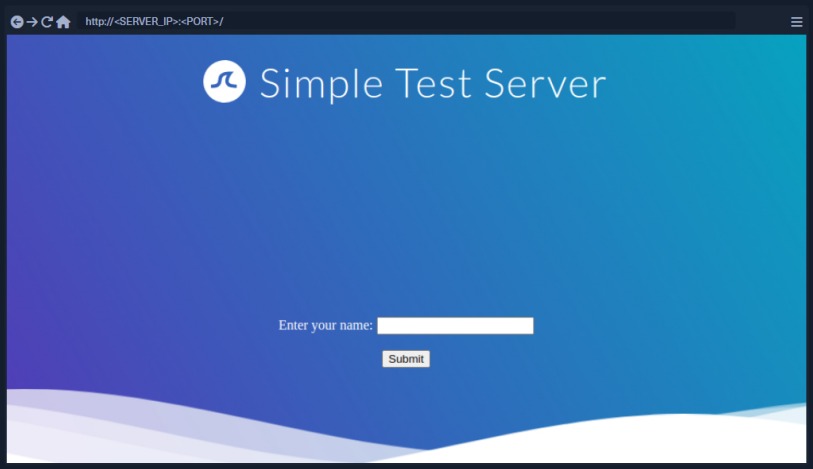
Exploiting SSI Injection

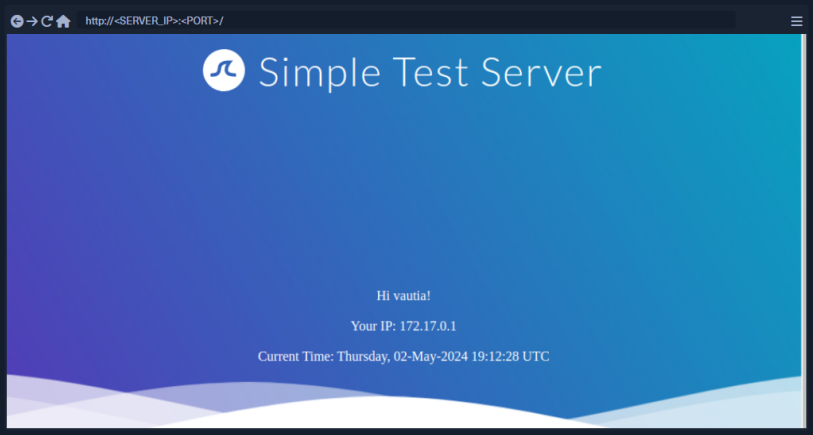
Now that we have discussed how SSI works in the previous section, let us discuss how to exploit SSI injection.

Exploitation

Let us take a look at our sample web application. We are greeted by a simple form asking for our name:

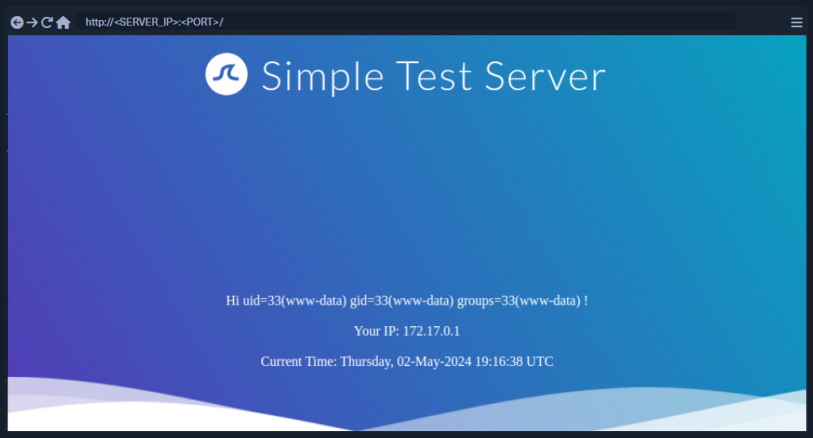


If we enter our name, we are redirected to `/page.shtml`, which displays some general information:



We can guess that the page supports SSI based on the file extension. If our username is inserted into the page without prior sanitization, it might be vulnerable to SSI injection. Let us confirm this by providing a username of `<!--#printenv -->`. This results in the following page:

As we can see, the directive is executed, and the environment variables are printed. Thus, we have successfully confirmed an SSI injection vulnerability. Let us confirm that we can execute arbitrary commands using the `exec` directive by providing the following username: `<!--#exec cmd="id" -->`:



The server successfully executed our injected command. This enables us to take over the web server fully.

Connect to Pwnbox
Your own web-based Parrot Linux instance to play our labs.

Pwnbox Location

UK 159ms

Terminate Pwnbox to switch location

Start Instance

∞ / 1 spawns left

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Cheat Sheet

Answer the question(s) below to complete this Section and earn cubes!

Target(s): Click here to spawn the target system!

+1 Exploit the SSI Injection vulnerability to obtain RCE and read the flag.

Submit your answer here...

+10 Streak pts Submit

Previous Next