**A Report**

**of**

# Mini project of Cyber Security


*Submitted by*

**Abhishek Kumar-202IT001**

**Mohd Asif Khan Khaishagi-202IT013**

*Submitted to*

**Dr. Jaidhar C.D.**

## Department of Information Technology

## National Institute of Technology, Surathkal

# Introduction

Phishing is a form of fraud in which the attacker tries to learn sensitive information such as login credentials or account information by sending as a reputable entity or person in email or other communication channels.

Typically a victim receives a message that appears to have been sent by a known contact or organization. The message contains malicious software targeting the user's computer or has links to direct victims to malicious websites in order to trick them into divulging personal and financial information, such as passwords, account IDs or credit card details.

Phishing is popular among attackers, since it is easier to trick someone into clicking a malicious link which seems legitimate than trying to break through a computer's defense systems.

So here we have tried to develop an APS(Anti phishing system) which takes an url as input and can tell whether that site is a phishing site or benign site.We know that to train a machine learning model, we need some features about these urls so that by learning about those features, machine can tell whether an url is phishing or benign. So we have considered three types of features.

### 1. URL based features:-

It is very important to take URL based features into consideration because phishers may manipulate the legitimate urls and make fraudulent urls and make people feel that this url is legitimate.

http://paypal.com-webappsuserid29348325limited.active-userid.com/webapps/89980/

| | |
|---|---|
| protocol | http:// |
| Domain name | active-userid.com |
| path | /webapps/89980/ |
| Subdomain item1 | com-webappsuserid29348325limited |
| Subdomain item2 | paypal |

## 2. Web document property:-

The Web document properties of a webpage are extracted from Document tag which includes the Title tag, Meta tag, Alt attribute of tags, Title attribute of tags, meta description etc. where keywords associated with a web page's product or services are defined. Thus, a web documents property can be acquired from its keyword's identity.

Now by using these document based property phishers may rank their website higher and manipulate users. so that if some user searches about that product then their website appears first and the user gets scammed.

## 3. The behaviour of a web page property:-

The behaviour of a web page describes the features of a webpage which are related to how the webpage handles its underground processes. Such underground processes may include how the transmission is made between the HTTP cookies and web server, the WHOIS history, port number type, certificate type e.g. selfissued or trusted third-party etc

# Datasets

- For the phishing dataset we crawled the phishtank website and extracted around 48.5K phishing urls and then among those urls we selected the urls which are valid phish and also online. After applying these 2 filters i.e validity and online then we got 2420 urls out of 48.5k.

- For Legitimate(benign) urls we have downloaded 1 million legitimate urls.Now since we got 2420 phish urls after the 2 filters for balancing the dataset for classification we'll also use around 2500 benign urls in the dataset.

# Pre-Processing

For generating the features corresponding to each url we checked five features corresponding to each property of url (i.e. url based , web document based and web behaviour based).

**URL based features**

1. **URL with '@' symbol:-** This involves using @ symbol in the URL path of a website. This symbol is used to redirect traffic to phishing sites whose domain name immediately followed the @ symbol. **For example, HTTP// mapoly.edu.ng@gatewaypoly.edu.ng** will direct a user to Gatewaypoly instead of Mapoly. The @ symbol usually comes with a shorter domain name unlike some other symbols such as ''-" or ''.".
   Therefore, if the URL contains @ symbol then phishing otherwise legitimate.

2. **Using the IP address as URL:-**This involves the use of IP address to represent the domain name of a website. Usually, this practice is very common for hiding the original information of a domain name. Hence, such IP addresses usually denote phishing or suspicious domains. If URL contains Domain path as IP address then Phishing, otherwise legitimate.

3. **URL with hexadecimal character code:**-Phishers usually hide phishing URLs by using hexadecimal codes to represent the numbers in the IP address. Each hexadecimal code usually begins with a ''%" symbol. **For instance, http://donefe.000webhostapp.com/auto/auto%20ferify/mail.php** which was reported in January 2018 by PhishTank used the hexadecimal character code. If URL contains hexadecimal characters then phishing, otherwise legitimate.

4. **URL length:-** This involves getting a URL length that is more than 35 characters. For example, HTTP// womenincoachingsuccess.com found on the Alexa database is a legitimate URL. A close observation of the Alexa database indicated that any length of more than 35 is likely to be phishing. If URL length greater than 35 then phishing Otherwise,legitimate.

5. **URL with multiple '//':-** This involves the use of more than one ''//" in the domain name path of a URL. A search query on the 1 million Alexa database of legitimate URLs in a.csv excel format returns 0 for this feature. If URL contains multiple ''//" then phishing Otherwise legitimate

**Web document based features**

1. **Domain name check:-**In most usual cases, website domain names (Dn) have a strong relationship with their contents (C) depicting the nature of products or services offered by the webpage. The keywords in this domain name are usually part of the base domain URL and should form the label for most links/anchors on the page. Therefore, if the keyword identity set of a page is not related to its contents (at least 70%), then it is phishing. Otherwise, it is legitimate.

2. **The domain name in the path of a URL: -**Some phishing URLs add the domain name of a legitimate website within the path segment of a URL in an attempt to scam users into believing that they are dealing with an authentic website.This implies that this feature can equally detect the use of prefix or suffix by phishers in reshaping suspicious domain name as the genuineness will be low due to inappropriate keyword identity set. Therefore, if the domain name in the path of a URL contains a prefix or suffix (Dps) that is not indicated in its contents then it is phishing. Otherwise, it is legitimate.

3. **Server Form Check (SFC)/Pop-Up Window:-**In a normal form processing operation, the domain name of a webpage is the same as the active form field address where the information is processed. But if there are any discrepancies between these two addresses or the domain name of the form is empty or missing, then it is likely to be phishing. Besides, a pop-up window can be activated by the phisher to circumvent this attribute. Since the goal of every phishing web page is to have access to user's details, they achieve this by sending the user's form field to their servers where they can have access to it through a pop-up window.Although most modern browsers allow window. open (i.e.one of the commands for creating pop-up window) to run only if it was called by user interaction, phishers can trigger on a mouse click event listener attached directly to the web documents to achieve their malicious intent. In this way, the call restriction to the mouse click events can be hijacked.Therefore, if a webpage contains a Pop-up window and the domain name/keyword identity set on the pop-up window is not related to the foreground URL,then it is phishing,Otherwise, it is genuine.

4. **Abnormal URL Shortening:-** Phishers use URL shorteners to obfuscate phishing URLs when requesting unsuspecting users to log-in their accounts through a link especially on social networking sites. If URL Shortening Service (USS) such as **Bit.ly, goo.gl, Owl.ly, Deck.ly, Su.pr** etc. then it is likely to be legitimate.Otherwise,it is phishing.

5. **Downloadable malicious code:-**Most phishing sites or emails contain an instruction to download certain files which are used to perpetrate crimeware-based attacks. If a webpage contains an active download link which contains specified extensions such as .aaa, .abc, .exx, .help_restore, 6–7 length extension of random characters, then it is phishing and suspicious. Otherwise, it is legitimate.

**The behaviour of web page property**

1. **Abnormal Cookie domain: -**This feature checks how the transmission of text data is done by a web server to a web client. Information about client machines/users is usually maintained in this text data, which is sometimes called HTTP cookies. If a website has a domain cookie (DC ) which is in a foreign domain, then it may be deceptive as most benign websites have their domain cookies or no cookies (Durl).

2. **Age of domain:-**This feature checks the age of the domain name of a particular URL or the URL extracted from the action attribute of a form using the WHOIS API search. Many phishing pages claim the identity of a known brand which has a relatively long history. If the age of the domain does not correspond to its WHOIS lookups, then it is likely to be deceptive.

3. **Port Number behaviour:-** This feature compares the port number part of a domain name with the stated protocol part of a URL. If the protocol does not match the port number,then the page is a phishing site.

4. **SSL Certificate:-** The Secure Socket Layer certificate is often used every time-sensitive information is being transferred by a user to an honest website. This certificate can either be self-signed by a website or offered by a trusted third party such as GeoTrust, VeriSign etc. There is a higher level of probity with a trusted third-party certificate than a self signed certificate. So, checking that the SSL certificate is offered by the trusted issuer is a feature of the legitimate website. Otherwise, it is suspicious or phishy.

