An optimized Fake Website Detection Using Regression and Machine Learning

[1]M Naveen kumar, [2]G kumari

[1] Research Scholar,

Department of CSE, Centurion University of Technology and Management, Visakhapatnam

[2] Research Scholar,

Department of CSSE, A.U.College of Engineering (A), Andhra University, Visakhapatnam

ABSTRACT

This work centers on the examination of phishing websites utilizing data mining and regression. There are number of users who purchase products online and make payment through various websites .There are multiple websites who ask user to provide sensitive data such as username, password or credit card details etc. often for malicious reasons. This type of websites are known as phishing websites. In order to detect and predict phishing website, we implemented an intelligent, flexible and effective system that is based on using Data mining algorithm. We implemented Logistic Regression algorithm and techniques to classify their legitimacy. The phishing website can be detected based on some important characteristics like URL and Domain Identity, and security in the final phishing detection rate. Once user wanted to check whether the website is legitimate or not, our system uses data mining algorithm to check its legitimacy. This application can be used by many internet users in order to save themselves from an ocean of phishing sites. Data mining algorithm used in this system provides better performance. . There are two components bunches in the calculation, URL lexical and page content highlights. This work is continuing to, add some values to the field malware combat, mitigate some threats, and improve Performance by enhancing the detection rate. With the help of this system user can also purchase products online without any hesitation. Admin can add phishing website URL or fake website URL into system where system could access and scan the phishing websites. By using algorithm, it will add new suspicious URLS to file when a user submits it.

Keywords: Phishing, regression, Spoofing, CSS Matching, Blacklisting, Whitelisting, Data Mining

1 INTRODUCTION

1.1 Phishing

Phishing is the attempt to obtain sensitive information such as usernames, passwords, and credit card details (and, indirectly, money), often for malicious reasons, by disguising as a trustworthy entity in an electronic communication. The word is a neologism created as a homophone of fishing due to the similarity of using a bait in an attempt to catch a victim. Phishing is typically carried out by email spoofing or instant messaging, and it often directs

users to enter personal information at a fake website, the look and feel of which are almost identical to the legitimate one. Communications purporting to be from social web sites, auction sites, banks, online payment processors or IT administrators are often used to lure victims. Phishing emails may contain links to websites that are infected with malware. Phishing is an example of social engineering techniques used to deceive users, and exploits weaknesses in current web security. Attempts to deal with the growing number of reported phishing incidents include legislation, user training, public awareness, and technical security measures. Many websites have now created secondary tools for applications, like maps for games, but they should be clearly marked as to who wrote them, and users should not use the same passwords anywhere on the internet. The web has gotten a Fundamental in our day by day life; it's the base of banking exchanges, shopping, diversion, asset sharing, news, and interpersonal interaction. The development of the web remunerated the digital lawbreakers towards it, with this development, additionally the structure and the utilization of the malware situation has changed, its more take their and polymorphic than harming the machines. Most of malware is planned to either take the client's private information, or power the injured individual framework to join a malware circulation organize. Web is a typical strategy for spreading malware, the aggressors abuse the vulnerabilities of internet browsers, web application, and working framework to oversee an injured individual's machine, which is utilized to have different vindictive exercises, for example, store shower, speck net, key lumberjacks, sending spam messages, etc. Further methods, for example, PHP language, adobe streak, and visual fundamental content are in like manner have capacity of download.

The web has gotten a basic in our day by day life, it's the base of banking exchanges, shopping, diversion, asset sharing, news, and interpersonal interaction. The development of the web remunerated the digital crooks towards it, with this development, additionally the plan and the utilization of the malware situation has changed, its more take their and polymorphic than harming the machines. Most of malware is planned to either take the client's private information, or power the injured individual framework to join a malware appropriation arrange. The purpose of this work is to stop the fraudulent practice of sending emails and to safe guard the personal information of the individuals relating to their cards number, credit, passwords and other important details. The aim of this work is to detect phishing websites using machine learning. This project is used to secure the websites from malicious users. The attack occurs when a user visits a suspected website. Therefore attacker focuses on a website that has become the center of attention, and then exploits the vulnerabilities in both client and server to launch the attack. Web attack is a challenge; it necessitates a careful understanding of the details and the behaviour.

