Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 44 No. 7
July 2023

Email Spoofing: In Today's Era

Prashant D. Chauhan[1],Apurva M. Shah[2]
Department of Computer Science and Engineering,
The Maharaja Sayajirao University of Baroda,
Vadodara, Gujarat, India
prashant.chauhan-cc@msubaroda.ac.in[1], apurva.shah-cse@msubaroda.ac.in[2]

Abstract-Email spoofing has been there since many years making it one of the main choices of the attackers or spammers who wish to gain access to some private information of victims to misuse it or to get some kind of financial benefit. We all know that email spoofing is possible only because of the fact that the entire email system works on SMTP which doesn't provide any mechanism to check authentication of the sender of the email. With the increasing research in the field of security many protocols have been designed to overcome the problem of spoofing of email address. In this paper, we try to find out whether email spoofing is still possible or not by the year 2023.
Keywords-Email Spoofing, SMTP, SPF, DMARC, DKIM

1. INTRODUCTION

Email has now a days become one of the most widely used means of communication which is quick as well as cost-efficient [1] [2]. With the increasing use of email, the cases of email spoofing have also been increasing. To prevent email spoofing many protocols have been developed like SPF, DKIM and DMARC [3]. With time, adoption of these protocols has also increased drastically. Now, it is a matter of debate, whether the problem of email spoofing has been resolved completely or partially after the use of these anti-spoofing protocols [4] [5].

In this paper, we try to find out whether as of today, spoofing of email is possible or not even after the usage of anti-spoofing protocols by majority of the email servers and the domain owners. For this, we first created our own email server, and configured the anti-spoofing protocols on it to prevent email spoofing. Then, we tried various approaches to bypass one or more of the anti-spoofing protocols in order to check whether our spoofed email reaches the inbox or spam of the recipient user or gets rejected completely by the receiving email server.

We identified top 10 email service providers and performed our experiment on them [6] [7]. The selected service providers for our experiment were Gmail, Outlook, Proton Mail, AOL Mail, Yahoo Mail, Zoho Mail, iCloud Mail, Rediff Mail, mail.com and Yandex Mail. We made various test cases based on varying scenarios taking into consideration all kind of possibilities and tried to spoof emails sent to these 10 email service providers.
By performing this experiment, we analyzed the success ratio of spoofed email on the selected email servers. The main purpose of this research is to find whether the existing protocols for authentication of email sender are effective in preventing or at least detecting spoofed emails or not. For this, we tried to think as an attacker and attempted all possible ways to spoof an email and bypass the existing authentication protocols.

2. EMAIL SPOOFING

A. What is Email Spoofing?

The act of sending an email that has been altered to appear as though it has come from a different, reliable source other than the sender is known as email spoofing [7] [8]. The use of email spoofing in spam and phishing attempts is common because it gives the recipient the impression that the email is authentic. Email spoofing's primary goal is to deceive users into opening or responding to emails received from a fictitious sender [9]. The purpose of email spoofing is to deceive recipients into thinking that the message is from someone they know and trust. The attacker attempts to gain access to some type of information that it can use in some way, once the receiver has been persuaded about the identity of the email's sender.

B. How Does Email Spoofing Works?

To spoof an email, main requirement is of an SMTP (Simple Mail Transfer Protocol) server and an application for sending an email. The attacker modifies the FROM, REPLY-TO, and RETURN-PATH

addresses in the message header while the email is being composed using this kind of application. Now, regardless of its actual source, any person who receives this updated email message will presume it has originated from the falsified address [10]. Email spoofing is only possible because SMTP was not initially intended to validate email sender addresses [11]. Although, certain methods have been created over time to avoid and detect email spoofing, their acceptance is quite low [12].

The “Reply-To” field can also be modified or even completely changed by the sender of an email and thus can result in spoofing of email phishing attack. Via the “Reply-To” field the receiving email server gets to know where to send a reply to, and this can be spoofed to be different email id instead of the original sending user’s email address.

The receiving email server or even the SMTP protocol cannot identify or take any action to identify the email as being a faked one in such circumstances, where the reply to address and the sender address are different. The receiving server now has full responsibility for comparing the two addresses and deciding whether it believes the email to be from a reliable sender or not [13]. A user risks falling prey to the attacker, if they fail to recognize an email with a spoofed sender. Before concluding that an email is authentic, users must analyze its source code. The IP address of the user who sent the email can be found in the email's source code, which is accessible to the recipient user. From the source code, the user can verify if the email message has successfully passed the SPF, DKIM and DMARC testes or not [14].

