DC Scanner: Detecting Phishing Attack

[1]Binay Kumar, [2]Pankaj Kumar, [3]Ankit Mundra, [4]Shikha Kabra
[1,2]Department of Computer Science and Engineering, Central University of Rajasthan
[3]Department of Computer Science and Engineering, Central University of Rajasthan,
[4]Department of Computer Science and Engineering, Manipal Univ., Jaipur
[1]2014MTCSE008@CURAJ.AC.IN, [2]2014mtcse015@curaj.ac.in, [3]ankitmundra8891@gmail.com,
[4]kabra.shikha1990@gmail.com

***Abstract*— Data mining has been used as a technology in various applications of engineering, sciences and others to analysis data of systems and to solve problems. Its applications further extend towards detecting cyber-attacks. We are presenting our work with simple and less efforts similar to data mining which detects email based phishing attacks. This work digs html contents of emails and web pages referred. Also domains and domain related authority details of these links, script codes associated to web pages are analyzed to conclude for the probability of phishing attacks.**

***Keywords*— Content analysis, Script scanning, URL verification, domain name, who is command.**

I. Introduction

The advent of phishing and key logging attacks has meant that some small proportion of bank accounts will at any time be under the control of criminals; money gets moved both from and through them [1].

Phishing attack is also known as the type of social engineering attack. This attack is performed to deceive legitimate users for the sack of committing some required/desired activities through internet/network. In terms of authors, phishing is a fraudulent attempt, usually made through email, to steal your personal information [2].Here the attackers pretend themselves as trustable web sources and convince the victims to act accordingly. The phishing attacks can be incurred by means of text, visual, voice contents [1]. In a press release Hong Kong MA (31 August, 2015) alarms public to classify the phone calls in the name of financial institutions [3] so that the customer can understand which one is the legitimate call and which one is fraudulent. Phishing in terms of web activities is found mostly in two forms, malware and non-malware phishing.

Usually victims are sent web links through emails or the links can be found in malicious websites. Whenever user clicks on web links, this will install malware in the victims' computers then the attack is known as malware based phishing. On the other hand in non-malware phishing, after clicking on the web links will redirect the user to some other malicious web pages. Here victims are convinced to fill in their credentials to complete the required/desired tasks. As the victims submit the forms carrying their credentials the data are sent to attacker web servers instead to trusted web servers for which the victims got convinced. In recent report on 24 Aug, 2015 Hong Kong MA notifies her public about fraudulent websites [4] and on 10 May, 2015 Bank of America alarms her customers by introducing phishing email methods [5].

Further, another type of Phishing attack is known as Email phishing attacks that can be seen in two common classes from higher perspective. One which deceive users by showing physical appearance of web pages exactly same as any trustable ones but have minor changes in the letters of web contents [6]. The changes are so minor and hidden that users fail to focus and lose their chances to stand again. The changes are actually found in URL of the websites what users open and get phished. e.g. *"gmaii.com"* for gmail.com, *"icieci.com"* for icici.com etc. Another one is something for which not users but the technology is responsible. Users see everything right and play in their ways but the data which are fed, redirected to malicious destinations. The users either don't understand till they don't know what they have lost or at end when data are redirected and behavior of websites become susceptible. In this type of attack a website is compromised and action attribute of html forms are changed with URL of malicious destinations. They are also compromised by cross scripting attacks which injects malicious script codes in the web pages. Resultantly form data are collected and sent to the malicious targets. A similar attack was seen in March 2004 [7]. A user see an email from ebay.com, she follows everything accordingly. She does activities accordingly and submits important credentials but finally it is noticed that information has gone to some other website rather than ebay.com.

Further, in order to defend phishing attacks we can classify the mechanisms in five classes i.e. Network level protection, Authentication, Client Side Tools, User Education, Server Side Filters and Classifiers [8]. All these approaches used Data Mining techniques in order to provide desired solution [9]. Further it has been used in phishing detection by heuristics [10], by human factors [11], by visual similarity [12], by blacklisted websites [13].

Analyzing domains of URLs, contents of html pages, links in emails to conclude websites or URLs as phishing ones with the help of data sets have also been seen as remarkable approaches in [14], [15] [16], [17].

