TryHackMe: Phishing Analysis Fundamentals

me0w4re · Follow
5 min read · May 18, 2023

Listen Share More

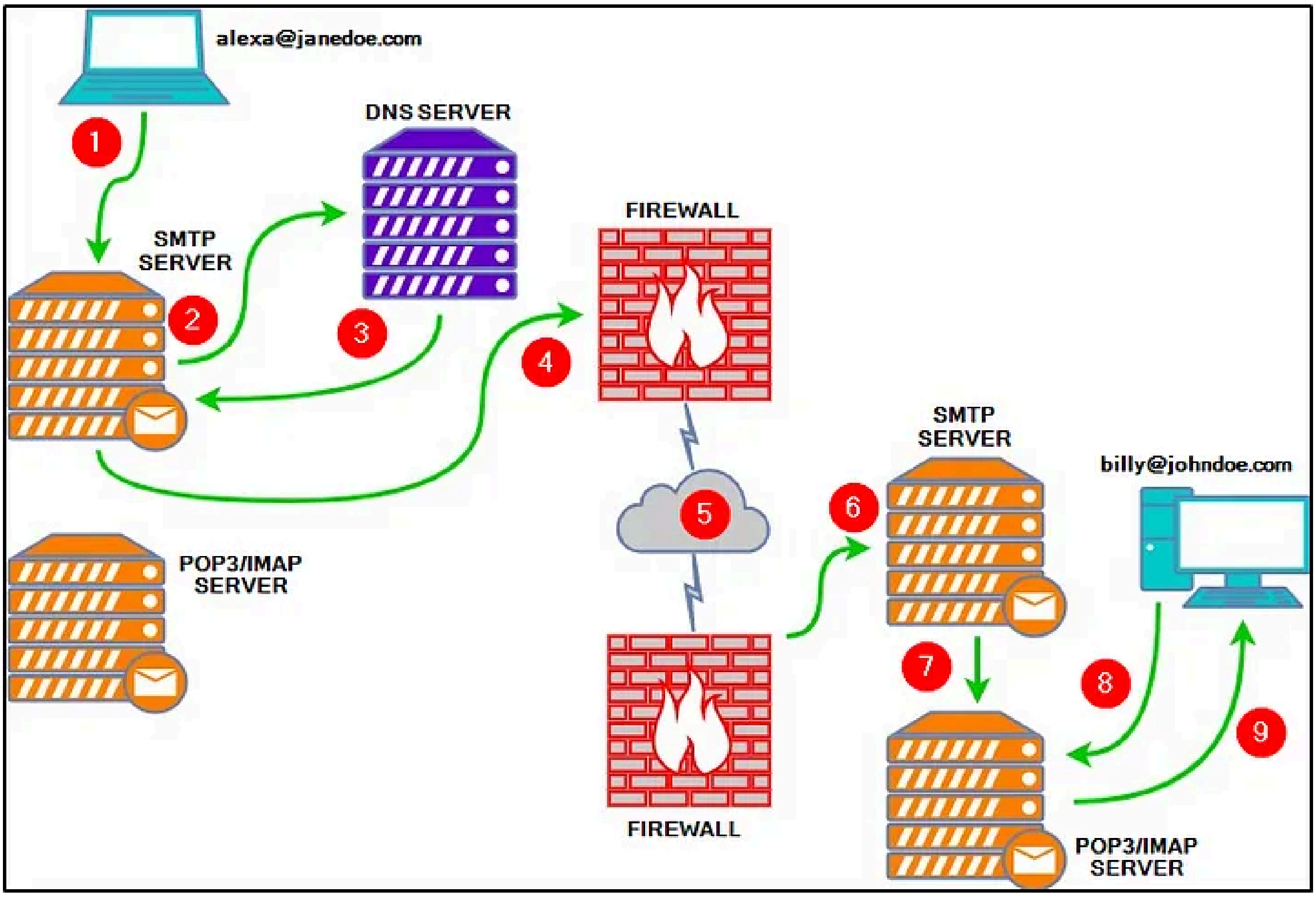
Well, hello there. This is my first blog in (hopefully) a series of blogs where I'll go through my experiences with THM labs. This is my attempt at keeping myself accountable and consistent in keeping up my technical competency as well as exploring other spheres in cybersecurity.

This writeup is meant for educational purposes, and as such I will format my articles to give you the learning points first, then the lab walkthrough.

Now let's dive into our first lab.

Key takeaways

1. There are 3 fundamental protocols to email traffic:
   — — SMTP [465]: To handle the sending of emails
   — — POP3, IMAP [995, 993] : Transferring emails between a client and a mail server



Credits: TryHackMe

2. When analyzing a phishing email, there are a few headers we will be interested in:
— X-Originating-IP: The IP Address this email was sent from
— smtp.mailfrom/header.from: The domain the email was sent from (within Authentication-Results)
— Reply-To: This is important. It tells you who the email will be replied to, which can be different from the sender email address.
— SPF, DKIM, DMARC

3. Email body:
— Content-Type: Tells us what type of file to parse it as
— Content-Disposition: Specifies if it is expected to be displayed 'in-line' or viewed as an attachment.
— Content-Transfer-Encoding: Encoding type (i.e. base64)

4. Types of malicious emails
— Spam/MalSpam
— Phishing
— Spear Phishing: Targeted Phishing

— Whaling: Phishing targeted as C-Suites
— Smishing: SMS-Phishing
— Vishing: Voice-call based Phishing

Now, let's move on to the labs.

