ACCIDENT ANALYSIS AND PREVENTION

Exposure to pedestrian crash based on household survey data: Effect of trip purpose

Team Members:

Anvesh Muppeda(R11840667)
Saran Kumar(R11847468)
Sai Manasa Kota(R11863589)
Saketh Rao Vardhineni(R11840584)
Vishwasena Raidu Nyaramneni(R11849314)
Nikhilesh Peketi(R11841634)
Farooq Abdulla Shaik(R11842924)

Contents

Timeline taken to accomplish the Project at different phases:

- Phase one: March 2nd, 2023 (Intro to frontend & training the ML model).
- Phase two: March 15th, 2023 (Completion of the frontend by linking the CSS, HTML, and JS).
- Phase three: April 10th, 2023 (Storing the trained model and loading the trained model by using joblib and using flask to create UI).
- Final Phase: April 30th, 2023 (Fully working Project)

What has been done so far?

Backend:

In this project Accident Analysis and prevention, the objective is to examine the causes and consequences of accidents and to identify the measures to prevent them and introduce the different types of attacks and mitigated them.

We have a dataset of accidents, casualties, and vehicles in which,

Accidents contain:

Accident_Index,Longitude,Latitude,Police_Force,Accident_Severity,Number_of_Vehicles,Number_of_Casualties,Date,Day_of_Week,Time,Local_Authority_(District),Local_Authority_(Highway),1st_Road_Class,1st_Road_Number,Road_Type,Speed_limit,Junction_Detail, 2nd_Road_Number,Pedestrian_Crossing-Human_Control,Pedestrian_Crossing-Physical_Facilities,Light_Conditions,Weather_Conditions,Road_Surface_Conditions,Special_Conditions_at_Site,Carriageway_Hazards,Urban_or_Rural_Area,Did_Police_Officer_Attend_Scene_of_Accident

Casualties contain:

Accident_Index,Vehicle_Reference,Casualty_Reference,Casualty_Class,Sex_of_Casualty,Age_of_Casualty,Age_Band_of_Casualty,Casualty_Severity,Pedestrian_Location,Pedestrian_Movement,Car_Passenger,Bus_or_Coach_Passenger, Casualty_Type, Casualty_Home_Area_Type,Casualty_IMD_Decile.

Vehicles contain:

Accident_Index,Vehicle_Reference,Vehicle_Type,Towing_and_Articulation,Vehicle_Manoeuvre,Vehicle_Location-Restricted_Lane,Junction_Location,Skidding_and_Overturning,Hit_Object_in_Carriageway,Vehicle_Leaving_Carriageway,Hit_Object_off_Carriageway,1st_Point_of_Impact,Was_Vehicle_Left_Hand_Drive?,Journey_Purpose_of_Driver,Sex_of_Driver,Age_of_Driver,Age_Band_of_Driver,Engine_Capacity_(CC),Propulsion_Code,Age_of_Vehicle,Driver_IMD_Decile,Driver_Home_Area_Type,Vehicle_IMD_Decile.

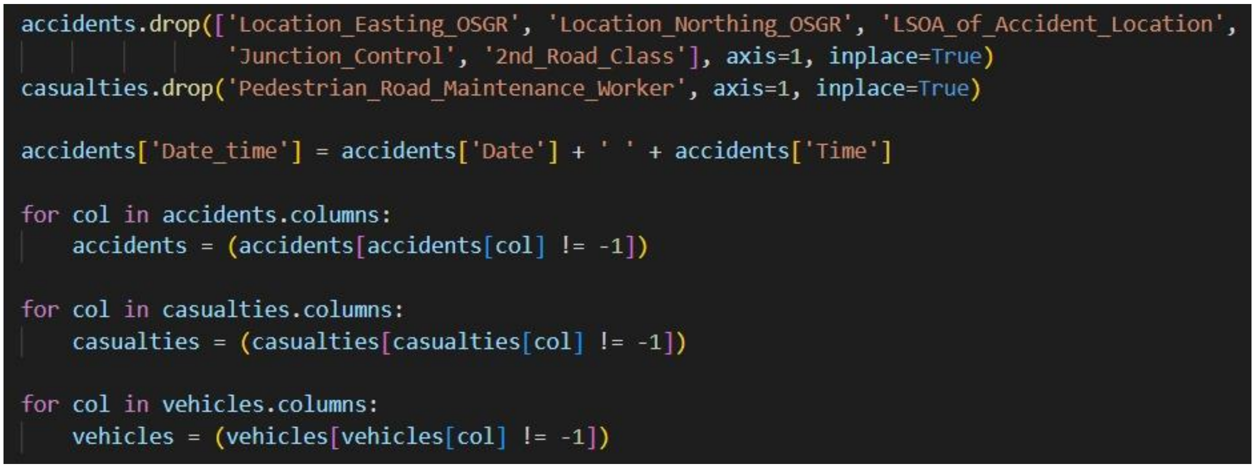
Then we have

- pre-processed the data:

  We have read the accidents, casualties, and vehicles by pd.read_csv and dropped the unwanted columns.

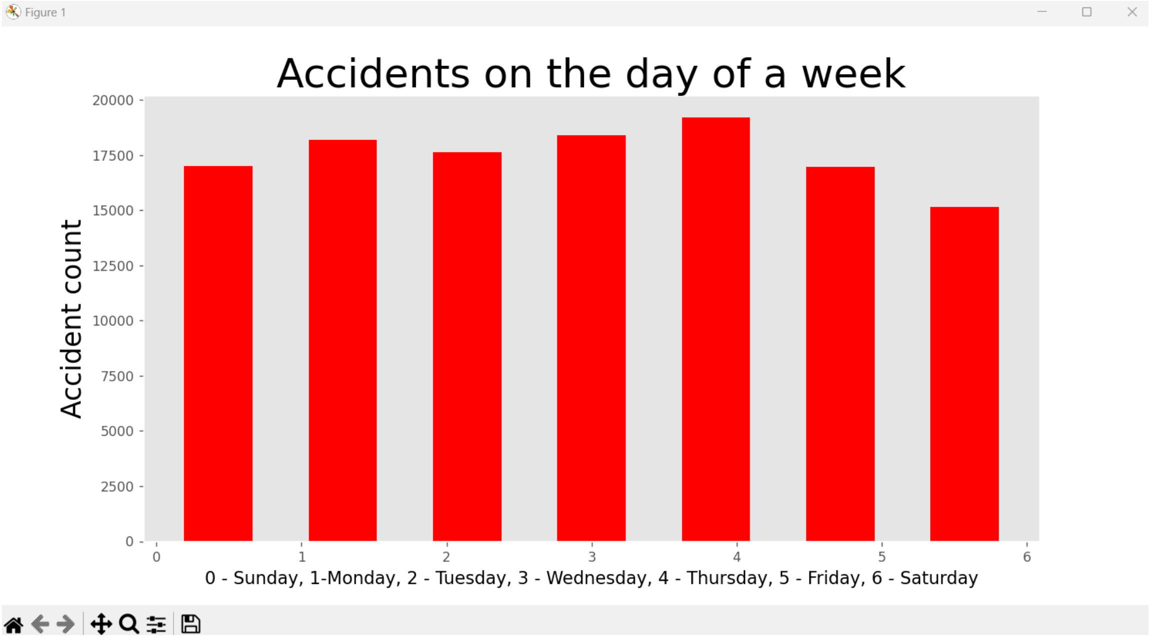
  Joining the date and time column in the accidents dataset by using pandas.

  Now we are removing the rows in the dataset which contains NULL values.

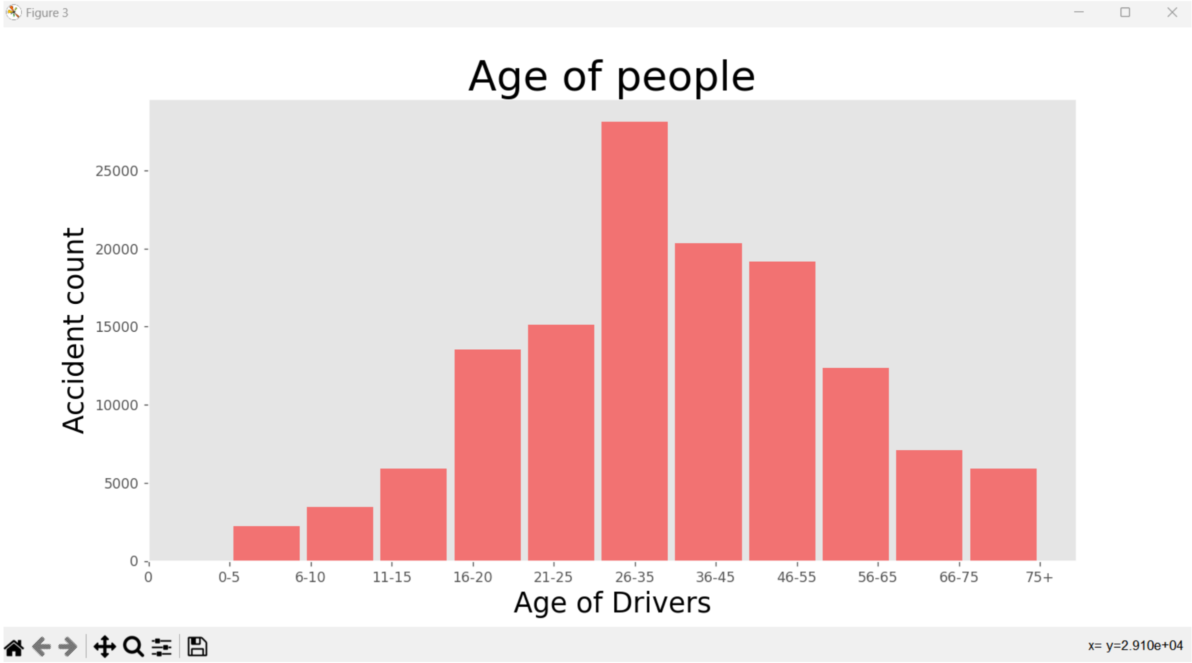


```
accidents.drop(['Location_Easting_OSGR', 'Location_Northing_OSGR', 'LSOA_of_Accident_Location',
                'Junction_Control', '2nd_Road_Class'], axis=1, inplace=True)
casualties.drop('Pedestrian_Road_Maintenance_Worker', axis=1, inplace=True)

accidents['Date_time'] = accidents['Date'] + ' ' + accidents['Time']

for col in accidents.columns:
    accidents = (accidents[accidents[col] != -1])

for col in casualties.columns:
    casualties = (casualties[casualties[col] != -1])

for col in vehicles.columns:
    vehicles = (vehicles[vehicles[col] != -1])
```

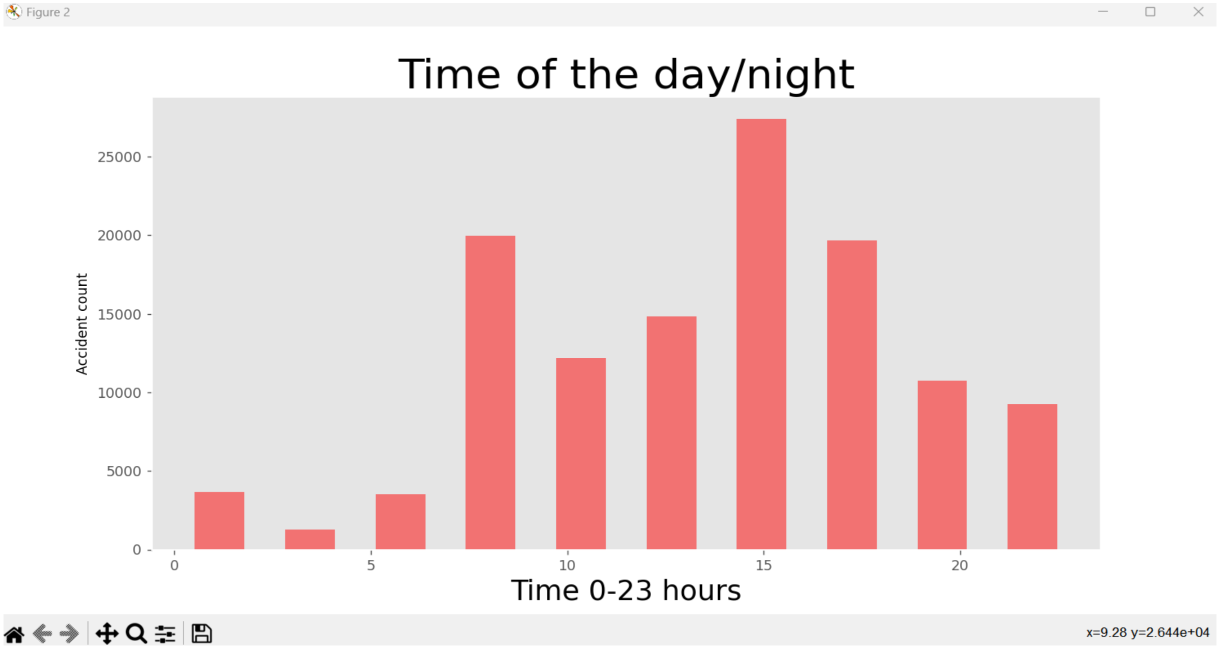
- Plotted a graph for accidents on the days of weeks.



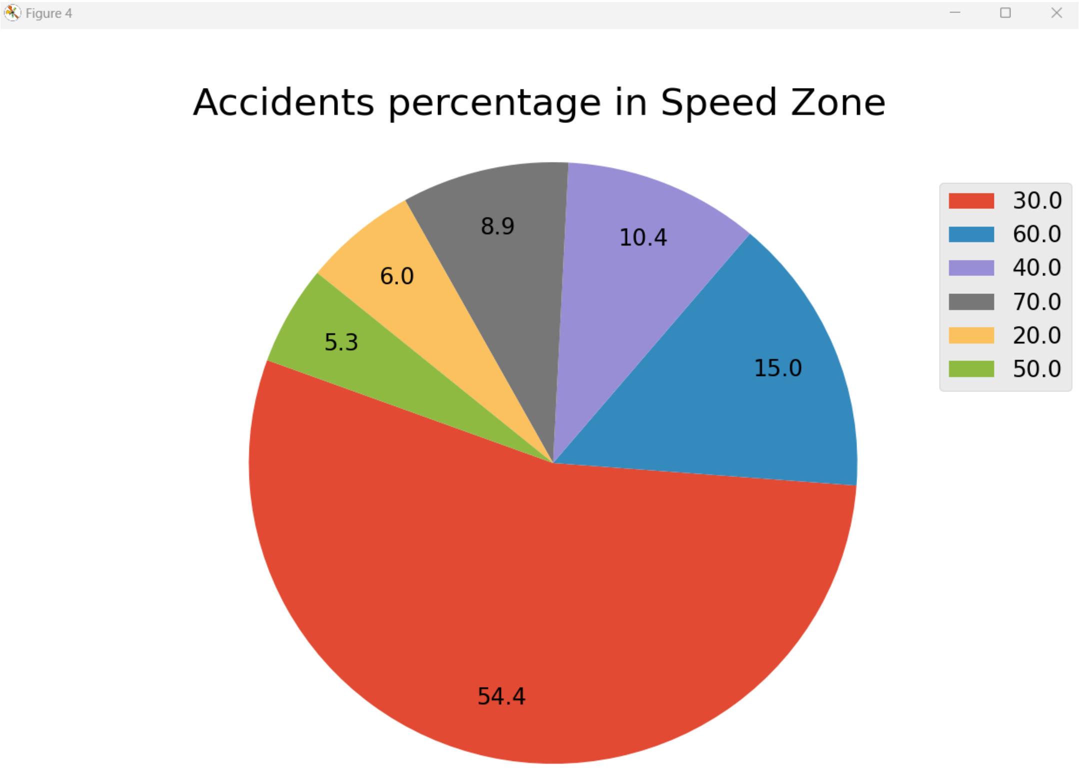
- Plotted a graph for accidents of different age groups.



