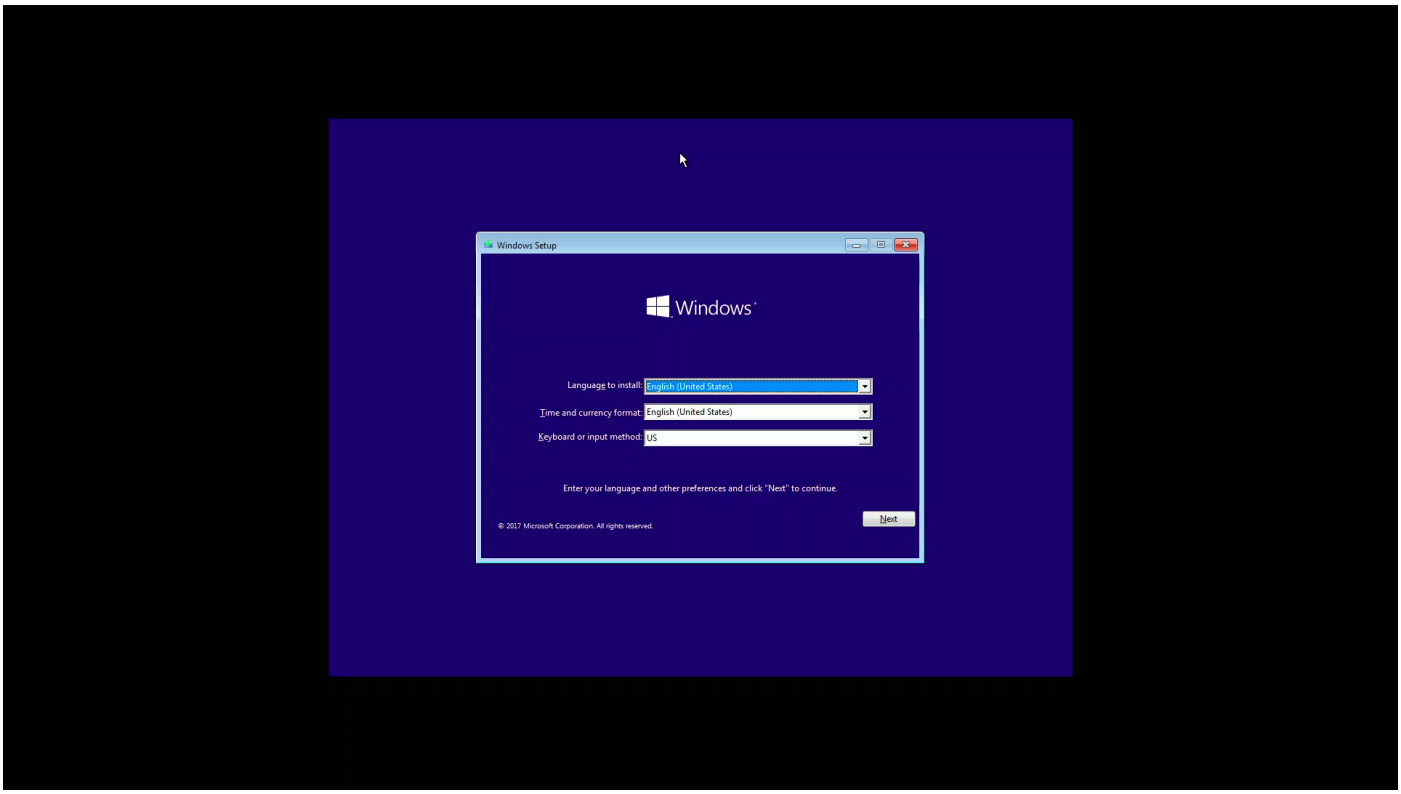
Practical – 1

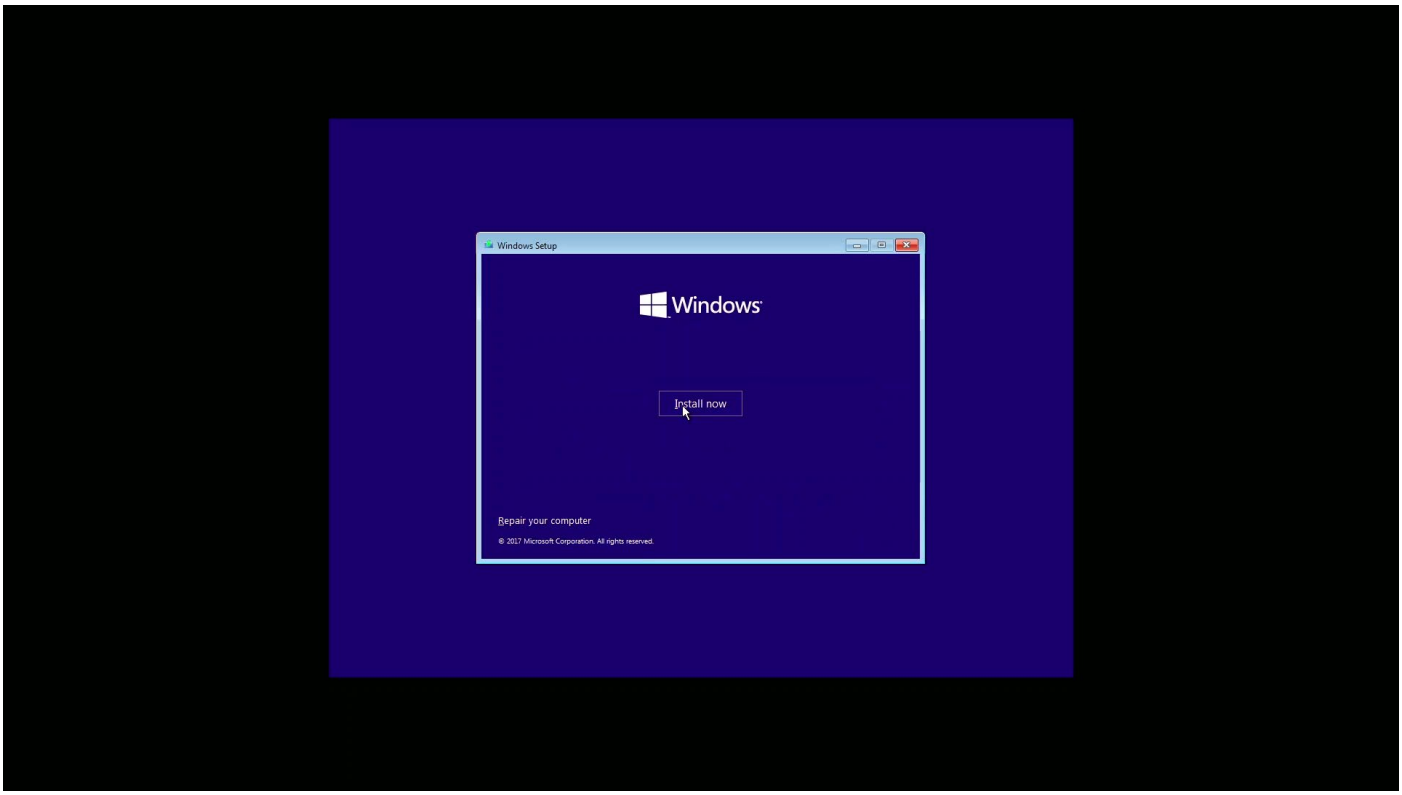
Aim: Install Multiple operating system in virtual machine.

Windows 10:

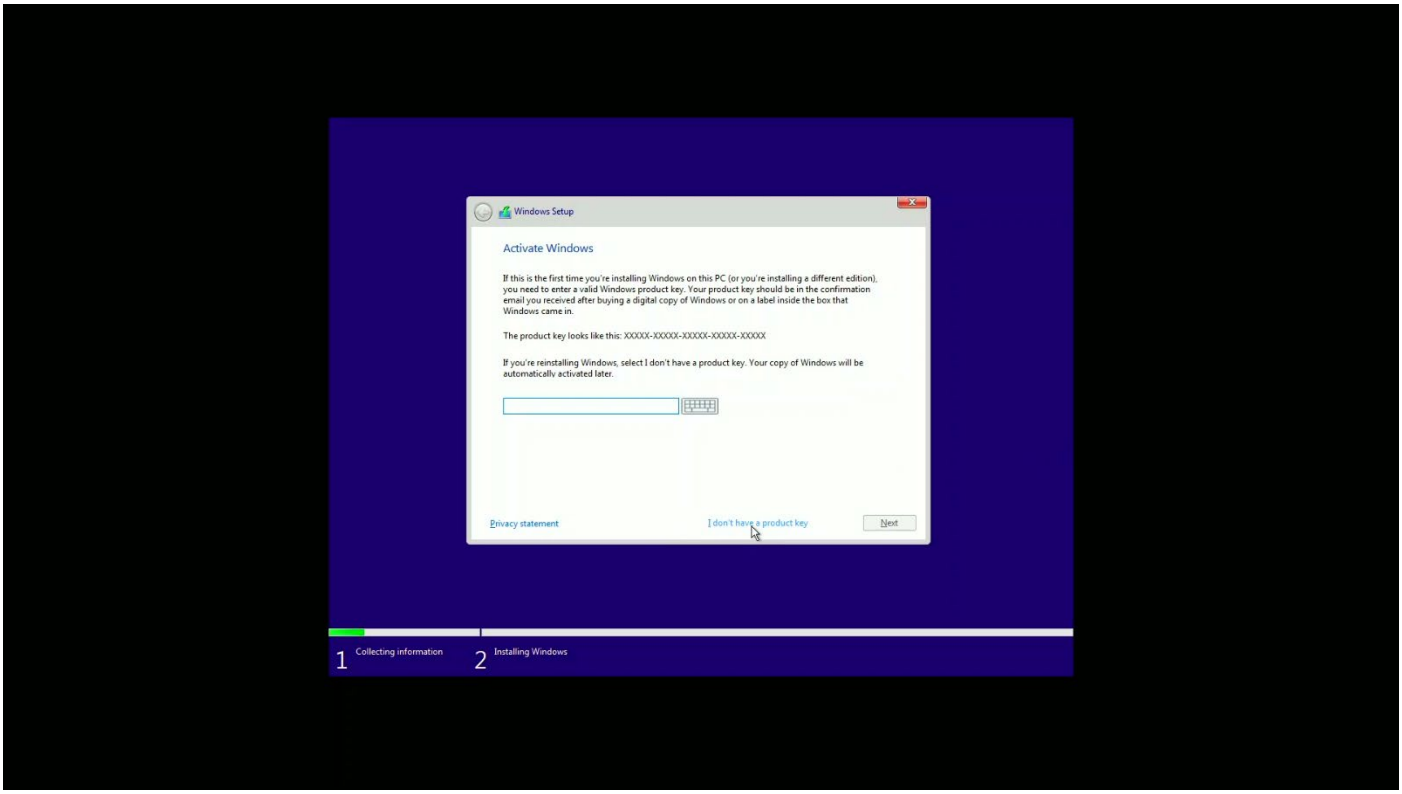
Step 1: Select Language, Time Zone and Keyboard Layout and click “Next”.



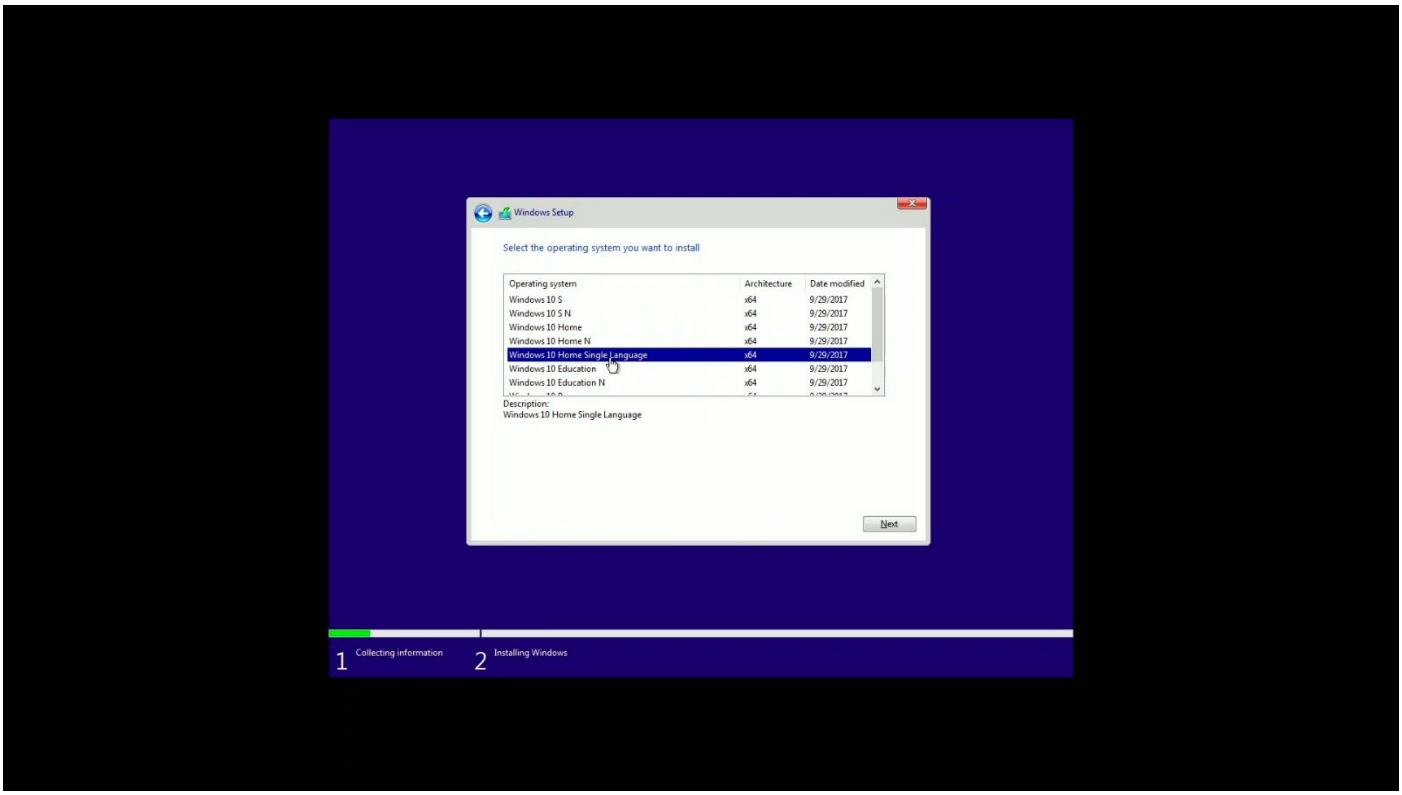
Step 2: Click “Install Now”



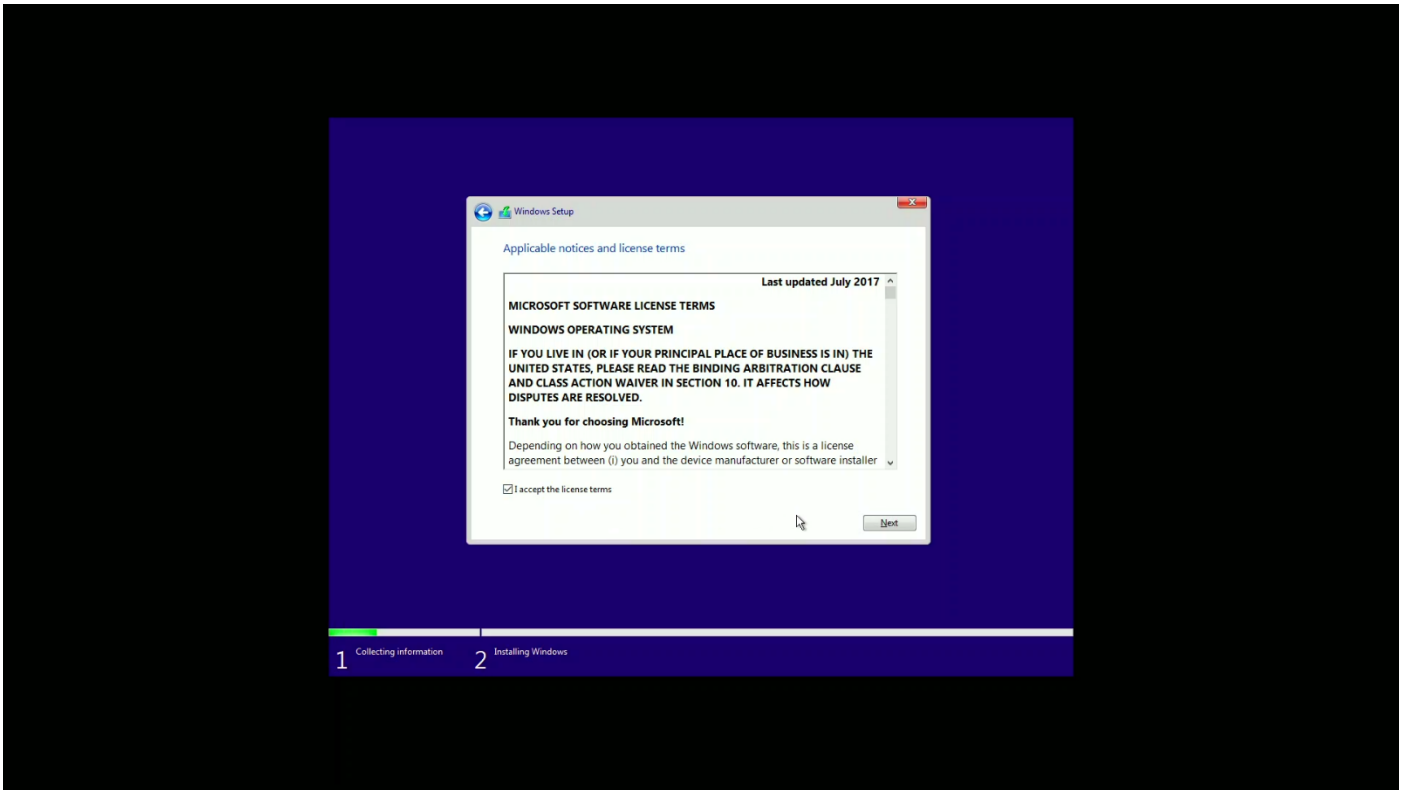
Step 3: Click “I don’t have product key”.



