

## E-mailing Security Issues

TEODOR TÎMPLARU, ROXANA TÎMPLARU

---

**ABSTRACT.** E-mail spoofing - as an important matter of actual security issues related to e-mailing activities - is a fast growing problem and has reached the point where you simply cannot rely on the information displayed in your e-mail client to tell you who really sent a message. Some authorities from countries throughout the world have already enacted laws against this form of "e-mail identity theft," but the more effective solution should be a technological one that makes it possible to authenticate the senders of e-mail messages. Some of the most popular mechanisms currently in development are SPF and DKIM. In this article, we will take a look at the problem and the proposed solutions so far.

---

### 1. Introduction

The term of e-mail spoofing is generally used to describe fraudulent e-mail activities, in which the sender address and other parts of the e-mail header are altered to appear as though the e-mail originated from a different source. In fact, this is a form of identity theft, as the sender pretends to be someone else in order to persuade the recipient to do something (from simply opening the message, to sending money or revealing personal information). Mail spoofing is a technique commonly used for spamming and phishing activities to hide the origin of an e-mail message.

By changing certain properties of the e-mail, such as the *From*, *Return-Path* and *Reply-To* fields (which can be found in the message header), ill-intentioned users can make the e-mail appear to be from someone else than the real sender. The result is that, although the e-mail appears to come from the address indicated in *From* field, it actually comes from another source. Thus, you can play some jokes to your friends or colleagues on different occasions, but apart from that, it is not hard to notice that the main part of some of these fake e-mails may contain annoying advertising banners or hidden little malicious code, ready to infect your computer (spamming) or links to some sites where trying to steal personal data from you (phishing).

### 2. E-mail spoofing by examples

You can do e-mail spoofing with some software (even created by you, if you read and understand the principles described later in this article and you have good programming skills too), but I will detail one simple e-mail spoofing technique in a basic way, which even the "ancient" computers will be able to handle.

For the moment, imagine you as taking part in a situation like this one. You are a young singled man, working in a company named Omniatech, for example. It is

---

*Received:* 9 July 2009.

Friday afternoon, it is 4 o'clock and you are looking forward to the weekend. Besides, Diane - the new gorgeous girl from HR Department - has finally accepted your invitation to dinner together tonight. While checking if everything is O.K. with your restaurant booking, the computer on your desk announces you just have received a new mail, from your boss. You have a bad feeling, but you have to open the mail and read it. The boss has a new task for you, indeed, a report for an important Monday morning meeting. He wants you to finalize the report as quickly as possible, because he will read it the next morning. All the data and the specifications you need are already archived and placed in the Public directory of the internal network. You consult immediately that archive and you soon realize that you cannot finish the job too quickly. It is clear now, that task will ruin your evening. None of your friends works here with you to ask for a little help, except Nick - the System Engineer; but Nick cannot write that report, of course, although you might need his help anyway. You have good relations with your colleagues, but it is Friday afternoon and you realize that it is almost impossible to convince one of them to write that report for you. Well, that is we can call bad luck.

But... What if you could send a "copy" of that mail to one of your colleagues, appearing to him as though sent by the boss himself? That means you could pass the entire task to your colleague. It would not be fair at all, of course, but you could save YOUR evening. Think again of Diane: is it really worth spending an entire evening with her in spite of the fact you may break the relation with your colleague? Or you may consider breaking that relationship it's another problem and decide to postpone its solving on Monday. Anyway, I assume you have answered "Yes" to the above question (Don't worry; if I were you, I would do the same. Because it's a thousand times easier to repair a broken relationship with a man than one with a woman Honestly, the real reason for acting like that might be quite different: did I mention what a gorgeous woman Diane was?)

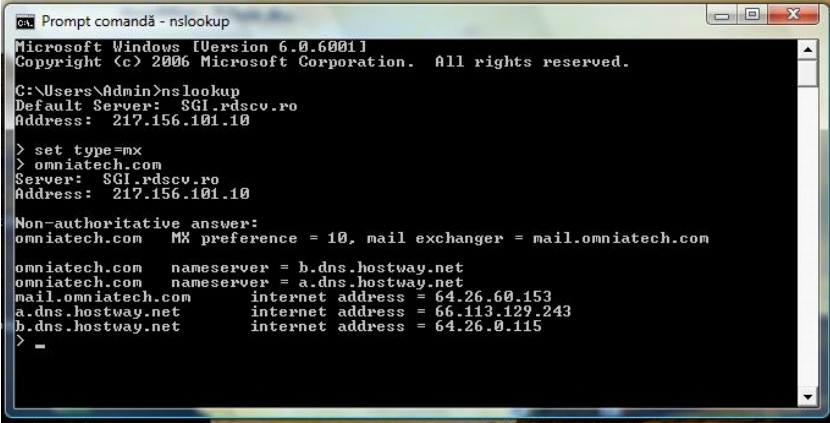
So, it seems you have to "fool" and even "exploit" one of your colleague, therefore you have to be careful. Choose one smart enough to write a good report and also gifted with a good sense of humor, because you'll have to tell him the truth later. Let's say Peter is the right person for "playing" that role. Now, let's see what we can do to save your evening.

