

PHISHING WEB SITES FEATURES

CLASSIFICATION ON MACHINE LEARNING

Tangella Ravi Kiran (MCA Scholar), B V Raju College, Vishnupur, Bhimavaram, West Godavari District, Andhra Pradesh, India, 534202.

S. K. Alisha, B V Raju College, Vishnupur, Bhimavaram, West Godavari District, Andhra Pradesh, India, 534202.

ABSTRACT:

Phishing are one of the most common and most dangerous attacks among cybercrimes. The aim of these attacks is to steal the information used by individuals and organizations to conduct transactions. Phishing websites contain various hints among their contents and web browser-based information. The purpose of this study is to perform Extreme Learning Machine (ELM) based classification for 30 features including Phishing Websites Data in UC Irvine Machine Learning Repository database. For results assessment, ELM was compared with other machine learning methods such as Support Vector Machine (SVM), Naïve Bayes (NB) and detected to have the highest accuracy of 95.34%

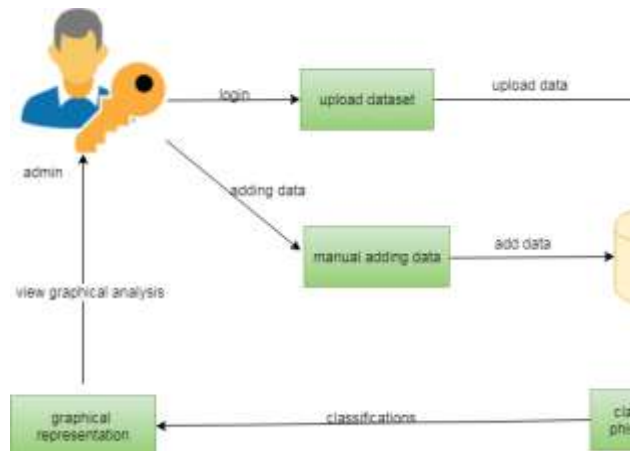
1. INTRODUCTION

Internet use has become an essential part of our daily activities as a result of rapidly growing technology. Due to this rapid growth of technology and intensive use of digital systems, data security of these systems has gained great importance. The primary objective of maintaining security in information technologies is to ensure that necessary precautions are taken against threats and dangers likely to be faced by users during the use of these technologies. Phishing is defined as imitating reliable websites in order to obtain the proprietary information entered into websites every day for various purposes, such as usernames, passwords and citizenship numbers. Phishing websites contain various hints among their contents and web browser-based information. Individual(s) committing the fraud sends the fake website or e-mail information to the target address as if it

comes from an organization, bank or any other reliable source that performs reliable transactions. Many articles have been published about how to predict the phishing websites by using artificial intelligence techniques. We examined phishing websites and extracted features of these web sites. Guidelines regarding the extracted features of this database are given below. In the first section we defined rules and we gave equations of web features. We need these equations in order to explain phishing attacks characterization. In this study, features in the database created for phishing websites are classified by determining the input and output parameters for the ELM classifier. Results obtained by ELM show that ELM has higher achievement compared to other classifier (SVM and NB) methods. This study is considered to be an applicable design in automated systems with high

performing classification against the phishing activity of websites.

2. ARCHITECTURE



3. EXISTING SYSTEM

Internet use has become an essential part of our daily activities as a result of rapidly growing technology. Due to this rapid growth of technology and intensive use of digital systems, data security of these systems has gained great importance. The primary objective of maintaining security in information technologies is to ensure that necessary precautions are taken against threats and dangers likely to be faced by users during the use of these technologies. Phishing is defined as imitating reliable websites in order to obtain the proprietary information entered into websites every day for various purposes, such as usernames, passwords and citizenship numbers. Phishing websites contain various hints among their contents and web browser-based information. Individual(s) committing the fraud sends the fake website or e-mail information to the target address as if it comes from an organization, bank or any other reliable source that performs reliable transactions. Many articles have been

published about how to predict the phishing websites by using artificial intelligence techniques. We examined phishing websites and extracted features of these web sites. Guidelines regarding the extracted features of this database are given below. In the first section we defined rules and we gave equations of web features. We need these equations in order to explain phishing attacks characterization.

