一种针对基于 SVM 入侵检测系统的毒性攻击方法

钱亚冠[1],卢红波[1],纪守领[2],周武杰[3],吴淑慧[1],雷景生[1],陶祥兴[1]
(1. 浙江科技学院理学院/大数据学院,浙江杭州 310023; 2. 浙江大学计算机学院,浙江杭州 310058;
3. 浙江科技学院信息与电子工程学院,浙江杭州 310023)

摘 要: 在机器学习被广泛应用的背景下,本文提出一种针对基于 SVM(Support Vector Machine)入侵检测系统的新颖攻击方法——毒性攻击. 该方法通过篡改训练数据,进而误导 SVM 的机器学习过程,降低入侵检测系统的分类模型对攻击流量的识别率. 本文把这种攻击建模为最优化问题,利用数值方法得到攻击样本. 通过包含多种攻击类型的 NSL-KDD 数据集进行实验,从攻击流量的召回率和精度这两个指标对攻击效果进行评估,与已有方法相比,实验结果表明本文方法可更有效地降低入侵检测系统的识别率. 本文希望通过该研究进一步认识针对机器学习的新颖攻击,为下一步研究对应的防御机制提供研究基础.

关键词: 机器学习; 支持向量机; 入侵检测; 毒性攻击; 双层优化
中图分类号: TP309.2 **文献标识码:** A **文章编号:** 0372-2112 (2019)01-0059-07
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2019.01.008

A Poisoning Attack on Intrusion Detection System Based on SVM

QIAN Ya-guan[1], LU Hong-bo[1], JI Shou-ling[2], ZHOU Wu-jie[3],
WU Shu-hui[1], LEI Jing-sheng[1], TAO Xiang-xing[1]
(1. *School of Science & Big Data Science, Zhejiang University of Science and Technology, Hangzhou, Zhejiang* 310023, *China*;
2. *College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang* 310058, *China*;
3. *School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou, Zhejiang* 310023, *China*)

Abstract: Machine learning is widely applied in various intelligent devices including intrusion detection systems (IDS). We propose a novel approach called poising attack on IDS based on SVM. This attack is to degrade detection rate of IDS by misleading the SVM learning process with poisoned training data set. We model the poisoning attack as an optimization problem and solve it with numerical approach to get poisoned data set. At last, NSL-KDD data including several real attacks is used in our experiments, and two measures of precision and callback are used to evaluate the effectiveness. The result shows the poisoning attack approach can significantly degrade the IDS performance. This study may further understand the possible new attacks on machine learning, and provide the basis for the next study of the corresponding defense methods.

Key words: machine learning; SVM; intrusion detection; poisoning attack; bilevel optimization

1 引言

以机器学习为代表的人工智能技术日趋成熟,逐渐被应用到入侵检测系统(Intrusion Detection System, IDS)等网络安全领域[1]. 由于入侵检测系统通常部署于极易遭到攻击的开放网络环境,也被称为对抗环境(Adversarial Environments)[2]. 当 IDS 被部署到这种对抗环境中时,对手可能首先攻击 IDS 自身的学习系统[3]. 因此,如何保护 IDS 自身的学习系统不被攻陷已成为一个重要的研究课题. Barreno M 等[3]提出毒性攻击(Poisoning Attacks)和探测攻击(Exploratory Attacks)这两类攻击策略. 其中毒性攻击是利用机器学习的训练机制和数据自适应特点,污染训练数据集,从而误导 IDS 分类模型往有利于对手攻击意图的方向生成.

支持向量机(Support Vector Machine, SVM)是一种基于统计学习理论的机器学习模型,相关研究表明 SVM 应用到入侵检测系统具有良好的识别能力[4,5]. 尽管 SVM 应用于入侵检测系统具有很多优势,但新研究

收稿日期:2017-11-09;修回日期:2018-05-30;责任编辑:孙瑶
基金项目:浙江省自然科学基金(No. LY17F020011, No. LY18F020012);国家自然科学基金(No. 61772466, No. 61672337, No. 11771399)

发现,像 SVM、神经网络这样的机器学习模型都存在安全脆弱点[6,7].

新一代的黑客可能谙熟机器学习原理,会利用机器学习自身的脆弱点攻击学习系统,不能寄希望于算法的保密性来保证系统的安全性,所以必须深入研究学习系统自身的脆弱点及对手可能的攻击策略[8].考虑到 IDS 被部署在风险极高的对抗环境中,因此本文针对基于 SVM 的 IDS 的可能攻击方法展开研究,为将来可能的防御策略提供科学依据.