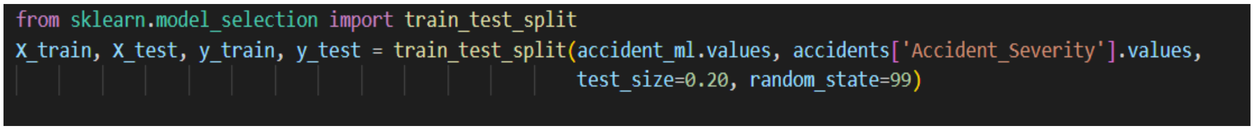
- Plotted a graph for accidents during the hours of the day.



- Plotted a graph for accident percentage in speed zone.

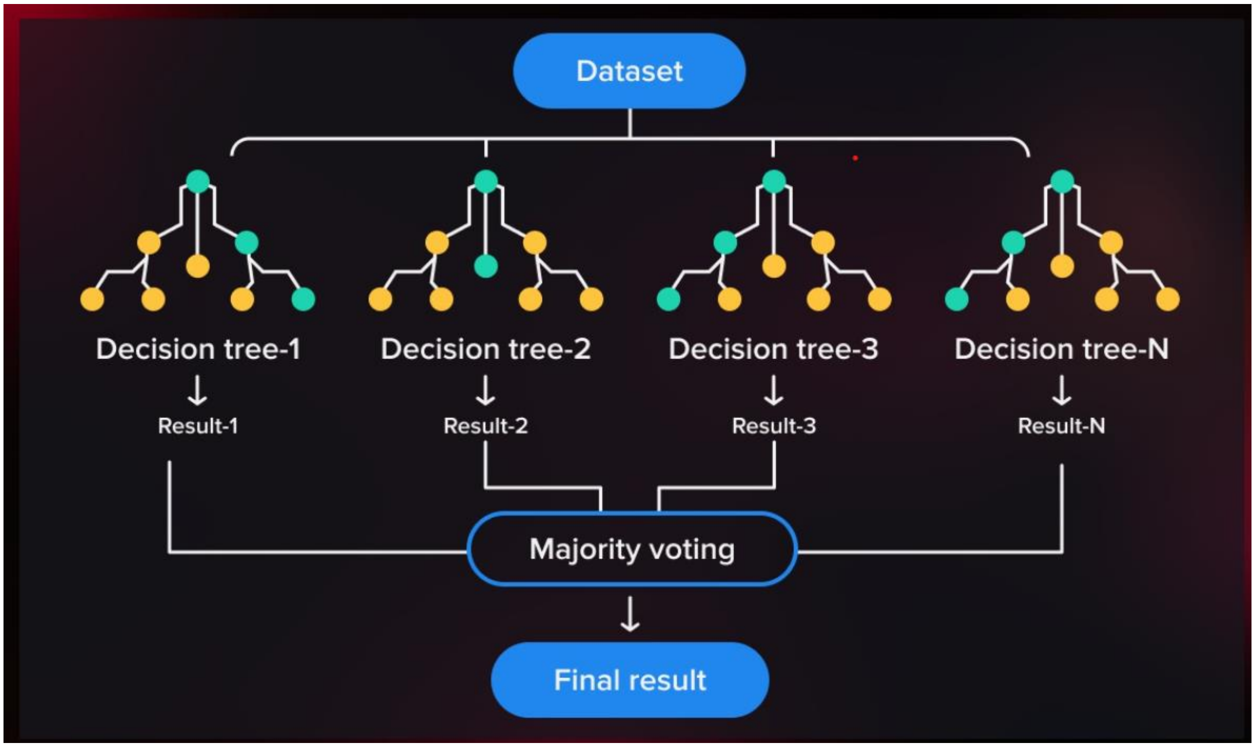


- We are creating the training data from accidents dataset which contains the columns Age_of_Driver, Vehicle_Type, Engine_Capacity_(CC), Day_of_Week, Weather_Conditions, Road_Surface_Conditions, Age_of_Vehicle, Light_Conditions, Sex_of_Driver, Speed_limit.
- Splitting the preprocessed data into 20 percent of test data and 80 percent of training data by using train_test_split.



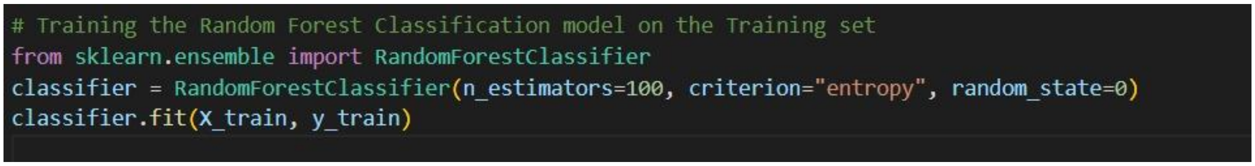
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(accident_ml.values, accidents['Accident_Severity'].values,
                                                    test_size=0.20, random_state=99)
```

- We are using the randomforestclassifier to train the model, so basically Random forest classifier is a bagging technique and a Supervised Machine Learning Algorithm that is used widely in **Classification and Regression problems**.
- random forest classifier is made up of numerous independent decision trees that work together as an ensemble. Each individual tree in the random forest spits out a class prediction, and the class with the most votes becomes our model's prediction.
- The data set contains a sample of rows and sample of features, initially we pick of some samples of rows called row sampling with replacement along with some columns called as feature sampling replacement sent to decision trees. So, the data sent to the decision tree i.e is row sampling with replacement along with some columns called as feature sampling replacement should be less than the total data set.
- Then the decision trees are trained on the data and when we send the test data the decision trees will be able to give the prediction. The predicted values are sent into the voting classifier where the prediction generated by most of the decision trees is the final prediction.



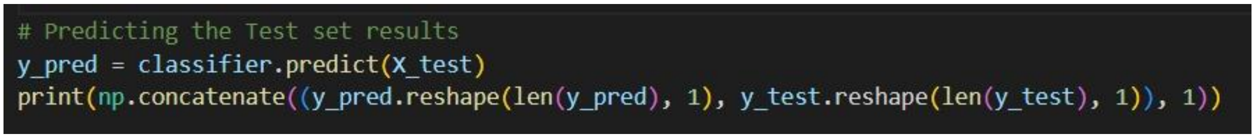
Where the decision tree has low bias and high variance which leads to overfitting of the data so to overcome this, we are using multiple decision trees.

- We have trained the random forest classifier model with the training data.



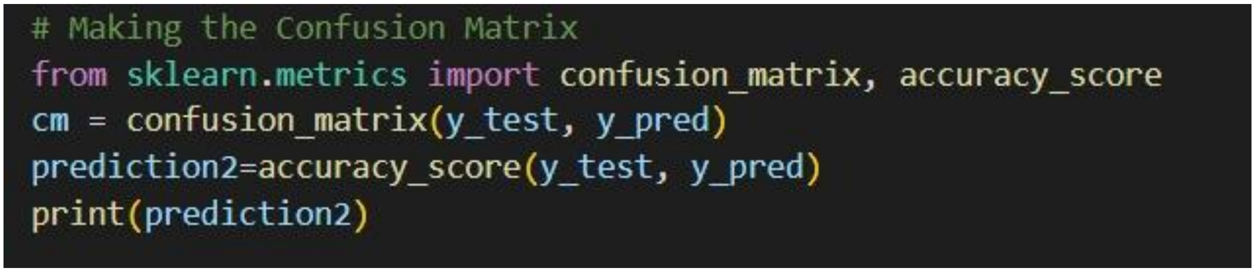
```
# Training the Random Forest Classification model on the Training set
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=100, criterion="entropy", random_state=0)
classifier.fit(X_train, y_train)
```

- We predicted the data by giving the test data as input to the model.