1.1.1 Link manipulation

Most methods of phishing use some form of technical deception designed to make a link in an email (and the spoofed website it leads to) appear to belong to the spoofed organization. Misspelled URLs or the use of subdomains are the common tricks used by phishers. In the following example URL, http://www.yourbank.example.com/, it appears as though the URL will take you to the example section of the yourbank website; actually this URL points to the "yourbank" (i.e. phishing) section of the example website. Another common trick is to make

the displayed text for a link (the text between the <A> tags) suggest a reliable destination, when the link actually goes to the phishers' site. Many email clients or web browsers will show previews of where a link will take the user in the bottom left of the screen, while hovering the mouse cursor over a link. This behavior, however, may in some circumstances be overridden by the phisher.

A further problem with URLs has been found in the handling of internationalized domain names (IDN) in web browsers that might allow visually identical web addresses to lead to different, possibly malicious, websites. Despite the publicity surrounding the flaw, known as IDN spoofing or homograph attack, phishers have taken advantage of a similar risk, using open URL redirectors on the websites of trusted organizations to disguise malicious URLs with a trusted domain. Even digital certificates do not solve this problem because it is quite possible for a phisher to purchase a valid certificate and subsequently change content to spoof a genuine website, or, to host the phish site without SSL at all. Once the user successfully login the authorized page will be displayed otherwise that shows the error messages. Login is compulsory. 3) URL Comparison: Many users unwittingly click phishing URL's every day and every hour. URL is created to identify the address of the web pages. When the URL is typed or clicked, the specific page is displayed on the screen. The most common method to detect malicious URLs deployed by many antivirus groups is the blacklist method. Blacklists are essentially a database of URLs that have been confirmed to be malicious in the past. The goal of machine learning for malicious URL detection is to maximize the predictive accuracy 4) Location Verification: Here we confirm the location. The information relating to the location and the latitude, longitude is also confirmed in location verification. The distance is measured automatically. Location verification enables location-based access control. Location verification provides the accuracy of the location. 5) Code Verification: It is the process for verifying the software code. It is the short numeric code. The objective of code verification process is to check the software code in all aspects. Through the help of code generation we can verify if the specific websites exist or if it fake or fraud website. Code verification is the shortest process in order to verify the code is valid or invalid. 6) IP Address Listing: It is an internet protocol address that is shown numerically. It is connected to each and every device. Every system has its own IP address.

II RELATED WORK

2.1 Django Framework:

Django follows the MVC pattern closely, however it does use its own logic in the implementation. Because the "C" is handled by the framework itself and most of the excitement in Django happens in models, templates and views, Django is often referred to as an MTV framework. In the MTV development pattern: M stands for "Model," the data access layer. This layer contains anything and everything about the data: how to access it, how to validate it, which behaviors it has, and the relationships between the data. T stands for "Template," the presentation layer. This layer contains presentation-related decisions: how something should be displayed on a Web page or other type of document.V stands for "View," the business logic layer. This layer contains the logic that accesses the model and

defers to the appropriate template(s). You can think of it as the bridge between models and templates.

A Django template is a string of text that is intended to separate the presentation of a document from its data. A template defines placeholders and various bits of basic logic (template tags) that regulate how the document should be displayed. Usually, templates are used for producing HTML, but Django templates are equally capable of generating any text-based format.

III LITERATURE REVIEW

World Wide Web Consortium (W3C) is the international standards organization for the World Wide Web (www). It develops standards, specifications and recommendations to enhance the interoperability and maximize consensus about the content of the web and define major parts of what makes the World Wide Web work. Phishing is a type of Internet scams that seeks to get a user's credentials by fraud websites, such as passwords, credit card numbers, bank account details and other sensitive information. There are some characteristics in webpage source code that distinguish phishing websites from legitimate websites and violate the w3c standards, so we can detect the phishing attacks by check the webpage and search for these characteristics in the source code file if it exists or not.

