

网络钓鱼 Web 页面检测算法

郭敏哲, 袁津生, 王雅超

(北京林业大学信息学院, 北京 100083)

摘 要: 网络钓鱼(Phishing)攻击在电子商务和电子金融中普遍存在。该文分析 Phishing 页面敏感特征, 提出一种防御 Phishing 攻击的 Web 页面检测算法。该算法通过分析 Web 页面的文档对象模型来提取 Phishing 敏感特征, 使用 BP 神经网络检测页面异常程度, 利用线性分类器判断该页面是否为 Phishing 页面。该算法成功过滤了 Phishing 页面, 有效地阻止了 Phishing 攻击。

关键词: 网络钓鱼; Web 页面检测算法; BP 神经网络

Phishing Web Page Detection Algorithm

GUO Min-zhe, YUAN Jin-sheng, WANG Ya-chao

(School of Information, Beijing Forestry University, Beijing 100083)

【Abstract】 With the rapid development of online business and online banking, Phishing attacks are becoming increasingly rampant. This paper proposes an algorithm to detect Phishing Web page. The algorithm extracts anomaly features from Web page based on DOM model, then uses Back Propagation(BP) neural networks to detect the abnormal levels of the page, and applies linear model for page classification. The implementation of the algorithm successfully detects most Phishing pages, protecting users from Phishing attacks effectively.

【Key words】 Phishing; Web page detection algorithm; Back Propagation(BP) neural network

1 概述

网络钓鱼(Phishing)攻击是一种在线身份伪造的欺诈方式, 使用社交工程和技术托词等攻击手段窃取客户的个人身份数据和金融账号等敏感信息。Phishing攻击给用户和机构带来经济损失, 使其不再信任电子商务和电子金融服务。目前, 对Phishing防御技术的研究大体上归为4类: Phishing攻击成功的研究, 如何教育用户的研究, 更安全用户界面的研究和Phishing攻击自动检测技术的研究^[1-2]。

Phishing页面防御技术是Phishing防御方案的一个重要组成部分。当前提出的防御技术大多采用黑名单或利用从以往攻击案例中归纳的特征进行Web页面过滤和检测。这些技术能够成功地检测已知攻击, 但攻击者对以前攻击特征稍加改动就可较容易地绕过这些防御机制^[1]。本文提出一种基于异常检测思想的Phishing页面检测算法, 通过分析Web页面中与Phishing攻击实现机制密切相关的数据, 概括合法站点正常页面中这些数据的特征和状态。从这些数据特征角度来分析, Phishing页面和正常页面就会产生偏差, 出现异常。本文算法独立于具体的Phishing攻击行为, 能够有效地检测以往攻击的变体甚至新的攻击。

2 Phishing 页面分析

Phishing 页面是本文算法所要防御的对象, 由 Phishing 攻击者构造, 并伪装成合法站点, 骗取用户信任使其泄露敏感信息的 Web 页面。

通过对 Phishing 攻击案例的研究, 可以发现 Phishing 页面通常存在 3 点共性:

(1) Web 页面包括地址栏信息、页面布局风格和商标图案等, 这些与合法站点的页面极其相似。

(2) Web 页面内容要求用户提交敏感信息并提供信息输

入的场所。

(3) Web 页面存在某种攻击机制, 并获取用户敏感信息, 比如: 修改表单提交给目的地, 使用恶意软件监视用户输入。

因此, Phishing 页面存在异常, 但直观上难以发现, 但可以从 Web 页面的文档对象模型(DOM 模型)和 HTTP 协议信息的角度进行分析。Phishing 页面主要异常如下:

(1) Web 页面 URL 地址异常

一个 Web 页面的 URL 地址是唯一的。对于正常页面, URL 地址通常包含与页面所声称的所有者相关的字符串;而对于 Phishing 页面, 其 URL 地址可能直接就是 IP 地址, 或者使用不易分辨的与合法域名类似的字符串作为域名。

(2) 链接对象异常

针对页面 DOM 模型中的<a>对象。对于正常页面, 大部分<a>对象应该指向页面文件所在的域内, 并且这个域与页面所声称的所有者所在的域一致。而对于 Phishing 页面, 存在较多数量<a>对象的指向异常, 比如空指向, 或者指向不一致的域。

(3) 表单异常

针对页面 DOM 模型中的<form>等可提交数据的对象。对于正常页面, 大部分<form>元素的 action 属性应该指向页面文件所在的域内, 并且这个域与页面所声称的所有者所在的域一致。对于 Phishing 页面, 存在较多数量<form>对象的 action 属性的指向异常, 比如空指向, 或者指向不一致的域。