```
# Predicting the Test set results
y_pred = classifier.predict(X_test)
print(np.concatenate((y_pred.reshape(len(y_pred), 1), y_test.reshape(len(y_test), 1)), 1))
```

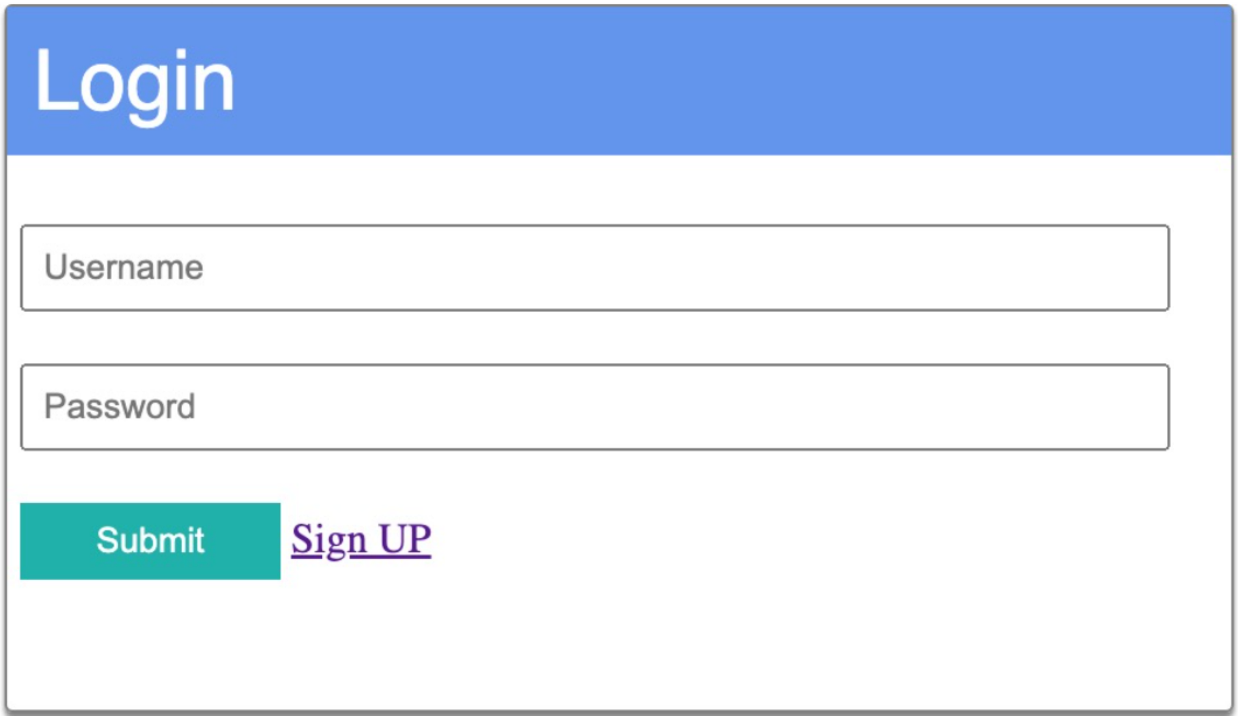
- Then we evaluated our model by finding accuracy. The accuracy of our model is 76%.



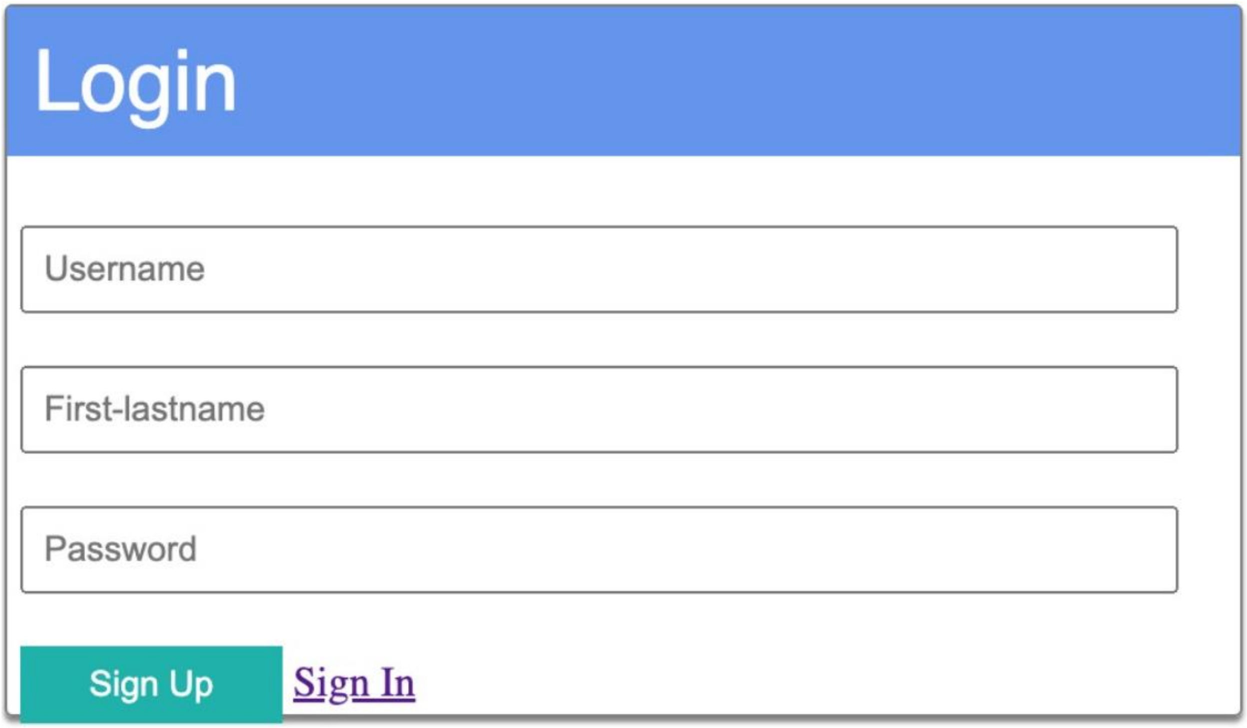
```
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
prediction2=accuracy_score(y_test, y_pred)
print(prediction2)
```

Front end:

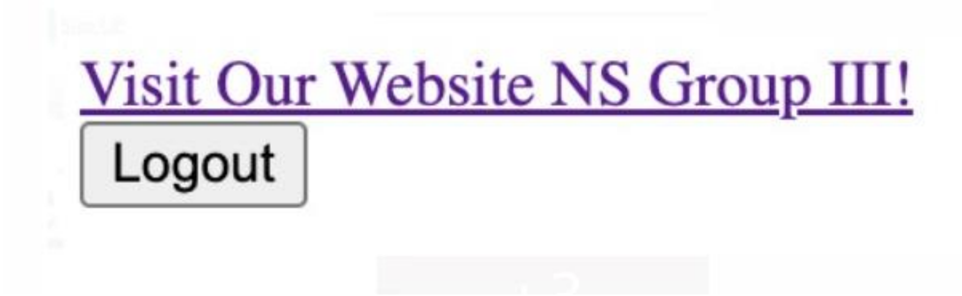
Written the front-end HTML code for the user interface in which we added the title of the page “Accident Analysis and Prevention”. Next few lines about how this website is going to detect the potential accidents in the About section.



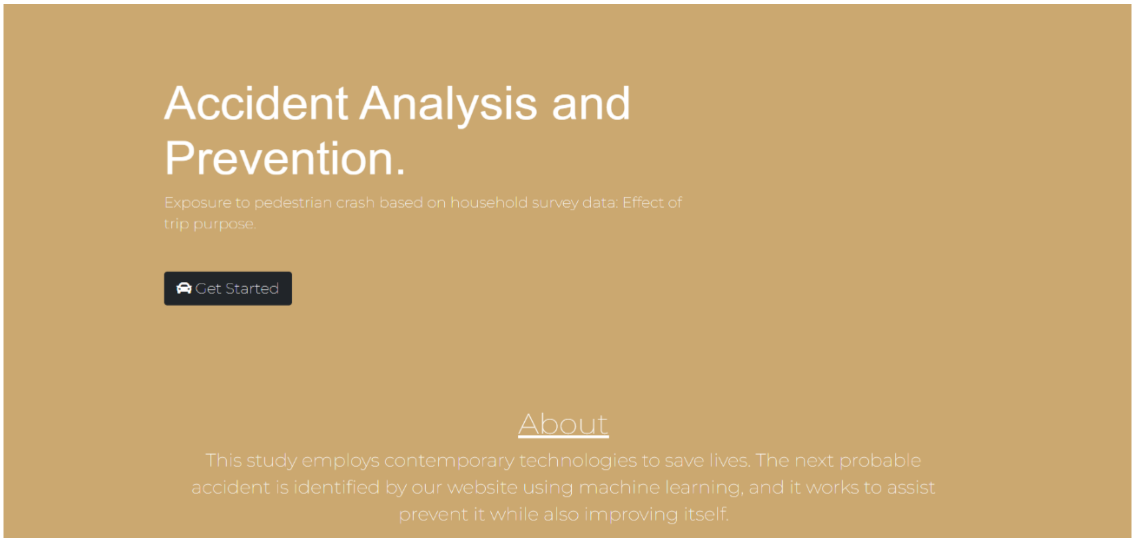
SIGN IN PAGE



SIGN UP PAGE



HOME PAGE



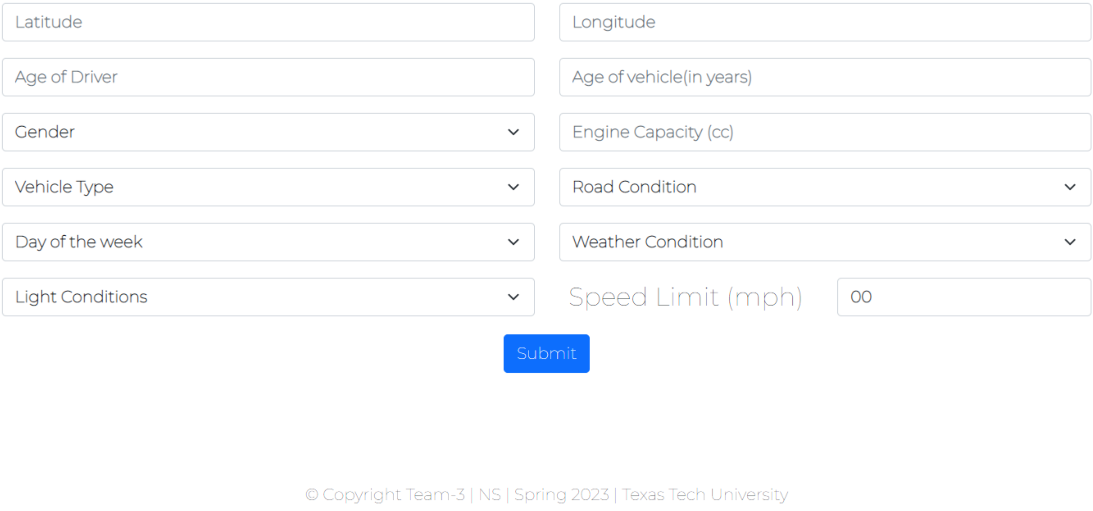
Accident Analysis and Prevention.

Exposure to pedestrian crash based on household survey data: Effect of trip purpose.

About

This study employs contemporary technologies to save lives. The next probable accident is identified by our website using machine learning, and it works to assist prevent it while also improving itself.

Then we added a few input fields which are used to predict accidents. Input fields include Age of the driver, gender, type of the vehicle, day of the week, age of the vehicle, Engine capacity (in CC), Road condition, weather condition and speed limit. Next there is a "submit" button upon clicking which will give the results about the accidents.



© Copyright Team-3 | NS | Spring 2023 | Texas Tech University

To this HTML page, we have added the styling with the Cascading Style sheet file which will give the proper look of the webpage.

Contributions:

ANVESH MUPPEDA: Networking, automation, Splunk setup

SPLUNK:

It is software which processes and brings out an insight from machine data and the other forms of big data.

It is a one-step solution as it automatically pulls out the data from various sources and accepts the data in any format i.e., structured, or semi-structured.

It is the easiest tool to install and allows various operations like – searching, analyzing, reporting, and visualizing system data.

Functionalities of the Splunk are:

- Analyzes system performance.
- Monitor business metrics.
- Troubleshoots any failure condition.
- Create dashboards to visualize and analyze the results.
- Search & investigate a specific outcome.

It has 2 types of components –

- Processing Components: Forwarder, Indexer, and Search Head.
- Management Components: License Master, Monitoring Console, Deployment Server, Indexer Cluster Master Node, and Search Head Cluster Deployer.

Splunk Universal Forwarder:

Splunk universal forwarder is free to the individuals to use.

Helpnet uses the Splunk Universal Forwarder, to gather data from different sources of inputs and forward it to the machine data of Splunk Indexers.

Thus, a central repository is achieved and the data is available for searching.

The universal forwarders communicate with the deployment services.

The data that is been forwarded will be encrypted to the indexers.

The use of Splunk Universal Forwarder is designed and framed in such a way that it can run on production servers where it will have minimal CPU and memory usage.

The best practices of Splunk Universal Forwarder are:

1. Use the universal forwarder as a data collection method when it is possible.

2. Perfect use of start and stop of universal forwarder can be controlled from the command line interface.

In general, Splunk Universal Forwarder is only used, or its primary purpose is to send or forward the data flow from different inputs.

Installing Splunk

```
sudo wget -O splunk-9.0.2-17e00c557dc1-Linux-x86_64.tgz
https://download.splunk.com/products/splunk/releases/9.0.2/linux/splunk-9.0.2-17e00c557dc1-Linux-x86_64.tgz
```

```
tar xvzf file.tgz -C /opt
```

```
/opt/splunk/bin/splunk start --accept-license
```

Opening port 8000 for splunk since splunk is running on port 8000

```
firewall-cmd --zone=public --add-port=8000/tcp --permanent
firewall-cmd --reload
```

installing the splunk forward

```
$wget -O splunkforwarder-9.0.2-17e00c557dc1-Linux-x86_64.tgz
"https://download.splunk.com/products/universalforwarder/releases/9.0.2/linux/splunkforwarder-9.0.2-17e00c557dc1-Linux-x86_64.tgz"
$tar -xvzf splunkforwarder-9.0.2-17e00c557dc1-Linux-x86_64.tgz -C /opt/
$sudo /opt/splunkforwarder/bin/splunk start --accept-license
```

NOTE: 8089 already in use, so better use 8088

splunk has limit on disk space for root directory search not executed: The minimum free disk space (5000MB) reached for /opt/splunk/var/run/splunk/dispatch

If we get above error, then we need to change the limit as required in below path

```
$/opt/splunkforwarder/etc/system/default/server.conf
```

Enable the boot start

```
$/opt/splunkforwarder/bin/splunk enable boot-start
```

Add the Splunk IP where you want to forward the logs

```
$/opt/splunkforwarder/bin/splunk add forward-server 18.117.242.146:9997
```

Sending logs to splunk

```
$/opt/splunkforwarder/bin/splunk add monitor /etc/httpd/logs/access_log -index main
```

To check the above configuration

```
/opt/splunkforwarder/etc/apps/search/local/inputs.conf
```

Enable the forward ports

```
$firewall-cmd --zone=public --permanent --add-port=9997/tcp
```

```
$firewall-cmd --zone=public --permanent --add-port=9997/udp
```

```
$firewall-cmd --zone=public --permanent --add-port=8088/tcp
```

```
$firewall-cmd --zone=public --permanent --add-port=8088/udp
```

Add the IP on splunk

Follow the below

Settings -> Data -> Forwarding and Receiving -> Receive data

Click on "+ Add new" Button
Then add the 9997 port. Then restart the splunk and splunk global forwarder.

Delete the index data from the splunk UI

```
$index=main sourcetype=syslog | delete
```

After restart of the server.

1. Restart the spunk on splunk forwarder
2. Update the new IP

   ```
   $/opt/splunkforwarder/bin/splunk add forward-server 18.117.242.146:9997
   ```

3. Restart the splunk-on-splunk forwarder.

NACL:

Network Access Control List.

It provides a layer of security to the amazon web services.

Two kinds of NACL - Customized and default.

Multiple subnets can be bound with a single NACL, but one subnet can be bound with a single NACL only, at a time.

It can be understood as the firewall or protection for the subnet.

It is stateless, meaning any change applied to an incoming rule isn’t automatically applied to an outgoing rule.

It is the second layer of defense, which helps in protecting the AWS stack.

It is an optional layer for VPC, which adds another security layer to the service.

This means every rule is evaluated based on the priority it has.

The rules are applied in the order of their priority, where priority is indicated by the number the rule is assigned.

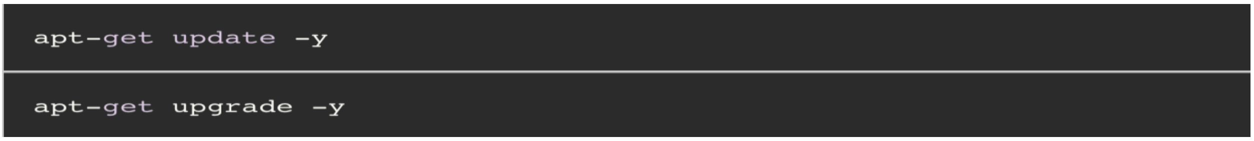
It can be used to support as well as deny rules.

Denial of rules can be explicitly mentioned, so that when the layer sees a specific IP address, it blocks connecting to it.

Example: If a request comes through port 80, it should be explicitly indicated that its outgoing response would be the same port 80.

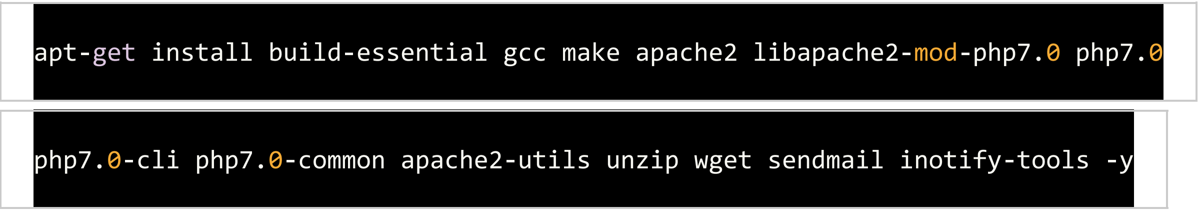
SAI MANASA KOTA: OSSEC , DDOS mitigation

First, we must update your system with the latest stable version. we can do this with the following command:



```
apt-get update -y
apt-get upgrade -y
```

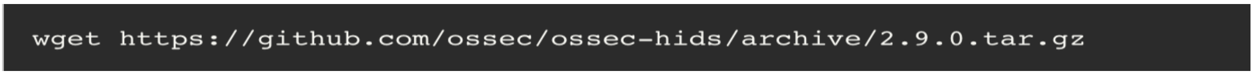
OSSEC requires gcc, libc, apache and PHP. We must install all these packages with the following command:



```
apt-get install build-essential gcc make apache2 libapache2-mod-php7.0 php7.0
php7.0-cli php7.0-common apache2-utils unzip wget sendmail inotify-tools -y
```

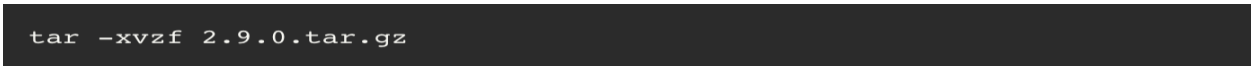
Install OSSEC

Then we must download the latest version of the OSSEC from GitHub repository with the following command:



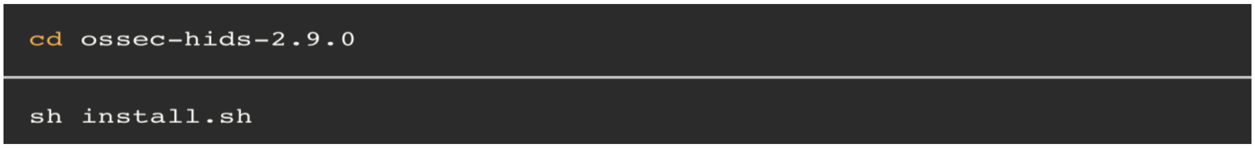
```
wget https://github.com/ossec/ossec-hids/archive/2.9.0.tar.gz
```

Once the download is completed, then extract the downloaded file with the following command:



```
tar -xvzf 2.9.0.tar.gz
```

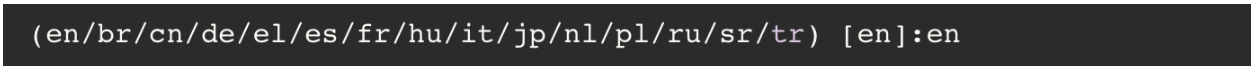
Next, change the directory to the extracted directory, then run `install.sh` to install OSSEC:



```
cd ossec-hids-2.9.0
sh install.sh
```

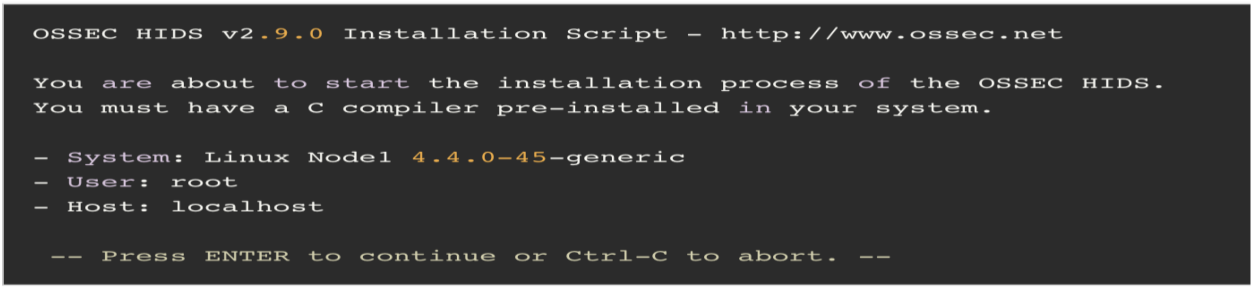
Then we'll be prompted to answer some questions,

Select your language, if your language is English then type en and press Enter:



```
(en/br/cn/de/el/es/fr/hu/it/jp/nl/pl/ru/sr/tr) [en]:en
```

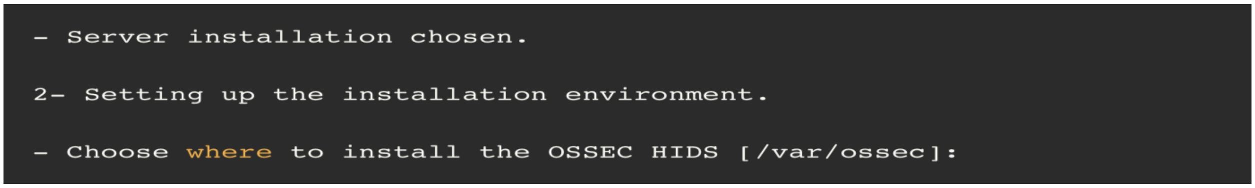
Then we'll see the following output:



```
OSSEC HIDS v2.9.0 Installation Script - http://www.ossec.net

You are about to start the installation process of the OSSEC HIDS.
You must have a C compiler pre-installed in your system.