(Mona Ghotaish Alkhozae, Omar Abdullah Batarfi, et al., 2011) proposed a phishing detection approach based on checking the webpage source code, they extracted some phishing characteristics out of the W3C standards to evaluate the security of the websites, and check each character in the webpage source code, if they find a phishing character, they would decrease from the initial secure weight.[1]

There are many phishing detection techniques available, but a central problem is that web browsers rely on a black list of known phishing website, but some phishing website has a lifespan as short as a few hours. These website with a shorter lifespan are known as zero day phishing website. Thus, a faster recognition system needs to be developed for the web browser to identify zero day phishing website. (Chandan, Chheda, Gosar, R. Shah, Bhave, et al., 2013)[2]

In existing Online Phishing Detection systems, usually the reference to the database is taken for making any conclusion about the degree of phishiness of the website. Concentrating on getting the necessary attributes in real time environment using Hadoop MapReduce, increases both speed & efficiency of the system. (Kaustubh A. Hiwarekar, Dr. R. C. Thool, et al., 2013)[3]

A heuristic is an algorithm to distinguish phishing sites from others based on users experience, that is heuristics checks if a site seems to be phishing site. A heuristic based solution employs several heuristics and converts each heuristics into a vector. (Miyamoto, Hazeyama, Kadobayashi, et al., 2008)[4]

IV FISHING WEBSITE DETECTION AND METHODOLOGY

4.1 Gathering Data

The first task was gathering data. We did some surfing and found some websites offering fake links. The next task was finding clear URLs. There was a data set available. [5]

We gathered around 400,000 URLs out of which around 80,000 were fake and others were clean. We have further added 50 URLS manually to the dataset. This forms the final dataset used in our project.

4.2 Analysis

We used Logistic Regression in our Project. The first part was tokenizing the URLs. We have written our own tokenizer function for this since URLs are not like some other document text. Some of the tokens we get are like 'virus','exe','php','wp','dat' etc. The next step is to load data into a list and to store it. We have vectorized URLs. We used tf-idf scores instead of using bag of words classification since there are words in urls that are more important than other words e.g 'virus', '.exe' ,'.dat' etc.
After vectorizing we converted it into test data and training data and performed logistic regression.