Lab Walkthrough

Task 3: Email Delivery

> *This one is pretty straightforward. Answers can be found in lesson material.*
>
> *Q: What port is classified as Secure Transport for SMTP?*
> *A: 465*
> *Q: What port is classified as Secure Transport for IMAP?*
> *A: 993*
> *Q: What port is classified as Secure Transport for POP3?*
> *A: 995*

Task 4: Email HeadersRead the additonal resource in the lesson material.

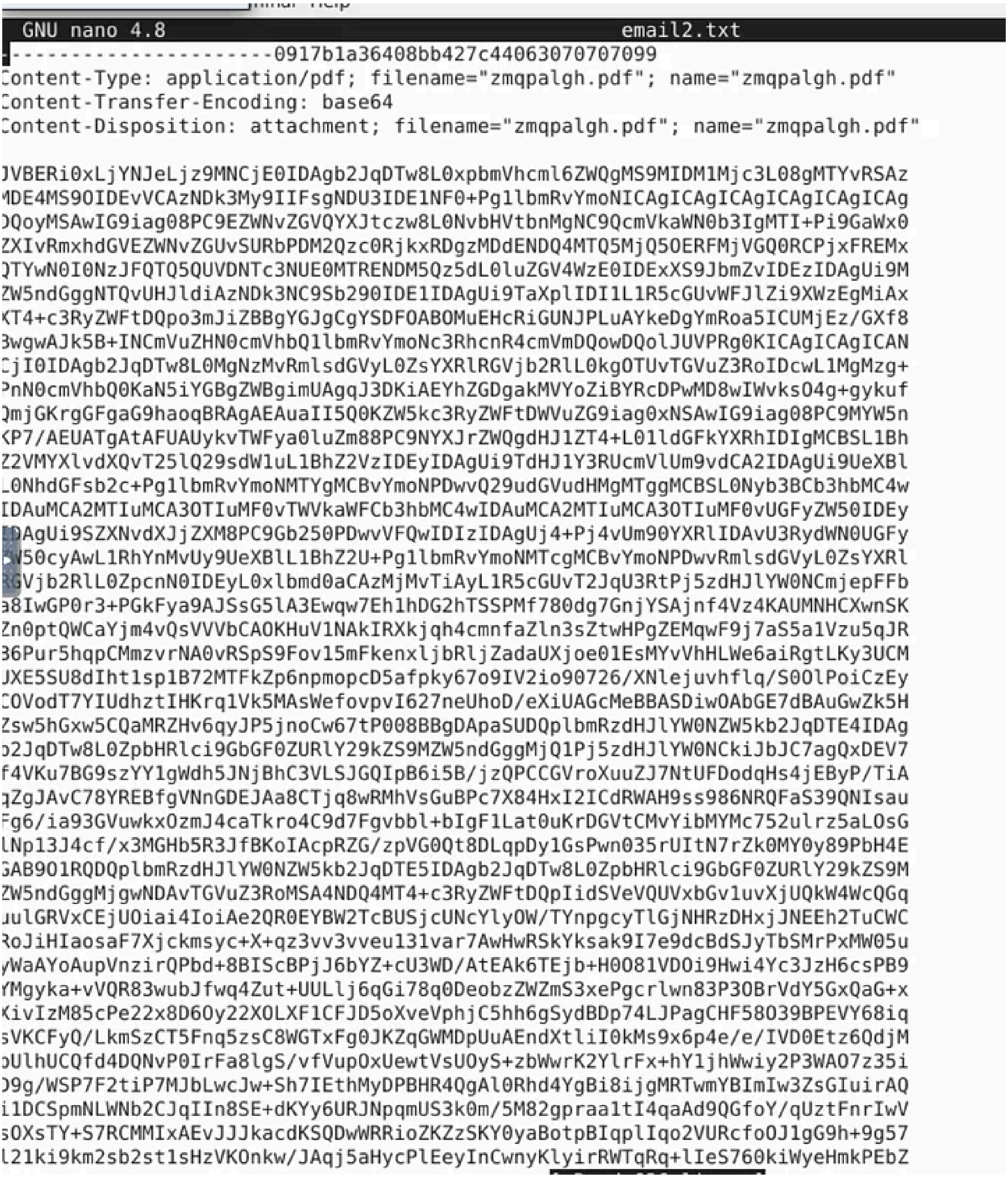
> *Q: What email header is the same as "Reply-to"?*
> *A: Return-Path*
> *Q: Once you find the email sender's IP address, where can you retrieve more information about the IP?*
> *A: http[:]//www.arin[.]net (**Defang your answer before submitting**)*

Task 5: Email Body

> *Q: In the above screenshots, what is the URI of the blocked image?*
> *A: https[:]//i.imgur[.]com/LSWOtDI.png **(Defang)***
> *Q: In the above screenshots, what is the name of the PDF attachment??*
> *A: Payment-updateid.pdf*
>
> *Q: In the attached virtual machine, view the information in email2.txt and reconstruct the PDF using the base64 data. What is the text within the PDF?*
> *A: THM{BENIGN_PDF_ATTACHMENT}*