4. PROPOSED SYSTEM

In this study, features in the database created for phishing websites are classified by determining the input and output parameters for the ELM classifier. Results obtained by ELM show that ELM has higher achievement compared to other classifier (SVM and NB) methods. This study is considered to be an applicable design in automated systems with high performing classification against the phishing activity of websites. Furthermore, in literature comparisons, this study is observed to be high-performing by having a high performance of 92.18% that is also the highest test performance in the publication. The topics addressed in this section are the two measures that affect the performance of the model and the algorithm used, the first one being the division of data set into training and test data set and the second one being the definition of expressions measuring the performance. In the first measure, the data set is divided into three parts as training, validation and test data by three-phase division in K-Fold method, and model selection and performance status are simultaneously performed. In the second measure, performance assessment of



classifier models generally uses a validation value. phishing attacks are vishing, smishing, search engine phishing, spear phishing, whaling attacks.

5. SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- ◆ **ECONOMICAL FEASIBILITY**
- ◆ **TECHNICAL FEASIBILITY**
- ◆ **SOCIAL FEASIBILITY**

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will

lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

SYSTEM TEST

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing



Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at



least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format

- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

6. CONCLUSION

In this paper, we defined features of phishing attack and we proposed a classification model in order to classification of the phishing attacks. This method consists of feature extraction from websites and classification section. In the feature extraction, we have clearly defined rules of phishing feature extraction and these rules have been used for obtaining features. In



order to classification of these feature, SVM, NB and ELM were used. In the ELM, 6 different activation functions were used and ELM achieved highest accuracy score.

7. REFERENCES

[1] AO Kaspersky lab. (2017). The Dangers of Phishing: Help employees avoid the lure of cybercrime. [Online] Available: <https://go.kaspersky.com/Dangers-Phishing-Landing-Page-Soc.html> [Oct 30, 2017].

[2] "Financial threats in 2016: Every Second Phishing Attack Aims to Steal Your Money" Internet:

<https://www.kaspersky.com/about/pressreleases/2017-financial-threats-in-2016>. Feb 22, 2017 [Oct 30, 2017].

[3] Y. Zhang, J. I. Hong, and L. F. Cranor, "Cantina: A Content-based Approach to Detecting Phishing Web Sites," New York, NY, USA, 2007, pp. 639-648.

[4] M. Blasi, "Techniques for detecting zero day phishing websites." M.A. thesis, Iowa State University, USA, 2009.

[5] R. S. Rao and S. T. Ali, "PhishShield: A Desktop Application to Detect Phishing Webpages through Heuristic Approach," *Procedia Computer Science*, vol. 54, no. Supplement C, pp. 147-156, 2015.

[6] E. Jakobsson, and E. Myers, *Phishing and Counter-Measures: Understanding the Increasing Problem of Electronic Identity Theft*. Wiley, 2006, pp.2-3.

[7] L. A. T. Nguyen, B. L. To, H. K. Nguyen, and M. H. Nguyen, "Detecting phishing web sites: A heuristic URL-based approach," in 2013 International Conference on Advanced Technologies for Communications (ATC 2013), 2013, pp. 597-602.

[8] Z. Zhang, Q. He, and B. Wang, "A Novel Multi-Layer Heuristic Model for Anti-Phishing," New York, NY, USA, 2017, p. 21:1-21:6.

[9] N. Sanglerdsinlapachai and A. Rungsawang, "Web Phishing Detection Using Classifier Ensemble," New York, NY, USA, 2010, pp. 210-215.

[10] G. Xiang, J. Hong, C. P. Rose, and L. Cranor, "CANTINA+: A Feature- Rich Machine Learning Framework for Detecting Phishing Web Sites," *ACM Trans. Inf. Syst. Secur.*, vol. 14, no. 2, pp. 21:1-21:28, Sep. 2011.

[11] R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting phishing websites based on self-structuring neural network," *Neural Comput & Applic*, vol. 25, no. 2, pp. 443-458, Aug. 2014.

[12] Pradeepthi K V and Kannan A, "Performance study of classification techniques for phishing URL detection," in 2014 Sixth International Conference on Advanced Computing (ICoAC), 2014, pp. 135-139.