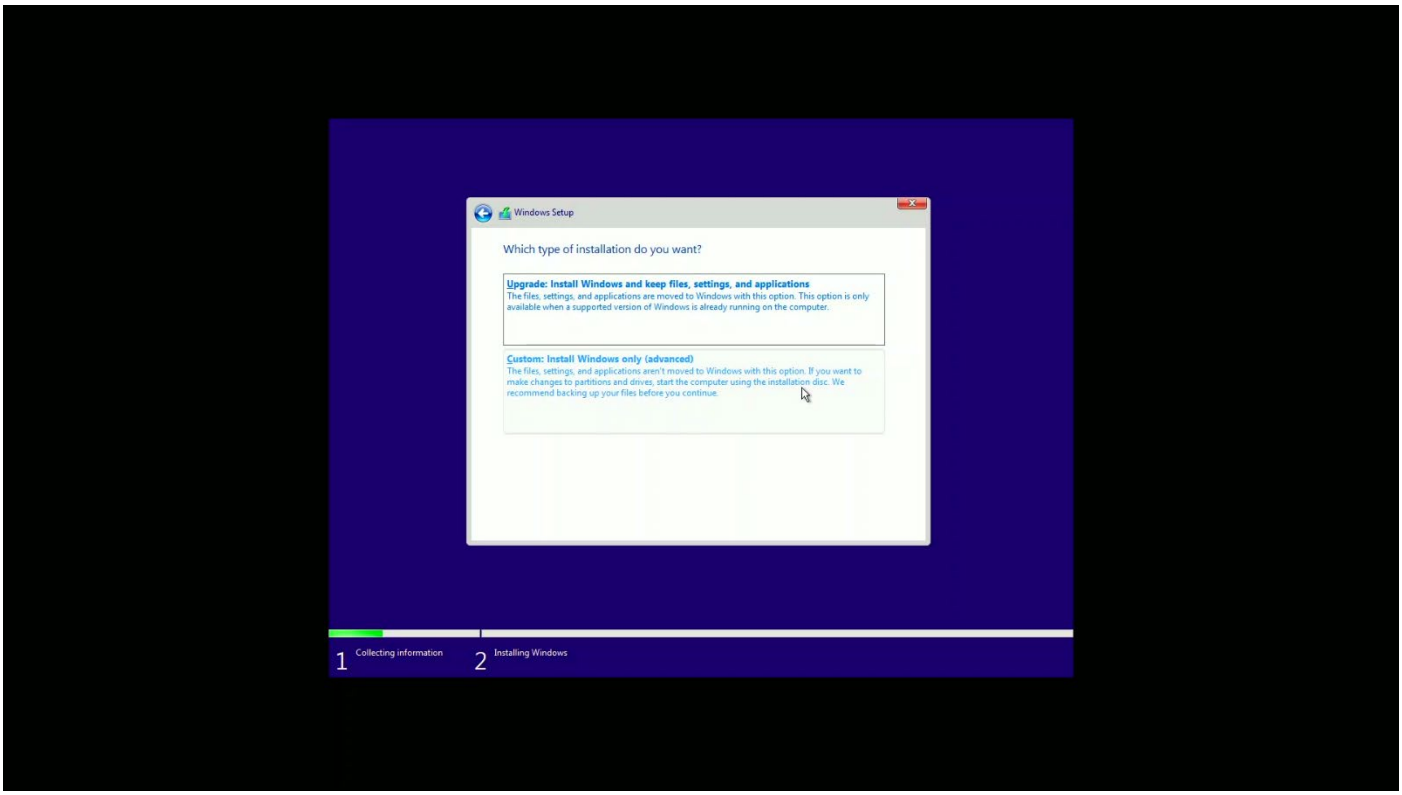
Step 4: Select Windows version and click “Next”.



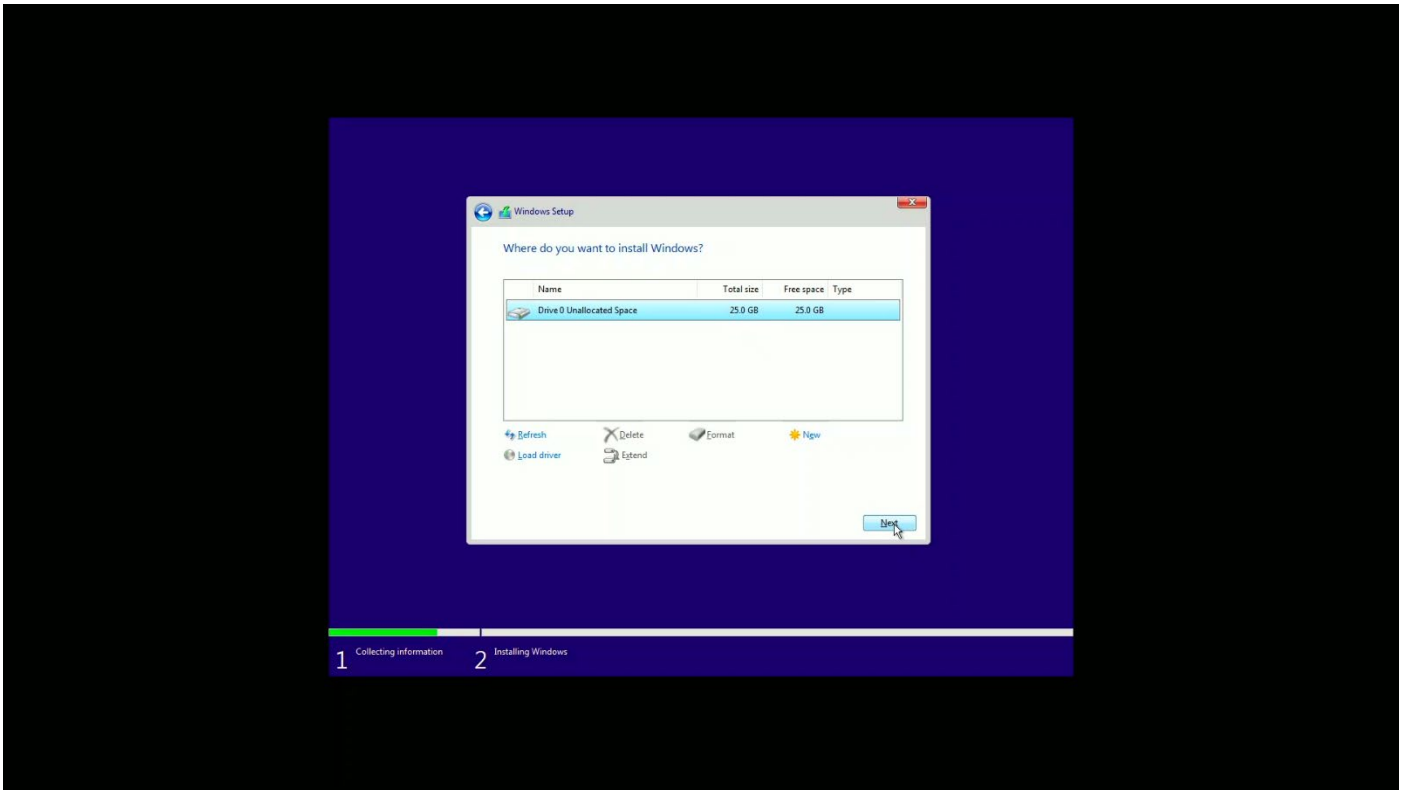
Step 5: Accept terms and conditions and click “Next”.



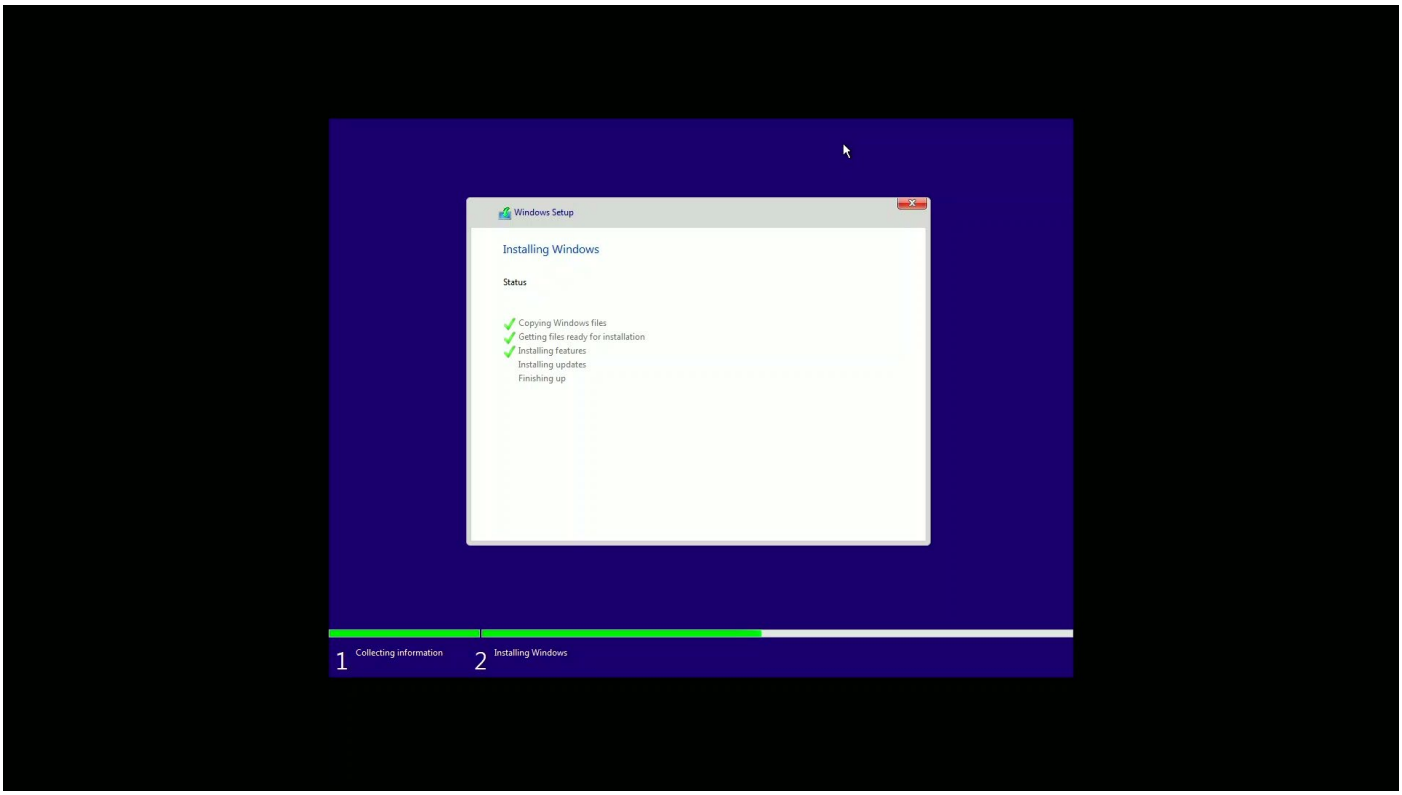
Step 6: Click “Custom: Install Windows only”.



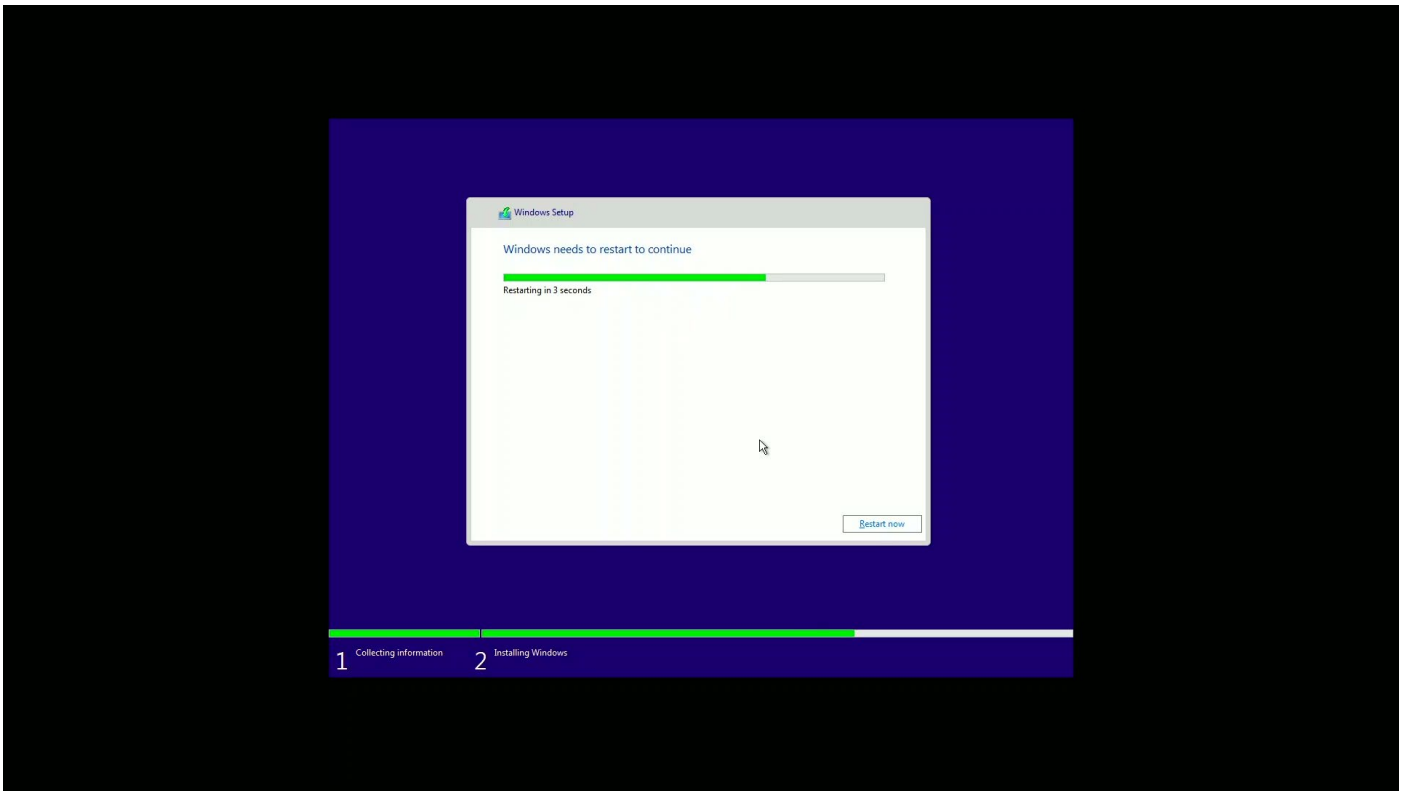
Step 7: Select the disk and click “Next”.



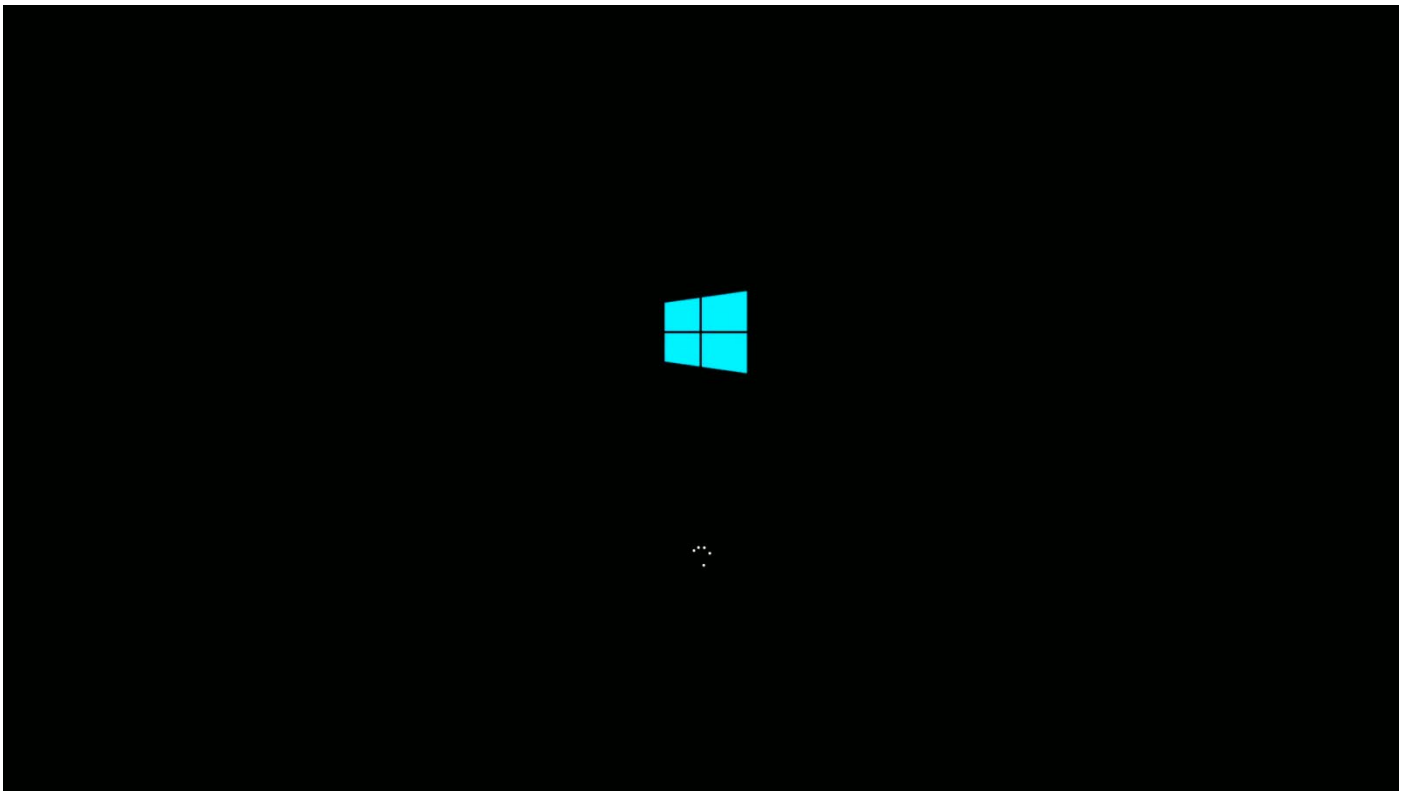
Step 8: Wait for the installation to get completed.



