OverTheWire – Leviathan Lab Walkthrough (Levels 0–7) Proof of Concept Report

Team Contributors

- **Piyush Babele – 386**
- **Sristi Dutta – 387**

This report documents the detailed step-by-step solutions of the OverTheWire Leviathan wargame (Levels 0–7). The lab was completed collaboratively by both team members, working together on each level from start to finish. As beginners, we often referred to online resources for guidance, especially to understand Linux commands, their functions, and practical usage. The report not only reflects our progress in solving the challenges but also highlights how teamwork, exploration, and external references helped us strengthen our foundation in Linux fundamentals, binary exploitation, and privilege escalation techniques.

Level 0 → Level 1

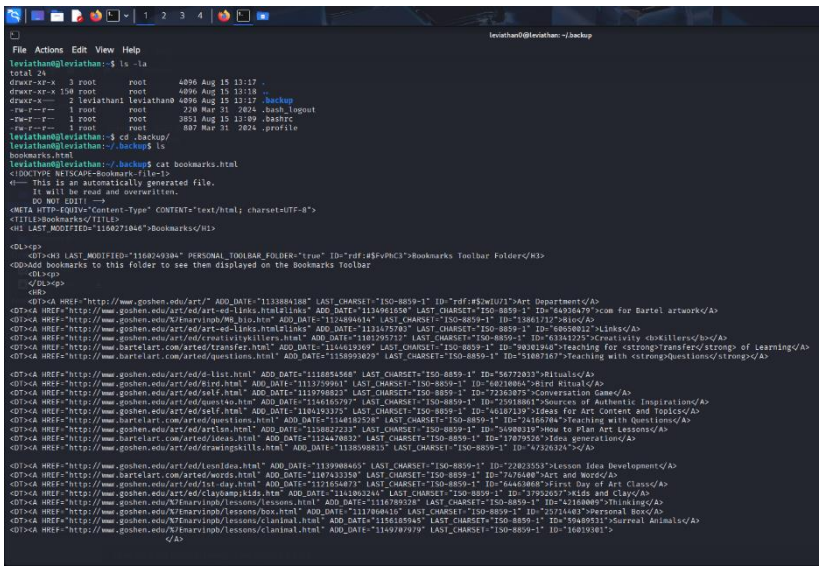
Tools Used: `ls`, `cat`, `grep`

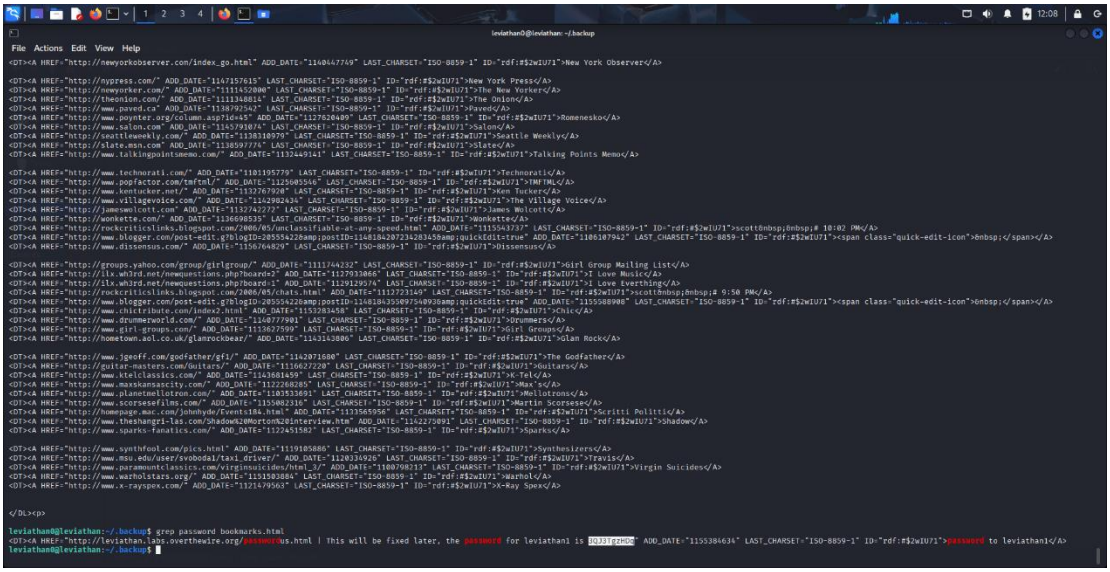
Objective: Extract password from backup HTML file.