Task 5.3

Ok Task 5.3 got me confused a little at the start as well.
To start off, you'll see the email body in email2.txt like this



```
-----------------------0917b1a36408bb427c44063070707099
Content-Type: application/pdf; filename="zmqpalgh.pdf"; name="zmqpalgh.pdf"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="zmqpalgh.pdf"; name="zmqpalgh.pdf"

JVBERi0xLjYNJeLjz9MNCjE0IDAgb2JqDTw8L0xpbmVhcml6ZWQgMS9MIDM1Mjc3L08gMTYvRSAz
MDE4MS9OIDEvVCAzNDk3My9IIFsgNDU3IDE1NF0+Pg1lbmRvYmoNICAgICAgICAgICAgICAgICAg
DQoyMSAwIG9iag08PC9EZWNvZGVQYXJtczw8L0NvbHVtbnMgNC9QcmVkaWN0b3IgMTI+Pi9GaWx0
ZXIvRmxhdGVEZWNvZGUvSURbPDM2Qzc0RjkxRDgzMDdENDQ4MTQ5MjQ5OERFMjVGQ0RCPjxFREMx
QTYwN0I0NzJFQTQ5QUVDNTc3NUE0MTRENDM5Qz5dL0luZGV4WzE0IDExXS9JbmZvIDEzIDAgUi9M
ZW5ndGggNTQvUHJldiAzNDk3NC9Sb290IDE1IDAgUi9TaXplIDI1L1R5cGUvWFJlZi9XWzEgMiAx
XT4+c3RyZWFtDQpo3mJiZBBgYGJgCgYSDFOABOMuEHcRiGUNJPLuAYkeDgYmRoa5ICUMjEz/GXf8
BwgwAJk5B+INCmVuZHN0cmVhbQ1lbmRvYmoNc3RhcnR4cmVmDQowDQolJUVPRg0KICAgICAgICAN
CjI0IDAgb2JqDTw8L0MgNzMvRmlsdGVyL0ZsYXRlRGVjb2RlL0kgOTUvTGVuZ3RoIDcwL1MgMzg+
PnN0cmVhbQ0KaN5iYGBgZWBgimUAgqJ3DKiAEYhZGDgakMVYoZiBYRcDPwMD8wIWvksO4g+gykuf
QmjGKrgGFgaG9haoqBRAgAEAuaII5Q0KZW5kc3RyZWFtDWVuZG9iag0xNSAwIG9iag08PC9MYW5n
KP7/AEUATgAtAFUAUykvTWFya0luZm88PC9NYXJrZWQgdHJ1ZT4+L01ldGFkYXRhIDIgMCBSL1Bh
Z2VMYXlvdXQvT25lQ29sdW1uL1BhZ2VzIDEyIDAgUi9TdHJ1Y3RUcmVlUm9vdCA2IDAgUi9UeXBl
L0NhdGFsb2c+Pg1lbmRvYmoNMTYgMCBvYmoNPDwvQ29udGVudHMgMTggMCBSL0Nyb3BCb3hbMC4w
IDAuMCA2MTIuMCA3OTIuMF0vTWVkaWFCb3hbMC4wIDAuMCA2MTIuMCA3OTIuMF0vUGFyZW50IDEy
IDAgUi9SZXNvdXJjZXM8PC9Gb250PDwvVFQwIDIzIDAgUj4+Pj4vUm90YXRlIDAvU3RydWN0UGFy
ZW50cyAwL1RhYnMvUy9UeXBlL1BhZ2U+Pg1lbmRvYmoNMTcgMCBvYmoNPDwvRmlsdGVyL0ZsYXRl
RGVjb2RlL0ZpcnN0IDEyL0xlbmd0aCAzMjMvTiAyL1R5cGUvT2JqU3RtPj5zdHJlYW0NCmjepFFb
a8IwGP0r3+PGkFya9AJSsG5lA3Ewqw7Eh1hDG2hTSSPMf780dg7GnjYSAjnf4Vz4KAUMNHCXwnSK
Zn0ptQWCaYjm4vQsVVVbCAOKHuV1NAkIRXkjqh4cmnfaZln3sZtwHPgZEMqwF9j7aS5a1Vzu5qJR
B6Pur5hqpCMmzvrNA0vRSpS9Fov15mFkenxljbRljZadaUXjoe01EsMYvVhHLWe6aiRgtLKy3UCM
UXE5SU8dIht1sp1B72MTFkZp6npmopcD5afpky67o9IV2io90726/XNlejuvhflq/S0OlPoiCzEy
COVodT7YIUdhztIHKrq1Vk5MAsWefovpvI627neUhoD/eXiUAGcMeBBASDiwOAbGE7dBAuGwZk5H
Zsw5hGxw5CQaMRZHv6qyJP5jnoCw67tP008BBgDApaSUDQplbmRzdHJlYW0NZW5kb2JqDTE4IDAg
b2JqDTw8L0ZpbHRlci9GbGF0ZURlY29kZS9MZW5ndGggMjQ1Pj5zdHJlYW0NCkiJbJC7agQxDEV7
f4VKu7BG9szYY1gWdh5JNjBhC3VLSJGQIpB6i5B/jzQPCCGVroXuuZJ7NtUFDodqHs4jEByP/TiA
qZgJAvC78YREBfgVNnGDEJAa8CTjq8wRMhVsGuBPc7X84HxI2ICdRWAH9ss986NRQFaS39QNIsau
Fg6/ia93GVuwkxOzmJ4caTkro4C9d7Fgvbbl+bIgF1Lat0uKrDGVtCMvYibMYMc752ulrz5aLOsG
lNp13J4cf/x3MGHb5R3JfBKoIAcpRZG/zpVG0Qt8DLqpDy1GsPwn035rUItN7rZk0MY0y89PbH4E
GAB9O1RQDQplbmRzdHJlYW0NZW5kb2JqDTE5IDAgb2JqDTw8L0ZpbHRlci9GbGF0ZURlY29kZS9M
ZW5ndGggMjgwNDAvTGVuZ3RoMSA4NDQ4MT4+c3RyZWFtDQpIidSVeVQUVxbGv1uvXjUQkW4WcQGq
uulGRVxCEjUOiai4IoiAe2QR0EYBW2TcBUSjcUNcYlyOW/TYnpgcyTlGjNHRzDHxjJNEEh2TuCWC
RoJiHIaosaF7Xjckmsyc+X+qz3vv3vveu131var7AwHwRSkYksak9I7e9dcBdSJyTbSMrPxMW05u
yWaAYoAupVnzirQPbd+8BIScBPjJ6bYZ+cU3WD/AtEAk6TEjb+H0O81VDOi9Hwi4Yc3JzH6csPB9
YMgyka+vVQR83wubJfwq4Zut+UULj6qGi78q0DeobzZWZmS3xePgcrlwn83P3OBrVdY5GxQaG+x
KivIzM85cPe22x8D6Oy22XOLXF1CFJD5oXveVphjC5hh6gSydBDp74LJPagCHF58O39BPEVY68iq
sVKCFyQ/LkmSzCT5Fnq5zsC8WGTxFg0JKZqGWMDpUuAEndXtliI0kMs9x6p4e/e/IVD0Etz6QdjM
bUlhUCQfd4DQNvP0IrFa8lgS/vfVupOxUewtVsUOyS+zbWwrK2YlrFx+hY1jhWwiy2P3WAO7z35i
D9g/WSP7F2tiP7MJbLwcJw+Sh7IEthMyDPBHR4QgAl0Rhd4YgBi8ijgMRTwmYBImIw3ZsGIuirAQ
i1DCSpmNLWNb2CJqIIn8SE+dKYy6URJNpqmUS3k0m/5M82gpraa1tI4qaAd9QGfoY/qUztFnrIwV
sOXsTY+S7RCMMIxAEvJJJkacdKSQDwWRRioZKZzSKY0yaBotpBIqplIqo2VURcfoOJ1gG9h+9g57
l21ki9km2sb2st1sHzVKOnkw/JAqj5aHycPlEeyInCwnyKlyirRWTqRq+lIeS760kiWyeHmkPEbZ
```