Similar to domain and content analysis we present our work, in easy and simple ways, to detect phishing web pages or URLs which usually come along emails. We perform contents scanning of emails; web pages referred, script codes, domains and associated details. Also we define some needs in designs of web pages what we have felt to make the work easier and reduce the complexity that have been seen in past related works. These needs we define as essentials ones to be in genuine web pages. We suggest whole cyber world to keep the definitions entertained in the websites. In section II we brief

978-1-5090-0148-4/15/$31.00© 2015 IEEE

some related works which mine contents and domains of web pages in their own ways. In section III we have discussed our proposed approach and algorithms. Later we have shown the implementation of proposed approach in section IV and section V summarizes the future work. Finally we conclude this work in section VI.

II. Related work

In this section we are elaborating the provisos research work which has been proposed to overcome or detect the phishing attacks. The basic techniques to avoid phishing are URL verification, domain check and html contents' scan of web pages referred by email links.

Earlier, J. I. H. Zhang proposed CANTINA [18] to analyze and verify HTML contents of web page refereed by links in emails, domains of URLs found in web pages, also URLs using heuristics approaches. Referring CANTINA, Gupta et. al.[14] has believed in some symbols like "-" that are rarely used in genuine websites. They check for such symbols in domains, URLs and also they check the domain details e.g. age. Also they have used a list of malicious websites against which the scanned URLs are matched. On this basis, web URLs are declared as malicious or genuine ones. But with growing number of websites and domains it will not be easy to update list of malicious or white domains [mahmoud]. Also it is personal to use symbols in websites. Therefore results declared may be unbalanced decision and some genuine URLs can be filtered as malicious.

Justin et. al. [16] detects malicious websites by extracting properties and features of URLs. The URLs are analyzed and classified thereafter these are matched with a large database which contains filtered malicious URLs. The classification process is performed on real time basis individually by different online classifier which works independently. The database is provided by a mail server which updates it too. Although this work guarantees 99% accuracy to filter URLs but according to Khonji. et. al.[2] large number of entries in data sets can cause performance and resource constraints.

Pradeepthi et. al.[15] has surveyed for classification based methods for detecting phishing URLs and finally proposed that tree based classifier can result with more accuracy. The tree based classifier they define by concluding machine learning and pattern recognition algorithms. This work analyses structure of the URL rather than domain verification and html content mining. However, they again use a data set which is updated at training phase while analyzing URLs.

Gautham et. al.[17] look in html pages, collect the associated direct and indirect links to create a domain set. Also they extract some keywords from the html contents and feed to a search engine which returns another domain set. Concluding a target domain set from the two domain sets they use a third party DNS lookup to check for the legitimacy of the URLs. Using search engine meant for collaborating with again a third party and accuracy of result will depend on results of search engine. Here results are fed by search engine, will be a subject of how the search engine has been designed and defined.

III. Proposed Work

This work is proposed regarding those problems which arise because of two well known facts about non-malware based email-phishing attacks. One, a genuine URL displayed in the address bar of a web browser is changed with URL of an attacker website when user submits his/her credentials. In other words the entered credentials are redirected to attacker website whereas user can see the correct URL of a trusted website before hitting submit button. Second, URL shown in the address bar appears like address of any genuine website whereas the URL has some letters different than the genuine URL. Here visual appearance of the web page is kept same as what users expect as to be of a trustable website. This deceives as users don't notice the URL properly because of their own reasons e.g. unawareness, time, illusion etc.

This work identifies the above mentioned two scenarios of attacks when users open their emails and then alert users referring phishing attacks. The identifications are done by means of scanning and analysis similar to data mining. We perform these activities on the HTML content of the URLs given in email links, website domains, domain authority details obtained using who is command, script codes associated with the URLs. For bringing correct result we depend on some essentials of web page design. We define these essentials and suggest the whole cyber society to carve these instructions as standard to design web pages. This is helpful to filter phishing web pages and URLs. The job is performed from scanning a received email for finding html links. We peep into html contents and script content to find malicious URLs and also those attempts which attackers can apply to deceive our approach in future. Attempt to deceive our approach is meant a malicious web page design which will follow our definitions but will hide them such that maliciously intentioned information will not be visible to users. i.e. we assume domain owner name as essentials expo-sable e.g. Google Inc. for any websites of google.com domain. Now the attacker web page will have its own domain owner e.g. Gougle Inc but it will hide the content which can deceive the basic approach we are applying. To prevent this malicious attempt we mine the web pages to detect whether the essentials of web page designs have been tortured. Now, we can define that our work is meant as an email scanner to filter phishing emails by scanning contents of web pages and verification of web domain, domain details. Hence, the name of this work is Domain and Content scanner *(DC Scanner)*.