Steps Followed:

1. Logged in as leviathan0 with default password.
2. Ran `ls -la` and found hidden `.backup` file.
3. Viewed contents with `cat bookmark.html`, which pointed to bookmark.html.
4. Searched for the keyword using `grep password bookmark.html`.
5. Retrieved the password for leviathan1.

Conclusion: Learned to inspect hidden files and extract sensitive data from backups.





Level 1 → Level 2

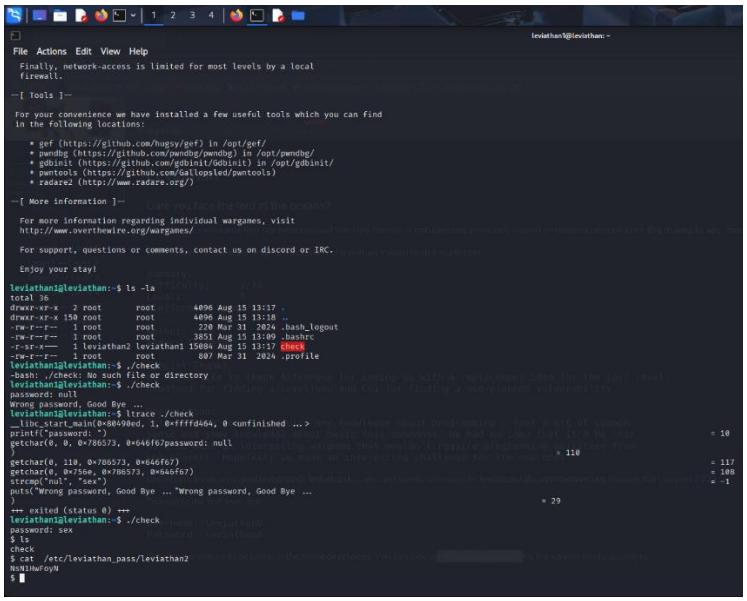
Tools Used: `ltrace`, `./binary`

Objective: Analyze binary execution to uncover a hardcoded password.

Steps Followed:

1. Found a binary named `check`.
2. Running it without arguments asked for a password.
3. Executed with `ltrace ./check` to trace function calls.
4. Observed a `strcmp()` comparison with the string **`sex`**.
5. Ran `./check sex` and successfully obtained the password.

Conclusion: Learned how to use `ltrace` to reveal hidden binary logic.