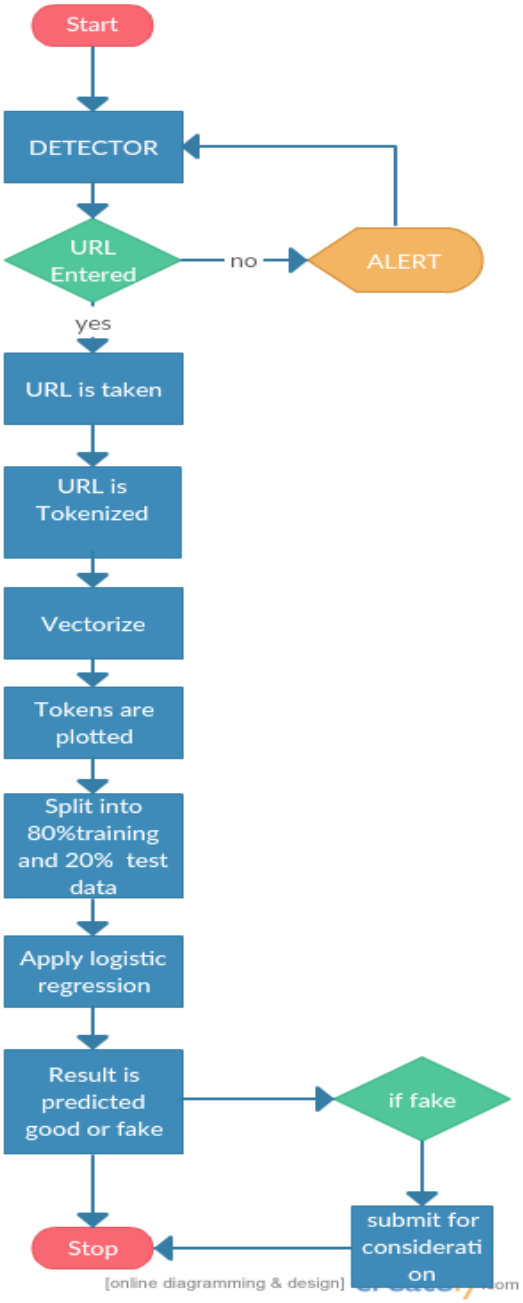

Fig 1 Flow chart for the proposed system

4.3 Implementation

The detector page is selected from the tab to the left of the page.

Then the URL is entered into the form page. If no URL is entered then an alert pops up asking to enter a URL.

After URL is entered it reaches the function. It passes through URLS.py and then to views.py.

The URLS in the data set are then tokenized first using the tokenizer function written and predefined.

The Tokenized URLs are then again vectorized and are plotted by splitting data into 80% training set and 20% testing data.

Now we apply logistic regression to detect whether our URL which is taken from the Webpage is fake or good.

If the webpage is good the page gets redirected and message dispalys it as good.

If the webpage is fake the page redirects to show that its bad and that website would be considered to be included in the further considerations by the admin.

4.3.1 Admin

Admin plays a crucial role in this type of system. The list of functions of an admin are given below.

The admin should update the data set periodically. The submitted URLS can be seen in text.txt file which will in the path of the file. The admin will have lowest maintenance cost as we maintained a file system rather than a database system.

4.3.2 Logistic regression

Logistic regression, or Logit regression, or logit model is a regression model where the dependent variable (DV) is categorical. This Logistic regression covers the case of a binary dependent variable—that is, where it can take only two values, "0" and "1", which represent outcomes such as pass/fail, win/lose, alive/dead or healthy/sick. Cases where the dependent variable has more than two outcome categories may be analyzed in multinomial logistic regression, or, if the multiple categories are ordered, in ordinal logistic regression. In the terminology of economics, logistic regression is an example of a qualitative response/discrete choice model.

We used the above algorithm in our project so that we classify our URLs as good or fake. We used this binary classifier since it is fast and gave an accuracy of greater than 95%.

4.3.3 Detector Methods:

The detector module is the heart of our project. The detector module will consist of one form field through which the URL is taken. The URL is then sent to test its legitimacy and the result is displayed in the output page. If no URL is entered then the page will display an alert to show to enter a URL. The form field is accessible in the views.py file which consists of the whole logic of the code.

4.3.4 Submit Method:

The submit module will also contain only a single field. The field present will take suggested URL form the user and will write it to the notepad directly which can be accessed by the admin. The admin further checks for any requirements and can update it to the master list of URL's directly. If no URL is entered an alert pops up saying to enter a URL. If URL is written to the file after submission then it displays a success message that the URL has been successfully added to the list.

VI RESULTS:

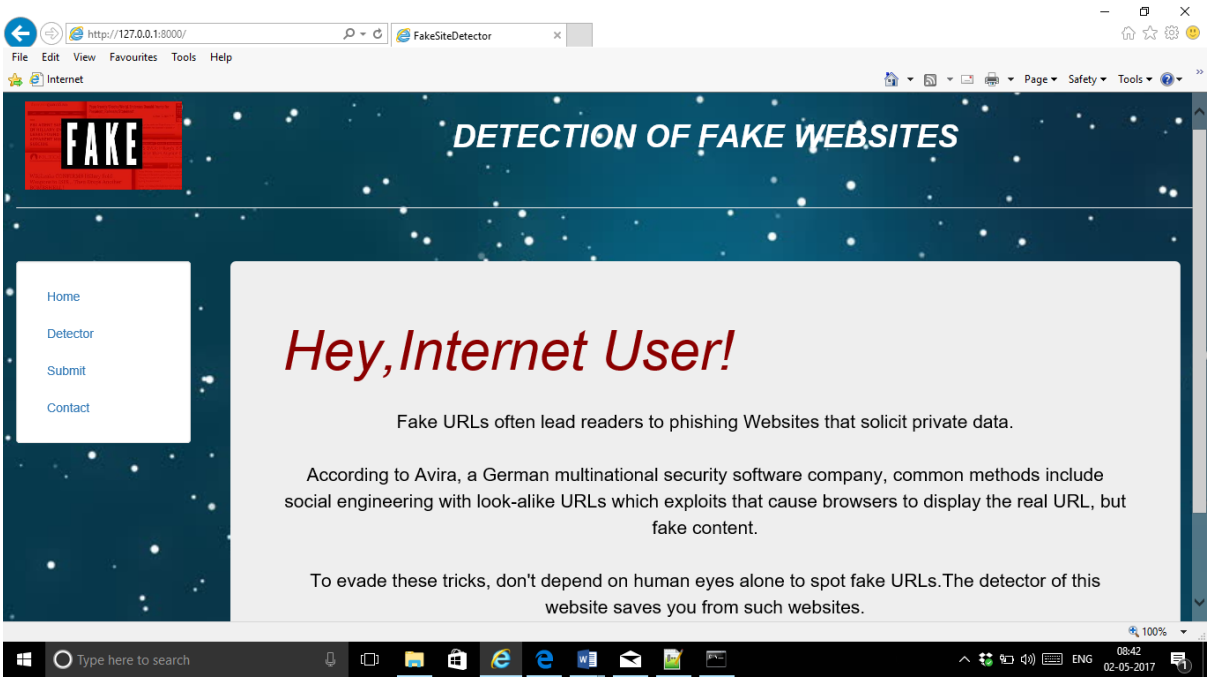

Fig Home method

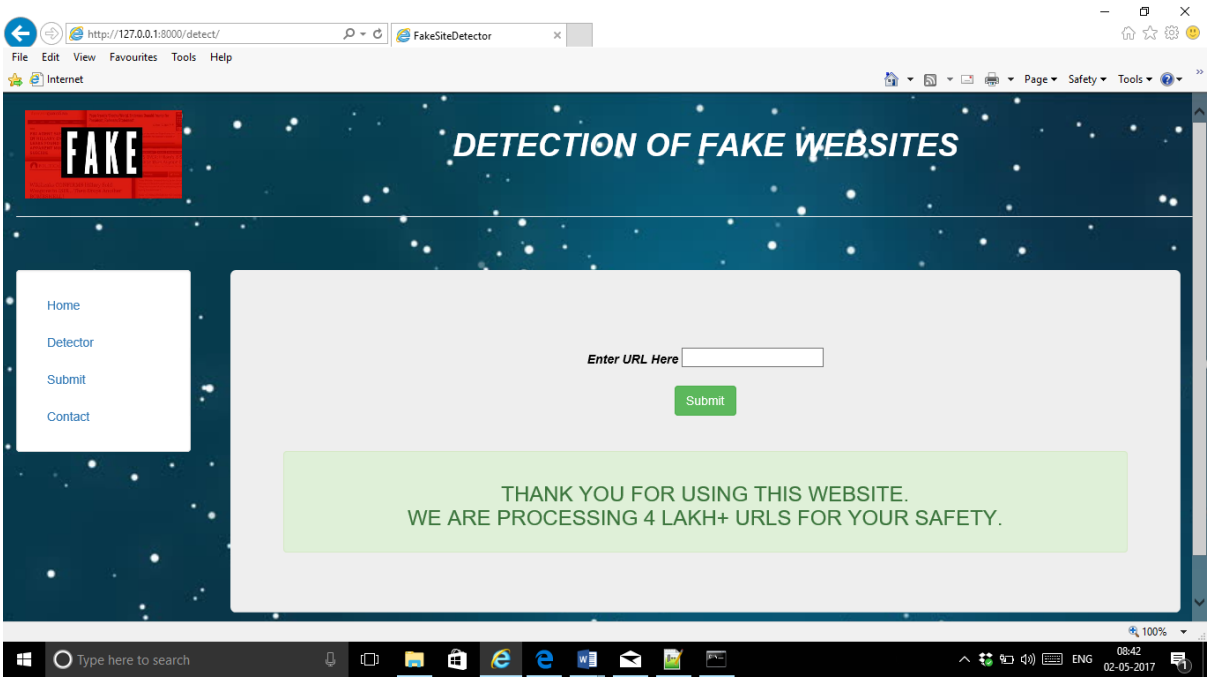

Fig Detector Method

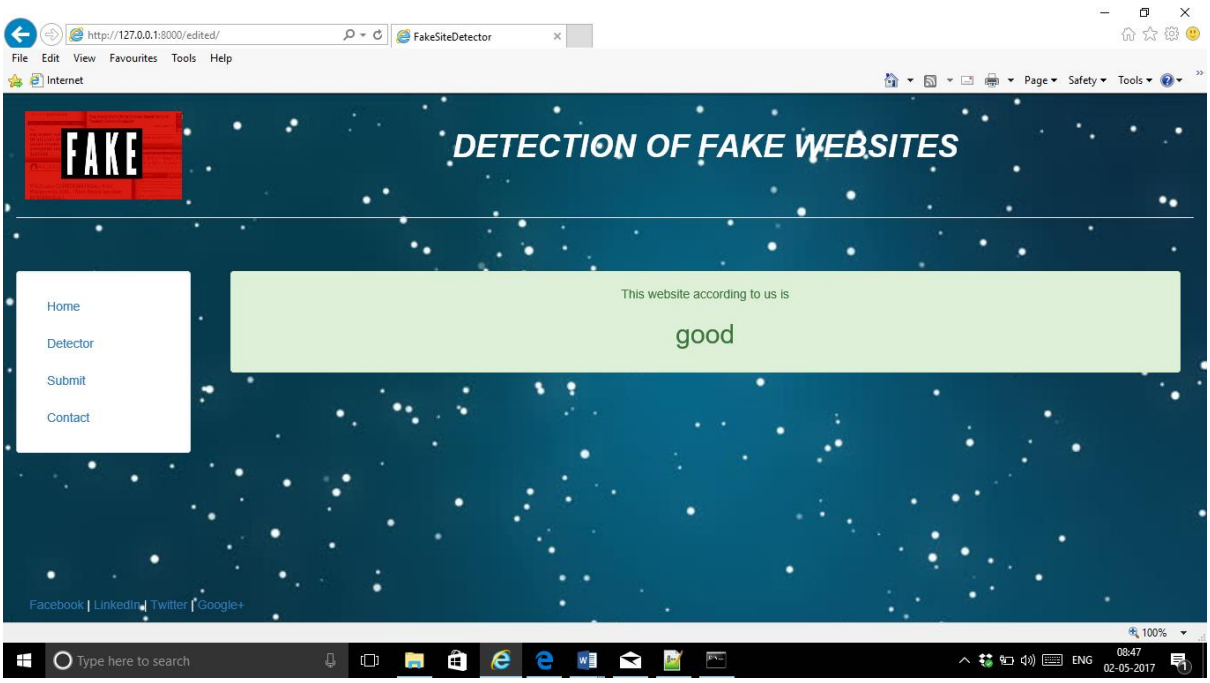

Fig Output for good URL

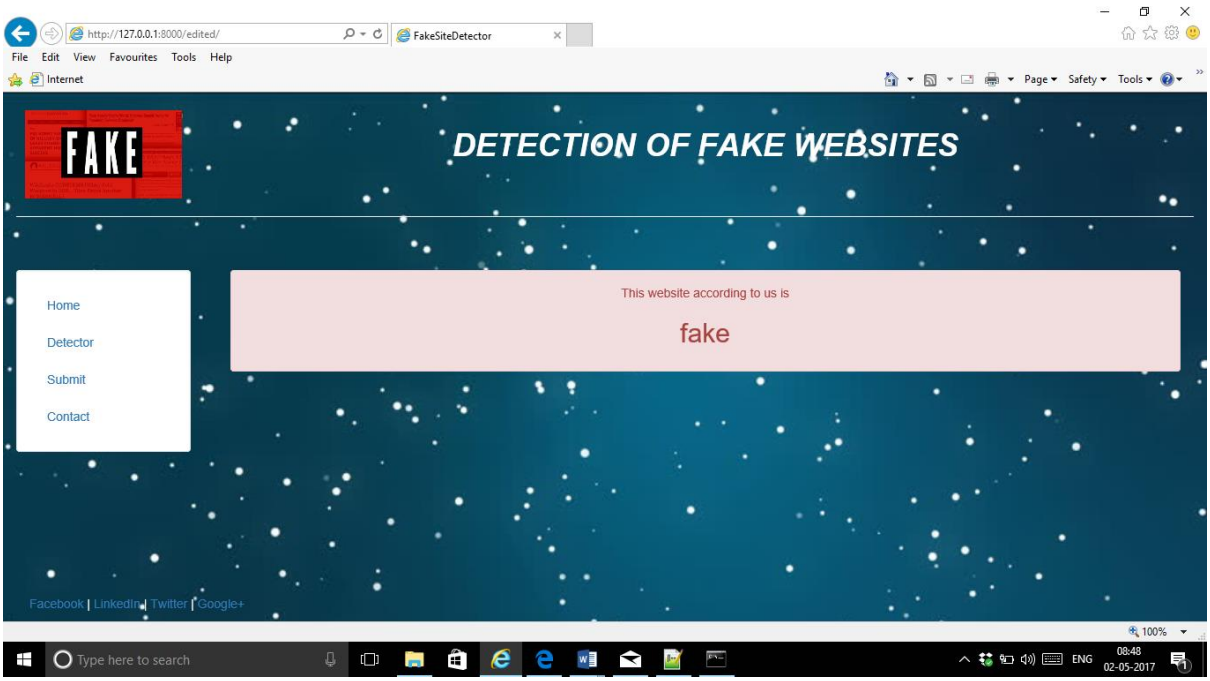

Fig Output for fake URL

VIII CONCLUSION AND FUTURE WORK

As the usage of internet increases, the number of users accessing the websites available on the net has also increased. Multiple websites ask the users to provide sensitive information like email ids, phone numbers, etc. for registration purposes. Some banking and utility websites require more personal information like bank account details and credit card numbers. Phishing websites deceive users and trick them into believing that they are using the original websites.