(4) 资源引用异常

针对页面 DOM 模型中的等可引用资源的对象。正

作者简介: 郭敏哲(1982-), 男, 硕士研究生, 主研方向: 网络安全; 袁津生, 教授; 王雅超, 硕士

收稿日期: 2007-12-21 **E-mail:** gmz881@gmail.com

常页面所引用的资源绝大部分来自页面文件所在的域内，并且这个域与页面所声称的所有者所在的域一致。而 Phishing 页面则存在相当一部分资源的来源异常，比如与页面文件不在同一个域内等。

(5) 域名信息异常

域名信息指页面所在域的注册信息。对于正常页面，域名注册信息通常较容易查询并且域名注册时间通常在较久之前。而对于 Phishing 页面，其域名通常是近期注册的或者查询不到其注册信息。

3 Phishing 页面检测算法

3.1 算法目的

算法的目的就是要求不但能够成功检测所有已知攻击的 Phishing 页面，而且能够检测出以往 Phishing 页面攻击的新的变体。通过第 2 节的分析可以发现，页面文档对象模型中某些对象和页面包含的某些 HTTP 协议数据与 Phishing 攻击的成功与否密切相关，Phishing 页面的攻击机制通常分布在这些对象和数据中，因此，可以将这些对象和数据称为 Phishing 敏感特征。算法对页面提取 Phishing 敏感特征，认为这些特征就可代表当前页面中与 Phishing 页面攻击相关的信息，将其作为算法检测的主要依据，同时这些敏感特征与具体的 Phishing 页面攻击无关。算法将从合法站点页面和以往的 Phishing 页面中概括出敏感特征的正常状态模型。当敏感特征中出现异常时，算法检测出其异常程度，判断待测页面是否为 Phishing 页面。

3.2 Phishing 页面检测算法的设计

Phishing 页面检测算法流程如图 1 所示。

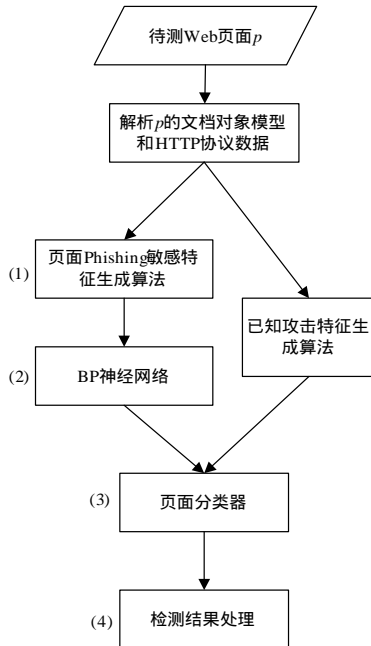


图 1 Phishing 页面检测算法流程

符号 p 表示一个Web页面。对于算法来说，一个Web页面 p 包括网络协议头部数据、HTTP协议头部数据、页面文档对象模型。其中， I 表示所有Web页面的集合； N 表示正常Web页面的集合； A_k 表示已知攻击的Phishing页面的集合； A_u 表示未知攻击的Phishing页面的集合，公式为

$$I = N \cup A_k \cup A_u \quad (1)$$

使用 c_p 表示Web页面 p 是正常页面($c_p=0$)还是Phishing页

面($c_p=1$)。 p 即为Phishing页面检测算法的输入， c_p 即为算法的输出。Phishing页面检测算法主要由Web页面敏感特征生成算法 F 、BP神经网络检测算法 S 、已知攻击特征生成算法 H 和页面分类算法 L 4 部分组成。Phishing页面检测算法为

$$c_p = L(S(F(p)), H(p)), p \in I, c_p \in \{0, 1\} \quad (2)$$

3.2.1 Web 页面 Phishing 敏感特征生成算法

在图 1 中，Phishing 页面检测算法步骤(1)提取页面 Phishing 敏感特征。步骤的输入为 Web 页面 p ，输出为 x 维敏感特征向量 V 。

算法共从Web页面中提取 8 个Phishing页面的敏感特征，特征提取对象为图片对象（如），链接对象（如<a>），框架对象（如<frame>，<iframe>等），嵌入式对象（如<script>，<applet>等），数据提交对象（如<form>等），页面URL地址，SSL证书，以及域名信息。使用特征函数来表示敏感特征，一共 8 个特征函数 f_1, f_2, \dots, f_8 ，每个特征函数的输出为实数值，表示待测页面中对应敏感特征的状态。以链接对象为例，其特征函数为

$$f_2 = \begin{cases} (L_{null} + L_{real})/L_{all} & L_{null} + L_{real} - L_{local} > 0 \\ 0 & L_{all} = 0 \\ -L_{local}/L_{all} & L_{local} > L_{null} + L_{real} > 0 \end{cases} \quad (3)$$

