xi_n \geq 0$$

ここでCは誤分類に対するペナルティの大きさを制御するパラメータであり,大きいほど誤分類を許さないような分類境界を求める.またξ_nはスラック変数であり,$0 \leq \xi_n \leq 1$となるデータは正しく分類され,$\xi_n > 1$となるデータは誤分類されている.

3.2 Random Forest[4]

Random Forest は複数の決定木による集団学習を行う機械学習アルゴリズムの一つである.全ての学習データからランダムに抽出した要素を用いて決定木を作成し,それらの分類の結果から,最終的な出力を決定する.

Random Forest において,マージンは以下の式で定義される.

$$mr(\mathbf{X}, Y) = P_{\boldsymbol{\Theta}}(h(\mathbf{X}, \boldsymbol{\Theta}) = Y) - \max_{j \neq Y} P_{\boldsymbol{\Theta}}(h(\mathbf{X}, \boldsymbol{\Theta}) = j)$$

ここで,$\mathbf{X}$は入力ベクトル,Yは正解のラベル,jは正解以外のラベル,$\boldsymbol{\Theta}$はランダムに抽出した学習データ,$P_{\boldsymbol{\Theta}}$は汎化誤差,$h(\mathbf{X}, \boldsymbol{\Theta})$は決定木である.このマージンが大きいほど,Random Forest 内の各決定木における信頼度が大きくなる.

3.3 Soft-Confidence Weighted Learning (SCW)[5]

SCW は逐次学習型の機械学習アルゴリズムの一つであり,分類は以下の式で定義される.

$$y(x) = \mathrm{sgn}(\boldsymbol{\mu}_{t-1}\mathbf{x}_t)$$
$$\text{if } \boldsymbol{\mu}\mathbf{x} \geq 0 : y(\mathbf{x}) = 1$$
$$\text{else} \quad : y(\mathbf{x}) = -1$$

ここで$\mathbf{x}$は入力ベクトル,$\boldsymbol{\mu}$は重みの平均ベクトルであり,バイアスパラメータは存在しない.また,損失関数l^{ϕ}は以下の式であらわされる.

$$l^{\phi} = \max\left(0, \phi\sqrt{\mathbf{x}_t^{\mathrm{T}}\Sigma\mathbf{x}_t} - y_t\boldsymbol{\mu}\mathbf{x}_t\right)$$

ここでΣは共分散行列,$\phi = \Phi^{-1}(\eta)$である(Φは正規分布の累積密度関数,ηは誤差を許容する程度を表すパラメータである).

μ, Σの更新式は以下の最適化問題であらわされる.

$$(\boldsymbol{\mu}_{t+1}, \Sigma_{t+1}) = \arg\min_{\boldsymbol{\mu},\Sigma} D_{KL}\left(\mathcal{N}(\boldsymbol{\mu}, \Sigma)\middle\|\mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t)\right) + Cl^{\phi}(\mathcal{N}(\boldsymbol{\mu}, \Sigma); (\mathbf{x}_t, y_t))$$

ここでD_{KL}はカルバック情報量,$\mathcal{N}$は平均ベクトルμと共分散行列Σの多変量正規分布,Cは重みの更新を制御するパラメータである.

SCW は同じ逐次学習の機械学習アルゴリズムである CW[6]を改良したものであり,特徴としてデータの信頼度による重み付けをを行う.また,マージンの最大化を行うことで,CW の弱点であるノイズに弱い面を克服している.

4. 機械学習を用いた検知実験

本章では,攻撃検知実験で使用するデータの作成法,機械学習のプログラム,評価指標について述べる.

4.1 使用するデータの作成

実験を行うにあたり,文献[7]や Web ページ[8]などを利用し,URL 形式の攻撃入力 915 個,正常入力 1290 個を用意し,特徴抽出を行い特徴ベクトルを生成する.その中からランダムに攻撃入力 500 個,正常入力 500 個を抽出し,攻撃と正常の入力が入り混じった要素数 1000 個のデータを作成,それを一つのデータセットとした.

4.2 特徴抽出

特徴抽出は,ASCII コードに基づいて次の 2 種類の特徴ベクトルを作成した.

