

LinkedIn: [Kelvin Kimotho](#)

## Time Machine



Easy

General Skills

picoCTF 2024

browser\_webshell\_solvable

git

AUTHOR: JEFFERY JOHN

### Description

What was I last working on? I remember writing a note to help me remember...

You can download the challenge files here:

- [challenge.zip](#)

### Hints ?

1

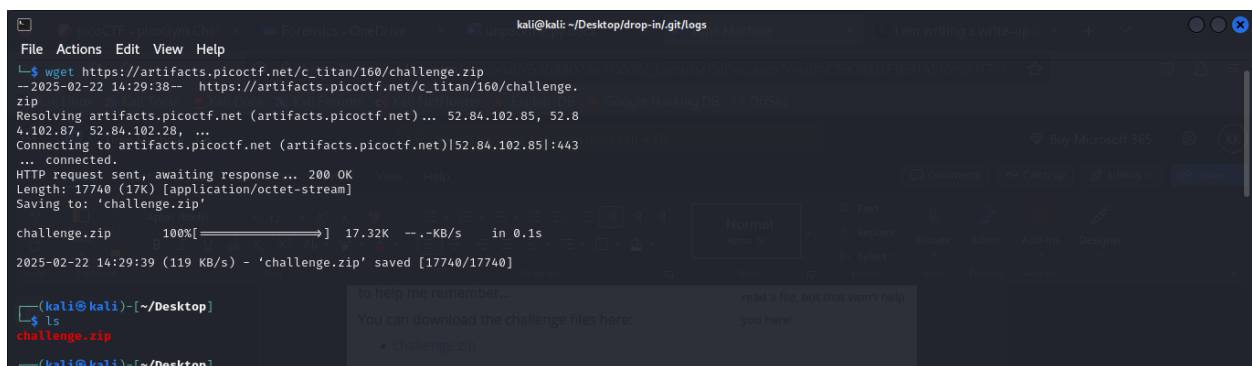
2

3

The `cat` command will let you read a file, but that won't help you here!

### Solution

I began by downloading the challenge file from the provided URL using `wget` tool. Once the download was complete, I checked my current directory by running the `ls` command to confirm that the `challenge.zip` file was present.



```
kali@kali: ~/Desktop/drop-in/.git/logs
File Actions Edit View Help
└─$ wget https://artifacts.picoctf.net/c_titan/160/challenge.zip
--2025-02-22 14:29:38-- https://artifacts.picoctf.net/c_titan/160/challenge.zip
Resolving artifacts.picoctf.net (artifacts.picoctf.net) ... 52.84.102.85, 52.84.102.87, 52.84.102.28, ...
Connecting to artifacts.picoctf.net (artifacts.picoctf.net)|52.84.102.85|:443
... connected.
HTTP request sent, awaiting response... 200 OK
Length: 17740 (17K) [application/octet-stream]
Saving to: 'challenge.zip'

challenge.zip  100%[=====] 17.32K  --.-KB/s  in 0.1s

2025-02-22 14:29:39 (119 KB/s) - 'challenge.zip' saved [17740/17740]

(kali@kali)-[~/Desktop]
└─$ ls
challenge.zip
```