目前在 IDS 的安全性上已有大量研究,而针对网络流量的毒性攻击研究包括 Barreno 等[3]提出的针对基于聚簇半径的异常检测模型,Rubinstein 等[9]提出的基于 PCA 子空间的异常检测模型,而本文主要研究针对基于 SVM 的 IDS 的毒性攻击方法. Biggio B 等[10]提出了一种针对 SVM 的毒性攻击方法,该方法把问题建模为最大化 SVM 合页损失(hinge loss)的优化问题,并在手写体识别中取得较好的攻击效果. 我们把毒性攻击建模为一个目标逼近代价和篡改代价之和为最小的优化问题. 由于数据篡改必须受 SVM 的训练过程的约束,反过来 SVM 的训练又是基于篡改后的数据,相互构成约束,被称为双层优化问题[11]. 针对 SVM 的线性约束条件,我们利用 KKT 条件将上述双层优化问题转化为易于求解单层优化问题. 最后,本文在入侵检测数据集 NSL-KDD 上,利用上述优化方法获得攻击样本,与 Biggio B 等[10]提出的方法进行了对比,实验结果表明本文提出毒性攻击可使大量攻击数据被 IDS 误判为正常流量.

2 基于 SVM 的 IDS

2.1 SVM 原理

对于数据集$\{\boldsymbol{x}_i, y_i\}, i=1,2,\cdots,N, \boldsymbol{x}_i$ 表示输入向量,$y_i=\{-1,+1\}$ 表示类别. SVM 的目标是寻找间隔最大的最优分离超平面. 该超平面表示如下

$$y=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b \tag{1}$$

其中,$\boldsymbol{w}, b$ 分别表示权重向量和偏置. 考虑到线性不可分的实际情况,引入松弛变量 ξ 将式(1)转化为如下二次规划问题

$$\begin{aligned}&\arg\min_{w}\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+C\sum_{i=1}^{N}\xi_i\\&\text{s.t.}\quad y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i+b)\geq 1-\xi_i,\\&\qquad \xi_i\geq 0, i=1,2,\cdots,N\end{aligned} \tag{2}$$

其中,C 为惩罚参数,ξ 为惩罚项. 当数据维度很大时,求解该问题时计算复杂度很高,为此将其转化为 Lagrange 对偶问题

$$\begin{aligned}&\arg\min_{\alpha}\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}y_i y_j \alpha_i \alpha_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j-\sum_{i=1}^{N}\alpha_i\\&\text{s.t.}\quad \sum_{i=1}^{N}y_i\alpha_i=0,\end{aligned}$$

$$0\leq\alpha_i\leq C \tag{3}$$

实际问题遇到的通常是线性不可分数据,因此需要将原始数据映射到高维空间 $\boldsymbol{x}\rightarrow\varphi(\boldsymbol{x})$ 获取线性超平面. 由于高维空间的计算复杂度会很高,可利用核函数 $K(\boldsymbol{x},\boldsymbol{x}')=\varphi(\boldsymbol{x})\varphi(\boldsymbol{x}')$ 降低计算复杂度,在原始数据空间得到非线性决策边界. 通过上述求解,从式(3)可以得到最终分类器的判决函数

$$f(\boldsymbol{x})=\operatorname{sign}\left(\sum_{i=1}^{N}\alpha_i y_i K(\boldsymbol{x}_i,\boldsymbol{x})+b\right) \tag{4}$$

2.2 IDS 的 SVM 方法[5]

基于 SVM 的 IDS 主要由 3 部分组成[5],分别为预处理器、SVM 分类器和决策系统组成. 预处理的功能是将流量数据处理成输入 SVM 的形式. SVM 分类器是 IDS 的核心组成,需要利用训练数据训练后获得性能良好的 SVM 分类器. 决策的功能是指对新到达的流量数据进行类别判断,标记为“正常”或者“入侵”.

整个 IDS 的工作过程分为两个阶段:训练阶段和检测阶段. 在训练阶段,根据已知的正常数据和异常数据按照式(2)式(3)来训练 SVM,最后得到支持向量和相应的参数. 在检测阶段,预处理器先将未知状态的流量数据处理成合适的形式,然后通过 SVM 分类器,根据判决函数式(4)做出最后的判断.

3 针对 SVM 的毒性攻击

3.1 攻击模型与策略

本文根据 Papernot N 等[12]提出的攻击模型,从攻击面、攻击能力和攻击目标三个方面进行设置. 攻击面包括整个数据处理流程的各个阶段:原始数据采集、数据预处理、模型训练、结果预测等,如图 1 所示. IDS 的学习系统通常采用在线学习方式,因此本文假设攻击面为数据采集阶段,通过构建伪造的毒性样本,利用 IDS 在线训练机制,影响 SVM 的决策边界. 攻击能力是指对手能获取的信息和采取的操作,根据攻击能力的大小分为黑箱攻击(black-box attacks)[7]和白箱攻击(white-box attacks). 本文假设对手仅知道 IDS 的分类器为 SVM,而不了解其他任何关于目标系统的信息,因此接近黑箱攻击. 攻击目标指破坏目标系统的机密性、完整性和可用性,本文的攻击目标为 IDS 系统的完整性,即把恶意流量误判为正常流量.