As I said, we need a mechanism allowing you (person A) to send an e-mail to someone (person B), but instead of you being listed as the (real) sender, someone else (person C) is being placed in that field. In other words, although A is the author and the sender of the mail, for B it appears as coming from C. Now, suppose that the two e-mail addresses we need to illustrate our procedure are *mark.bossman@omniatech.com* (as boss' address) and *peter.henman@omniatech.com* (as Peter's address), respectively. Notice that your e-mail address is not even necessary, on one hand, and that both e-mail addresses considered are related to the Internet domain owned by Omniatech Company, on the other hand. But, as you will see later, this spoofing procedure will work with any two e-mail addresses, not necessarily created into the same Internet domain. The only condition is the recipient's address to be a valid one. A connection to the Internet is also required, of course.

OK, let us go ahead! As we know, there are many various protocols to handle different types of data on the Internet. One of them is SMTP (Simple Mail Transfer Protocol) and it is the most common protocol for sending and receiving e-mails (*follow the link [22] for more detailed information*). Each site with e-mail capabilities has one or more SMTP servers for handling this kind of traffic. Everyone who had the curiosity to analyze an e-mail header more deeply, could see the whole path the mail travelled, including the mail servers the message came from (i.e. *mta181.mail.re3.yahoo.com* or *mx.google.com* or any other valid mail server address from the Internet).

To find the proper mail servers you need for the e-mail spoofing procedure, in the way described as follows, you might need some little help and assistance from your friend Nick (the System Engineer/Network Administrator). In fact, you need two little tools that probably are not installed or not available on your Windows machine; that depends on the Security Policy (*consult [21] as an excellent book about this matter*) adopted and implemented by Nick, for your company. So, write him quickly a message, asking to allow you running directly from your computer a *Command Prompt* session and a *Telnet* session, respectively. He can do that in an easy manner, by remote logging to your computer from his one and then installing the applications you asked for and/or setting all the rights you need to run them. Of course, Nick will become a little nervous receiving that request coming from you. Ask him to trust you and promise him to telling the whole story later.

Now, being able to open a *Command Prompt* window (*Start → Run → cmd*), enter the command *nslookup* (the acronym comes from *Name Server Lookup*). This command should help us finding the mail servers we are looking for. After executing this command, the name and the DNS related to the Web Server of your ISP (Internet Service Provider) are both listed on your display, as you can see in the following screen-capture image. Of course, the information displayed varies, depending on your actual location and ISP. All the screen-captures images through this article are from one computer from the network I administer myself and are inserted as illustrative examples for the explanations associated.



```
Prompt comandã - nslookup
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Users\Admin>nslookup
Default Server:  $GI.rdscv.ro
Address:  217.156.101.10

> set type=mx
> omniatech.com
Server:  $GI.rdscv.ro
Address:  217.156.101.10

Non-authoritative answer:
omniatech.com    MX preference = 10, mail exchanger = mail.omniatech.com

omniatech.com   nameserver = b.dns.hostway.net
omniatech.com   nameserver = a.dns.hostway.net
mail.omniatech.com internet address = 64.26.60.153
a.dns.hostway.net internet address = 66.113.129.243
b.dns.hostway.net internet address = 64.26.0.115
>
> -
```

Now, remember that we try to find some mail servers, so we need to specify that in our request, somehow. You have to type in *set type=mx* (*mx* comes from *mail*

*exchanger*). Of course, you can type some other acronyms after the equal sign, depends on the type of the servers you search for (i.e. *set type=all*, *set type=cname*, *set type=soa*, etc.), but for now, you are concerned only with finding the mail servers.

Analyzing again the recipient's address - *peter.henman@omniatech.com* - it is obvious that you need finding the mail server for Omniatech Company in order to connect to. You have to type in the Internet domain name of the company you work for - *omniatech.com* - and then wait for the answer to your request. This time there is just one mail server listed - *mail.omniatech.com* - as you can see in the image above, but that generally depends on the dimension of the company involved.

On certain occasions, you might be interested in finding the mail servers for a popular e-mail service provider like Gmail. Then, you should type *gmail.com* and there would be much more information displayed that time, as shown in the next image.

```

Prompt comandă - nslookup
> gmail.com
Server: SCL_rdcv.ro
Address: 217.156.101.10

Non-authoritative answer:
gmail.com      MX preference = 40, mail exchanger = alt4.gmail-smtp-in.l.google
.com
gmail.com      MX preference = 5, mail exchanger = gmail-smtp-in.l.google.com
gmail.com      MX preference = 10, mail exchanger = alt1.gmail-smtp-in.l.google
.com
gmail.com      MX preference = 20, mail exchanger = alt2.gmail-smtp-in.l.google
.com
gmail.com      MX preference = 30, mail exchanger = alt3.gmail-smtp-in.l.google
.com

gmail.com      nameserver = ns2.google.com
gmail.com      nameserver = ns3.google.com
gmail.com      nameserver = ns4.google.com
gmail.com      nameserver = ns1.google.com
gmail-smtp-in.l.google.com internet address = 209.85.220.27
alt1.gmail-smtp-in.l.google.com internet address = 209.85.135.27
alt2.gmail-smtp-in.l.google.com internet address = 209.85.210.97
alt3.gmail-smtp-in.l.google.com internet address = 209.85.222.26
alt4.gmail-smtp-in.l.google.com internet address = 74.125.93.114
ns1.google.com internet address = 216.239.32.10
ns2.google.com internet address = 216.239.34.10
ns3.google.com internet address = 216.239.36.10
ns4.google.com internet address = 216.239.38.10
>