Level 2 → Level 3

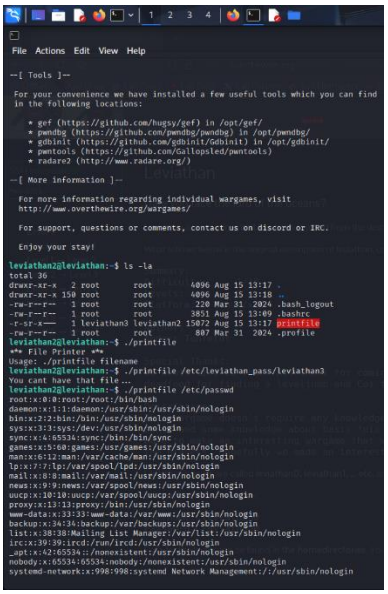
Tools Used: `ltrace`, command injection

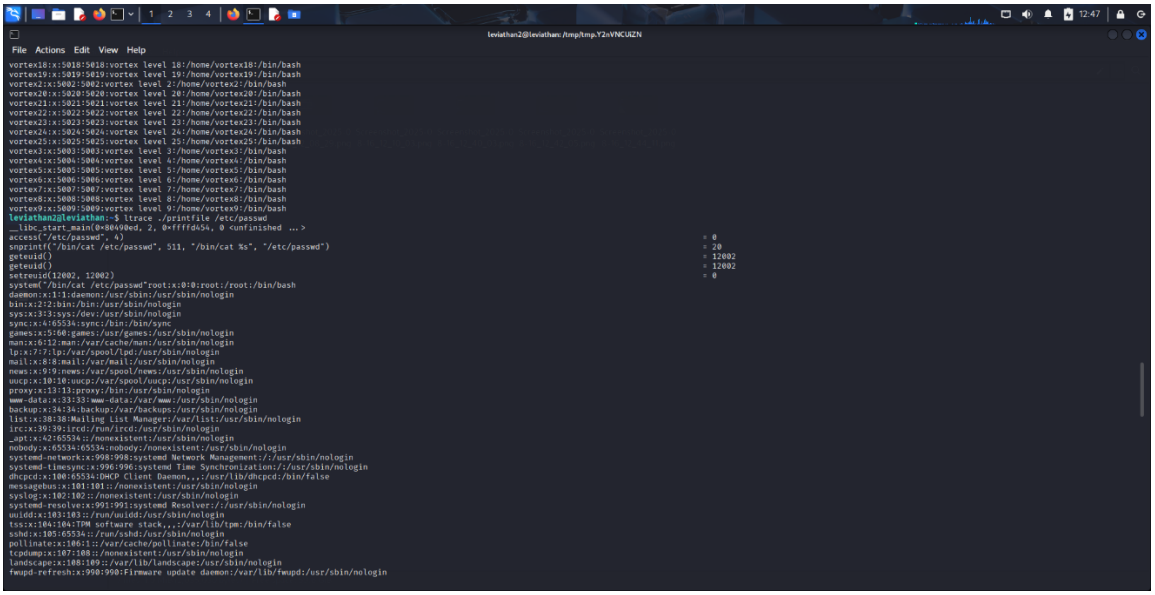
Objective: Exploit insecure command usage inside a binary.

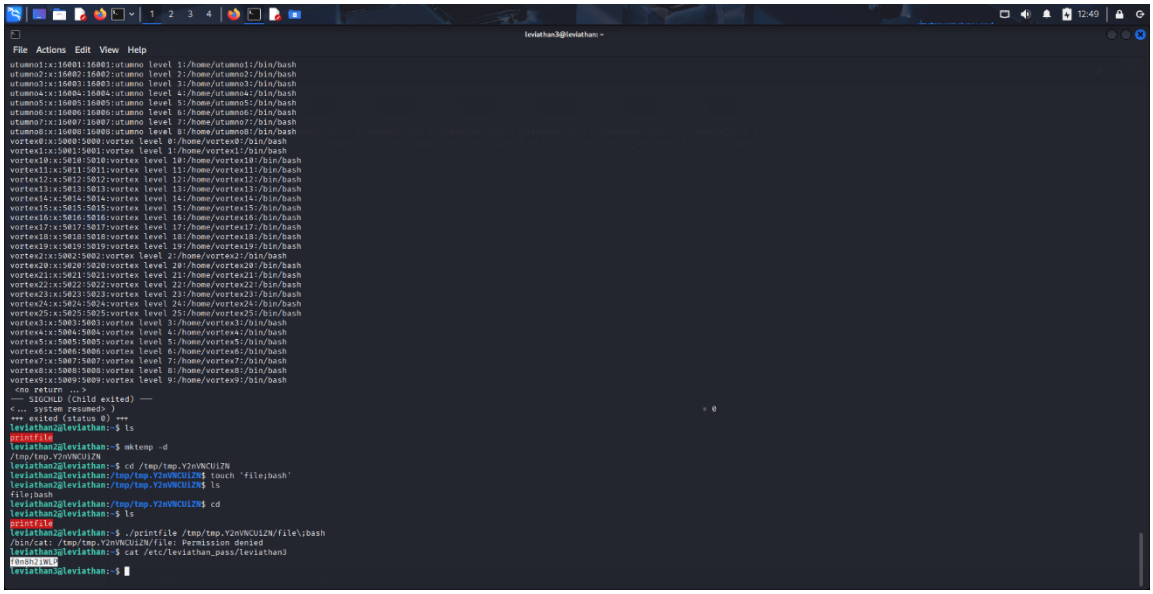
Steps Followed:

1. Found the binary `printfile`.
2. Running with `/etc/leviathan_pass/leviathan3` returned “you can’t have that file.”
3. Using `ltrace`, discovered it calls the cat command.
4. Created a malicious filename `/tmp/file;bash` to inject commands.
5. Executed binary with this input, escalated privileges, and accessed `/etc/leviathan_pass/leviathan3`.

Conclusion: Gained experience in exploiting command injection vulnerabilities.







