Neural Comput & Applic (2017) 28:3629–3654
DOI 10.1007/s00521-016-2275-y

REVIEW

# Fighting against phishing attacks: state of the art and future challenges

**B. B. Gupta[1] · Aakanksha Tewari[1] · Ankit Kumar Jain[1] · Dharma P. Agrawal[2]**

Received: 8 January 2016 / Accepted: 2 March 2016 / Published online: 17 March 2016


**Abstract** In the last few years, phishing scams have rapidly grown posing huge threat to global Internet security. Today, phishing attack is one of the most common and serious threats over Internet where cyber attackers try to steal user's personal or financial credentials by using either malwares or social engineering. Detection of phishing attacks with high accuracy has always been an issue of great interest. Recent developments in phishing detection techniques have led to various new techniques, specially designed for phishing detection where accuracy is extremely important. Phishing problem is widely present as there are several ways to carry out such an attack, which implies that one solution is not adequate to address it. Two main issues are addressed in our paper. First, we discuss in detail phishing attacks, history of phishing attacks and motivation of attacker behind performing this attack. In addition, we also provide taxonomy of various types of phishing attacks. Second, we provide taxonomy of various solutions proposed in the literature to detect and defend from phishing attacks. In addition, we also discuss various issues and challenges faced in dealing with phishing attacks and spear phishing and how phishing is now targeting the emerging domain of IoT. We discuss various tools and datasets that are used by the researchers for the evaluation of their approaches. This provides better understanding of the problem, current solution space and future research scope to efficiently deal with such attacks.

**Keywords** Bag-of-word · Data mining · Key logger · Machine learning · Malware · Phishing · Social engineering · Soft computing · Spam · Visual similarity

## 1 Introduction

One of the most profitable crimes since past is "identity theft", which means to steal any person's identity. In a traditional term [1], criminals commit these either by killing the victim and pretend to be that person or steal confidential information from the garbage by accessing information from discarded letters, financial record, electricity bills and many others bills which are dumped without shredding them properly [2]. The term "phishing" is derived from the analogy of "fishing" for victims' passwords and credentials in the web. The phrase "ph" comes from "phone phreaking", which was very common technique that attacked telephone systems during 1970s. The word "phishing" was used for the first time over the Internet by a group of hackers in 1996, who stole America Online (AOL) accounts by tricking unaware AOL users into disclosing their passwords [1].

*Phishing* can be referred to as an automated identity theft, which takes the advantage of human nature and the Internet to trick millions of people and take a large amount of money. An IT industry research group Gartner showed in April 2004 that about 1.8 million American people had already given their information to phishers. It has been observed that in last few years phishing attacks have grown rapidly posing a real threat to global security. The main aim of these campaigns is to exploit the vulnerabilities

✉ B. B. Gupta
gupta.brij@gmail.com

Dharma P. Agrawal
dpa@cs.uc.edu

[1] National Institute of Technology Kurukshetra, Kurukshetra, India

[2] Center for Distributed and Mobile Computing, EECS University of Cincinnati, Cincinnati, OH, USA

present in the system, which may be either technical or due to user unawareness, which means that researchers have to provide defense against these attacks at both the technical and the user level. Researchers have tried to achieve the former by employing various approaches, and the latter can be feasible by increasing awareness and educating the Internet users. Phishing campaigns attempt to extract secret data from the victims, which may lead to substantial financial losses. Studies have shown that one-third of all the phishing attempts in 2013 were intended toward bank accounts or to gain other financial information [3]. Since 2012, financial phishing attacks were increased by 8.5 % as compared to 2011, an all-time high responsible for phishing attack [4] (Fig. 1).

In spite of causing severe financial damages to the users across the Internet, spam and phishing are still growing at a faster rate, and it will continue to do so as long as 1 out of 100,000 recipients actually responds to the phrases like "Click here" in spam emails. According to the Anti-Phishing Working Group (APWG) reports, phishing scams will keep growing with the use of more advanced technologies, and it will become the main threat over Internet, surpassing spam behind, as phishing scams are increasing 56 % per month [5].

In this paper, we present an overview of phishing attacks and many possible defense schemes. This survey gives a broader classification of defense mechanisms, and we provide a set of features used for phishing detection associated with these features ranked according to their ability to classify the phishing emails effectively. We also provide taxonomy of various solutions proposed in the literature that can detect and defend from phishing attacks. In addition, we also discuss various issues and challenges to deal with phishing attacks. We also summarize various tools and datasets used by the researchers for evaluating of their approaches.

The rest of the paper is organized as follows: Sect. 2 of the paper contains the history, background and statistics of phishing attacks. Section 3 describes phishing life cycle. Section 4 presents performance evaluation metrics for judging the anti-phishing system. Section 5 contains various tools and dataset used for evaluation. Section 6 presents taxonomy of various types of phishing attacks. Section 7 provides taxonomy of various types of phishing defense mechanism. Section 8 presents open issues and challenges against phishing detection, and finally, Sect. 9 concludes the paper.

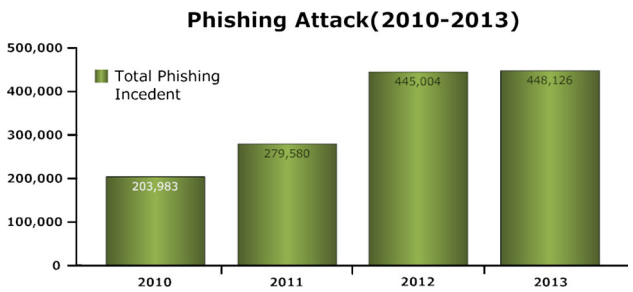


**Fig. 1** Phishing attack year over year (according to RSA Online Fraud Report, January 2014)

## 2 History, background and statistics

The word "phishing" was used for the first time over the Internet by a group of hackers in 1996, who stole America Online (AOL) accounts by tricking unaware AOL user into giving their passwords [1]. Table 1 shows growth rate of phishing starts from 1996 to 2014. According to the APWG report [6], the total number of unique phishing websites detected was 125,215 in the first quarter of 2014, which has increased approximately by 11 % in the last quarter of 2013 [7]. It is the second highest number of websites attacked in a quarter, and the first highest was 164,032 in Q1 of 2012. USA remains the most targeted country for these attacks [6]. Most of the phishing campaigns used maliciously registered domains and subdomains. The number of domains has increased from 260 million in April 2013 to 272 million in November 2013 [7, 8]. The attacks targeted 82,163

**Table 1** Evolution of phishing during 1996–2014

| Year | Events |
|---|---|
| 1996 | Term "phishing" was first used |
| 1997 | Media declared the evolution of a new attack called "phishing" |
| 1998 | Attackers started using message and newsgroups |
| 1999 | Use of mass mailing to escalate the phishing attacks |
| 2000 | First use of key loggers, phishers used it for getting login credentials |
| 2001 | Use of URLs to direct victim to a fake site |
| 2002 | Use of screen loggers |
| 2003 | Use of IM and IRC |
| 2004 | Evolution of "pharming" |
| 2005 | Term "spear phishing" was first used |
| 2006 | First phishing over VoIP |
| 2007 | More than $3 billion lost to phishing scams |
| 2009 | Symantec Hosted Services blocked phishing attacks impersonating 1079 different organizations |
| 2010 | Facebook attracted more phishing attacks than Google and IRS |
| 2012 | 6 million unique malware samples were identified |
| 2013 | Red October operation attacked more than 69 countries |
| 2014 | 750,000 malicious emails were sent using IoT devices, i.e., refrigerators and smart TVs [123] |
| 2015 | Spear phishing reached its peak in manufacturing and wholesale industries |

unique domain names, which is again significantly larger than 53,685 during the first half of 2013. Out of the 22,831 registered fraud domains, 1541 used well-known brand names. Rather than using domain names, some of the attack attempts used IP address, and statistics showed about 2400 such attacks used around 840 IP addresses [9].

Accordingly to a survey in 2013 [10], 62 % organizations were found to be a victim of spear phishing, whereas the survey by InfoSecurity [11] showed that 42 % organizations had faced these attacks. Overall 20 % (18 % in RSA and 32 % in InfoSecurity) said that they have not faced such attacks and 21 % did not know whether that happened or not. The organizations with more than 1000 employees have a higher probability to become a victim of spear phishing [10, 12].

The United States Computer Emergency Readiness Team collected security incident reports from federal, state and local government agencies and processed 107,655 incident reports in 2011, with 43,889 of them involving federal agencies. After processing these incident reports, they found that more than half of those incident reports (Approx. 51.2 %) came from phishing (as shown in Fig. 2). Therefore, for getting a foot into the door of a government network, most popular means is phishing to the hackers by a wide margin [5].

We also studied statistics of phishing attacks using eCrime Trend Reports. According to [13], in fourth quarter of 2013, .com is the most uses domain for phishing attacks with 41 %, followed by.net with 6 %, .org with 5 %, .br with 4 % and remaining IP address based with 4 % (as shown in Fig. 3).

We also found that USA is the most popular country for hosting phishing websites with 45 %. Next most popular phishing websites hosting country is Germany with 6 %, Canada with 3 %, France with 5 %, UK with 4 %, Brazil with 3 %, Russia with 2 % and Poland with 2 % (as shown in Fig. 4).

According to APWG's recent report about phishing scams during January to September in 2015 [14], Business Email Compromise (BEC) was dominant, and these attacks deploy spear phishing to trick big organizations or a specific employee. The global rate of infected computers was found to be 36.51 % in first quarter, 32.21 % in the second and 32.12 % in the third. During the first three quarters of 2015, Internet Service Providers (ISPs) were the most targeted sector.

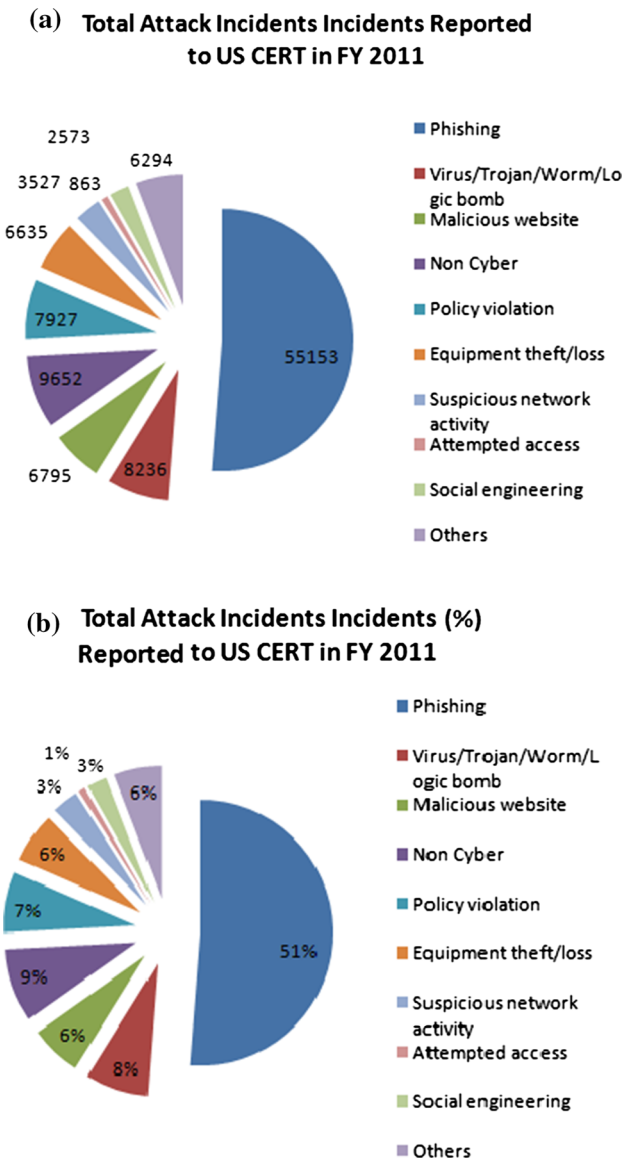


**Fig. 2** **a** Summary of total incidents reported to US CERT in FY 2011 and **b** summary of total incidents (%) reported to US CERT in FY 2011

## 3 Motivation and phishing life cycle

### 3.1 Motivation

Phishers always take advantage of human nature that generally ignores critical warning messages. Lack of awareness about the phishing attacks in the society is also the main reason why phishing attacks have been so much successful. Whenever any researcher came with some technique to prevent these attacks, phishers try to find out associated loophole to commit successful attacks. Remembering the fact that phishing mainly used for financial gains, there are other factors that also motivate phishers to commit the crime. Motivations behind these activities are as below:

- Theft of login credentials: Phisher steals login credentials of online services like eBay, Amazon and Gmail from the user using spoofed email as warning message to change password and provided hyperlink.

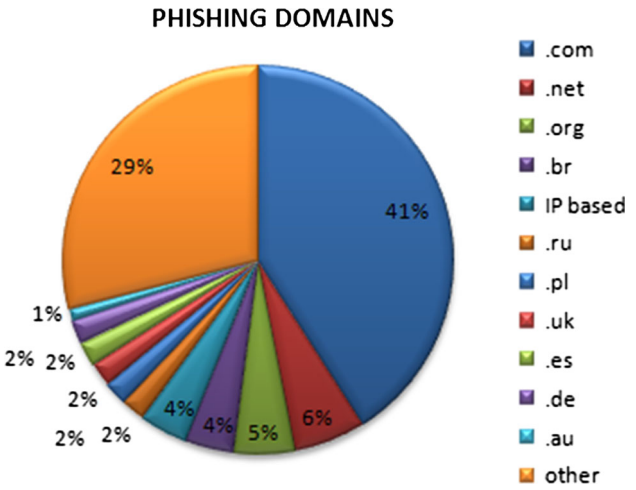


**Fig. 3** Statistics of phishing websites based on domain (E-crime report 2013 Q4)

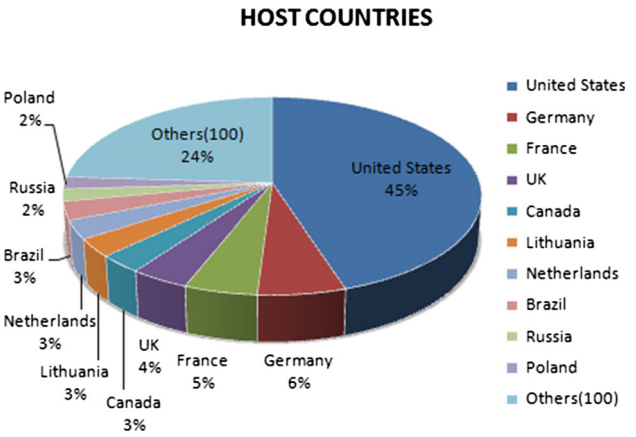


**Fig. 4** Statistics of phishing websites based on host countries

- Theft of banking credentials: Online login credentials and credit card details such as card number, expiry and issue dates, cardholder's name, CCV number and several other popular banking organizations like PayPal, OnlineSBI, HDFC and Citibank.
- Capture of personal information: Personal information, such as address and telephone number, is highly saleable and in constant demand by direct marketing companies.
- Theft of trade secrets and confidential documents: With spear phishing techniques, phishers are targeting specific organizations for acquisition of proprietary information and used directly or sold to interested parties.
- Fame and notoriety: A very interesting psychological aspect of phishing in which information is phished not for financial gain but carried out mainly to gain recognition and notoriety among their peers.
- Exploit security holes: People who are curious to find out how robust a particular system is may try to write programs to break somebody else's system to launch phishing attacks or to sell the compromised systems to other phishers.
- Attack Propagation: Through a mixture of spear phishing and bot agent installations, phishers can use a single compromised host as an internal "jump point" within the organization for future attack.