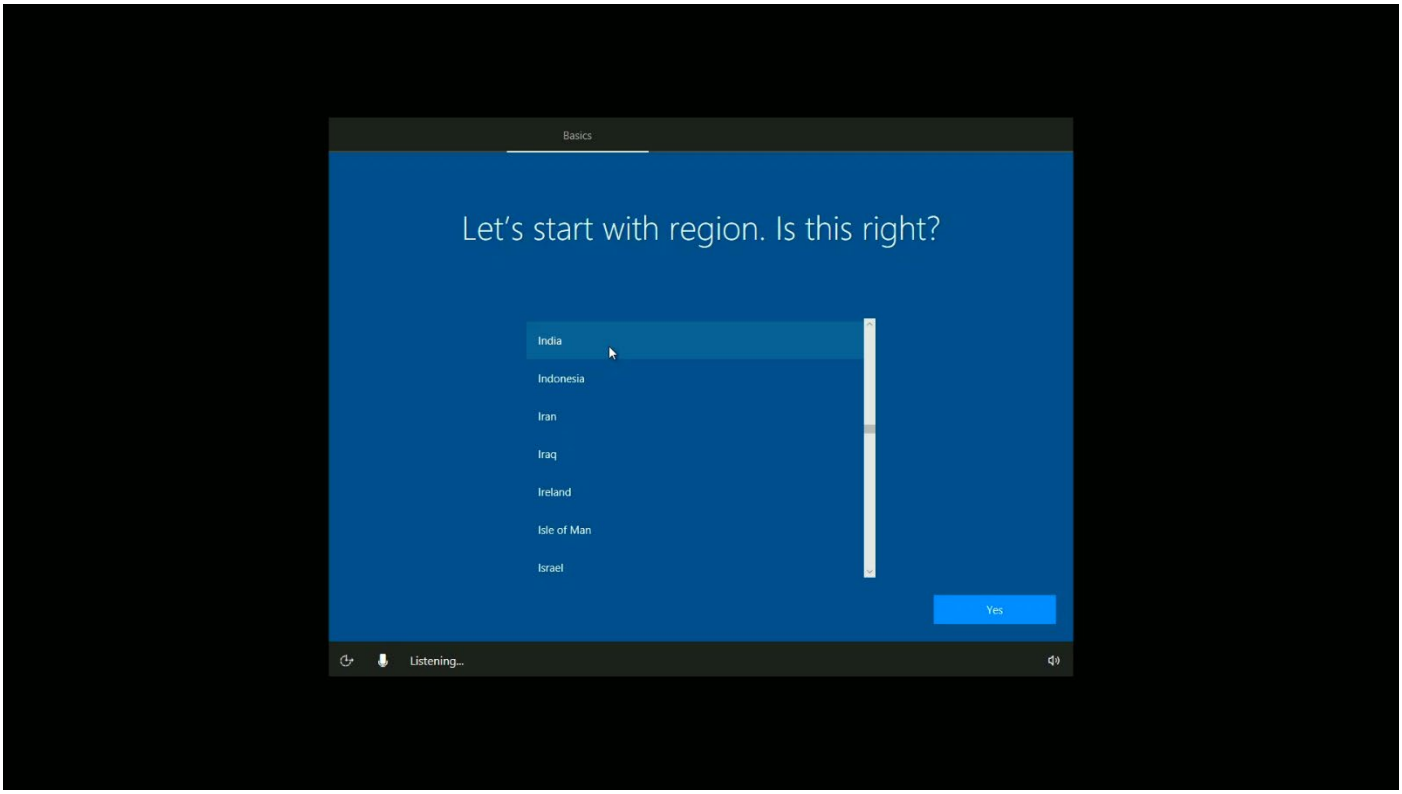
Step 9: Click “Restart Now”.



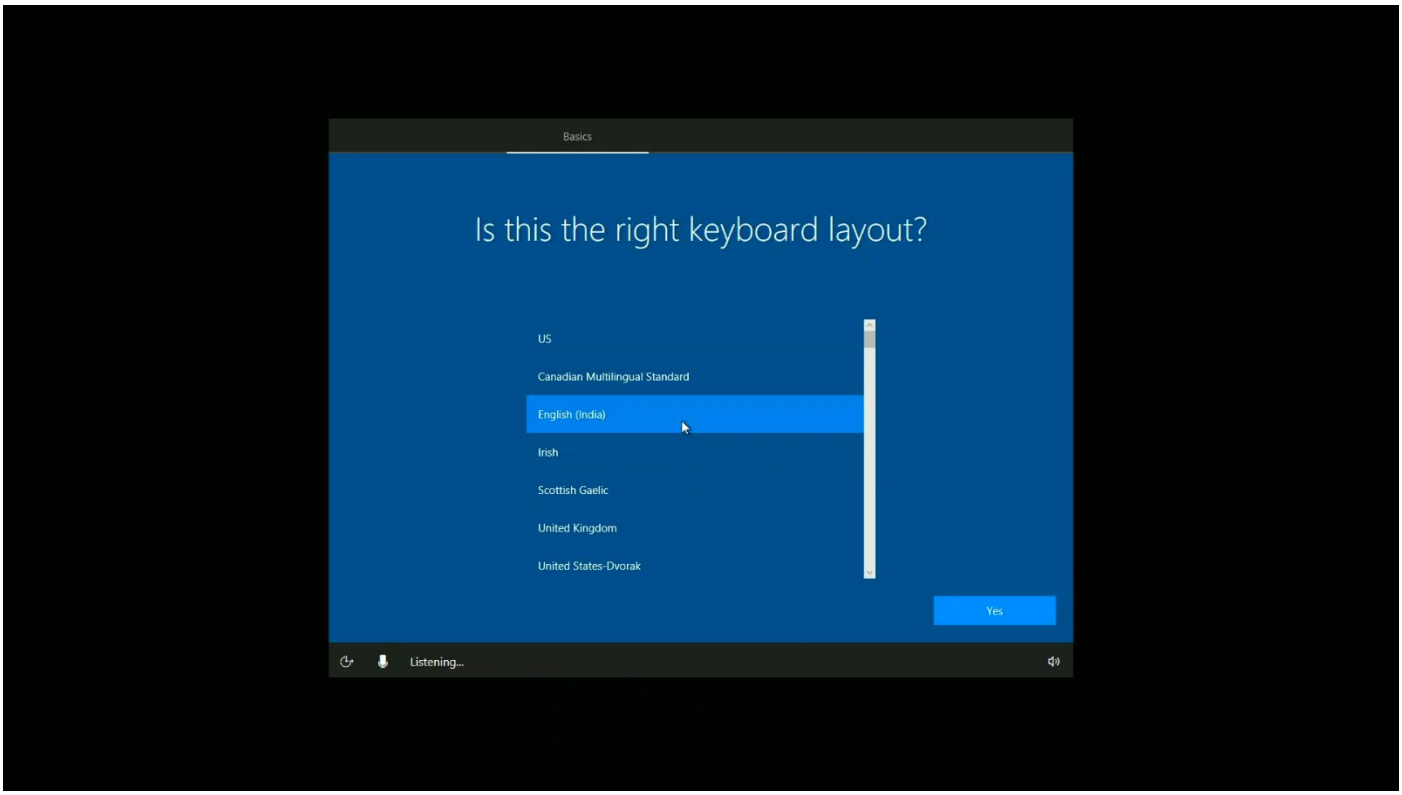
Step 10: System Reboots.



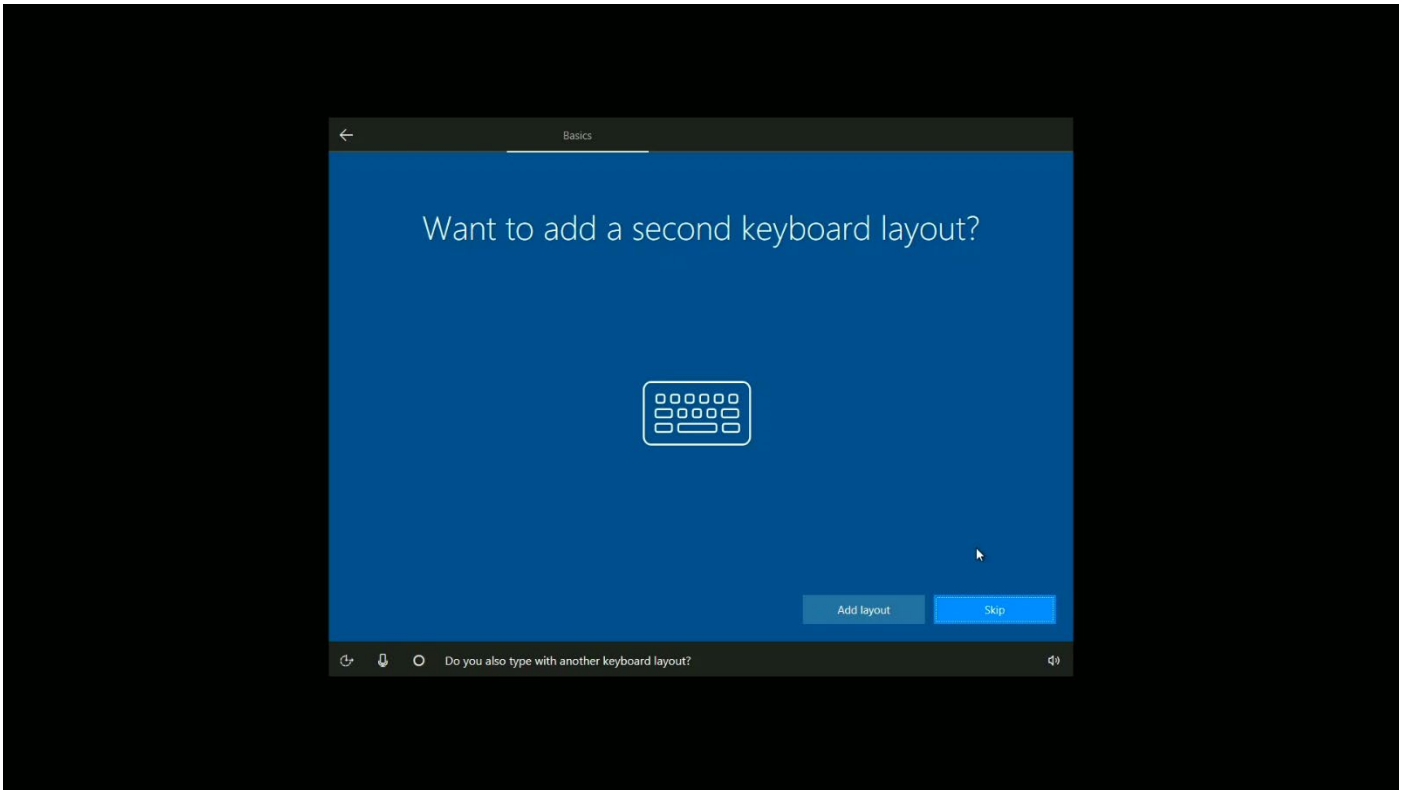
Step 11: Select your region and click “Yes”.



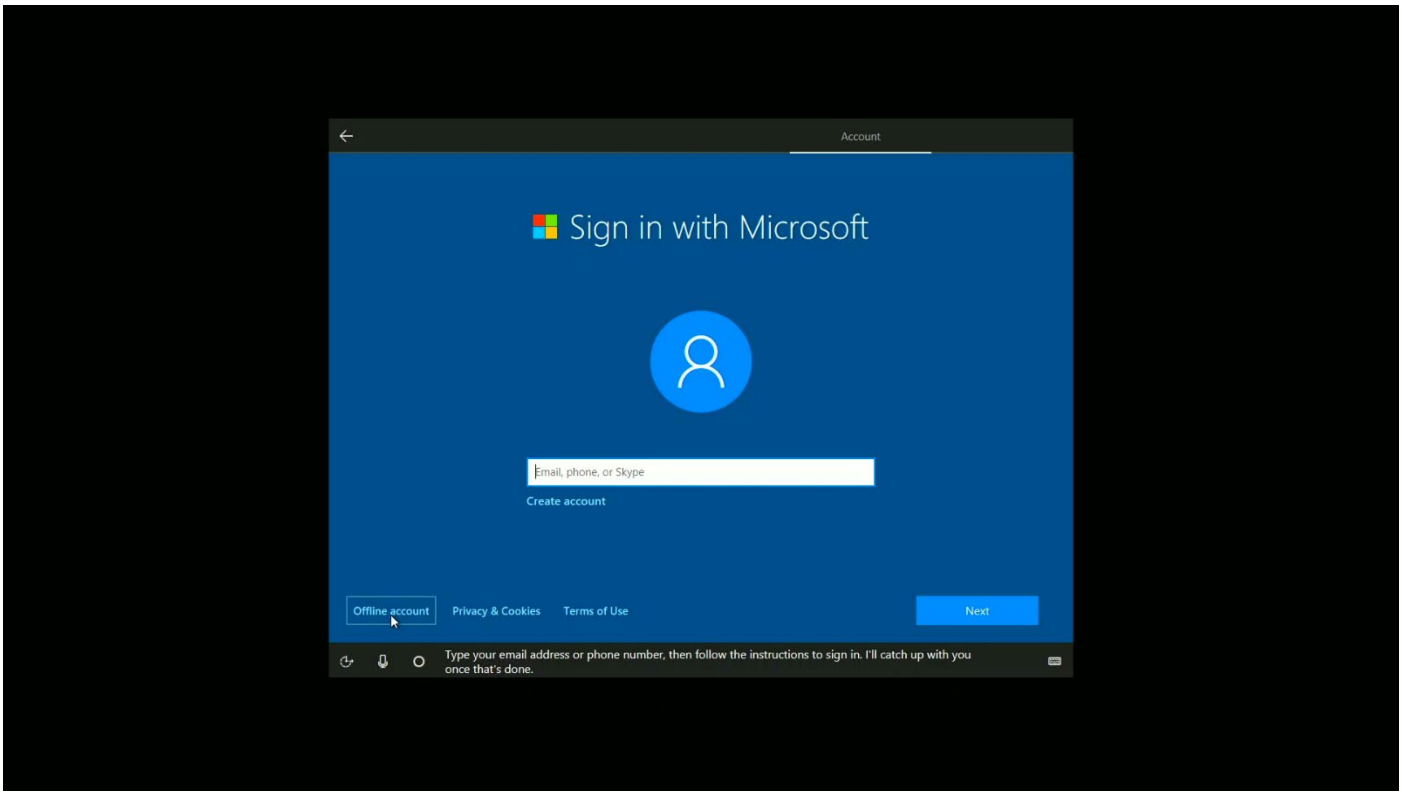
Step 12: Select Keyboard Layout and click “Yes”.



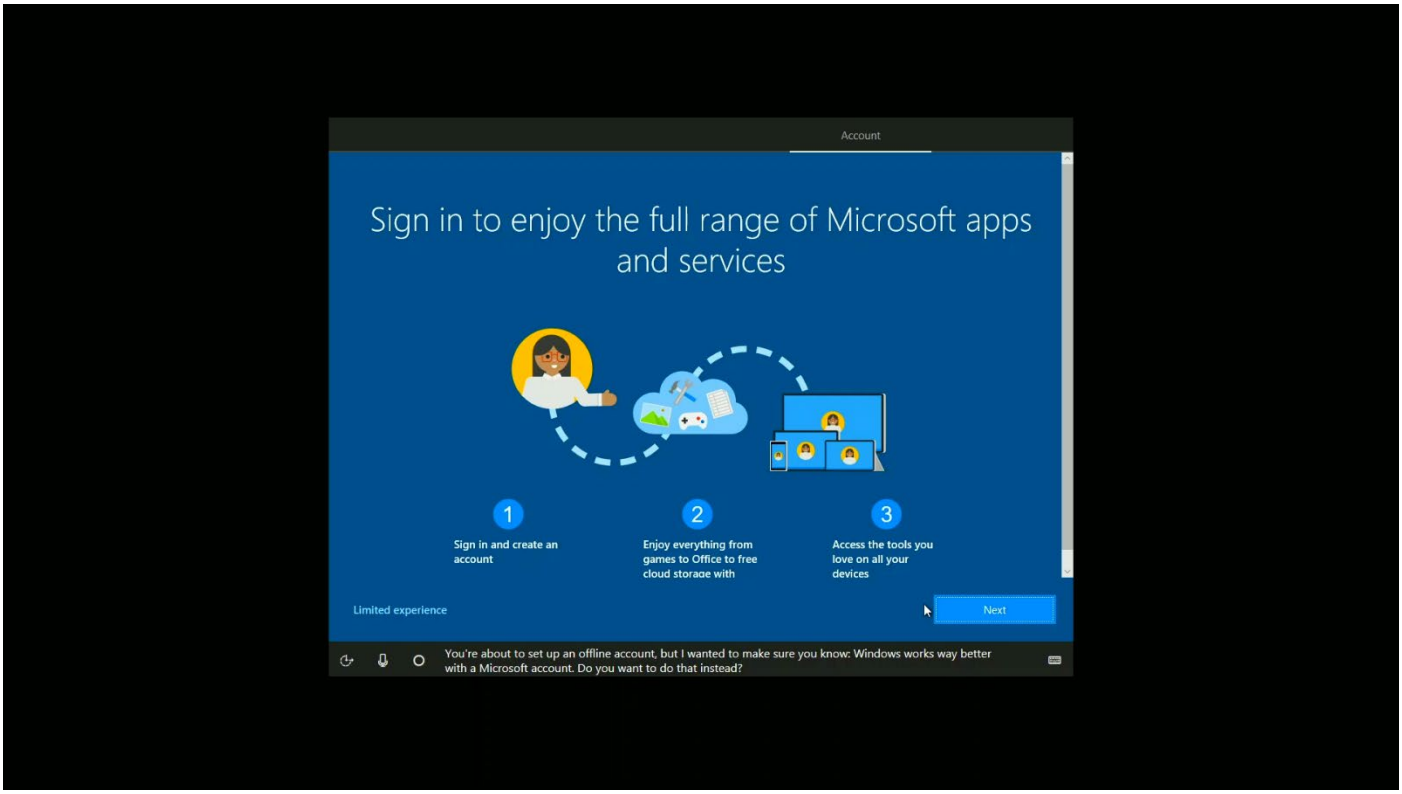
Step 13: Click “Yes”.



