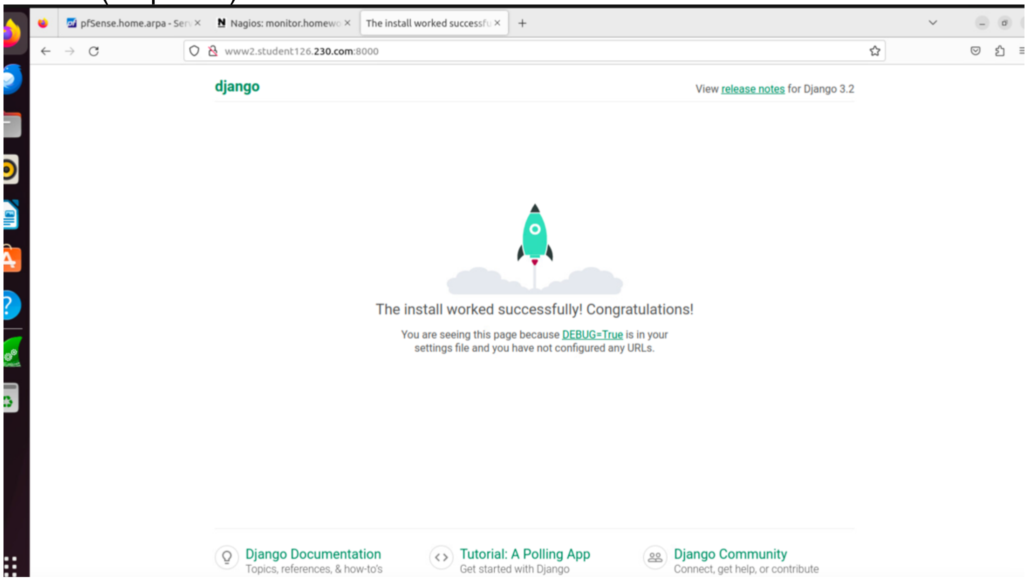
Lab 09 Template – Ethan Roepke

1. **Screenshot of default Django landing page**
(10 points)



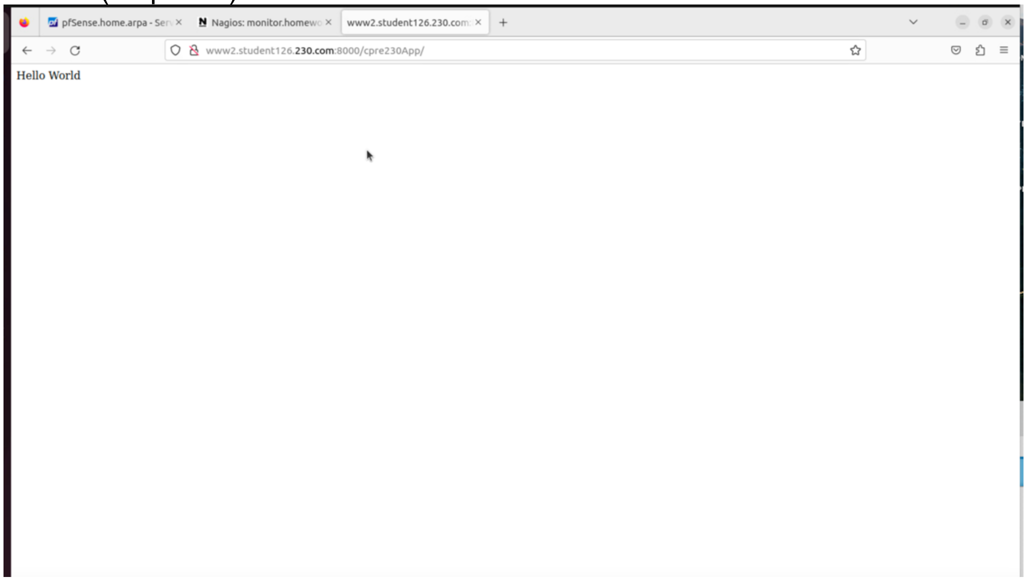
django

View release notes for Django 3.2

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

2. **Screenshot of testapp "Hello World" page**
(10 points)