Level 3 → Level 4

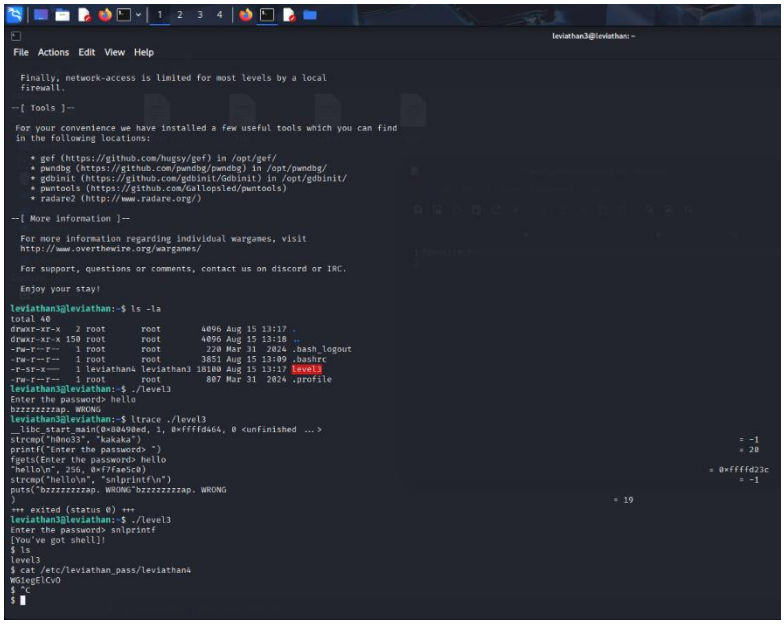
Tools Used: `ltrace`, `./binary`

Objective: Reveal password through function call comparison.

Steps Followed:

1. Found binary level3.
2. Running it prompted for a password.
3. Traced with `ltrace ./level3` and saw a `strcmp()` comparison with **`snlprintf`**.
4. Executed `./level3 snlprintf` which granted shell access.
5. Retrieved password from `/etc/leviathan_pass/leviathan4`.

Conclusion: Learned to identify hardcoded credentials via function tracing.



Level 4 → Level 5

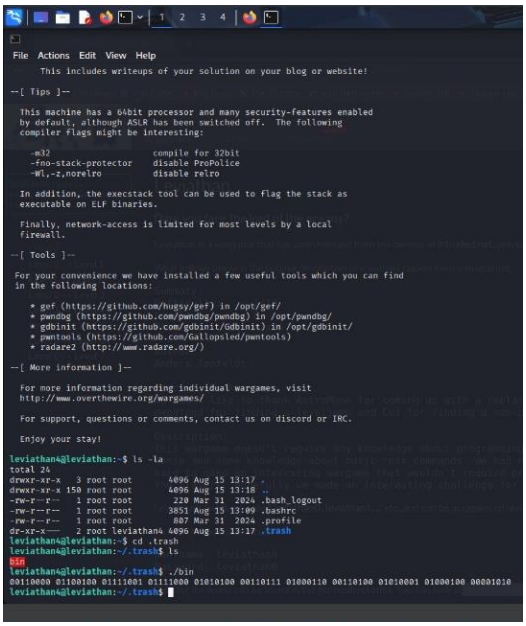
Tools Used: Hidden file inspection, ASCII conversion