Notice at the start and end of the files, there are delimiters that look something like ———————————-*0917b1a36408bb427c44063070707099*

These are called MIME boundaries, and are used to clearly define boundaries between parts of an email body, be it to separate multi-part email bodies, or in this case to define the payload of the attachment.

Now, we have to do 2 things: First, we want to only decode (base64 as defined in the email body) the PDF contents. And secondly, we will want to remove any unnecessary 'newline' characters. We can do the following:

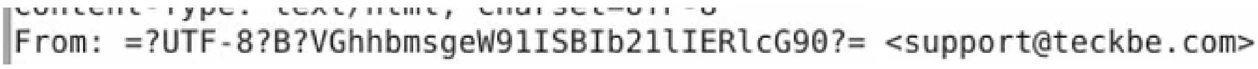
1. cp email2.txt email2.tmp
2. nano email2.tmp
   Now remove all the unnecessary contents. You should only be left with the base64 encoded portion. Save the file.
3. tr -d '\n' < email2.tmp | base64 -d > email2.pdf
   This basically trims away all the 'newline' characters to get one long string of the base64 encoded contents, decode them using the 'base64' command, and then output the results into email2.pdf
4. Now, using the File Explorer, open email2.pdf and voila the answer.

Task 6: Types of Phishing

Task 6.1
Q: What trusted entity is this email masquerading as?
A: Home Depot

There are many ways to do this, but I prefer to stick to the shell version as much as I can.

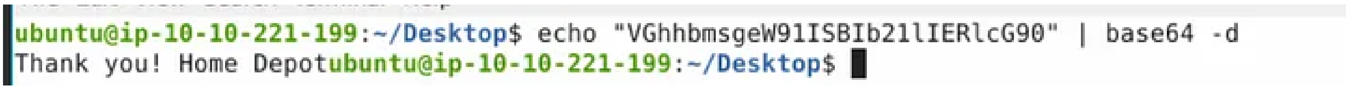
1. cat email3.eml
2. find the From header



```
Content-Type: text/html; charset=UTF-8
From: =?UTF-8?B?VGhhbmsgeW91ISBIb21lIERlcG90?= <support@teckbe.com>
```

To understand this, you can follow this format from RFC 2047:
encoded-word = "=?" charset "?" encoding "?" encoded-text "?="

This actually tells us that the email header is *encoded in UTF-8* and the B refers to Base64. Now, you want to take the encoded-text and decode it.



```
ubuntu@ip-10-10-221-199:~/Desktop$ echo "VGhhbmsgeW91ISBIb21lIERlcG90" | base64 -d
Thank you! Home Depotubuntu@ip-10-10-221-199:~/Desktop$ 
```

Task 6.2

Q: What is the sender’s email?