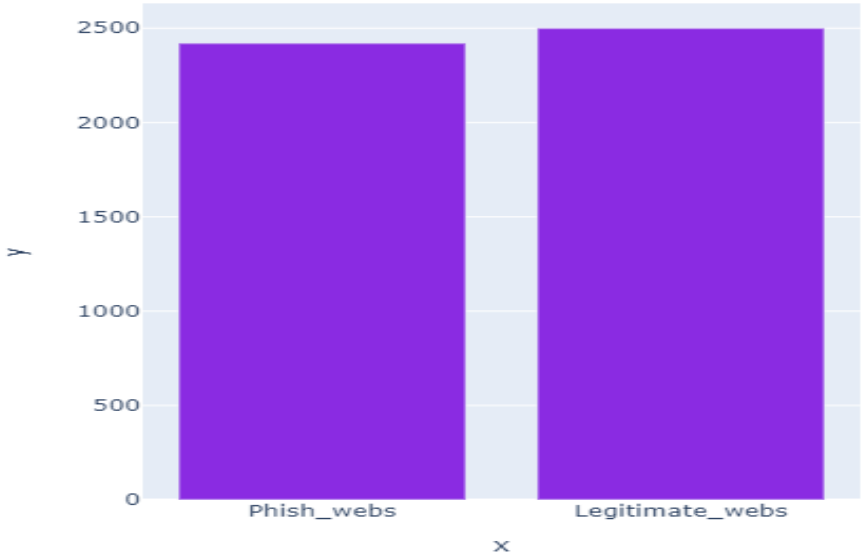
5. **Blacklisted domain:-** Since blacklisting of suspicious URLs has produced promising results in phishing detection, the blacklist domain is used as a feature in our approach to managing a list of locally detected phishing sites to bypassed superfluous computation on an already known malicious domain. This feature provides the advantages of providing resources for updating our feature corpus

and reducing overheads. If the domain name used in the action field of a login form or URI (DC ) is found in the blacklist domain, then it is likely to be phishing. Otherwise, it is legitimate.

**Note: -**For each feature corresponding to each filter we have written a function and if according to that feature **url is phishing then we returned 1** corresponding to that and **if legitimate then returned 0.**

# Data Visualization

First of all, Out of 48.5k urls which we have crawled from phishtank we have selected those urls which are online as well as phishing url.So we got 2420/48.5k urls and for making balance in dataset we selected random 2500 legitimate urls.



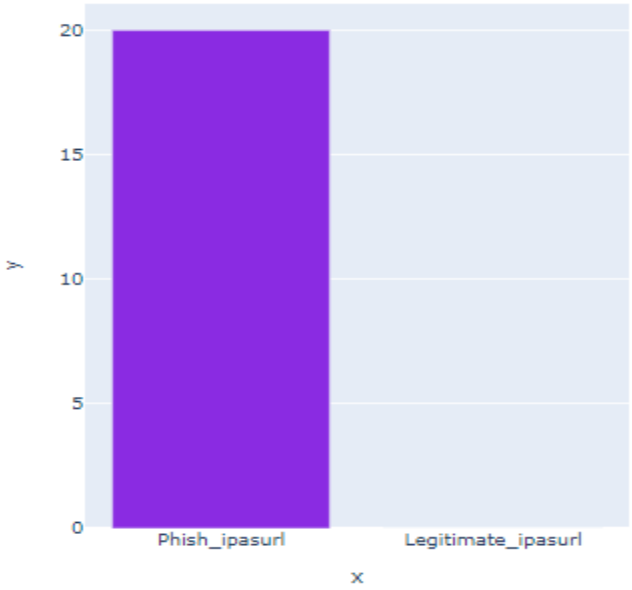

Now, we have checked corresponding to each feature how many urls are phishing or legitimate according to that feature.

1. **URL with '@' symbol:-**

```
No of phishing urls having @ symbol :  35
No of legitimate urls having @ symbol:  0
```

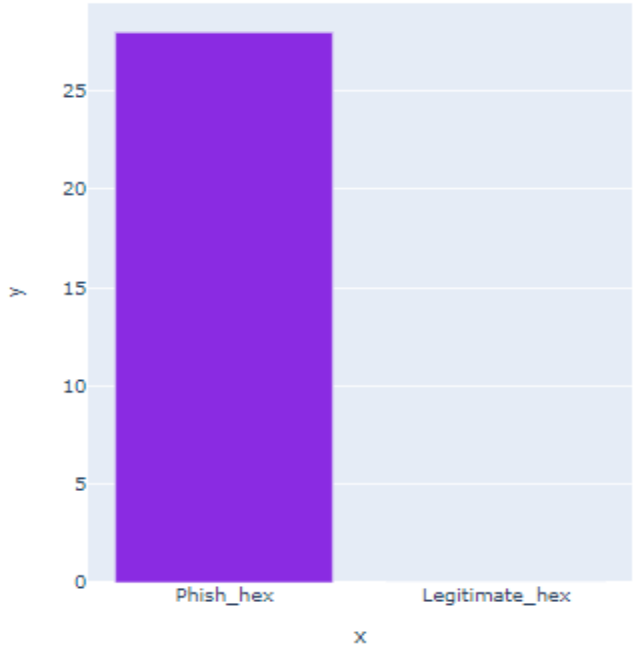
## 2. Using the IP address as URL:-

```
No of phishing urls having ip in url :  20
No of legitimate urls having ip in url:  0
```



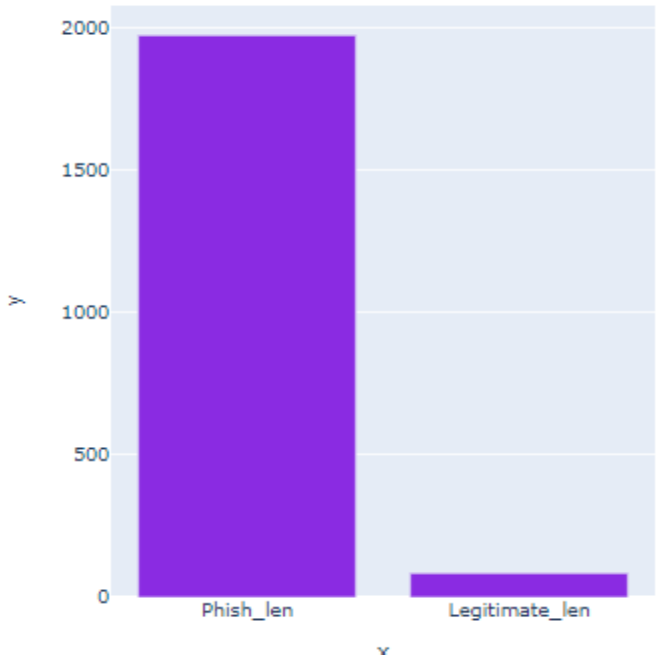

## 3. URL with hexadecimal character code:-

```
No of phishing urls having hex characters :  28
No of legitimate urls having hex characters:  0
```



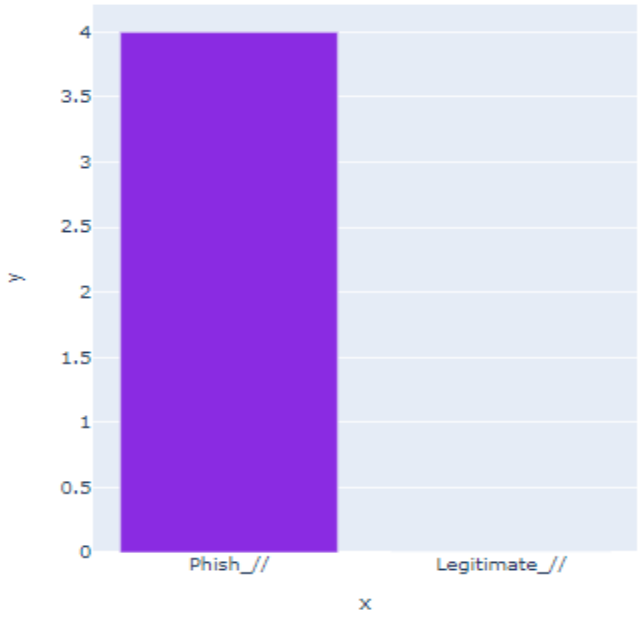
## 4. URL length:-

```
No of phishing urls having length greater than 35 :  1973
No of legitimate urls having length greater than 35:  85
```



