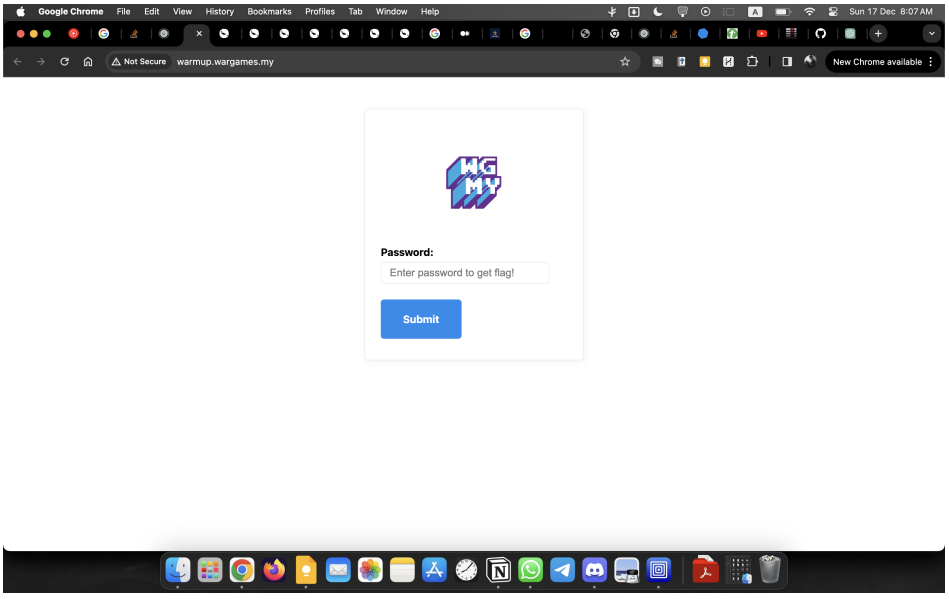
Wargames 2023 (student category)

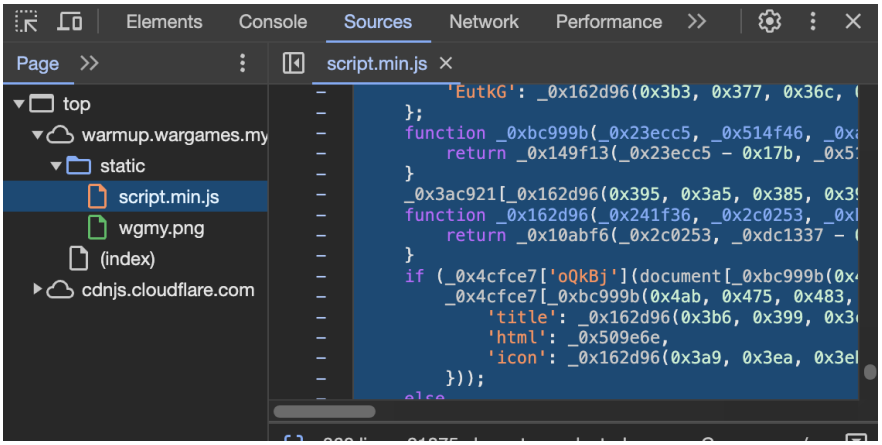
Created	@December 17, 2023 8:06 AM
By	Wahba Kamaluddin Nurdin (sud0su)
LinkedIN	https://www.linkedin.com/in/wahbakamaluddin/
CTF link	https://wargames.my/2023/

▼ Web

▼ Warmup



- Opening the link, there's sort of a login page asking for password.

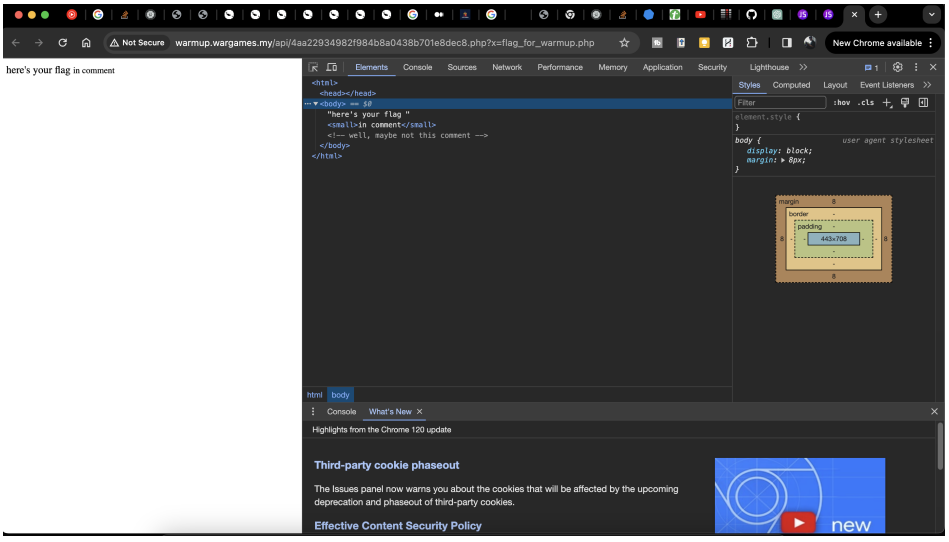


- Upon viewing the page source we can see there's script.min.js where it verifies the password we typed in. However, it's obfuscated, so we need to de-obfuscate it using online tool