REFERENCES

[1] Research paper on Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code by Mona Ghotaish Alkhozae, Omar Abdullah Batarfi.

[2] Research paper on A Machine Learning Approach for Detection of Phished Websites Using Neural Networks by Charmi J. Chandan, Hiral P. Chheda, Disha M. Gosar, Hetal R. Shah.

[3] Research paper on Phishing Detection System Using Machine Learning and Hadoop-MapReduce Kaustubh A. Hiwarekar, Dr. R. C. Thool.

[4] Research paper on An Evaluation of Machine Learning-based Methods for Detection of Phishing Sites by Daisuke Miyamoto, Hiroaki Hazeyama, and Youki Kadobayashi.

[5] https://archive.ics.uci.edu/ml/datasets/URL+Reputation (for dataset)

[6] http://sysnet.ucsd.edu/projects/url/ (for dataset)

[7] Python 2.7.13 documentation.

[8] Django 1.11 framework documentation.

[9] BootStrap documentation.

[10] Jinja Templating documentation.

[11] Animate.css documentation.

[12] Han J and Kamber M, Data Mining, Concepts and Techniques, 2nd ed, San Fransisco, Morgan Kuffmann Publishers, 2001

Authors Profile

***M Naveen Kumar**, pursuing full time Ph.D from Centurion University of Technology & Management.He has 10 Years of extensive teaching experience in various domains such as Data mining, machine learning and Big Data analytics including various cloud models.*

***G Kumari**, pursuing full time Ph.D from Andhra University. She has 10 years of teaching experience in various areas such as Machine Learning, Big Data Analytics, Data Mining and various cloud computing models.*