### 3.2 Phishing life cycle

The following stages are involved in phishing Attack as shown in Fig. 5.

*Stage 1*: *Planning and setup* In the first step, the attackers identify the target organization or individual or a nation. Then, their task is to get details about the organization and its network. It can be done by visiting the place physically or monitor the traffic going in and out of the network. The next step is to set up the attacks by using a feasible means, e.g., website or emails having malicious links, which may redirect the victim to some fraud web page.

*Stage 2*: *Phishing* The next step is to send these spoofed emails, e.g., masqueraded as some reputed banking organization to the victim using the collected email addresses, which ask user to update some information urgently by clicking on some malicious link. The emails might be sent to individuals or specific person in an organization.

*Stage 3*: *Break-in/infiltration* As soon as the victim opens the fraud link, either a malware is installed on the system which allows the attacker to intrude the system and change its configuration or access rights are changed accordingly. In other cases, it might lead to some fake page that asks for credentials.

*Stage 4*: *Data collection* Once the attackers get access to the user's system, the required data are extracted, and if the user gives his account details to the attacker, they can now access his/her account, and this may led to financial losses to the victim. In case of malware attacks, now the attacker may get remote access to the system and get the data he wants what so ever, or the compromised systems could be used for DDos attacks, etc. Phishers use rootkits to hide their malwares.

*Stage 5: Break-out/exfiltration* After getting the required information, the phisher now removes all the evidences, i.e., the false websites accounts. It is also observed that they track the degree of success of their attack for refining future attacks.

## 4 Performance evaluation metrics

The goal of most classifiers is to perform binary classification, i.e., into phishing or a legitimate category where four possibilities exist. Assume that $N_H$ denotes the total

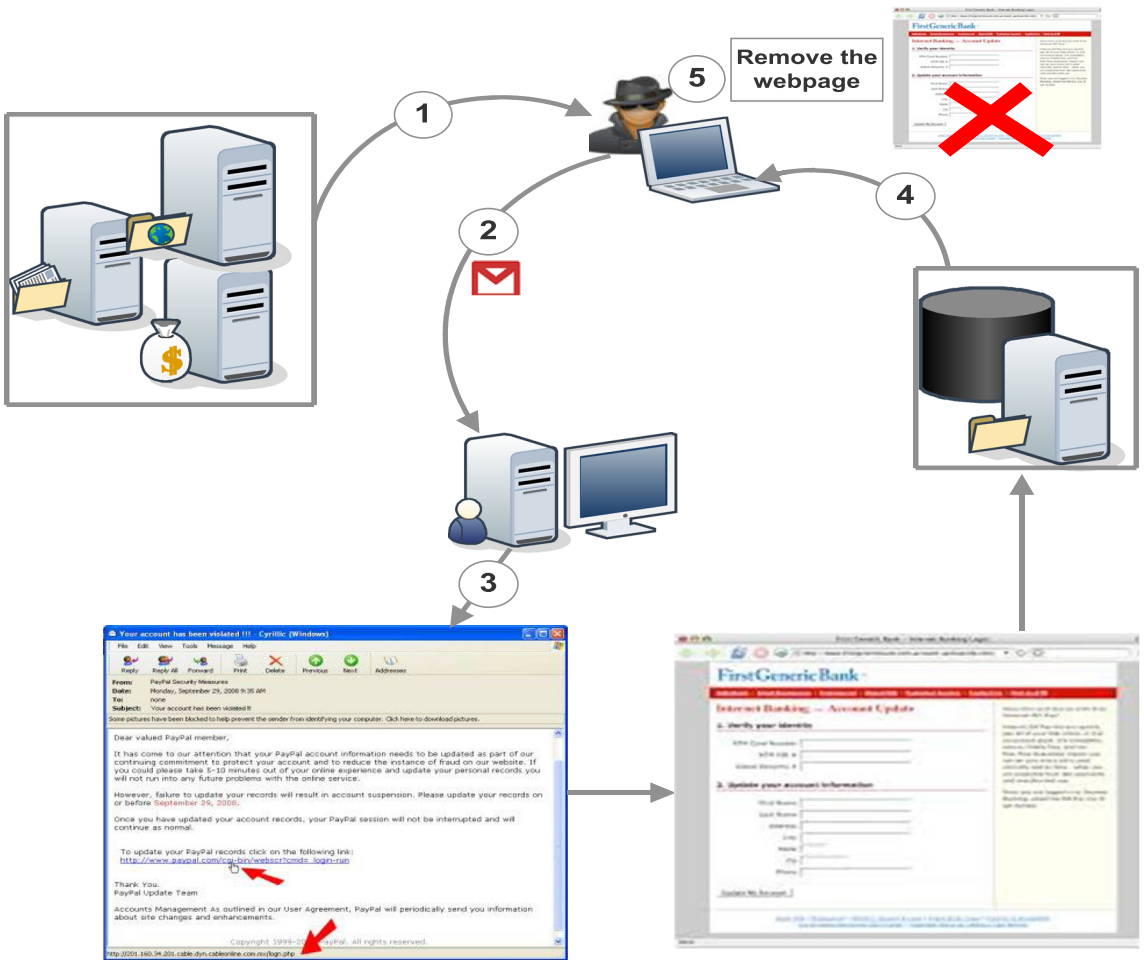


**Fig. 5** Phishing life cycle

number of ham emails and $N_P$ denotes the total number of phishing emails. If $(n_h \rightarrow H)$ denotes ham messages, then $(n_p \rightarrow H)$ denotes phishing emails classified as ham $(n_h \rightarrow P)$ denotes ham mails classified as phishing and $(n_p \rightarrow P)$ denotes phishing emails classified as phishing. The evaluation metrics used in this case are [15, 16]:

1. *True positive* (*TP*): This denotes the ratio of the number of phishing emails identified correctly as:

$$\mathrm{TP} = \frac{n_p \rightarrow P}{N_P}. \tag{1}$$

2. *True negative* (*TN*): This denotes the ratio of the number of ham emails identified correctly as:

$$\mathrm{TN} = \frac{n_h \rightarrow H}{N_H}. \tag{2}$$

3. *False positive* (*FP*): This denoting the ratio of the number of ham emails classified as phishing, as:

$$\mathrm{FP} = \frac{n_h \rightarrow P}{N_H}. \tag{3}$$

4. *False negative* (*FN*): Ratio denoting the number of phishing emails classified as ham, as:

$$\mathrm{FN} = \frac{n_p \rightarrow H}{N_P}. \tag{4}$$

5. *Precision* (*p*): Measures the rate of phishing emails which are identified correctly as the emails detected as phishing:

$$p = \frac{n_h \rightarrow P}{n_p \rightarrow P + n_h \rightarrow P}. \tag{5}$$

6. *Recall* (*r*): Measures the rate of phishing emails which are identified correctly as existing phishing emails:

$$r = \frac{n_p \rightarrow P}{n_p \rightarrow P + n_p \rightarrow H}. \tag{6}$$

7. *$f_1$ score*: This is the harmonic mean of Precision and Recall:

$$f_1 = \frac{2p.r}{p+r}. \tag{7}$$

8. *Accuracy* (*ACC*): Measures overall correctly identified emails:

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{8}$$

9. *Specificity* (*S*): Measures correctly identified ham emails:

$$S = \frac{\text{TP}}{\text{TN} + \text{FP}}. \tag{9}$$

In some of the studies where specific features are used to identify the category of an email, the evaluation metrics used are [17]:

1. *Entropy* (*E*): Measures the amount of disorder or disturbance in the system. It can be calculated as:

$$E(S) = \sum_{i=1}^{N} -p_i \log_2 p_i, \tag{10}$$

where $N$ number of classes in the dataset, $S$ dataset, and $p_i$ probability of an email belonging to class $i$.

2. *Information gain* (*IG*): Measures decrease in the value of entropy when a particular feature is used. IG($S$, $A$) is the information gain of dataset $S$ over the attribute $A$ and can be obtained as:

$$\text{IG}(S, A) = E(S) - \sum_{v \in \text{value}(A)} \frac{S_\text{v}}{s} E(S_\text{v}), \tag{11}$$

where $S_\text{v}$ the number of attributes in $S$ with $A$ has the value of $v$, and $E(S_\text{v})$ entropy of the subset $S_\text{v}$ in $S$.

## 5 Datasets and tools used for evaluation

Many datasets are freely available on the Internet, which are commonly used for the experimentation and evaluation of phishing detection algorithm. This section presents a brief description of most popular phishing and ham datasets.

### 5.1 Standard datasets

*Phishing archive*: The Anti-Phishing Work group's "Phishing Archive", is a record of phishing attacks, which are either reported to APWG or detected by APWG [18]. This dataset is used by Dhamija et al. [19] and Abburous et al. [20] in performing their evaluation.

*PhishTank*: PhishTank website stores the phishing data reported by the user. The phishing data are shared via the website and can also be accessed through an API [21].

*Corpora*: The corpora of the SpamAssassin project contains three parts of spam corpora, easy ham which could be easily distinguishable from spam and hard ham which are hard to be distinguished from spam [22]. A recent addition to this corpus is easy ham_2, a ham dataset, and spam_2, a spam dataset. This dataset is used by I. Fette et al. [23] for the evaluation of their algorithm PILFER and M. Khonji et al. [24] for implementing LUA algorithm.

*Enron dataset*: The Enron Dataset [25] has been collected by the CALO [26] project consisting of more than 150 employees. Initially, the dataset had some integrity problems, but it was fixed by Bryan Klimt and Yiming Yang [27]. This dataset is used by Georgala et al. [28]. The dataset includes comprises of personal emails. The ham messages are collected from six Enron employees and the TREC 2005 Spam Track public corpus. The dataset contains approximately 50,000 spam and 43,000 ham emails. It is considered as a benchmark dataset.

*TREC*: Other commonly used dataset are TREC corpus [29] used by Al-Daeef et al. with the copyright being held by waterloo university. The TREC 2005 corpus has been created for spam evaluation, and it contains 92,189 emails ordered chronologically. The dataset contains 39,399 ham emails and 52,790 spam emails. TREC 2006 and 2007 are also available from their websites.

*IronPorts*: It has been designed in 2000 by Scott Banister and Scott Weiss, to provide defense against any Internet threat. It was acquired by Cisco in 2007. Iron Port's corpus [30] was used by Tyler Moore et al. [31]. Dataset is a collection of messages that arrive at their spam traps and emails submitted by the customers.

Iron Port's SpamCop [32], founded by Jullian Haight in 1998 and acquired by Iron Port in 2003, is a service which keeps a record of spam reported by the recipients of commercial emails or UBEs (Unsolicited Bulk Emails). It has a number of spam traps at geographically different locations, thereby acting as a major contributor to Iron Port corpus. SpamCop processes all these reported spam and creates a list of systems used in sending those emails which are blacklisted by SpamCop.

*Phishload*: In 2012, the phishing database was created by Max-Emanuel Maurer, which is named as Phishload [33] and contains HTML code, URL, and other information related to phishing websites. It also has about a 1000 legitimate and target websites (Table 2).

### 5.2 Various tools used for evaluation

This section describes various tools used for experimental purpose and to judge the accuracy of an anti-phishing

**Table 2** Dataset description

| Dataset | Description | No. of instances |
|---|---|---|
| APWG's phishing archive | A record of phishing attacks which are reported to or detected by APWG | Dynamic dataset |
| Phish tank | Stores the phishing data reported by the user | Dynamic dataset |
| SpamAssassin's corpora | A collection of emails, appropriate for testing spam filtering system, it is divided into three parts | 4150 ham and 1897 spam messages |
| TREC corpus | Provides a standard evaluation of current and proposed SPAM filtering approaches | 39,399 ham and 52,790 spam messages |
| Enron | Collected by the CALO project from more than 150 employees | 43,000 ham and 50,000 spam messages |

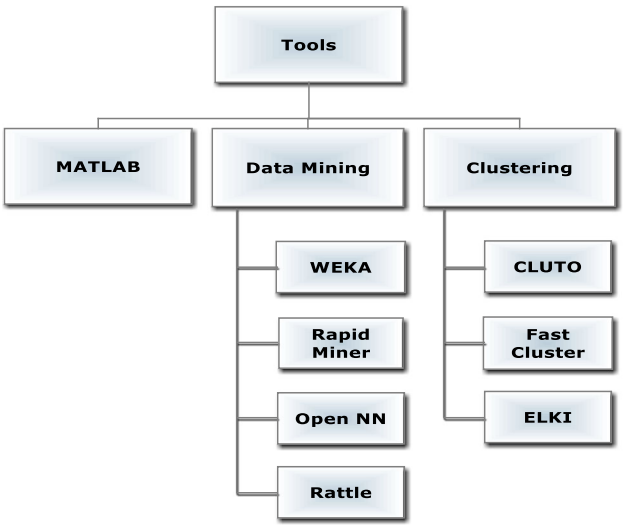


**Fig. 6** Tools used for evaluation

system. A researcher can select a tool depending on various parameters and algorithms used, e.g., if an approach uses a data mining algorithm for phishing detection, then WEKA tool can be used. Figure 6 shows numerous tools that can be used for evaluation of phishing detection, and Table 3 gives an overview of these tools and their possible application in various fields.