An example of source code of a forged email can be seen in Fig. 1. As seen in the figure, the “Return-Path” field has the spoofed email address “prashant@gmail.com” which is shown to the user in inbox, while “Received-From” field has domain “mail-server.in” which is the original sending domain.

```
ARC-Authentication-Results: i=1; mx.google.com;
       dkim=pass header.i=@mail-server.in header.s=modoboa header.b="MM2L8A5/";
       spf=softfail (google.com: domain of transitioning prashant@gmail.com does
       dmarc=fail (p=NONE sp=QUARANTINE dis=NONE) header.from=gmail.com
Return-Path: <prashant@gmail.com>
Received: from mail-server.in (mail-server.in. [117.240.215.158])
        by mx.google.com with ESMTPS id j70-20020a638049000000b0040d40aa9825si133
        for <prashant.chauhan-cc@msubaroda.ac.in>
        (version=TLS1_3 cipher=TLS_AES_256_GCM_SHA384 bits=256/256);
        Mon, 04 Jul 2022 22:47:45 -0700 (PDT)
Received-SPF: softfail (google.com: domain of transitioning prashant@gmail.com do
Authentication-Results: mx.google.com;
       dkim=pass header.i=@mail-server.in header.s=modoboa header.b="MM2L8A5/";
       spf=softfail (google.com: domain of transitioning prashant@gmail.com does
       dmarc=fail (p=NONE sp=QUARANTINE dis=NONE) header.from=gmail.com
```

Fig. 1. Example of a spoofed email

A number of the fields in the email message header are modified by the attacker to spoof emails [15]. A person who wants to spoof an email address can change the address in the "From" field of email header to any desired email address. Then, as the "Return-Path" field, this fake "From" field is added to the email message header.

The email address that is displayed to the recipient user in their inbox is the one in the "From" field of the email header. Now, when a person receives a spoof email, he or she will see the fake "From" email address rather than the real one [16].Numerous SMTP protocols have been developed in order to stop and detect email spoofing. The three most efficient ones are SPF, DKIM, and DMARC. The Internet Engineering Task Force (IETF) has published and standardized each of these protocols [17].

C. SPF

The owner of a domain specifies the IP address of the email servers used to send emails purporting to be from that domain for SPF or Sender Policy Framework. The domain owner then enters these IP addresses as a specially structured TXT record in the SPF record on his DNS server. The SPF record of the transmitting domain will be checked by the receiving email server whenever it gets an email from that domain.

An Email is deemed to have originated from authentic sender, if the IP address used to connect to the receiving server is same as the IP address mentioned in the SPF record of the domain described in “From” field of the email header. The SPF check will fail, if it doesn't match any of the IP addresses specified in the domain's SPF record, which signifies that the sending email address is not authentic [18] [19].

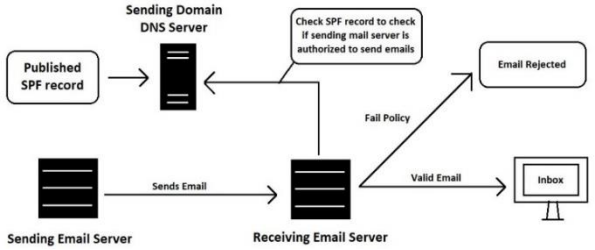

Fig. 2. Sender Policy Framework (SPF) Process Flow

The receiving email server has the option of either delivering the email to the recipient's inbox or entirely rejecting it [20]. An attacker can easily pass the SPF test by using different emails in “From” field and “Return-Path” field. In a sense, we might claim

that the domains with published SPF records are more vulnerable to email spoofing than domains without published SPF records [21] [22].

D. DKIM

Another one of the techniques used for email address authentication is DKIM or DomainKeys Identified Mail. It is designed to stop emails with counterfeit sender addresses, a phishing and email spam tactic that is frequently employed. It enables the receiving email server to determine if emails claiming to be sent from a particular domain are actually authenticated or not [23].

To verify the sending email address for an email is authenticated, DKIM uses a private-key and public key combination. The receiving email server can determine whether or not the email appears to have come from an authenticated user by confirming the DKIM signed email. Additionally, it makes sure that no alterations were made between the time the email was sent and when it was received by the receiving email server [23].