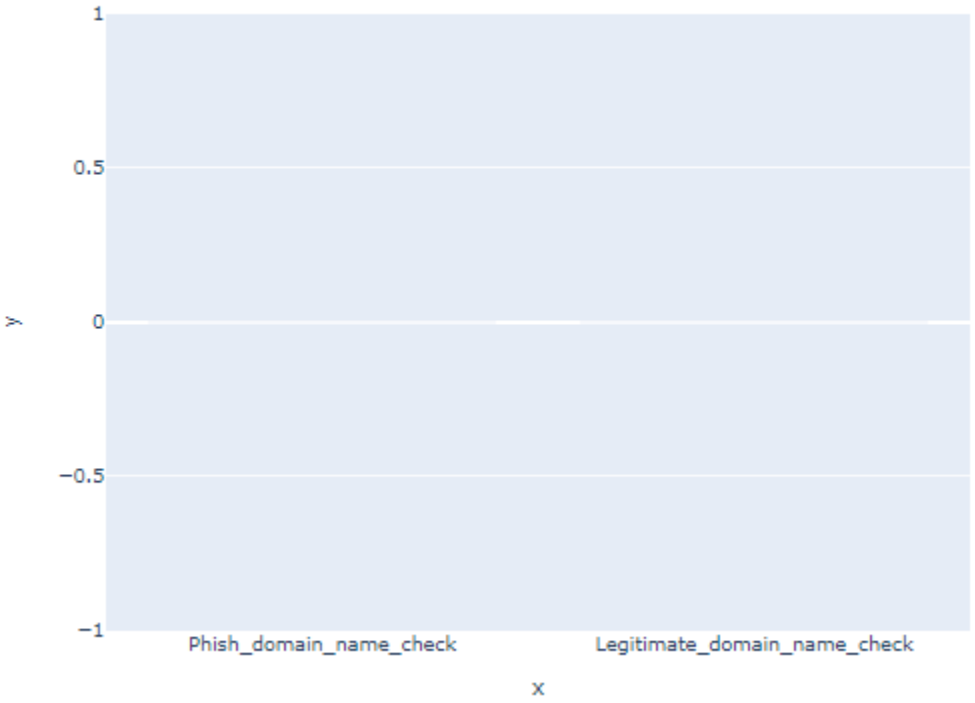

## 5. URL with multiple '//':-

```
No of phishing urls having multiple // :  4
No of legitimate urls having multiple //:  0
```



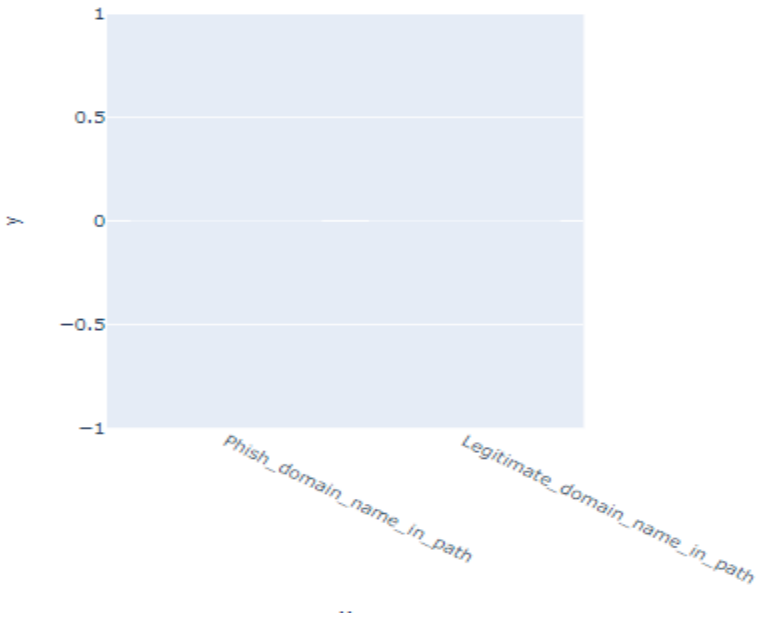
## 6. Domain name check:-

```
No of phishing urls having  keyword identity set of a page is not related to its contents :  0
No of legitimate urls having  keyword identity set of a page is not related to its contents:  0
```



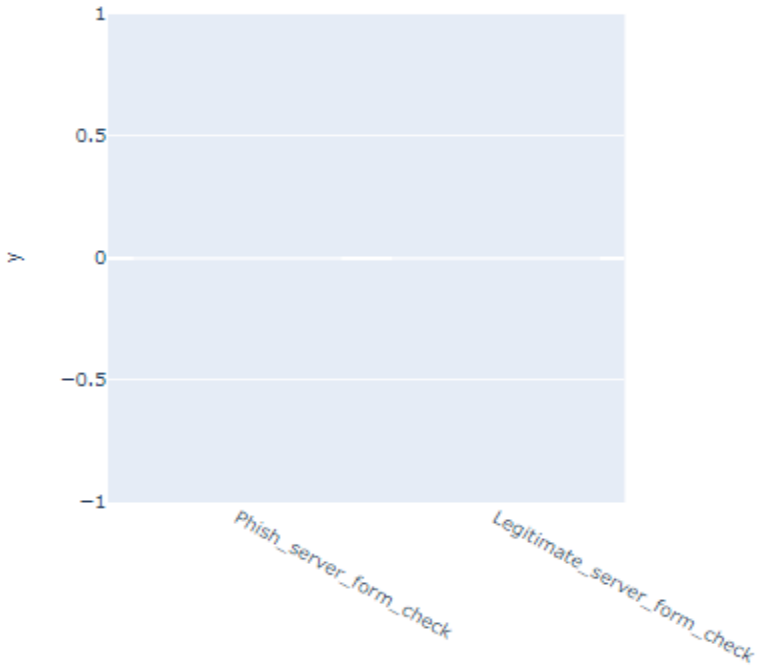

## 7. The domain name in the path of a URL: -

```
No of phishing urls  contains prefix or suffix (Dps) that is not indi□cated in its content :  0
No of legitimate urls contains prefix or suffix (Dps) that is not indi□cated in its content:  0
```



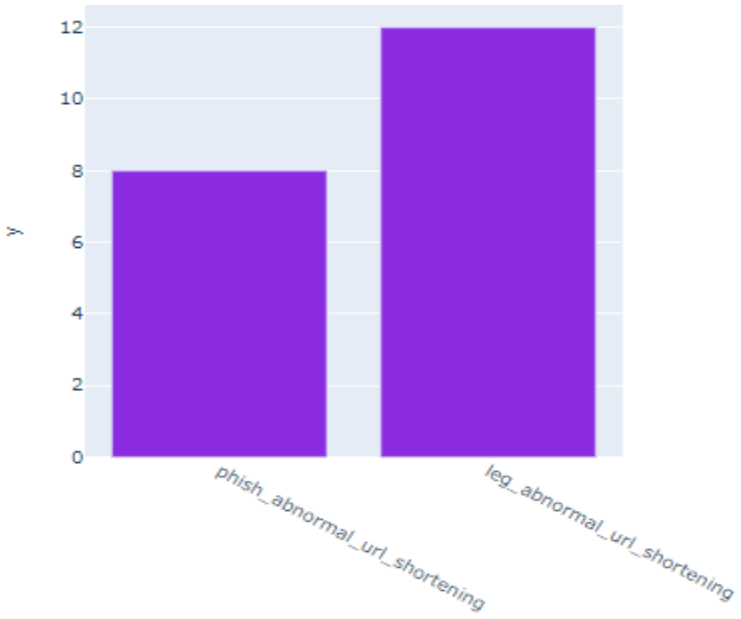
## 8. Server Form Check (SFC)/Pop-Up Window:-

```
No of phishing urls  contains where active field of from not matches with domain :  0
No of legitimate urls contains where active field of from not matches with domain:  0
```



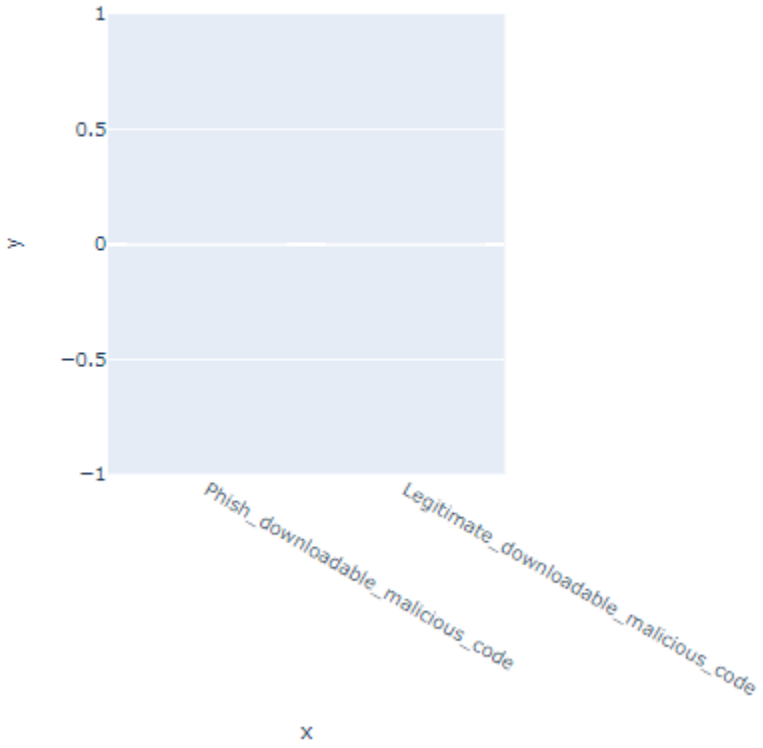

## 9. Abnormal URL Shortening:-

```
No of phishing urls not shorted with known shotner :  8
No of legitimate urls not shorted with known shotner :  12
```



## 10. Downloadable malicious code:-

```
No of phishing urls  contains downloadable_malicious_code :  0
No of legitimate urls contains downloadable_malicious_code:  0
```