在上述攻击模型的基本假设下,本文采取如下的攻击策略:①通过向远程目标 IDS 系统重放流量数据,获得分类标签,建立与目标系统同分布的本地训练集;②利用本地数据集训练 SVM,以代理方式模拟目标系统的 SVM;③在本地代理 SVM 上生成攻击样本. 假设原始的流量数据空间表示为 $\mathscr{D}$,$\mathscr{D}$ 通过数据预处理后变换为特征空间 $\mathscr{X}$ SVM 是在特征空间 $\mathscr{X}$ 上进行训练,因

此本文提出的算法是基于特征空间 $\mathscr{X}$ 生成攻击样本 $\boldsymbol{X}\in\mathscr{X}$. 一旦获得 $\mathscr{X}$ 上的攻击样本 $\boldsymbol{X}$,将流量特征反变换为数据包数量、长度等参数,利用流量生成工具[13,14]很容易重建 $\mathscr{D}$ 空间上的攻击 TCP 流.

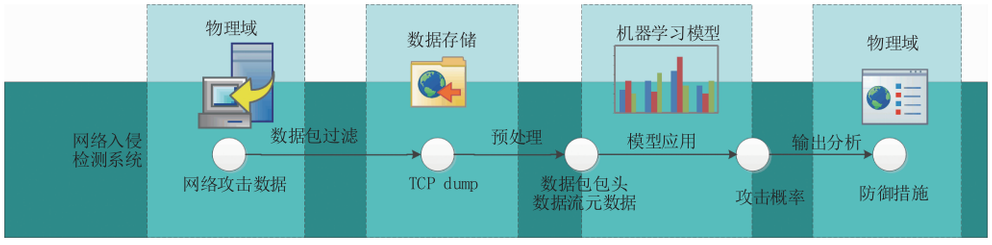

图1　基于机器学习的IDS的攻击面[12]

3.2　问题的建立

假设某一攻击流量企图躲避 SVM 的检测,它希望通过毒性攻击篡改训练数据,使得 SVM 的分离超平面发生改变. 我们把对手希望得到的分离超平面定义为攻击目标超平面 $\boldsymbol{w}^*$(用超平面的法向量表示). 毒性攻击的过程建模为 $\boldsymbol{w}$ 逼近 $\boldsymbol{w}^*$,且数据篡改代价最小的最优化问题,该优化问题的决策变量为训练数据 $\boldsymbol{X}_0\in\mathscr{X}$,目标函数为目标逼近代价和数据篡改代价. 其中,目标逼近代价定义为 $\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2$,数据篡改代价定义为 $\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2$,$\boldsymbol{X}\in\mathscr{X}$ 为篡改后的数据. 可以发现,数据篡改后并不能直接得到 $\boldsymbol{w}$,$\boldsymbol{w}$ 是 SVM 在 $\boldsymbol{X}$ 上的学习结果. 因此目标逼近代价受 SVM 优化学习的约束,反过来 SVM 的优化学习又受数据篡改后的 $\boldsymbol{X}$ 的约束. 两种优化互为约束,又成交替优化的过程,可以建模为如下的双层优化问题

$$\begin{aligned}&\arg\min_{\boldsymbol{X},\boldsymbol{w}}\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2+\frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2\\&\text{s.t.}\quad \arg\min_{\boldsymbol{w}}\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+C\sum_{i=1}^{N}\xi_i\\&\text{s.t.}\quad y_i(\boldsymbol{w}^{\mathrm{T}}x_i+b)\geq 1-\xi_i,\\&\qquad \xi_i\geq 0,i=1,2,\cdots,N\end{aligned}\tag{5}$$

这里 λ 是对篡改代价的惩罚系数. 从式(5)可以发现,初始状态是原始训练数据 $\boldsymbol{X}_0$ 及在 $\boldsymbol{X}_0$ 上训练得到的 SVM 权重向量 $\boldsymbol{w}_0$. 对手将 $\boldsymbol{X}_0$ 篡改为 $\boldsymbol{X}_1$,SVM 从 $\boldsymbol{X}_1$ 训练得到 $\boldsymbol{w}_1$,如此反复,最终使 $\boldsymbol{w}_0$ 逼近 $\boldsymbol{w}^*$ 的交替优化的过程.

3.3　数值方法

由于上述双层优化问题是 NP 难问题[11],因此,本文通过 KKT 条件将双层优化转换成单层优化进行求解,式(6)给出了单层优化的表示方法.

$$\begin{aligned}&\arg\min_{\boldsymbol{X}}\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2+\frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2\\&\text{s.t.}\quad \sum_{i=1}^{N}\alpha_i y_i\mathrm{I}(y_i(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{w}+b)\leq 1)x_{ij}=\boldsymbol{w}\end{aligned}\tag{6}$$

其中,$\mathrm{I}(z)$ 为指示函数,当 z 为真时 $\mathrm{I}=1$,否则 $\mathrm{I}=0$.