- System: Linux Node1 4.4.0-45-generic
- User: root
- Host: localhost

 -- Press ENTER to continue or Ctrl-C to abort. --
```

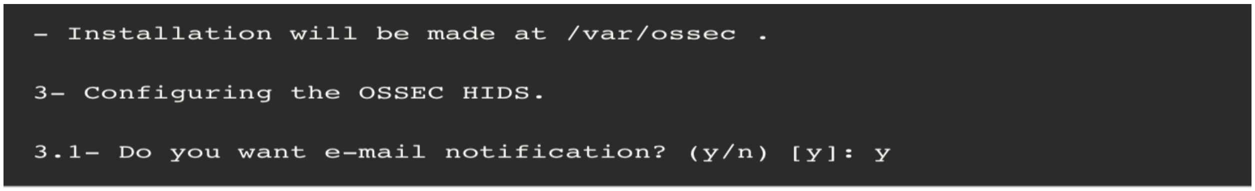
Now press enter, and we'll see the following output:

1- What kind of installation do you want (server, agent, local, hybrid or help)? Local
Choose local to monitor the server it has been installed on then press Enter:



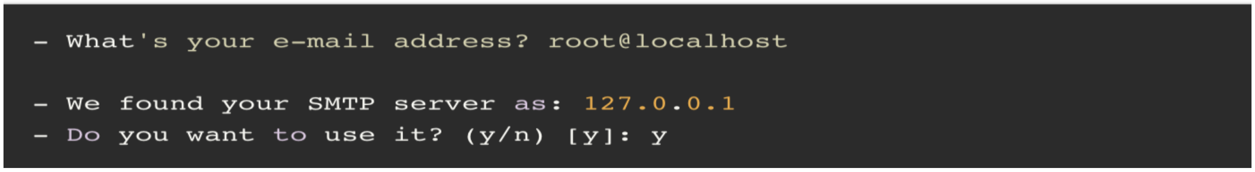
```
- Server installation chosen.

2- Setting up the installation environment.

- Choose where to install the OSSEC HIDS [/var/ossec]:
```

Choose OSSEC install location and press Enter:



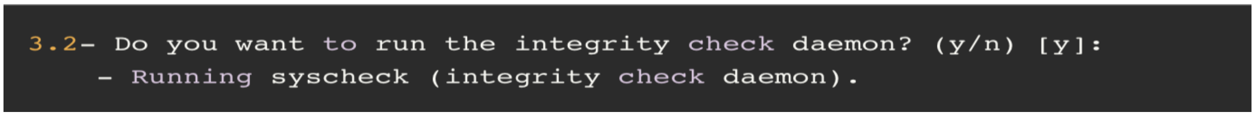
```
- Installation will be made at /var/ossec .

3- Configuring the OSSEC HIDS.

3.1- Do you want e-mail notification? (y/n) [y]: y
```

Type y and press Enter if you want to get e-mail notification:



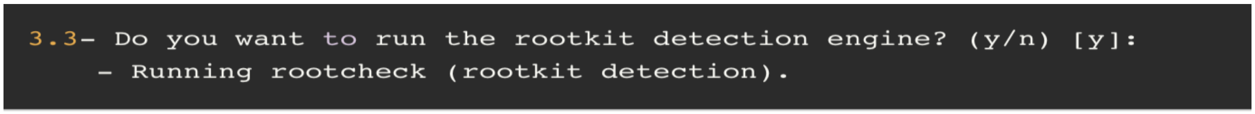
```
- What's your e-mail address? root@localhost

- We found your SMTP server as: 127.0.0.1
- Do you want to use it? (y/n) [y]: y
```

Type your local e-mail address and press Enter:



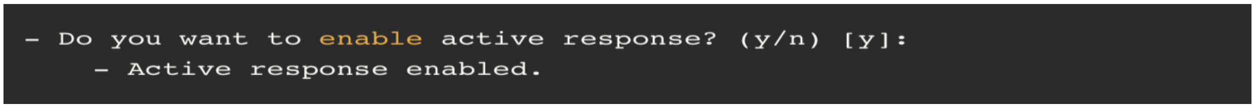
```
3.2- Do you want to run the integrity check daemon? (y/n) [y]:
   - Running syscheck (integrity check daemon).
```

Press Enter for integrity check daemon:



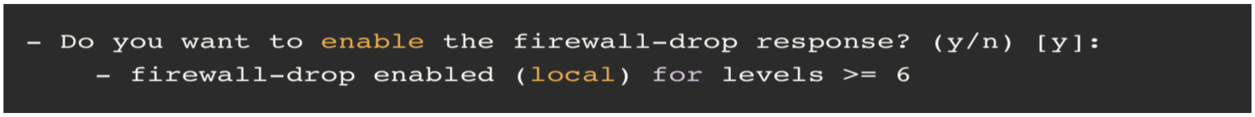
```
3.3- Do you want to run the rootkit detection engine? (y/n) [y]:
   - Running rootcheck (rootkit detection).
```

Press Enter for rootkit detection engine:



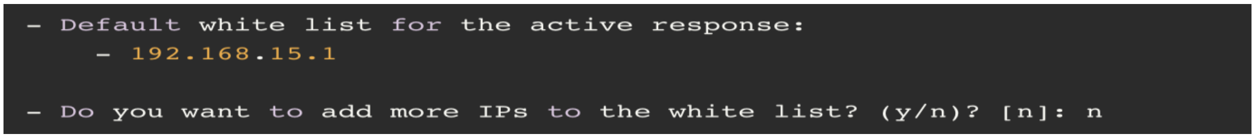
```
- Do you want to enable active response? (y/n) [y]:
   - Active response enabled.
```

Press Enter to enable active response:



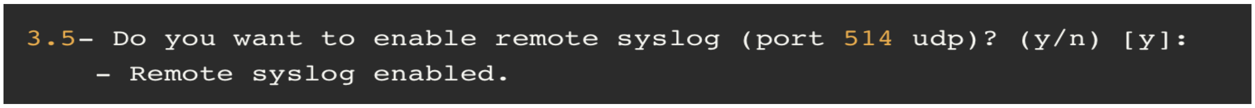
```
- Do you want to enable the firewall-drop response? (y/n) [y]:
    - firewall-drop enabled (local) for levels >= 6
```

Press Enter to enable the firewall-drop response:



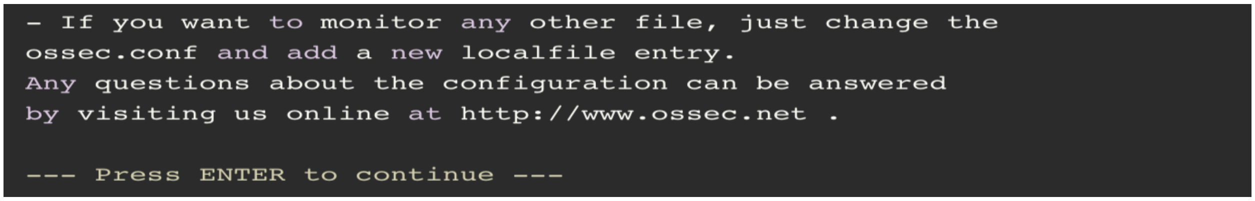
```
- Default white list for the active response:
    - 192.168.15.1

- Do you want to add more IPs to the white list? (y/n)? [n]: n
```

Type n and press Enter if you don’t want to add white list:



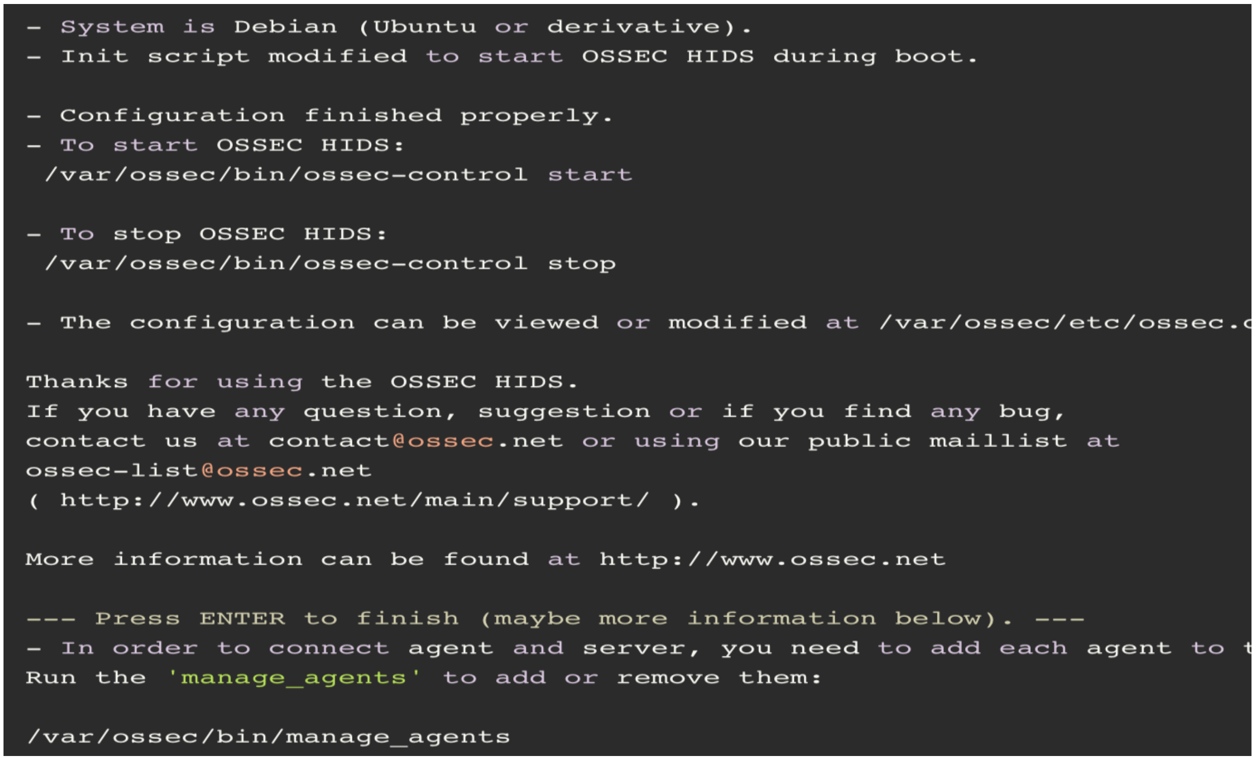
```
3.5- Do you want to enable remote syslog (port 514 udp)? (y/n) [y]:
    - Remote syslog enabled.
```

Press Enter to enable remote Syslog:



```
- If you want to monitor any other file, just change the
ossec.conf and add a new localfile entry.
Any questions about the configuration can be answered
by visiting us online at http://www.ossec.net .

--- Press ENTER to continue ---
```

Finally, Press Enter to start installation. Once the installation succeeds, you should see the following output:



```
- System is Debian (Ubuntu or derivative).
- Init script modified to start OSSEC HIDS during boot.

- Configuration finished properly.
- To start OSSEC HIDS:
 /var/ossec/bin/ossec-control start

- To stop OSSEC HIDS:
 /var/ossec/bin/ossec-control stop

- The configuration can be viewed or modified at /var/ossec/etc/ossec.c

Thanks for using the OSSEC HIDS.
If you have any question, suggestion or if you find any bug,
contact us at contact@ossec.net or using our public maillist at
ossec-list@ossec.net
( http://www.ossec.net/main/support/ ).