A: support@teckbe.com

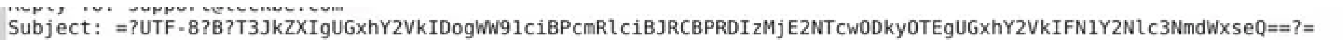
Refer to the earlier screenshot with the encoded text. It shows the sender email next to the encoded display text.

Task 6.3

Q:What is the subject line?

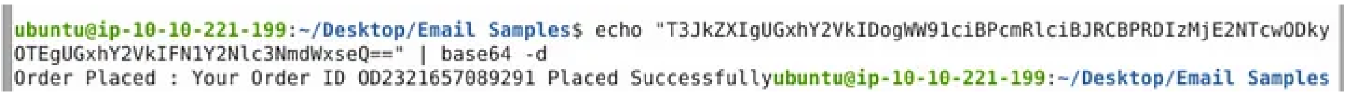
A: Order Placed : Your Order ID OD2321657089291 Placed Successfully

To do this, refer to the raw email and look for the Subject Header.



```
Subject: =?UTF-8?B?T3JkZXIgUGxhY2VkIDogWW91ciBPcmRlciBJRCBPRDIzMjE2NTcwODkyOTEgUGxhY2VkIFN1Y2Nlc3NmdWxseQ==?=
```

As you can tell, it is encoded in the same format as earlier (RFC 2047). Decoding this string as we did just now will bring us to the answer.



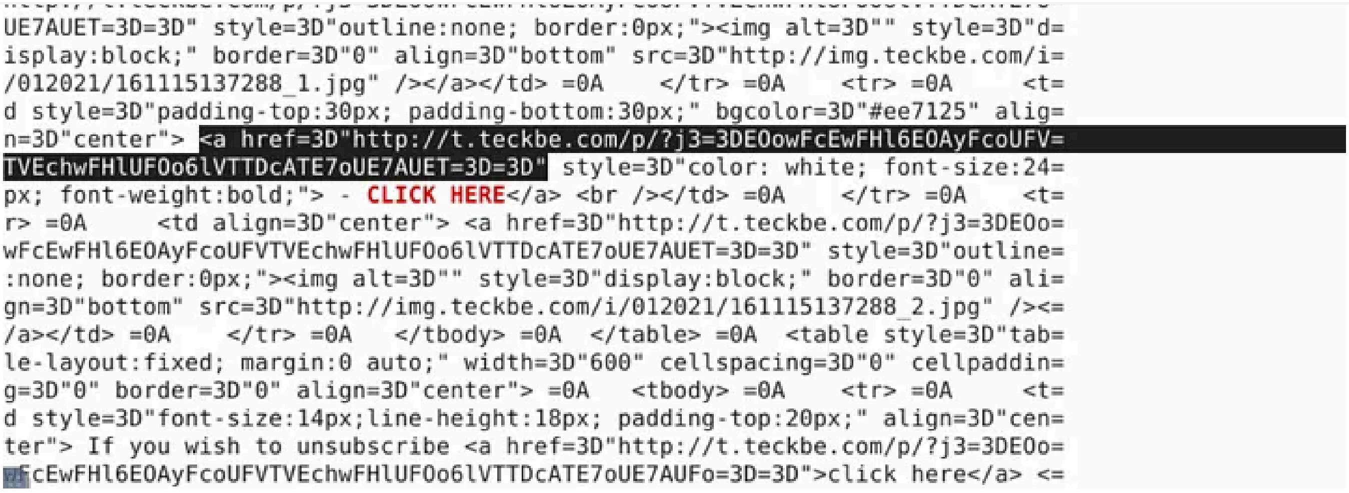
```
ubuntu@ip-10-10-221-199:~/Desktop/Email Samples$ echo "T3JkZXIgUGxhY2VkIDogWW91ciBPcmRlciBJRCBPRDIzMjE2NTcwODky
OTEgUGxhY2VkIFN1Y2Nlc3NmdWxseQ==" | base64 -d
Order Placed : Your Order ID OD2321657089291 Placed Successfullyubuntu@ip-10-10-221-199:~/Desktop/Email Samples
```

Task 6.4

Q: What is the URL link for — CLICK HERE? (Enter the defanged URL)

A: hxxp[://]t[.]teckbe[.]com/p/?j3=EOowFcEwFHl6EOAyFcoUFV=TVEchwFHlUFOo6lVTTDcATE7oUE7AUET==