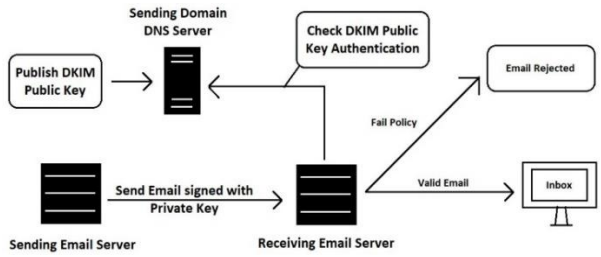

Fig. 3. DomainKeys Identified Mail (DKIM) Process Flow

For the user who wants to send an email, a public-key and private-key pair is first generated on the sender side. The public key is then added to the transmitting domain's DNS server so that the receiving email server can use it later to check the email's authenticity. The private key of the domain is used to encrypt all emails before they are transmitted to the recipient email server [24]. The email server at the receiving end, first reads the sending domain's DKIM record to determine its public key. The receiving email server then attempts to decode the email message using this public-key of the sending domain. If it is successfully decrypted, two things are guaranteed: first, that the email message was not altered during transmission and that it is unaltered from the time it was sent by the sender email server; and second, that the email genuinely originated from the same domain as is claimed in the email. The DKIM test is considered to have passed, if the email message is successfully decrypted; otherwise, it is said to have failed.

E. DMARC

DMARC or Domain-based Message Authentication, Reporting and Conformance is one of the protocols designed to prevent the spoofing of email. In the event that a received email is thought to be a forgery, it gives users the tools to recognise it and take the appropriate action. SPF and DKIM, two other existing protocols, work in conjunction with DMARC to help the receiving email server determine what to do if an email fails one or both of the SPF and DKIM checks. DMARC merely gives instructions on what to do if email spoofing is discovered and does not offer any means for detecting it. SPF and DKIM are always checked first when an email is received. Then, the DMARC policy is used to determine what should be done based on the results of the SPF and the DKIM checks [25]. The sender email address identifier alignment is also checked by DMARC. If DMARC is unsuccessful, the transmitting domain user's DMARC policy is used for deciding, and if it is successful, the email is delivered to the intended recipient [26].

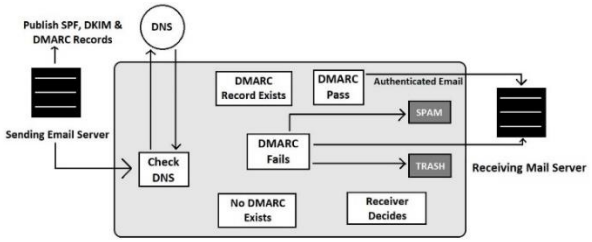

Fig. 4. Domain-based Message Authentication, Reporting and Conformance (DMARC) Process Flow

If the SPF, DKIM, and DMARC tests fail, one of the following options may be used to deliver the email: (i) send it to the inbox; (ii) send it to the spam or junk or trash; or (iii) reject it entirely. When one or both of the test fails, DMARC takes the guesswork out of the receiving email server's decision-making process [21] [22] and help it take appropriate decision as per the choice of the originating domain owner.

3. THE EXPERIMENT AND RESULTS

For the experiment of email spoofing, we used Modoboa version 1.17.0 installed on Ubuntu 20.04.3 LTS as our email server. The SMTP protocol accepts two kinds of sender email ids: one in the email header and the other in the body of the email [16].

For spoofing of email and to bypass the SPF check, we placed our own email id i.e. "prashant@mail-server.in" as the sender email address in the SMTP email header. This value is used while checking of the SPF record at the receiving email server and thus our spoofed email passes the SPF check. The spoofed email id i.e. "prashant@gmail.com" in this case, is placed in the "from" field of the SMTP email body which is displayed to the recipient user of the receiving email server as shown in Fig 5. Thus, the victim users see spoofed email id in the "From" field in their inbox on receipt of the spoofed email instead of the original sender mentioned in the "Return-Path" field.

```
sender = 'prashant@mail-server.in'
receivers = ['prashantchauhan25@gmail.com']

message = """From: Prashant Chauhan <prashant@gmail.com>
Subject: Meeting Reminder 12

Sir, reminder again..
Please find below the updated schedule for the meeting:
Date: 10 Nov 2020
Time: 10 am onwards
```

Fig. 5. Part of python script used to send spoofed email

To bypass the DKIM test, we encrypted the email using our own private key and at the receiving side, the email server uses the "Return-Path" field to find the domain used to encrypt the email, which is our original email server and not the actual email id that is displayed to the recipient user mentioned in the "From" field of email body. For this, we modified the Opendkim Key Table and Signing Table config files so as to sign all the spoofed domains with the private key of our own server and not with the actual spoofed domain. In this way, the receiving email server decrypts our email using our public key and passes the DKIM test.