More information can be found at http://www.ossec.net

--- Press ENTER to finish (maybe more information below). ---
- In order to connect agent and server, you need to add each agent to t
Run the 'manage_agents' to add or remove them:

/var/ossec/bin/manage_agents
```

Once the installation is completed, start OSSEC with the following command:

```
/var/ossec/bin/ossec-control start
```

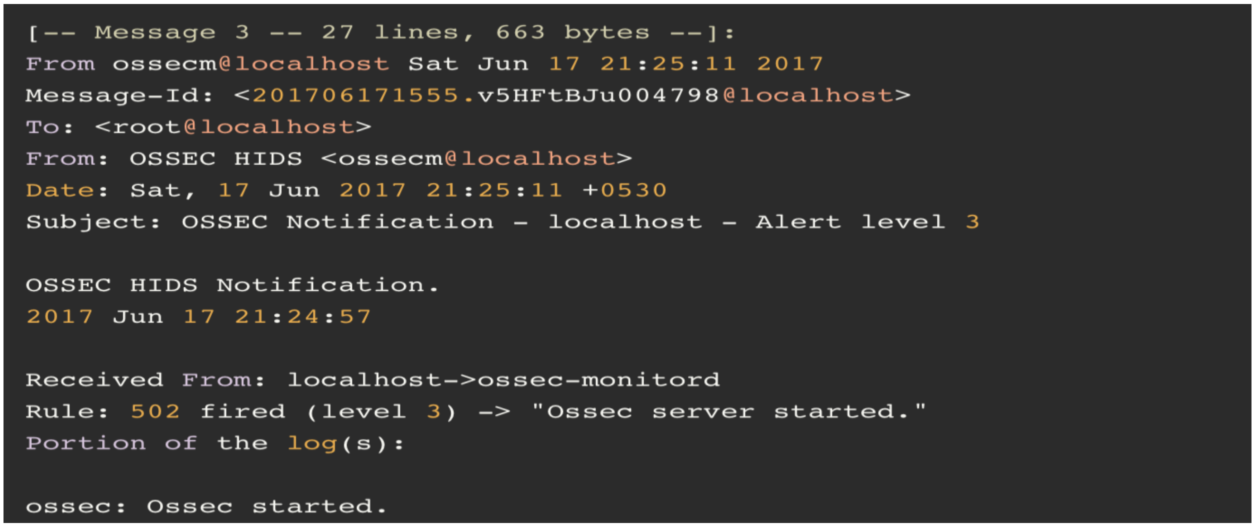
You should see the following output:

```
Starting OSSEC HIDS v2.9 (by Trend Micro Inc.)...
Started ossec-maild...
Started ossec-execd...
Started ossec-analysisd...
Started ossec-logcollector...
Started ossec-syscheckd...
Started ossec-monitord...
Completed.
```

After starting OSSEC, you should also get an e-mail alert. You can check this with the following command:

```
mail
```

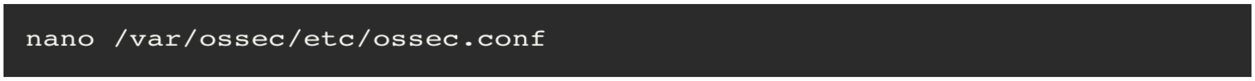
You should see the e-mail looks like the following:



```
[-- Message 3 -- 27 lines, 663 bytes --]:
From ossecm@localhost Sat Jun 17 21:25:11 2017
Message-Id: <201706171555.v5HFtBJu004798@localhost>
To: <root@localhost>
From: OSSEC HIDS <ossecm@localhost>
Date: Sat, 17 Jun 2017 21:25:11 +0530
Subject: OSSEC Notification - localhost - Alert level 3

OSSEC HIDS Notification.
2017 Jun 17 21:24:57

Received From: localhost->ossec-monitord
Rule: 502 fired (level 3) -> "Ossec server started."
Portion of the log(s):

ossec: Ossec started.
```

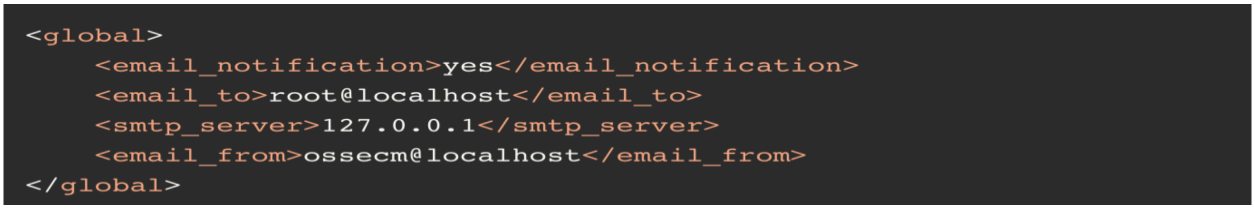
Configure OSSEC

The default configuration of OSSEC works fine. The OSSEC mail configuration file is located inside /var/ossec/etc/ directory.
Now, open the OSSEC main configuration file ossec.conf using the following command:



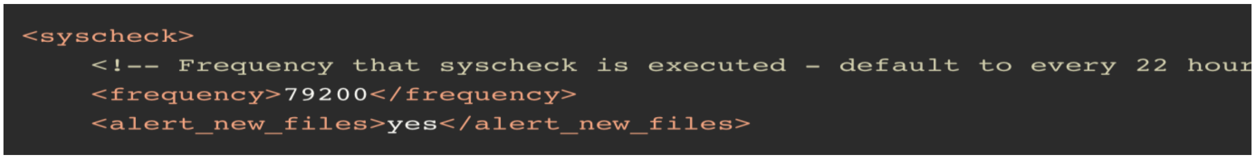
```
nano /var/ossec/etc/ossec.conf
```

The first configuration options is the E-mail configurations which you specified during installation. You can change this setting at any time:



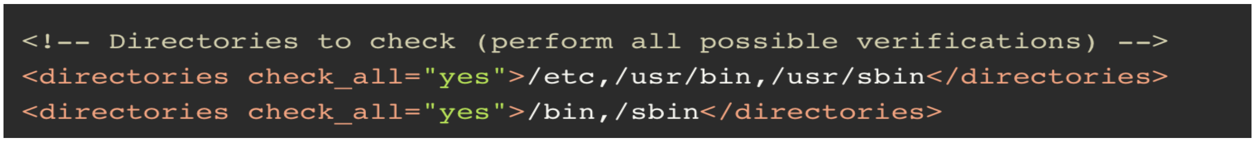
```
<global>
    <email_notification>yes</email_notification>
    <email_to>root@localhost</email_to>
    <smtp_server>127.0.0.1</smtp_server>
    <email_from>ossecm@localhost</email_from>
</global>
```

By default, OSSEC does not alert when a new file is added to the server. You can change that by adding a new line just under the section as shown below:



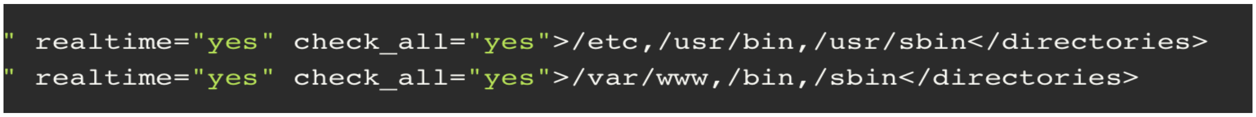
```
<syscheck>
    <!-- Frequency that syscheck is executed - default to every 22 hour
    <frequency>79200</frequency>
    <alert_new_files>yes</alert_new_files>
```

OSSEC does not send real-time alerts by default. You will also need to change is in the list to directories that OSSEC should check. By default, the directories are shown below:



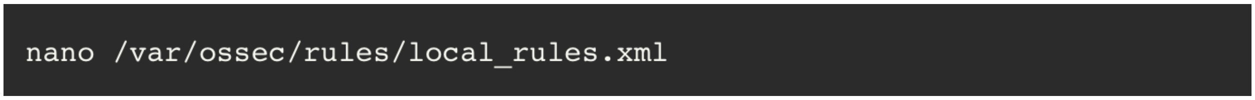
```
<!-- Directories to check (perform all possible verifications) -->
<directories check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
<directories check_all="yes">/bin,/sbin</directories>
```

You will need to modify the above two lines to make OSSEC report changes in real-time. Replace both lines with the following:



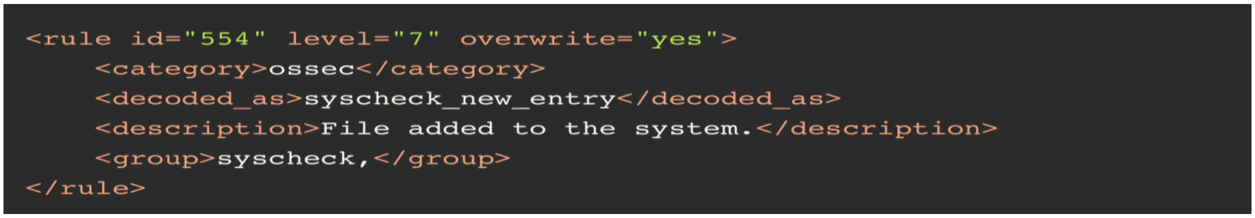
```
" realtime="yes" check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
" realtime="yes" check_all="yes">/var/www,/bin,/sbin</directories>
```

Next, you will need to modify the rules file local_rules.xml located inside /var/ossec/rules directory. This file contains rules for new file added to the system.



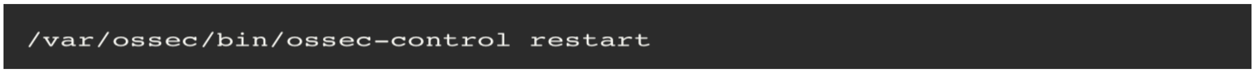
```
nano /var/ossec/rules/local_rules.xml
```

Add the following lines between ... sections:



```
<rule id="554" level="7" overwrite="yes">
    <category>ossec</category>
    <decoded_as>syscheck_new_entry</decoded_as>
    <description>File added to the system.</description>
    <group>syscheck,</group>
</rule>
```

Save and close the file when you are finished. Then restart OSSEC with the following command:



```
/var/ossec/bin/ossec-control restart
```

If all is well OSSEC restarts with no errors.

DDOS Attack:

DOS – Denial of Service

The main idea of a DoS attack is to make a certain service unavailable.

Since every service is in reality, running on a machine, the service can be made unavailable if the performance on the machine can be brought down.

This is the main fundamental behind every DOS and DDOS.

Some examples are:

- Hijacking a server.

- Port Overloading.
- De-authenticate wireless.
- Denying internet-based services.

Types of DOS attacks:

- Ping of Death
- Reflected Attack
- Mailbomb
- Teardrop Attack

DDOS – Distributed Denial of Service

Sending multiple requests from a web-resource or the machine.

Saturates the server capability of managing requests.

Attack is mostly carried out using a botnet of multiple devices.

How does DDOS attack work?

- A hacker must create a network of zombie bots, that can be used to attack the targeted victim when called upon, using malware infusion.
- These bots then flood the target with the continuous requests that cause the server system to crash.

Types of DDOS attacks:

- Volumetric / Network based attack.
- Protocol based attack.
- Application based attack.

Its main aim is:

- Competitive advantage against rival business.
- Ransom demands for releasing data.
- Activist behavior for protests and upstaging.

Prevention:

- Employ load balancers and firewalls.
- Detect an attack early and mitigate the damage beyond that point.
- Switch to cloud service providers like AWS and Azure.
- Allocate more bandwidth to prevent the clogging of data.
- Using Content Delivery Networks (CDNs) that have redundant servers.

SAKETH RAO VARDHINENI:

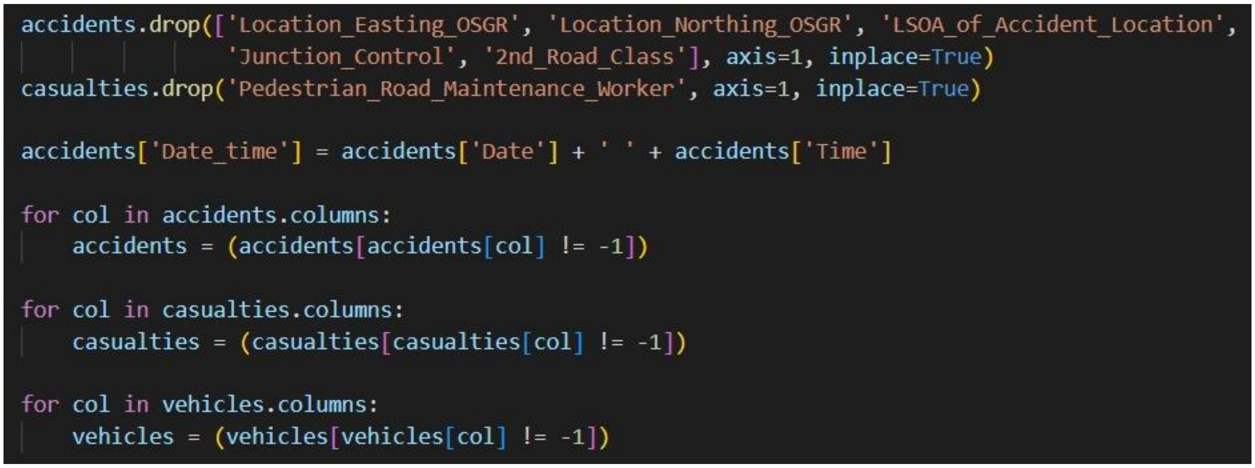
One of the back-end developers who worked on the total backend development. And preprocessing and implementing the randomforestclassifier.

- pre-processed the data:

  We have read the accidents, casualties, and vehicles by pd.read_csv and dropped the unwanted columns.

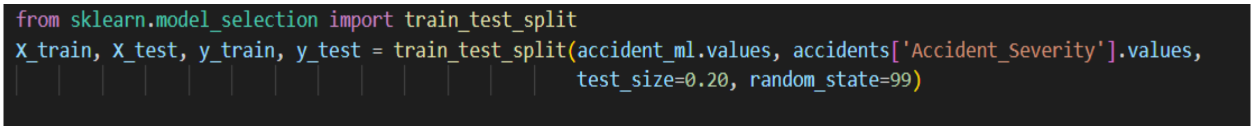
  Joining the date and time column in the accidents dataset by using pandas.

  Now we are removing the rows in the dataset which contains NULL values.



```
accidents.drop(['Location_Easting_OSGR', 'Location_Northing_OSGR', 'LSOA_of_Accident_Location',
                'Junction_Control', '2nd_Road_Class'], axis=1, inplace=True)
casualties.drop('Pedestrian_Road_Maintenance_Worker', axis=1, inplace=True)

accidents['Date_time'] = accidents['Date'] + ' ' + accidents['Time']

for col in accidents.columns:
    accidents = (accidents[accidents[col] != -1])

for col in casualties.columns:
    casualties = (casualties[casualties[col] != -1])

for col in vehicles.columns:
    vehicles = (vehicles[vehicles[col] != -1])
```

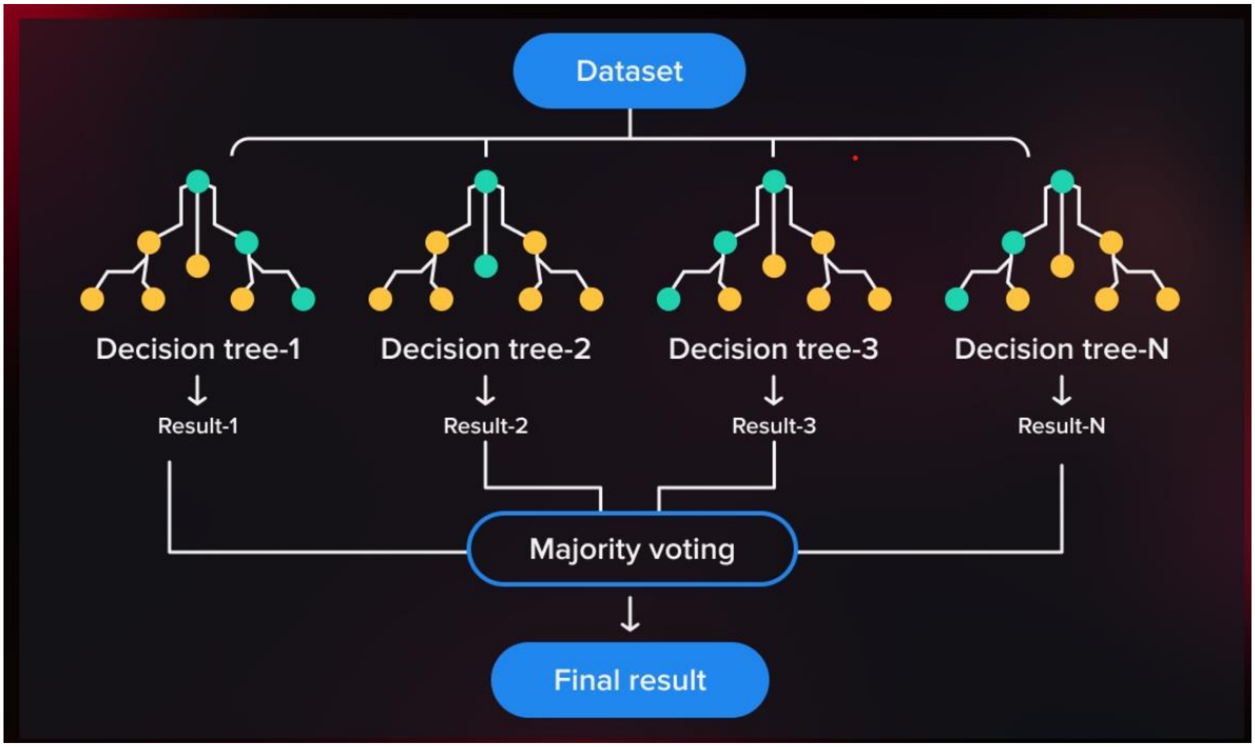
- Plotted a graph for accidents on the days of weeks.
- Plotted a graph for accidents of different age groups.
- Plotted a graph for accidents during the hours of the day.
- Plotted a graph for accident percentage in speed zone.
- We are creating the training data from accidents dataset which contains the columns Age_of_Driver, Vehicle_Type, Engine_Capacity_(CC), Day_of_Week, Weather_Conditions, Road_Surface_Conditions, Age_of_Vehicle, Light_Conditions, Sex_of_Driver, Speed_limit.
- Splitting the preprocessed data into 20 percent of test data and 80 percent of training data by using train_test_split.