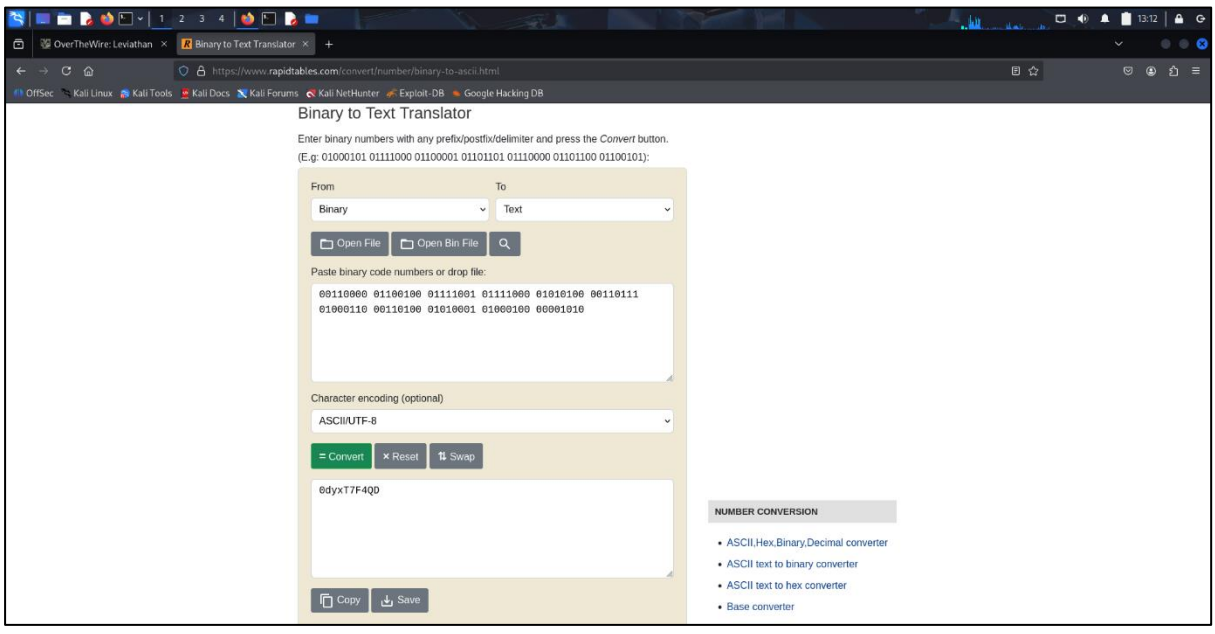
Objective: Decode hidden binary file to extract password.

Steps Followed:

1. Logged in as leviathan4.
2. Searched directories and discovered a hidden `.trash` folder.
3. Found a binary file inside named `bin`.
4. Converted binary content to ASCII using online tools.
5. Extracted the password for leviathan5.

Conclusion: Learned how hidden binary files can store encoded secrets.





Level 5 → Level 6

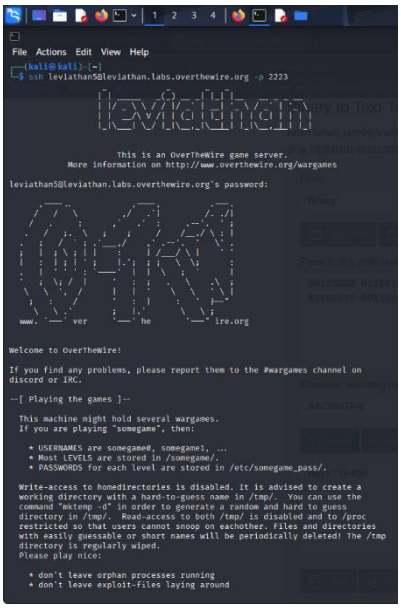
Tools Used: `ln -s`, symbolic links

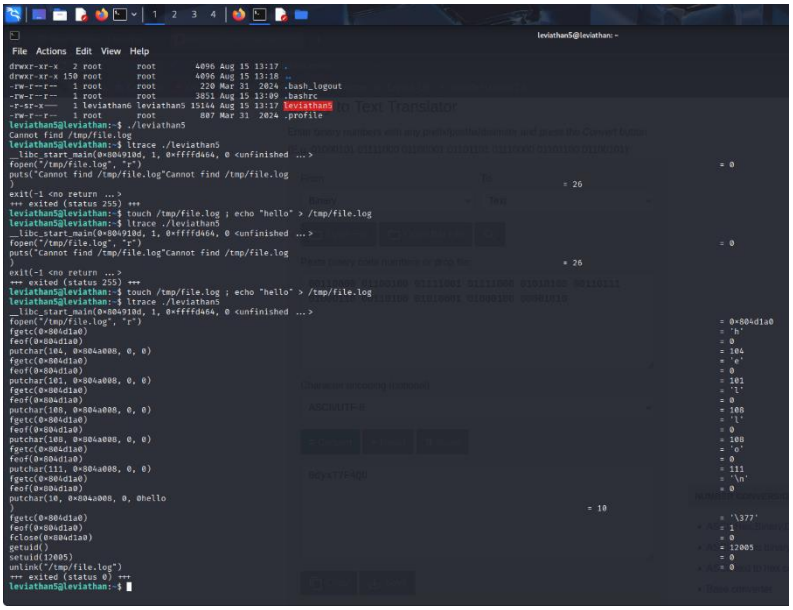
Objective: Exploit SUID binary with symlink trick.