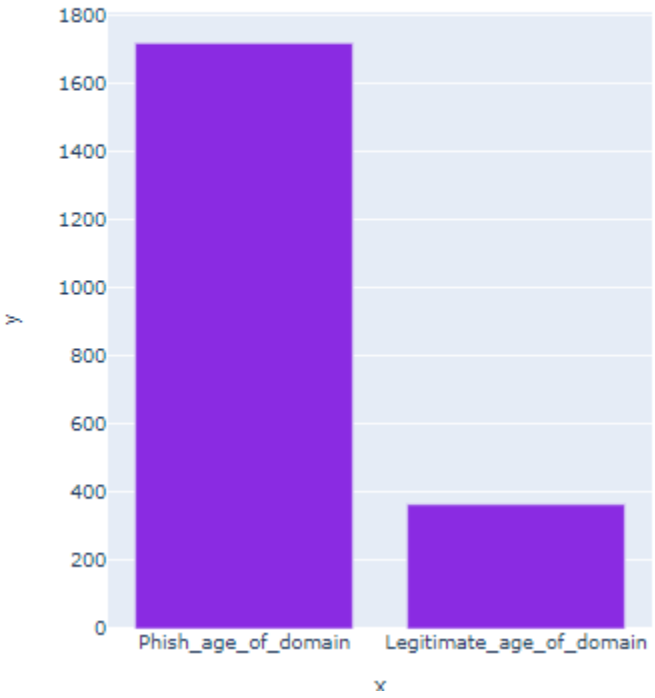



## 11. Abnormal Cookie domain: -

```
No of phishing urls  contains abnormal_cookie_domain :  0
No of legitimate urls contains abnormal_cookie_domain:  0
```

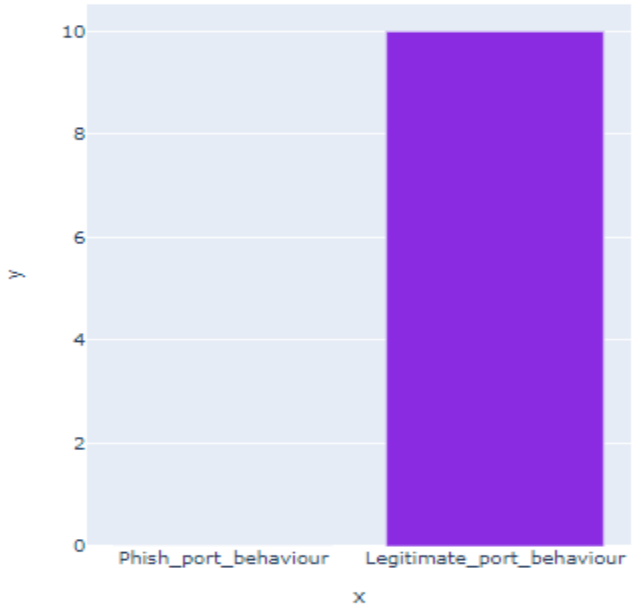
## 12. Age of domain:-

```
No of phishing urls  having age <1:  1717
No of legitimate urls  having age <1:  365
```




## 13. Port Number behaviour:-

```
No of phishing urls  contains port in url different than 80 :  0
No of legitimate urls port in url different than 80 :  10
```

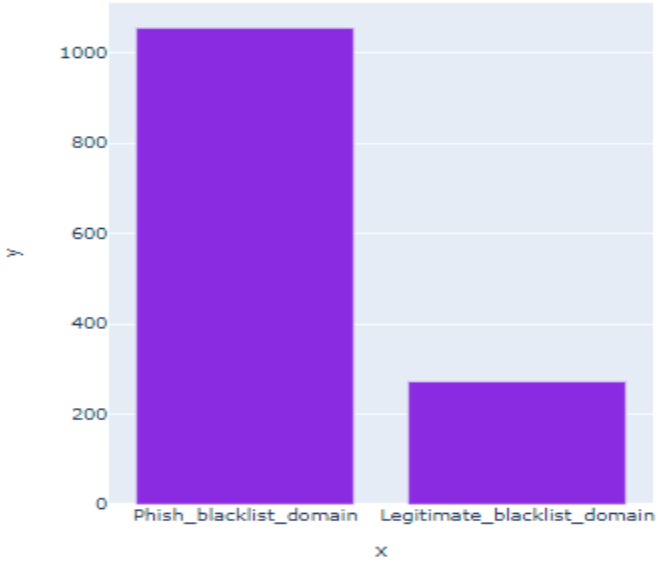


## 14. SSL Certificate:-

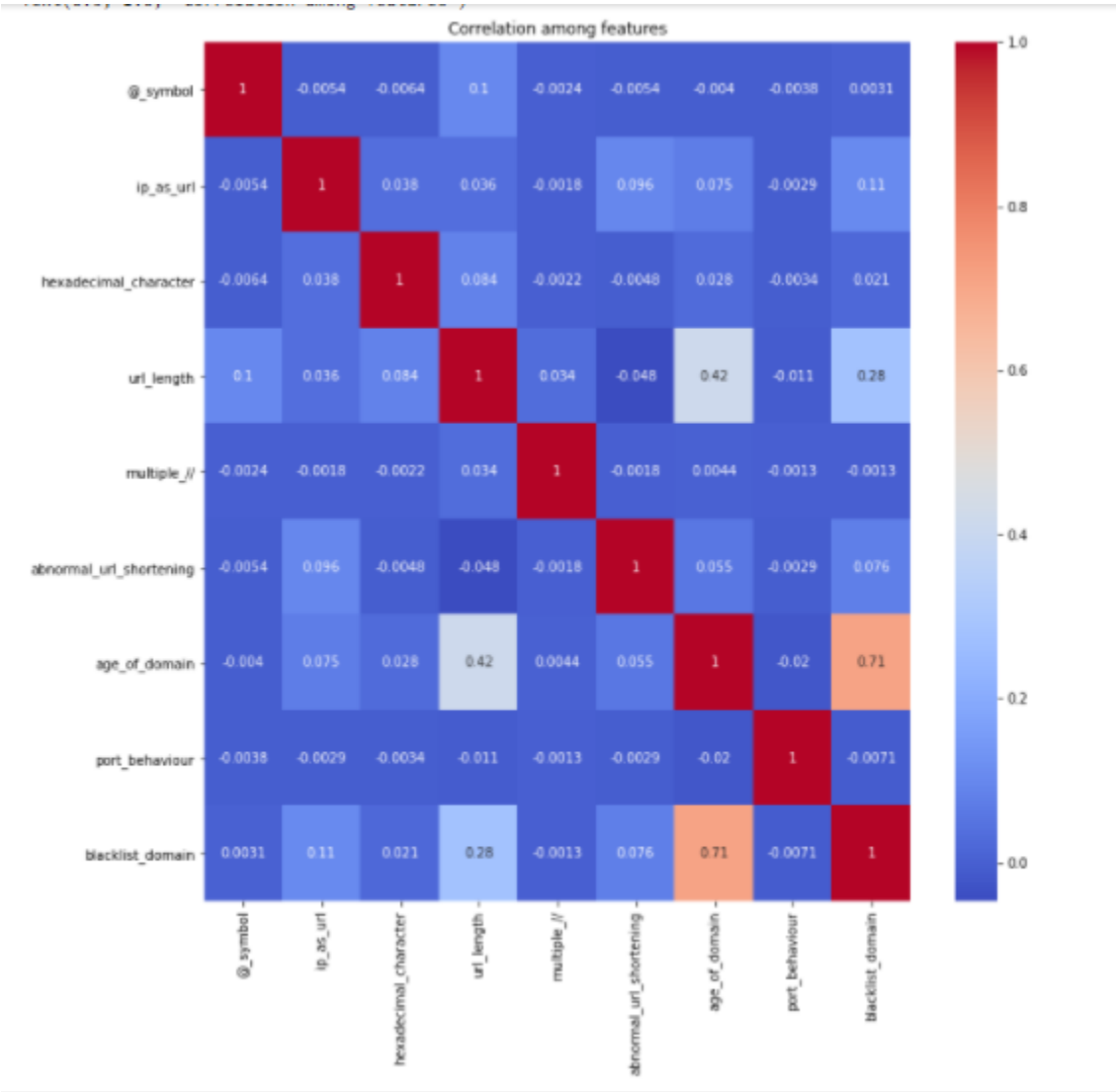
```
No of phishing websites signed by trusted 3rd party:  0
No of legitimate websites signed by trusted 3rd party:  0
```


## 15. Blacklisted domain:-

```
No of phishing urls  haveing action field of login form in blacklisted domain :  1055
No of legitimate urls haveing action field of login form in blacklisted domain:  272
```



## Correlation matrix:-



# Results and Conclusion

## ❖ Results

We have applied various classification algorithms like(**Naive bayes, Linear regression,SVM,K-nearest neighbor,Random forest**) since then we also did hyper-parameter tuning for finding the best parameter by using which we could get the optimal results.

| Algorithms | Test Accuracy |
|---|---|
| Linear regression | 90.91% |
| Support vector machine | 91.59% |
| K-nearest neighbor | 90.98% |
| Naive bayes | 88.61% |
| Random Forest | 90.98% |

## ❖ Conclusion

1. Five features out of 15(**Domain name check, The domain name in the path of a URL,Server Form Check (SFC)/Pop-Up Window ,Downloadable malicious code,SSL Certificate**) are not correlated with the dependent variable for these datasets. So we can remove these features while doing the classification.
2. Using **support vector machine** ,we got the best test accuracy around 91.6%.