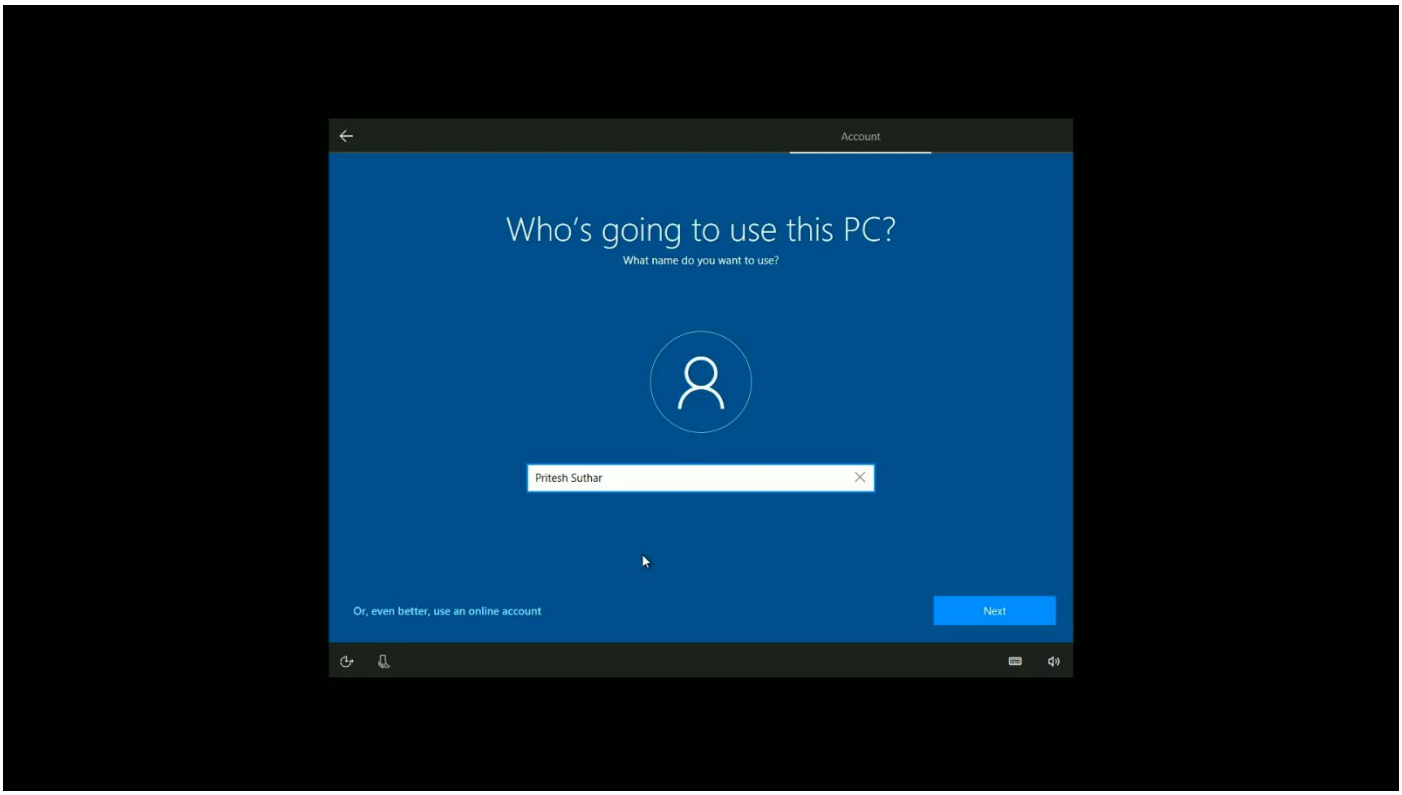
Step 14: Click “Offline Account”.



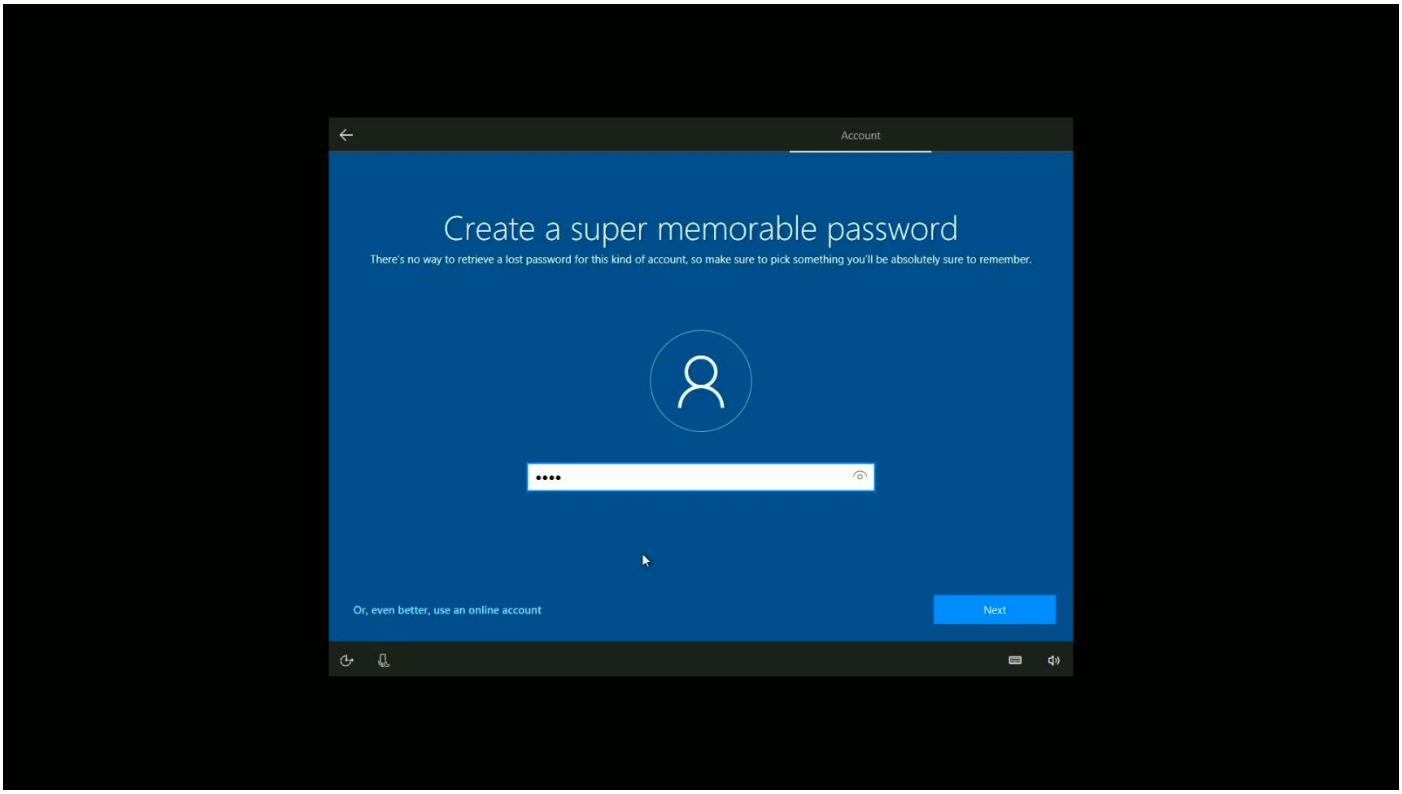
Step 15: Click “Limited Experience”.



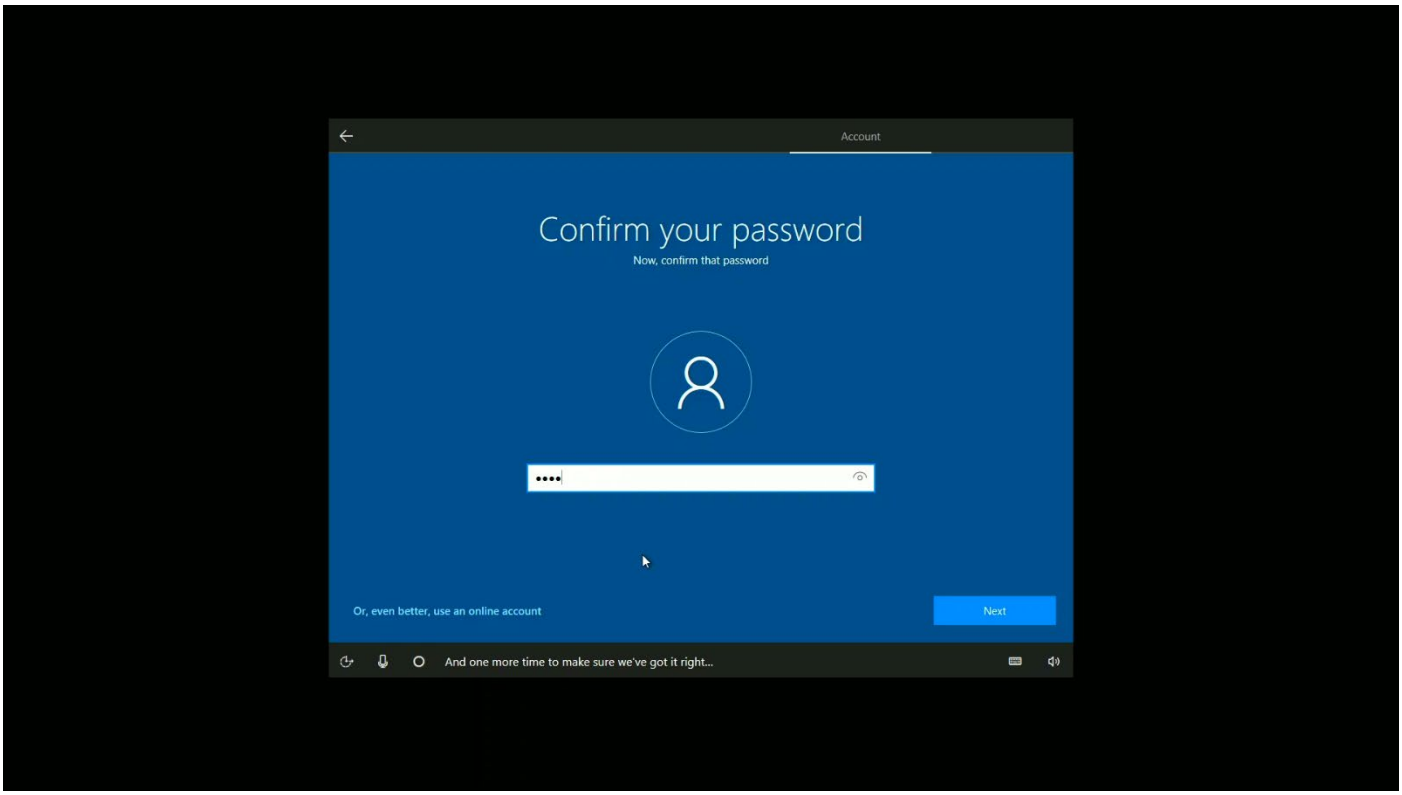
Step 16: Enter username and click “Next”.



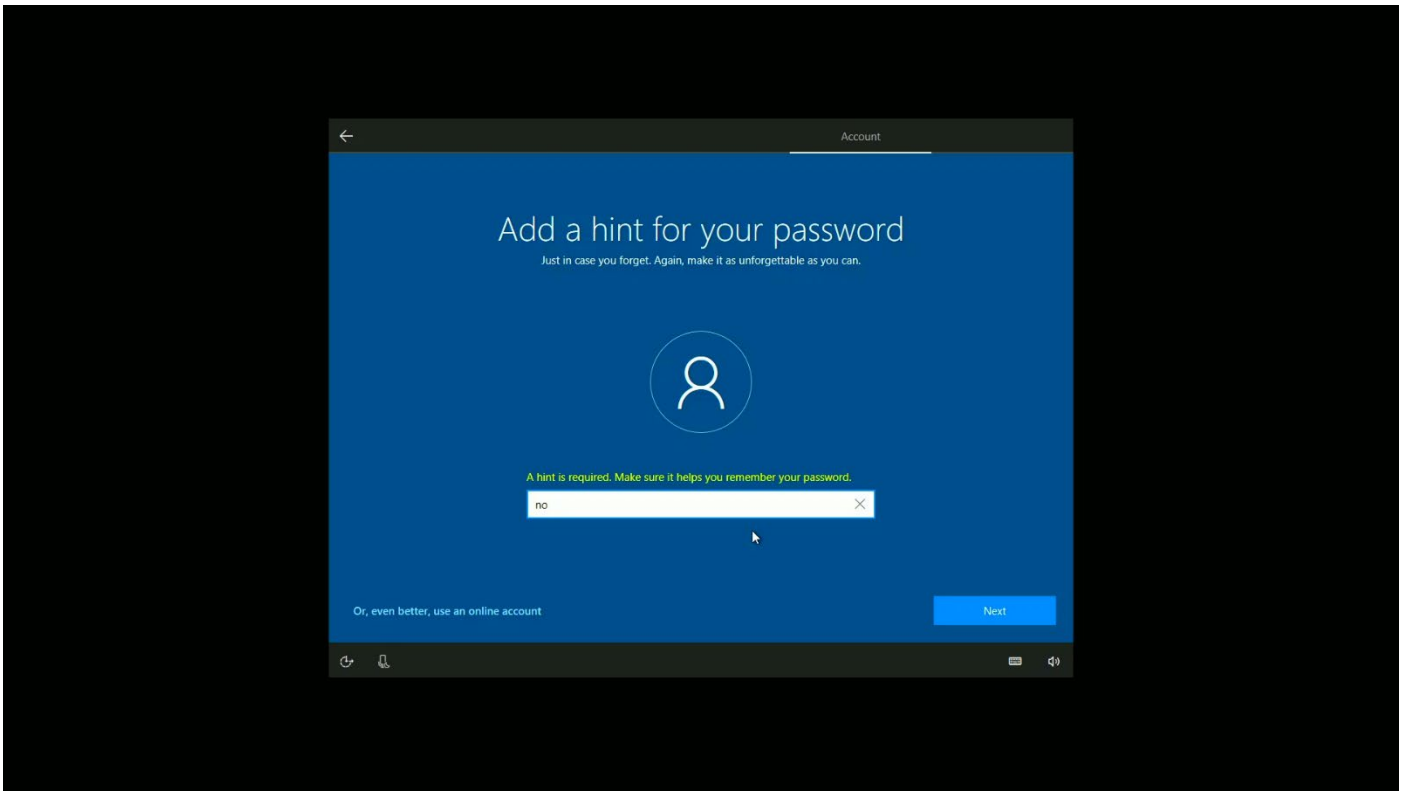
Step 17: Enter the password and click “Next”.