## 6 Taxonomy of phishing attacks

Phishing attacks are broadly classified into two categories: social engineering- and malware-based phishing attacks. In social engineering-based phishing, the attackers try to acquire the targets' credentials by using some fake website or sending fake emails that appear to be legitimate to trick the user [34–36]. Social engineering, also known as deceptive phishing, can be further classified as: (1) email phishing and (2) website phishing. Similarly, malware-based phishing attacks use a variety of malicious programs, which are primarily unwanted software running on target's system. These attacks can be further classified as: key loggers/screen loggers, session hijacking, host file poisoning, DNS phishing and content injection. The classification of phishing attacks is shown in Fig. 7.

Some of the mechanism or measures used to carry on these phishing attacks are summarized in the following paragraphs.

### 6.1 Phishing using compromised web servers

In these categories, the attackers search for vulnerable servers and install a secret exit or a backdoor that enables them to access a compromised server if the server is a web server. Phishing websites are downloaded, which then start receiving traffic and victim begins to access the malicious contents [37]. In the study conducted by Tyler Moore et al. [38], it was found that 76 % of the examined phishing websites were hosted on a compromised web server.

### 6.2 Phishing through botnets

A botnet refers to a network of infected computers, which are controlled by an attacker from a remote location, being large in size such that they pose a serious threat when used as denial-of-service (DDoS) attacks the attackers these days also make use of botnets for sending spam emails and phishing attacks [37]. In a study by Cipher Trust (an email security company) in October 2004, it has been shown that 70 % of recorded phishing mails are sent using one of five active botnets. Other observations also showed that a large number of unrecorded botnets are in use causing such attacks. Botnets can generate traffic to consume very high bandwidth, and a collection of 20,000 such machines can take down about 90 % of the websites [39, 40].

### 6.3 Phishing through port redirection

This mechanism makes it difficult to trace the location of the source of attack. Here, no phishing content is uploaded directly. Instead port redirection services are used by

**Table 3** Experimental tools and their applications

| Tool | Description | Applications |
|---|---|---|
| MATLAB [124] | Provides built-in math functions and tools by which one can get results faster | Image processing, signal processing and communication, computational biology |
| WEKA [125] | A collection of machine learning algorithms which are directly applicable to the datasets | Preprocessing, classification, clustering, etc., of datasets |
| Rapid Miner [126] | Implemented as a client–server model. The functionalities of RapidMiner can be extended by using plug-ins | Business and industrial application and research, training and application development |
| Open NN [127] | An open-source library which implements neural networks | Pattern recognition, function regression, time series prediction |
| Rattle [128] | Open-source data mining software written in R statistical programming language | Application and model generation, statistical analysis |
| CLUTO [129] | A software toolkit which can be used to cluster and analyze the properties of low as well as high-dimensional datasets | GIS, purchase and transactions, web science |
| Fast cluster [130] | Provides two interfaces for the standard software: R and Python | Developing hierarchical clustering on data |
| ELKI [131] | Provides data structures such as the R*-tree | Environment for developing KDD-applications supported by index-structures |

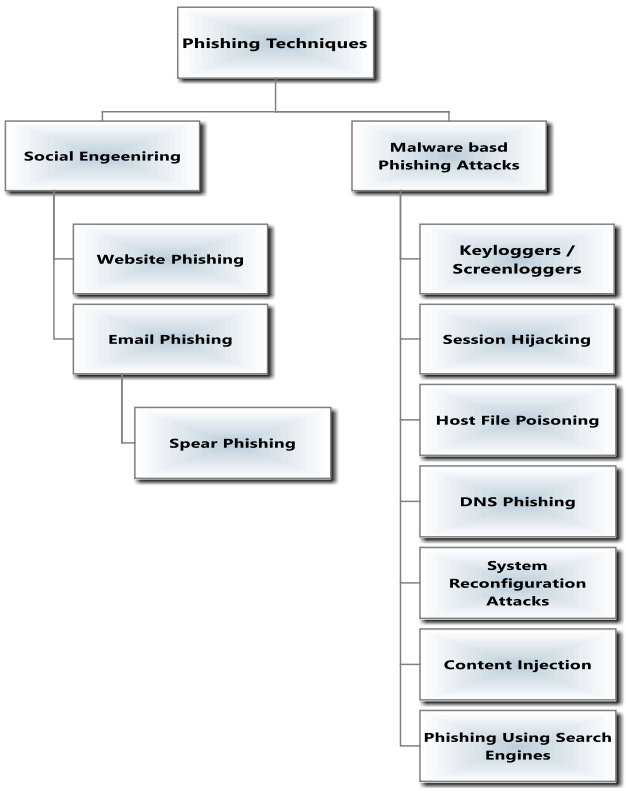


**Fig. 7** Types of phishing attacks

the attackers designed for purpose of re-routing web requests sent to the server to some other remote web server [37]. The tool generally used for port redirection is Fpipe [41].

## 6.4 Social engineering

Social engineering attacks intend to acquire victim's identity or other confidential information through spoofed or fake emails. Social engineering attacks are brought into action with similar motive as that of hacking, i.e., to acquiring illegal access to a system or gain confidential information about an organization or an individual, network intrusion, etc. The common targets include big corporations, military and government agencies [42]. Social engineering attacks target at two levels: physical and psychological [43].

The first target is the physical setup where the attacks are to take place, and it may be the workplace, the phone and even online. The attackers lay stress on finding a way to create a favorable psychological environment in order to make that attack work. Despite the method used, the primary goal is to make the victim believe that the attacker is a genuine person so that they can give him the required details [44]. They never try to get a lot of information at one time from one user. Instead, the attackers try to get small details from many people in order to gain their trust.

### *6.4.1 Website phishing*

Website phishing is a phonological attack with an objective of targeting a particular person rather than a system. These attacks are very easy to put into action due to the fact that creating a phishing website that is an exact replica of some legitimate site is not a problem for the attacker [45]. The main aim is to defraud people in order to gain their personal and financial details. Moreover, it is a very complex

task to detect phishing websites as it is primarily a combination of technical and social problems. These attacks aim to compromise an individual or an organizations' confidential information.

### 6.4.2 *Email phishing*

The phisher's first step is to launch a phishing website; then, it sends large volume of fake mails which may ask the user to click on a link within that mail and may give away his/her identity or other personal information which is passed onto the phisher by a phishing server. The phisher then uses the victim's identity to gain illegal financial benefits or for some other purpose. Despite the fact that phishing emails have been developed in a better style over time, there are still a number of measures or clues that indicate their deceptive nature. A new variant of email phishing that has become increasingly common these days is spear phishing in which the *type and the targets* of the phishing attack are the main concern.

*6.4.2.1 Protection from phishing emails* Phishing email message transportation is represented in Fig. 8 [12]. To detect phishing emails from ham emails, the framework in an online mode is set between message transfer agent (MTA) and mail user agent (MUA) so as to stop phishing email from reaching to victim's account (before it get to user).

*MTA* (*message transfer agent*): Acts as a post office for storing and acting as an email carrier.

*MUA* (*mail user agent*): A software program using for retrieving emails like "Microsoft outlook".

*MDA* (*message delivery agents*): Act as mailboxes, which store messages (as much as their volume will allow) until the recipients check the box.

*Phisher*: Malicious user who sends a phishing message to a potential victim.

*Victim*: User who may open for phishing email and become the target of a phisher.

An overview of email data parts is shown in Fig. 9 [46].

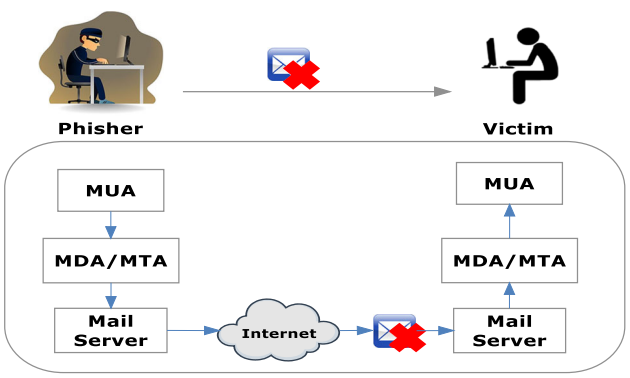


**Fig. 8** Phishing email message transportation [12]

*6.4.2.2 Spear phishing* Spear phishing may be defined as "highly targeted phishing aimed at specific individuals or groups within an organization" [47]. Spear phishing was first studied in 2005 as context aware phishing by Jacobson et al. [43]. A few years later, Jagatic et al. [42] found that the number of users who become a victim in spear phishing is 4.5 times larger than the general phishing attacks.

*6.4.2.3 Stages in spear phishing attacks* The spear phishing attacks start with the attacker collecting information about the organization and end with getting access to the required information. Figure 10 shows the steps involved in spear phishing [48].

*6.4.2.4 Spear phishing: one of the biggest cyber security threats* Spear phishing has become more common these days. The phisher disguises the email to look as it came from someone in the same organization, which increases the probability for the attack to be successful as people are more to open an email if appears to be coming from someone familiar making these attacks very hard to detect and the attacker is successful in breaching the organization without anyone knowing it. The US Department of Defense had also once been a target of spear phishing which caused the Joint Task Force-Global Network to educate the employees about these kinds of attacks [47, 49].

## 6.5 Malware-based phishing

Malware is malicious software that is usually installed on a machine without the knowledge of a victim, and sometimes even the victim can be tricked into downloading anti-virus software while it really is the virus or malware itself [50, 51]. Malware can access user's confidential data and send it to the phisher. Malware takes advantage of the gaps in the browser software and operating systems, or make use of deceptive techniques to encourage the victim to execute the malicious code.

According to APWG first quarter report of 2013 [3], around 5 million malware samples are reported by the APWG company PandaLabs which has increased set of new malware samples to be 15 million. Trojans remain the most commonly used malware, which is 72 % of all malware samples found. The total number of infected systems across the globe is about 33 %. China has the highest number of infected systems (Table 4).

### 6.5.1 *Key loggers and screen loggers*

Key loggers also pose severe threat to the systems as people are unable to detect their presence, and the screen recording software has made it even worse the situation due to key logging as virtual keyboards have no utility.

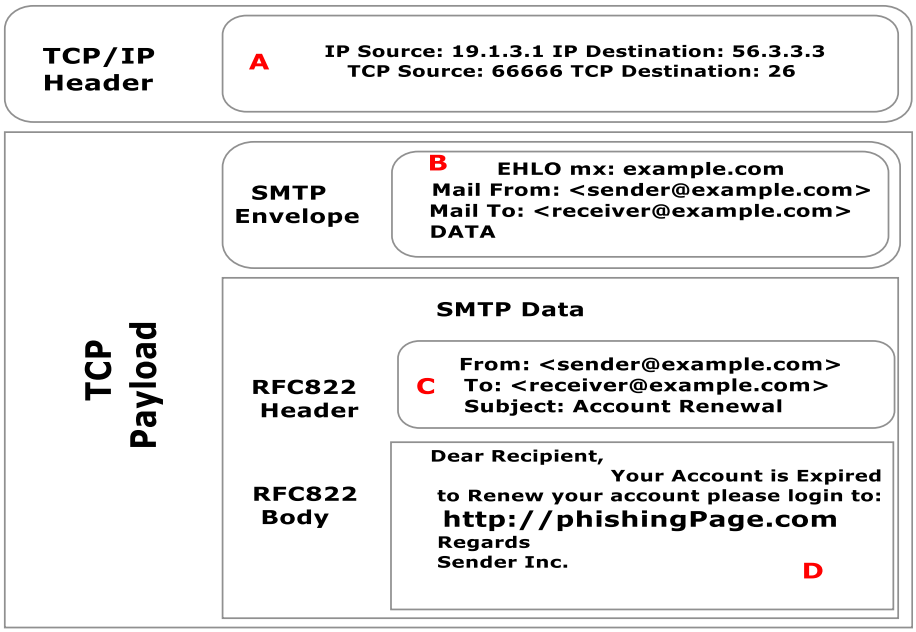


**Fig. 9** An overview of email data parts [46]


**Fig. 10** Stages in phishing attacks

**Table 4** Percentage of all malwares captured in Q1 of 2014 [6]

| Malware type | Overall % in malware samples |
|---|---|
| Trojan | 79.70 |
| Virus | 6.71 |
| Worms | 6.06 |
| Adware/spyware | 3.62 |
| Others | 3.91 |

Key loggers can be categorized as: hardware key loggers and software key loggers [52].

1. *Hardware key loggers*: Hardware key loggers are devices that record the data that are entered via keyboards into their memory and do not make use of any resources of the system as any anti-viral software cannot recognize them, and also these are very small in size and hide themselves well in a system, and they may send key hits to some attacker at remote location [48].
2. *Software key loggers*: Software key loggers examine the data of the operating system and the data entered through keyboard of the victim's system and record them at remote locations which are later sent to the attacker. Virtual keyboards are helpful and are not affected by these softwares as virtual keyboard data are entered by a mouse [48, 52].
3. *Screen recording software*: Screen recording software captures the screen and mouse movements for monitoring. But these are also used adversely which makes virtual keyboard not a safe option. This software records the activities going on at the screen like keystrokes using virtual keyboards [48].

### 6.5.2 Session hijacking

Session hijacks can occur either at *Network Level* or *Application Level*. At network layer session, hijack involves interfering TCP and UDP sessions, whereas session hijack at application level involves interfering HTTP sessions [53].