上述问题是一个典型的凸优化问题,可以采用梯度下降算法求解. 由于优化变量为数据集,需要通过如下处理才能进行求解. 根据式(6)可以得到如下优化迭代式

$$\boldsymbol{X}_{t+1}=\boldsymbol{X}_t-\alpha_t\nabla_{\boldsymbol{X}}\left(\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2+\frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2\right)\tag{7}$$

考虑到上式无法对 $\boldsymbol{X}$ 直接求导,因此采用链式法则表示如下

$$\nabla_{\boldsymbol{X}}(\cdot)=\nabla_{\boldsymbol{w}}\left(\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2+\frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2\right)\frac{\partial\boldsymbol{w}}{\partial\boldsymbol{X}}\tag{8}$$

式(8)中 $\partial\boldsymbol{w}/\partial\boldsymbol{X}$ 则由 SVM 的优化过程得到. 为了更好地说明上述生成攻击样本的优化迭代过程(如图 2 所示),在二维平面上随机生成数据点,“○”表示原始正例,“●”表示已被篡改的正例,“×”表示原始负例,“+”表示被篡改的负例. 假设在初始状态时,所有的数据点都已被正确分类,即超平面能将每个数据分类为 +1 或者 −1(图 2(a)所示). 假设攻击目标是使某些特定的负例(图中用☆表示)变为正例(★表示),即使最后的分离超平面平行于 X 轴. 考虑到实际情形,又假设对手只能篡改负例. 训练数据受到毒性污染之后,由图 2 可观察到超平面的变化:数据的毒性污染是个迭代过程,SVM 生成的超平面随着变化也进行不断的调整,超平面不断趋于平坦,接近 X 轴,最终逼近攻击目标. 算法过程如算法 1 所示.

算法 1　攻击样本生成算法(BOPA)

输入:$\boldsymbol{X}_0$ 是原始正常数据,y 为类标签;n 是迭代总次数;λ 是篡改惩罚参数;$\boldsymbol{w}^*$ 为攻击目标;
输出:$\boldsymbol{X}$ 是攻击样本,r 是目标逼近代价,e 是篡改代价步骤:
1:　初始化攻击样本 $\boldsymbol{X}\leftarrow\boldsymbol{X}_0$
2:　for $t\leftarrow 1,2,\cdots,n$ do
3:　$\boldsymbol{w}\leftarrow$ train_svm($\boldsymbol{X},y$) //训练支持向量机
4:　if $\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2<0.001$ then
5:　$r\leftarrow\frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}^*\|_2^2$
6:　$e\leftarrow\frac{\lambda}{2}\|\boldsymbol{X}-\boldsymbol{X}_0\|_F^2$
7:　return ($\boldsymbol{X},r,e$)

8： end if

9： $\alpha \leftarrow 1/t$;

10： $\boldsymbol{X} \leftarrow \boldsymbol{X} - \alpha \nabla_{\boldsymbol{X}} \left(\frac{1}{2} \| \boldsymbol{w} - \boldsymbol{w}^* \|_2^2 + \frac{\lambda}{2} \| \boldsymbol{X} - \boldsymbol{X}_0 \|_F^2 \right)$

11： end for

12： $r \leftarrow \frac{1}{2} \| \boldsymbol{w} - \boldsymbol{w}^* \|_2^2, e \leftarrow \frac{\lambda}{2} \| \boldsymbol{X} - \boldsymbol{X}_0 \|_F^2$

13： return $(\boldsymbol{X}, r, e)$)

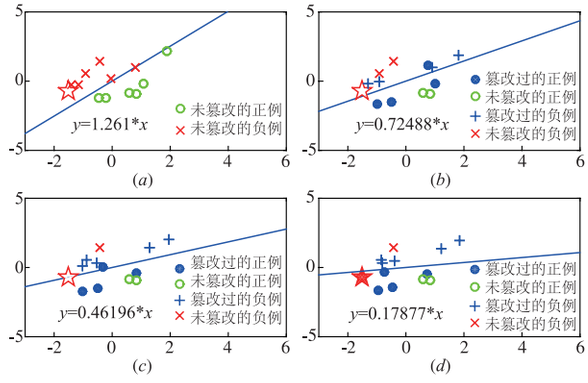

图2 训练数据受攻击后的超平面（此处是直线）变化过程

4 实验

Biggio B 等[10]提出的毒性攻击方法在初始训练集中注入特定的数据达到攻击效果，本文称之为 BPA（Biggio's Poisoning Attacks）方法. 它把 SVM 的攻击建模为"合页误差"的最大化问题，采用梯度上升方法求解，仅对一个攻击点进行迭代构造. 本文方法的不同之处在于将流量数据矩阵视为决策变量，实现整体构造攻击样本；不仅考虑分类误差，同时考虑代价的最小化. 方便起见，把本文提出的方法称为 BOPA 方法（Bi-level Optimization Poisoning Attacks）. 通过公开数据集 NSL-KDD[15]对两个方法进行了实验对比分析.

4.1 NSL-KDD 数据集

KDD-CUP-99 数据集广泛用于测试 IDS 系统，但大量分析发现其存在冗余、有偏等缺陷[16]. NSL-KDD 数据集则是其改进，包含了它的重要记录，并去除冗余，具有数据量大、无偏等优点.