I then unzipped the downloaded file, which created a directory called `drop-in`. This directory contained several files and folders, including a `message.txt` file and a `.git` directory.

```
(kali@kali)-[~/Desktop]
$ unzip challenge.zip
Archive: challenge.zip
  creating: drop-in/
  inflating: drop-in/message.txt
  creating: drop-in/.git/
  creating: drop-in/.git/branches/
  inflating: drop-in/.git/description
  creating: drop-in/.git/hooks/
  inflating: drop-in/.git/hooks/applypatch-msg.sample
  inflating: drop-in/.git/hooks/commit-msg.sample
  inflating: drop-in/.git/hooks/fsmonitor-watchman.sample
  inflating: drop-in/.git/hooks/post-update.sample
  inflating: drop-in/.git/hooks/pre-applypatch.sample
  inflating: drop-in/.git/hooks/pre-commit.sample
  inflating: drop-in/.git/hooks/pre-merge-commit.sample
  inflating: drop-in/.git/hooks/pre-push.sample
  inflating: drop-in/.git/hooks/pre-rebase.sample
  inflating: drop-in/.git/hooks/pre-receive.sample
  inflating: drop-in/.git/hooks/prepare-commit-msg.sample
  inflating: drop-in/.git/hooks/update.sample
  creating: drop-in/.git/info/
  inflating: drop-in/.git/info/exclude
  creating: drop-in/.git/refs/
  creating: drop-in/.git/refs/heads/
  extracting: drop-in/.git/refs/heads/master
  creating: drop-in/.git/refs/tags/
  extracting: drop-in/.git/HEAD
  inflating: drop-in/.git/config
  creating: drop-in/.git/objects/
  creating: drop-in/.git/objects/pack/
  creating: drop-in/.git/objects/info/
  creating: drop-in/.git/objects/43/
  extracting: drop-in/.git/objects/43/246218ab4fc7b30e9a9dff073e012316851469
  creating: drop-in/.git/objects/25/
  extracting: drop-in/.git/objects/25/16effb8d70e33bdd0023629b164a77225e1ec2
  creating: drop-in/.git/objects/89/
  extracting: drop-in/.git/objects/89/d296ef533525a1378529be66b22d6a2c01e530
  inflating: drop-in/.git/index
  extracting: drop-in/.git/COMMIT_EDITMSG
```

I changed into the drop-in directory using the `cd` command to explore its contents further. Inside, I found the `message.txt` file. I opened the message.txt file, which contained a note indicating that I needed to look at the commit history to understand more about the work that had been done.

```
(kali@kali)-[~/Desktop]
$ ls
challenge.zip  drop-in

(kali@kali)-[~/Desktop]
$ cd drop-in

(kali@kali)-[~/Desktop/drop-in]
$ ls
message.txt

(kali@kali)-[~/Desktop/drop-in]
$ ls -al
total 4
drwxr-xr-x 3 kali kali 80 Mar 12 2024 .
drwxr-xr-x 3 kali kali 80 Feb 22 14:29 ..
drwxr-xr-x 8 kali kali 260 Mar 12 2024 .git
-rw-r--r-- 1 kali kali 87 Mar 12 2024 message.txt

(kali@kali)-[~/Desktop/drop-in]
$ cat message.txt
This is what I was working on, but I'd need to look at my commit history to know why ...

(kali@kali)-[~/Desktop/drop-in]
```

Following the hint in the message, I navigated into the `.git` directory to find relevant files related to the commit history. Inside the `.git` directory, I found a `logs` folder and entered it to look for commit information.

```
(kali@kali)-[~/Desktop/drop-in]
$ cd .git

(kali@kali)-[~/Desktop/drop-in/.git]
$ ls
branches  config  HEAD  index  logs  refs
COMMIT_EDITMSG  description  hooks  info  objects

(kali@kali)-[~/Desktop/drop-in/.git]
$ cd logs

(kali@kali)-[~/Desktop/drop-in/.git/logs]
$ ls
HEAD  refs

(kali@kali)-[~/Desktop/drop-in/.git/logs]
```

Within the logs, I discovered a file named `HEAD`, which I suspected contained commit

information. I opened this file to check its contents. The output from the HEAD file revealed the commit history, including a commit message that contained the flag

**picoCTF{t1m3m@ch1n3\_186cd7d7}**.

```
(kali@kali)-[~/Desktop/drop-in/.git/logs]
$ file HEAD
HEAD: ASCII text

(kali@kali)-[~/Desktop/drop-in/.git/logs]
$ cat HEAD
00000000000000000000000000000000 89d296ef533525a1378529be66b22d6a2c01
e530 picoCTF <ops@picoctf.com> 1710202042 +0000 commit (initial): picoCTF{t1m
3m@ch1n3_186cd7d7}

(kali@kali)-[~/Desktop/drop-in/.git/logs]
$ cat HEAD | grep pico
00000000000000000000000000000000 89d296ef533525a1378529be66b22d6a2c01
e530 picoCTF <ops@picoctf.com> 1710202042 +0000 commit (initial): picoCTF{t1m
3m@ch1n3_186cd7d7}

(kali@kali)-[~/Desktop/drop-in/.git/logs]
$
```

With this, I successfully captured the flag for the picoCTF challenge!