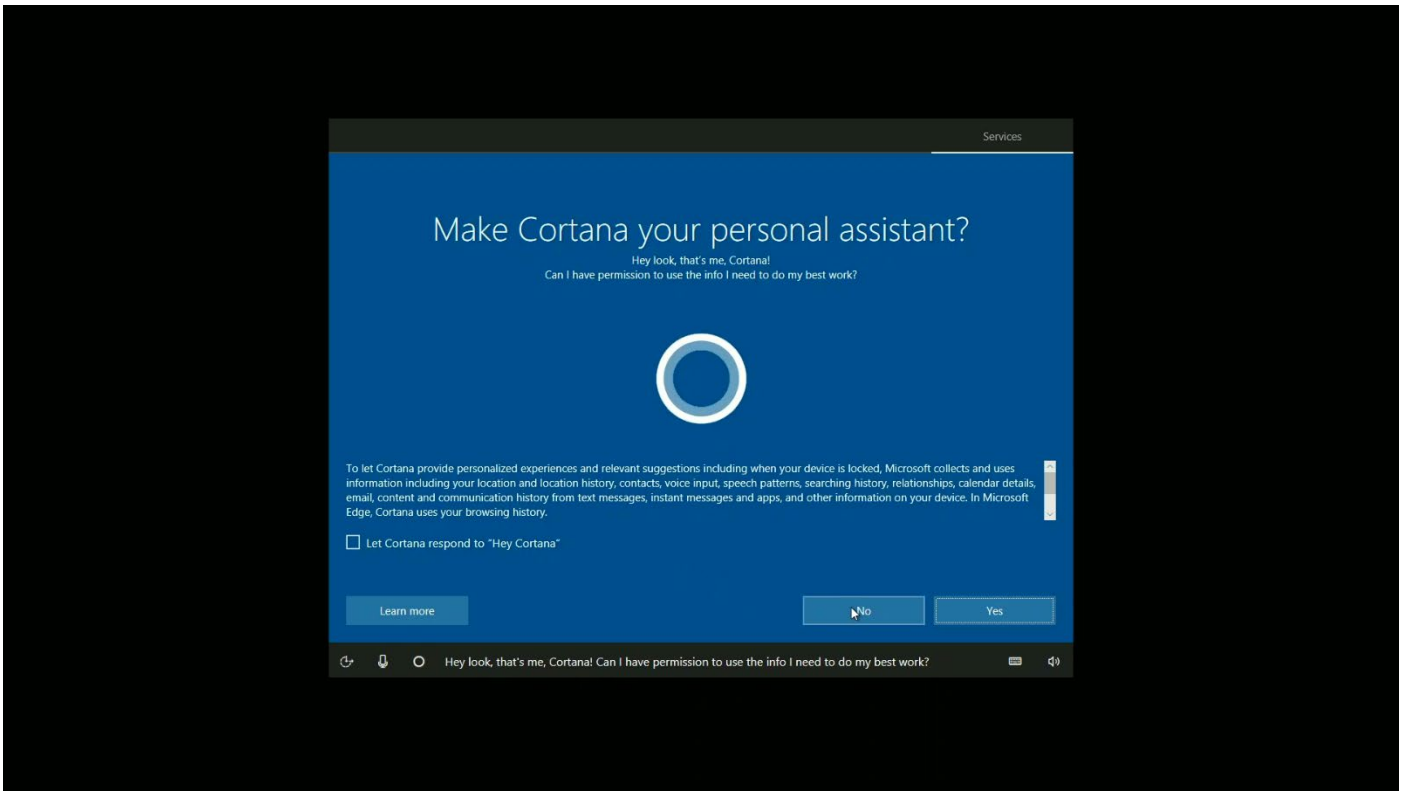
Step 18: Re-enter the password and click “Next”.



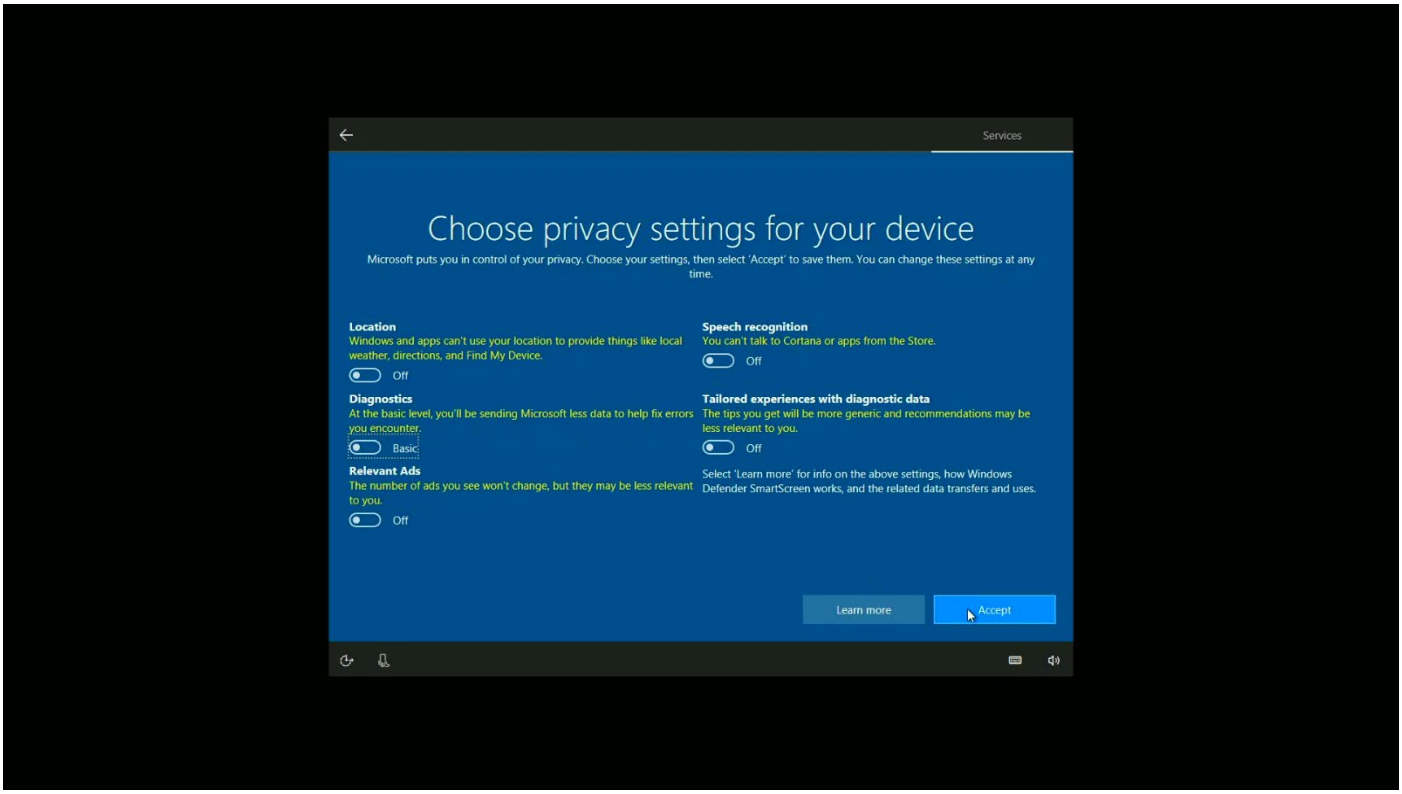
Step 19: Enter a hint for your password click “Next”.



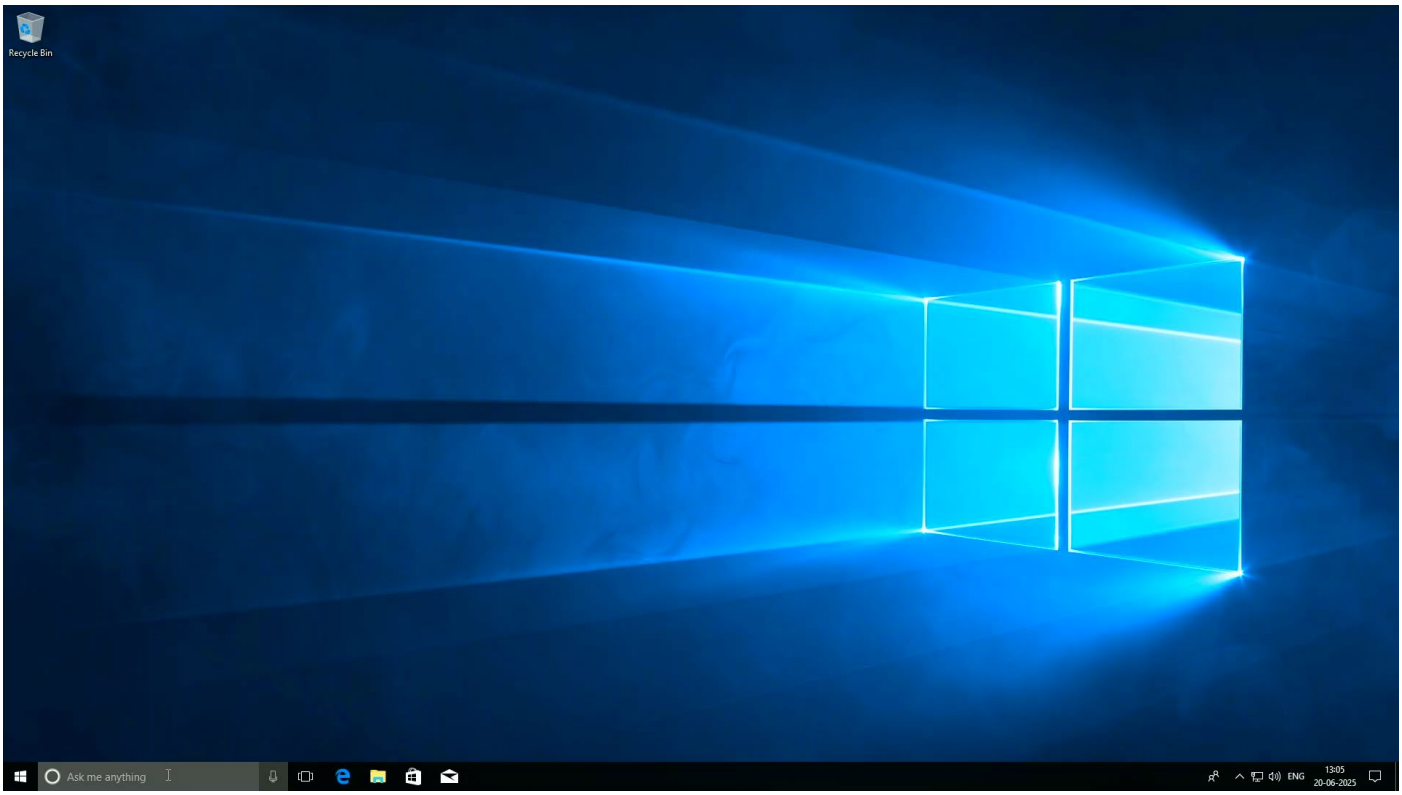
Step 20: Click “No”.



Step 21: Uncheck all the checkboxes and click “Next”.

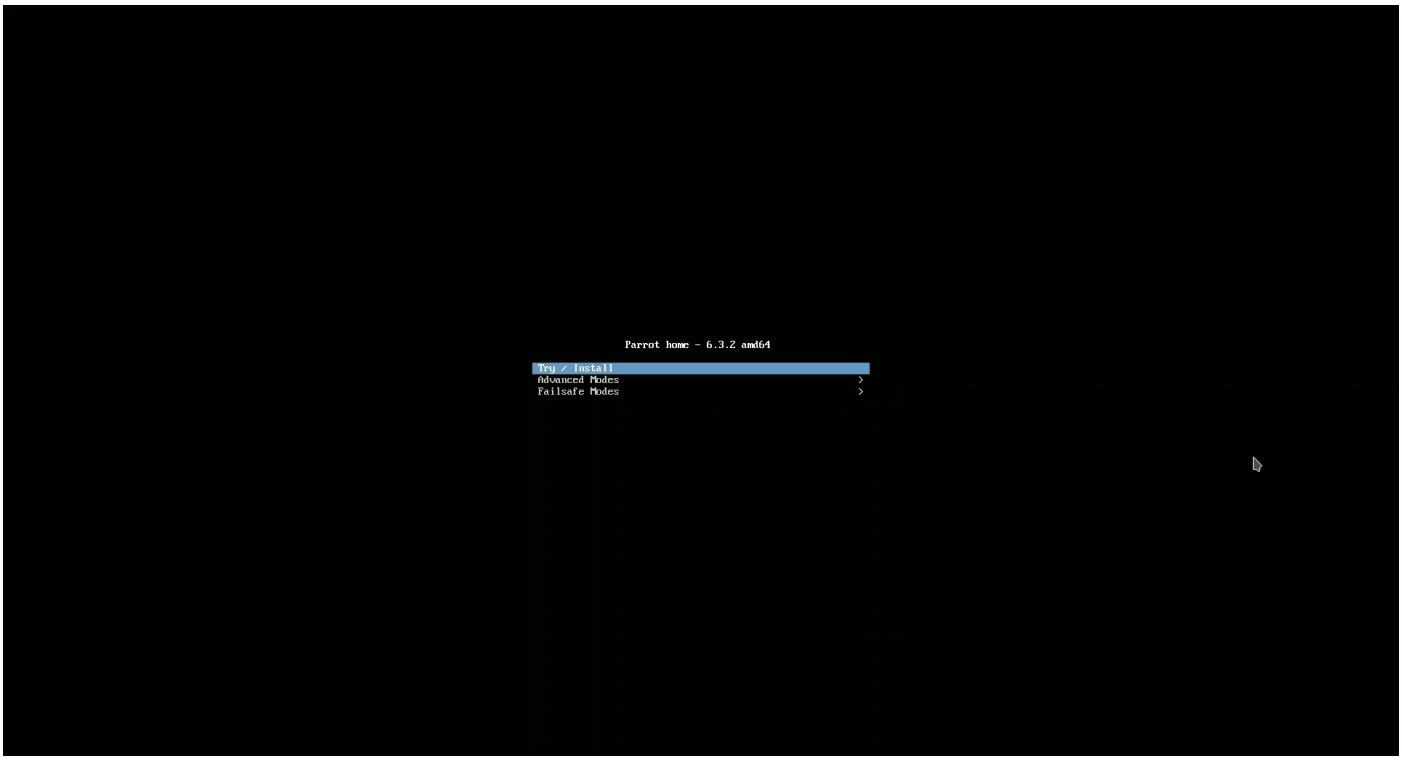


Step 22: Windows is successfully installed.

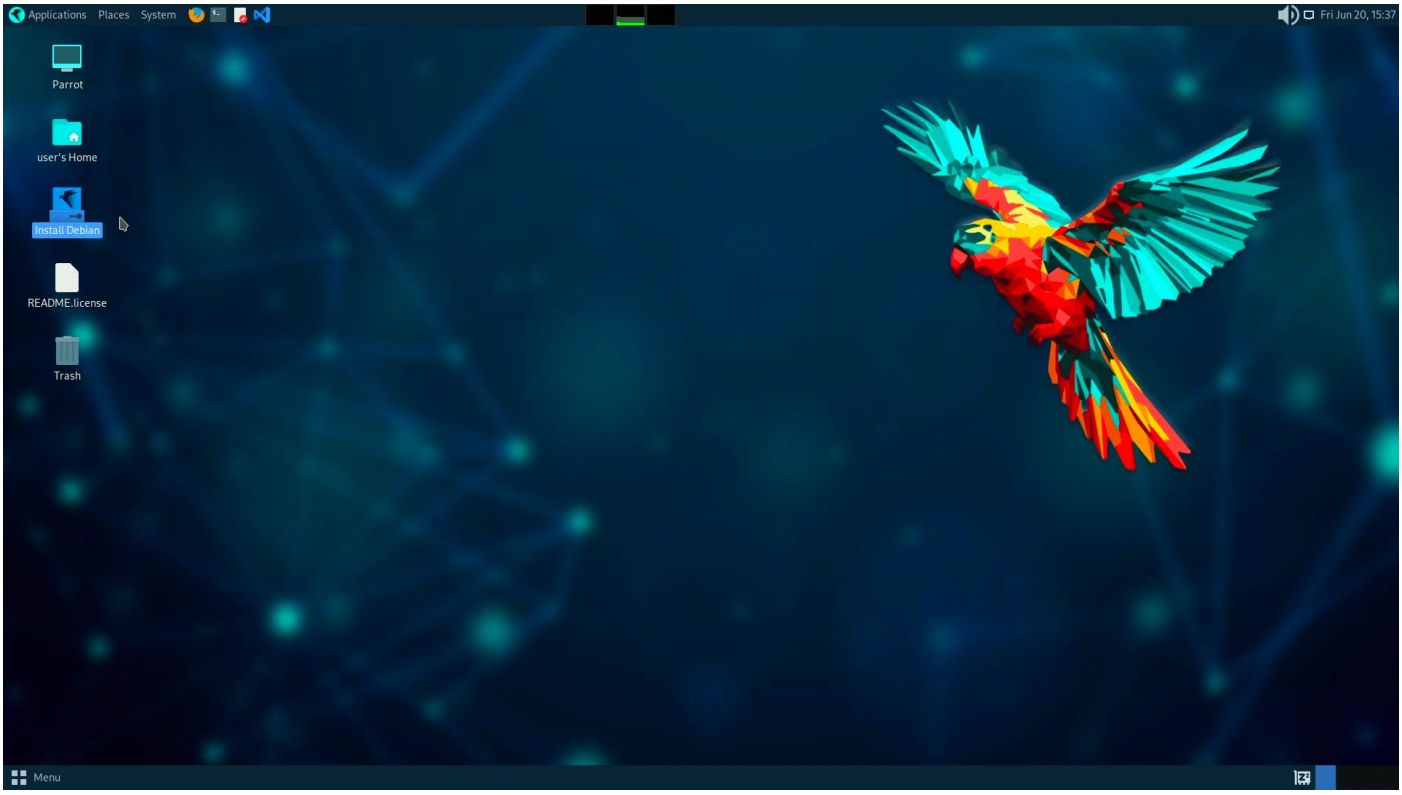


Parrot OS:

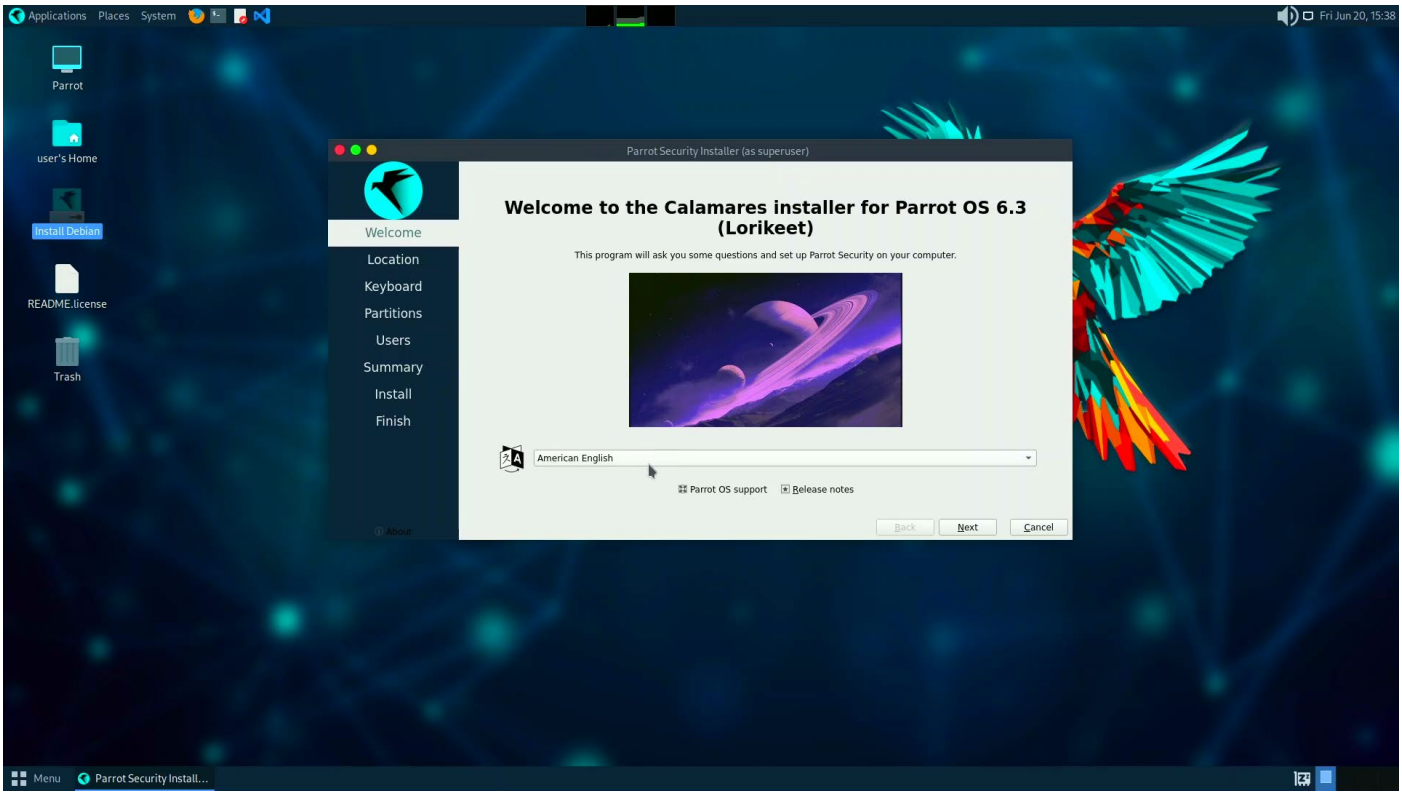
Step 1: Select “Try/Install” and press Enter.