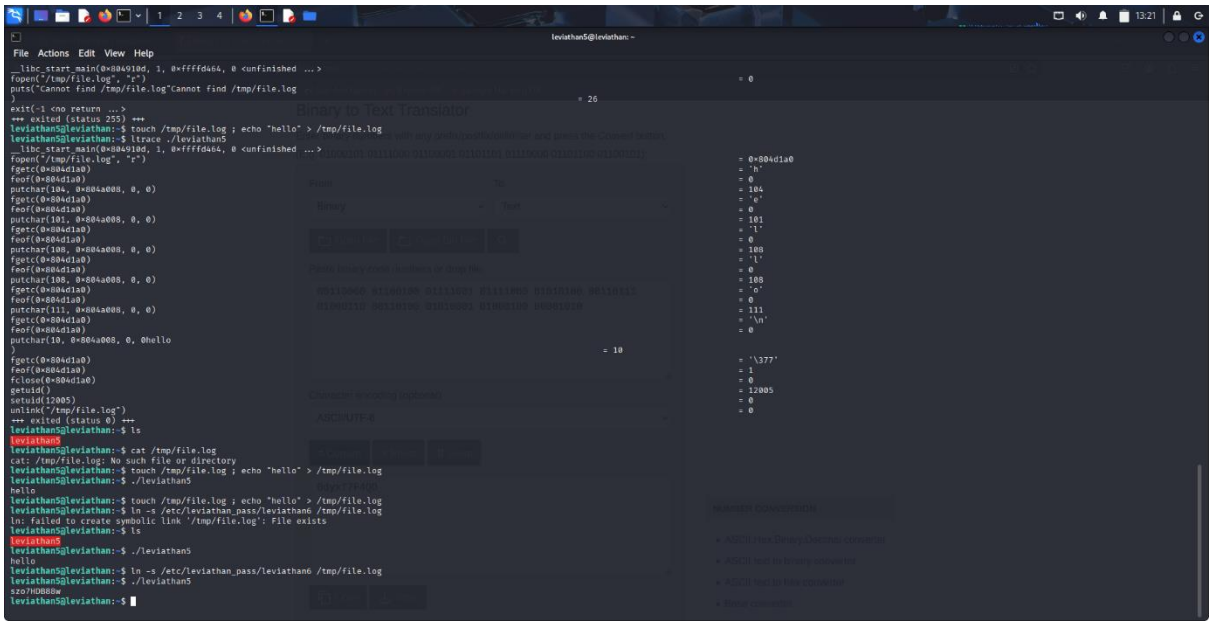
Steps Followed:

1. Found a binary that attempted to use `/tmp/file.log`.
2. Created a symbolic link:
3. `ln -s /etc/leviathan_pass/leviathan6 /tmp/file.log`
4. Ran the binary, which printed the contents of the password file.
5. Retrieved the password for leviathan6.

Conclusion: Gained understanding of symlink attacks against SUID binaries.