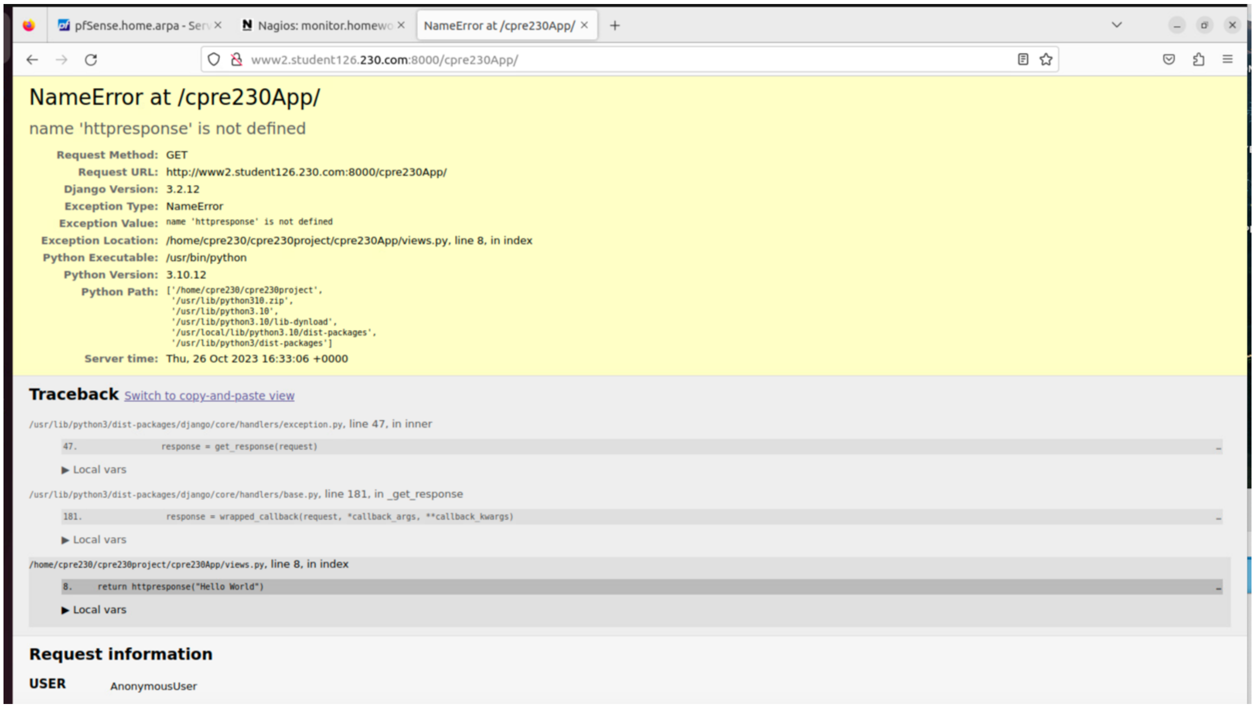
Hello World

3. **Screenshot of testapp debug information**
a. Identify 5 potentially dangerous facts revealed by the debug page.

b. How do you turn off the display of information?
(10 points)

A) 5 dangerous facts that are revealed on the debug page are giving away python path, exception location, python executable, variable names, and how the application works.