1. 攻撃,正常入力中の記号の出現頻度から,正常入力に多い記号(x_1)と攻撃に多い記号(x_2)を選択し,それらから作成した 5 次元特徴ベクトル(表 1).
2. 10 進 ASCII コードにおける 0-127 までのそれぞれの記号における,入力中の出現頻度を特徴とした 128 次元特徴ベクトル.

表 1　5 次元特徴ベクトル(x_1, x_2, x_3, x_4, x_5)

x_1	x_2	x_3	x_4	x_5					
/	<	文字	数字	!	"	#	$	&	'
.	>			(	)	*	+	,	[
_	=			¥¥	]	^	`	{	|
?	;			}	~	:	@		
%	SP			その他制御文字					

ここで,正常・攻撃に多い記号は,片方の出現頻度に対して,もう片方の出現頻度が約 2 倍程度となるものから選択した.

4.3 機械学習プログラム

実験に用いた機械学習アルゴリズムは SVM, Random Forest, SCW の 3 種類である.SVM, Random Forest の実行プログラムには,Python の機械学習ライブラリ scikit-learn0.15.2 の関数 SVC, RandomForestClassifier を使用した.以下に示した SVM のカーネル関数もこれに準拠するものである.

$$\text{ガウスカーネル: } \exp\left(\frac{1}{2\sigma^2}|x - x'|^2\right)$$

SCW の実行プログラムは,[5]に記載されたアルゴリズムを参考に Python で実装した.Python のバージョンは 2.7.8 である.

4.4 評価

評価を行うにあたって,特徴抽出を行ったデータセットを 5 分割し,交差確認を行った.SVM, Random Forest の評価項目として,以下のものを用いた.

1. データ全体に対して予測が正しかったものの割合(正解率,Accuracy)
2. 予測が実際に正しいものの割合(精度,Precision)
3. 真の結果に対して,その結果であると予測されたものの割合(再現率,Recall)
4. 精度と再現率の調和平均(F 値,F-measure)
5. ROC 曲線の下の面積(AUC)

SCW の評価項目としては,上記項目の 1 から 4 を用いた.

5. 結果

本章では,各機械学習アルゴリズムによる攻撃検知実験の結果について,記述する.

5.1 SVM

5 次元特徴ベクトル,128 次元特徴ベクトルにおいて,2 つのデータセットに対して SVM を用いた分類を行った.結果を表 2 に示す.今回の実験では,パラメータ$C = 100$,ガウスカーネルのパラメータ$\sigma = 10$とした.

表 2　SVM による実験結果

	5-dimension		128-dimension	
	dataset1	dataset2	dataset1	dataset2
正解率	0.963	0.977	0.982	0.989
攻撃精度	0.953	0.976	0.973	0.982
正常精度	0.973	0.978	0.992	0.996
攻撃再現率	0.975	0.978	0.992	0.996
正常再現率	0.953	0.976	0.972	0.982
攻撃 F 値	0.963	0.977	0.982	0.989
正常 F 値	0.963	0.977	0.982	0.989
AUC	0.964	0.977	0.982	0.989

傾向として,5 次元特徴ベクトルを用いた検知よりも,128 次元特徴ベクトルを用いた時の方が,検知結果が向上している.128 次元ベクトルの際の平均正解率は 98.5%程度となり,F 値や AUC も高い値を示している.

5.2 Random Forest

5 次元特徴ベクトル,128 次元特徴ベクトルにおいて,2 つのデータセットに対して Random Forest を用いた分類を行った.結果を表 3 に示す.今回の実験では,決定木の数は 40,木の深さの最大値は 4 とした.

表 3　Random Forest による実験結果

	5-dimension		128-dimension	
	dataset1	dataset2	dataset1	dataset2
正解率	0.963	0.969	0.991	0.992
攻撃精度	0.962	0.976	0.996	0.996
正常精度	0.964	0.962	0.986	0.988
攻撃再現率	0.964	0.963	0.986	0.988
正常再現率	0.963	0.976	0.996	0.996
攻撃 F 値	0.963	0.969	0.991	0.992
正常 F 値	0.963	0.969	0.991	0.992
AUC	0.964	0.969	0.991	0.992

SVM と同じように,5 次元特徴ベクトルよりも,128 次元特徴ベクトルを用いた際に,検知性能が向上する結果となった.128 次元ベクトルの際には,平均正解率が 99.15%程度となることから,ほぼ全ての入力に対して攻撃と正常を確実に分離できていることが分かる.