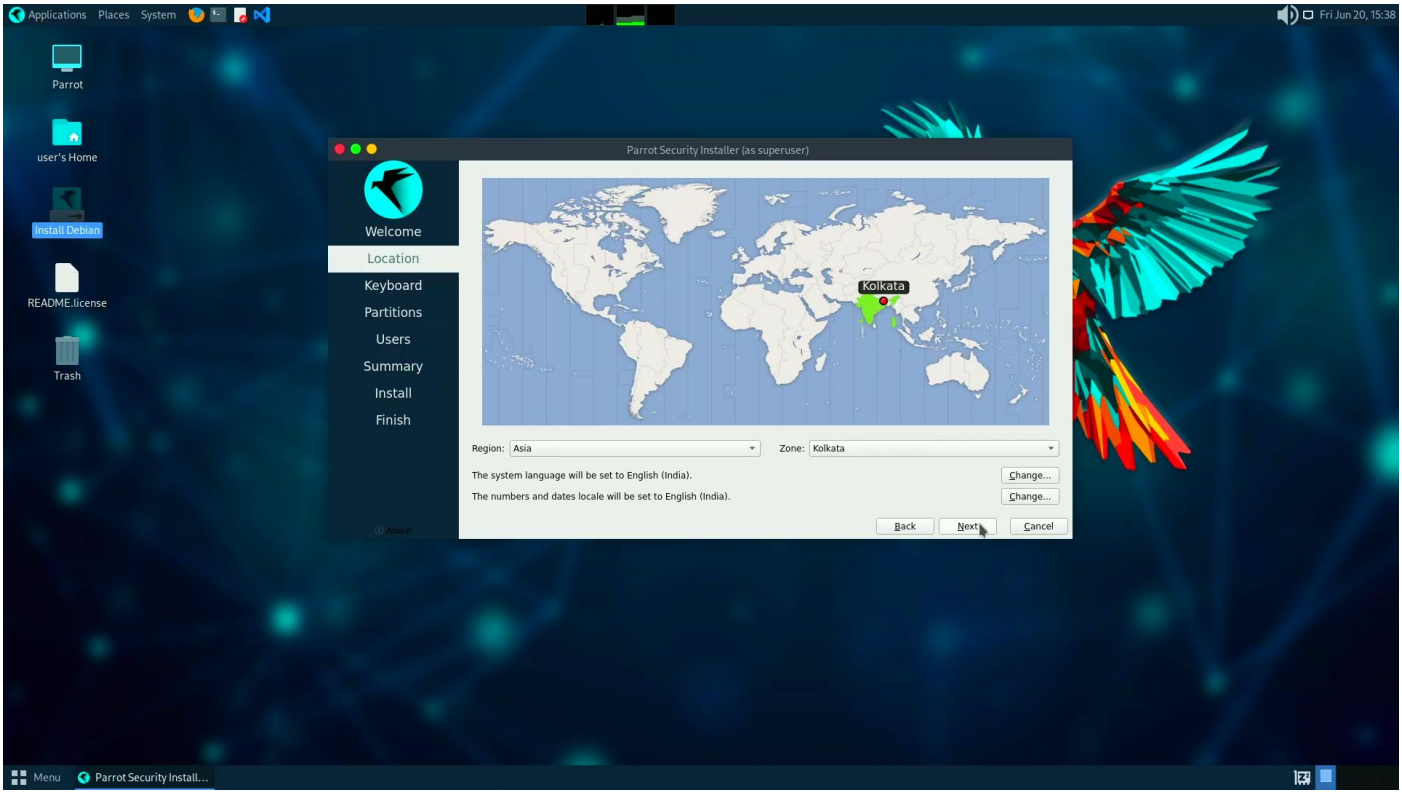
Step 2: Open “Install Debian”.



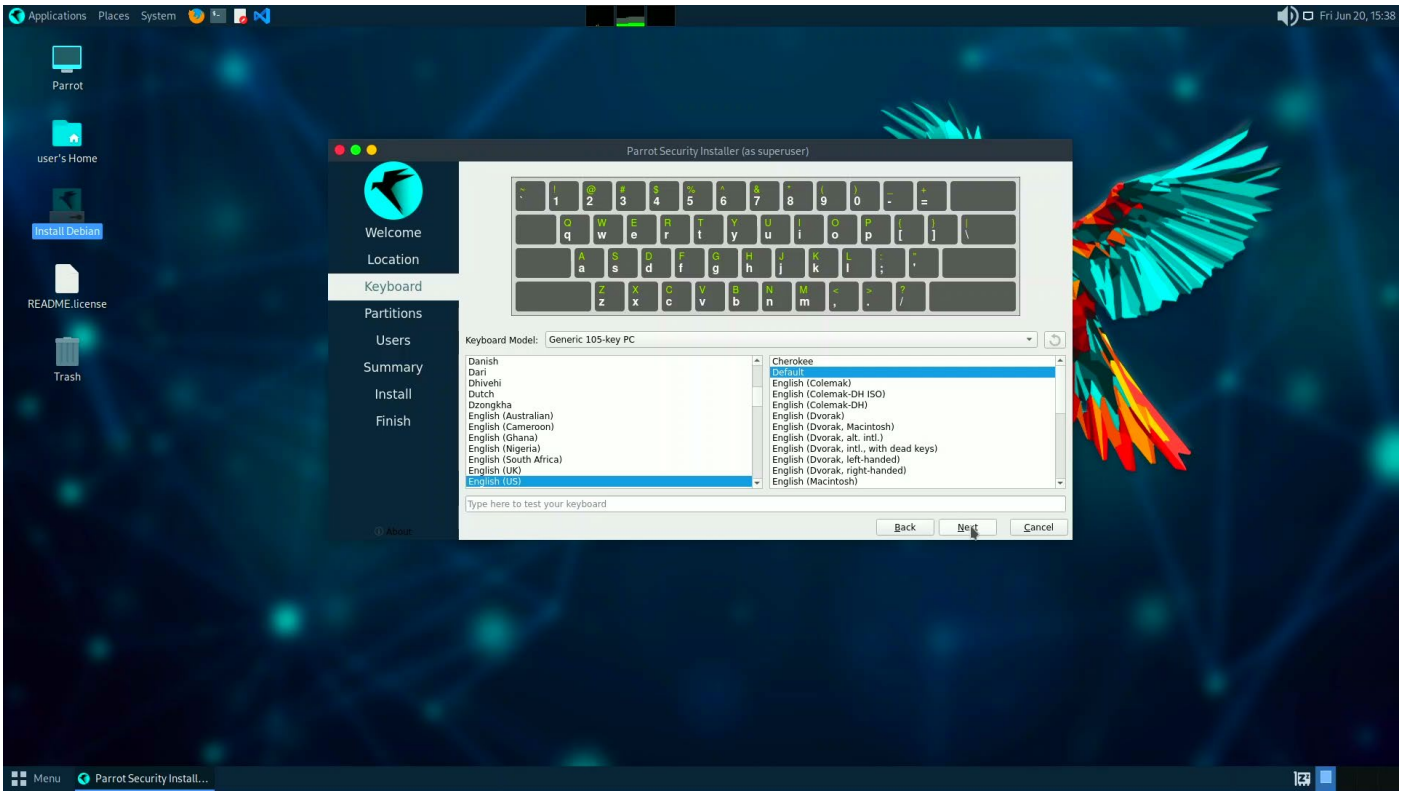
Step 3: Select system language and click “Next”.



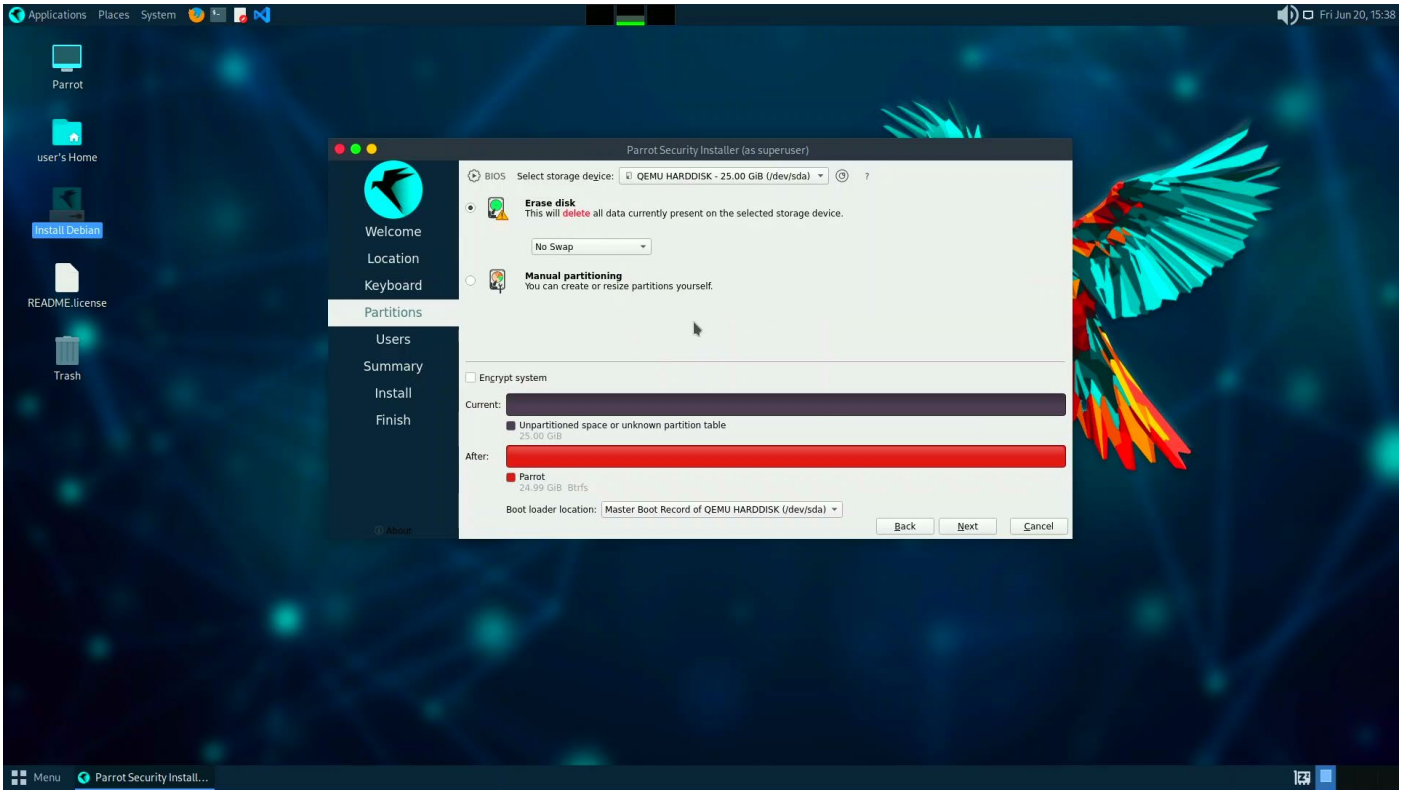
Step 4: Select region, time zone and click “Next”.



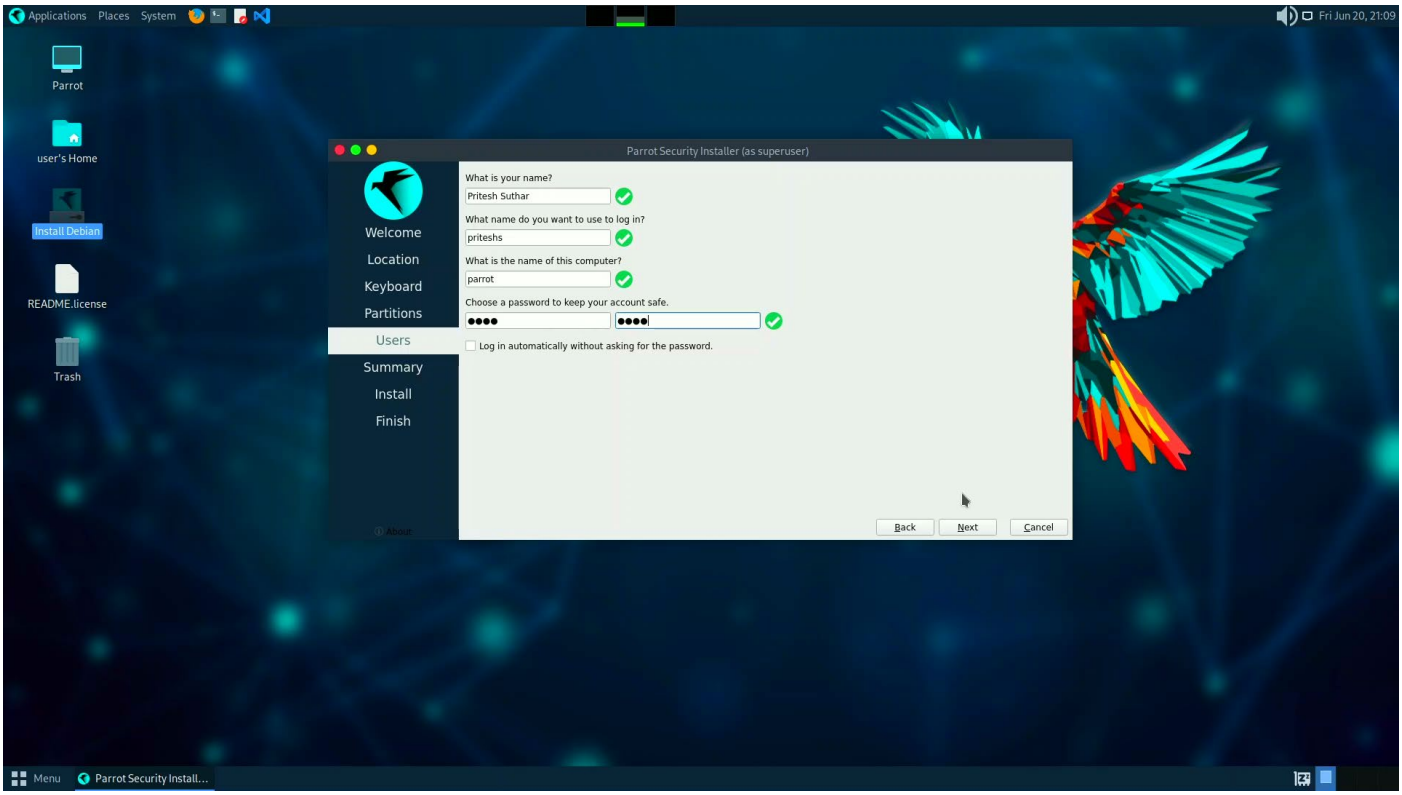
Step 5: Select keyboard language, layout and click “Next”.