From our experiment, it is found that, for our test spoofed emails, we are successfully able to bypass SPF and DKIM using the above-mentioned methods for majority of the email servers, but DMARC test gets failed in case where DMARC record is published by a domain owner in its DNS records. This is due to the facility of identifier alignment checking in DMARC, which ensures that the "From" field in email body and the "Return-Path" field in email header must be similar otherwise DMARC is failed. To overcome this, we tried to send multiple email ids in the "From" field of email body as shown in Fig 6.

```
sender = 'prashant@mail-server.in'
receivers = ['prashantchauhan25@gmail.com']

message = """From: Prashant Chauhan <prashant@gmail.com>
<prashant@mail-server.in>
To: <prashantchauhan25@gmail.com>
Subject: Meeting Reminder 12
```

Fig 6: Part of python script used to send spoofed email with multiple email id's in "From" field

A. Scenario – 1: Spoofing our own domain having SPF, DKIM and DMARC records

In this case, we selected our own email server domain as the sending email domain i.e., **mail-server.in.** The selected sending domain has implemented all three authentication protocols i.e. Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM) and Domain-based Message Authentication, Reporting and Conformance (DMARC). We selected **prashant@mail-server.in** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC
Gmail	Inbox	Pass	Pass	Pass
Outlook	Spam	Pass	Pass	Pass
Proton Mail	Inbox	Pass	Pass	Pass
AOL Mail	Inbox	Pass	Pass	Pass
Yahoo! Mail	Inbox	Pass	Pass	Pass
Zoho Mail	Inbox	Pass	Pass	Pass
iCloud Mail	Inbox	Pass	Pass	Pass
Rediffmail	Inbox	Pass	Pass	Pass
Mail.com	Inbox	Pass	Pass	Pass
Yandex Mail	Inbox	Pass	Pass	Pass

Table 1: Experiment results for prashant@mail-server.in

From the above table, we can conclude that our email server can successfully deliver email to the inbox of all major email service providers and the sent email passed all three authentication protocols successfully. Only in case of outlook.com, our email was sent to spam instead of inbox despite passing all three protocols.

B. Scenario – 2.1: Spoofing other domain having SPF, DKIM and DMARC records

In this case, we selected **gmail.com** as the sending email domain which uses all three authentication protocols SPF, DKIMandDMARC. We selected **prashant@gmail.com** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Fail	via shown
Outlook	Spam	Pass	Pass	Fail	-
Proton Mail	Inbox	Pass	Pass	Fail	warning
AOL Mail	Inbox	Pass	Pass	Fail	-
Yahoo! Mail	Inbox	Pass	Pass	Fail	-
Zoho Mail	Spam	Pass	Pass	Fail	via shown
iCloud Mail	Inbox	Pass	Pass	Fail	-
Rediffmail	Inbox	Pass	Fail	Fail	warning
Mail.com	Inbox	Pass	Pass	Fail	-
Yandex Mail	Spam	Pass	Pass	Fail	-

Table 2: Experiment results for prashant@gmail.com with single sender email id

From the above table we found that DMARC was failed in all the cases due to identifier alignment issue, while both SPF and DKIM were passed by all the email servers used in the experiment for the spoofed email. Gmail and Zoho Mail shows "via" message along with the sender email id. Proton Mail and Rediffmail displays a warning message to the user along with the email and in most cases, the email is successfully delivered to inbox without any kind of warning to the user.To further try to bypass the authentication protocols, we tried to send the spoofed email with multiple senders, the results of which are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	Spam	Pass	Pass	Fail	-
Proton Mail	Inbox	Pass	Pass	Pass	warning
AOL Mail	Inbox	Pass	Pass	Fail	-
Yahoo! Mail	Inbox	Pass	Pass	Fail	-
Zoho Mail	Spam	Pass	Pass	Fail	via shown
iCloud Mail	Inbox	Pass	Pass	Fail	-
Rediffmail	Inbox	Pass	Fail	Fail	warning
Mail.com	Not delivered				
Yandex Mail	Spam	Pass	Pass	Fail	-

Table 3: Experiment results for prashant@gmail.com with multiple sender email id

From this table, we can see that by using multiple sender ids we were able to pass all three protocols in case of Gmail and Proton Mail which was not possible with single sender id. Even no "via" message was shown in case of Gmail for the spoofed email with multiple sender email ids which was displayed in case of spoofed email with single sender email id. Also, in case of mail.com the email was completely rejected and not delivered to the recipient user when multiple sender email ids were used.