NSL-KDD 数据集分训练集和测试集，其中训练集包含 125973 条记录，测试集包含 22544 记录. 每条记录有 42 个特征，其中第 42 个特征为类标签，标记为正常类型（normal）或者具体的攻击方法. 这些攻击方法可进一步归属到 DoS、Probe、R2L 和 U2R 等四种攻击大类. 本文实验包括训练用的 KDDTrain + . TXT（125973 条记录）和测试用的 KDDTest + . TXT（125973 条记录）两个数据集. 数据集中的每一条记录包含 42 个特征，特征类型包含 nominal、binary 和 numeric. 所选数据集中正常类型和攻击大类的记录比例如表 1 所示.

表 1 正常类型、攻击大类的数据统计

数据集	Normal	DoS	Probe	U2R	R2L
KDDTrain +	53.46%	36.46%	9.25%	0.04%	0.79%
KDDTest +	43.08%	33.08%	10.74%	0.89%	12.22%

在实验中，我们选择 DoS 攻击类中所占比例最大的 neptune，smurf 及 Probe 攻击类中的 satan，nmap，portsweep，共计 5 种攻击方法.

4.2 生成攻击样本

本文将攻击前、后的样本分别称为原始样本和攻击样本. 为生成攻击样本，首先对原始样本 NSL-KDD 数据集中的 nominal 和 numeric 数据进行标准化处理. 对 nominal 型数据，我们通过计算各个 nominal 值所占的比例，将比例作为新的特征值. 然后对数据集中的 numeric 及数值化后的 nominal 数据作标准化处理，即将数据集中每个特征转化为均值为 0，标准差为 1 的数值型变量. 考虑到数值可计算性，剔除特征值为 0 的比例超过 99% 的特征，共计 14 个. 最后我们实际实验所用的数据的特征维度为 27.

实验采用 LIBLINEAR SVM 作为标准的支持向量机学习算法. 首先通过交叉验证确定正则化参数 C 为 1，迭代次数为 1000 次，在原始数据集 $\boldsymbol{X}_0$ 上获得初始 SVM. 毒性攻击样本的生成过程和支持向量机的训练过程交替进行，这样一次交替优化过程我们称之为一个迭代期. 第 t 个迭代周期得到训练好的 SVM 的权重向量 $\boldsymbol{w}$，并用该权重向量为参数优化第 t 个迭代周期的攻击样本. 迭代周期通常在 30 次以上能收敛到很小的值，所以我们在实验中设置迭代周期为 50 次. 然后确定毒性攻击目标的权重向量 $\boldsymbol{w}^*$，具体方式如下：①获取数据集中正常流量和单一攻击（如 neptune）合成子集；②采用贪心策略选择若干个有效的特征，即与攻击目标相关性最强的特征；③根据攻击的数据分布设置阈值 τ，高于阈值的记录的类标签改为 +1，否则为 −1；④将标签重置后的数据集进行训练，得到的权重向量即为 $\boldsymbol{w}^*$.

实验分析以 neptune 攻击为例，限于篇幅，satan、smurf、nmap、portsweep 等多种攻击具有类似的实验结果. 从实际情况考虑，期望得到的攻击样本：一方面正常类型的分类准确率在较高水平，避免被 IDS 觉察；另一方面能显著降低 neptune 攻击的检出率，达到躲避检测的目的. 显然目标函数中的惩罚系数 λ 和搜索步长 α 起着关键作用. 不同惩罚力度对算法的敏感度大为不同，我们取多个 $\lambda = 0.05$、0.25、0.45 进行实验，利用交替优化的方式尽可能使支持向量机得到的 $\boldsymbol{w}$ 能够逼近 $\boldsymbol{w}^*$，逼近代价的变化随迭代过程的趋势变化如图 3 所示. 由图 3 可发现，λ 取 0.45 时，逼近代价在前期振荡收敛不稳定，易受篡改代价的干扰. 而 λ 取 0.05 和 0.25 时更容易稳定收敛，但

λ 取 0.05 的收敛速度更快.

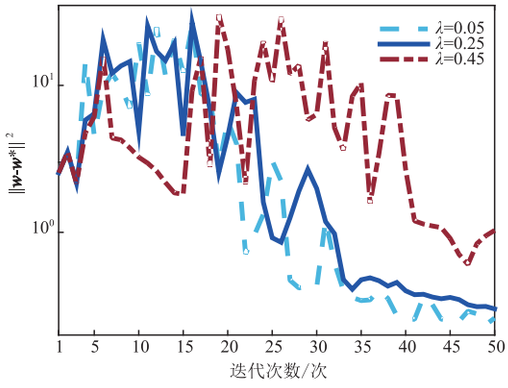

图3　不同λ下逼近代价的收敛趋势

对于搜索步长 α 采用自适应的方式设定. 由图 4 发现,不同的步长设置方案,目标逼近代价(Y 轴)的下降趋势不同. 当 α 取很小的固定值 0.01 时,代价值在前 40 个迭代周期几乎保持不变,前 80 次中也仅小幅下降,而在第 90 个迭代期开始不降反升,逼近代价大幅度上升. 当 α 取较大的固定值 0.7 时,逼近代价在前 40 个迭代期有良好的下降趋势,逐渐收敛. 然而从第 80 个迭代期左右开始,代价再次上升,缺乏收敛稳定性. 上述收敛不稳定性的原因,一方面过大的步长容易跨过最优值,另一方面过小的步长易被目标函数中的篡改代价干扰. 实验发现,当 α 的取值随迭代次数逐渐减小,因此取 $\alpha = 1/t$,逼近代价会以较稳定的方式下降,最终收敛到极小值.