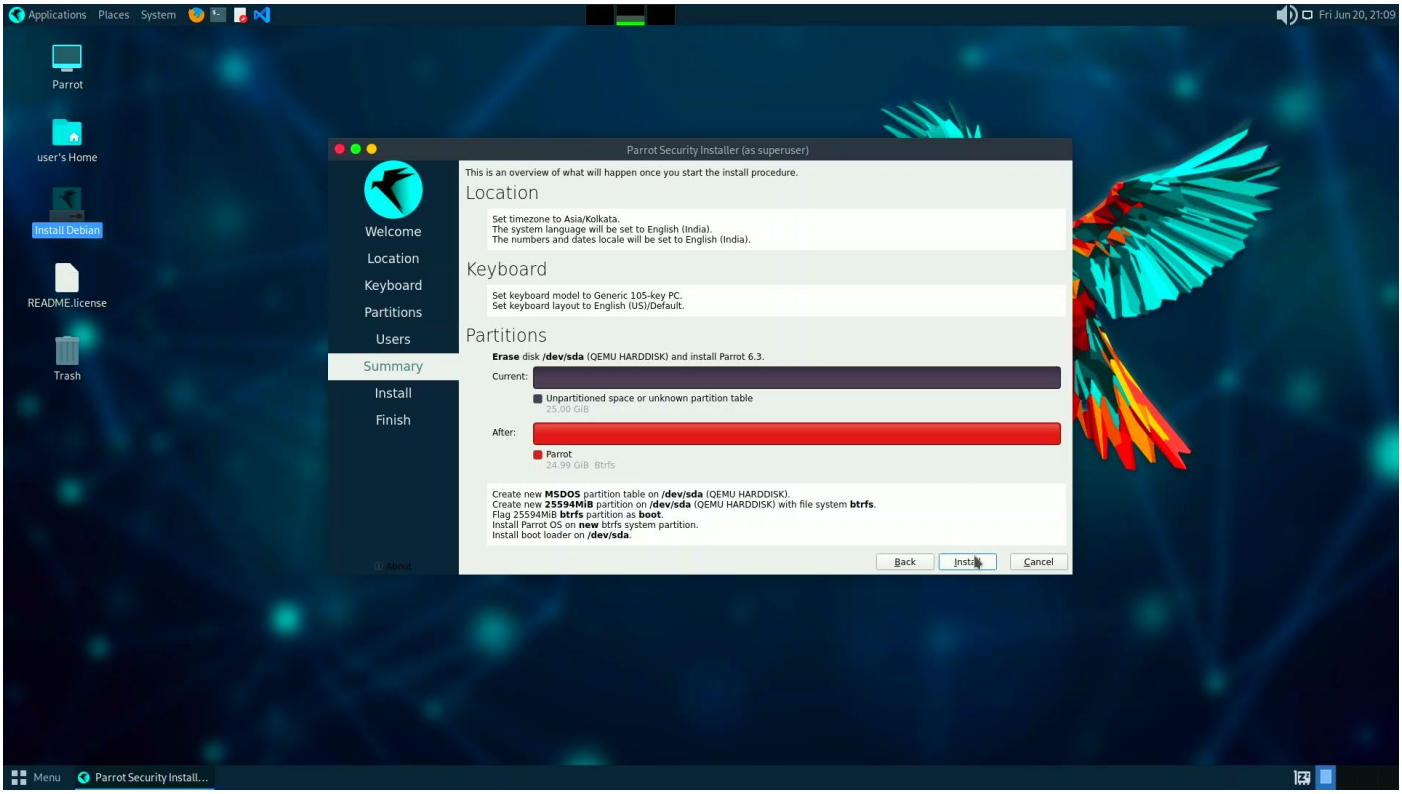
Step 6: Select “Erase Disk” click “Next”.



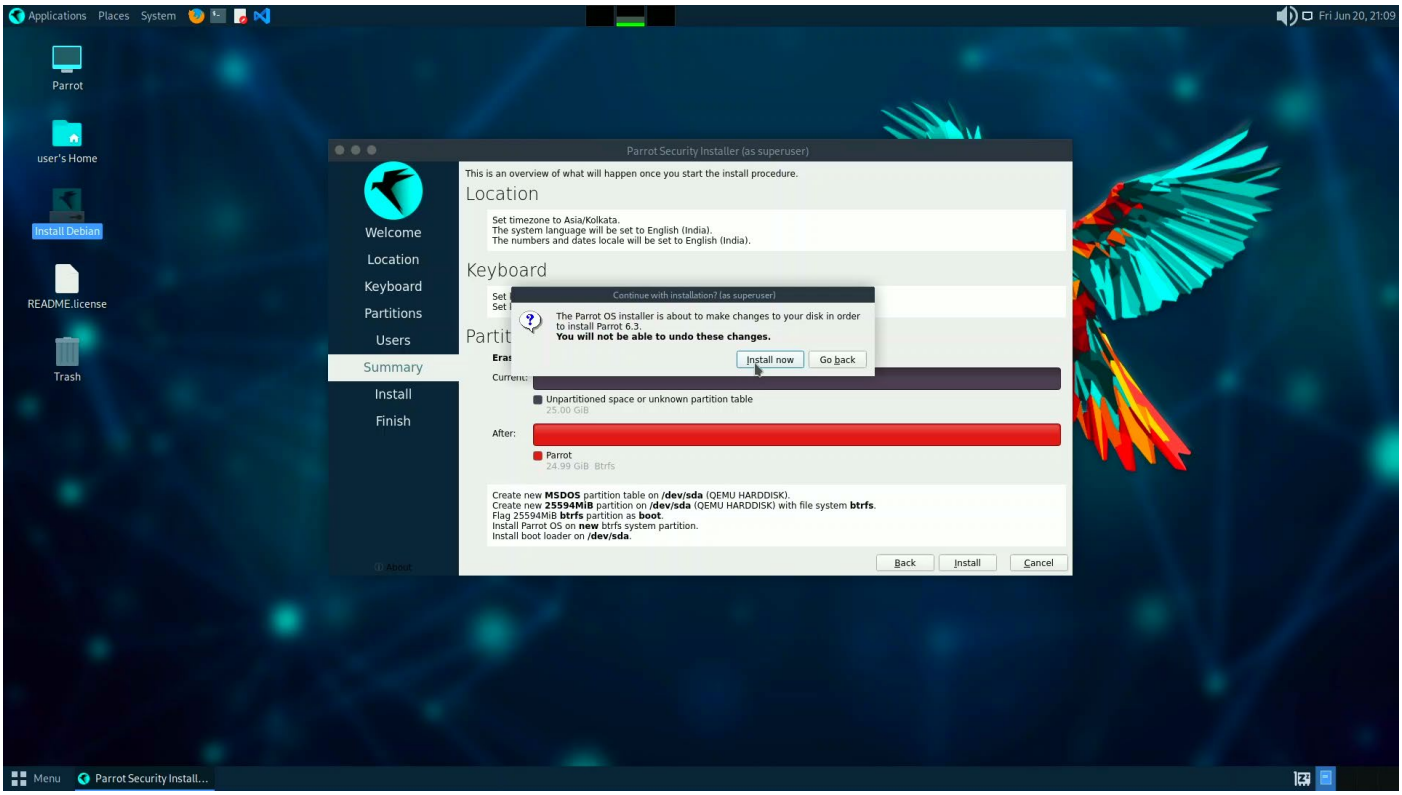
Step 7: Enter your name, username, device name, password click “Next”.



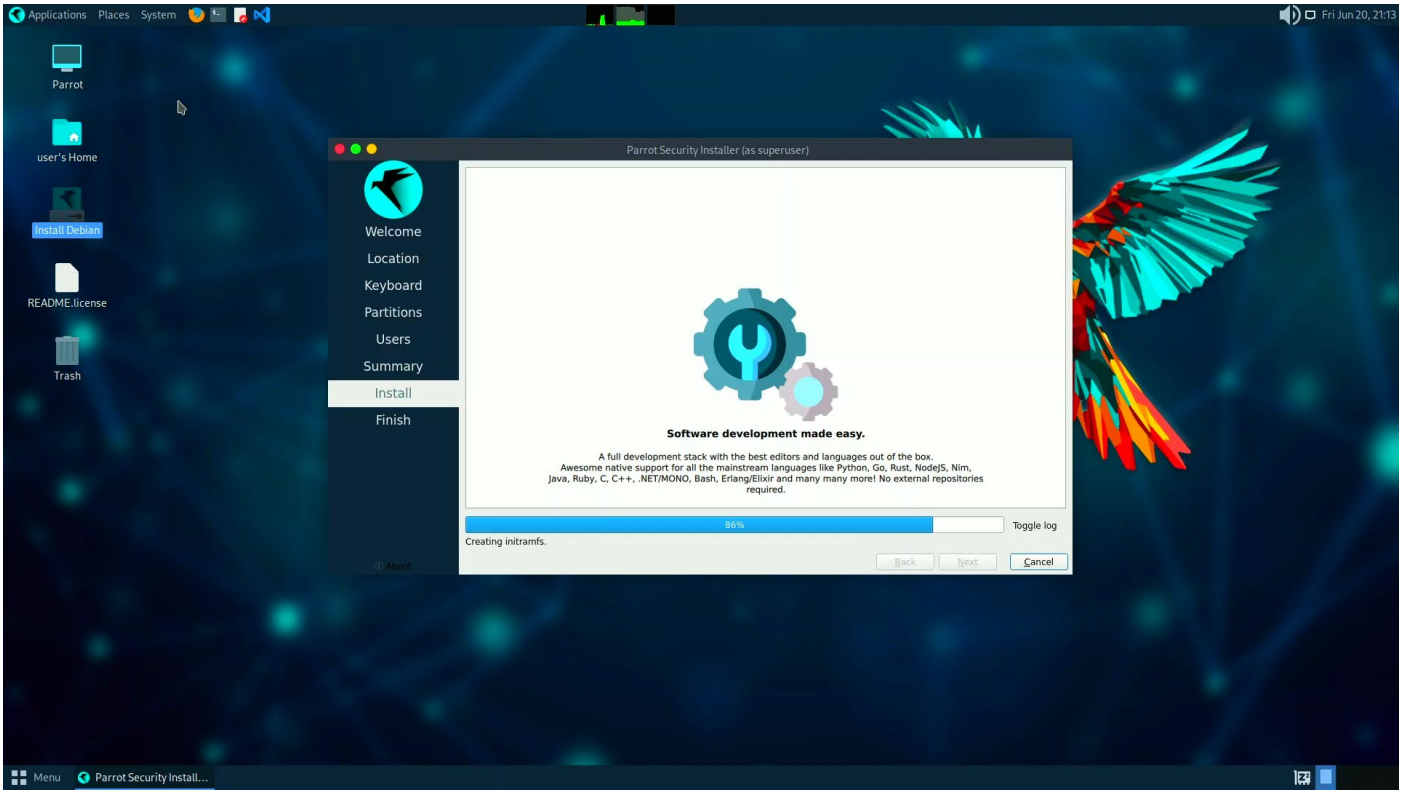
Step 8: Verify the information click “Next”.



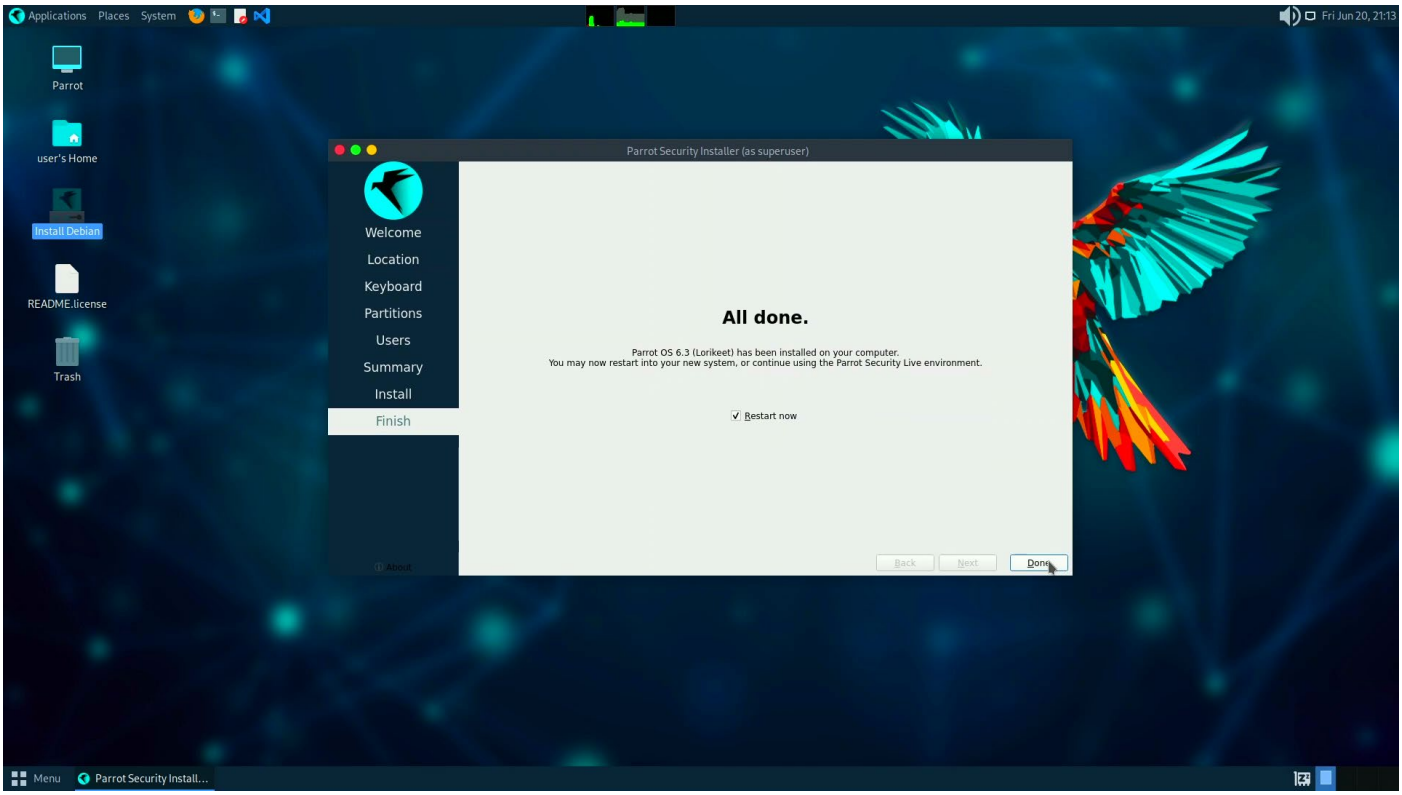
Step 9: Click “Install Now”.



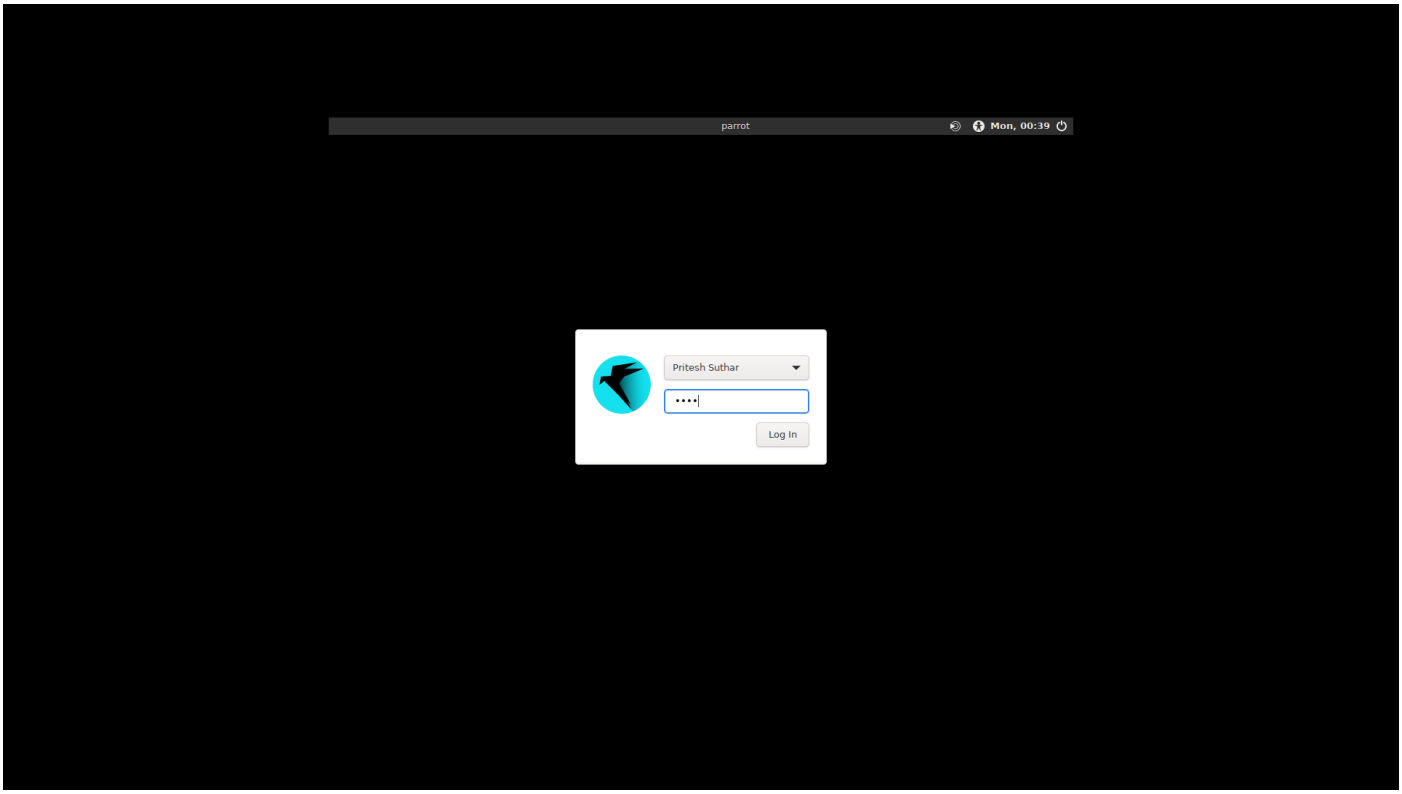
Step 10: Wait for the installation to get completed.