C. Scenario – 2.2: Spoofing other domain having SPF, DKIM and DMARC records

In this case, we selected **outlook.com** as the sending email domain which uses all authentication protocols SPF, DKIMandDMARC. We selected **prashant@outlook.com** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Fail	via shown
Outlook	spam	Pass	Pass	Fail	-
Proton Mail	Inbox	Pass	Pass	Fail	warning
AOL Mail	Inbox	Pass	Pass	Fail	-
Yahoo! Mail	Inbox	Pass	Pass	Fail	-
Zoho Mail	Spam	Pass	Pass	Fail	via shown
iCloud Mail	Inbox	Pass	Pass	Fail	-
Rediffmail	Inbox	Pass	Fail	Fail	-
Mail.com	Inbox	Pass	Pass	Fail	-
Yandex Mail	Spam	Pass	Pass	Fail	-

Table 4: Experiment results for prashant@outlook.com with single sender email id

From the above table, we can see that DMARC was failed in all the cases due to identifier alignment issue while both SPF and DKIM were passed by all the email servers used in the experiment for the spoofed email. Gmail and Zoho Mail shows "via" message along with the sender email id and Proton Mail displays a warning message to the user along with the email and in majority of the cases, the email is successfully delivered to inbox without any kind of warning to the user.

To further try to bypass the authentication protocols, we tried to send the spoofed email with multiple senders, the results of which are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	spam	Pass	Pass	Fail	-
Proton Mail	Inbox	Pass	Pass	Pass	warning

AOL Mail	Inbox	Pass	Pass	Fail	-
Yahoo! Mail	Inbox	Pass	Pass	Fail	-
Zoho Mail	Spam	Pass	Pass	Fail	via shown
iCloud Mail	Inbox	Pass	Pass	Fail	-
Rediffmail	Inbox	Pass	Pass	Fail	-
Mail.com	Not delivered				
Yandex Mai	Spam	Pass	Pass	Fail	-

Table 5: Experiment results for prashant@outlook.com with multiple sender email id

From this table, we can see that by using multiple sender ids, we were able to pass all three protocols in case of Gmail and Proton Mail which was not possible with single sender id. Unlike the case with single sender email id, even no "via" message was shown in case of Gmail for the spoofed email with multiple sender email ids. Also, in case of Mail.com the email was completely rejected and not delivered to the recipient user when multiple sender email ids were used.

D. Scenario – 3: Spoofing a domain having only DKIM record

In this case, we selected **msubaroda.ac.in** as the sending email domain which usesonly DKIM protocol. We selected **prashant@msubaroda.ac.in** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	none	via shown
Outlook	Spam	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	none	-
AOL Mail	Inbox	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Spam	Pass	Pass	none	-
Rediffmail	Inbox	Pass	Fail	none	-
Mail.com	Inbox	Pass	Pass	none	-
Yandex Mai	Spam	Pass	Pass	none	-

Table 6: Experiment results for prashant@msubaroda.ac.in with single sender email id having only DKIM record

From the above table, we can see that in all the cases, the result of DMARC is shown as "none" as the selected domain has not published DMARC record. It is to be noted that in all these cases, SFP was passed even though the selected domain has not published any SPF record. Also, Gmail and Zoho Mail shows "via" message along with the sender email id.

To further try to bypass the authentication protocols, we tried to send the spoofed email with multiple senders, the results of which are shown in Table7.

From this table, we can see that by using multiple sender ids we were able to pass all three protocols in case of Gmail and Proton Mail which was not possible with single sender id. Also, a "via" message was shown in case of Zoho Mail and warning was shown only in Proton Mail to alert the recipient user of possibility of spoofing of the sender email id.

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	Spam	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	Pass	warning
AOL Mail	Inbox	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Spam	Pass	Pass	none	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Not delivered				
Yandex Mai	Inbox	Pass	Pass	none	-

Table 7: Experiment results for prashant@msubaroda.ac.in with multiple sender email ids having only DKIM record

E. Scenario – 4: Spoofing a domain having only SPF record

In this case, we selected **orthocarehospital.in** as the sending email domain and we published only SPFrecords for this domain for the purpose of our experiment. We selected **prashant@orthocarehospital.in** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	none	via shown
Outlook	Inbox	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	none	-
AOL Mail	Inbox	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	via

					shown
iCloud Mail	Inbox	Pass	Pass	none	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Inbox	Pass	Pass	Pass	-
Yandex Mai	Spam	Pass	Pass	none	-

Table 8: Experiment results for prashant@orthocarehospital.in with single sender email id having only SPF record