```

Look again at these two images. There are some differences we need to discuss about. Five mail servers (mail exchangers) are displayed for Gmail, just one for Omniatech. Besides, every mail exchanger has a specific attribute, that *MX preference*: 10 for the single Omniatech mail server, five distinct numbers (40, 5, 10, 20 and 30, in that order) for each of Gmail servers, respectively. When you send an e-mail to the mail servers of one important mail service provider like Gmail, usually it is allocated to that mail server with the lowest MX preference, while the rest of servers, those with greater MX preference, generally "play" the role of back-up servers and "act" only on intense traffic conditions. In case of a DOS / DDoS (Denial of Service / Distributed Denial of Service) attack - which means "bombing" one such server with a huge data flow, beyond its normal capability to handle it - the sever(s) with the lowest MX preference should "consume" all that traffic generated by the attack, while the rest of the servers should continue to handle the common and real e-mail traffic activity (*more information related to this subject in [20]*).

As a consequence of what we have discussed above, in order to proceed with our mail spoofing procedure, we have to try connecting to those mail servers with greater

MX preference, because we have seen those servers are rarely (or even not) used, depending on the dimension of the traffic involved at that time. Thus, we could have a more stable connection, for a longer time period, permitting us sending the fake message successfully. Look again at the last image above, that with Gmail servers listed. Except that mail server with MX preference = 5, all the others (with greater MX numbers) have the associated names starting with the string 'alt'. In my opinion, that comes from the word 'alternative', suggesting us that they are alternative (or back-up, as we have previously called) mail servers. Thus, the mail server named *alt4.gmail-smtp-in.l.google.com*, which has the greatest MX number (40), should be our first choice when trying to e-mail spoofing with Gmail servers involved. Ironically, it is the first Gmail server listed, when executing *nslookup* command. As for Omnitech, there is just one mail server, so nothing else to discuss further.

Therefore, we have exactly found the name of the Omnitech server: *mail.omnitech.com*. However, how can we connect to it? As I said earlier, there are several methods to do it, some of them quite simple, while others a bit more complicated (such as writing one or more dedicated PHP scripts, for example), involving much more technical notions and pretty good programming skills. But for now, I want to show you, probably, the easiest technique for performing some e-mail spoofing. It is based on *telnet*, a little and "ancient" software package which allows connection to any other computer on the Internet. It is rarely used nowadays (mainly because of poor security reasons), but during the "Stone Age" of the Internet, it was the most common tool to remote accessing other computers within the network (*for more details, follow the link [23]*).

First, you have to quit *nslookup* command, therefore you have to type *exit*. Now, you can use the same Command Prompt window, already opened, for executing the next few commands. The syntax for the command you need now is: *telnet [server] [port]*. In our case you will type: *telnet mail.omnitech.com 25*, where 25 is the number of the port which SMTP protocol "listen" to, by default. *Telnet* tool might be also used independently (so not through a Command Prompt session); the executable file is placed into the Windows\System32 folder and then, the above command become *o mail.omnitech.com 25* ('o' comes from 'open'), but it is hard to believe you normally have access to it, on a computer within a company network, in a situation like ours assumed here; except the case the Network Administrator is your friend. However, from your personal computer at home (or from your laptop) you can run it directly, if properly installed before. It is very important not to forget specifying the port number (i.e. 25), otherwise the connection is made using the port identified by the number 23, which is the default port for *telnet* program, but unhelpful in our case.

Now, it would be useful to know how the accessed mail server can answer to your *telnet* requests or commands. The most important and relevant answers (in fact some numerical codes) are presented in the following listing.

220 - *Successfully established connection*  
 221 - *Voluntary closing of a previously opened connection*  
 250 - *O.K. status, "You may continue the session"*  
 354 - *Main message, "Start data input"*  
 421 - *Unsuccessfully trying to establish a connection, "Service not available"*  
 451 - *Connection dropout*  
 501 - *Syntax error message, after wrong editing a command*  
 502 - *Error message, after trying to use a non-existent command*  
 550 - *Error message, after trying to send a mail to a non-existent address*

Well, you have finally reached the stage for editing your fake e-mail effectively. Some mail servers (such as Yahoo or even Gmail or Hotmail ones) are very busy and thus the connection to them is limited in time. Therefore, you have to type in rapidly and without errors. In case of syntax or editing errors, type *reset* and re-edit that command.

But in our case, it is hard to believe that the mail server of Omniatech Company is quite busy on Friday afternoon, close to the end of the working program. Therefore, you will not have any problem while trying to connect to that mail server. Now, first of all, you will type the following two lines, thus specifying to the mail server both the sender's and the recipient's mail addresses. (You must use the angular parentheses; otherwise, you receive an error message from the system: *501 syntax error in parameters or arguments.*)

*mail from:* <mark.bossman@omniatech.com >  
*rcpt to:* <peter.henman@omniatech.com >

Now it is time for editing the header of the message. First type in the command *data* on a separate line and you will receive a confirmation code immediately: *354 go ahead*. Three lines are absolutely needed here, as you can see below.

*subject:* *I need your expertise . . .*  
*to:* < peter.henman@omniatech.com >  
*from:* < mark.bossman@omniatech.com >

Some other lines could be added. For example, the spammers and other ill-intentioned users have to add at least two lines more (see the below example), to be able to insert annoying commercial banners or to hide some malicious code (just becoming active after a simple mouse click) or to write some HTML code for commuting you to sites where trying to thieve personal data from you.