5.3 SCW

5 次元特徴ベクトル,128 次元特徴ベクトルにおいて,2 つのデータセットに対して SCW を用いた分類を行った.結果を表 4 に示す.今回の実験では,パラメータ$C = 100, \eta = 0.5$とした.

表 4　SCW による実験結果

	5-dimension		128-dimension	
	dataset1	dataset2	dataset1	dataset2
正解率	0.667	0.591	0.887	0.898
攻撃精度	0.493	0.748	0.931	0.902
正常精度	0.847	0.429	0.837	0.893
攻撃再現率	0.839	0.571	0.870	0.895
正常再現率	0.616	0.662	0.933	0.902
攻撃 F 値	0.599	0.643	0.891	0.898
正常 F 値	0.701	0.492	0.882	0.897

こちらは,SVM, Random Forest ほど正解率が高いとはいえないものの,特徴次元を 128 に増やすことで,飛躍的に結果が向上した.128 次元での平均正解率は,89.25%となっている.

6. 考察・まとめ

今回の実験では,どの機械学習アルゴリズムにおいても,5 次元特徴抽出より 128 次元特徴抽出を用いた場合の方が結果が向上することが分かった.特に,Random Forest では 128 次元特徴ベクトルでの平均正解率が 99%程度となり,その他の値も非常に良い結果となったことから,攻撃と正常の入力を特徴空間上で十分分離することができていると考えられる.

しかし,特徴次元数をむやみに増やすことが必ずしも最善であるとは限らないと考えられる.一つ考えられる問題点として,特徴次元数が増えることによる計算時間の増加がある.

機械学習アルゴリズムにおけるパラメータを有効に決定する主な手法として Grid Search がある.Grid Search はパラメータチューニング用のデータに対して,指定した複数のパラメータを用いて学習・分類を行い,最も良い結果が出るパラメータの組を総当たりで探す手法である.特徴次元が大きくなる場合,Grid Search における 1 回の学習ごとの計算時間が増加する可能性があり,結果としてパラメータの決定時間が大きく増加してしまうことが考えられる.

そのため,今後の課題として,より少ない特徴次元数で 128 次元特徴ベクトルを用いた際の検知結果と同程度,またはそれ以上の結果を出すことが考えられる.そのためには,5 次元の特徴抽出法をさらに改良していくことが必要である.各入力内の出現頻度について,着目する部分を変更する,特徴抽出前後のデータに対して加工を行うなど,より効果的な特徴抽出の方法を検討していくことが,今後の課題として挙げられる.

参考文献

[1]"IE8 Security Part Ⅳ: The XSS Filter - IEBlog - SiteHome - MSDN Blogs",http://blogs.msdn.com/b/ie/archive/2008/07/02/ie8-security-part-ⅳ-the-xss-filter.aspx,最終閲覧日:2015/1/7

[2]梅原 章宏, 松田 健, 園田 道夫, 水野 信也, 趙 晋輝,"線形分類器によるクロスサイトスクリプティング(XSS)攻撃の検知に関する考察",FIT2015 第 14 回情報科学技術フォーラム A-020

[3] Christopher M.Bishop,"PATTERN RECOGNITION AND MACHINE LEARNING",Springer(2006)

[4]Leo Breiman, "RANDOM FORESTS",Statistics Department University of California Berkeley, CA 94720(January 2001)

[5] Jialei Wang,Peilin Zhao,Steven C.H. Hoi,"Exact Soft Confidence-Weighted Learning",ICML(2012)

[6] Mark Dredze,Koby Crammer,Fernando Pereira,"Confidence-weighted Linear Classification",ICML,pp.264-pp.271(2008)

[7] Jeremiah Grossman,Robert "Rsnake" hansen,Petko "pdp" D.petkov,Anton Rager,Seth Fogie,"XSS ATTACKS",SYNGRESS(2007)

[8] "ページ一覧取得",http://tshinobu.com/lab/get-page-link/,最終閲覧日:2015/1/4