1. *TCP session/hijack:* In this case, an existing TCP connection between any two communicating systems is intercepted and the attacker disguises himself to be one of them and redirects the TCP traffic toward him/her by inserting fake IP packets so that the processing of the commands is done by the authenticated host. It desynchronizes the session between the actual communicating hosts. Authentication is done only at the time of the establishment of a connection; thus, an

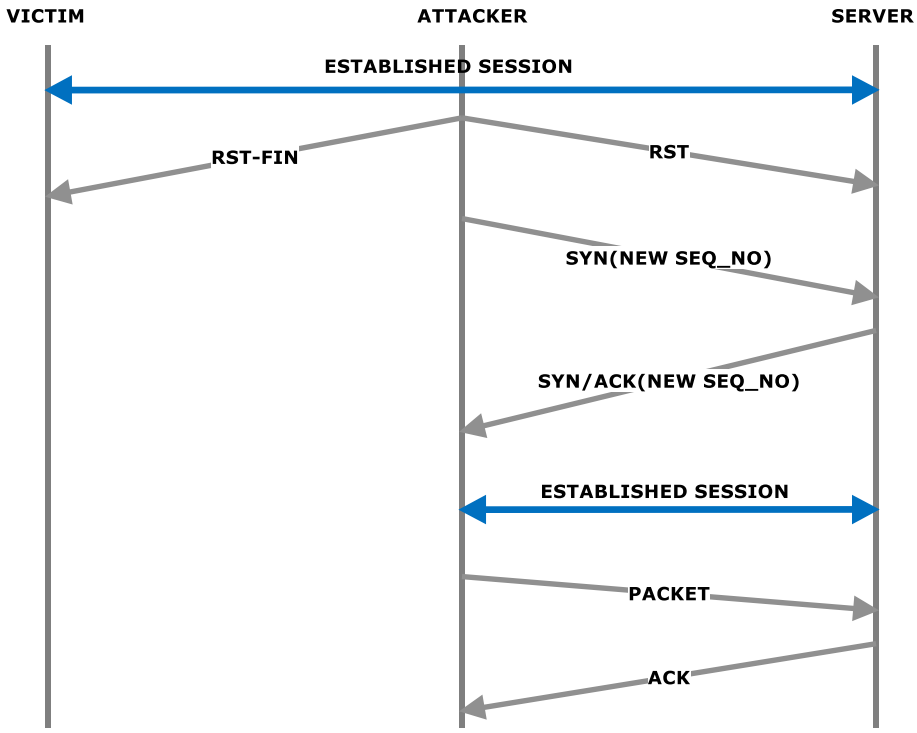


**Fig. 11** TCP session hijack

already established connection can be hijacked without any authentication (Fig. 11).

2. *UDP session hijacking:* It is easier to hijack a UDP session as compared to a TCP session as synchronization, and sequencing of packets is not required. In this case, the attacker sends a fake UDP reply to the host on the behalf of server before the server can respond.
3. *Hijacking application levels*: At application level, either a new session is initiated by using a stolen data or hijacking an existing session can also take place.

*HTTP session hijack:* IDs are obtained corresponding to a session. The IDs are only unique identifier for an HTTP session and can be extracted from the URL that the browser receives for the HTTP GET request, cookies at the clients system and the form fields as shown in Fig. 12.

4. *Session hijacking in WLANs:* Session hijacking has become a serious threat to WLANs as it exploits vulnerabilities of the network and can be brought into action using commercial off-the-shelf tools. The attacker forces a legitimate station to disconnect itself from its access point and using that station's address; the attacker then connects itself to that access point [54].

### 6.5.3 Host file poisoning

Hosts file poisoning refers to injecting new entries for websites into a machine's host file, which redirects the websites to another site. When a client inputs a URL, it is converted into an IP address before sending over the Internet; hackers have bogus address transmitted by poisoning the host files, redirecting the user to a fraud website where they are required to give their personal information [55].

### 6.5.4 DNS phishing

In DNS phishing attacks, initially a reprobate access point is created where the attacker runs a fake DNS server which tempts the client to connect to it. Using this fake DNS server, particular sites are redirected to the attacker's phishing server [56]. Figure 13 shows the mechanism of DNS phishing attack.

### 6.5.5 System reconfiguration attacks

In system reconfiguration attack, the user receives a message asking to reconfigure the computer settings from the attacker. That message may come from a web address which appears to be a reliable source [57].

System reconfiguration attack can be categorized as:

1. *Pharming*: It is also referred to as host name lookup attacks, which interfere with the host file of the victim's system and modifies the target address so that it directs to some other malicious location.

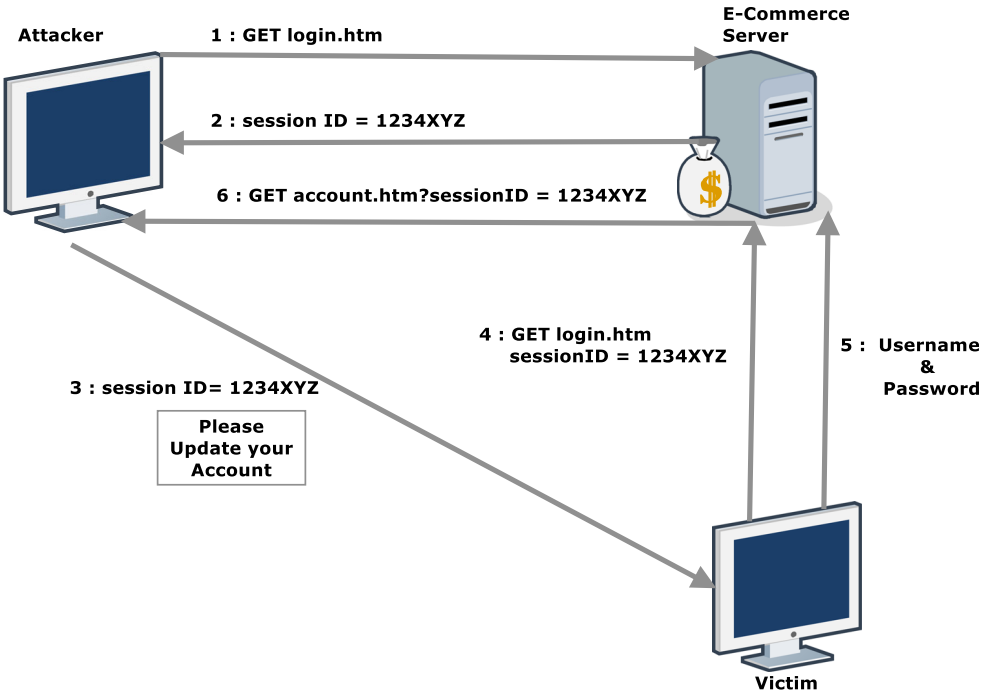


**Fig. 12** HTTP session hijack

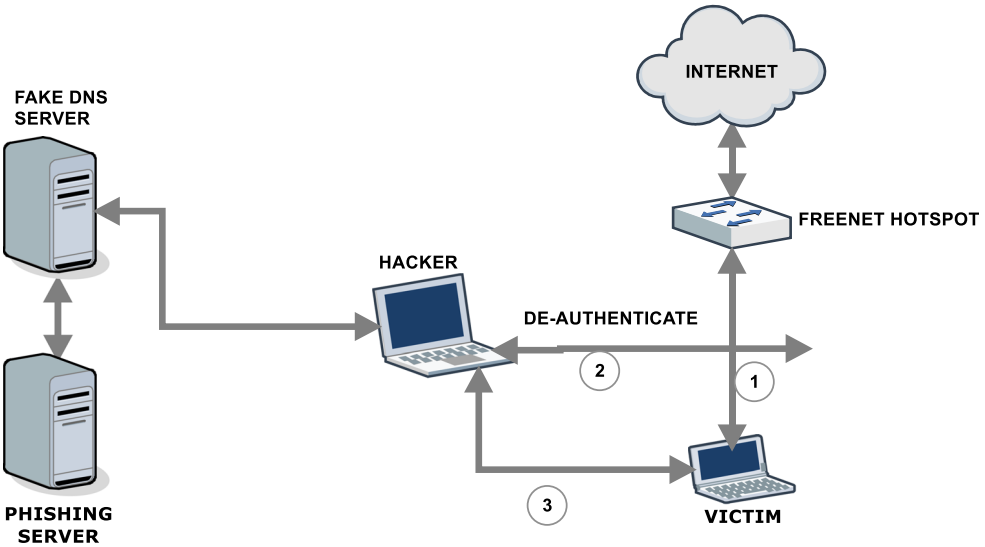


**Fig. 13** DNS phishing attacks mechanism

2. *Proxy attack*: Another type of reconfiguration attack is the installation of a proxy through which the Internet traffic is passed from the victim to the server so that the attacker is able to extract confidential information from it.

### 6.5.6 Content injection

These attacks can easily take place if an application is unable to handle given data by the user in a proper manner; attacker can give contents of that application through some parameter value, which results into a modified page that appears to be from a trusted domain. An attacker first identifies a vulnerable parameter and then crafts a link by making a little change in a valid request which is sent to the user [58].

### 6.5.7 Phishing through search engines

Phishing attacks involving search engines direct the user to certain online shopping sites where cost for the products or services is low which may tempt the user to buy the product by giving the credit card details at that


**Fig. 14** Spamdexing stages

**Table 5** Overview of malware attacks

| Type of attack | Description | Safety mechanisms |
|---|---|---|
| Key loggers/screen loggers | Record the data that is entered via keyboards or take snapshots of screen into their memory | Signature or heuristic based anti-key/screen loggers, e.g., zemana, spy shelter and data guard |
| Session hijack | Can occur either at *Network Level* involves interfering TCP and UDP sessions or at *Application Level* interfering HTTP sessions | Use of HTTPS, i.e., obtaining and deploying SSL certificate and one-time cookies |
| Host file poisoning and DNS poisoning | New entries for websites into a machine's hosts file which redirects the websites to another site | Securing the internal machine, use of DNS SEC, use of IDS, one should not rely on DNS for system security |
| System reconfiguration | The user receives a message from the attacker asking to reconfigure the computer settings that may appear to be a reliable source | Use of modified DHCP, e.g., DHCP with secure state estimation |
| Content injection | Attacker can give some contents to that application through some parameter value which results into a modified page that seems to be from a trusted domain | Context-sensitive string evaluation, use of runtime heap-spraying detector, use of web application firewalls, limiting data privileges by context, etc. |
| Phishing using search engines | A malicious technique used in web pages in order to achieve good rank in search results | Use of blacklists and whitelists |

phishing site. Nowadays there are many existing fraud websites that offer credit cards or loans to users at a low rate. The attackers can make use of spam indexing, which is a technique that could for the manipulating indexing in search engines so as to enhance the rank of the malicious web page to fool the user. It can be done by inserting some keywords that make the web page seem relevant or legitimate [59]; the steps followed are shown in Fig. 14 (Table 5).

## 7 Taxonomy of phishing defense mechanisms

Phishing email is a kind of spam mail, which is a criminal mechanism relying on fake email claims for a legitimate organization. The basic objective is to steal personal or confidential information from the victims. In this section, we discuss various approaches to filter phishing emails and detect malicious web pages.

### 7.1 Features used for identification of phishing scams

Toolan and Carthy [60] studied the utility of about 40 such features and evaluated their effectives using information gain and entropy. They have categorized common features used for detection of phishing emails as:

1. *Body-based features*: These features are extracted from the email body. They include binary features such as presence of forms, HTML or certain phrases and links in the email body.
2. *Subject-based features*: Some features are extracted from the subject of an email such as whether it is a reply to some previous mail, or the presence of certain words like verify, debit.
3. *URL-based features*: These features check whether an IP address is used instead of domain name, the presence of @ in the links, number of images, external and internal links in the email text, the count of periods in the links, etc.
4. *Script-based features*: These features check for the presence of JavaScript, pop-up window code, onClick events, etc., in the email.
5. *Sender-based features*: These features include the sender's details such as difference between the sender's address and the reply to address.

The most effective features are extracted by analyzing data parts C and D of an email message (as shown in Fig. 15). A common approach in extracting features found in data parts A and B is to use a blacklist, which is not efficient as blacklists perform poorly against zero-day phishing attacks [46]. The groups of most effective features of an email are discussed in Table 6.

Table 6 shows four groups of features: external features (group 1), body-based features (group 2), URL-based features (group 3) and features header (group 4). Phishing emails a traditional and one common way for phishing frauds. Users through Mail User Agent (MUA) transfer any

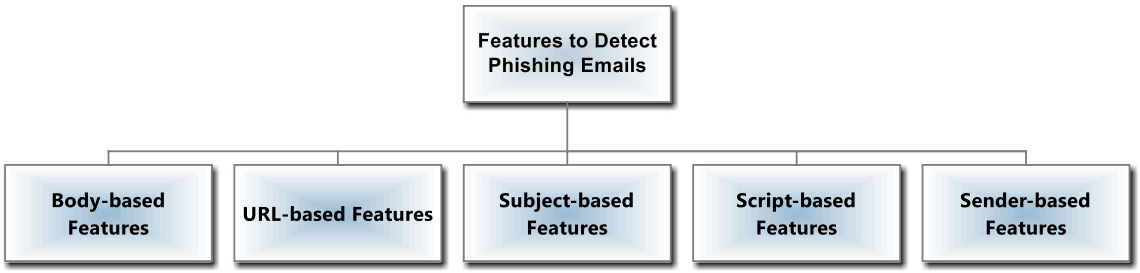


**Fig. 15** Features types for phishing detection

**Table 6** Most effective features of an email

| Group features | No | Features | Abbreviation of features |
|---|---|---|---|
| External features | 1 | Spam features (included 50 subfeatures) | Spamfeatures |
| Body-based features | 2 | HTML email | body_html |
| | 3 | Body of multipart | body_multipart |
| | 4 | Verify your account phrase | body_Verifyphrase |
| | 5 | "OnClick" JavaScript event | body_JSonclick |
| | 6 | Code of JavaScript to change the status bar | body_JSchangebar |
| | 7 | Code of JavaScript | body_javascript |
| | 8 | Code of JavaScript to open popup windows | body_JSpopup |
| URL-based features | 9 | HTML links | url_htmllink |
| | 10 | Number of dots in a link | url_nodots |
| | 11 | Non-matching between target and text of urls | url_ TarDiflink |
| | 12 | URL IP address | url_ip |
| | 13 | Image links | url_imagelink |
| | 14 | URL bag-of-word links | url_bagword |
| | 15 | URL has two domains | url_twodomain |
| | 16 | Non-standard port in the URL | url_nonstport |
| | 17 | URL containing hexadecimal characters or @ symbol | url_hexorat |
| Header-based features | 18 | Subject replay word | sub_replay |
| | 19 | Difference between the sender domain from the domain of the embedded links | Diffsenlindom |
| | 20 | Subject (bank, verify, debit) | sub_words |
| | 21 | Sender email address uses different replay address | Sendiffreplyto |

phishing mail from Mail Transfer Agent (MTA), which transferred email to Mail Delivery Agents (MDA) and then finally received. Figure 8 as used in [61] shows the procedure of phishing email transferred to a computer network. These features are also ranked in [17] based on their information gain based on the overall best, worst and median features as given in Table 7.

## 7.2 Classification of protection against phishing attacks

Classification of different protection mechanisms against phishing attack is shown in Fig. 16.

### *7.2.1 User education*

User education refers to spreading awareness and education about phishing among Internet users. Education-based approaches offer online information about risk of such attacks and their prevention techniques [62]. Some approaches also provide online training and testing to the users.