*mime-version:* 1.0  
*content-type:* text/html; charset=iso-8859-1

Of course, the specified set of characters could be also *utf-8* or any other set of characters needed for editing and then sending an e-mail not as simple text, but in HTML format. One other line used by some spoofer here, in the header section,

is that related to *reply-to* attribute. Think about: if the field *from* would contain a non-real address, you would probably want reading somehow a possible *reply* to your message. So, for the *reply-to* field of the message header you should write not only a valid mail address, but also an address you could access later. It is obvious you should be very careful, trying to minimize the differences between the addresses specified in *from* and *reply-to* fields, respectively.

In practice, such task is hard to be accomplished. Consider our particular situation assumed here. After finishing the report (some hours later), Peter will have to send it to your boss. There are two possibilities for doing that: writing him a completely new mail, on one hand or using *reply* facility provided by most client mail software, on the other hand. For the first case issued above, you can do nothing to get Peter's report, except for knowing your boss' credentials to be able to break into his mail box; but it is absurd to consider such hypothesis. For the second case issued, there might be one possibility; depending on the permission you might have yourself to create a new internal mail address within the system of Omnitech. Then you could create another box mail as though it would be your boss'. You could use names like *mark.bosman* (a single 's' instead of two) or *mark.bossmann* (two 'n' instead of one), so with minor differences from the original, hardly observed on a quick examination. Thus, you should place one of these addresses into the *reply-to* field of your fake mail and you would have access to the report created by Peter. Finally, you would download the report attached and then you would compose your own message to your boss, in an absolutely normal way this time, sending him the wanted report. It would seem to him as though the entire job would be done by you. That would be quite pretty, don't you think? Another one possibility would be creating a new e-mail address using a well-known web-based mail service provider as Yahoo, Hotmail or Gmail - as an alternative and more personal sender's address - and then somehow convincing the recipient sending his response to that address. You will exactly see later how you could apply this idea in our particular case.

So, you have learned quite a lot by now about e-mail spoofing, but you have not finished the job yet. Let us turn again to our scenario. I have another idea for you to determine Peter doing his best on creating that report. I think you could password the entire archive indicated by the boss with a well-chosen name. What do you think about "promotion"? I think it is a good idea, of course, as it is mine. . .

Now it is time to edit the main part (the message itself) of your spoofed e-mail, but first you have to separate correctly the end of the previous section (the message header) from the beginning of the next section (the main part of the message or the body of the message). You can do that by leaving an empty line between the two last sections mentioned.

After editing the message itself, you have to leave another empty line for mail server to understand that you have finished that part. It follows immediately another special line, containing just the character '.' (period), without quotes, of course. The '.' character plays the role of ending character for a SMTP command session and thus your message is already delivered to Omnitech mail server, being ready to reach the recipient's address. To close the connection simply type in the command *quit*.

The entire procedure is shown in the following listing. All your commands are bolded, while the answers from the system are not. However, do not try to run this example exactly as shown, as the names and mail addresses associated are imaginary.

```

telnet mail.omniatech.com 25
220 mx05.mfg.onr.siteprotect.com ESMTP MailFilterGateway <3.1>
helo mail.omniatech.com
250 mx05.mfg.onr.siteprotect.com
mail from: <mark.bossman@omniatech.com>
250 Ok
rcpt to: <peter.henman@omniatech.com>
250 Ok
data
354 Start message
subject: I need your expertise...
to: <peter.henman@omniatech.com>
from: <mark.bossman@omniatech.com>
mime-version: 1.0

As you probably know, I have an important meeting on Monday morning and I need your expertise and your cooperation. To be more precise, I need a report from you to present it to our clients on Monday morning, attempting to obtain the best results from that meeting. I have to leave the company now, but all the data and the specifications you need are already archived and placed into the Public directory within our network (\Public\Urgent\mondaymeeting.rar). You made good reports in the past, on similar occasions; so I count on you this time too. It is true that my request comes to you too late; but, please, make some efforts and finish that report as quickly as you can. I will need it tomorrow morning for doing all my own homework, too.

Because, in the last few days, I experienced some problems with my Omniatech domain related mail address, when trying to access it from home, please send me the report to the following Gmail address of mine: markb.omniatech@gmail.com.

Mark B.
Omniatech Corp.

P.S. I almost forgot the password for the archive: PROMOTION.

.
250 Ok
quit
221 Bye

Connection to host lost.

```



It is obvious you have to create that Gmail address previously, in order to access it later (with credentials by your choice) for downloading the Peter's report. Of course, if the user name suggested here is not available by any reason, you have to find another one, but also carefully chosen.

Before leaving the company, you might want to salute Peter and wish him having a nice weekend. In fact, you want to see if your e-mail spoofing procedure has succeeded. You might expect Peter to be a little nervous or even irritated. But you find him quite calm and editing something on his computer. Do you remember that password suggested for the boss' archive? I told you it was a good idea! As you can see, the report will be ready some hours later this evening. If you have a modern mobile device (even a cell phone) with Internet access, you can consult later that Gmail address (supposed to be your boss' one), then download the Peter's report (poor Peter!), then write your name on it instead of Peter's, if necessary (poor Peter, again!), then enter your real Gmail (or Yahoo, or Hotmail, etc.) personal account and finally send that report to your boss (using his real Omniatech mail address this time, of course). You can do all that even at the restaurant, while dinning with Diane, if you have five free minutes.