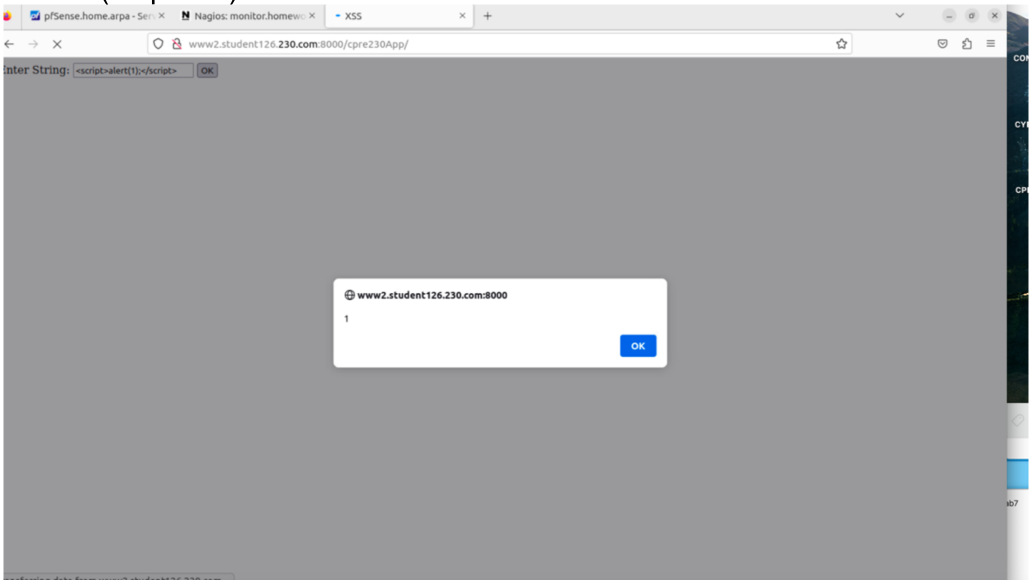
B) To turn off the display of information open the web.config file for the application and set the debug attribute to false for the compilation element


pfSense.home.arpa - Serv
Nagios: monitor.homewo
NameError at /cpre230App/
www2.student126.230.com:8000/cpre230App/

```
NameError at /cpre230App/

name 'httpresponse' is not defined

      Request Method: GET
         Request URL: http://www2.student126.230.com:8000/cpre230App/
      Django Version: 3.2.12
      Exception Type: NameError
     Exception Value: name 'httpresponse' is not defined
  Exception Location: /home/cpre230/cpre230project/cpre230App/views.py, line 8, in index
   Python Executable: /usr/bin/python
      Python Version: 3.10.12
         Python Path: ['/home/cpre230/cpre230project',
                       '/usr/lib/python310.zip',
                       '/usr/lib/python3.10',
                       '/usr/lib/python3.10/lib-dynload',
                       '/usr/local/lib/python3.10/dist-packages',
                       '/usr/lib/python3/dist-packages']
         Server time: Thu, 26 Oct 2023 16:33:06 +0000

Traceback Switch to copy-and-paste view

/usr/lib/python3/dist-packages/django/core/handlers/exception.py, line 47, in inner
    47.                 response = get_response(request)
    ▶ Local vars
/usr/lib/python3/dist-packages/django/core/handlers/base.py, line 181, in _get_response
    181.                response = wrapped_callback(request, *callback_args, **callback_kwargs)
    ▶ Local vars
/home/cpre230/cpre230project/cpre230App/views.py, line 8, in index
    8.     return httpresponse("Hello World")
    ▶ Local vars

Request information
USER        AnonymousUser
```

4. **Screenshot of XSS javascript alert**

a. Explain why the javascript isn't being displayed on the page
(10 points)