*7.2.1.1 User's response to phishing attacks* Downs et al. [63] performed a study and concluded that the user should be educated about phishing rather than warning them about the negative consequences of attacks. In this study, 232

**Table 7** Classification of features according to their efficiency

| Best | Median | Worst |
|---|---|---|
| Character/word ratio (body-based feature) | Use of IP address (URL-based feature) | Presence of pop-up window code (script-based feature) |
| Character/word ratio (subject-based feature) | Use of @ symbol (URL-based feature) | Use of the word "debit" (subject-based feature) |
| Presence of HTML (body-based feature) | Use of word "bank" (subject-based feature) | Is the email forwarded (subject-based feature) |
| Total no. of links (URL-based feature) | No. of characters in the subject | No. of internal links (URL-based feature) |
| No. of external Links (URL-based feature) | No. of links in the email body | Presence of scripts in the email (script-based feature) |

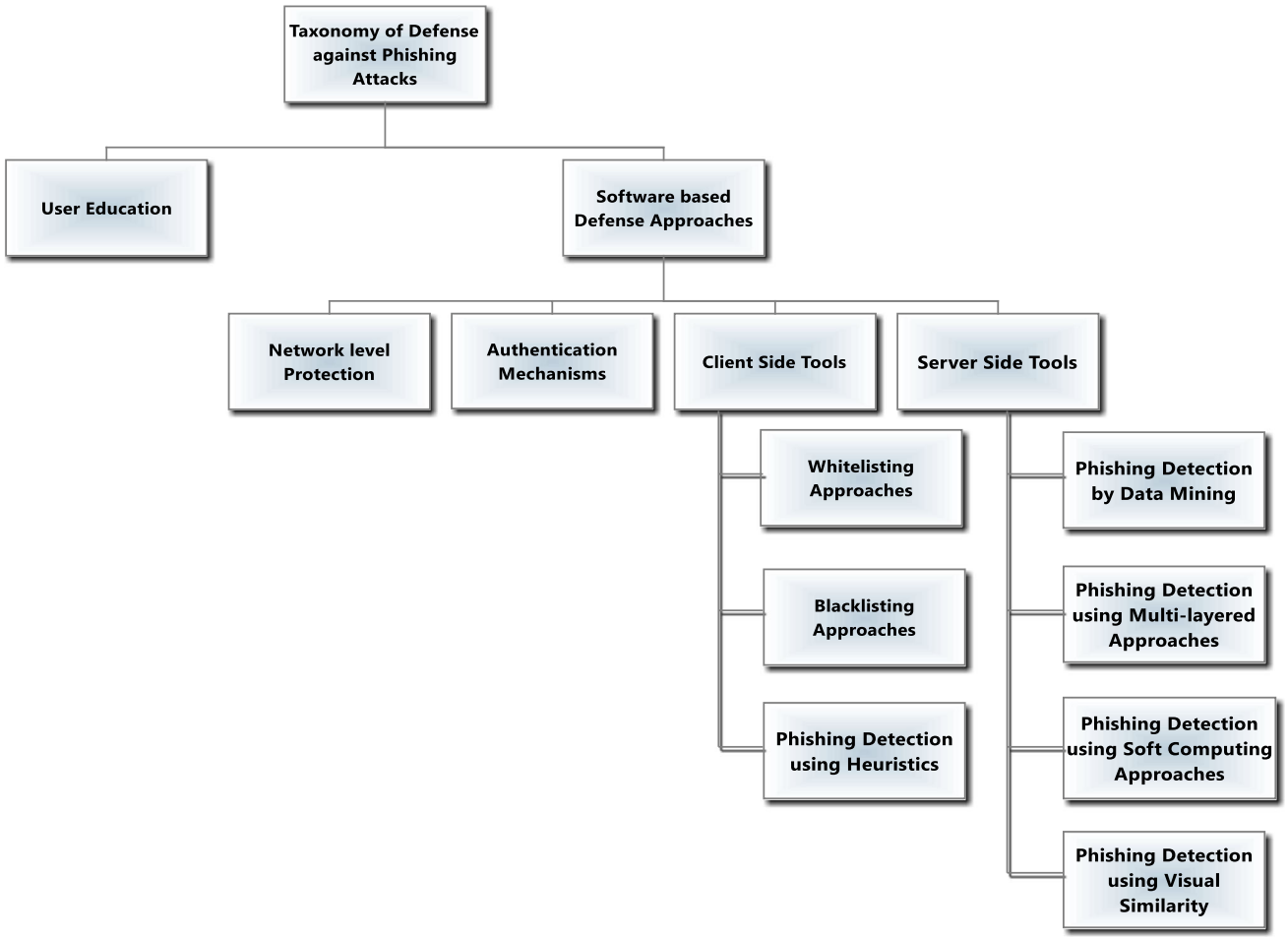


**Fig. 16** Taxonomy of phishing detection approaches [4, 122]

computer users were asked to view some emails and answer few question related to those. This study shows that user having knowledge about URLs and locks is less likely to fall for phishing attacks, whereas understanding of other web tools, e.g., cookies and malicious software, did not reduce the likelihood to fall for phishing attacks. Huang et al. [64] studied that users fall for such attacks because either they are not able to differentiate between a legitimate or phishing website or due to ignorance of the warnings and toolbars indicators.

Shen et al. [65] showed some of the indirect characteristics, i.e., females are more likely to fall victims to phishing attacks than males. The same is true for people between 18 and 25 years of age due to the lack of awareness and technical knowledge. Don et al. [66] proposed a model that describes user interaction with respect to

decision making, which starts as soon as the user views the phishing mail or web page and stops when the user ends its activities. The goal is to detect phishing campaigns by understanding the way users react to phishing web pages or mails.

*7.2.1.2 Prevention of attacks by warning the user* Whenever a user clicks on a malicious link or views a phishing website, web browsers issue security warnings that can be of two types: (1) active warnings, which blocks malicious contents preventing the user from viewing it and (2) passive warnings, which display a pop-up window warning the users while the contents are being viewed. It is shown by studies that active warnings are more effective than passive ones, as user tends to ignore the warnings unless they are blocked from viewing the content [67].

Moore and Clayton [20] stated that service take down is the most common method used by the service providers to handle security problems. Service providers are also required to enforce the rules strictly against illegal use of services of the services provided by them. Egelman et al. [68] showed that passive warnings are ineffective as only about 13 % participants notice passive warnings given by the browser, while active warnings noticed by 79 % of the participants. This justifies failure of the security toolbars as they mostly follow passive warnings. The study conducted by Kumaraguru et al. [69] shows that generated periodic security warnings are ineffective as they can only enhance user's knowledge but cannot force to change their attitude. They also proposed a design of a new method of sending educational notices to users frequently.

Arachchilage and Cole [70, 71] developed a mobile game for Internet users to make them familiar with phishing attacks, and they used Technology Threat Avoidance Theory (TTAT) for the designing of the game. The main objective is to enhance the user knowledge and remove the ignorant behavior among the users. In [72], another TTAT-based model is presented, showing that a well-designed user education program is always helpful in prevention of attacks (Table 8).

*7.2.1.3 Online training* Training the web users to identify malicious emails and websites can be very effective in preventing phishing attacks. Through last few years, various methods have been proposed to train the users online, and training and testing the users using games have also been proved to be an effective method. Kumarguru et al. [69] proposed a design of a new method in which educational notices are sent to users at fixed intervals. They also propose a training method embedded into daily tasks of the user so that he/she is not required to read from any other outside sources. This study showed that user training programs are more successful as compared to educational notices as 89 % of users who train through educational notices fall for phishing attacks, whereas only 30 % of users had training messages included in their daily tasks, were victim of such attacks. But, this approach requires an administrator to handle the messages who should be aware of all the latest aspects of phishing which is a limitation of this approach.

### *7.2.2 Software-based defense approaches*

*7.2.2.1 Protection at network level* In this approach, certain range of IP addresses or a set of domain is not allowed to enter the network. DNSBLs [73] make use of the DNS protocol and are created and updated regularly by observing the network traffic. An open-source software Snort can also be used at the network level although these require continuously updated.

*7.2.2.2 Authentication-based mechanisms* In this approach, it is confirmed whether or not the message was sent by a valid path and domain name and can be employed at both the user and domain level. These techniques enhance the security of email communication. The

**Table 8** Overview of defense against phishing techniques

| Techniques | Approaches | Advantages | Disadvantages |
|---|---|---|---|
| Network level | DNSBL | List is instantaneously updated | Easy for attackers surpass this technique and is required to be manually updated |
| Authentication | TANs, Froogle, Domain level authentication, digital signature for email authentication | Low complexity. Appropriate for social networks | Both sender and receiver must use the same technique. Cannot be secured from MITM attacks |
| Client side | Google safe browsing API, PhishNet, AIWL, SpoofGuard, Phishguard, Phishwish, CANTINA | Heuristic techniques are able to detect zero-hour attacks to certain extent | High FP rates and bandwidth requirements |
| Server side | TF-IDF, SVM, K-NN, DBSCAN, PHONEY, FRALEC, PILFER, robust classifier model | High accuracy | Time-consuming |

authentication schemes are fairly simple and can be done at the domain level or by digitally signing the document before sending. But these require the same technology to be used at both the sender and receivers' sides. Another authentication mechanism called transaction authentication numbers is used by the banks. But, they do not ensure security from man-in-the-middle attacks and are costly in terms of time and computation [74].

*7.2.2.3 Client-side tools* These include user profile filter and browser-based toolbars. Other techniques are domain checks, URL examination, etc. These tools also depend on blacklisting and whitelisting techniques where a list of detected phishing or legitimate websites is downloaded with updates at standard intervals. The limitations of these techniques are their fail to detect zero-day attack.

I. *Phishing detection by blacklists and whitelists* Blacklists contain URLs and IP addresses, which are found to be suspicious and are frequently updated. But they do not provide any protection from zero-day phishing attacks and can only detect only 20 % of these attacks. The conducted studies conclude that 47–83 % of phishing URL are blacklisted after 12 h. This delay is significant as 63 % of phishing attacks end within the first 2 h [75]. Some of the approaches making use of blacklists are: Google safe browsing API, DNS-based blacklists, predictive blacklisting and automated individual whitelist.

(1) *Google safe browsing API* Google provides a service for safe browsing that allows the applications to verify the URLs using a list of suspicious pages which are regularly updated by Google. It is an experimental API and is used by Google Chrome and Mozilla Firefox. The Safe Browsing service provides two experimental APIs:

(a) *Safe browsing lookup API* The Safe Browsing Lookup API [74, 76] allows the clients to send suspicious URLs to Safe Browsing service which tells whether the URL is legitimate or malicious. The client API sends the URLs with GET or POST request, which are checked using the malware and phishing lists provided by Google with the current version being used is 3.1. Some of the shortcomings of Safe Browsing Lookup API are: (i) no hashing is performed before sending URLs and (ii) there is no limit on the response time by the lookup server.

(b) *Safe browsing API v3* Using the Safe Browsing API [77], the client can download a table of URLs for client-side lookups. The Safe Browsing API version v3 was introduced in 2014, and afterward the Safe Browsing API version v2 was deprecated. Phishing and malware URLs are published in two different blacklists which are goog-pub-phish-shavar and goog-malware-shavar, both of which have SHA-256 hash values ending with a 4-byte hash prefix.

(2) *DNS-based blacklist (DNSBL)* A DNSBL is a zone that contains resource records for the identification of hosts present in the blacklist and uses DNS protocol. Hosts undergo an IP address or domain name transformation to be encoded into DNSBL zones. There must be an A record and TXT record, which gives the reason for blacklisting for each entry in the DNSBL [75]. The standard value of A record contents is 127.0.0.2, but they may have other values too. DNSBLs can use the same TXT records for all entries or a different for each entry.

(a) *IPv6 DNSBLs* The structure of DNSBLs using IPv6 addresses is defined as a domain in [78]. The entry names are IPv6 address with DNSBL domain as their suffix. The A and TXT records are used similar to that of IPv4 DNSBLs. A single DNSBL can have IPv4 and IPv6 addresses. The representation of IPv6 lists is similar to IPv4, where the 4 octet address is replaced by 32 nibbles of IPv6 address.

(b) *Domain name DNSBLs* Domain names are less frequently used by DNSBLs than the IP addresses. The interpretation of records and TXT is the same as that of the IPv4 DNSBLs. The system manager must be cautious while choosing DNSBLs; the management policies of the server management and the system manager should be consistent. If it is not the case, then the addresses from which the system is expecting mail might get blocked (Fig. 17).

(3) *PhishNet: Predictive blacklisting* The attacker applies some simple changes to the URL; PhishNet [79] detects these changes using two components:

(a) Malicious URL prediction: PhishNet examines the blacklisted URLs and uses some heuristics to create new variations of that URL, which are as follows:

- Replacing top-level domains (TLDs) with 3209 different TLDs resulting into child URLs which ought to be examined.
- To generate new URLs clusters of host equivalence classes having the same IP address are maintained, all combinations of these hostnames and paths are then used to create new URLs.
- The URLs with the same directory are grouped together, and new URLs are created by exchanging filenames within that group.
- If two URLs have the same directory structure with different query part, the query part can be swapped to create new URLs.
- New URLs are created by substituting the brand names in the phishing URLs.

Once the children URL are generated, they are subjected to a validation process which eliminates legitimate URLs.

(b) *URL matching* Incoming URL are matched with the blacklisted URLs using regular expressions and hash maps.

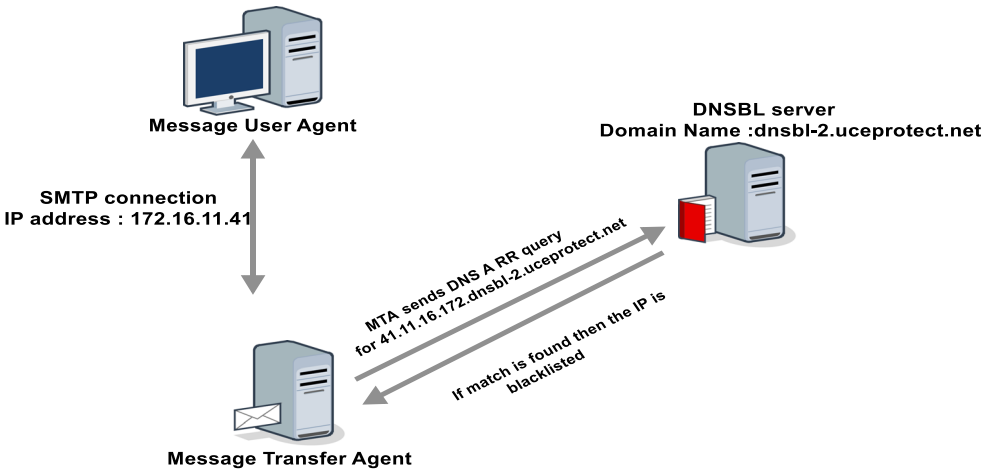


**Fig. 17** DNS-based blacklisting mechanism

(4) *Automated individual whitelist* AIWL [80] keep records of legitimate login user interfaces (LUIs) of web pages. Whenever a user submits his/her credentials to LUI, the whitelist is checked for it and if it is not on the list, a warning is given to the user.