This work focus on defending against non-malware based web phishing attacks. As we have described above that we depend on specific design of web pages which we wish to follow as web page design standards for the sake of preventing phishing web pages. Below we give design instructions what we wish to add in the web pages and also we describe other constituents of this work.

A. Web Page Design

The design standards what we add in our work have following definitions.

1) Definition 1

Every web page must have an html tag which alone displays the correct and exact full website address of the website under which it is being hosted.

2) Definition 2

Every web page must have an html tag which clearly and precisely shows the domain owner name.

3) Definition 3

The tags following defintion1 and definition2 must bear a name or id whose value is same as the domain name.

4) Definition 4

Any of the tags used by above definition1, definition 2 and definition3 are not modified by script codes.

The definition1 is meant for proving that web page opened lies to the website whose address is shown in the web page. Definition 2 is meant for proving the webpage pretending as a genuine web page that it lies to the same domain whose authority information users can notice. Although it is possible that any tag with proper styles can be used to display authority information as per the definition 2 but we have used <h1>,<h2>or <h3>. The definition 3 makes this mechanism easier in order to locate the defined information. The definition4 prevents future malpractices against this approach.

B. Domain Verification

A phishing webpage has visual appearance bearing domain details in the web page same as of the trustable website. This helps us in verifying the web URL whether it is a genuine one which users trust or something else. The definitions instruct to bear a line of text to display the following things: (a) Website's full address. The phishing webpage will also put this information in the webpage to prove its fake genuineness. (b) Domain Associated Authority name in the webpage in the head section. These definitions help us in order to do correct verification of the web URL whether it is meant for the same web page users get visually or it is different. The verification is done by getting domain of the webpage from URL and getting another domain by looking into line of the website address. We compare these two, whether the domain obtained from the DNS is same what is shown to users or not. Again it is possible that the URL of the webpage is very similar to a trustable website with only one or two letters changes which may not be noticeable slightly. The domain authority information mentioned in the head section of the web page helps here against hiding the eyes. We dig the URL and obtain domain authority details which we compare with the ones mentioned in the web page. Here if the URL is same, which users notice in the web page then there will be a match otherwise it shows mismatch.

C. Webpage Scanning

Despite of incorrect web URL there are other anomalies also that deceive users. The deceive comes true when phishing web page which has its visual appearance and URL same as of a genuine web page, asks for credentials from users. The web page offers input controls to users and users getting deceived feed their original information. The technical aspects: Input controls are designed using html which are found in the web page. This states that webpage will surely have form tag and other input tags which will be required inside the form tag. Although it is possible that input controls can be used without forms and make them active using script codes so we analyses them in next sub section. This is possible that web page have more than one form control as per the needs of websites. Again it is to say that webpage may not mean as phishing webs if they don't ask for credentials or these don't have form controls. All the credentials are sent to other web servers when action attributes of the forms have malicious URLs. Also html tags may have hidden values to not show the content. The malicious web page will get advantage here that users will not see the information given in the web page but our approach will test it successfully. These all things are cared by scanning the web pages.

D. Script Scanning

We scan associated script codes e.g. java script to check whether the open web page has been compromised by any malicious websites or the active website itself can send users' credentials to some other web servers. It is also checked whether the tags which follow the definitions are tortured e.g. getting hidden. We first search for URLs in the associated script codes then checks their domains and associated authority details with the domain and its associated details of active website as well as details mentioned visually for users.

E. Working Approach

In order to explain our work we have divided our proposed approach into two phases. First phase scans emails received in user's mail box and detects whether the links given may be phishing. The second phase works when user clicks on any link of the email. We scan the script codes and look for URLs as described in following figure1.