Enter String: <script>alert(1);</script>

www2.student126.230.com:8000

1

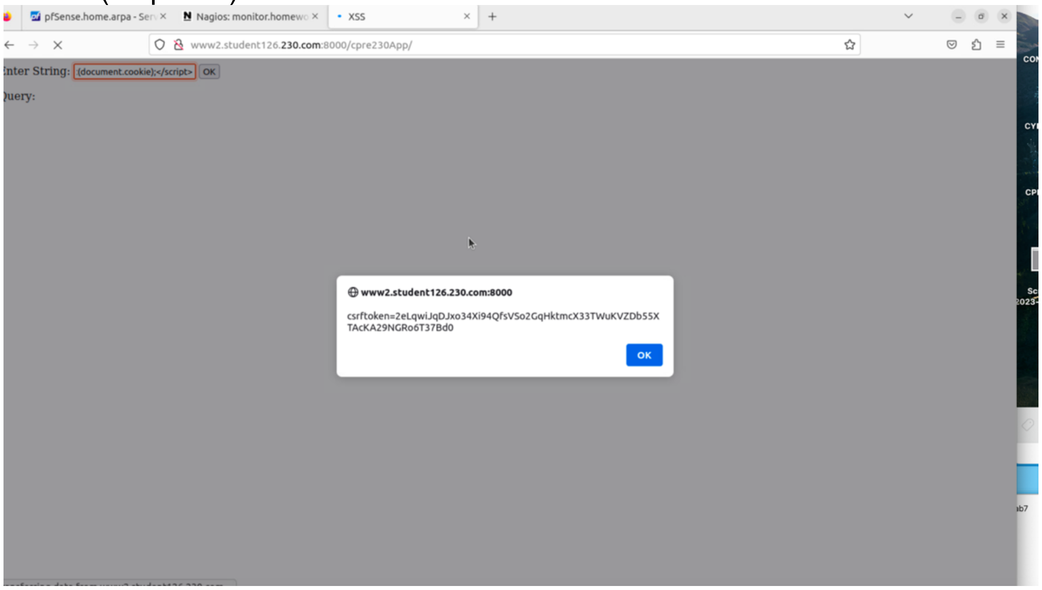
The JavaScript isn’t being displayed on the page because the line of code is giving an alert and inside the parenthesis will act like a message box so that’s why we get 1. If we were to change “1” to “Hello World” the alert message box would say “Hello World”.

5. **Screenshot of alert(document.cookie)**

a. How did we get this value?

b. Is it good or bad that we have that value?

(10 points)



:(document.cookie);</script>

www2.student126.230.com:8000

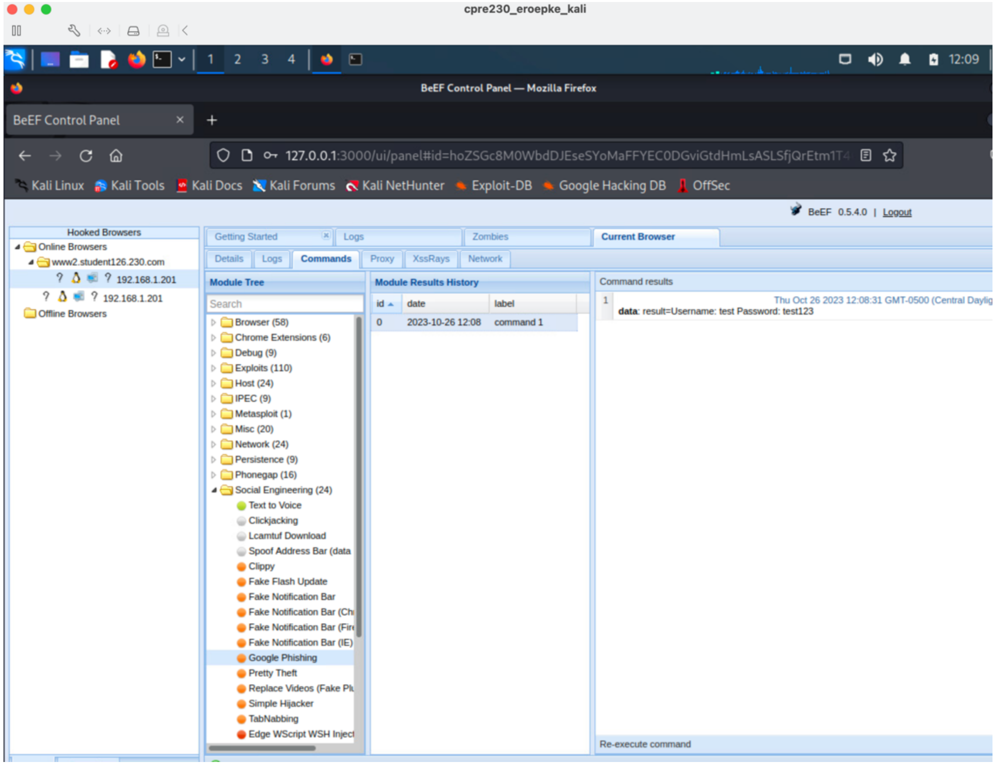
csrftoken=2eLqwiJqDJxo34Xi94QfsVSo2GqHktmcX33TWuKVZDb55X
TAcKA29NGRo6T37Bd0