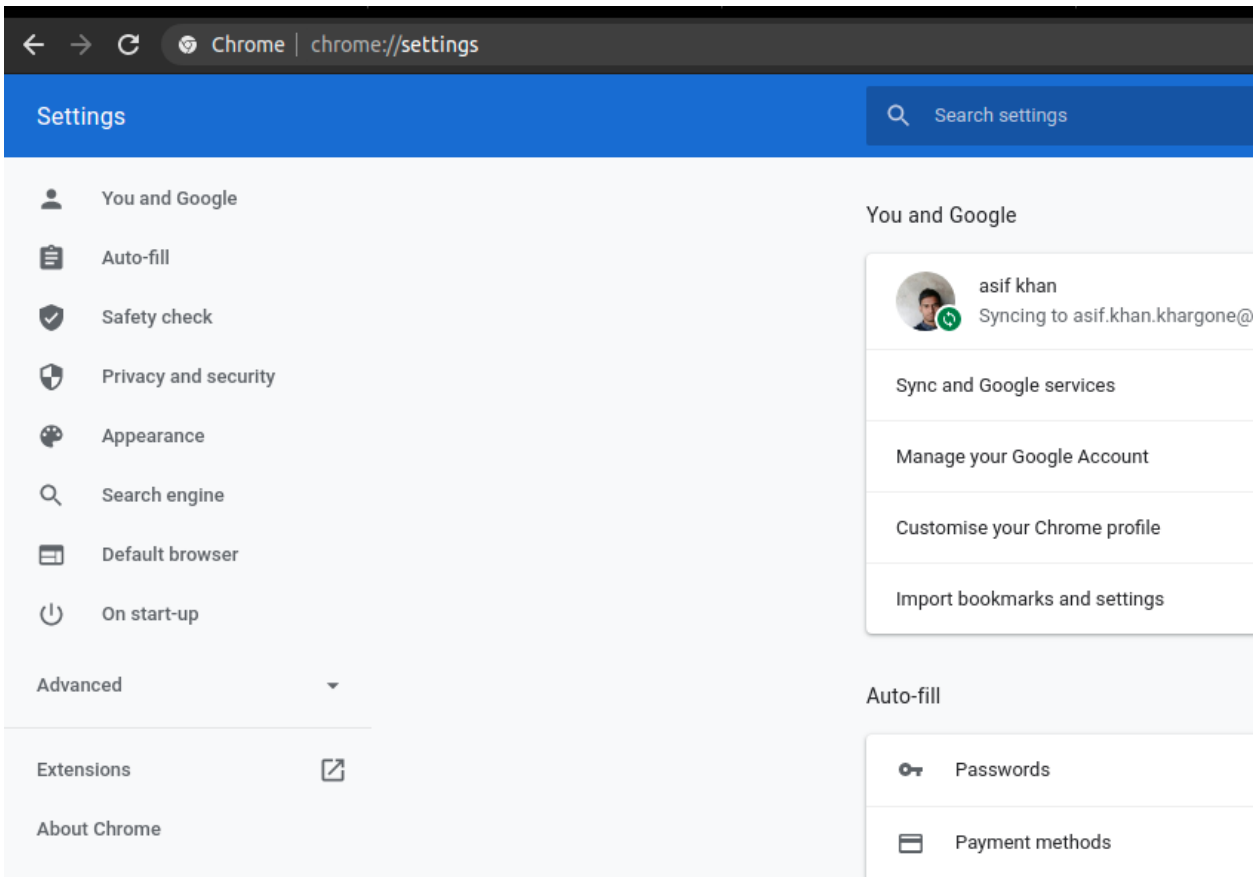
# Chrome Extension Installation

Two Components : Chrome Extension Frontend, Phishing API Backend

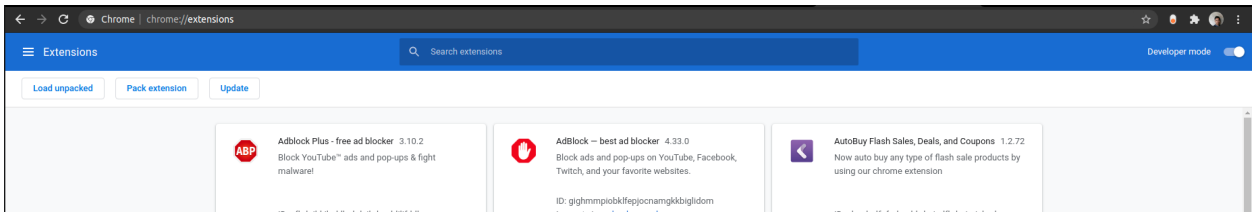
## 1st component : Chrome Extension Frontend

To Install the chrome extension :

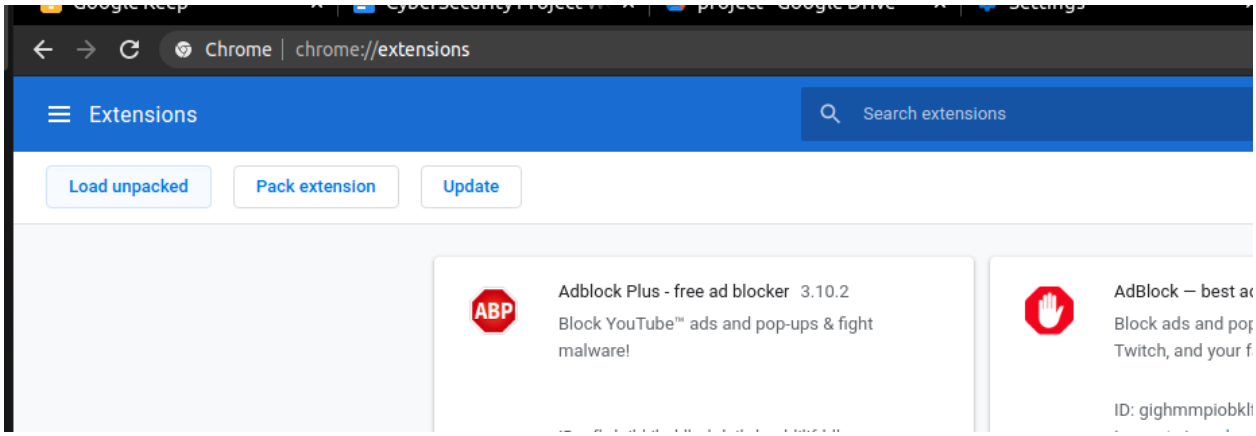
1. In chrome.
2. Open settings
3. Click Extensions