```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(accident_ml.values, accidents['Accident_Severity'].values,
                                                    test_size=0.20, random_state=99)
```

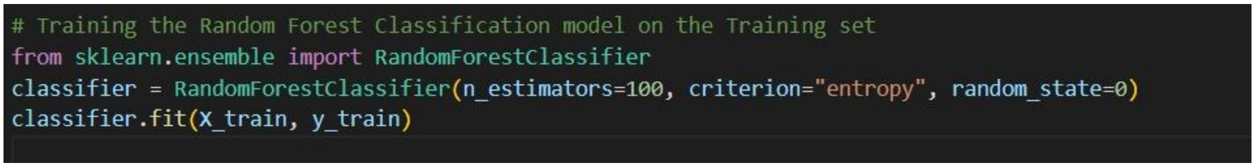
- We are using the randomforestclassifier to train the model, so basically Random forest classifier is a bagging technique and a Supervised Machine Learning Algorithm that is used widely in **Classification and Regression problems**.
- random forest classifier is made up of numerous independent decision trees that work together as an ensemble. Each individual tree in the random forest spits out a class prediction, and the class with the most votes becomes our model's prediction.

- The data set contains a sample of rows and sample of features, initially we pick of some samples of rows called row sampling with replacement along with some columns called as feature sampling replacement sent to decision trees. So, the data sent to the decision tree i.e is row sampling with replacement along with some columns called as feature sampling replacement should be less than the total data set.
- Then the decision trees are trained on the data and when we send the test data the decision trees will be able to give the prediction. The predicted values are sent into the voting classifier where the prediction generated by most of the decision trees is the final prediction.



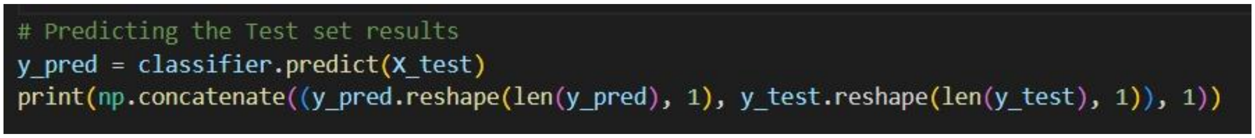
Where the decision tree has low bias and high variance which leads to overfitting of the data so to overcome this, we are using multiple decision trees.

- We have trained the random forest classifier model with the training data.



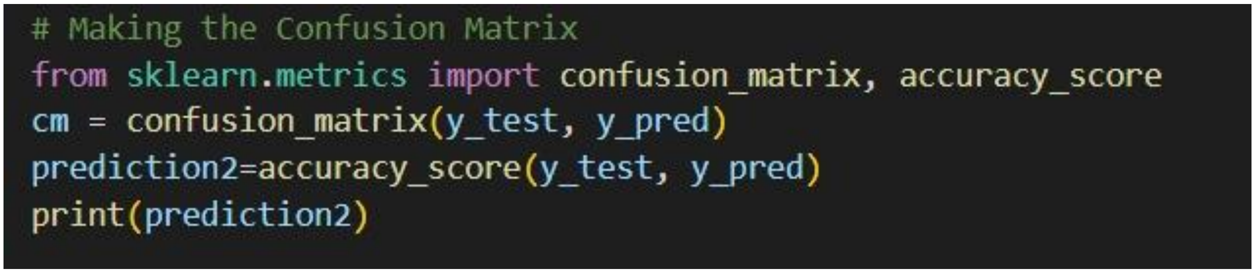
```
# Training the Random Forest Classification model on the Training set
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=100, criterion="entropy", random_state=0)
classifier.fit(X_train, y_train)
```

- We predicted the data by giving the test data as input to the model.



```
# Predicting the Test set results
y_pred = classifier.predict(X_test)
print(np.concatenate((y_pred.reshape(len(y_pred), 1), y_test.reshape(len(y_test), 1)), 1))
```

- Then we evaluated our model by finding accuracy. The accuracy of our model is 76%.



```
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, y_pred)
prediction2=accuracy_score(y_test, y_pred)
print(prediction2)
```

SQL Injection:

It is a code injection technique used to execute malicious SQL statements.

Types –

- Time – based: It relies on sending an SQL query to the database which forces the database to wait for a specified amount of time before responding.
- Out – of – bound: It is not very common, because it depends on features being enabled on the DBMS server being used by the web application.
  This type is used when an attacker is unable to use the same channel to attack and gather the result.

Prevention:

- Use prepared statements and parameterized queries – It makes sure that the parameters passed into SQL statements are treated in a safe manner.
- Object relational mapping – Most of the development teams prefer to use this framework to make the translation of SQL result sets into code objects more seamless.
- Escaping inputs – A simple way to protect. Many languages have standard functions to achieve this.
- Password hashing
- Third party authentication
- Web application firewall
- Buy better software
- Always update and use patches
- Continuously monitor SQL statements and database

Index.php page will have the sign in details like username and password, when we enter correct login details it will redirect to homepage, or else it will notify us that we have entered wrong login details and Index.PHP will also provide sign up option where we are going to create a new account by inserting our details like First and Last name, UserID and password into the table.

So SQL injection occurs when we enter (‘#) at the end of our userID then the verification of the password will not be done because the #, the user ID’s password from the database table is not verifying with the password which we have entered in the sign-in page, and the page is directed to homepage for any given password, This is one of a type SQL Injection.

When we enter ‘or 1=1’ in addition to the user ID we give, then the user id of the table is assigned to True because 1=1 is always true , so it’ll match our user id which is in our database table so it will always accept any user id we give, same goes with the password entry,(When we enter ‘or 1=1’ in addition to the password we give, then the password of the table is assigned to True because 1=1 is always true , so it’ll match our password which is in our database table so it will always accept any password we give) this is another type of SQLi

When we enter our user id with; DROP TABLE table name, then the table will be dropped from the database and we can’t access the homepage anymore, this is another example of SQLi

SARAN KUMAR KAMANDULA: AWS setup , Installing PHP

Get started with Amazon EC2 Linux/Ubuntu instances.

Step 1: Launch an instance

You can launch a Linux instance using the AWS Management Console as described in the following procedure.

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.

2. From the EC2 console dashboard, in the Launch instance box, choose Launch instance, and then choose Launch instance from the options that appear.

3. Under Name and tags, for Name, enter a descriptive name for your instance.

4. Under Application and OS Images (Amazon Machine Image), do the following:

- Choose Quick Start, and then choose Amazon Linux. This is the operating system (OS) for your instance.

- From Amazon Machine Image (AMI), select an HVM version of Amazon Linux 2. Notice that these AMIs are marked Free tier eligible. An Amazon Machine Image (AMI) is a basic configuration that serves as a template for your instance.

Under Instance type, from the Instance type list, you can select the hardware configuration for your instance. Choose the t2.micro instance type, which is selected by default. The t2.micro instance type is eligible for the free tier. In Regions where t2.micro is unavailable, you can use a t3.micro instance under the free tier. For more information, see AWS Free Tier.

5. Under Instance type, from the Instance type list, you can select the hardware configuration for your instance. Choose the t2.micro instance type, which is selected by default. The t2.micro instance type is eligible for the free tier. In Regions where t2.micro is unavailable, you can use a t3.micro instance under the free tier. For more information, see AWS Free Tier.

6. Under Key pair (login), for Key pair name, choose the key pair that you created when getting set up.

7. Next to Network settings, choose Edit. For Security group name, you'll see that the wizard created and selected a security group for you. You can use this security group, or alternatively you can select the security group that you created when getting set up using the following steps:

- Choose Select existing security group.

- From Common security groups, choose your security group from the list of existing security groups

8. Keep the default selections for the other configuration settings for your instance.

9. Review a summary of your instance configuration in the Summary panel, and when you're ready, choose Launch instance.

10. A confirmation page lets you know that your instance is launching. Choose View all instances to close the confirmation page and return to the console.

11. On the Instances screen, you can view the status of the launch. It takes a short time for an instance to launch. When you launch an instance, its initial state is pending. After the instance starts, its state changes to running and it receives a public DNS name. If the Public IPv4 DNS column is hidden, choose the settings icon ( ) in the top-right corner, toggle on Public IPv4 DNS, and choose Confirm.

12. It can take a few minutes for the instance to be ready for you to connect to it. Check that your instance has passed its status checks; you can view this information in the Status check column.

Step 2- Connect to your instance.

Step 3- Clean up your instance

After you've finished with the instance that you created for this tutorial, you should clean up by terminating the instance.

To terminate your instance-

1. In the navigation pane, choose Instances. In the list of instances, select the instance.

2. Choose Instance state, Terminate instance.

3. Choose Terminate when prompted for confirmation.

Amazon EC2 shuts down and terminates your instance. After your instance is terminated, it remains visible on the console for a short while, and then the entry is automatically deleted. You cannot remove the terminated instance from the console display yourself.

Installing PHP:

The Apache web server is among the most popular web servers in the world. It’s well documented, has an active community of users, and has been in wide use for much of the history of the web, which makes it a great choice for hosting a website.

Start by updating the package manager cache. If this is the first time you’re using sudo within this session, you’ll be prompted to provide your user’s password to confirm you have the right privileges to manage system packages with apt:

Then Install Apache with

You’ll be prompted to confirm Apache’s installation. Confirm by pressing Y, then ENTER.

Once the installation is finished, you’ll need to adjust your firewall settings to allow HTTP traffic. Ubuntu’s default firewall configuration tool is called Uncomplicated Firewall (UFW). It has different application profiles that you can leverage. To list all currently available UFW application profiles, execute this command:

Apt Setup:

1) grab the package from the web

```
wget http://security.ubuntu.com/ubuntu/pool/main/a/apt/apt_1.0.1ubuntu2.17_amd64.deb -O apt.deb
```

2) install it with dpkg

```
sudo dpkg -i apt.deb
```

if everything goes right, that should be enough.

Probably you may find that some dependencies are not satisfied, to deal with that you may need to create a folder (something like "apt-installer") and drop all the dependencies of the aptpackage there. The list of dependencies required will be shown by the dpkg output and the download process is similar to the one explained above. Last, if you don't feel confident of what you are doing, my suggestion is that you may want to do a fresh install of your system.

VISHWASENA RAIDU NYRAMNENI**:**

One of the two back-end supports that worked on total backend development and integrations with the front end. Contributed on report writing.

PHP

Step 1- Create a HTML PHP Login Form

The code consists of HTML elements. The basic elements are:

<title>

The element displays the heading of the document. The label can be only in the text format, and it shows the text in the browser's tab.

<link>

It creates a link between the working document and an external resource.

<form>

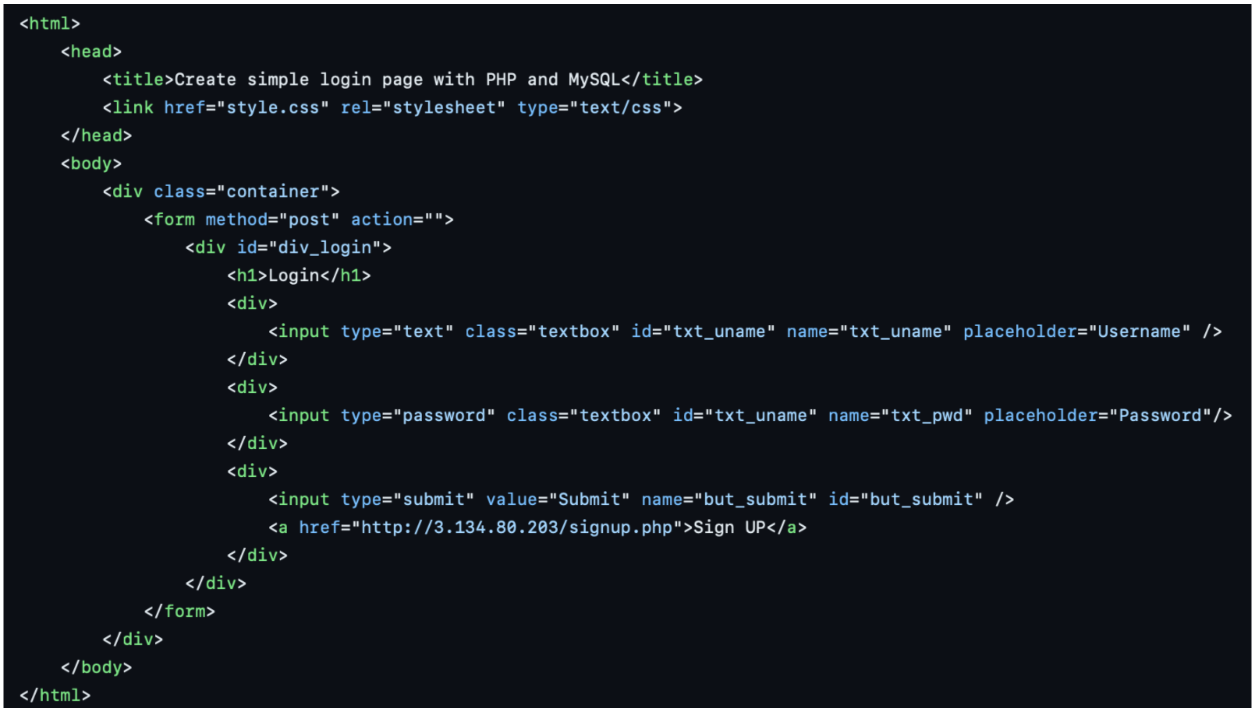
The element defines the user input included while creating an HTML.

<input type>

It displays a one-line input field.

<label>

It defines a label for many form elements. It majorly indicates the radio button/checkbox.