Level 6 → Level 7

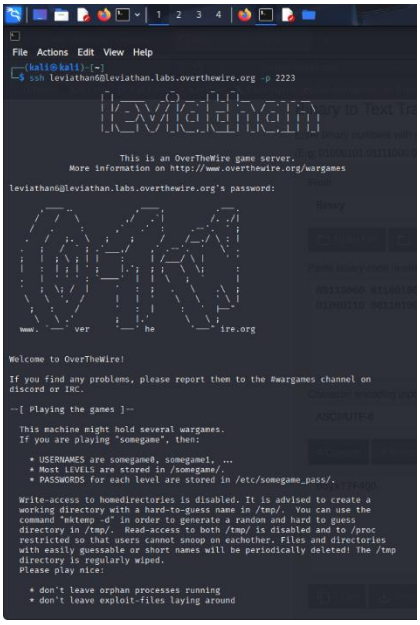
Tools Used: Bash scripting, brute force

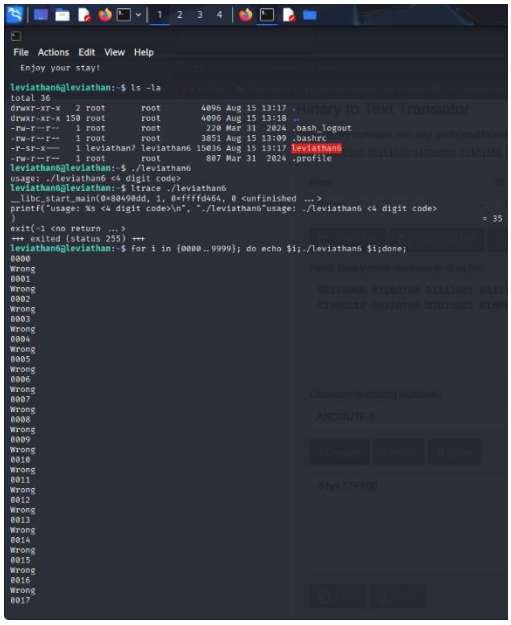
Objective: Crack a 4-digit PIN to gain shell access.

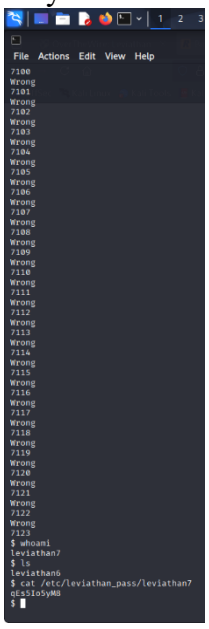
Steps Followed:

1. Binary leviathan6 asked for a 4-digit code.
2. Wrote a brute-force script to test all combinations from 0000 to 9999: `for i in {0000..9999}; do echo $i;./leviathan6 $i;done;`
3. When the correct PIN was entered, a shell was granted.
4. Accessed `/etc/leviathan_pass/leviathan7` to retrieve the password.

Conclusion: Learned brute-force automation for PIN-based security.







Level 7 (Final)

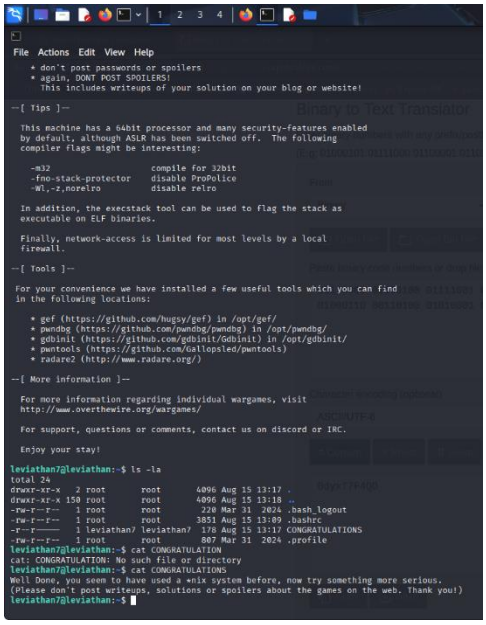
Tools Used: File inspection

Objective: Verify game completion.

Steps Followed:

1. Logged in as leviathan7.
2. No files were visible in the home directory.
3. Found and opened the file congratulations.
4. The file confirmed successful completion of Leviathan.

Conclusion: Completed all Leviathan levels successfully.



Summary Table

Level	Approach	Key Command / Method
0	Backup file inspection	`grep password bookmark.html`
1	Binary tracing with ltrace	`./check sex`
2	Command injection in filename	`/tmp/file;bash`
3	Hardcoded password via ltrace	`./level3 snlprintf`
4	Hidden binary decoding	ASCII conversion (using any site of conversion in website)
5	Symlink attack on temp file	`ln -s /etc/leviathan_pass/leviathan6 ...`
6	Brute-force PIN with loop	Bash brute force `for i in {0000..9999}; do echo $i;./leviathan6 $i;done;`
7	Completion check	`cat CONGRATULATIONS`

Conclusion

Through the successful completion of the OverTheWire Leviathan wargame (Levels 0–7), we gained valuable hands-on experience in Linux system navigation, privilege escalation, and binary exploitation. Each level introduced us to different aspects of cybersecurity problem-solving; from inspecting hidden files and analyzing binaries, to exploiting SUID programs, command injection, symbolic links, and brute-force techniques.