From the above table, we can see that result of DMARC is shown as "none" as the selected domain has not published DMARC record. It is to be noted that in all the cases both SFP and DKIM were passed even though the selected domain has only published its own SPF record and all the email were delivered to the inbox except Yandex Mail which delivered the spoofed email in spam folder. Also, Gmail and Zoho Mail shows a "via" message along with the sender email id except which warning was not shown in any other of the selected email servers.The results for multiple sender ids in spoofed email are shown in Table 9. From this table, we can see that by using multiple sender ids we were able to pass all three protocols in case of Gmail and iCloud Mail which was not possible with single sender id. The "via" message was shown only in case of Zoho Mail and the spoofed email was delivered to spam folder for AOL Mail, Yahoo Mail, iCloud Mail and Yandex Mail. For the rest, all our spoofed emails were successfully delivered to the inbox of the recipient user.

Receiving Server Name	**Delivery Location**	**SPF**	**DKIM**	**DMARC**	**Remarks**
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	Inbox	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	none	-
AOL Mail	Spam	Pass	Pass	none	-
Yahoo! Mail	Spam	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Spam	Pass	Pass	Pass	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Not delivered				
Yandex Mai	Spam	Pass	Pass	none	-

Table 9: Experiment results for prashant@orthocarehospital.in with multiple sender email id with only SPF record

F. Scenario – 5: Spoofing a domain having both SPF and DKIM records

In this case, we selected **orthocarehospital.in** as the sending email domain and publishedboth SPF and DKIM records in the DNS for this domain. Then, we selected **prashant@orthocarehospital.in** as the spoofed sending email id for this scenario. The results of our experiment for this case are as follows:

Receiving Server Name	**Delivery Location**	**SPF**	**DKIM**	**DMARC**	**Remarks**
Gmail	Inbox	Pass	Pass	none	via shown
Outlook	Inbox	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	none	-
AOL Mail	Inbox	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	-
iCloud Mail	Inbox	Pass	Pass	none	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Inbox	Pass	Pass	Pass	-
Yandex Mai	Spam	Pass	Pass	none	-

Table 10: Experiment results for prashant@orthocarehospital.in with single sender email id having SPF and DKIM records

From the above table, we can see that result of DMARC is shown as "none" as the selected domain has not published any entry for DMARC in its DNS records. It is to be noted that in all the cases both SFP and DKIM were passed even though the selected domain has published its own SPF and DKIM records as our spoofed email passed both SPF and DKIM tests. All the emails were successfully delivered to the inbox of the recipient user except for Yandex Mail, where the spoofed email was delivered in the spam folder. Also, only Gmail shows a "via" message along with the sender email id to alert the recipient user of possibility of spoofed sender email address in the received email. Rest all other servers delivered our spoofed email in the inbox of the recipient user without giving any kind of warning to the user.

Then, we performed the same experiment by sending multiple sender ids in spoofed email whose results can be seen in Table 11. From this table, we can see that by using multiple sender ids we were able to pass all three protocols in case of Gmail, Proton Mail and iCloud Mail which was not possible with single sender id. Also, a "via" message was shown only in case of Zoho Mail to alert the

recipient user. Our spoofed email was delivered to spam folder only for AOL mail and Yandex mail, for the rest it was successfully delivered to the inbox of the recipient user.

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	Inbox	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	Pass	-
AOL Mail	Spam	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	Pass	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Inbox	Pass	Pass	Pass	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Not delivered				
Yandex Mai	Spam	Pass	Pass	none	-

Table 11: Experiment results for prashant@orthocarehospital.in with multiple sender email id having only SPF and DKIM records

G. Scenario – 6: Spoofing a domain having none of the authentication records

In this case, we selected **gujarattourism.com** as the sending email domain which uses none of the authentication protocols. We selected **prashant@gujarattourism.com** as the spoofed sending email id in this case. The results of our experiment in this case are as follows:

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	none	via shown
Outlook	Spam	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	none	-
AOL Mail	Inbox	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	none	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Inbox	Pass	Pass	none	-
Rediffmail	Inbox	Pass	Fail	none	-
Mail.com	Inbox	Pass	Pass	none	-
Yandex Mai	Spam	Pass	Pass	none	

Table 12: Experiment results for prashant@gujarattourism.com with single sender email id

From the above table, we can see that result of DMARC is shown as “none” as the selected domain has not published DMARC record. It is to be noted that in all the cases both SFP and DKIM were passed even though the selected domain has not published its own SPF and DKIM records and all the emails were delivered to the inbox except for Yandex Mail and Outlook. Also, only Gmail and Zoho Mail shows a “via” message along with the sender email id to warn the user.