```
<html>
    <head>
        <title>Create simple login page with PHP and MySQL</title>
        <link href="style.css" rel="stylesheet" type="text/css">
    </head>
    <body>
        <div class="container">
            <form method="post" action="">
                <div id="div_login">
                    <h1>Login</h1>
                    <div>
                        <input type="text" class="textbox" id="txt_uname" name="txt_uname" placeholder="Username" />
                    </div>
                    <div>
                        <input type="password" class="textbox" id="txt_uname" name="txt_pwd" placeholder="Password"/>
                    </div>
                    <div>
                        <input type="submit" value="Submit" name="but_submit" id="but_submit" />
                        <a href="http://3.134.80.203/signup.php">Sign UP</a>
                    </div>
                </div>
            </form>
        </div>
    </body>
</html>
```

Step 2: Create a CSS Code for Website Design

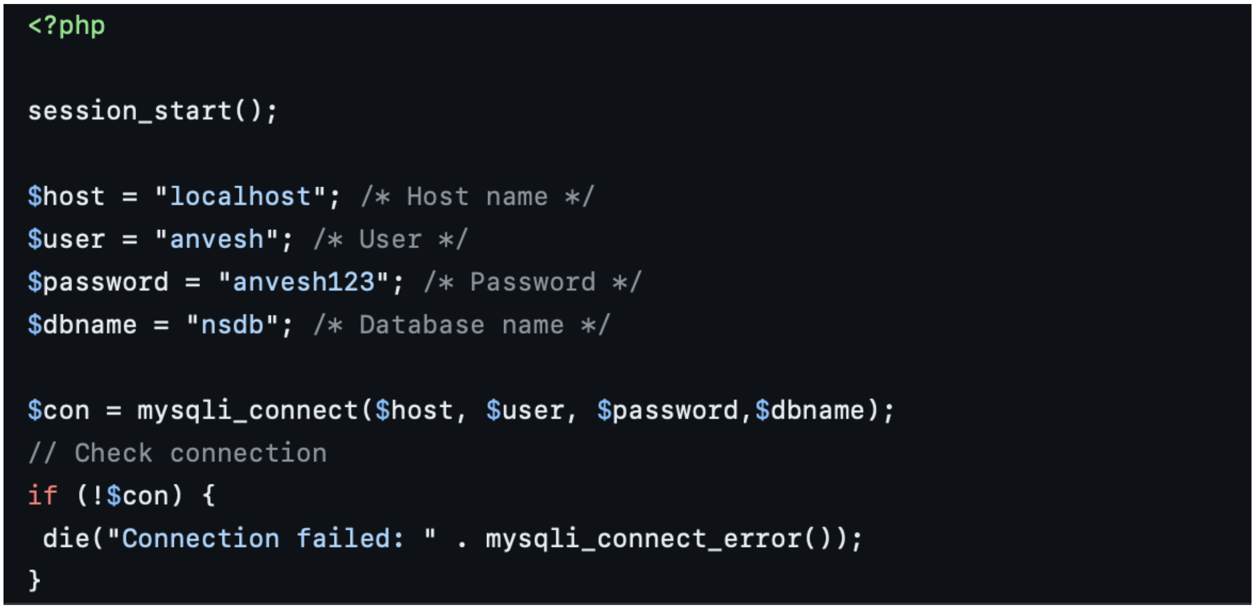
Next comes the CSS sheet. Now, the aim is to create a cascading style sheet (CSS) file to improve the design of the webpage.

STEP 3

In ubuntu server we are creating a MySQL database with the database name as nsdb.

Then, we connect to the nsdb data base by the command use nsdb, then we are creating the table with the table name as user. The query we used to create the table is “Create table user(varchar username, varchar name, varchar password);”

Step 4: we create a “config.php” page. In this page we connect the database to the form by using mysqli_connect( ).



```php
<?php

session_start();

$host = "localhost"; /* Host name */
$user = "anvesh"; /* User */
$password = "anvesh123"; /* Password */
$dbname = "nsdb"; /* Database name */

$con = mysqli_connect($host, $user, $password,$dbname);
// Check connection
if (!$con) {
 die("Connection failed: " . mysqli_connect_error());
}
```

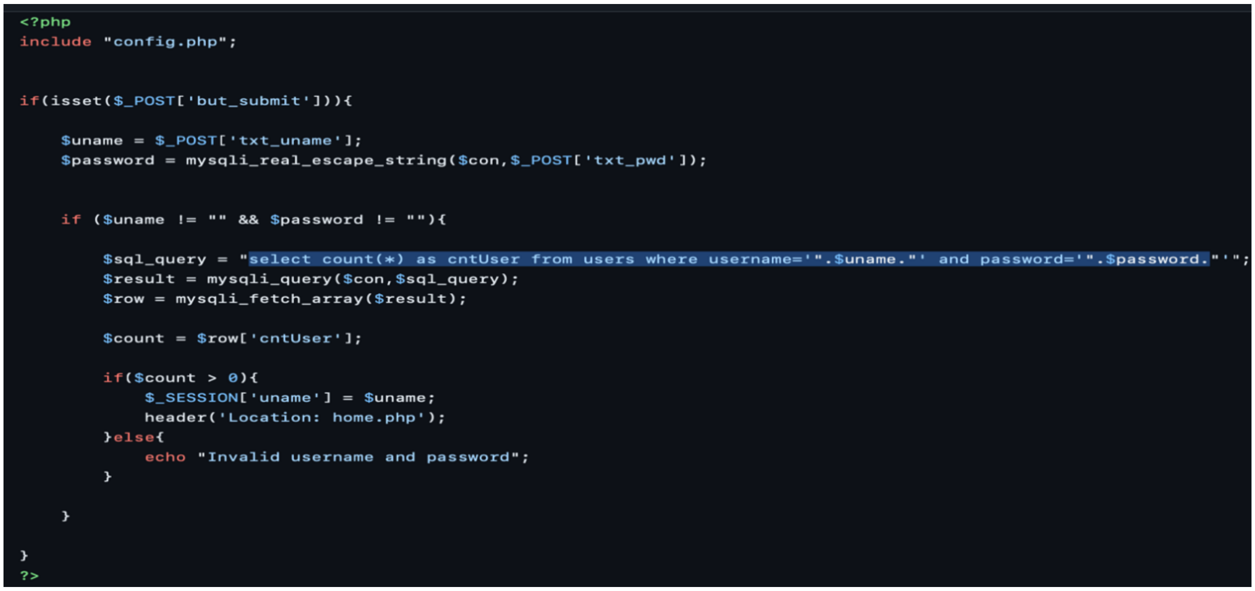
Config.php

Step -4 Creating a index.php page

The html form contains userid and password when we click the submit button an action is performed where the userid and password is stored in the variable and then we perform and SQL query “select count(*) as cntUser from users where username='".$uname."' and password='".$password.”

$uname is the variable which store the text input of userid from html form. $password is the variable which store the text input of userid from html form.

Checking weather the given userid and password matches with the credentials in the database table.



```php
<?php
include "config.php";


if(isset($_POST['but_submit'])){

    $uname = $_POST['txt_uname'];
    $password = mysqli_real_escape_string($con,$_POST['txt_pwd']);


    if ($uname != "" && $password != ""){

        $sql_query = "select count(*) as cntUser from users where username='".$uname."' and password='".$password."'";
        $result = mysqli_query($con,$sql_query);
        $row = mysqli_fetch_array($result);

        $count = $row['cntUser'];

        if($count > 0){
            $_SESSION['uname'] = $uname;
            header('Location: home.php');
        }else{
            echo "Invalid username and password";
        }

    }

}
?>
```

Step –5

Create a signup.php

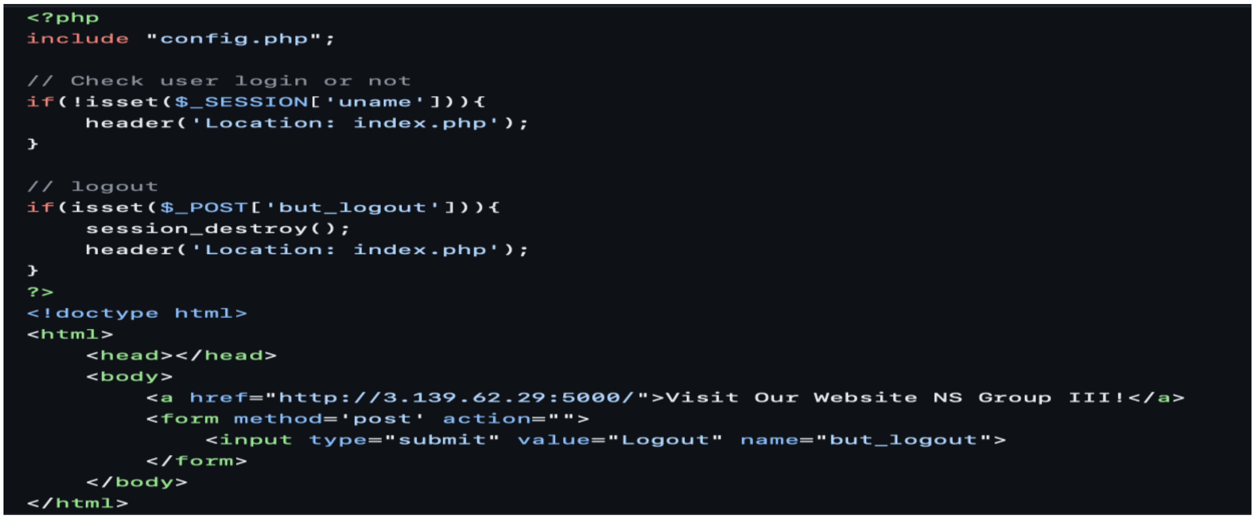
Where we are creating a signup page where we are inserting the data userid , first and lastname into the database table

Step 6- Create a Logout Session

In this section, you will create a "home.php" file.

When a user selects the logout option, the code mentioned below will automatically redirect the user back to the login page.

When we click on the link which is in the homepage, then it will redirect to the Accident Prevention Website



```php
<?php
include "config.php";

// Check user login or not
if(!isset($_SESSION['uname'])){
    header('Location: index.php');
}