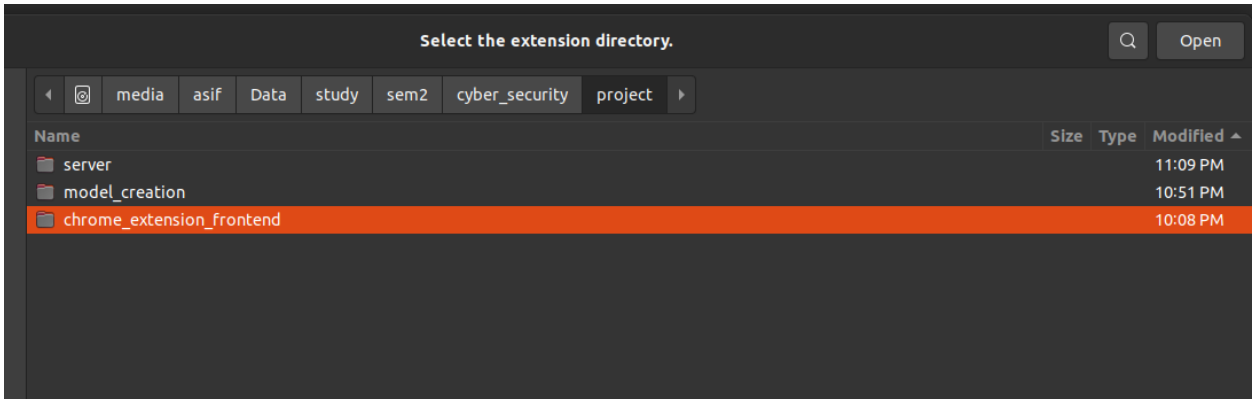
4. Enable Developer Mode



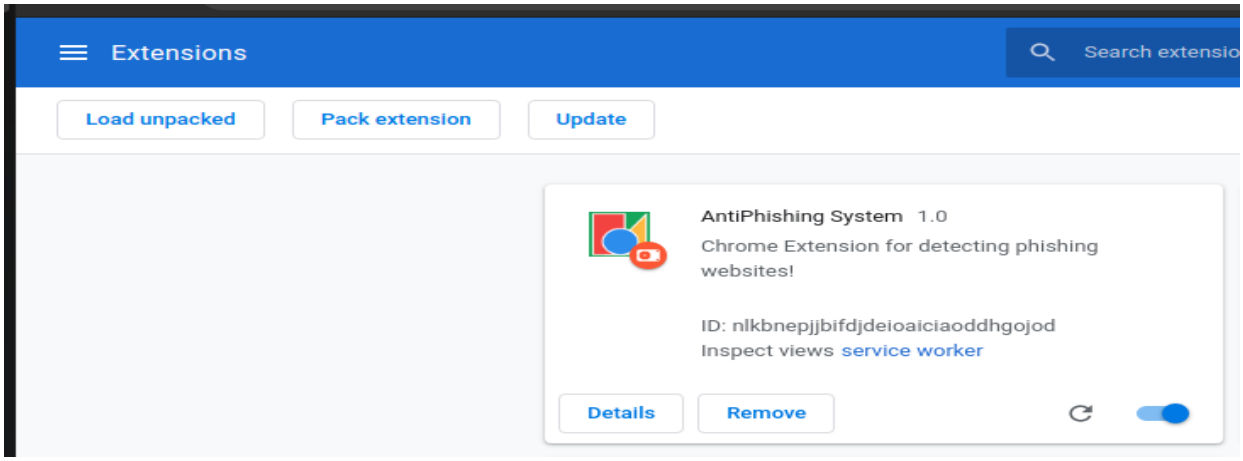

5. Load Unpacked




6. Load the CHROME_EXTENSION_FRONTEND folder.




7. And the chrome extension is installed in your system



# 2nd component : Phishing API Backend

It is a flask based api.

First install whois on your system

a. System dependencies
b. Python Virtual Environment Setup
c. Python Dependencies Install
d. Flask Variables Setup
e. Running Flask Server

## a. System dependencies

1. sudo apt-get install whois

```
Terminal:  Local  +
asif@asif-Predator-PH315-51:/media/asif/Data/study/sem2/cyber_security/project/server$ sudo apt-get install whois
[sudo] password for asif:
Reading package lists... Done
Building dependency tree
Reading state information... Done
whois is already the newest version (5.5.6).
The following packages were automatically installed and are no longer required:
  libfprint-2-tod1 libllvm10:i386 libnvidia-common-455
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 136 not upgraded.
asif@asif-Predator-PH315-51:/media/asif/Data/study/sem2/cyber_security/project/server$
```

2. sudo apt-get install netbase

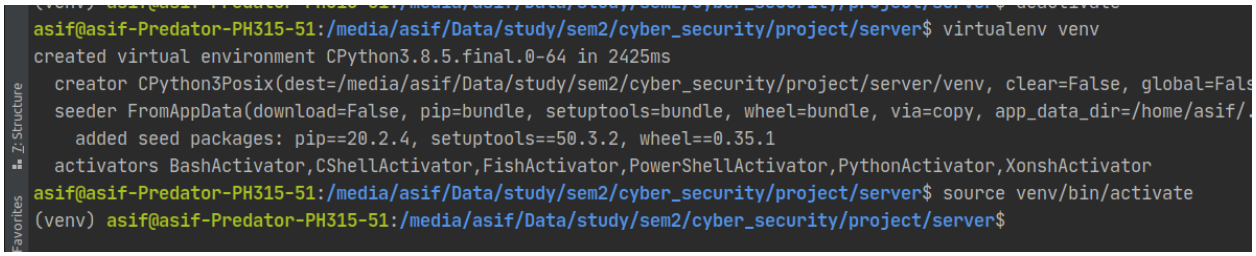
```
asif@asif-Predator-PH315-51:/media/asif/Data/study/sem2/cyber_security/project/server$ sudo apt-get install netbase
Reading package lists... Done
Building dependency tree
Reading state information... Done
netbase is already the newest version (6.1).
The following packages were automatically installed and are no longer required:
  libfprint-2-tod1 libllvm10:i386 libnvidia-common-455
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 136 not upgraded.
asif@asif-Predator-PH315-51:/media/asif/Data/study/sem2/cyber_security/project/server$
 TODO  6: Problems  Terminal  Python Console
 2: Favorites  7: Structure
```

b. Python Virtual Environment Setup

Prerequisites : Python 3 and virtualenv

Go inside project/server folder

1. virtualenv venv(Create virtual environment)
2. source venv/bin/activate (Activate virtual environment)

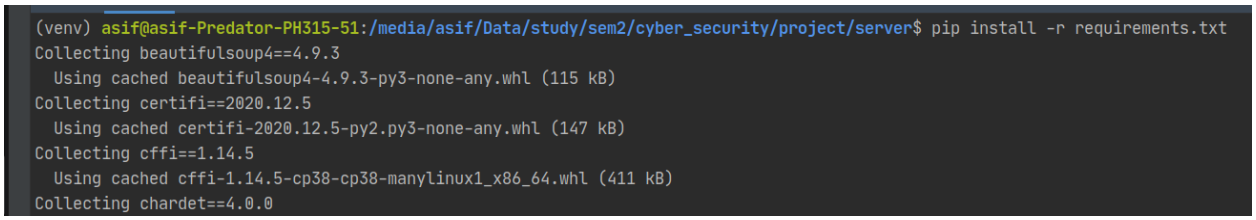



After this step virtual environment is created now we can install python packages.

c. Python Dependencies Install

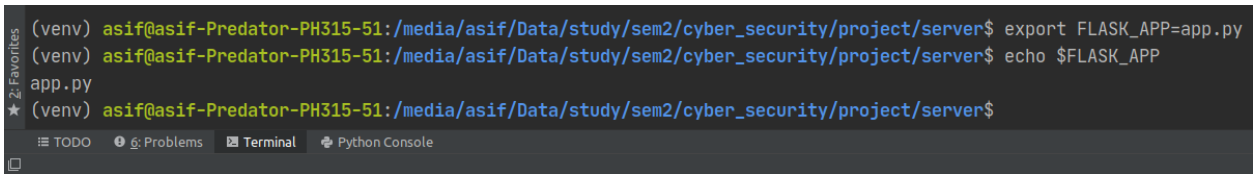
All the dependencies are list in project/server/requirements.txt

1. pip install -r requirements.txt
   It will install all the python package dependencies



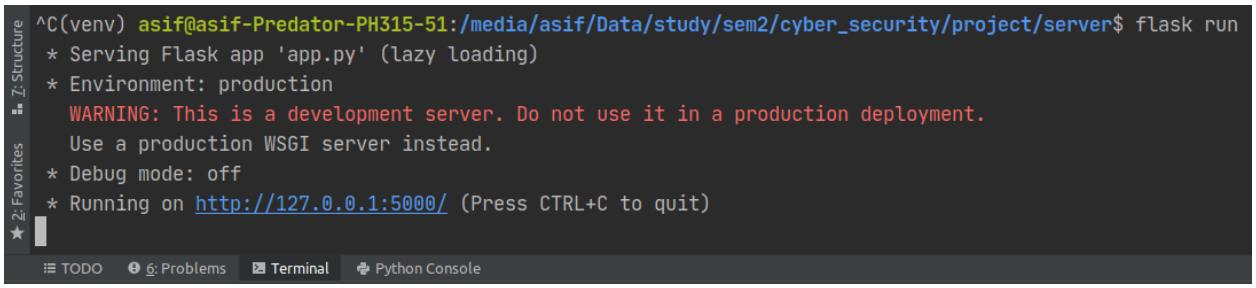
d. Flask Variables Setup

1. export FLASK_APP=app.py




e. Running Flask Server

1. flask run



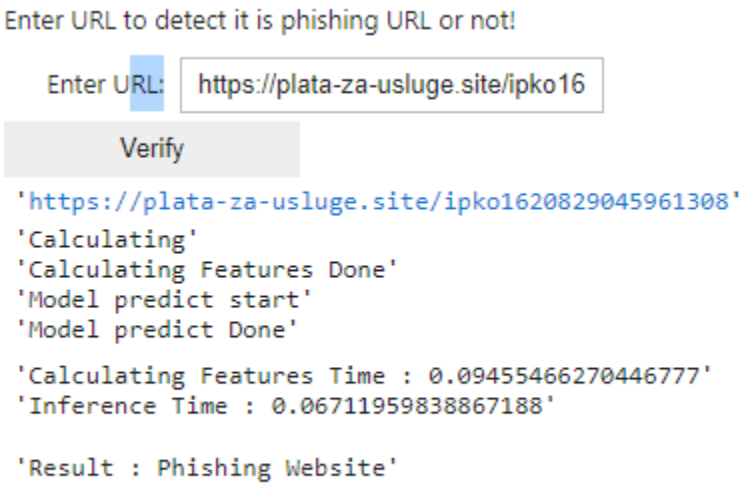

This will run flask server on port 5000. Please make sure port 5000 is not occupied by any other service.