图 5、图 6 是 BPA 方法从训练集的一个样本点开始构造攻击样本的过程. 由于初始样本点的选择具有随机性,因此我们进行多次实验,对不同的初始样本点选择结果进行对比. 为保持数据一致性,将实验结果缩放到[0,100]的范围. 从图 5 可看出随着迭代构次数的增加,得到的 SVM 分类误差变化趋势. 可以发现,不同的初始点选择都能使构建的攻击样本让 SVM 分类误差达到最大值. 图 6 给出了不同初始样本点下的算法收敛

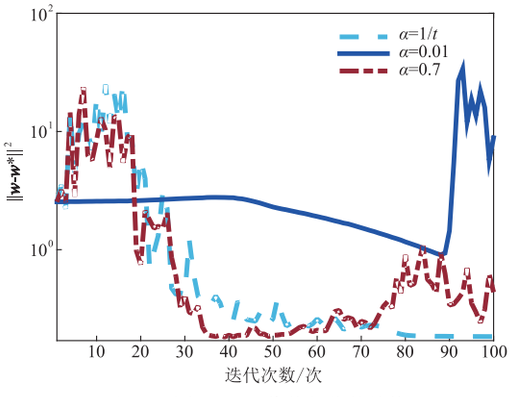

图4　不同α下逼近代价的收敛趋势

过程. 由此可以发现,BPA 方法和本文提出的 BOPA 方法均能比较稳定的构建出攻击样本.

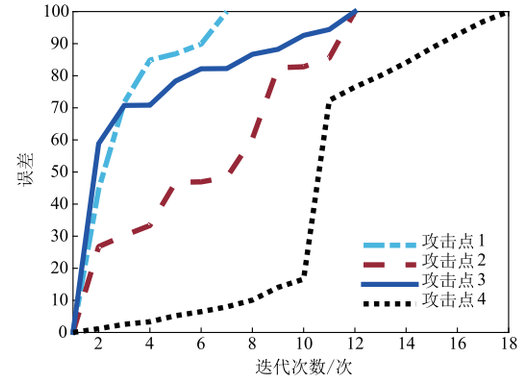

图5　不同攻击点的误差趋势图

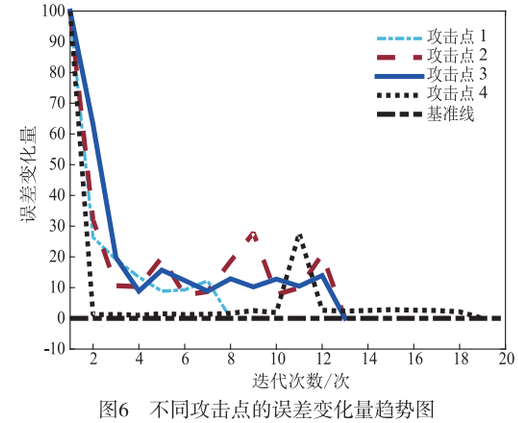

图6　不同攻击点的误差变化量趋势图

4.3　攻击效果分析

本文从攻击样本训练 SVM 的效果和逼近目标两个方面评价. 首先将原始样本和攻击样本分别训练 SVM,用相同测试集测试 SVM 的效果. 评估指标采用召回率(recall)与精度(precision):

$$\text{recall} = \text{TN}/\text{N} \tag{9}$$

$$\text{precision} = \text{TN}/(\text{TN} + \text{FN}) \tag{10}$$

设定正常、攻击流量分别为正、负例. N 为测试集标识为负例数,TN,FN 分别为正确预测为负例,错误预测为负例个数. 攻击 SVM 的目的是使攻击流量不被检测到,同时不影响正常流量通过. 上述指标是保证正常流量不被错误标记,即不增大式(10)的 FN,精度在较高水平;而对手期望攻击流量标记为正常,即式(9)TN 变小,从而降低召回率.

表 2 列出了攻击前后 SVM 召回率与精度,本文提出的 BOPA 方法与 Biggio B[15] 提出的 BPA 方法在攻击效果上相比具有优势. 首先,BPA 方法在分类精度上表现不稳定,例如,在 neptune 攻击的分类精度从 86.7% 上升到 98.44%,而 nmap 攻击却从 93.93% 降至 55.19%. 而 BOPA 方法在各种攻击上精度均有提升,这