A) We got this value by injecting HTML without any problems and alerted document.cookie to verify that XSS is present, hence the value we get.

B) This is bad that we have that value because this is how hackers use Reflected XSS and steal his cookies to gain access to ones account.

6. **Screenshot of username/password captured with BeEF**

(10 points)



Module Results History

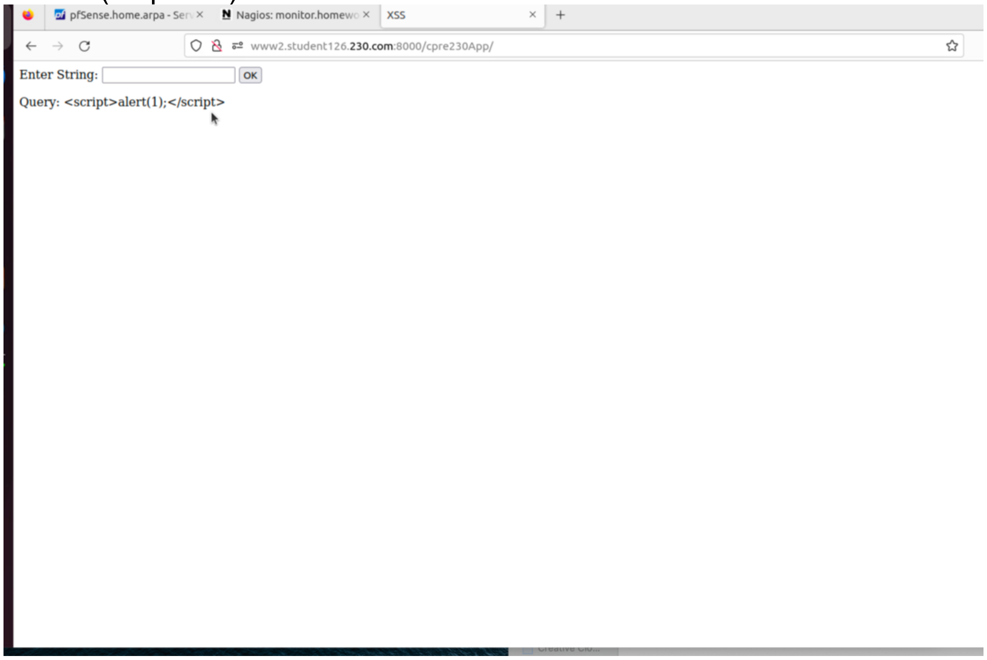
id	date	label
0	2023-10-26 12:08	command 1

Command results

1 Thu Oct 26 2023 12:08:31 GMT-0500 (Central Daylig
data: result=Username: test Password: test123

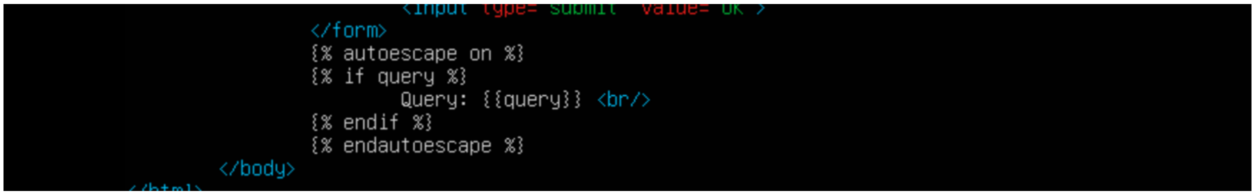
7. **Screenshot of fixed XSS page**

a. Screenshot and explanation of the problem and how you fixed it. (10 points)