AIWL has two primary components:

- *Whitelist*: It contains a list of legitimate LUIs and is used to check whether a URL is familiar or suspicious and so that warnings are suppressed. In the whitelist, each LUI is stored as a vector comprises of URL address, page feature, DNS-IP mapping, etc.
- *Automated whitelist maintainer*: It is a classifier, i.e., naive Bayes, which decides whether to store LUI in the whitelist. The whitelist maintainer checks the number of logins for a specific LUI; if it exceeds a certain threshold, then that LUI is whitelisted.

*II. Phishing detection toolbars and plug-ins* Phishing heuristics are characteristics that are found in phishing attack; however, the characteristics are not guaranteed to exist in each case. If a set of general heuristics can be identified, it is possible to detect zero-hour phishing attacks. Some of the methods to detect these heuristics are: SpoofGuard, PhishGuard, Phishwish, CANTINA, etc.

(1) *PhishGuard: A browser plug-in* PhishGuard [81] is a browser plug-in non-tunneled phishing attacks that do not involve legitimate sites being used for tunneling the output to the victim and is achieved by testing HTTP Digest authentications. The plug-in can possibly incorporate other authentication mechanism too. PhishGuard triggers into action when it detects the start of an authentication process where a user will submit a user ID and a password (or some equivalent data). PhishGuard would forward the real user ID to the page but some (random) incorrect password instead of the real one repeatedly, a certain number of times. If the page replies negatively, i.e., if the HTTP response code is 401, then there is a good chance it is a legitimate site. On the other hand, if the page replies positively, i.e., if the HTTP response code is 200, then the site can be considered as a phishing site. The final result is reached on the basis of password hashes maintained by the tool, which declares the site a phishing site if it already has the hash of the entered password, otherwise the user is prompted to reenter the password.

(2) *Phishwish* Phishwish [82] is a mechanism consisting of 11 rules to detect phishing message or email. The idea is to provide better protection against zero-hour attacks than blacklists with minimal false positives. It requires lesser resources (11 rules) as compared to SpamAssassin which uses 795 rules. Phishwish analyzes the email header and URL that is contained in the email's body.

The message is referred to as phishing in the following cases (rules):

1. If the URL present in the message redirects the user to a login page which is not authentic (some organizations' original login page) and checked by the help of a search.
2. If URL uses Transport Layer Security (TLS) in a HTML formatted email, but not in the actual HREF attribute.
3. If the URL has an IP address instead of a domain name for the host.
4. If the name of the organization (e.g., eBay) is given in the URL, but not in the domain name.
5. If domain name in the URL does not match the domain name in the HREF attribute.
6. If the organization's domain name is not present in the received SMTP header.
7. If there are inconsistencies in URL's domain portion, the result is positive.
8. If there are inconsistencies in the image link's domain part.

9. If there are inconsistencies in the WHOIS records of non-image URL's domain part.
10. If there are inconsistencies in the WHOIS information of image link's domain portion.
11. If the page cannot be accessed.

The email score is calculated by the weighted mean of these rules. If the score of a given email is greater than 50 %, then it is considered as phishing, otherwise as legitimate.

(3) *CANTINA* CANTINA [83] is a content-based approach that decides whether a visited page is legitimate or phishing. CANTINA deploys term frequency–inverse document frequency (TF-IDF), rule-based heuristics and search engine output to reduce false positives. CANTINA when making use of TF-IDF along with some simple heuristics is able to detect about 90 % of the phishing websites with 1 % false positive rate. But it suffers from performance issues due to the delay in querying from search engine.

(4) *Other browser-based toolbars* Some other browser-based toolbars are: SpoofGuard [84], which a browser plug-in developed at the Stanford University. It detects phishing attacks based on HTTP(S), by taking certain irregularities found in the HTML content into account against a previously defined threshold. Some other browser-based toolbars are NetCraft [85], CloudMark [86], IE phishing Filter [87], eBay toolbar [88] used at the client side. These toolbars are developed and trained using the URLs of phishing web pages, and they give warning to the user when encounter a suspicious pages are encountered.

*7.2.2.4 Server-side filters and classifiers* These are based on content filtering approaches and are appropriate to fight zero-day attacks. These filters are based on machine learning techniques and are categorized as:

(I) *Phishing detection based on machine learning* In this method, input data are considered to be an unordered set of words and are based on machine learning classifiers such as naive Bayes classifiers, support vector machines (SVM), k-nearest neighbors, Boosting and TF-DIF. SVM is the most popular method of all, and Chandrashekharan et al. [89] used one-class SVM to train some email samples in an already defined plane and used features to map the samples into a new transformed space. The two classes of the emails, e.g., ham and phishing, are separated by a hyper plane. TF-IDF [90] uses document frequency of a word, i.e., in how many documents the words occur. K-nearest neighbors algorithm uses a similarity function for training the datasets; then, the mails are labeled to one of the created previously cluster [91]. Naïve Bayes is also commonly used for the purpose of text classification [92]. It uses the Bayes theorem for classification, and the features should be statistically independent to get good results.

(II) *Phishing detection by data mining* The techniques that come under this category consider phishing to be a classification or clustering problem, and algorithms such as machine learning, k-means clustering or SVM are applied to them.

Gang Liu et al. [93] proposed an approach to detect phishing websites by finding websites similar to it and comparing it with them. If the websites are similar to the suspicious site but having different domain name are found, then the website is said to be phishing. They extracted features from URL and keywords, i.e., similarity in the text, layout, number of links in the web page, etc. DBSCAN algorithm is used to detect similarity by comparing the suspected website with all the websites, and the website with highest similarity is said to be the target website. For the evaluation of their approach, the authors used 8745 websites from PhishTank and 1000 legitimate websites were collected from Random yahoo link.

Bazarganigilani [94] used ontology concept for the classification of text of phishing emails and considers every word to be an attribute and its number of occurrences to be the value (which is referred to as the ontology concept) for the classification purpose. The term frequency variance, information gain and adaptive Naïve Bayes technique are used that gives 94.87 % accuracy.

Chandrashekharan et al. [95] used structural features for detection of phishing emails. From each email, features, such as a ratio of total number of words to total number of characters and frequency of certain words, are extracted. This approach made use of simulated annealing for feature selection and SVM as classifier with 95 % of accuracy. But the approach was tested only on a small dataset.

Kim et al. [96] devised an algorithm that detects DNS-based poisoning attacks on the basis of network level features. The data they used for processing were 10,000 routing information instances destined toward phishing and legitimate servers. Each instance consists of mean round-trip time, hop count between the user and the service accessed and whether the service was behind a firewall. Their study showed that only 19 % of phishing websites are behind firewalls, while 79 % of the legitimate websites were behind firewall. Thus, the phishing websites are hosted in less secure hosts than the legitimate target sites. The routing information has been processed by algorithm like SVM and k-NN.

*PHONEY* [97] detects links and HTML forms in the incoming email. Then, the control transfers to the content scanner that analyses and takes the data from the web page. The data thus obtained are compared with the DB entries, e.g., usernames and passwords. As the technique was tested on a very small amount of data, it cannot be determined whether it can address real-time phishing scams. Haijun Zhang et al. [98] proposed techniques that used the content

information to give a class label to a suspected website. They proposed a number of techniques but the most effective of them made use of naïve Bayesian classifier and image processing to compare both the textual and visual characteristics of the sites. The naïve Bayes classifier gives a normalized number, which specifies similarity between the text of suspected and legitimate websites. The image processing technique measures the similarity between the appearances of both websites. The outputs of both the approaches are then examined and normalized for a selected interval and have the highest probability of belonging to that interval decides the label of the suspected website.

Ma et al. [99] proposed a model to detect phishing emails using hybrid features. It has five stages:

- A feature generator to extract seven features out of the email.
- For feature selection, an adaptive machine learning algorithm is used which is a combination of five algorithms.
- Calculation of information gain.
- A small vector of features for evaluation.
- In the last stage, a matrix of features is created for optimization of features. The decision tree algorithm gives the best results for the short feature vectors.

*PILFER* [23] technique is used to detect phishing emails. Here, the emails are represented using 10 features, and spam filter output is also considered to be a feature. For a classification, tenfold cross-validation with random forest is used. For training and testing, the dataset SVM is used. Since the phishing sites are short-lived, many of the features cannot be extracted from old emails, but still PILFER can classify emails with 99.5 % accuracy. It is more accurate than spam filter alone with the *false positive* rate of 0.0013(approximately) and *false negative* rate of 0.035(approximately), while without spam filter the output PILFER has comparable accuracy by spam filter.

*Cluster phishing emails automatically* uses features such as document size, message content and HTML features. For extracting these features, an adaptive k-means clustering is used. An objective function is produced with final value determined by the optimal cluster [100].

Beuskova et al. [101] proposed an approach that combines both supervised and unsupervised learning classification approaches. In order to randomize the input data, independent unsupervised clustering is used. The next step is to build a consensus clustering which is a combination of trained clusters on which supervised clustering is applied for the classification of the already clustered data.

In [102], a country-based model for phishing detection has been proposed; the objective was to develop an anti-phishing framework in accordance with a particular country's Internet infrastructure (Saudi Arabia in this case). The provided exposure of their model prototype to the victims within the country and for deployment, their aim was to detect phishing web pages instead of blocking them.

In [103], two features were used for the determination of web pages' identity solely; they do not need any other services. They proposed a model called PhishDetector, which is rule-based for obtaining hidden knowledge, and it is also able to detect zero-day phishing attacks.

Some other proposed techniques made use of multidimensional feature vectors and used information gain for feature selection, and any classification model can be used. Other techniques are Bayesian anti-phishing toolbar or the use of natural language processing for intrusion detection.

(III) *Phishing detection based on soft computing techniques* In this approach, knowledge discovery is used to simplify the evolution process, which can be a group of networks that are executing continuously and changing their architecture and functions in parallel and are also consistent with the environments and systems related to them. There are no dimensions fixed, and the system grows in free space learning continuously as an individual and a part of a system [104]. Evolving clustering method for classification is used in [105] to develop a model for phishing detection which performs the classification using some features. Other approaches by Almomani et al. [106, 107] are based on fuzzy neural networks to classify phishing and legitimate emails.

(IV) *Multilayered phishing detection system* This approach uses different classifier algorithms to improve the results. Some of them are:

An approach by Castillo et al. [108] referred to as FRALEC classifies an email into ham or phishing classes. It makes use of three filters: (1) using Naïve Bayes classifier which scrutinizes the emails' text content. (2) Classifies an email into fake, legitimate or suspicious category using non-grammatical features using rule-based classifier. (3) Emulator-based filter to re-classify the suspicious emails. This technique gives 99.8 % accurate results.

Multitier classification [109] makes use of three classifiers in a layered fashion which extracts and classifies features in a sequential manner, and the output thus obtained is sent to the decision classifier. In case of any misclassification by the upper two layers, the last classifier will make the final decision. The approach gave best results of 97 % accuracy when the sequence of classifiers used from top to bottom was: SVM, AdaBoost and Naïve Bayes, respectively.

Profiling of phishing email [110] is done by obtaining structural features to get the links embedded in emails and then to represent the emails in the form of features' WHOIS information. To get multilabel class predictions, SVM followed by a boosting algorithm (Table 9).

**Table 9** Phishing detection techniques

| Technique | Description | Advantages | Drawbacks |
|---|---|---|---|
| Google API [77, 78] | Multiple versions of a URL are produced using heuristics | Only the data portions are to be considered | Very high FP (5 %) and FN (3 %) rates, high bandwidth requirements |
| DNSBL [74] | Provides network-level protection and uses DNS protocol | Created and updated regularly | If policies of server and system are not consistent address from which system may get mail might get blocked |
| AIWL [80] | Creates a whitelist on the basis of heuristics and visual Similarity | Accurate results | Dependent on how user trains the browser |
| PhishGuard [82] | Uses a set of heuristics for phishing detection | Effective in the detection of zero-hour attacks | Updation of rule is required |
| PhishWish [83] | Uses 11 rules for phishing detection (heuristics) | Provides better results than blacklists in zero-hour attacks | Fails in the case of changing phishing patterns |
| CANTINA [84] | Content-based approach; uses TF-IDF, rule-based heuristics and Search Engine output to reduce FPs | Detect 90 % of phishing websites correctly with 1 % of FP rate | Delay in queuing from search engine |
| Gang Liu et al. [89] | Detect phishing websites by comparing them with other websites using DBSCAN clustering algorithm | Detect up to 91.4 % of websites correctly | 3.4 % False alarm rate. Network connection is also required for search engine querying which can be time- and cost consuming |
| Structural feature-based detection [98] | Simulated annealing is used for feature selection and SVM for classification | Minimal performance overhead | Dataset used is small in size |
| Ontology concept [100] | Uses heuristics for phishing email detection | Accuracy reached up to 94.8 % | Dataset size is small; only ontology concepts of features are considered which lowers the accuracy |
| PHONEY [101] | Copies user actions | | Tested only on a set 20 emails and thus real-time output is not known |
| PILFER [104] | Uses 10 features for selection of phishing email and SVM is used for classification | Can also be used website detection (excluding the SA output) | 0.12 and 7.4 % of FN show that significant number emails were mis-classified |
| Beuskova et al. [108] | Expert system with a set of dynamic rules uses Artificial Intelligence | Suitable for real-time applications | Time- and cost consuming |
| FRALEC [112] | Uses three classifiers: Naïve Bayes, rule-based and emulator-based | Gives 99.8 % accuracy | Consumes time due layered design |
| Ma et al. [105] | Hybrid detection model consists of five stages and decision tree for classification | Accuracy of 96 % | Non-standard dataset used |
| Profiling of phishing emails [114] | Twelve features are used to represent phishing emails. SVM and boosting is used for classification | Detection accuracy is good using hyperlink features | Only hyperlink features are considered |
| Fu et al. [115] | Detection of web pages using earth mover's distance to calculate visual similarity | Computation complexity is low | Only considers pixel features thus text-based similarity cannot be detected |
| Medvet et al. [103] | Visual similarity-based detection method using text- and image-based features | No false positives were raised | Only 41 real-world images are used for testing |
| Hara et al. [72] | Visual similarity mechanism compares websites which completely mimic victim pages | | Very high false positive (18 %) rate |

(V) *Phishing detection by visual similarity* Phishing Detection through this method refers to identifying phishing web pages by checking their resemblance with legitimate web pages [106, 111]. As we are aware of the fact that most of the phishing websites are almost same as that of their target websites, these techniques make use of the view of the web page rather than the code behind it. One such approach proposed by Fu et al. [112] used Internet Explorer to collect websites whose snapshots are then converted to $100 \times 100$ image; a feature vector is formed using that image which is then normalized to a number from 0 to 1. Whenever a comparison between two images is performed

if the images are different it retuned normalized number is 0, and if it is 1, then they are same. If the value is between 0 and 1, then threshold values are used to categorize the web page.