说明攻击后不但实现攻击,而且不影响正常流量通过,比 BPA 方法更稳定. 两者在召回率上均出现下降,表明一定数量的攻击流量未被 IDS 正确识别,但 BOPA 方法比 BPA 方法的召回率下降显著,尤其在 smurf 攻击中,BOPA 方法从 98.10% 下降到 58.18%,而 BPA 方法仅降至 98.36%. 这说明采用本文的毒性攻击方法后,有大量攻击未被 IDS 检测到,效果上优于 BPA 方法. 其次,不同攻击类型对原始样本训练的 SVM 在精度上相差较为明显,neptune 和 portsweep 分别为 86.70% 和 93.78%,相差超过 7%,而在攻击样本中仅 0.64%,相应的在 BPA 方法中仅相差 2.03%,由式(10)得,FN 的值很小,即正常流量被误判为攻击样本的比例很小. 而在召回率上则不然,两者差距分别为 10.64%、9.85%,表明攻击样本与原始样本虽然在数据上进行篡改,但 SVM 在正例预测为负例的比例仍维持在原始样本的水平上. 综合来看,训练 SVM 的攻击样本具有迅速降低召回率,同时维持正常流量不被影响的特性,实现了攻击目标,并在攻击效果上明显优于 BPA 方法.

表 2 原始样与攻击样本训练的 SVM 在测试集上的分类性能比较

攻击类型	原始样本训练的 SVM		BOPA 攻击样本训练的 SVM		BPA 攻击样本训练的 SVM	
	召回率	精度	召回率	精度	召回率	精度
neptune	99.06%	93.78%	74.47%	93.78%	74.47%	98.44%
smurf	98.10%	93.85%	93.86%	93.85%	93.86%	82.89%
satan	93.78%	94.01%	87.05%	94.01%	87.05%	98.06%
portsweep	88.42%	94.42%	70.55%	94.42%	70.55%	96.41%
nmap	90.25%	94.37%	34.93%	94.37%	34.93%	55.19%

图 7 是从逼近攻击目标的角度对攻击效果进行评价. 从 satan、smurf、nmap、portsweep 攻击的目标逼近代价趋势程度看,代价下降趋势明显,不同 λ 对该攻击类型最终都能够收敛到 0 值附近,实现攻击目标. nmap 攻击对 λ 敏感,在 λ 取较大值 0.25 和 0.45 时收敛过程不稳定,难以保证收敛结果,而在 λ 取较小值 0.05 时,收敛过程明显有利于实现最终的攻击目标. portsweep 攻击与 nmap 类似,当 λ 取较小值 0.05 时能较快的稳定收敛到攻击目标值. 综上所述,针对不同攻击类型,设置合理的自由参数,都可实现预期的攻击效果.

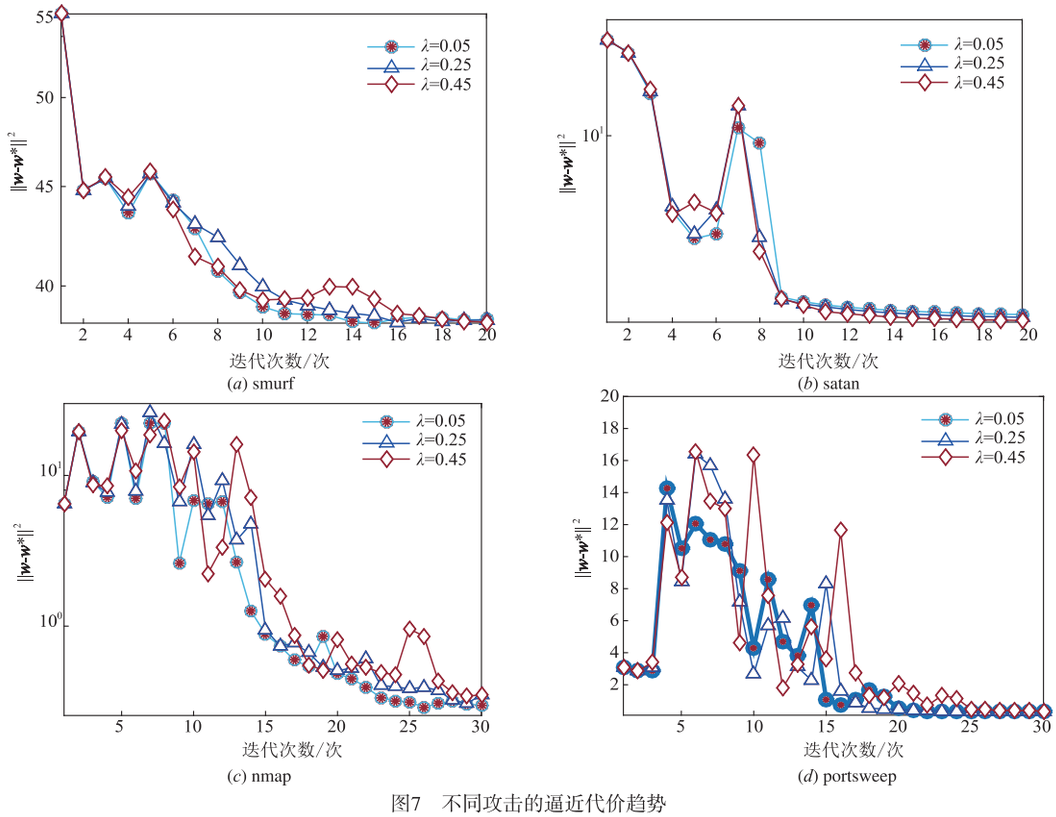

图7 不同攻击的逼近代价趋势

5 结论

本文基于最优化理论对 SVM 毒性攻击进行建模,并在 NSL-KDD 数据集模拟对 IDS 的攻击,利用独立的测试集进行了攻击效果的评估. 基于 SVM 的 IDS 在正常条件下的检测准确率超过 99%. 当对其进行毒性攻击时,系统的检测率会受到严重影响,表现在攻击数据的召回率大幅降低,即假阴性率明显增加,大量攻击数据被误判为正常流量. 实验显示被毒性攻击后的训练数据建立的 SVM 模型的检测准确率下降程度最高接近

40%,比 Biggio B 等[15]提出的方法攻击效果更好,实验证明了其攻击的有效性.