Enter String:

Query: <script>alert(1);</script>



```
                <input type="submit" value="OK">
            </form>
            {% autoescape on %}
            {% if query %}
                Query: {{query}} <br/>
            {% endif %}
            {% endautoescape %}
        </body>
    </html>
```

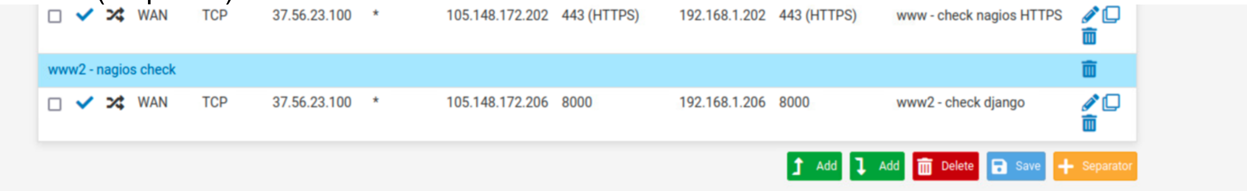
In out html file, I have changed {% autoescape off %} to {% autoescape on %}. Changing from off to on causes HTML code in variables to be escaped. So it will print out the query exactly.

8. **Lookup the three types of XSS**

a. Which did we use in this lab?
(10 points)

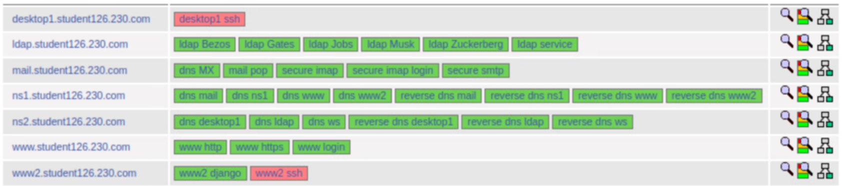
The three types of XSS are Stored XSS, Reflected XSS, and DOM-based XSS. Stored XSS is when an untrusted sources receives data and includes that data with HTTP responses in an unsafe way(permanently stored). Reflected XSS is when the web server is injected with a script that would give an error message, search result. DOM-based XSS is when the attacker modifies the users document object model environment in the browser.
We used reflected XSS in this lab because when we would open the website, everything seemed normal but, we were able to view the contents in BEEF for the attacker.

9. **Screenshot of NAT rule**
(10 points)



□	✓	WAN	TCP	37.56.23.100	*	105.148.172.202	443 (HTTPS)	192.168.1.202	443 (HTTPS)	www - check nagios HTTPS	
www2 - nagios check											
□	✓	WAN	TCP	37.56.23.100	*	105.148.172.206	8000	192.168.1.206	8000	www2 - check django	

10. **Screenshot of Nagios with django services successfully running**
(10 points)



desktop1.student126.230.com	🟥 desktop1 ssh
ldap.student126.230.com	🟩 ldap Bezos 🟩 ldap Gates 🟩 ldap Jobs 🟩 ldap Musk 🟩 ldap Zuckerberg 🟩 ldap service
mail.student126.230.com	🟩 dns MX 🟩 mail pop 🟩 secure imap 🟩 secure imap login 🟩 secure smtp
ns1.student126.230.com	🟩 dns mail 🟩 dns ns1 🟩 dns www 🟩 dns www2 🟩 reverse dns mail 🟩 reverse dns ns1 🟩 reverse dns www 🟩 reverse dns www2
ns2.student126.230.com	🟩 dns desktop1 🟩 dns ldap 🟩 dns ws 🟩 reverse dns desktop1 🟩 reverse dns ldap 🟩 reverse dns ws
www.student126.230.com	🟩 www http 🟩 www https 🟩 www login
www2.student126.230.com	🟩 www2 django 🟥 www2 ssh