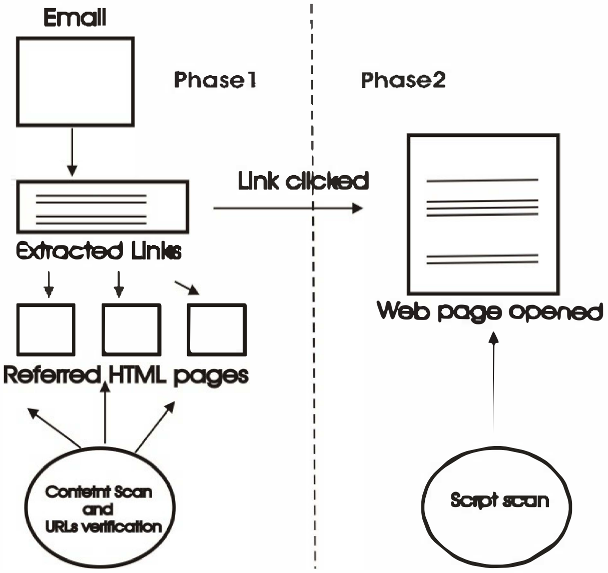


Figure 1: Phases in DC Scanner

Finding the URLs we get their domains and match them with the domain of active website. So the algorithms for phase1 and phase2 which have been given in Algorithm1 and Algorithm2 respectively detect phishing web pages.

```
DC Scanner: Phase I
Input: Link Addresses in an Email.
Output: Verified Links
Used variables are bold, italicized and underlined
I email_link[]=Anchor tags in Email
II found_URL[]=value(href attribute of email-link[])

Repeat below steps for every found_URL[]
1. domain_name=domain(found_URL)
2. page_content=html content(found_URL)
3. domain_tag[]= A body tag with (body . id/name =
                 domain_name)//As per defintion3
4. If domain_tag[0]=="" then // The web page doesn't follow
the design instruction
—Alert "This link may have phishing page";
—Stop
—Endif
5. If Array_Length(domain_tag[])>1 then
—Alert "This link may have phishing page";
—Stop
—Endif
6. If domain_name!=domain(domain_tag[0]) then // The link
url is different than what is claimed in the webpage
—Alert "This link may have phishing page";
—Stop
—Endif
7. author_tag=A head(h1,h2,h3) tag with (id/name =
              domain_name)
8. domain_info_url=innerHTML of author_tag
9. domain_info_web=Filtered whois(found_URL) // Use who
is command on the web URL and find a unique information
about domain authority
10. if domain_info_web!=domain_info_url then // what
organisation is being shown here is not the exactly
—-Alert("This link may have phishing page")
—-Endif
11. web_forms[]=HTML forms in page_content;
12. If web_forms[0]=="" then
—-Stop // No forms then no chances of attacks
—-Endif
13. while(web_forms[])
—directed_url=value of action attribute in web_forms[]
—if url_format(directed_url) then // url_format is a function
   to check if the action attribute contains a web URL.
—-if domain(directed_url)!=domain_name then // The action
     attribut directs to another website
——-Alert("This link may have phishing page")
——-End if
—-End if
—End if
—End while
```

1) *Phase 1:*

This phase starts when users open emails. The phase gets all the links associated in the emails and perform detailed analysis on every URLs and html pages referred one by one.

Here the scanner finds domains of the links selected and then looks for the design standards whether they have maintained or not. If not maintained then it alert the users. If the standards are maintained then it checks whether the head section of the web pages contain exactly the same domain's unique authority records what stored in DNS of Registrar or something else. Again html forms are looked for action attributes in the web pages. It is checked whether the action values are URLs or only a string like folder name or web page names.

The domains of the URLs obtained are checked whether they match with domains of active web pages or not. This way finally first phase completes its share of work to suspect over the web pages and URLs.

```
DC Scanner: Phase II
Inputs: Source codes of open web page
Output: Verified web page

1. page=HTML contents of open web page
2. script_sections[]=Parse page for <script> tags
3. page_domain=domain(page) // Find the domain of open
                              web page
4. while(script_sections[])
        —script_url[]=scan script_sections[] for urls
        —while(script_url[])
        —-url_domain[]=domain(script_url[]) // Get
        domains of urls found in script sections
                —-if url_domain[]!=page_domain then //
                The script_url redirects to malicious web
                servers
                —-Alert("This may be a Phishing page")
                —-Stop
                —-Endif
        —Endwhile
–Endwhile
5. domain_tag=A body tag with (id/name==page_domain)
 // As per the definiton3
6. author_tag=A head(h1,h2,h3) tag with
   (id/name=page_domain)
7. if ( domain_tag==hidden) then // website address is
   available in page but it is hidden
        —-Alert("This may be a Phishing page")
—-Endif
8. if ( author_tag==hidden) then // website authority info is
not shown as it is to forge the DC Scanner
—-Alert("This may be a Phishing page")
—Endif
```