参考文献

[1] SOMMER R, PAXSON V. Outside the closed world: Onusing machine learning for network intrusion detection[A]. IEEE Symposium on Security and Privacy(SP)[C]. USA: IEEE, 2010. 305 – 316.

[2] ZHANG R, ZHU Q. Secure and resilient distributed machine learning under adversarial environments[A]. Proceedings of the 18th International Conference on Information Fusion (Fusion)[C]. USA: IEEE, 2015. 644 – 651.

[3] BARRENO M, NELSON B, SEARS R, et al. Can machine learning be secure? [A]. Proceedings of the ACM Symposium on Information, Computer and Communications Security[C]. USA: ACM, 2006. 16 – 25.

[4] 高妮,高岭,贺毅岳,等. 基于自编码网络特征降维的轻量级入侵检测模型[J]. 电子学报, 2017, 45(3): 730 – 739.
GAO Ni, GAO Lin, HE Yi-yue, et al. Light weight intrusion detection model based on dimension reduction of self-coding network features[J]. Acta Electronica Sinica, 2017, 45(3): 730 – 739. (in Chinese)

[5] 尚文利,张盛山,万明,等. 基于 PSO-SVM 的 Modbus TCP 通讯的异常检测方法[J]. 电子学报, 2014, 42(11): 2314 – 2320.
SHANG Wen-li, ZHANG Sheng-shan, WAN Ming, et al. Abnormal detection method of Modbus TCP communication based on PSO-SVM[J]. Acta Electronica Sinica, 2014, 42(11): 2314 – 2320. (in Chinese)

[6] XIAO H, BIGGIO B, NELSON B, et al. Support vectormachines under adversarial label contamination[J]. Neuro Computing, 2015, 160(C): 53 – 62.

[7] PAPERNOT N, MC-DANIEL P, GOODFELLOW I, et al. Practical black-box attacks against machine learning[A]. Proceedings of the ACM on Asia Conference on Computer and Communications Security[C]. USA: ACM, 2017. 506 – 519.

[8] MC-DANIEL P, PAPERNOT N, CELIK Z B. Machine learning in adversarial settings[J]. IEEE Security & Privacy, 2016, 14(3): 68 – 72.

[9] RUBINSTEIN B I P, NELSON B, HUANG L, et al. ANTIDOTE: Understanding and defending against poisoning of anomaly detectors[A]. ACM SIGCOMM Conference on Internet Measurement[C]. USA: DBLP, 2009. 1 – 14.

[10] BIGGIO B, NELSON B, LASKOV P. Poisoning attacks against support vector machines[A]. International Coference on International Conference on Machine Learning[C]. USA: Omnipress, 2012. 1467 – 1474.

[11] COLSON B, MARCOTTE P, SAVARD G. An overview of bilevel optimization[J]. Annals of Operations Research, 2007, 153(1): 235 – 256.

[12] PAPERNOT N, MCDANIEL P, SINHA A, et al. Towards the Science of Security and Privacy in Machine Learning[OL]. http://arxiv.org/abs/1611.03814v1, 2016.

[13] BEHAL S, KUMAR K. Characterization and comparison of DDoS attack tools and traffic generators: A review[J]. International Journal of Network Security, 2017, 19(3): 383 – 393.

[14] WEIGLE M C, ADURTHI P, HERNÁNDEZ-CAMPOS F, et al. Tmix: A tool for generating realistic TCP application workloads in ns-2[J]. ACM SIGCOMM Computer Communication Review, 2006, 36(3): 65 – 76.

[15] NSL-KDD 数据集[OL]. http://nsl.cs.unb.ca/NSL-KDD, 2017-11-5/2018-5-29.

[16] DHANABAL L, SHANTHARAJAH S P. A study on NSL-KDD dataset for intrusion detection system based on classification algorithms[J]. International Journal of Advanced Research in Computerand Communication Engineering, 2015, 4(6): 446 – 452.

作者简介

钱亚冠(通信作者)　男. 1976 年生于浙江嵊县. 博士、副教授. 主要研究方向为机器学习的安全性、人工智能与深度学习.

卢红波　男. 1993 年生于浙江宁波. 硕士研究生,研究方向为对抗性机器学习、基于深度学习的图像处理.

纪守领　男. 1986 年生于山东菏泽. 博士、研究员. 研究方向为人工智能安全、数据驱动安全、隐私保护.