AFTER THIS OUR FRONTEND AND BACKEND FOR CHROME EXTENSION IS INSTALLED AND RUNNING

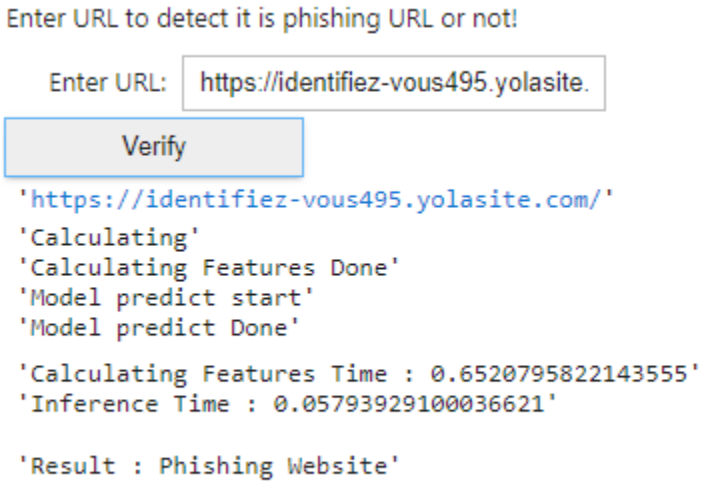
# Binder Demo

Here, We did try  with some phishing as well as benign  urls randomly and  got some results which are attached below.

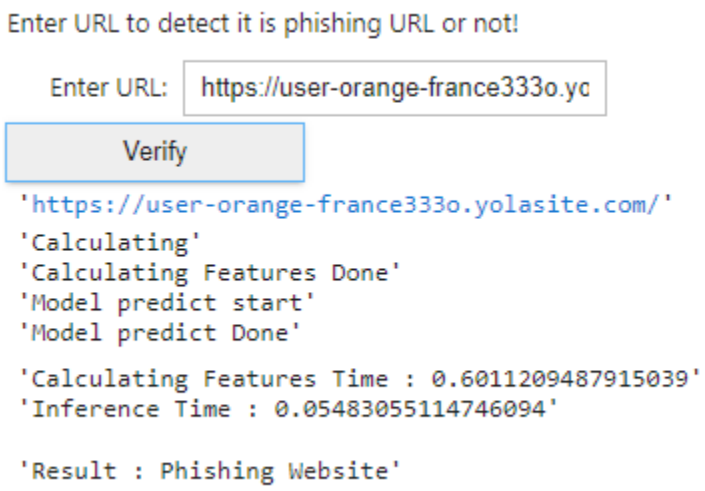
1. https://plata-za-usluge.site/ipko1620829045961308



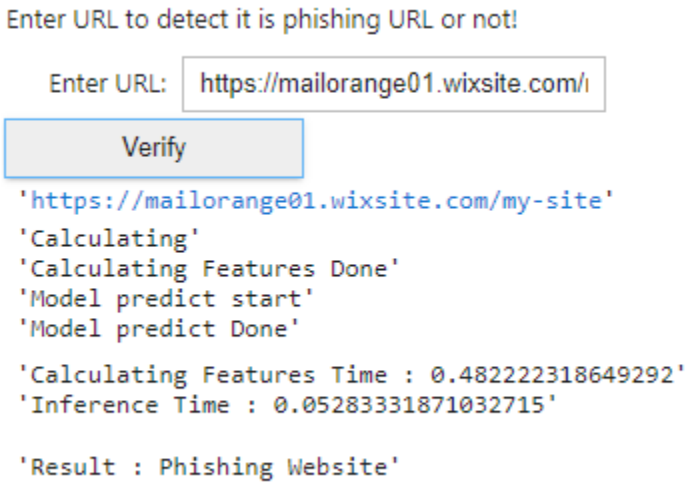

2. https://identifiez-vous495.yolasite.com/



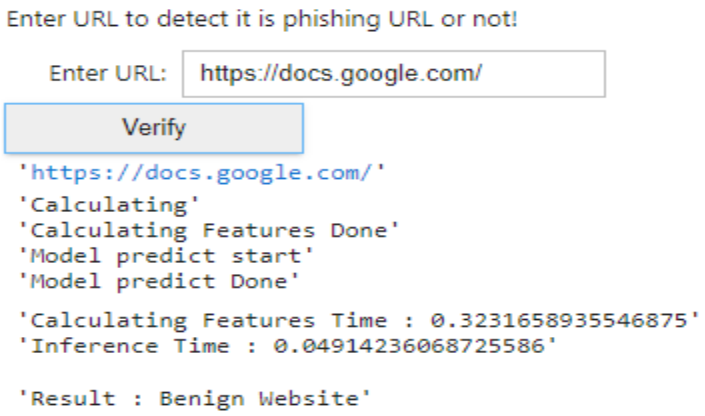
3. https://user-orange-france333o.yolasite.com/



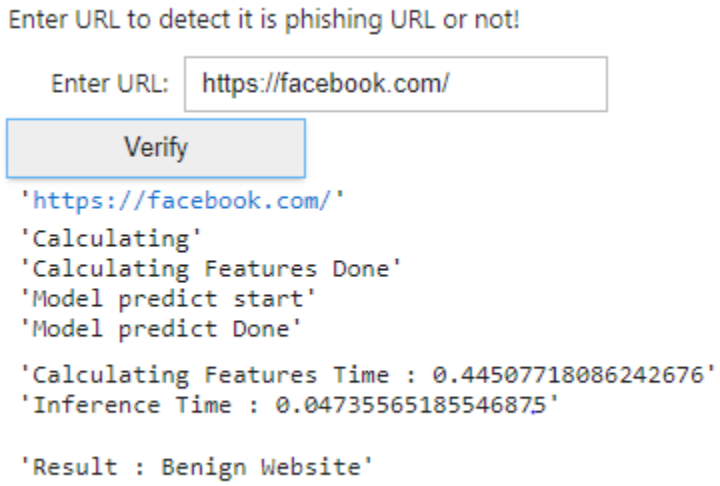

4. https://mailorange01.wixsite.com/my-site



5. https://docs.google.com/




6. https://facebook.com/



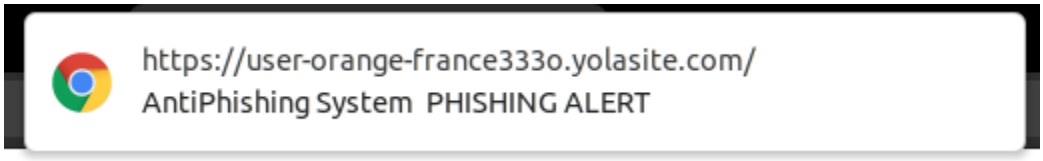

**Note:-** The link for our deployment using binder is given here. So you can play with some urls.

# Chrome Extension Demo

The chrome extension basically shows notification whenever we visit some phishing website, (no user intervention is required once the extension is involved).

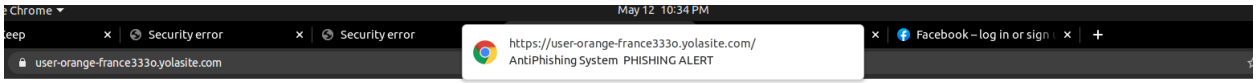
The lifespan of phishing websites are very short so the websites which we demo here may die. So they may not be alerted as phishing websites in future .

Notification looks like this




On top there is URL of the website you visited.
Then AntiPhishing System is name of the chrome extension
PHISHING ALERT is the notification.