2) *Phase 2:*

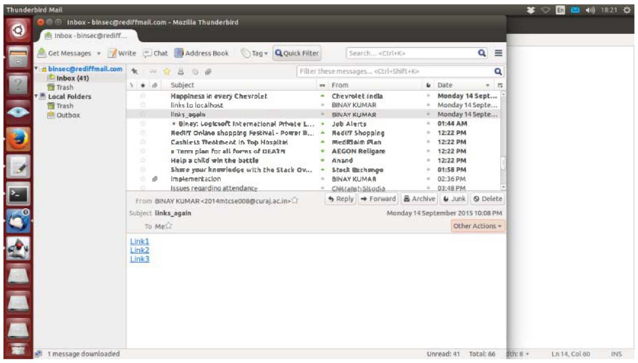
The second phase checks script of web pages to conclude whether the contents of forms can be sent to some other web servers. Also whether the html tags have been tried to get modified on events like key up, mouse hover etc. When the page is opened clicking link in the email then the second phase starts working. The scanner locates URLs in the script codes and again checks for their domains whether they are same as the domain of active web page or not. If phase1 and phase2 completes the task without any suspicion then website or URL are considered non-phishing ones and users not alerted.

IV. Implementation And Results

We implement our work in our university LAB on Intel core i5 computers with 2 GB RAM and Ubuntu 14.04 LTS as operating system. The programming language we use is java jdk1.8.

As the definitions what we propose are subjects of implementations so we design different web pages and keep them in different computers of the university Lab. For domain purpose we install bind9 and configure DNS servers in more than one computer.

Finally, we send emails to different students containing different links referring to different domains. Although, it is possible to develop a web browser plugin to scan the email automatically and alerts users with proper messages upon opening emails. We have used Mozilla Thunderbird, the email client download the emails as shown in figure2 and copy source codes of the email's message body. In figure2 the lower part showing Link1, Link2 and Link3 have URLs. The copied html contents we feed to a java program for phase1 of the DC Scanner. This works in three steps as per the algorithm1. The result of phase1 is shown in figure3.

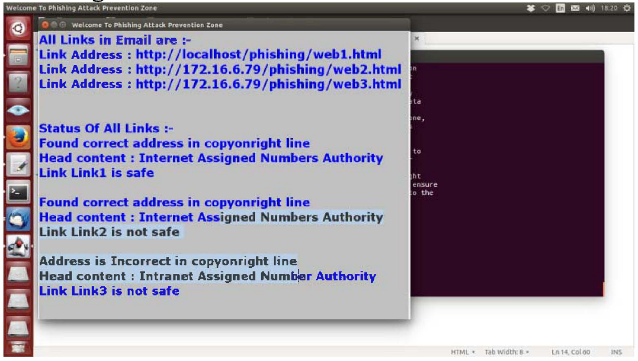


Subject	From	Date
Happiness in every Chevrolet	Chevrolet India	Monday 14 Sept...
links to localhost	BINAY KUMAR	Monday 14 Septe...
links_again	BINAY KUMAR	Monday 14 Septe...
Binay: Logicsoft International Private L...	Job Alerts	01:44 AM
Rediff Online shopping Festival - Power B...	Rediff Shopping	12:22 PM
Cashless Treatment in Top Hospital	MediClaim Plan	12:22 PM
Term plan for all forms of DEATH	AEGON Religare	12:22 PM
Help a child win the battle	Anand	12:22 PM
Share your knowledge with the Stack Ov...	Stack Exchange	01:58 PM
Implementation	BINAY KUMAR	02:36 PM
Issues regarding attendance	Chitransh Sisodia	03:48 PM

From BINAY KUMAR <2014mtcse008@curaj.ac.in>
Subject links_again
Monday 14 September 2015 10:08 PM
To Me

Link1
Link2
Link3

Figure 2: Emails collected in the Thunderbird


Welcome To Phishing Attack Prevention Zone
18:20
Welcome To Phishing Attack Prevention Zone
HTML
Tab Width: 8
Ln 14, Col 60
INS

```
All Links in Email are :-
Link Address : http://localhost/phishing/web1.html
Link Address : http://172.16.6.79/phishing/web2.html
Link Address : http://172.16.6.79/phishing/web3.html


Status Of All Links :-
Found correct address in copyonright line
Head content : Internet Assigned Numbers Authority
Link Link1 is safe

Found correct address in copyonright line
Head content : Internet Assigned Numbers Authority
Link Link2 is not safe

Address is Incorrect in copyonright line
Head content : Intranet Assigned Number Authority
Link Link3 is not safe
```

Figure 3: Results of Phase I

Figure 3 shows all URLs received in the email shown in figure2. After scan results show that Link2 has a form which redirects its form contents to other domain so it is not safe, here the URL where the form's contents are redirected is not shown. Link3 does not follow the definition properly. As result of who is command on its URL is "Internet Assigned Numbers Authority" which we have not shown in figure3 but content displayed in its head part is "Intranet Assigned Number Authority" so it is not safe again[19].

