ACROPOLIS INSTITUTE OF TECHNOLOGY & RESEARCH, INDORE

INDEX

S.no	CONTENTS	Page no
1.	Introduction to technology Undertaken……....	1
2.	Objectives ……..…….....	2
3.	Project undertaken ……..	4
4.	Screenshots of Project and Certificates……..	4
5.	Github Links (Project/certificate/video/copy of report.… …….	10
7.	Conclusion…….. …….	10
8.	References/ Bibilography……..……	10

INTRODUCTION

E-Mail Spoofing

In the context of computers, to spoof one's email address means that the sender is acting as if the email is coming from someone it is not.How someone (or something) sends an email made to look like it comes from somewhere or somewhere it does not, is a little more technical to explain. The spoofing process involves:Spoofing email addresses is rather easy. All a person needs to spoof an email address is an SMTP (Simple Mail Transfer Protocol) server (a server that can send email) and the appropriate email software. Most website hosting services will even provide an SMTP server in their hosting package. It is also possible to send emailfrom your own computer if you load an SMTP server on it, however most ISPs will block port 25 (which is required to send out email).Many of the available free SMTP servers will allow you to show a different "from" address than the actual registered domain that the email is transmitting from. However, to the recipient of said message, they will see that it actually came from the address you specified. Now, there are special checks in place (and more being put into place) to prevent exactly this problem. One is called SPF or "Sender Policy Framework" which was developed by Meng Weng Wong in 2003. Basically, each time an email is sent, the receiving server compares the IP of the origin with the IP listed in the SPF record with the appropriate domain.

OBJECTIVES

Although most well-known for phishing purposes, there are actually several reasons for spoofing sender addresses. These reasons can include:

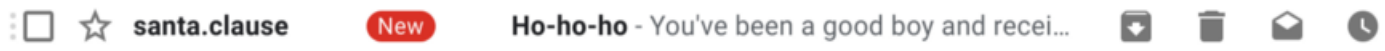
- Hiding the sender's true identity – though if this is the only goal, it can be achieved more easily by registering anonymous mail addresses.
- Avoiding spam block lists. If a sender is spamming, they are bound to be block-listed quickly. A simple solution to this problem is to switch email addresses.
- Pretending to be someone the recipient knows, in order to, for example, ask for sensitive information or access to personal assets.

- Pretending to be from a business the recipient has a relationship with, as means of getting ahold of bank login details or other personal data.
- Tarnishing the image of the assumed sender, a character attack that places the so-called sender in a bad light.
- Sending messages in someone's name can also be used to commit identity theft, for example, by requesting information from the victims financial or healthcare accounts.

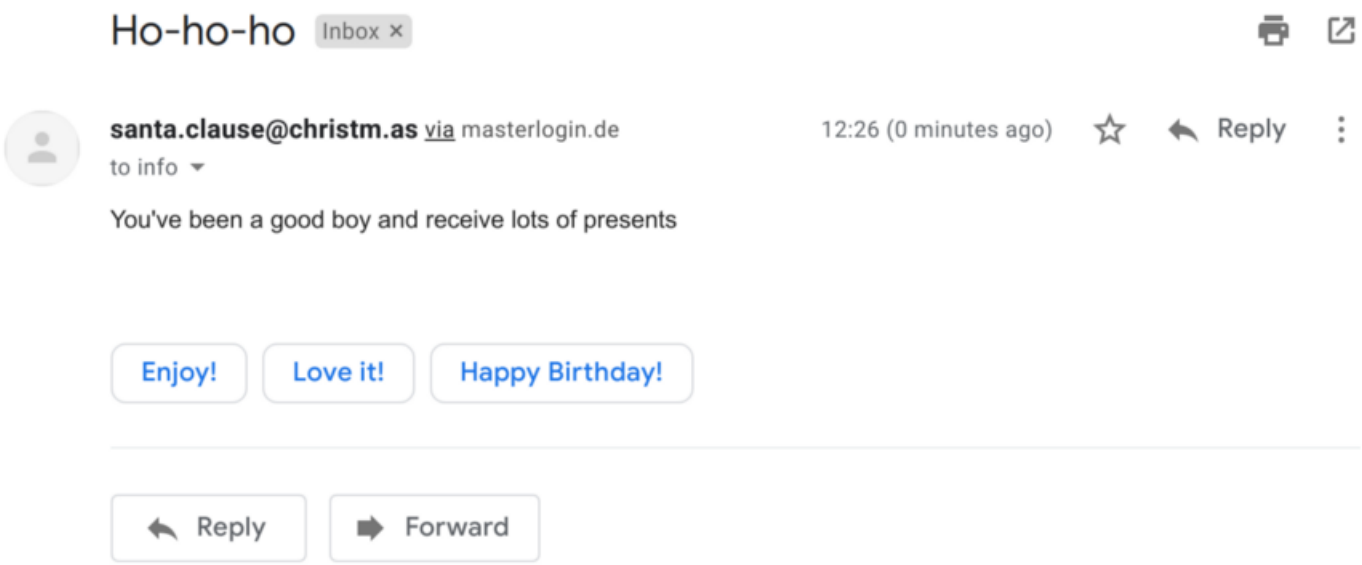
To make it clear that this is super easy and the attackers are not doing rocket science, here is how an email can be sent with Python:

```
import smtplib
from email.message import EmailMessage

msg = EmailMessage()
msg.set_content("You've been a good boy")

msg["Subject"] = "Ho-ho-ho"
msg["From"] = "santa.clause@christm.as"  # The fake sender
msg["To"] = "victim@example.com"  # The actual receiver
# msg.add_header("reply-to", "phishy@phising.com")  # The attackers
address

# Send the message via our own SMTP server.
# On Ubuntu, you need to install sendmail:
#     $ apt-get install sendmail
s = smtplib.SMTP("localhost")
s.send_message(msg)
s.quit()
```