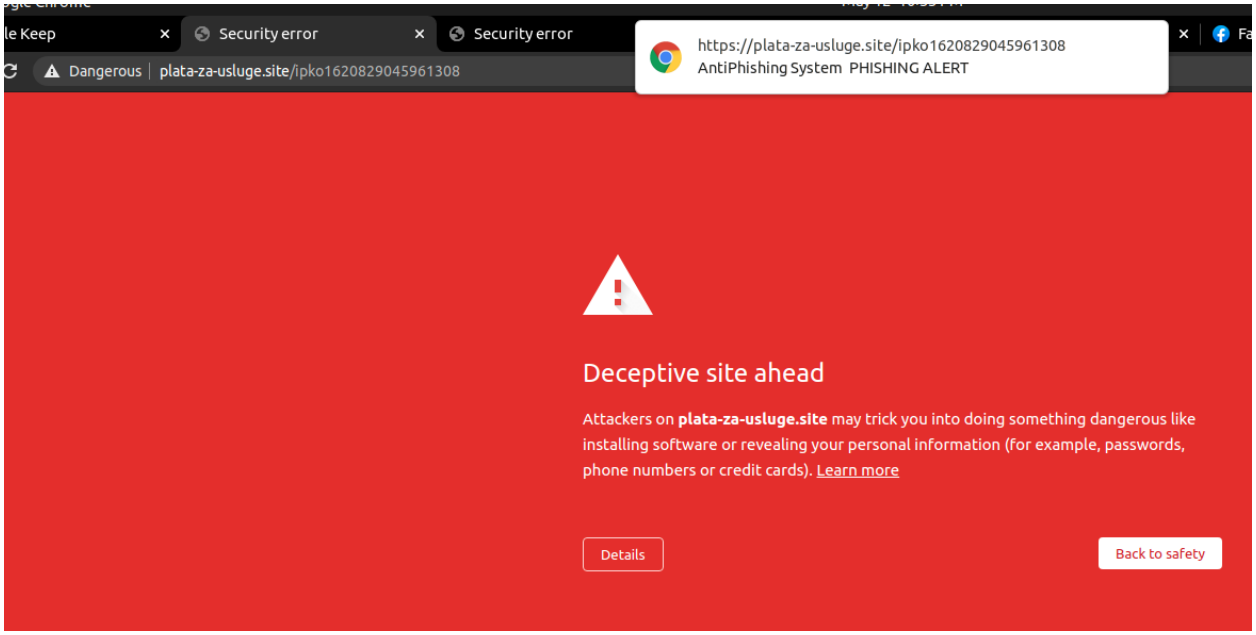
**Phishing Website Demo 1 :**



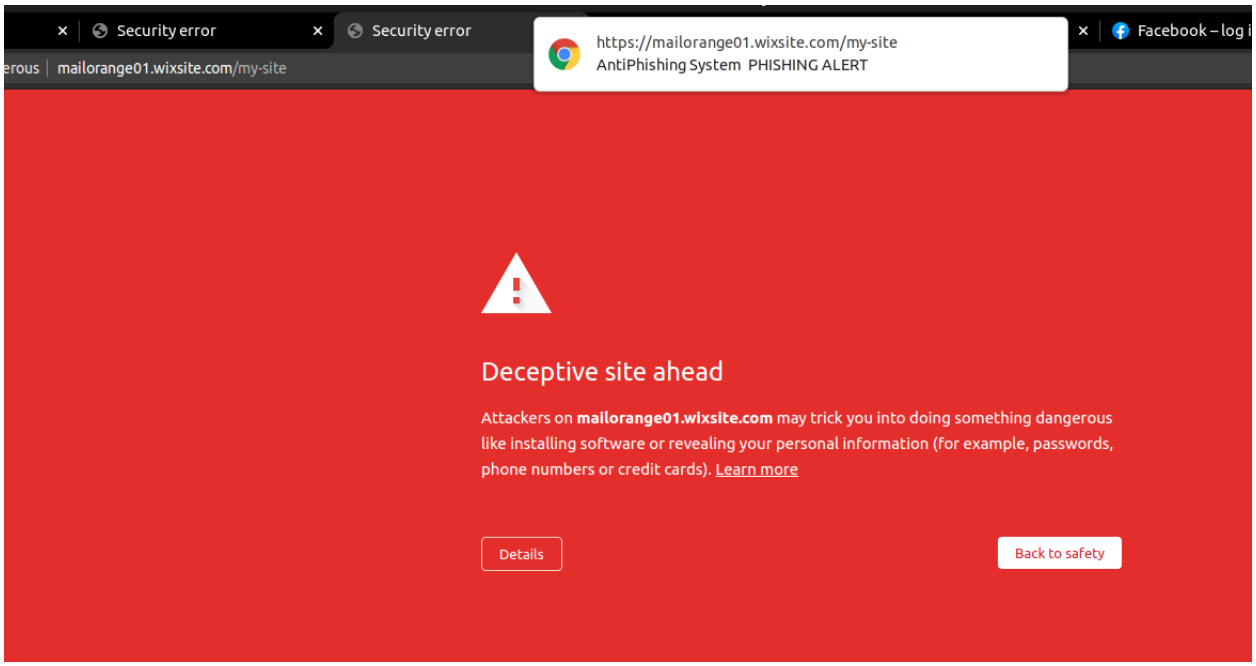

Oops!

Site disabled

## Phishing Website Demo 2.

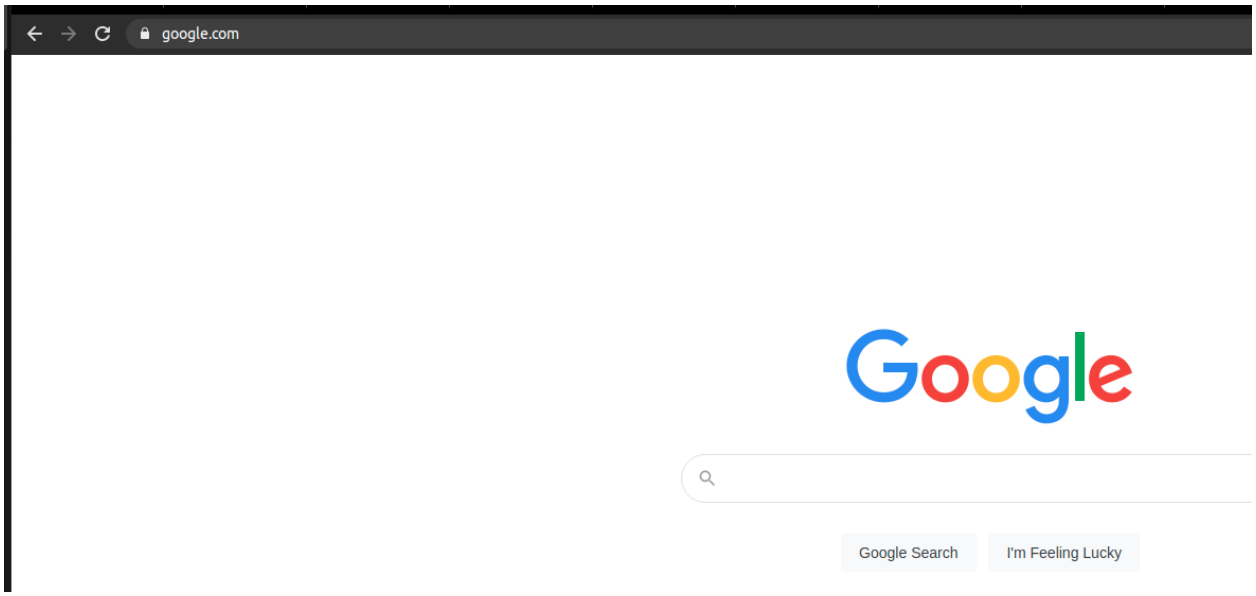



## Phishing Website Demo 3 :

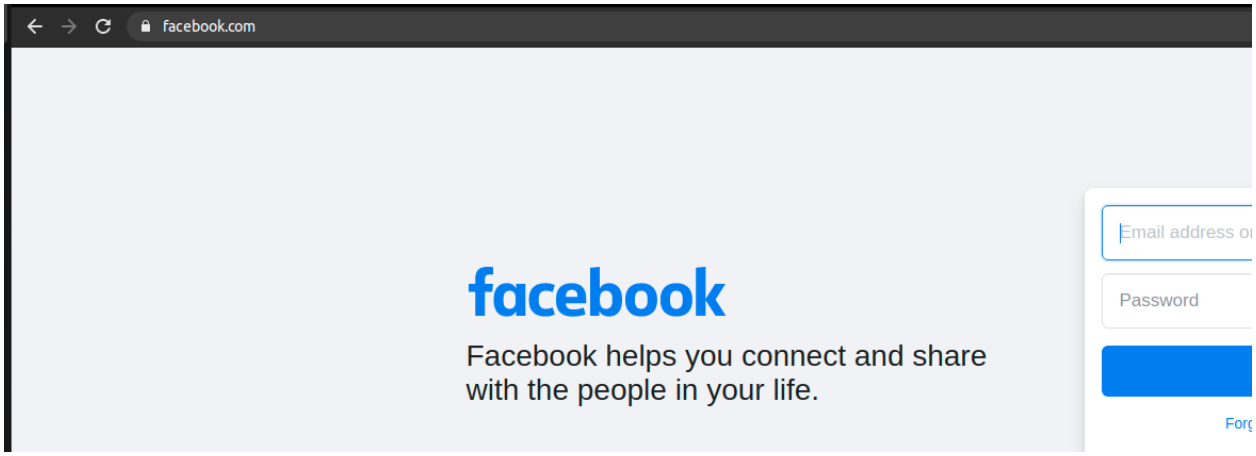



**On benign website there is no warning notification**

**Benign Website Demo 1:**




**Benign website Demo 2:**




On benign websites there is no notification for phishing websites as expected.

**Note:-** Google chrome has inbuilt a dangerous website detection tool. More details are given here. It also has similar working. It stops users by saying a deceptive site ahead.