For phase2 we copy html contents of open page when a non-phishing link is clicked in email. This content as a file again we feed to another program written in java which parse the contents and analyze script codes to find URLs, tags and their attributes' modifiable codes. Finally this verifies the URLs' domains whether they are same as domains of web pages URLs. This properly alerts users regarding phishing nature of open web pages.

V. Future Work

Several vulnerabilities in web browsers have provided capabilities to phishers to install malware in the victim computers [6]. Of the attacks mentioned here, malware based phishing attack has become more exploitative today. We will extend our work as phishing scanner to identify and prevent malware based phishing attacks in next attempt.

VI. CONCLUSION

DC Scanner is an email scanner which identifies malicious URLs received in the email message opened by users. This work in two phases: first to get html contents of every links of the email body and mine the contents to verify domains of every links. For detailed verification of URLs we use domain registration information and compare it with the domain authority unique information displayed in the web pages. The second phase checks for malicious URLs in the script codes of the web pages. Also it has been checked whether the phishers have tried to modify the html tags and their attributes so that they could not be traced.

For bringing efficiency and correctness we have suggested some standards for web pages' design. All what we have proposed have been implemented and tested among our university students.

References

[1] R. Anderson, Security engineering, 2nd ed. John Wiley & Sons, 2008.

[2] Y. I. Khonji, Majid and A. Jones, "Phishing detection: a literature survey." in Communications Surveys & Tutorials. IEEE, 2013, pp. 2091–2121.

[3] [Online]. Available: http://www.hkma.gov.hk/eng/key-information/press-releases/2015/20150831-8.shtml

[4] [Online]Available:http://www.info.gov.hk/gia/general/201508/24/P201508240782.htm

[5] [Online]. Available: http://www.fraudwatchinternational.com/individualalert?fa_no=241555&mode=alert

[6] J. Milletary, "Technical trends in phishing attacks," CERT Coordination Center, Tech. Rep., December 2007.

[7] [Online]. Available: http://www.millersmiles.co.uk/email/ebay-notice--obligatory-verifying-invalid-user-ebay

[8] Almomani, Ammar, "A survey of phishing email filtering techniques," in Communications Surveys & Tutorials, 2013, pp. 2070–2090

[9] B. Q. Liu, Gang and L. Wenyin, "Automatic detection of phishing target from phishing webpage," in Pattern Recognition (ICPR), 2010 20th International Conference on. IEE, 2010.

[10] E.A.Chou, Neil, “Client-side defense against web-based identity theft,” in NDSS Symposium 2004, 2004.

[11] S.Görling, “The myth of user education,” in Virus Bulletin Conference, 2006.

[12] Chen, Kuan-Ta et. al. , “"fighting phishing with discriminative keypoint features,” in Internet Computing. IEE, 2009.

[13] Sheng, Steve et. al. „ “"an empirical analysis of phishing blacklists,” in Proceedings of the 6th Conference in Email and Anti-Spam, 2009.

[14] Ma, Justin et. al. , “Learning to detect malicious urls,” in ACM Transactions on Intelligent Systems and Technology (TIST), 2011.

[15] I. K. Ramesh, Gowtham and K. S. S. Kumar, “An efficacious method for detecting phishing webpages through target domain identification,” Decision Support Systems, 2014.

[16] K. V. Pradeepthi and A. Kannan, “Performance study of classification techniques for phishing url detection,” in Advanced Computing (ICoAC). IEE, 2014.

[17] J. J. Gupta, Anjali and K. Thakker, “Content based approach for detection of phishing sites,” in International Research Journal of Engineering and Technology, vol. 2. IRJET, 2015.

[18] J. I. H. Zhang, Yue and L. F. Cranor, “Cantina: a content-based approach to detecting phishing web sites,” in Proceedings of the 16th international conference on World Wide Web. ACM, 2007.

[19] Priyanka Gupta, Ankit Mundra,” Online in-auction fraud detection using online hybrid model”, 2015 International Conference on Computing, Communication & Automation (ICCCA), IEEE,2015.