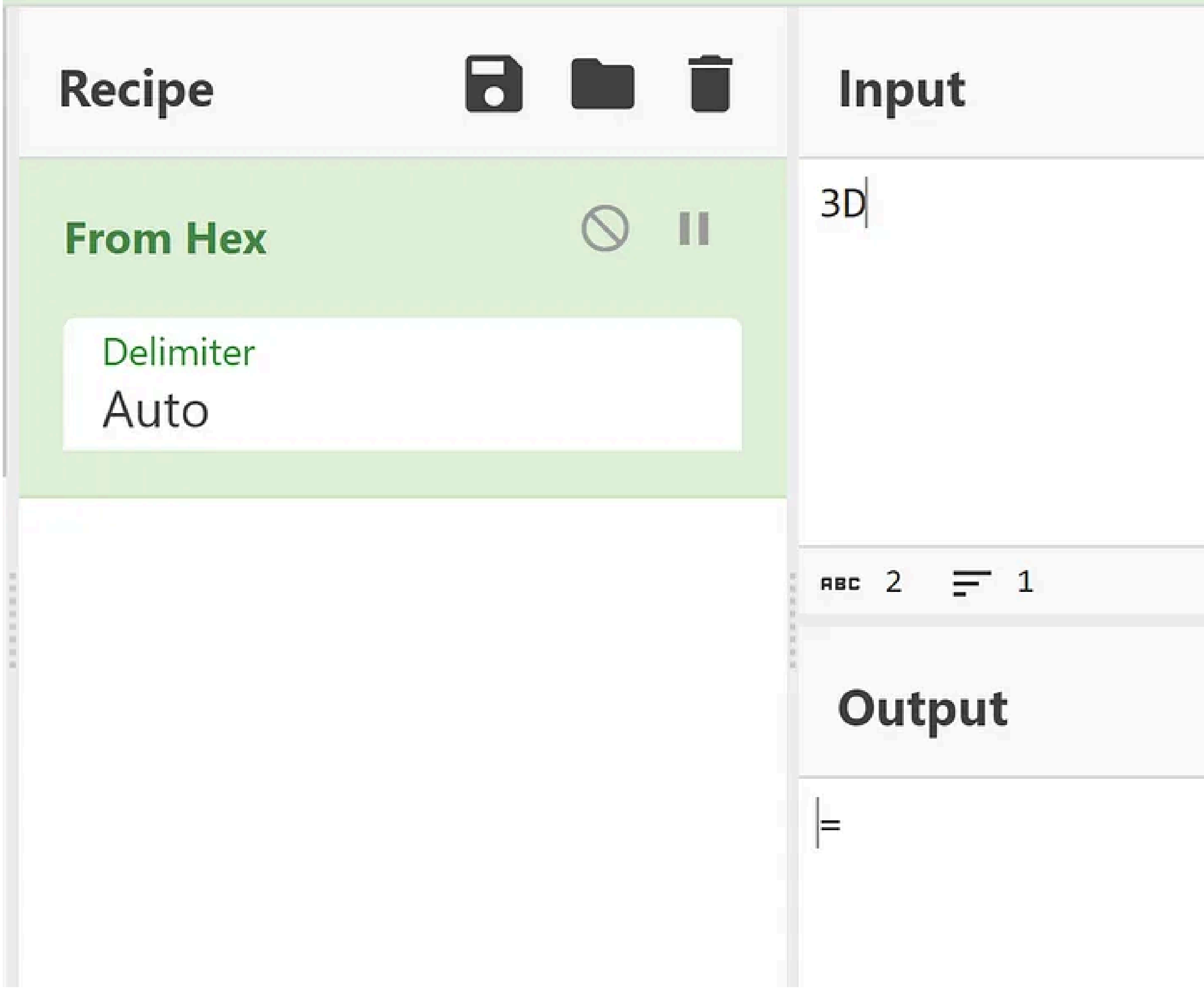
First, we want to look for the CLICK HERE string within the raw email.
You can do that by doing
grep “CLICK HERE” email3.eml -A 10 -B 10



```
UE7AUET=3D=3D" style=3D"outline:none; border:0px;"><img alt=3D"" style=3D"d=
isplay:block;" border=3D"0" align=3D"bottom" src=3D"http://img.teckbe.com/i=
/012021/161115137288_1.jpg" /></a></td> =0A    </tr> =0A    <tr> =0A    <t=
d style=3D"padding-top:30px; padding-bottom:30px;" bgcolor=3D"#ee7125" alig=
n=3D"center"> <a href=3D"http://t.teckbe.com/p/?j3=3DEOowFcEwFHl6EOAyFcoUFV=
TVEchwFHlUFOo6lVTTDcATE7oUE7AUET=3D=3D" style=3D"color: white; font-size:24=
px; font-weight:bold;"> - CLICK HERE</a> <br /></td> =0A    </tr> =0A    <t=
r> =0A     <td align=3D"center"> <a href=3D"http://t.teckbe.com/p/?j3=3DEOo=
wFcEwFHl6EOAyFcoUFVTVEchwFHlUFOo6lVTTDcATE7oUE7AUET=3D=3D" style=3D"outline=
:none; border:0px;"><img alt=3D"" style=3D"display:block;" border=3D"0" ali=
gn=3D"bottom" src=3D"http://img.teckbe.com/i/012021/161115137288_2.jpg" /><=
/a></td> =0A    </tr> =0A   </tbody> =0A  </table> =0A  <table style=3D"tab=
le-layout:fixed; margin:0 auto;" width=3D"600" cellspacing=3D"0" cellpaddin=
g=3D"0" border=3D"0" align=3D"center"> =0A   <tbody> =0A    <tr> =0A     <t=
d style=3D"font-size:14px;line-height:18px; padding-top:20px;" align=3D"cen=
ter"> If you wish to unsubscribe <a href=3D"http://t.teckbe.com/p/?j3=3DEOo=
wFcEwFHl6EOAyFcoUFVTVEchwFHlUFOo6lVTTDcATE7oUE7AUFo=3D=3D">click here</a> <=
```

Notice the </a> behind the string “CLICK HERE”? This tells us that the href hyperlink tag we are looking for is **before** the “CLICK HERE” string, as highlighted above.

