

Normandy: A Framework for Implementing High Speed Lexical Classification of Malicious URLs

Shaun P. Egan, Dr. Barry Irwin

Security and Networks Research Group, Department of Computer Science

Rhodes University, P. O. Box 94, Grahamstown 6140

Tel: +27 46 6038111, Fax: +27 46 6037350

email: g10e4008@campus.ru.ac.za ; b.irwin@ru.ac.za

Abstract—Research has shown that it is possible to classify malicious URLs using state of the art techniques to train Artificial Neural Networks (ANN) using only lexical features of a URL. This has the advantage of being high speed and does not add any overhead to classifications as it does not require look-ups from external services. This paper discusses our method for implementing and testing a framework which automates the generation of these neural networks as well as testing involved in trying to optimize the performance of these ANNs.

Index Terms—Artificial Intelligence, Neural Networks, Network Security

I. INTRODUCTION

Phishing is a malicious activity where attackers try to manipulate users into entering identifying information by providing falsified resources which mimic the resource that users are attempting to access. In the example of banking, a fraudulent website is created that closely, if not perfectly, resembles the legitimate website for that bank. An email containing a link to the fraudulent page is sent out to users and attempts to have them log onto the resource using their details. This is a simple example of the attack type, with many more complex methods in use today. The Anti-Phishing Work Group has shown that even though the number of unique phishing sites has dropped in the period of the first half and second half of 2011, there is still a large volume of phishing attacks that occur every month, with a total of 83 083 attacks recorded for the second half of 2011 [1], with the month of March having 26 402 phishing attacks reported to the Anti-Phishing Work Group [2] alone.

It has been shown by several researchers that it is possible to identify malicious phishing URLs by using lexical elements of these URLs alone [3, 4]. The justification for this is that experienced users can often identify malicious URLs by their odd textual properties, the way that they appear when compared to benign URLs. These URLs are not easy to identify programmatically as attackers try to mask the intent of these URLs using several methods of obfuscation.

Authors have used the Perceptron model from the Artificial Neural Network paradigm, also known as classifiers, to identify these URLs [3, 4]. The authors compared the accuracy of these classifiers and have found them to be as accurate as other approaches, including blacklisting, content analysis and spam filtering, but with

several advantages over these methods.

There are two methods of classification that may be used for perceptrons, the first being a lexical only classification which relies solely on the use of the features of the URL itself. The second method is known as a fully featured classification and uses features that are taken from external services (such as blacklists and WHOIS data) via lookups in addition to lexical features. Lexical only classification has an advantage in that no overhead is incurred by adding the latency required to perform these lookups. These lookups include blacklists, spam filters and reputation services, depending on the implementation of the classifier.

Another factor impacting the performance of a classifier is the use of training method, ranging from the Online Perceptron model to the Adaptive Regularization of Weights (AROW) training method. Classifiers trained using the Online Perceptron method show to be 94% accurate using the lexical only approach, where fully featured classifications are 98% accurate. This small drop in accuracy has been shown to be almost completely mitigated by using the AROW technique to train the classifier [3].

II. NEURAL NETWORKS

Neural Networks are a family of mathematical models which try to mimic, in a simplified manner, the way in which the human brain works. They consist of a series of nodes (neurons) which serve to combine inputs taken from connections called synapses. While this is a very simple analogy, they have shown to be versatile in many situations, with the ability to be trained and adapt. All classifiers used for the purpose of identifying malicious URLs have been based on the perceptron model which is a single neuron used as a linear combiner for a series of weighted inputs.

A. Inputs and training data

All neural networks require training data that serves as examples of real world data from which to learn. There are several training algorithms that may be employed, each with specific benefits. Training data for our perceptron comes primarily from two sources.

Open Directory provides a database of known benign sources and is available for download. This source of training data includes 9 198 532 benign URLs to choose from. Not all of these URLs will be used, but allows for a good random selection of benign URLs for training the classifier with. The second source of information is Phishtank, a manually vetted database of phishing URLs

which are submitted by users. Phishtank also has a database which is downloadable by the public. Using the Phishtank API, our framework has downloaded a total of 13 867 malicious URLs. It is important to note that these URLs are phishing URLs. The classifier may be adapted to detect malicious URLs as well due to the apparent difference in structure as shown in [3].