Receiving Server Name	Delivery Location	SPF	DKIM	DMARC	Remarks
Gmail	Inbox	Pass	Pass	Pass	-
Outlook	Inbox	Pass	Pass	none	-
Proton Mail	Inbox	Pass	Pass	Pass	-
AOL Mail	Spam	Pass	Pass	none	-
Yahoo! Mail	Inbox	Pass	Pass	Pass	-
Zoho Mail	Inbox	Pass	Pass	none	via shown
iCloud Mail	Inbox	Pass	Pass	Pass	-
Rediffmail	Inbox	Pass	Pass	none	-
Mail.com	Not delivered				
Yandex Mai	Spam	Pass	Pass	none	-

Table 13: Experiment results for prashant@orthocarehospital.in with multiple sender email id

Then, we performed the same experiment by sending multiple sender ids in spoofed email whose results are displayed in Table 13. From this table, we can see that by using multiple sender ids we were able to pass all three protocols in case of Gmail, Proton Mail and iCloud Mail which was not possible with single sender id. Also, a “via” message was shown only in case of Zoho Mail. The spoofed email was delivered to spam folder only for AOL Mail and Yandex Mail, for rest all, the email was successfully delivered to the inbox of the recipient user. Also, the email was completely rejected by Mail.com server, when multiple sender email ids were used in the spoofed email.

4. DISCUSSION

It can be said that identifier alignment used in DMARC is quite helpful in marking an email as DMARC fail. It checks if the email id in “From” field of the email body is same as the one in email header. If both these email ids are not similar, DMARC test is failed, and decision is made by the recipient email server based on the DMARC policy established by the sending email domain owner. It is found that even when DMARC test is failed, still none of the email servers rejected our spoofed

email because of their selected DMARC policy. The liberal DMARC policy of allowing the email even in case of failure of DMARC, leads the email to the inbox of the user which might be dangerous for the recipient user. Also, with our experiment we can say that using multiple email ids in the "From" field of email body, identifier alignment test can be passed, if one of the email ids in the "From" field of email body is kept same as the one in email header. There must be some techniques to ensure identifier alignment is checked and passed before any email is received at the receiving email server which will ensure prevention of email spoofing.

Also, it is found that some of the email service providers like Gmail and Zoho Mail shows a "via" message along with the sender email id which is quite helpful in detecting the source of the email and warn the user for possibility of spoofing. And, in few cases of spoofing of email, only Proton Mail and Rediffmail displayed a warning message to the user to warn them of possibility of spoofing of email, while rest others displayed no warning to the user regarding possibility of spoofing. A normal user is not expected to go to email details and see the raw email message to check the status of SPF, DKIM and DMARC tests. In such cases, where users are not very tech-savvy, they will only see whatever is displayed on their main screen of the inbox and won't in detail to check the authenticity of the received email and may get in the trap of the attacker. In some cases, and by few of the email servers, this kind of warning message is displayed to the user but not by majority of them. We suggest the email receivers to provide such warning message to the user in all cases where any of the tests are failed or in case spoofing of email is presumed by the receiving email server.

5. THE CONCLUSION

With our experiment, we found that in present scenario the only options for preventing spoofing of email are authentication protocols namely SPF, DKIM and DMARC which are not sufficient to prevent email spoofing completely. We also found that there are multiple ways by which we may bypass all the three authentication protocols and make way for the spoofed email to the inbox of the recipient user instead of marking the email as spam or rejecting the email all together. As a conclusion of this experimental study, we confirm that spoofing of email is still possible by the attackers and there is a critical need of some additional security measures to ensure prevention and detection of email spoofing in addition to the existing protocols like SPF, DKIM and DMARC.

REFERENCES

[1] Asif Karim, Sami Azam, Bharanidharan Shanmugam, Krishnan Kannoorpatti and Mamoun Alazab, "A Comprehensive Survey for Intelligent Spam Email Detection," in *IEEE Access*, 2019.

[2] Bridget Opazo, Don Whitteker and Chen-Chi Shing, "Email Trouble: Secrets of Spoofing, the Dangers of Social Engineering, and How We Can Help," in *13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, 2017.

[3] "DKIM Frequently Asked Questions," [Online]. Available: https://dkim.org/info/dkim-faq.html.

[4] "DMARC Overview," [Online]. Available: https://dmarc.org/overview/.

[5] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami and Michael Bailey, "You've Got Vulnerability: Exploring Effective Vulnerability Notifications," in *25th USENIX Security Symposium*, 2016.