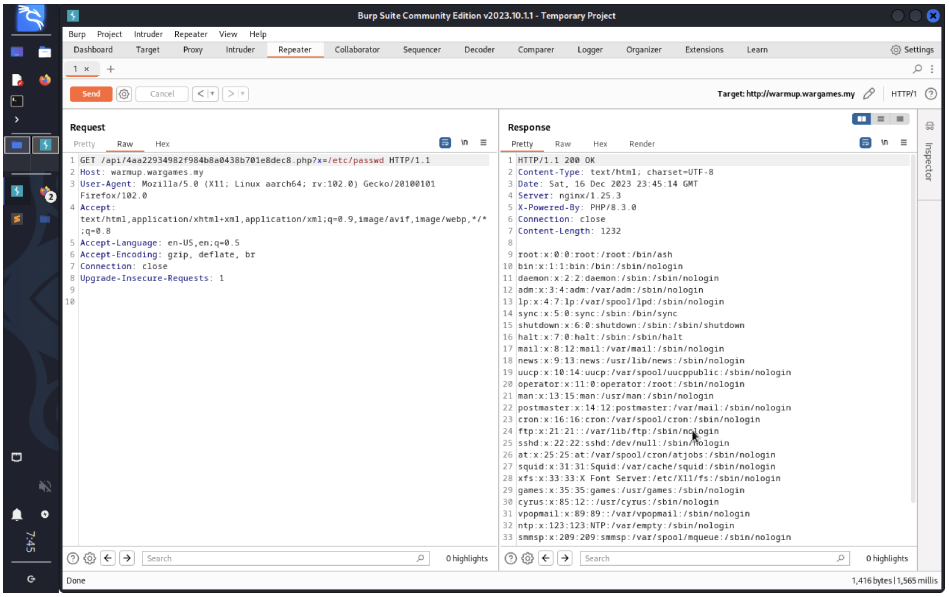
```
document.querySelector('button').addEventListener("click", _0x3ac921 => {
  _0x3ac921.preventDefault();
  if (document.querySelector("input").value === "this_password_is_so_weak_i_can_cr
    fetch("/api/4aa22934982f984b8a0438b701e8dec8.php?x=flag_for_warmup.php").then(
      'title': "Good job!",
```

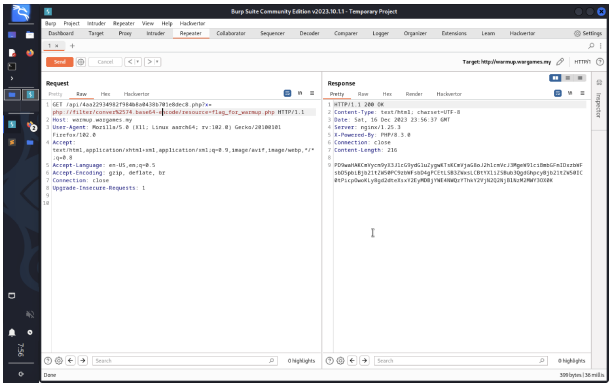
```
      'html': _0x509e6e,
      'icon': "success"
    }));
  } else {
    Swal.fire({
      'title': "Oops...",
      'text': "wrong password",
      'icon': "error"
    });
  }
});
```

- At the bottom of the script, we can see that it matches the password with `this_password_is_so_weak_i_can_crack_in_1_sec!`, and then fetch `/api/4aa22934982f984b8a0438b701e8dec8.php?x=flag_for_warmup.php`
- We can choose either to enter `this_password_is_so_weak_i_can_crack_in_1_sec!` as password or to use `http://warmup.wargames.my/api/4aa22934982f984b8a0438b701e8dec8.php?x=flag_for_warmup.php` as the url

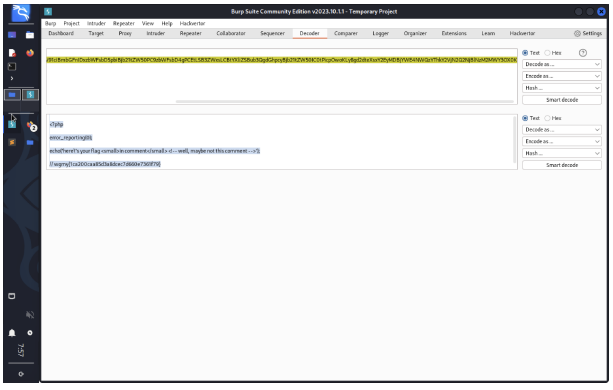


- Either way, we'll end up here, where the flag is supposed to be in the comment, but as we can see it is not in the HTML comment
- The statement is partially true. When you open a .php file in a web browser, the server processes the PHP code and sends the resulting HTML to the browser. In this process, the PHP code is executed, and only the output (usually HTML) is sent to the browser. This means that if there are comments or code in the PHP file that doesn't generate output, you won't see it in the browser.
- Since the file is a .php, it might suggest that the comment is in the .php file itself. The problem is, every time we open a .php file, the server will process the PHP code and sends the resulting HTML to the browser, as a result, the comment in the .php file will not be generated as an output. Therefore. we need to find a way to read the raw .php file.





- To do that, we can try to modify the x value in the `api/4aa22934982f984b8a0438b701e8dec8.php?x=flag_for_warmup.php` (To test if Local File Intrusion)(LFI) work, use `...x=/etc/passwd/` ; It return the content of etc/passwd, means LFI does work)
- The server execute PHP files when it detect opening php tag, `<?php` ; The theory is, if we can somehow convert the syntax, the server may not recognise it and won't execute it i.e: by encoding it
- To do it, modify the x value into `...x=php://filter/conver%2574/base64.encode/resource=flag_for_warmup.php` (the `t` is doubly encoded into `%2574` to bypass the restriction)



- Finally, we can decode the response and retrieve the flag