Now you can realize what you have done. You have sent an e-mail to your colleague Peter, but that e-mail has appeared to him as sent by your boss and not by you. Thus, you have forced Peter to read it and to write that report for you, although the real boss' request for writing it has been addressed to you, not to Peter. In fact you have used him, therefore do not forget to tell him the entire story Monday evening. Buy him a couple of beers after work, excuse you and promise him returning such service on similar situation.

Also, do not forget about Nick and tell him the entire story too! You will give him some serious reasons to worry about. Think a little further! Instead of you, it could have been someone else from outside (even one from a concurrent company). Following the procedure as described above, he could have got that report, thus obtaining some (relatively) secret information about your company. As you can see, that is a very serious problem and Nick has to be more careful and find an adequate solution.

As of Diane... Well, you may imagine as you wish the continuation of the relationship with Diane, because it is time to end that scenario. And I could do it in a Hollywood-like style, mentioning that all the facts, people and names depicted above were pure fiction (so, do not ask me to insert pictures of Diane, here!). Any similarities with actual facts, people and names are just simple coincidences.

That was a form of pure e-mail spoofing; in fact, it was a form of identity theft, a bad thing, certainly. Of course, I do not encourage you to follow such practices as described through this article; all examples presented here are for informational and educational purpose only. In order to learn how we can protect against mail spoofing, we have to know as much as we can about it.

To learn even more about e-mail spoofing, let me expose you some other facts, inspired by reality, this time. Some evenings ago, while editing this article, I received the following mail, coming from my boss.

*"I've just received some samples of the clips needed for promoting our next week action. But I've experienced some problems, trying to play them: no image, just sound. I've tried on other computers too... Same trouble! At the Agency, they expect a quick response of acceptance from me, tomorrow morning, but I want to see them first... So, I ask you to come earlier tomorrow morning at the office to fix the problem. Much earlier than you usually did. I'll be there at 8 o'clock.*

*Maria R."*

Yes, my boss is a she. And, as any other boss does, she gives me some bad news, from time to time, just like this one. I normally wake up at 15' past 8 in the morning and hardly arrive in time (just after 9 o'clock) at the office. I have to admit that, but that habit of mine is normal somehow, as I often work on my computer until late at the night. Therefore, imagine what an effort she asks me to do by waking up at 7 o'clock next morning! And all that, because of some missing updated codec packs on her computer, most probably...

Irritated enough, I told my wife what happened to me. And then it seemed to me noticing a vague trace of a smile (barely perceptible) on her pretty face. Suddenly, I became suspicious and I suspected my wife of playing some jokes on me. Then I remembered we contradictorily discussed before about my late wake up in the mornings, I also remembered she knew a lot (quite from me, of course) about my job and about my colleagues (even about my boss) and finally I remembered she was teacher of Computer Science and therefore perfectly able to try e-mail spoofing me with ease.

Probably, she "spied" me in the last few days and then she quickly learned the procedure for e-mail spoofing, thus being ready to test it right on me. Of course, I wasn't quite sure about all that, but I decided not to change anything in my behavior. So, next morning I woke up again at 15' past 8, I arrived at my office at 5' past 9, I asked my boss if everything was OK with her computer and those clip samples ("Yes, everything's fine! Why do you ask me?" she said) and then I followed exactly the steps shown in the listing bellow for e-mail spoofing my wife too, while savoring my coffee.

```

telnet b.mx.mail.yahoo.com 25
220 mta112.mail.re3.yahoo.com ESMTP Ysmtp service ready
mail from: <dan_the_doorman@yahoo.com>
250 sender <dan_the_doorman@yahoo.com> ok
rcpt to: <roxana.timplaru@yahoo.com>
250 recipient <roxana.timplaru@yahoo.com> ok
data
354 go ahead
subject: Mission accomplished!
to: <roxana.timplaru@yahoo.com>
from: <dan_the_doorman@yahoo.com>
mime-version: 1.0

```

*I wrote you because your husband told me it was your idea asking my help to update that codec pack on our boss' computer, this morning. Thank you for your confidence! Even if a little scared at the beginning, just following the instructions your husband gave me, finally the entire process seemed to be quite simple...*

*I enjoyed so much that task, thus you could count on me again, when needed.*

*Sincerely,  
Dan, the Doorman*

*.  
250 Ok dirdel  
quit  
221 mta112.mail.re3.yahoo.com*

*Connection to host lost.*

Although Dan - one man from the security staff of my office - is a good person, I doubt that he has minimal skills to properly operate a computer. I do not even know if he has a Yahoo mail address. Or if he has ever used a web based mail service like Yahoo. But, even if he did, how could he choose a "user name" like the one above? Of course, that mail address is a fake one and therefore the entire mail is a fake; it is just an answer from me to my wife's trying to fool me.

I don't know how other wives are, but mine always wants to have the last *reply*. This is her message (a real one this time) sent to me, a couple of hours later. (In terms of e-mailing it is not exactly a reply, because she realized of course that was a joke of mine and sent that mail to me, not to Dan.)