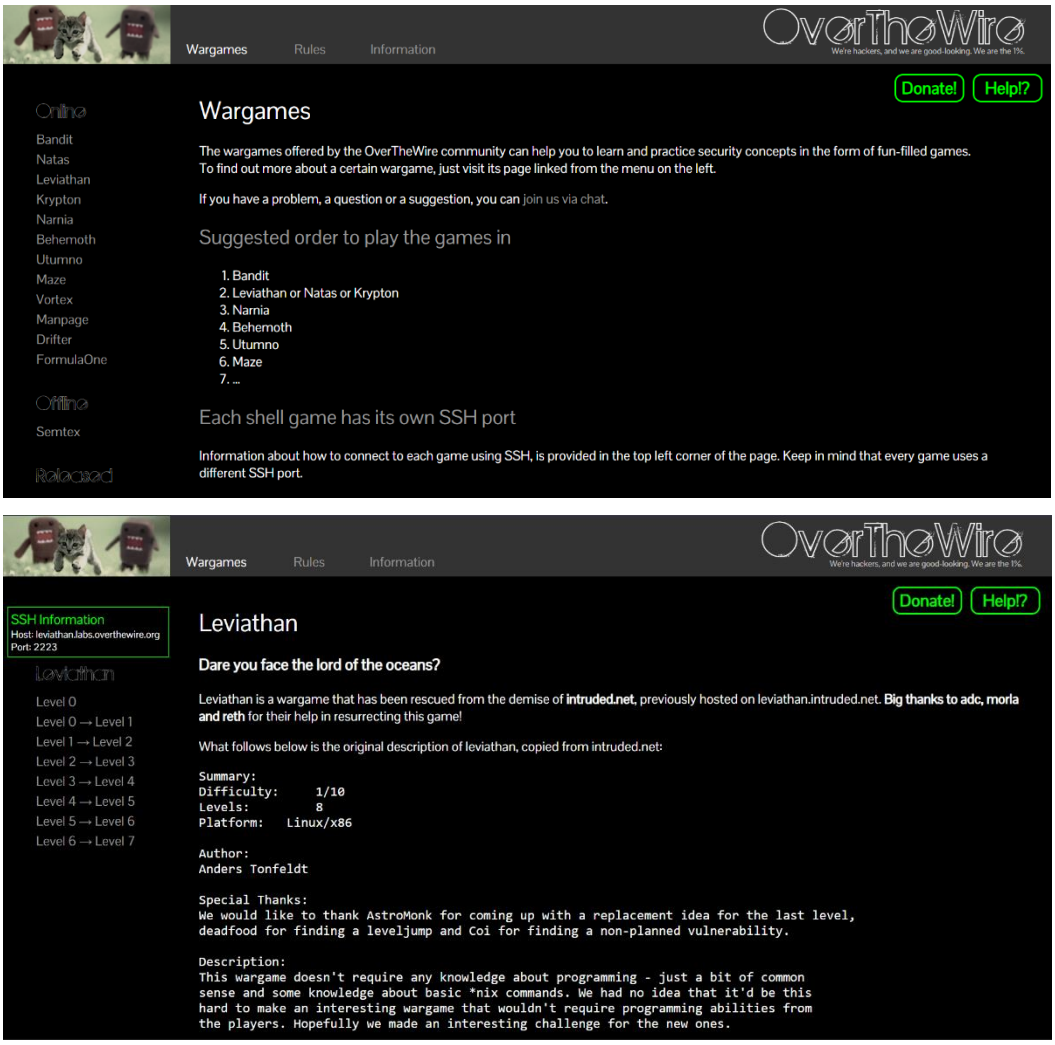
Working together as a team allowed us to share ideas, troubleshoot errors collaboratively, and develop a structured approach to solving challenges. As beginners, we enhanced our

understanding of Linux commands and their practical applications with the help of online references, which further deepened our confidence in applying theoretical knowledge to real-world scenarios.

Learning Outcomes

- Improved proficiency in **basic Linux commands** and their functional uses.
- Practical exposure to **binary analysis tools** such as `ltrace`, `strings`, etc.
- Understanding of **command injection** and symlink exploitation.
- Hands-on practice with **SUID binaries** and privilege escalation.
- Developed skills in **automation and brute-force scripting** using Bash.
- Strengthened teamwork, research ability, and problem-solving mindset.

This project not only helped us clear the Leviathan lab but also built a solid foundation for advancing further in Capture The Flag (CTF) challenges and cybersecurity learning.



OverTheWire Website Interface