// logout
if(isset($_POST['but_logout'])){
    session_destroy();
    header('Location: index.php');
}
?>
<!doctype html>
<html>
    <head></head>
    <body>
        <a href="http://3.139.62.29:5000/">Visit Our Website NS Group III!</a>
        <form method='post' action="">
            <input type="submit" value="Logout" name="but_logout">
        </form>
    </body>
</html>
```

NIKHILESH REDDY PEKETI: Frontend, SQL mitigation, software setup of python and gitbash

Why Python???

- Solves complex problems in less time with fewer lines of code.
- High-level
- Huge Community
- Cross-Platform
- Large Ecosystem
- A multi-purpose language with a simple, and beginner friendly syntax.
- Extremely rich libraries.
- Extensive online documentation
- Diverse applications

Steps to install:

1. Download the installer from the link: https://www.python.org/downloads/
2. Run the executable installer i.e., .exe file.
3. Once it is installed, check the version using the "python -version" command either in the command prompt or in the Git Bash.
4. Check the path in the system variables under environment variables for any issues in the installation.

Installing Python in Ubuntu

The steps are.

1. Open the terminal by pressing Ctrl + Alt +T.

2. Update your local system's repository.

Command- sudo apt update

3. Download the latest version of Python.

Command- sudo apt install python3

4. APT will automatically find the package and install it on your computer.

Git:

It is a free and open-source version control system.

Version Control – The management of changes to documents, computer programs, large web sites, and other collections of information.

Few terms –

- Directory – Folder
- Terminal or command line – Interface for text commands
- CLI – Command Line Interface
- cd – Change directory.
- Code Editor – Word processor for writing code

- Repository – Project
- GitHub – Website to host our repositories online.

Few commands –

- clone – Bring a repository that is hosted in GitHub into a folder on our local machine.
- add - track the files and changes in git.
- commit – saves files in git.
- push – upload git commits to a remote repo like GitHub.
- pull – downloads changes from remote repo to the local machine, it's just the opposite of push.

Why Git???

- Free
- Open source
- Super-fast
- Scalable
- Cheap branching/ merging

Steps to install –

1. Navigate to the latest https://gitforwindows.org/ and download the latest version.
2. Once the .exe file is downloaded, follow the installation instructions as provided in the Git Setup wizard until the installation is complete.
3. Open the command prompt or Git Bash.

Check the version of the git using "git version" command (if it displays the version in the output then the git is installed successfully)

Installation of Git Bash

1. Download the Git Bash setup from the official setup- https://git-scm.com/

2. Download the Installer.

3. Run the .exe file you just downloaded and follow the instructions in the installer.

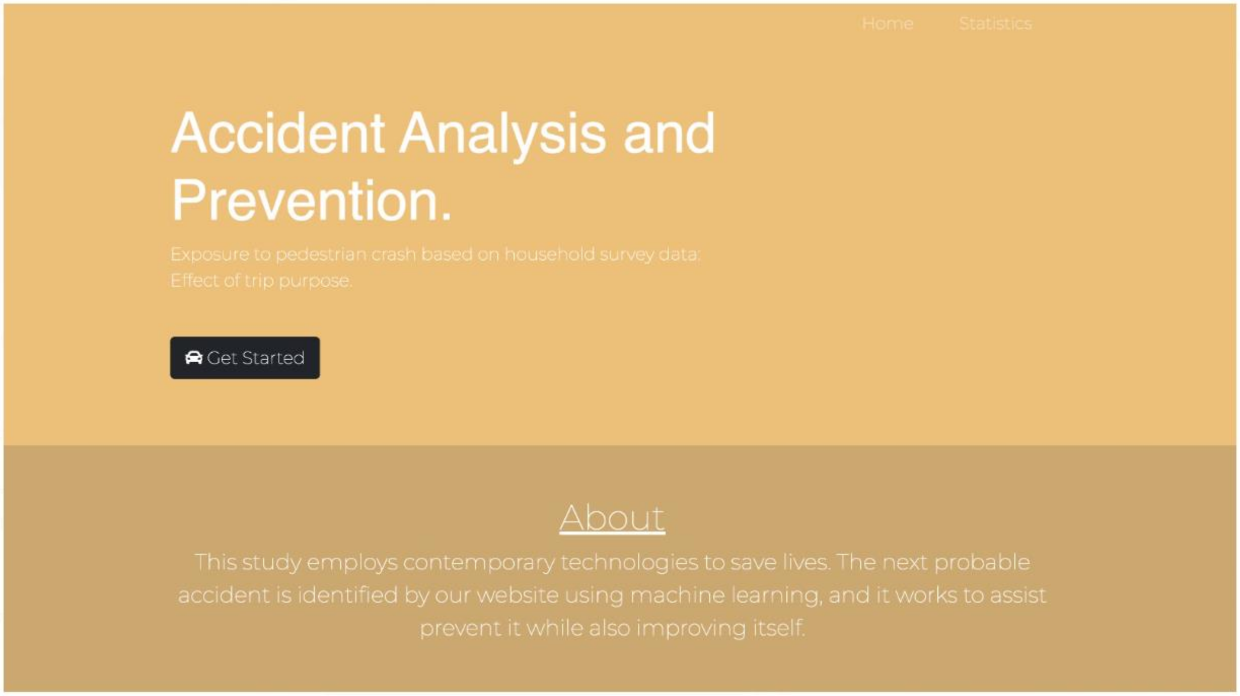
4. Git Bash will be installed in the system. By right clicking you can run as an administrator.

Usage of Git

1. To implement the DDOS attacks.

2. To update in the GitHub.

Example- git clone, git commit, git checkout, git push, and more.



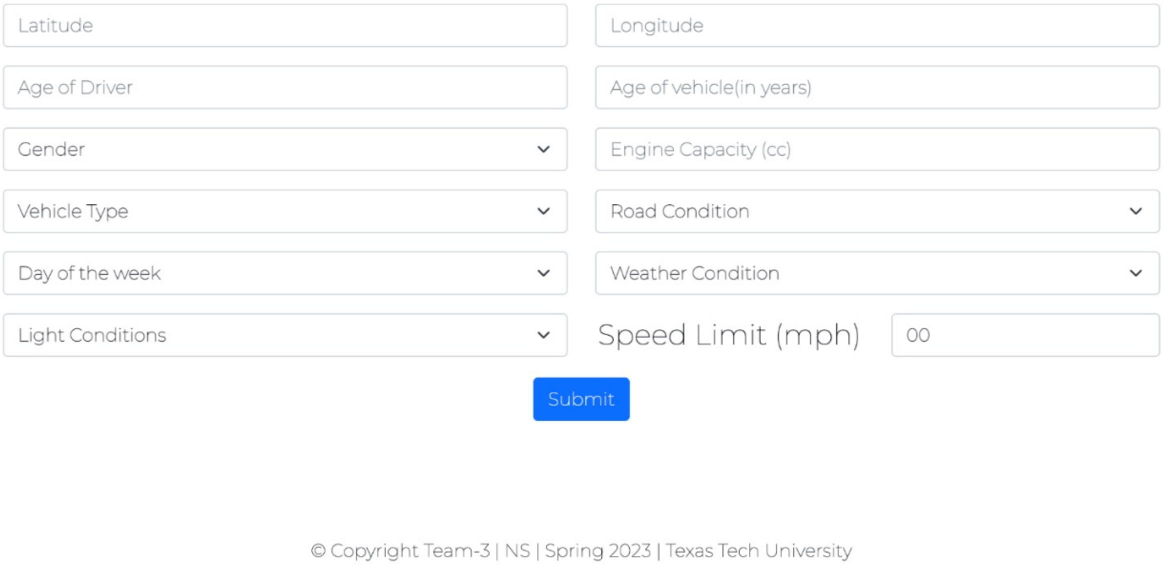
Accident Analysis and Prevention.

Exposure to pedestrian crash based on household survey data: Effect of trip purpose.

About

This study employs contemporary technologies to save lives. The next probable accident is identified by our website using machine learning, and it works to assist prevent it while also improving itself.

HOME PAGE



© Copyright Team-3 | NS | Spring 2023 | Texas Tech University

FAROOQ ABDULLA SHAIK: GIT, VS code

VS code Setup:

Why VS Code –

- Edit, Build, and Debug with ease.
- Make it our own.
- Built with love for the web.
- Robust and extensible architecture.
- Ready, Set, Code!

Steps to install:

1. Download the VS code from the link: https://code.visualstudio.com/download
2. Run the installer i.e., .exe file.
3. Follow the installation instructions and complete the installation process.
4. By default, it will be installed under this path: "**C:\username\AppData\Local\Programs\Microsoft VS Code**"
5. Check the path in the system variables under environment variables for any issues in the installation.

Yum Repo Setup

Step 1- Upgrading the system

We have executed the upgrade command for getting the latest package information and updating package repositories:

1.
   ```
   $ sudo apt upgrade
   ```

Step 2- Install YUM

1.
   ```
   $ sudo apt-get install yum
   ```

Installing MySQL

Now that you have a web server up and running, you need to install the database system to be able to store and manage data for your site. MySQL is a popular database management system used within PHP environments.

Again, use apt to acquire and install this software:

```
$ sudo apt install mysql-server
```

When prompted, confirm installation by typing Y, and then ENTER.

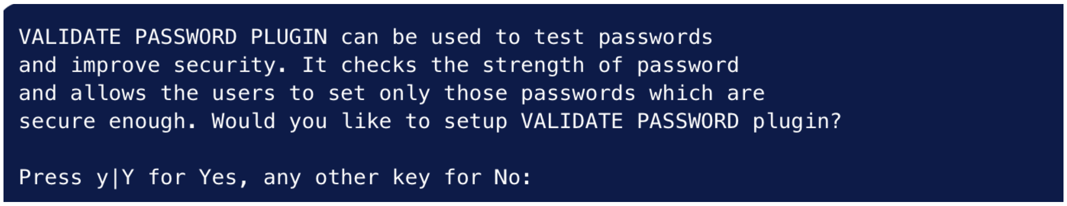
When the installation is finished, it's recommended that you run a security script that comes pre-installed with MySQL. This script will remove some insecure default settings and lock down access to your database system.

Start the interactive script by running:

```
$ sudo mysql_secure_installation
```

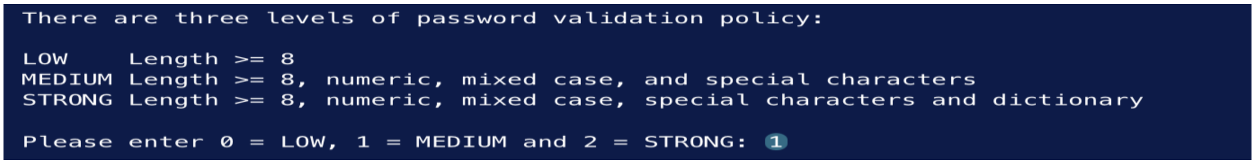
This will ask if you want to configure the VALIDATE PASSWORD PLUGIN.

Answer Y for yes, or anything else to continue without enabling.



```
VALIDATE PASSWORD PLUGIN can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD plugin?

Press y|Y for Yes, any other key for No:
```

If you answer “yes”, you’ll be asked to select a level of password validation. Keep in mind that if you enter 2 for the strongest level, you will receive errors when attempting to set any password which does not contain numbers, upper and lowercase letters, and special characters:



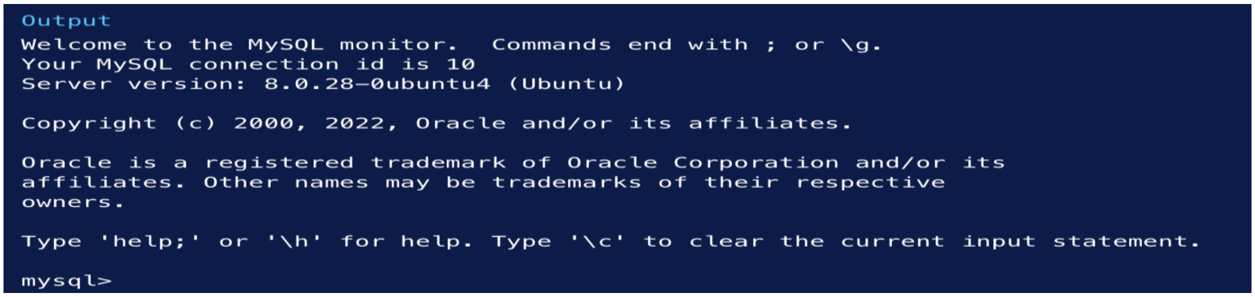
```
There are three levels of password validation policy:

LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 1
```

If you enabled password validation, you’ll be shown the password strength for the root password you just entered and your server will ask if you want to continue with that password. If you are happy with your current password, enter Y for “yes” at the prompt:

When you’re finished, test whether you’re able to log in to the MySQL console by typing:

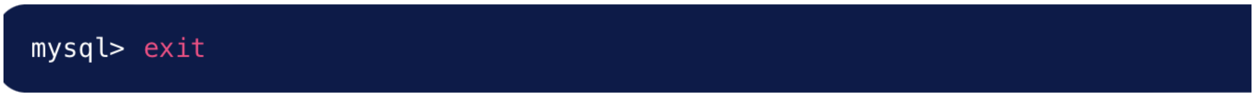
```
$ sudo mysql
```

This will connect to the MySQL server as the administrative database user **root**, which is inferred by the use of sudo when running this command. Below is an example output:



```
Output
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.28-0ubuntu4 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

To exit the MySQL console, type:



```
mysql> exit
```

Httpd:

Step 1 — Installing Apache

#Apache is available within CentOS’s default software repositories, which means you can install it with the yum package manager.

#As the non-root sudo user configured in the prerequisites, update the local Apache httpd package index to reflect the latest upstream changes:

```
$sudo yum update httpd
```

#Once the packages are updated, install the Apache package:

```
$sudo yum install httpd
```

#After confirming the installation, yum will install Apache and all required dependencies

#If you completed the Additional Recommended Steps for New CentOS 7 Servers guide mentioned in the prerequisites section, you will have installed firewalld on your server and you’ll need to open up port 80 to allow Apache to serve requests over HTTP. If you haven’t already done so, you can do this by enabling firewalld’s http service with the following command:

```
$sudo firewall-cmd --permanent --add-service=http
```

#If you plan to configure Apache to serve content over HTTPS, you will also want to open up port 443 by enabling the https service:

```
$sudo firewall-cmd --permanent --add-service=https
```

#Next, reload the firewall to put these new rules into effect:

```
$sudo firewall-cmd --reload
```

##Step 2 — Checking your Web Server

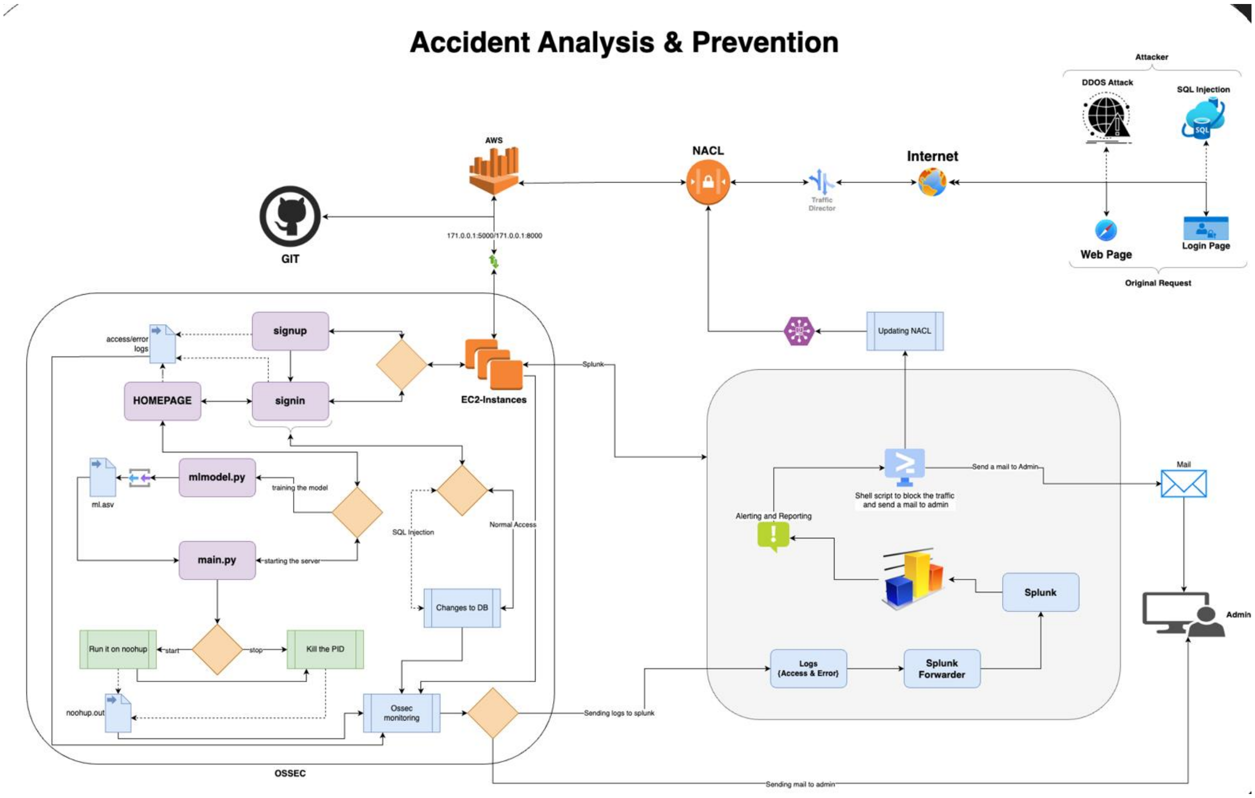
#Apache does not automatically start on CentOS once the installation completes. You will need to start the Apache process manually:

```
$sudo systemctl start httpd
```

#Verify that the service is running with the following command:

```
$sudo systemctl status httpd
```

Activity Diagram:



Flow:

1. Sign up to the page in this https://127.0.0.1:5000 URL.
2. Then, sign in.
3. Access the web page, here the request will go to NACL from the web page.
4. From NACL to Security Groups (AWS)
5. It runs with the main.py file which is present in the launched EC2 Instance.
6. Parallelly, logs are generated when the request is running on the main.py.
7. These logs are stored in the noohup.out file.
8. Using the Splunk Universal Forwarder, we move the logs which are present in the noohup.out file to the Splunk.
9. Based on the traffic generated, the Splunk feature i.e., "Reporting and Alerting" it will trigger the shell script which is present in the EC2 instance (based on high or low traffic w.r.t our desired requirement)
10. When the shell script is triggered (as mentioned in the above step), it also sends an email to the administrator about the status of the traffic.
11. Parallelly, it also blocks the IP address (attacker) which is generating the high traffic in the NACL.
12. Till this step, the DDOS attack is triggered and mitigated.
13. When an attacker accesses the web login page (https://172.0.0.1/index.php)
14. He injects the SQL injection with the help of the login and sign-up page.
15. Once he injects successfully, then the database gets altered completely.
16. This means, we can login to the web page with all the credentials (including fake credentials)
17. All these processes are monitored by OSSEC, if there's any unusual action it performs according to it.
18. Firstly, it sends an email to the administrator.

19. Secondly, it generates logs for the unusual activity and categorizes them according to the OSSEC rules (i.e., from level 00 to level 15)
20. Then the administrator changes the DB manually and solves the issue.
21. So, here we are mitigating the SQL injection attack manually.