The authors in [113] proposed a visual similarity-based strategy in order to identify phishing websites. The first step checks emails at the mail server for suspicious words and phrases and URLs. The second step monitors the suspicious web pages by comparing them to legitimate pages and measure their similarity with respect to layout, page style, etc. Medvet et al. [114] compared suspected web pages to legitimate pages using three features which can be text or style related that make both the pages appear to be similar. For evaluation, they used real phishing web pages along with their targets. Hara et al. [115] proposed a technique based on visual similarity. A collection of legitimate websites are used to train the classifier and stored in a database. Whenever a suspected website is found, its snapshot is compared to websites in the database, and threshold values are used for the similarity between the websites so as to get a label.

## 8 Open issues and challenges

Various solutions to control phishing attacks have been given in the literature. However, we can say that no solution is a "bullet of silver" against phishing. With time, phishing threat is increasing and becoming a common fraud to commit e-crime. Every time, when researchers come with any idea to control this problem, phishers change their attack strategy by exploiting vulnerabilities found in the current solution. Therefore, we can say that it is a very tight race between phishers and researchers. Phishing frauds could be committed either by social engineering or by using malicious codes. In social engineering scheme, phisher used either spoofed emails or fake websites to fool the users and commit fraud. Therefore, solutions are also based on these observations.

The blacklisting and whitelisting approaches have low FP rates and are very inefficient for the detection of zero-hour phishing attacks, i.e., these approaches are able to detect only about 20 % of such attacks. They also require communication over the network, which lowers the performance. PhishNet [79] requires high bandwidth so as to increase the blacklist. The Google safe browsing API [77] aims to lower the bandwidth requirements. In case of AIWL [80], the efficiency totally depends on how the user trains his/her browser. The machine learning and data mining approaches give the best results in phishing detection. Chandrashekharan et al. [85] used structural features with SVM to detect phishing attacks with 95 % accuracy. However, this approach is very time-consuming, even for a small dataset. The accuracy of the system using SVM can be increased up to 97 %. PILFER [104] also gives about 95 % accuracy. But the FP and FN rate show that considerable number of emails is not well classified. Similarly, robust classifier model [105] is 99.8 % accurate. But, it is a time-consuming process as it requires due to its five stages and used datasets are not standard.

Phishing detection by heuristics also gave good results. But some of them have very high FP rates, e.g., SpoofGuard [81] and PhishWish [83]. In Phishwish, since there are 11 rules to be followed, it is not adaptive to changes in the scenario. CANTINA [84] also has high FP rate in addition to its time-consuming processing. Another challenge with these approaches is the frequent update time which makes it quite expensive. User awareness is an important issue, for defense against phishing attacks. Along with an increase in the user education, some other remedies could be enhancement in the user interfaces, i.e., giving active warnings and automatically detecting malicious messages.

Recently, one of the newest areas, i.e., IoT, has also become a victim of phishing attacks. IoT is a very fast evolving architecture these days connecting every day-to-day object making our lives more comfortable. But, due to limited resources available to the IoT devices, their security mechanism is not very strong which makes them a very easy target for the attackers [116–118]. In January 2014, Proofpoint unleashed the first spam and phishing attacks on IoT devices such as refrigerators and smart TVs; the attackers used these devices as a medium to send about 100,000 emails containing malwares. Once infected, the IoT devices are required to be bought offline to remove malware and those which were not are still infected. In the year 2013, 20 billion devices were connected to Internet, and this number will increase to 32 billion by the year 2020. Smart thing are the future, and everyone is appreciating it but these devices are also making the job of attackers easy [119–121].

## 9 Conclusion

It has been almost 20 years since the phishing problem was identified. But, still it is used to steal personal information, online credentials and credit card details. There are various solutions available, but whenever a solution is proposed to overcome these attacks, phishers come up with the vulnerabilities of that solution to continue with such an attack. Phishing attacks can be classified broadly into two categories: Social engineering, which refers to acquiring user's credentials using emails or fake websites, and malware attacks, which use malicious code or software to acquire the data required. There are several approaches to defend

the user from email and website phishing and were discussed in this document.

Our survey helps new researchers to understand the history, current trends of attacks and failure of various available solutions. Defense against phishing attacks is one of the hardest challenges faced by the network security these days. A good defense mechanism should be able to detect phishing attacks with low false positives. The defense techniques discussed in this survey are blacklisting, data mining and heuristics, machine learning and soft computing algorithms. Blacklisting techniques have minimal FP rates but consume a lot of bandwidth and should be avoided if there is a possibility of zero-hour attacks. The heuristic and data mining techniques have high FP rates than blacklists with high computational costs but better at detecting zero-hour attacks. The machine learning techniques give the best results as compared to other techniques as they are able to mitigate zero-hour phishing attacks better than the other. Some of the machine learning techniques [104, 105] are able to detect TP up to 99 %.

We know that lack of awareness among the users is also a factor that relates to success of phishing attacks. Thus, educating the user is also a requirement to lower the phishing attacks, besides improvements in the interfaces that give warnings or the automatic removal of malicious content before the end-users would be a more promising approach. After the classification, we also described various issues and challenges in current solutions to understand new researcher about the idea for future study by defending against phishing attacks.

## References


1. The Phishing Guide Understanding & Preventing Phishing Attacks By: Gunter Ollmann, Director of Security Strategy, IBM Internet Security Systems, 2007
2. Phishing: Cutting the Identity Theft Line Published by Wiley Publishing, Inc. 10475 Crosspoint Boulevard Indianapolis, IN 46256 www.wiley.com, 2005, Rachael Lininger and Russell Dean Vines
3. Anti-Phishing Working Group (APWG), "Phishing activity trends report—first quarter 2013. http://antiphishing.org/reports/apwgtrendsreportq12013.pdf, accessed September 2014
4. Aloul F (2010) The need for effective information security awareness. Int J Intell Comput Res 1(3):176–183
5. James L (2005) Phishing exposed. Syngress Publishing, Burlington
6. Anti-Phishing Working Group (APWG) (2014) Phishing activity trends report—first quarter 2014. http://antiphishing.org/reports/apwgtrendsreportq12014.pdf. Accessed Sept 2014
7. Anti-Phishing Working Group (APWG) (2014) Phishing activity trends report—fourth quarter 2013. http://antiphishing.org/reports/apwgtrendsreportq42013.pdf. Accessed Sept 2014
8. Anti-Phishing Working Group (APWG) (2014) Phishing activity trends report—second quarter 2013. http://antiphishing.org/reports/apwgtrendsreportq22013.pdf. Accessed Sept 2014
9. Anti-Phishing Working Group (APWG) (2014) Global Phishing Survey—second half 2013. http://antiphishing.org/reports/apwgglobalphishingreport2h2013.pdf. Accessed Sept 2014
10. IT Business Edge (2014) Spear phishing, targeted attacks and data breach trends. http://www.itbusinessedge.com/slideshows/spear-phishing-targeted-attacks-and-data-breach-trends-04.html. Accessed on Sept 2014
11. Pierluigi Paganini (2014) Phishing: a very dangerous cyber threat. http://resources.infosecinstitute.com/phishing-dangerous-cyber-threat/2012. Accessed on Sept 2014
12. Krebs B (2014) HBGary federal hacked by anonymous. http://krebsonsecurity.com/2011/02/hbgary-federal-hacked-by-anonymous/2011. Accessed Sept 2014
13. eCrime Trends Report: Fourth Quarter (2013) http://Internetidentity.com/resource-tags/quarterly-ecrime-reports/. Accessed Sept 2014
14. Anti-Phishing Working Group (APWG) (2016) Phishing activity trends report—first-third quarter 2015. http://antiphishing.org/reports/apwgtrendsreportq12013.pdf. Accessed Feb 2016
15. Husna H, Phithakkitnukoon S, Palla S, Dantu R (2008) Behavior analysis of spam botnets. In: Communication systems software and middleware and workshops, 2008. COMSWARE 2008. 3rd International Conference, Bangalore, India, 2008, pp 246–253
16. Toolan F, Carthy J (2009) Phishing detection using classifier ensembles. In: eCrime researchers summit, IEEE conference Tacoma, WA, USA, 2009, pp 1–9
17. Toolan F, Carthy J (2010) Feature selection for spam and phishing detection. E-Crime Researchers Summit, Dallas, pp 1–12
18. Anti-Phishing Working Group Phishing Archive (2014) http://anti-phishing.org/phishing_archive.htm. Accessed Sept 2014
19. Dhamija R, Tygar JD (2005) The battle against phishing: dynamic security skins. Proceedings of symposium usable privacy and security
20. Aburrous M, Hossain MA, Dahal K, Thabtah F (2010) Predicting phishing websites using classification mining techniques with experimental case studies. In: Seventh international conference on information technology. IEEE Conference, Las Vegas, Nevada, USA, 2010, pp 176–181
21. PhishTank Phishing Archive (2014) http://www.phishtank.com/phisharchive.php. Accessed Sept 2014
22. Apache Software Foundation (2014) Spamassassin public corpus, 2006. http://spamassassin.apache.org/publiccorpus/. Accessed Sept 2014
23. Fette I, Sadeh N, Tomasic A (2007) Learning to detect phishing emails, In: Proceedings of 16th international world wide web conference (WWW 2007). ACM Press, Banff, Alberta, Canada, pp 649–656
24. Khonji M, Iraqi Y (2011) Lexical URL analysis for discriminating phishing and legitimate email. 6th IEEE international conference on internet technology and secure transaction, London, UK, pp 422–427
25. Cohen WW (2014) Enron email dataset. https://www.cs.cmu.edu/~./enron/. Accessed Sept 2014
26. "The Enron Spam Datasets" (2014) AEUB natural language processing group, Athens, Greece. http://www.aueb.gr/users/ion/data/enron-spam/. Accessed Sept 2014
27. Klimt B, Yang Y (2004) The enron corpus: a new dataset for email classification research. In: Proceedings of 15th European conference on machine learning, Nancy, France, 2004, pp 217–226
28. Georgala K, Kosmopoulous A, Paliouras G (2014) Spam filtering: an active learning approach using incremental clustering. In: Proceedings of ACM 4th international conference on web intelligence, mining and semantics, article no. 23, Greece, ACM
29. Cormack GV, Lynam TR (2005) TREC 2005 spam track overview. In: TREC