其中， L_{all} 代表页面中链接对象的总数； L_{null} 代表页面中空链接的个数； L_{real} 代表页面中指向真实站点链接的个数； L_{local} 代表页面中指向本域的链接的个数。此特征函数所要表达的是Web页面中链接对象的异常程度。

则Phishing页面检测算法的第(1)步页面敏感特征生成算法 $F=(f_1, f_2, \dots, f_8)$ ，输入值 p 经过 F 计算，得到一个 8 维的敏感特征向量 $V=\langle v_1, v_2, \dots, v_8 \rangle$ ，其中， $v_i=f_i(p)$ 。

3.2.2 页面 Phishing 异常检测

Phishing页面检测算法步骤(2)检测待测页面中Phishing敏感特征的异常信息，以此来衡量页面异常程度。该步骤的输入为 8 维的敏感特征向量 V ，输出为 0~1 之间的实数值 e_p 。算法利用BP神经网络的自适应和自学习的能力，通过对其训练，使其从合法站点页面和以往的Phishing页面中概括出敏感特征的正常状态模型，作为敏感特征异常检测的模型。

算法所使用的神经网络为 3 层BP神经网络。网络结构为：输入层包含节点个数与页面检测算法步骤(1)所产生的敏感特征向量的维数相同；输出层节点个数为 1，输出 0~1 之间的实数，表示被检测页面的异常程度；一个隐藏层，节点个数根据式(4)^[3]计算：

$$N_h = \sqrt{0.43N_i N_o + 0.12N_o^2 + 2.54N_i + 0.77N_o + 0.35 + 0.5} \quad (4)$$

其中， N_h 为隐藏层节点数； N_i 为输入层节点数； N_o 为输出层节点数。

在将此神经网络应用于检测算法之前，需要对其进行训练。为了克服传统 BP 算法的收敛速度慢和容易陷入局部最小点的缺陷，页面检测算法使用 $L-M$ 算法作为神经网络的学习算法来获取高收敛速度和高精确度。关于训练样本，从 PhishTank.com 中记录的 2007 年 7 月—2007 年 8 月间已确认的 Phishing 站点中随机选取 150 个，提取相关信息作为 Phishing 站点样本；另外根据 millersmiles.co.uk 中的记录提取 100 个易受 Phishing 攻击的合法站点的相关信息作为合法站点样本，一共 250 个训练和测试样本。

Phishing页面检测算法经过步骤(1)的Web页面敏感特征的提取算法后，生成一个 8 维的敏感特征向量 $V=\langle v_1,$

v_2, \dots, v_8 , 作为训练好的神经网络 S 的输入。

$$e_p = S(v_1, v_2, \dots, v_8) \quad (5)$$

神经网络的输出 e_p 是 0~1 之间的模糊数, 表示当前检测的Web页面的异常程度, 其值越接近 1, 异常程度越大, 可认为该Web页面很有可能是Phishing页面; 越接近 0, 异常程度越小, 可判为正常页面。采取模糊化的描述有利于更精确的描述Web页面发生异常的程度, 也能为后续的防御机制提供更多的信息。

3.2.3 页面分类器

Phishing页面检测算法步骤(3)使用前面步骤的结果来判定Web页面 p 是否为Phishing页面。该步算法的输入为BP神经网络的输出 e_p 和Web页面 p , 输出 c_p 为 0 或 1, 分别表示页面 p 为正常页面和页面 p 为Phishing页面。 c_p 是整个页面检测算法的最后输出。

步骤(1)所提取的敏感特征向量和步骤(2)所得到的页面异常程度可以为后续的防御措施提供很多信息。步骤(3)使用页面分类器是诸多后续方案之一。该算法主要目的是处理异常检测技术中的传统难题——误判率高的问题。

从现有已发生的攻击中概括出一系列已知攻击特征 (h_1, h_2, \dots, h_n) 和神经网络的输出 e_p 一起组成分类特征向量 H , 传入前向线性模型 L , 得出分类值。

$$c_p = L\left(\sum_{i=1}^{n+1} w_i * h_i\right) \quad (6)$$

$$L(x) = 0 \text{ if } x > 0, \quad L(x) = 1 \text{ if } x = 0 \quad (7)$$

式(6)中, n 是从页面 p 中提取的攻击特征的个数。参考文献[2], 本文选择对页面提取 7 个攻击特征, 分别为: 域名第 1 次注册时间, 常用商标图案, 可疑URL, 可疑链接, URL 中包含IP地址, URL中点的个数以及是否包含表单。 h_i ($i=1, 2, \dots, n$)表示当前页面中攻击特征的取值, $h_i=0$ 表示使用 h_i 来检测页面时, 判断该页面为正常, 而 $h_i=1$ 表示使用 h_i 来检测页面时, 判断该页面为Phishing页面。 w_i 是各分量的权值, 其中 h_{n+1} 和 w_{n+1} 分别表示 e_p 及其权值。 w_i ($i=1, 2, \dots, n$)由式(8)和式(9)确定:

$$w_i = \text{eff}_i / \sum_{i=1}^{n+1} \text{eff}_i \quad (8)$$

$$\text{eff}_i = RT_i - RF_i \quad (9)$$

其中, RT_i 和 RF_i 分别表示单独使用分量 h_i 进行页面检测时的正确率和误判率。攻击特征 h_i 是否能准确表述以往Phishing攻击的特征及其权值 w_i 直接影响最后的检测结果。由于Phishing攻击手段的快速发展, 因此已知攻击特征向量 H 应

该及时更新而权值在 H 变动和到达一定周期时也应更新。

(上接第 160 页)

5 结束语

密码算法实现形式多样化, 同一种算法的实现往往呈现出不同的语法形态, 单从静态数据特征的角度来识别密码算法有一定的局限性。在可执行代码中提取密码算法一方面要进一步深入研究各种密码算法的加密机制, 建立各种密码算法抽象语义库; 另一方面要尽可能地可从可执行代码中提取语义信息。

步骤(4)由式(6)~式(9)计算出 c_p , 其取值为 0 或 1, 即表示页面 p 是正常页面还是Phishing页面。将此判断提交给后续Phishing防御机制。

4 实验结果与分析

实验将实时收集的站点信息作为测试用例, 2007 年 8 月 10 日—2007 年 8 月 14 日间持续关注 PhishTank.com 中 Phishing URL 的最新报告, 随机选取并进行验证, 收集经确认的 Phishing 站点的相关信息。一共收集了 250 个 Phishing 站点作为测试用例。使用 Google 搜索在线服务相关的站点, 随机选取搜索结果并进行验证, 收集经确认的合法站点的相关信息。本文共收集了 170 个合法站点作为测试用例, 将总共 420 个测试用例随机混合在一起, 使用算法逐个进行检测。

实验结果如下: 算法正确判断出 166 个合法站点页面和 243 个 Phishing 站点页面, 但是将 4 个合法站点页面误判为 Phishing 页面, 将 7 个 Phishing 站点页面漏判为合法页面。由此可得误判率为 2.4%, 漏报率为 2.8%, 正确率为 97.4%。对于误判的 4 个合法站点, 经过分析, 发现它们都是内容聚合站点, 与算法所针对提供在线金融类站点在内容组织上有较大差别, 因此, 会出现误判。而对于漏报的 7 个 Phishing 站点, 经过分析发现, 它们的页面大部分是由图片拼接而成, 算法不容易提取敏感特征, 因此, 会出现漏报。

5 结束语

本文研究并实现了一种基于敏感特征异常检测的 Phishing 页面检测算法。实验表明, 该算法具有较高的效率: 正确率>97%, 误判率<3%, 同时具有实时检测的速度。

目前算法中敏感特征的提取对于绝大部分的 Phishing 页面来说是有效的。针对新近出现的使用图片来构造 Phishing 页面的攻击方式, 将来算法可能需要结合图像相似性检测技术来提取这种攻击方式的敏感特征, 进一步提高检测效率。

参考文献

- [1] Pan Ying, Ding Xuhua. Anomaly Based Web Phishing Page Detection[C]/Proc. of the 22nd Annual Computer Security Applications Conference. New Orleans, LA, USA: [s. n.], 2006: 381-392.
- [2] Zhang Yue, Jason I. A Content-based Approach to Detecting Phishing Web Sites[C]/Proc. of the 16th International Conference on World Wide Web. Budapest, Hungary: [s. n.], 2007: 639-648.
- [3] White H. Commentionist Nonparametric Regression: Multilayer Feed Forward Networks Can Learn Arbitrary Mapping[J]. Neural Networks, 1990, 3(1): 47-51.

参考文献

- [1] Alias C. Program Optimization by Template Recognition and Replacement[D]. Versailles, France: University of Versailles Saint-Quentin. 2005.
- [2] Schneier B. 应用密码学[M]. 北京: 机械工业出版社, 2001.
- [3] Menezes A J, Oorschot P C. 应用密码学手册[M]. 北京: 电子工业出版社, 2005.
- [4] 杨广正, 吴 岷, 张晓莉, 等. 模式识别[M]. 合肥: 中国科学技术大学出版社, 2003.