*"Poor Dan! I heard he missed just by little to be hired by the local subsidiary of Google. He made a small mistake by completing his CV. When required to fill in a contact mail address, he wrote a Yahoo address. I've noticed he hasn't changed it yet..."*

I did not depict all the facts above just for adding some fun to a technical article. In fact, I want to point out some other new important things, when e-mail spoofing with Yahoo mail servers involved. If you run again *nslookup* command, attempting to find the Yahoo mail servers this time, the result looks as bellow.

```

ca: Prompt comandă - nslookup
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Users\Admin>nslookup
Default Server: SGI.rdsrv.ro
Address: 217.156.101.10

> set type=mx
> yahoo.com
Server: SGI.rdsrv.ro
Address: 217.156.101.10

Non-authoritative answer:
yahoo.com MX preference = 1, mail exchanger = g.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = a.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = b.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = c.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = d.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = e.mx.mail.yahoo.com
yahoo.com MX preference = 1, mail exchanger = f.mx.mail.yahoo.com

yahoo.com nameserver = ns4.yahoo.com
yahoo.com nameserver = ns5.yahoo.com
yahoo.com nameserver = ns6.yahoo.com
yahoo.com nameserver = ns0.yahoo.com
yahoo.com nameserver = ns1.yahoo.com
yahoo.com nameserver = ns2.yahoo.com
yahoo.com nameserver = ns3.yahoo.com
a.mx.mail.yahoo.com internet address = 67.195.168.31
b.mx.mail.yahoo.com internet address = 66.196.97.250
c.mx.mail.yahoo.com internet address = 216.39.53.3
c.mx.mail.yahoo.com internet address = 216.39.53.2
d.mx.mail.yahoo.com internet address = 209.191.88.247
d.mx.mail.yahoo.com internet address = 68.142.202.247
e.mx.mail.yahoo.com internet address = 216.39.53.1
f.mx.mail.yahoo.com internet address = 98.137.54.237
g.mx.mail.yahoo.com internet address = 206.190.53.191
g.mx.mail.yahoo.com internet address = 209.191.118.103
ns6.yahoo.com internet address = 202.43.223.170
ns8.yahoo.com internet address = 202.165.104.22
>

```

All the servers listed now have *MX preference = 1*. I have already said that, in order to succeed when trying to e-mail spoofing, we had to choose those servers with the greatest MX preference. Otherwise, the connection could not be established or could not be stable for a reasonable period of time. In case of Google/Gmail, I had not even one missed attempt, when connecting to their server named *alt4.gmail-smtp-in.l.google.com*, which had the greatest MX number (40), while trying to connect to others (with lower MX) was not always possible or even at all. In case of Yahoo, all their servers have the same MX number and furthermore they probably are the busiest in the world. Therefore, the MX is set to 1 for all of them and that means any of them is prepared at any moment to handle every new message received. Thus, it is more complicated to find the proper one to connect to it. After a lot of testing, I can state now that their server identified by *b.mx.mail.yahoo.com* was the only one that allowed me stable connections, even if there were needed more connection attempts for each successful one.

Before writing this article, I tested intensively some of the major e-mail service providers. The best e-mail spoofing was in case of Gmail. Their servers were not so busy; therefore, it was very easy to establish quick and stable connections. Although some fake e-mails go to the Spam box, their filtering system is perfectible however. In case of Hotmail servers, trying to connect to some of them was a little more difficult, but in case of successful attempt, the connection proved to be stable. I also noticed that the answers to all my *telnet* commands were often very long and filled with unhelpful information.

The most difficult to spoof were Yahoo servers. Their filtering system is quite good; therefore, all fake e-mails go directly to the Spam box, except when the sender's address is included in the recipient's contact list. Then, their servers are busy all the

time, so you have to be patient when trying to connect and have to repeat the procedure many times. In case of success, the connection will never last too much, so you will have to be fast for finishing the entire spoofing process.

### 3. Anti-spoofing protection technologies

If all examples exposed through this article were rather some harmless jokes and did not affect at all the "spoofed" persons involved, neither the security of their computers, nor the security of their confidential and personal data, in case of all those ill-intentioned users addicted to fraudulent activities such as spamming, phishing, cracking, hacking and so on, the damages they can produce are not at all negligible. We are all "bombed" daily with various forms of spam, many of them hiding some malicious little code such as spyware, Trojans, viruses, etc. In addition, we are all informed, quite often lately, that something wrong happened with our bank accounts (even if we never opened any account at the banks mentioned there); in fact, they try to steal our credentials for breaking our real accounts instead. Other e-mails inform us that we are the lucky winners of some lottery we did not buy any ticket for, and so on.

One way to protect against such dangerous e-mails consists of using that mail client software which implements and supports digital signature formats, such as S/MIME (Secure / Multipurpose Internet Mail Extension). It starts with checking the correctness of the sender's address and then the digital signature is checked, too. (*[1], [2], [8], [9] are very good sources to find much more about digital signature.*) This is also a very good anti-phishing protection, because the digital signature is practically generated by the mail client after opening and authenticating the message, on one hand and the entire procedure is based on certain robust and well-certified cryptographic techniques, on the other hand. (*You can find more information about S/MIME by following the links [24], [25] and about designing new efficient protocols in [19], [20].*)