30. IronPort Anti-Spam (2014) http://www.ironport.com/technology/ironport/antispam.html. Accessed Sept 2014
31. Moore T, Clayton R, Stern H (2009) Temporal correlations between spam and phishing websites. In: Proceedings of 2nd USENIX LEET, Boston
32. SpamCopWiki: SpamTrap (2014) 21 July 2006. http://forum.spamcop.net/scwik/SpamTrap/. Accessed Sept 2014
33. The Phishload Phishing Test Database. http://www.medien.ifi.lmu.de/team/max.maurer/files/phishload/
34. Jakobsson M, Myers S (2007) Phishing & countermeasures: understanding the increasing problem of electronic identity theft. Wiley, New York
35. Sheng S, Magnien B, Kumaraguru P, Acquisti A, Cranor LF, Hong J, Nunge E (2007) Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish. In: Proceedings of the SOUPS, Pittsburg, pp 88–99
36. Markus Jakobsson SM (2007) Phishing and countermeasures, Microsoft's anti-phishing technologies and tactics. 18 MAY 2007, pp 551562
37. Project H, Alliance R (2005) Know your enemy: tracking botnets. http://www.honeynet.org/papers/bots/. Accessed Sept 2014
38. Moore T, Clayton R (2007) Examining the impact of website take-down on phishing. In: eCrime'07: proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit. ACM, New York, NY, USA, pp 1–13
39. Chhabra M, Gupta BB (2013) A novel solution to handle DDOS attack in MANET. J Inf Secur 4(3):165–179
40. Gupta BB, Joshi RC, Misra M (2009) Defending against distributed denial of service attacks: issues and challenges. Inf Secur J A Global Perspect 18(5):224–247
41. NPM (2014) Fpipe. https://www.npmjs.org/package/fpipe. Accessed Sept 2014
42. Jagatic T, Johnson N, Jakobsson M, Menczer F (2007) Social phishing. Commun ACM 50(10):94–100
43. Granger S (2001) Social engineering fundamentals, part I: hacker tactics. vol 2006: SecurityFocus
44. Tom NAJ, Jagatic N (2007) Markus Jakobsson, FilippoMenczer, "Social phishing". Commun ACM 50:94–100
45. Spear Phishing Attacks—Why They are Successful and How to Stop Them. Combating the Attack of Choice for Cyber criminals, Fire Eye Inc (Whitepaper)
46. The Internet Protocol Journal, June 2000, vol 3, no 2. http://www.cisco.com/web/about/ac123/ac147/ac174/ac196/about_cisco_ipj_archive_article09186a00800c8901.pdf
47. Spear Phishing Email: Most favored APT attack bait (2012) Trend micro incorporated research paper 2012
48. Adhikary N, Shrivastava R, Kumar A, Verma SK, Bag M, Singh V (2012) Battering keyloggers and screen recording software by fabricating passwords. I. J. Computer Network and Information Security, June 2012
49. CPNI (2013) Spear phishing: understanding the threat. Sept 2013
50. Sullivan D (2005) The definitive guide to controlling malware, spyware, phishing and spam. Realtime Publishers
51. Emigh A (2006) The crimeware landscape: malware, phishing, identity theft and beyond. J Digit Forensic Pract 1(3):245–260
52. Sagiroglu S, Canbek G (2009) Keyloggers. IEEE technology and society magazine, pp 10–17
53. Kapoor S (2014) Session hijacking exploiting TCP, UDP and HTTP sessions. http://www.bindview.com/Services/Razor/Papers/2001/tcpseq.cfm. Accessed Sept 2014
54. Gill R, Smith J, Clark A (2006) Experiences in passively detecting session hijacking attacks in IEEE 802.11 networks. In: ACSW frontiers '06: proceedings of the 2006 Australian workshops on grid computing. Darlinghurst, Australia, 2006. Australian Computer Society, Inc, pp 221–230
55. Christin N, Weigend AS, Chuang J (2005) Content availability, pollution and poisoning in file sharing peer-to-peer networks. In: EC '05: proceedings of the 6th ACM conference on electronic commerce. ACM Press, New York, NY, USA, pp 68–77
56. Perdisci R, Antonakakis M, Luo X, Lee W (2009) "WSEC DNS: protecting recursive DNS resolvers from poisoning attacks", in DSN. IEEE, Lisbon, pp 3–12
57. Azad HS, Zomaya AY (2014) Large scale network centric distributed systems. Wiley, New York
58. Yang LT, Rana OF, Martino BD, Dongarra J (2006) High performance computing and computing. First international conference, HPCC, Springer, Munich, Germany, Sept 2006
59. Moore T, Clayton R (2008) Evil Searching: compromise and recompromise of internet hosts for phishing
60. Dhamija R, Tygar JD, Hearst MA (2006) Why phishing works," in proceedings of the 2006 conference on human factors in computing systems (CHI). ACM, Montréal, Québec, Canada, pp 581–590
61. ALmomani A, Gupta BB, Wan T, Altaher A, Manickam S (2013) Phishing dynamic evolving neural fuzzy framework for online detection zero-day phishing email. Indian J Sci Technol 6(1):3960–3964
62. Chou N, Ledesma R, Teraguchi Y, Mitchell JC (2004) Client-side defense against web-based identity theft. In: NDSS. The Internet Society
63. Downs JS, Holbrook M, Cranor LF (2007) Behavioral response to phishing risk. Presented at the proceedings of anti-phishing working groups 2nd annual eCrime researchers summit. ACM Conf, Pittsburgh, Pennsylvania, pp 37–44
64. Huang H, Tan J, Liu L (2009) Countermeasure techniques for deceptive phishing attack. In: International conference on new trends in information and service science, 2009. NISS'09, Korea, pp 636–641
65. Sheng S, Holbrook M, Kumaraguru P, Cranor LF, Downs J (2010) Who falls for phish? A demographic analysis of phishing susceptibility and effectiveness of interventions. In the proceedings of 28th ACM international conference on human factors in computing systems (CHI'10), New York, NY, USA, pp 373–382
66. Dong X, Clark J, Jacob J (2008) Modelling user-phishing interaction. In: Human system interactions, 2008 conference, Austria, May 2008, pp 627–632
67. Wu M, Miller RC, Garfinkel SL (2006) Do security toolbars actually prevent phishing attacks? In: Proceedings of the SIGCHI conference on human factors in computing systems, ser. CHI'06, New York, NY, USA, pp 601–610
68. Egelman S, Cranor LF, Hong J (2008) You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In: Proceeding of the twenty-sixth annual SIGCHI conference on human factors in computing systems, ser. CHI'08. ACM, New York, NY, USA, pp 1065–1074
69. Kumaraguru P, Rhee Y, Acquisti A, Cranor LF, Hong J, Nunge E (2007) Protecting people from phishing: the design and evaluation of an embedded training email system. In: Proceedings of CHI, ACM Conf, California, USA, pp 905–914
70. Arachchilage NAG, Love S (2013) A game design framework for avoiding phishing attacks. Comput Hum Behav 29(3):706–714
71. Arachchilage NAG, Cole M (2011) Designing a mobile game for home computer users to protect against "phishing attacks". Int J e-Learn Secur 1(1/2)
72. Arachchilage NAG, Love S (2014) Security awareness of computer users: a phishing threat avoidance perspective. Comput Hum Behav 38:304–312
73. Levine J (2008) DNS blacklists and whitelists, IRTF anti-spam research group, Nov 2008, Internet Draft draft-irtf-asrg-dnsbl-08.txt

74. Microsoft, Sender ID, 2008. http://www.microsoft.com/. Accessed on Sept 2014
75. Sheng S, Wardman B, Warner G, Cranor LF, Hong J, Zhang C (2009) An empirical analysis of phishing blacklists. In: Proceedings of the 6th conference in email and anti-spam, ser. CEAS'09, Mountain view, USA, CA, July 2009
76. Google (2014) Google safe browsing API. http://code.google.com/apis/safebrowsing/. Accessed Oct 2014
77. Google (2014) Google safe browsing lookup API. https://developers.google.com/safe-browsing/lookup_guide/. Accessed Oct 2014
78. RFC 3596—Internet Engineering Task Force. https://www.ietf.org/rfc/rfc3596.txt. Accessed Oct 2014
79. Prakash P, Kumar M, Kompella RR, Gupta M (2010) Phishnet: predictive blacklisting to detect phishing attacks. In: Proceedings of the 29th conference on information communications INFOCOM'10. IEEE Press, Piscataway, NJ, USA, pp 346–350
80. Cao Y, Han W, Le Y (2008) Anti-phishing based on automated individual white-list. In DIM'08: proceedings of the 4th ACM workshop on digital identity management. ACM, New York, NY, USA, pp 51–60
81. Likarish P, Dunbar D, Hansen TE (2008) Phishguard: a browser plug-in for protection from phishing. In: 2nd International conference on internet multimedia services architecture and applications, IMSAA, Bangalore, India, pp 1– 6
82. Cook DL, Gurbani VK, Daniluk M (2008) Phishwish: a stateless phishing filter using minimal rules. In: Tsudik G (ed) Financial cryptography and data security. Springer, Berlin, pp 182–186
83. Zhang Y, Hong JI, Cranor LF (2007) Cantina: a content-based approach to detecting phishing web sites. In: Proceedings of the 16th international conference on World Wide Web, ser. WWW'07. ACM, New York, NY, USA, pp 639–648
84. Chou N, Ledesma R, Teraguchi Y, Mitchell JC (2004) Client-side defense against web-based identity theft. In *NDSS*. The Internet Society
85. Netcraft (2014) Netcraft toolbar, 2006. http://toolbar.netcraft.com/. Accessed Sept 2014
86. CloudMark (2014) http://www.cloudmark.com/en/products/cloudmark-desktopone/index. Accessed Sept 2014
87. Filter IP (2014) http://support.microsoft.com/kb/930168. Accessed Sept 2014
88. E. toolbar (2014) http://download.cnet.com/eBay-Toolbar/3000-125124-10153544.html?tag=contentMain;downloadLinks. Accessed Sept 2014
89. Chandrasekaran M, Narayanan K, Upadhyaya S (2006) Phishing email detection based on structural properties. In: New York state cyber security conference (NYS), Albany, NY
90. Dazeley R, Yearwood JL, Kang BH, Kelarev AV (2010) Consensus clustering and supervised classification for pro ling phishing emails in internet commerce security. In: Knowledge management and acquisition for smart systems and services. Springer Conf, Berlin, Heidelberg, vol 6232, pp 235–246
91. Gansterer WN, Polz D (2009) E-Mail classification for phishing defense. Presented at the proceedings of 31st European conference on IR research on advances in information retrieval, Springer conference, Toulouse, France, pp 449–460
92. Robichaux P, Ganger DL (2006) Gone phishing: evaluating anti-phishing tools for windows. Technical report Sept 2006
93. Liu G, Qiu B, Wenyin L (2010) Automatic detection of phishing target from phishing webpage. In: Pattern recognition (ICPR), 2010 20th international conference, Istanbul, Turkey, Aug 2010, pp 4153–4156
94. Bazarganigilani M (2011) Phishing E-Mail detection using ontology concept and naive Bayes algorithm. Int J Res Rev Comput Sci 2(2):1–4
95. Chen J, Guo C (2007) Online detection and prevention of phishing attacks. Communications and networking in China IEEE, 2007, pp 1–7
96. Kim H, Huh J (2011) Detecting DNS-poisoning-based phishing attacks from their network performance characteristics. Electron Lett 47(11):656–658
97. Chandrasekaran M, Chinchani R, Upadhaya S (2006) Phoney: mimicking user response to detect phishing attacks. In: Symposium on world of wireless, mobile and multimedia networks, IEEE computer society, pp 668–672
98. Zhang H, Liu G, Chow T, Liu W (2011) Textual and visual content based anti-phishing: A Bayesian approach. IEEE Trans Neural Netw 22(10):1532–1546
99. Ma L, Ofoghi B, Watters P, Brown S (2009) Detecting phishing emails using hybrid features. IEEE conference on UIC-ATC '09, Brisbane, pp 493–497
100. Ma L, Yearwood J, Watters P (2009) Establishing phishing provenance using orthographic features. IEEE conference on eCrime'09, Tocoma pp 1–10
101. Benuskova L, Kasabov N (2007) Evolving connectionist systems (ECOS). In: Computational neurogenetic modeling.: Springer, US, pp 107–126
102. Alnajim A (2015) A country based model towards phishing detection enhancement. Int J Innov Technol Explor Eng 5(1):52–57
103. Moghimi M, Varjani AY (2016) New rule-based phishing detection method. Exp Syst Appl 53:231–242
104. Angelov PP, Filev DP, Kasabov N (2010) Evolving intelligent systems: methodology and applications, vol 12. Wiley, New York
105. ALmomani A, Wan T, Al-Saedi K, Altaher A, Ramadass S, Manasrah A (2011) An online model on evolving phishing E-mail detection and classification method. J Appl Sci 11(18):3301–3307
106. Almomani A, Wan T, Altaher A, Manasrah A, ALmomani E, Anbar M, ALomari E, Ramadass S (2012) Evolving fuzzy neural network for phishing emails detection. J Comput Sci 8(7):1099–1107
107. Almomani BB, Gupta TWan et al (2013) Phishing dynamic evolving neural fuzzy framework for online detection "Zero-day" phishing email. Indian J Sci Technology 6(1):3960–3964
108. del Castillo M, Iglesias A, Serrano JI (2007) An integrated approach to filtering phishing emails computer aided systems theory. EUROCAST 2007, vol 4739. Springer, Berlin, pp 321–328
109. Islam MR, Abawajy J, Warren M (2009) Multi-tier phishing email classification with an impact of classifier rescheduling. In: The international symposium on pervasive systems, algorithms, and networks, IEEE conference, Kaohsiung, Taiwan, pp 789–793
110. Yearwood J, Mamadov M, Banerjee A (2010) Profiling phishing emails based on hyperlink information. In: 2010 International conference on advances in social networks analysis and mining, IEEE conference, Odense, Denmark, pp 120–127
111. Liu W, Huang G, Liu X, Zhang M, Deng X (2005) Detection of phishing web pages based on visual similarity. In: The proceedings of 14th international world wide web conference Chiba, pp 1060–1061
112. Fu AY, Wenyin L, Deng X (2006) Detecting phishing web pages with visual similarity assessment based on earth mover's distance (emd). IEEE Trans Dependable Secur Comput 3(4):301–311
113. Liu W, Deng X, Huang G, Fu AY (2006) An anti-phishing strategy based on visual similarity assessment. IEEE Internet Comput 10(2):58–65

114. Medved E, Kirda E, Kruegel C (2008) Visual-similarity-based phishing detection. In: The proceedings of the 4th international conference on security and privacy in communication networks, NY, USA, pp 234–245
115. Hara M, Yamada A, Miyake Y (2009) Visual similarity-based phishing detection without victim site information. In: IEEE symposium on computational intelligence in cyber security, CICS 2009 Nashville, pp 30–36
116. Atzori L, Iera A, Morabito G (2010) The internet of things: a survey. Comput Netw 54:2787–2805
117. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of things (IoT): a vision, architectural elements, and future directions. Future Gener Comput Syst 29(7):1645–1660
118. Roman R, Najera P, Lopez J (2011) Securing the internet of things. Computer 44(9):51–58
119. Koroneous GL (2015) Enterprise tech spotlight: IoT tipping point, phishing scams, retail breaches. http://news.verizonenterprise.com/2015/08/iot-retail-breaches-phishing-security/
120. Bertlucci J. Internet of thingbots: the new security worry http://www.informationweek.com/big-data/big-data-analytics/internet-of-thingbots-the-new-security-worry/d/d-id/1234973
121. Gorman M. The internet of things isn't safe: thousands of smart gadgets hacked to send spam and phishing emails. http://www.engadget.com/2014/01/17/internet-of-things-hacked-malicious-email-phishing/
122. Almomani A, Gupta BB, Atawneh S, Meulenberg A, Almomani E (2013) A survey of phishing email filtering techniques. IEEE Commun Surveys Tutor 15(4):2070–2090
123. Proofpint. Proofpoint uncovers internet of things (IoT) cyber-attack. http://investors.proofpoint.com/releasedetail.cfm?releaseid=819799
124. Mathworks (2014) MATLAB: the language of technical computing. http://www.mathworks.in/products/matlab/. Accessed Oct 2014
125. WEKA—University of Waikato, New Zealand, EN (2006) Weka-data mining with open source machine learning software in java. http://www.cs.waikato.ac.nz/ml/weka, (2006/01/31). Accessed Sept 2014
126. Rapidminer (2007) Rapidminer: predictive analysis and data mining. https://rapidminer.com/. Accessed Mar 2016
127. Rattle: A Data Mining toolkit in R (2013) https://code.google.com/p/rattle/. Accessed Mar 2016
128. Open NN: An Open Source Neural Networks C Library (2006) http://opennn.cimne.com/. Accessed Sept 2014
129. Karypis Lab (2006) CLUTO: data clustering software. http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview. Accessed Mar 2016
130. Müllner D (2013) fastcluster: Fast hierarchical, agglomerative clustering routines for R and Python. J Stat Softw 53(9):1–18
131. Munchen (2008) ELKI: environment for developing KDD-application supported by index structures. http://elki.dbs.ifi.lmu.de/. Accessed Mar 2016