Now, you can’t just copy and paste this string as it is poorly formatted. Notice how end of each line in the body is delimited with a “=”? You will have to strip that first. Now, notice that there are randomly insert “=3D” characters all along the string? Well, you can read more about it here, but it’s a formatting standard called “quoted-printable”, where you “3D” is the hex value of the actual character. Which coincidentally happens to be the character “=”.



Delimiter
Auto

```
3D
```

```
=
```

Conclusion

And with that, that’s the end of our first lab. Hope you enjoyed it and found this article useful. I hope to keep this up and look forward to putting out more content. Cheers!

Tryhackme Walkthrough

Phishing Email

Follow

Written by me0w4re

13 Followers · 26 Following

No responses yet

What are your thoughts?

Respond

More from me0w4re

me0w4re

TryHackMe: Linux Forensics

Well, it’s been awhile since my previous THM series. Life has been busy. But here’s the latest drop!

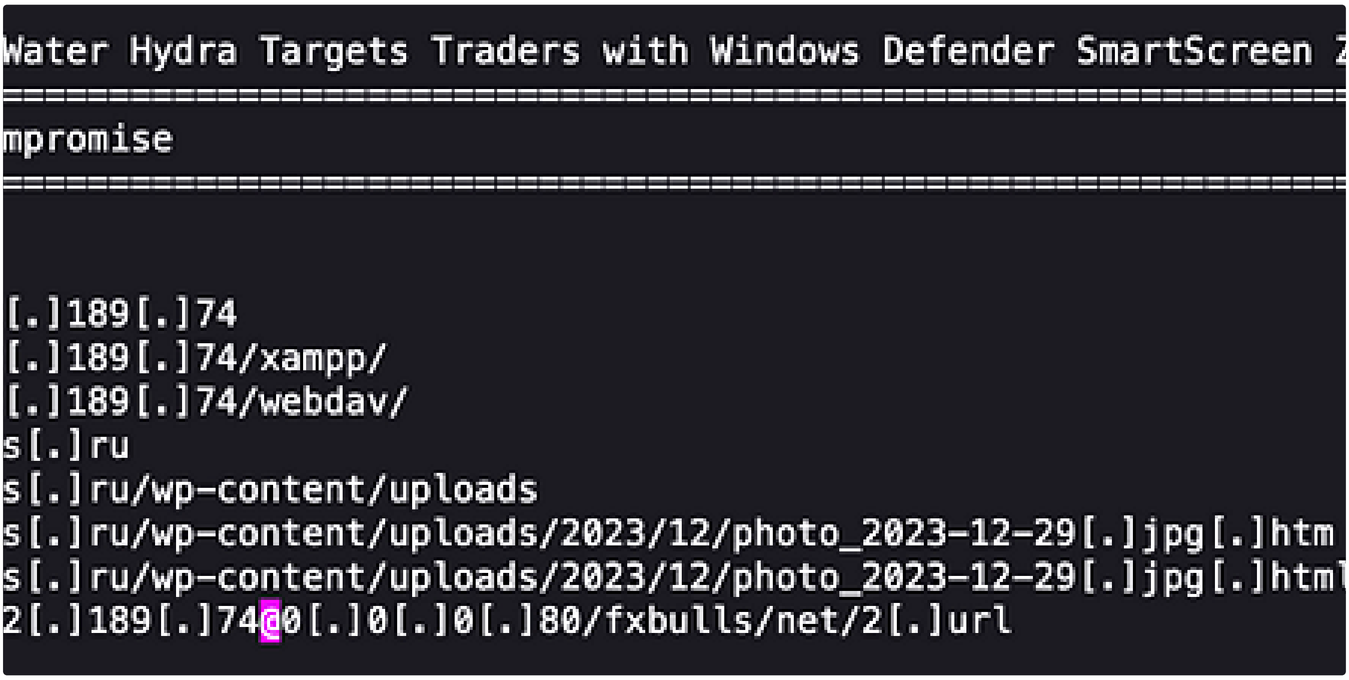
Nov 10, 2023 1

me0w4re

SIEM Engineering Ideas: Elasticsearch Enrich Processor

The Problem

Dec 13, 2023 12



```
Water Hydra Targets Traders with Windows Defender SmartScreen Z
=================================================================
mpromise
=================================================================



[.]189[.]74
[.]189[.]74/xampp/
[.]189[.]74/webdav/
s[.]ru
s[.]ru/wp-content/uploads
s[.]ru/wp-content/uploads/2023/12/photo_2023-12-29[.]jpg[.]htm
s[.]ru/wp-content/uploads/2023/12/photo_2023-12-29[.]jpg[.]html
2[.]189[.]74@0[.]0[.]0[.]80/fxbulls/net/2[.]url
```

me0w4re

Threat Hunting: Odd looking '@' symbol in URLs

While reading Trend Micro's report on Water Hydra (aka DarkCasino) (you can find the article here), something struck out to me in the list…

Mar 1, 2024 1

See all from me0w4re

Recommended from Medium

Advent of Cyber 2024
Dive into the wonderful world of cyber security by engaging in festive beginner-friendly exercises every day in the lead-up to Christmas!
If you'd like to WPA, press the star key!
Day 11
Answers
cyberw1ng.medium.com

In System Weakness by Karthikeyan Nagaraj

Advent of Cyber 2024 [Day 11] Writeup with Answers | TryHackMe Walkthrough

If you'd like to WPA, press the star key!