[6] Geethapriya Liyanage and Shantha Fernando, "A comprehensive secure email transfer model," in *IEEE International Conference on Industrial and Information Systems (ICIIS),*, 2017.

[7] Hang Hu and Gang Wang, "End-to-End Measurements of Email Spoofing Attacks," in *27th USENIX Security Symposium*, 2018.

[8] Hang Hu, Peng Peng and Gang Wang, "Towards Understanding the Adoption of Anti-Spoofing Protocols in Email Systems," in *IEEE Cybersecurity Development (SecDev)*, 2018.

[9] Hongming Che, Qinyun Liu, Lin Zou, Hongji Yang, Dongdai Zhou and Feng Yu, "A Content-Based Phishing Email Detection Method," in *IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 2017.

[10] I. Dolnák, "Secure mutual exchange of messages between network nodes inspired by security technologies for electronic mail exchange," in *19th International Conference on*

Emerging eLearning Technologies and Applications (ICETA), 2021.

[11] Ian D. Foster, Jon Larson, Max Masich, Alex C. Snoeren, Stefan Savage and Kirill Levchenko, "Security by Any Other Name: On the Effectiveness of Provider Based Email Security," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[12] Jeremy Clark, P.C. van Oorschot, Scott Ruoti, Kent Seamons and Daniel Zappala, "SoK: Securing Email -- A Stakeholder-Based Analysis (Extended Version)," arXiv, 2018.

[13] Kaiwen Shen, Chuhan Wang, Minglei Guo, Xiaofeng Zheng, Chaoyi Lu, Baojun Liu, Yuxuan Zhao, Shuang Hao, Haixin Duan and Qingfeng Pan, "Weak Links in Authentication Chains: A Large-scale," in *30th USENIX Security Symposium*, 2021.

[14] Kenya Dan, Naoya Kitagawa, Shuji Sakuraba and Nariyoshi Yamai, "Spam Domain Detection Method Using Active DNS Data and E-Mail Reception Log," in *43rd Annual Computer Software and Applications Conference (COMPSAC)*, 2019.

[15] M. H. Jalalzai, W. B. Shahid and M. M. W. Iqbal, "DNS security challenges and best practices to deploy secure DNS with digital signatures," in *12th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, 2015.

[16] Piotr Malek, "Everything You Need to Know About SMTP Security," [Online]. Available: https://mailtrap.io/blog/smtp-security/.

[17] Sarah Scheffler, Sean Smith, Yossi Gilad and Sharon Goldberg , "The Unintended Consequences of Email Spam Prevention," in *International Conference on Passive and Active Network Measurement*, 2018.

[18] Shuji Sakuraba, Minami Yoda, Yuichi Sei, Yasuyuki Tahara and Akihiko Ohsuga, "Improvement of Legitimate Mail Server Detection Method using Sender Authentication," in *IEEE/ACIS 19th International Conference on Software Engineering Research, Management and Applications (SERA)*, 2021.

[19] Zakhar Yung, "SPF Record Explained," [Online]. Available: https://mailtrap.io/blog/spf-records-explained.

[20] Shuji Sakuraba, Minami Yoda, Yuichi Sei, Yasuyuki Tahara and Akihiko Ohsuga, "Sender Reputation Construction method using Sender Authentication," in *IEEE International Conference on Data Science and Computer Application (ICDSCA)*, 2021.

[21] Sourena Maroofi, Maciej Korczyński, Arnold Hölzel and Andrzej Duda, "Adoption of Email Anti-Spoofing Schemes: A Large Scale Analysis," in *IEEE Transactions on Network and Service Management*, 2021.

[22] Wissem Soussi, Maciej Korczynski, Sourena Maroofi and Andrzej Duda, "Feasibility of Large-Scale Vulnerability Notifications after GDPR," in *IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, 2020.

[23] Piotr Malek, "DKIM Explained," [Online]. Available: https://mailtrap.io/blog/dkim/.

[24] Zakir Durumeric, David Adrian, Ariana Mirian, James Kasten, Elie Bursztein, Nicolas Lidzborski, Kurt Thomas, Vijay Eranti, Michael Bailey and J. Alex Halderman, "Neither Snow Nor Rain Nor MITM...: An Empirical Analysis of Email Delivery Security," in *Proceedings of the 2015 Internet Measurement Conference*, 2015.

[25] Zakhar Yung, "DMARC Explained," [Online]. Available: https://mailtrap.io/blog/dmarc-explained/.

[26] Yuanyuan Grace Zeng, "Identifying email threats using predictive analysis," in *International Conference on Cyber Security And Protection Of Digital Services (Cyber Security)*, 2017.