One other technology for fighting against e-mail spoofing is SPF (Sender Policy Framework), an open standard that specifies methods for not allowing faking the sender's address. Any owner of an Internet domain can specify exactly his Security Policy related to mailing activities (into a special section of the DNS attributes), by clearly indicating all the mail exchangers intended to be used for the e-mailing activity within that domain. Thus, the mail server of one recipient - who receives a message pretended to come from that domain - will start with checking the identity of the sender's mail server and, if checking not passed, it will label that message as a fake one and it will treat it accordingly. (*For much more details on SPF follow the link [26].*)

Another one anti-spoofing technique is DKIM (DomainKeys Identified Mail). DKIM adds a header named "DKIM-Signature" that contains a digital signature of the contents (headers and body) of the mail message. The receiving SMTP server then uses the name of the domain from which the mail originated, the string "\_domainkey", and also a selector from the header to perform a DNS lookup. The returned data includes the domain's public key. The receiver can then decrypt the hash value in the header field and, at the same time, recalculate the hash value for the mail message (headers and body) that was received. If the two values match, this cryptographically

proves that the mail originated at the purported domain and has not been corrupted in transit. (*For more details on DKIM, follow the link [27].*) DKIM uses public-key cryptography to allow the signer to electronically sign legitimate e-mails in a way that can be verified by recipients. (*Good information on some public-key cryptography schemes, can be found in [5], [6], [13], [14], [15].*)

One other protection technology - rather related to minimizing the damages some e-mail spoofing can produce - is TFA (Two Factors Authentication), successfully used so for fighting against phishing and so for enhancing the security of the commercial transactions on the Internet. It refers to all those specific mechanisms which involve two components for a complete and safe authentication: the first factor from its name refers to "something you know", usually a user name and a password, while the second factor from its name refers to "something you possess", or "something you have", normally a small electronic device, generating numerical sequences, frequently changed (usually, 30 seconds intervals) by one complex and secret algorithm; furthermore, the device is also protected by a classic PIN code which must be entered before the number sequences generating process to proceed. As you can see, TFA assumes knowing and providing four elements, in order to complete the authentication: two classic ones, which you have to know, of course, the user name and the password, and two other ones, the PIN code and the number sequence, by which you prove you possess that previously mentioned electronic device.

Also related to TFA, I want to point here out the newly developed biometric devices for authentication, a little more expensive so far. In that case, the second factor from the definition refers no more to "something you have", but to "something you are". It is about scanning and identifying some unique (distinct) features for each user: fingerprints, eyes, face geometry, voice inflexions, etc. (*Some very interesting materials related to such relatively new technologies can be found in [3], [4].*)

One commonly used method for a more general protection consists of installing and permanently updating any good anti-virus/anti-spam/anti-spy software suite on our computers. It should be very helpful to fight against Trojans, viruses, key loggers, etc., well hidden through the body of some of the e-mails we receive constantly (or as already infected files attached) and ready to produce great damages, just after a simple mouse click. Using any of such tools does not guarantee us 100% protection, but it is easy to realize how many dangers our computers and our personal data would be exposed to, without having them installed at all.

#### 4. Conclusion

Finally, I want to point out just another opinion, as a conclusion of e-mail spoofing subject. Everyone can create an outbound e-mail message and define all of its characteristics, including *from* field, using a scripting language or, even more simply, a *telnet* session (as shown here, through this article). This particular functionality is due to the inherent insecurity in the SMTP protocol - a standard that was created over 20 years ago and probably its creators never imagined that it would achieve such longevity.

I know that prominent e-mail service providers as Yahoo and Google have already implemented protection technologies such as SPF or DKIM (or maybe both). However, why does not prevent this e-mail spoofing completely? When testing the e-mail spoofing procedure previously described and thus playing jokes with my friends and my colleagues, in most cases they did not treat those mails as fake ones. There were mails sent by me, but with *from* field in header altered for pretending as sent by someone else - though someone included in the recipient's contact list. Most of those fake e-mails passed successfully the spam filtering and other protection systems of the recipients' e-mail service providers and finally appeared to them as ordinary mails. However, when non-real addresses were used as senders' e-mail addresses, I have to admit that, the vast majority of such messages gone directly to the Spam boxes, especially in case of Yahoo.

It seems that if someone sends you a message but does not have anti-spoofing technology in place, the message will still deliver. On the other hand, if someone sends you a message and does have anti-spoofing technology in place, your SMTP server will validate the message and make certain that it comes from where it claims to come from. So, the simple presence of any anti-spoofing technology does not prevent the spoofer/spammer from sending mail, it just provides the recipient a way to verify the mail. Even far from perfect, if such validating methods are not implemented by the recipient, there is nothing he can do to prevent e-mail spoofing.

**Acknowledgement.** This material represents a continuation of the studies performed by graduating the Master in "Artificial Intelligence and Communication Technologies", within the Faculty of Mathematics and Computers Science - Department of Computer Science, from University of Craiova. We want to thank to Prof. Dr. Nicolae Tandareanu - the Dean of the Faculty of Mathematics and Computers Science - and Lect. Dr. Nicolae Constantinescu -within the same Faculty - for their permanent guidance and support, from simple technical discussions and useful advices to providing us excellent reference materials, many of them created by themselves.