Dec 11, 2024 855 1

The Story
14
"It's the Mayor" said the Glitch, he said it while sighing,
"The people of Wareville, their browsing he's spying!"
"That sounds like him", McSkidy then said,
"Back to work then", while scratching her head.

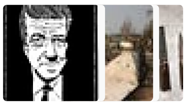
In InfoSec Write-ups by Nanda Siddhardha

TryHackme's Advent of Cyber 2024 — Day 14 Writeup

Day 14: Even if we're horribly mismanaged, there'll be no sad faces on SOC-mas!

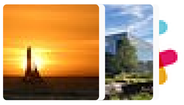
Dec 14, 2024

Lists



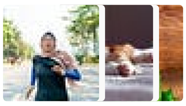
Staff picks

800 stories · 1569 saves



Stories to Help You Level-Up at Work

19 stories · 920 saves



Self-Improvement 101

20 stories · 3227 saves

Productivity 101

20 stories · 2726 saves

Simple CTF

Beginner level ctf

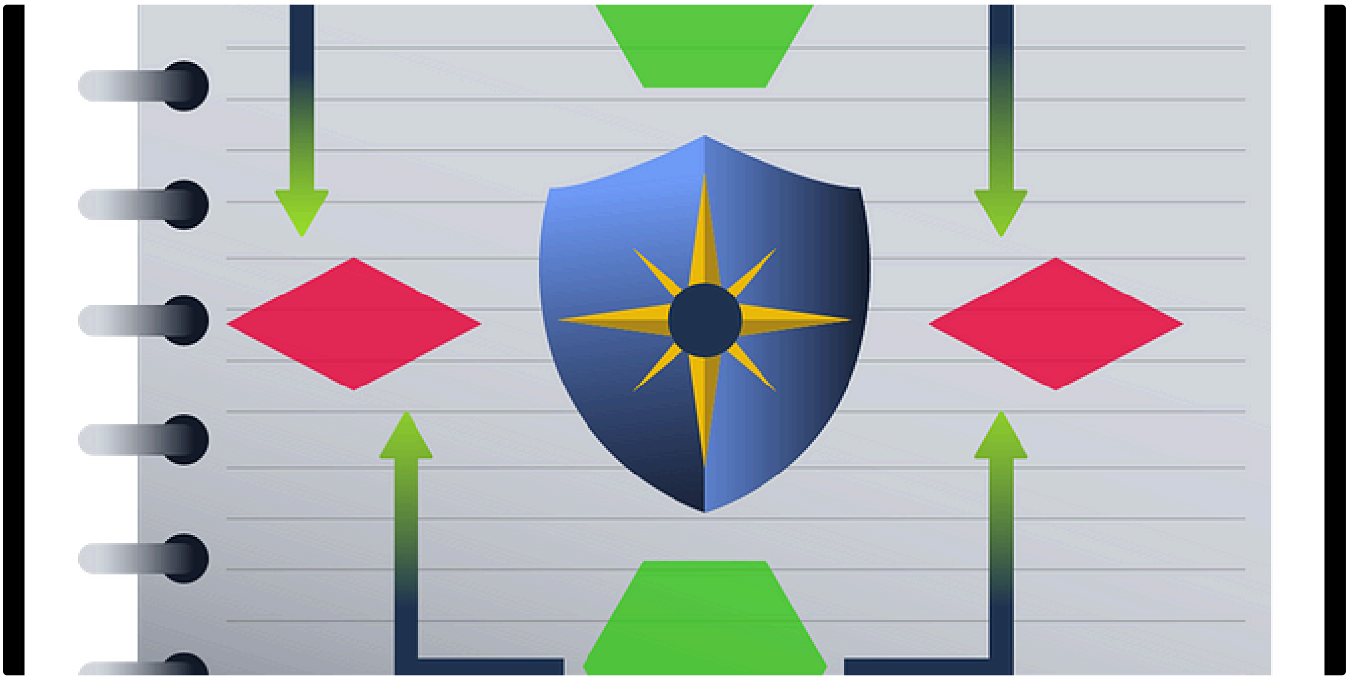
Easy 0 min

In InfoSec Write-ups by Momal Naz

TryHackMe | Simple CTF | Walkthrough | By HexaHunter

Step-by-step guide to solving the Simple CTF room for beginners.

Sep 9, 2024 5



Sunny Singh Verma [SuNnY]

IR Playbooks TryHackMe Walkthrough Writeup THM |—SuNnY

Kudos to The Creators of this Room :

Sep 13, 2024 100 1

IritT

Phishing Analysis Fundamentals

Learn all the components that make up an email.

Nov 26, 2024 👏 1

0xMan1sh 🚀

Unattended Writeup TryHackMe || Medium Level || Detailed Walkthrough 🔥

Use your Windows forensics knowledge to investigate an incident.

Dec 17, 2024 50 1

See more recommendations