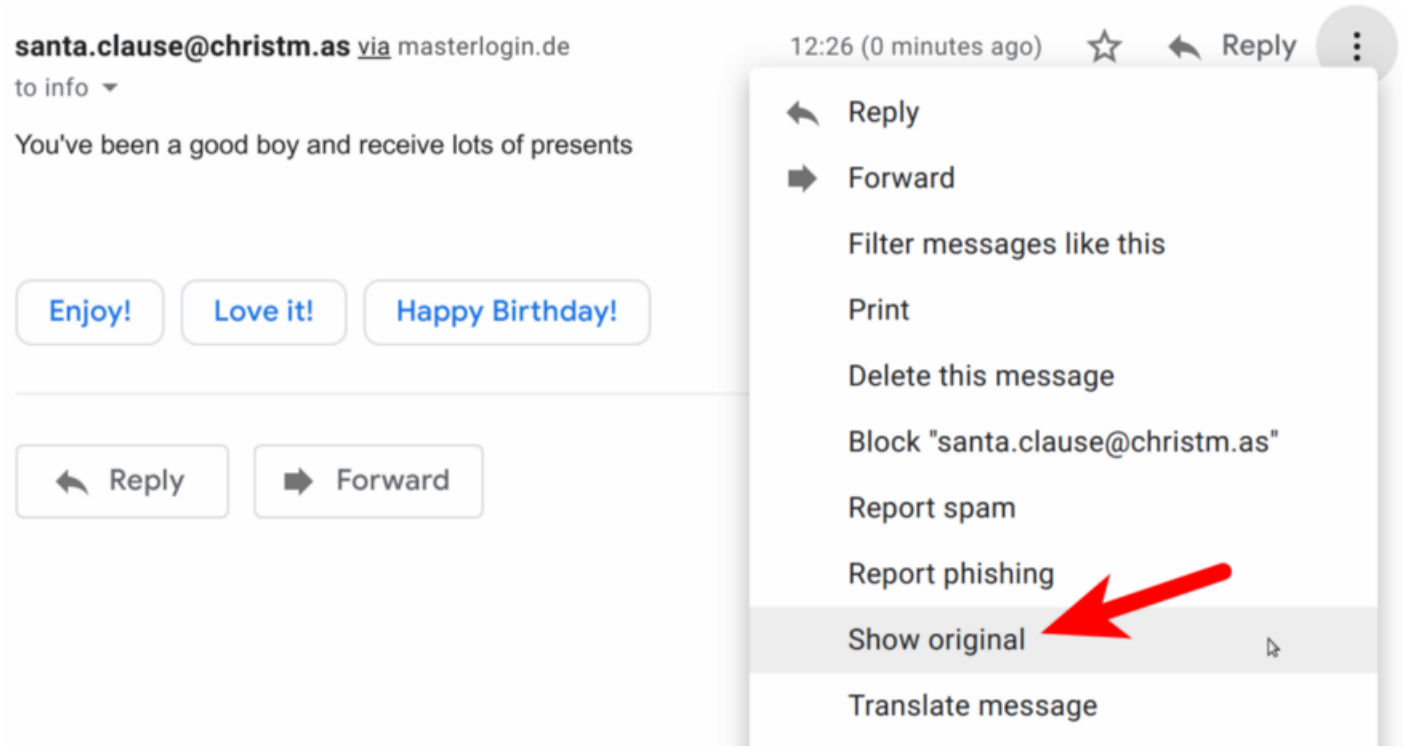
Which then looks like this in Gmail:

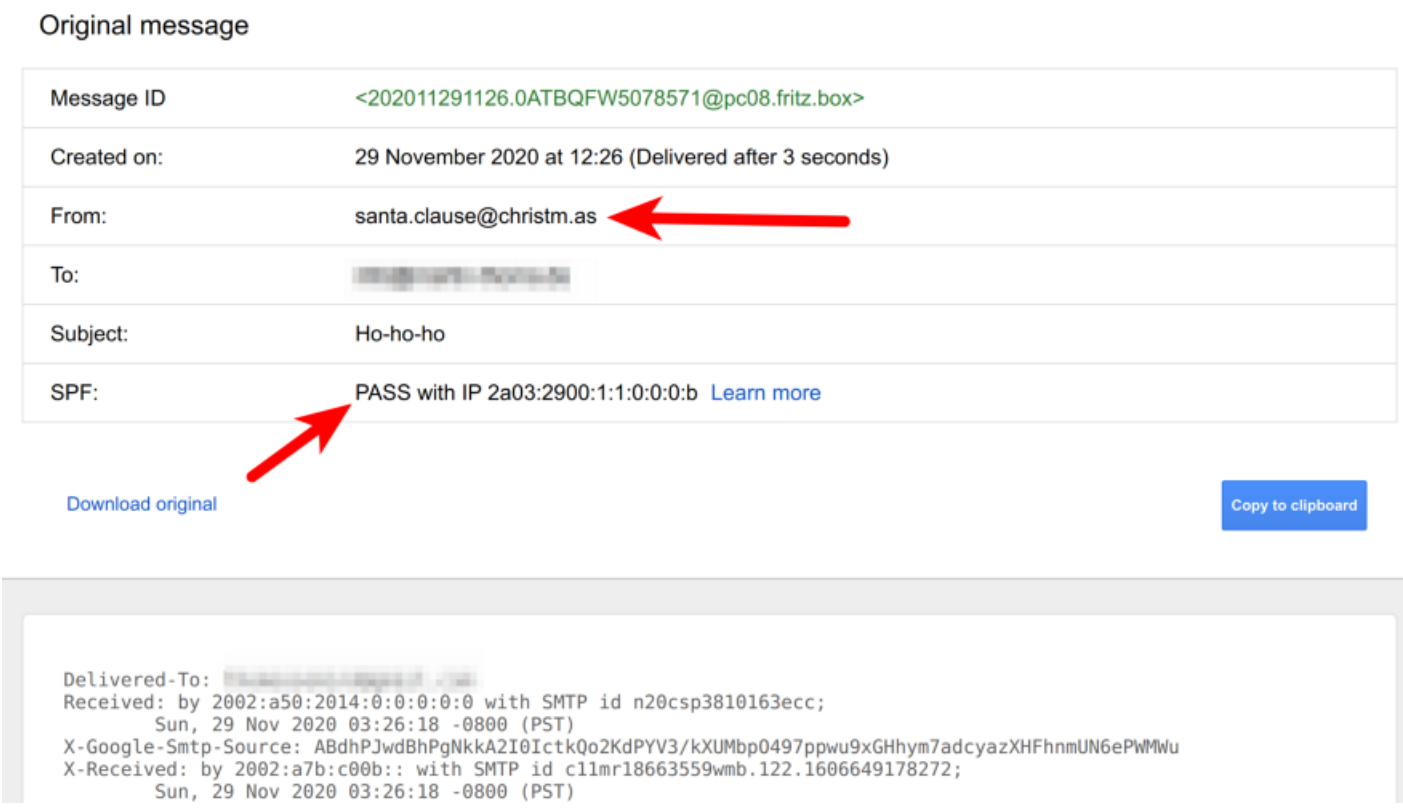


When I click on it, I see this:



Even when I go on the details, I see:






```
Delivered-To: [redacted]
Received: by 2002:a50:2014:0:0:0:0:0 with SMTP id n20csp3810163ecc;
        Sun, 29 Nov 2020 03:26:18 -0800 (PST)
X-Google-Smtp-Source: ABdhPJwdBhPgNkkA2I0IctkQo2KdPYV3/kXUMbpO497ppwu9xGHhym7adcyazXHFhnmUN6ePWMWu
X-Received: by 2002:a7b:c00b:: with SMTP id c11mr18663559wmb.122.1606649178272;
        Sun, 29 Nov 2020 03:26:18 -0800 (PST)
```

The attacker might also put a reply-to in the mail:

```
msg.add_header("reply-to", "phishy@phising.com")
```

The attacker can also add a name to the email address:

```
from email.utils import formataddr

fake_address = "santa.clause@christm.as"
msg["From"] = formataddr(("Santa Clause", fake_address))
```

Interestingly, that triggered Gmails spam detection:

REFERENCE

- www.researchgate.com
- www.security.net
- www.google.com
- www.youtube.com