To use these URLs with the neural network, a descriptor has to be generated. Each URL is analysed and turned into an array where each element serves as an input to the classifier. The bulk of the elements are binary inputs indicating the presence of specific words, called a bag-of-words. The rest of the descriptor represents values which represent obfuscation resistant features such as URL length, number of delimiters and number of arguments passed. There are 20 of these obfuscation resistant features in total and a variable amount of bag-of-words features since it is generated by the URLs in the training data.

Each URL is fed through the perceptron and a classification is made. The neural network then attempts to correct itself based on the input for that iteration.

III. FRAMEWORK

The primary output of this research is to create a framework which will automate the process of gathering testing data, training new classifiers and making them accessible to the public through the use of browser extensions, proxy plugins and spam filters which may use a web service to check for regular updates. Even though neural networks are adaptable, this allows for low response times to changes in malicious URL obfuscation trends as well as unforeseen changes that may occur.

The framework, currently referred to as Normandy, is being built in python to allow for rapid prototyping and cross platform compatibility, but is however being built to target the Linux platform. The class hierarchy is shown in figure one and shows the basic components which make up Normandy. The Observer design pattern is being used to construct the framework as it allows for fairly complex scheduling orders and work flows without requiring a high level of complexity.

The work flow follows logically from the process required to train these classifiers. Firstly, new training data is downloaded every 12 hours from data sources defined by the installation. This data is preformatted and inserted into a database with a timestamp and an associated hash of the URL. The training data is then used to create descriptors for each of the URLs and is used to train a neural network. Once the network has reached its optimal classification accuracy it is published to the web service via another table, along with the bag-of-words and normalization vectors used to generate a descriptor for any URL that the client may encounter. Separate to this process is a statistics generator which runs every time new training data is received. This is primarily for research purposes and will allow researchers to identify shifting trends in URL obfuscation techniques.

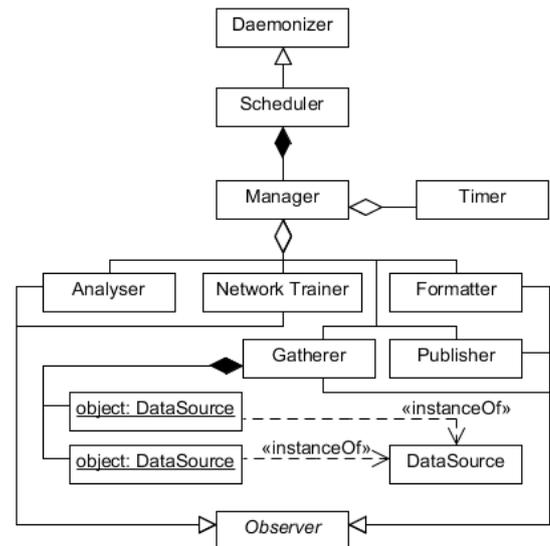


Figure 1: Class Hierarchy

IV. CONCLUSION

Perceptrons have been shown to be effective methods of identifying different types of malicious URLs. This research will deliver a method by which these perceptrons may be deployed to end users with very little, if any human involvement. This allows for more stable protection and is computationally inexpensive to maintain.

V. REFERENCES

- [1] G. Aaron and R. Rasmussen. (2012, April) Global phishing survey: Trends and domain name use in 2h2012. Anti Phishing Work Group. <http://www.antiphishing.org>. [Online]. Available: <http://www.antiphishing.org/reports/APWGGlobalPhishingSurvey2H2011.pdf>; Last accessed: 06/05/2012
- [2] (2012, April) Phishing activity trends report 1st half 2011. Anti Phishing Work Group. [Online]. Available: http://www.antiphishing.org/reports/apwg_trends_report_h1_2011.pdf; Last accessed: 06/05/2012
- [3] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious web sites from suspicious urls," in Proceedings of the SIGKDD Conference. Paris, France, 2009.
- [4] S. Egan and B. Irwin, "An evaluation of lightweight classification methods for identifying malicious URLs," in Internet Security South Africa, 2011.
- [5] Phishtank. [Online]. Available: <http://www.phishtank.com/>; Last accessed: 27/04/2011

Shaun Egan received his undergraduate degree in 2010 from the University of South Africa and is presently studying towards his Master of Science degree at Rhodes University where he received his honours degree. His research interests include Artificial Intelligence, Security, Networks and Frameworks.