## References

- [1] Z-W. Tan, Z-J. Liu, *Cryptanalysis of Threshold Proxy Signature Schemes*, Lecture Notes in Computer Science, vol. 3786, pag. 226-233, 2006
- [2] National Institute of Standards and Technology (NIST), *Digital Signature Standard (DSS)*, FIPS PUB 186-2, 2000
- [3] N. Tandareanu, *Communication by Voice to Interrogate an Inheritance Based Knowledge System*, Research Notes in Artificial Intelligence and Digital Communications, Vol.107, 7th International Conference on Artificial Intelligence and Digital Communications, pag. 1-15, 2007
- [4] N. Tandareanu, M. Ghindeanu, *Image Synthesis from Natural Language Description*, Research Notes in Artificial Intelligence and Digital Communications, Vol.103, 3rd Romanian Conference on Artificial Intelligence and Digital Communications, Craiova, 2003, pag. 82-96
- [5] N. Constantinescu, G. Stephanides, *Identification of parts in identity-based encryption*, Research Notes in Data Security, Wessex Institute of Technology, UK, developed with University of Bergen, Norway, ISBN 1-85312-713-2, 2004
- [6] N. Constantinescu, G. Stephanides, *Secure Key-Exchange*, Recent Advances in Communications and Computer Science, Vol. 7, pag. 162-166, WSEAS Press, Greece, 2003
- [7] M. Bellare, P. Rogaway, *Entity Authentication and Key Distribution*, In Crypto 93, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Germany, September 1994, vol. 773, num. 1, pag. 232-249

- [8] M. Bellare, P. Rogaway, *The Exact Security of Digital Signatures: How to Sign with RSA and Rabin*, In EuroCrypt 96, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Germany", August 1997, vol. 773, num.1, pag. 399-416
- [9] S. Even, O. Goldreich, S. Micali, *On-Line/Off-Line Digital Signatures*, Journal of Cryptology, Springer-Verlag New York, USA, January 1996, vol. 1, pag. 35-67
- [10] O. Goldreich, V. Rosen, *On the Security of Modular Exponentiation with Application to the Construction of Pseudorandom Generators*, Journal of Cryptology, Springer-Verlag New York, USA, February 2003, vol. 16, num. 2, pag. 71-93
- [11] S. Myers, *Efficient Amplification of the Security of Weak Pseudo-Random Function Generators*, Journal of Cryptology, Springer-Verlag New York, USA, Mars 2003, vol. 16, num. 1, pag. 1-24
- [12] J. Dj. Golic, R. Menicocci, *Edit Probability Correlation Attacks on Stop/Go Clocked Keystream Generators*, Journal of Cryptology, Springer-Verlag New York, USA, August 2003, vol. 16, num. 1, pag. 41-68
- [13] S. S. Magliveras, D. R. Stinson, T. van Trung, *New Approaches to Designing Public Key Cryptosystems Using One-Way Functions and Trapdoors in Finite Groups*, Journal of Cryptology, Springer-Verlag New York, December 2002, vol. 15, pag. 285-297
- [14] S. Murphy, M. J.B. Robshaw, *Essential Algebraic Structure within the AES*, Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference Santa Barbara, California, Aug. 18-22, 2002; Proceedings, Springer-Verlag, New York, USA
- [15] W. Diffie, P. C. van Oorschot, M. J. Wiener, *Authentication and Authenticated Key Exchanges*, Journal of Designs, Codes and Cryptography, Kluwer Academic Publisher, Boston, MA, USA, 1992, vol. 2, num. 2
- [16] S. Blake-Willson, D. Johnson, A. Menezes, *Key Agreement Protocols and their Security Analysis*, Proceedings of the 6th IMA International Conference on Cryptography and Coding, Springer-Verlag, Berlin, Germany, 1997, vol. 1355, pag. 30-45
- [17] C. Lim, P. Lee, *A key recovery attack on discrete log-based schemes using a prime order subgroup*, Advances in Cryptology, Springer-Verlag, Lecture Notes in Computer Science, Berlin, Germany, 1997, vol. 1294, pag. 275-288
- [18] Y. Yacobi, *A key distribution paradox*, Journal of Designs, Codes and Cryptography, Kluwer Academic Publisher, Boston, MA, USA, 1991, vol. 537, pag. 268-273
- [19] M. Bellare, P. Rogaway, *Random oracles are practical: A paradigm for designing efficient protocols*, ACM Press, Computer & Communications Security, Washington DC, USA, 1993, vol. 57, pag. 62-73
- [20] B. W. Aiello, S. M. Bellovin, M. Blaze, R. Canetti, J. Ioannidis, A. D. Keromytis, O. Reingold, *Efficient, DoS-Resistant, Secure Key Exchange for Internet Protocols*, Proceedings of the 9th ACM Conference on Computer and Communications security, ACM, Washington DC, USA, 2002, vol. 2, pag. 48-58
- [21] M. Bishop, *Computer Security: Art and Science*, Addison-Wesley, 2004
- [22] <http://www.faqs.org/rfcs/rfc821.html>
- [23] <http://www.faqs.org/rfcs/rfc854.html>
- [24] <http://www.ietf.org/rfc/rfc2633.txt>
- [25] <http://www.smime.org/>
- [26] <http://www.openspf.org/>
- [27] <http://www.dkim.org/>

(Teodor Tîmplaru, Roxana Tîmplaru) DEPARTMENT OF INFORMATICS, UNIVERSITY OF CRAIOVA,  
AL.I. CUZA STREET, NO. 13, CRAIOVA RO-200585, ROMANIA, TEL. & FAX: 40-251412673  
E-mail address: [teotimplaru@yahoo.com](mailto:teotimplaru@yahoo.com)