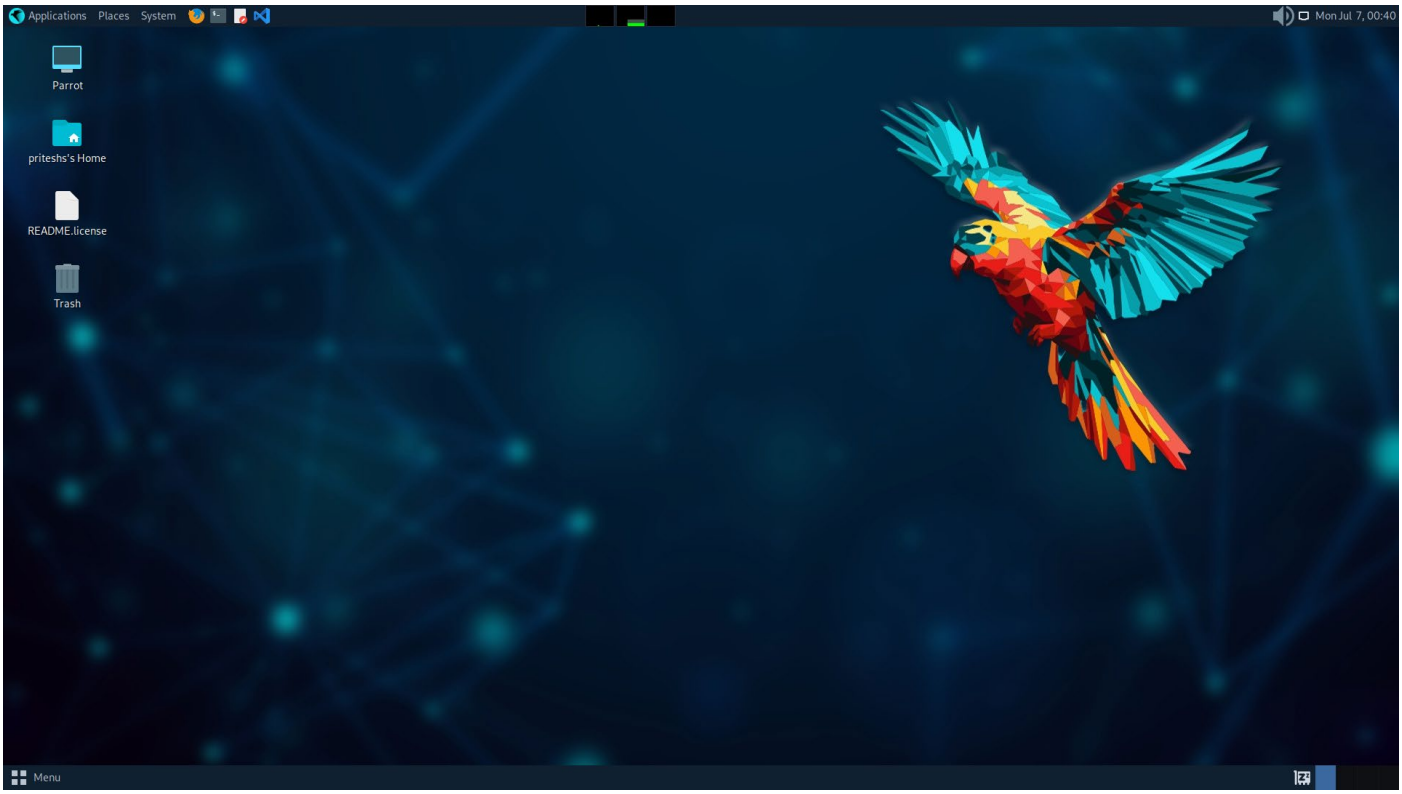
Step 11: Click “Done”



Step 12: Enter your password.

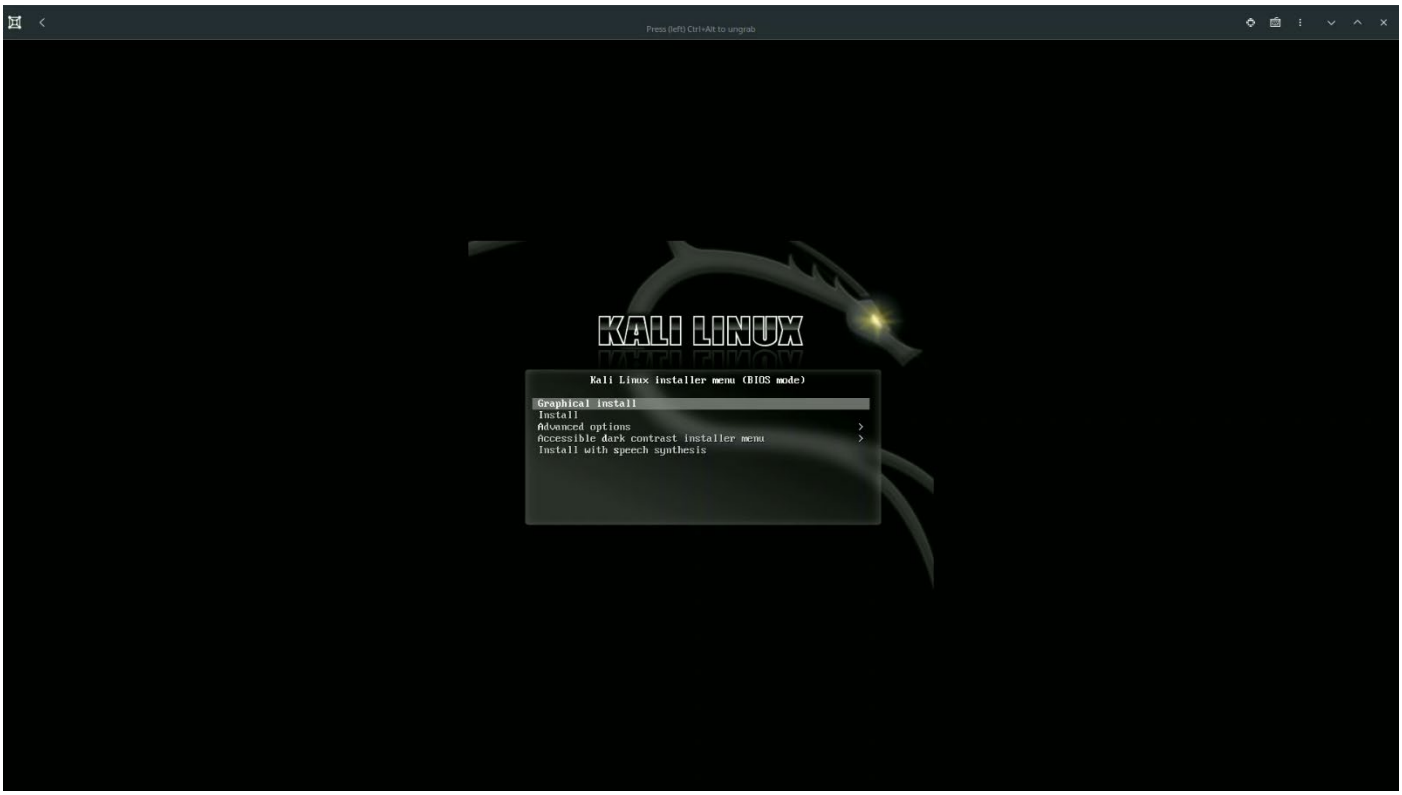


Step 13: Parrot OS is successfully installed.

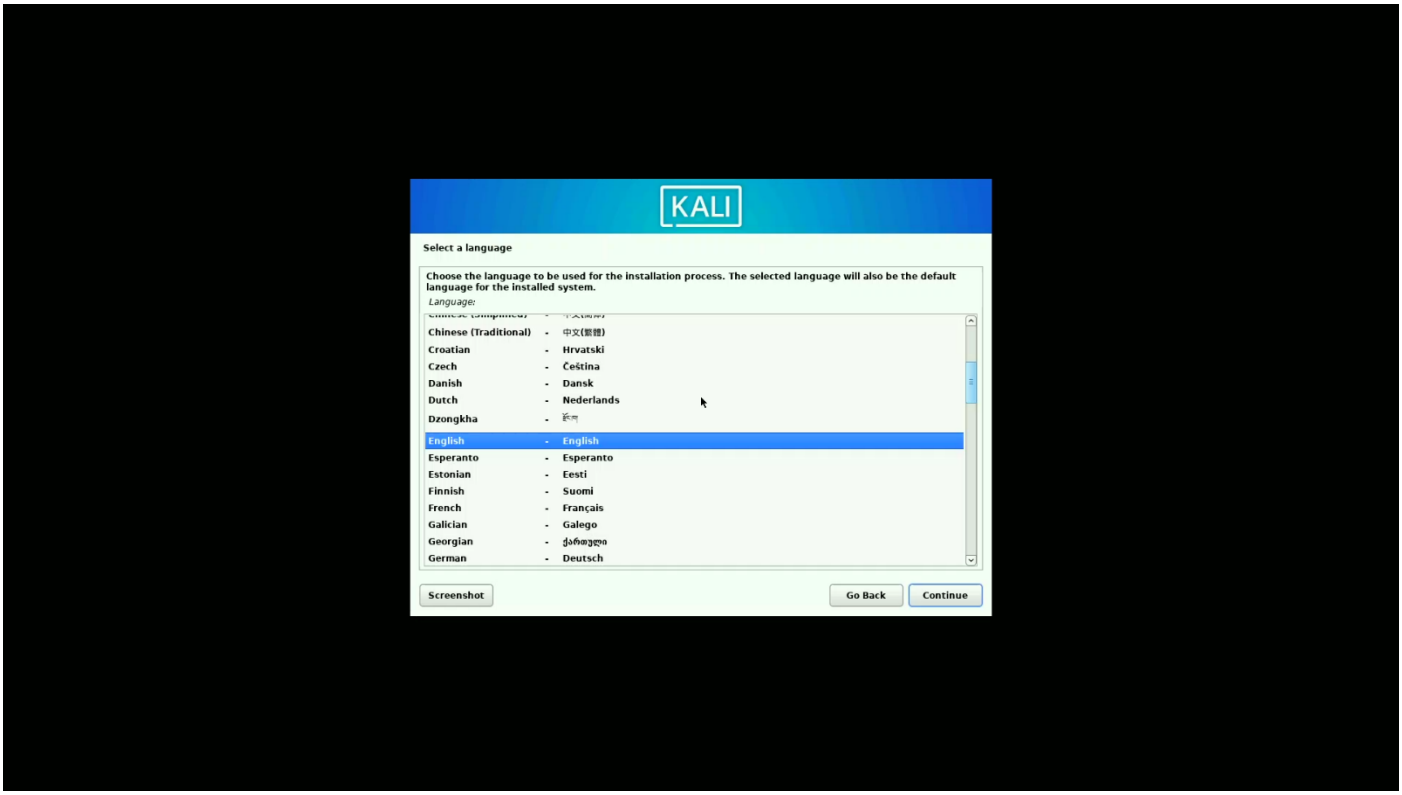


Kali Linux:

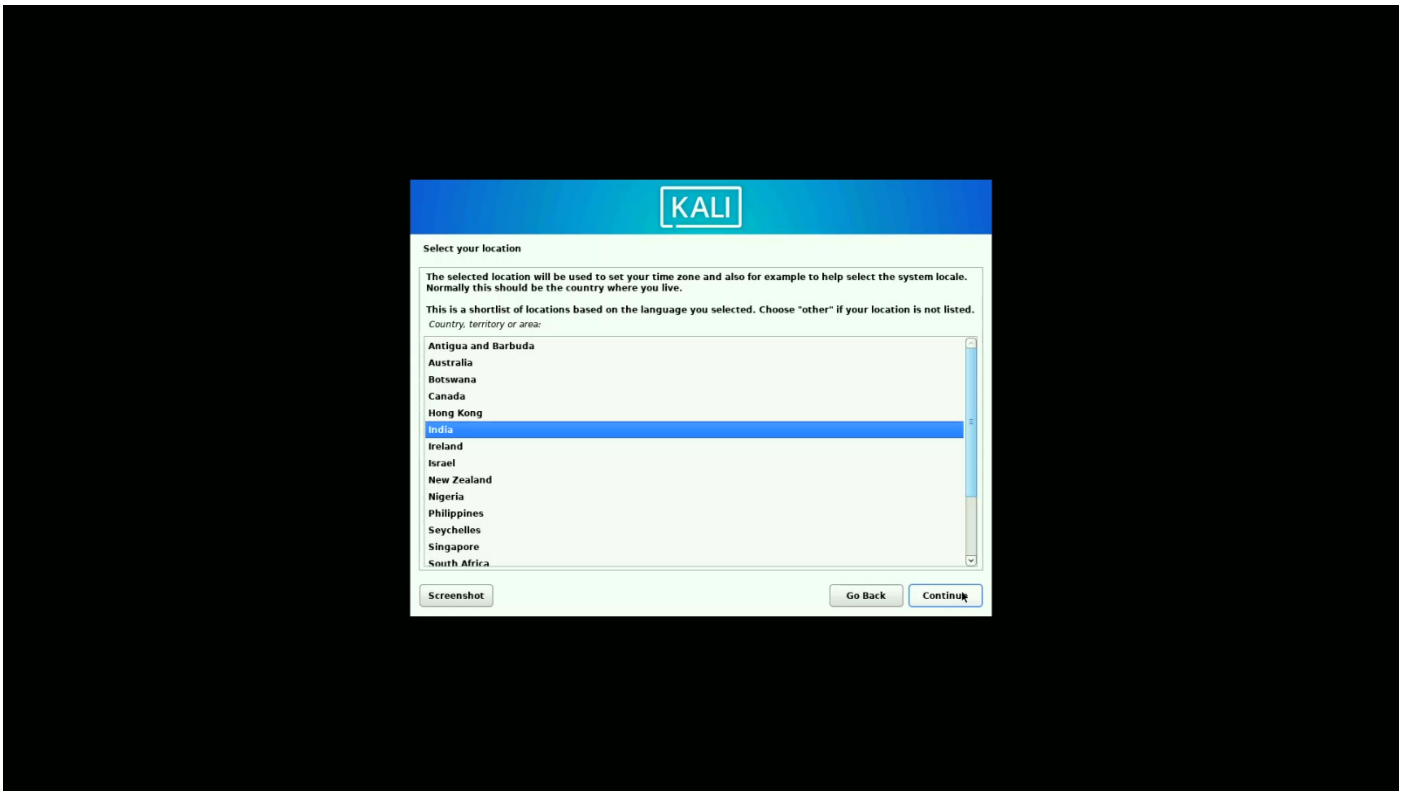
Step 1: Select “Graphical Install” and press Enter.



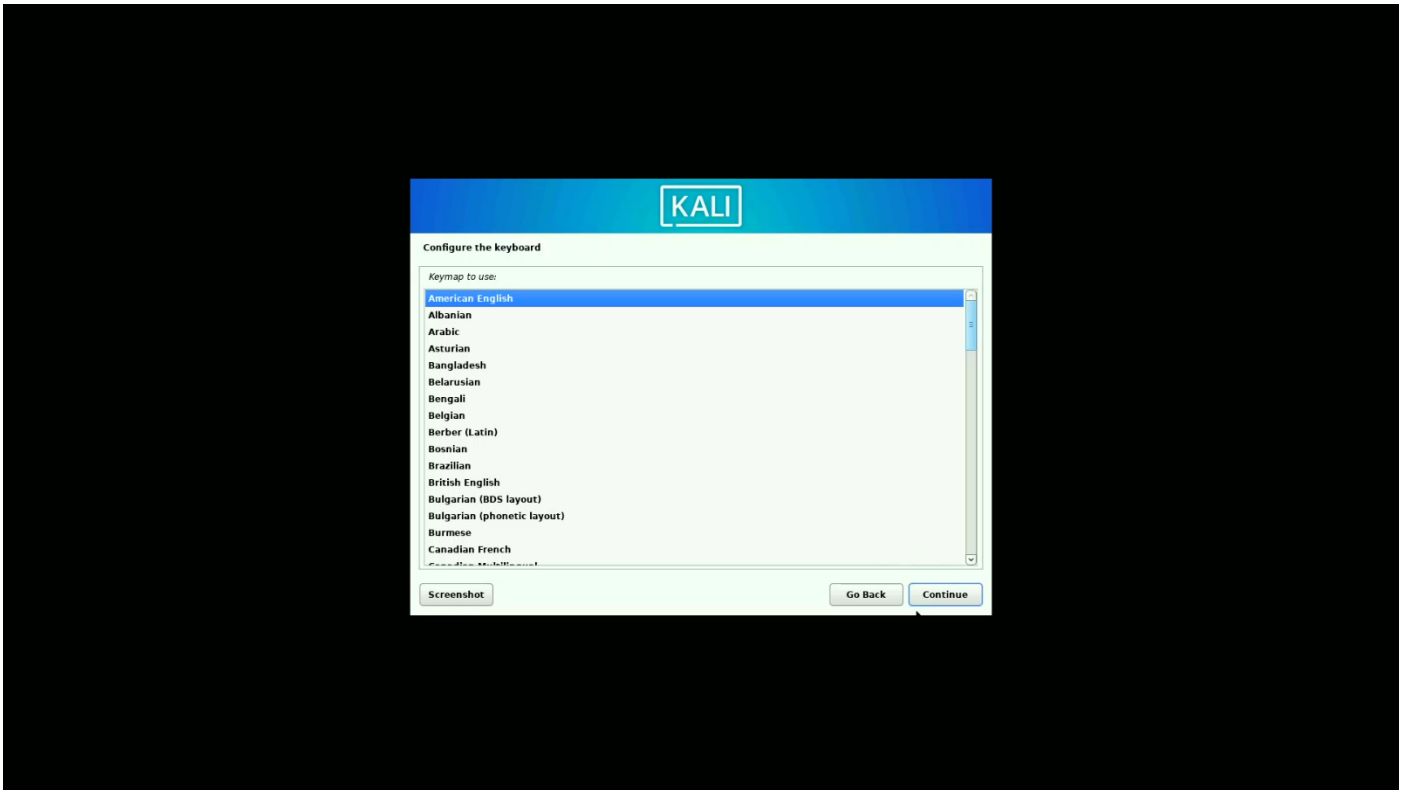
Step 2: Select language and click “Continue”.



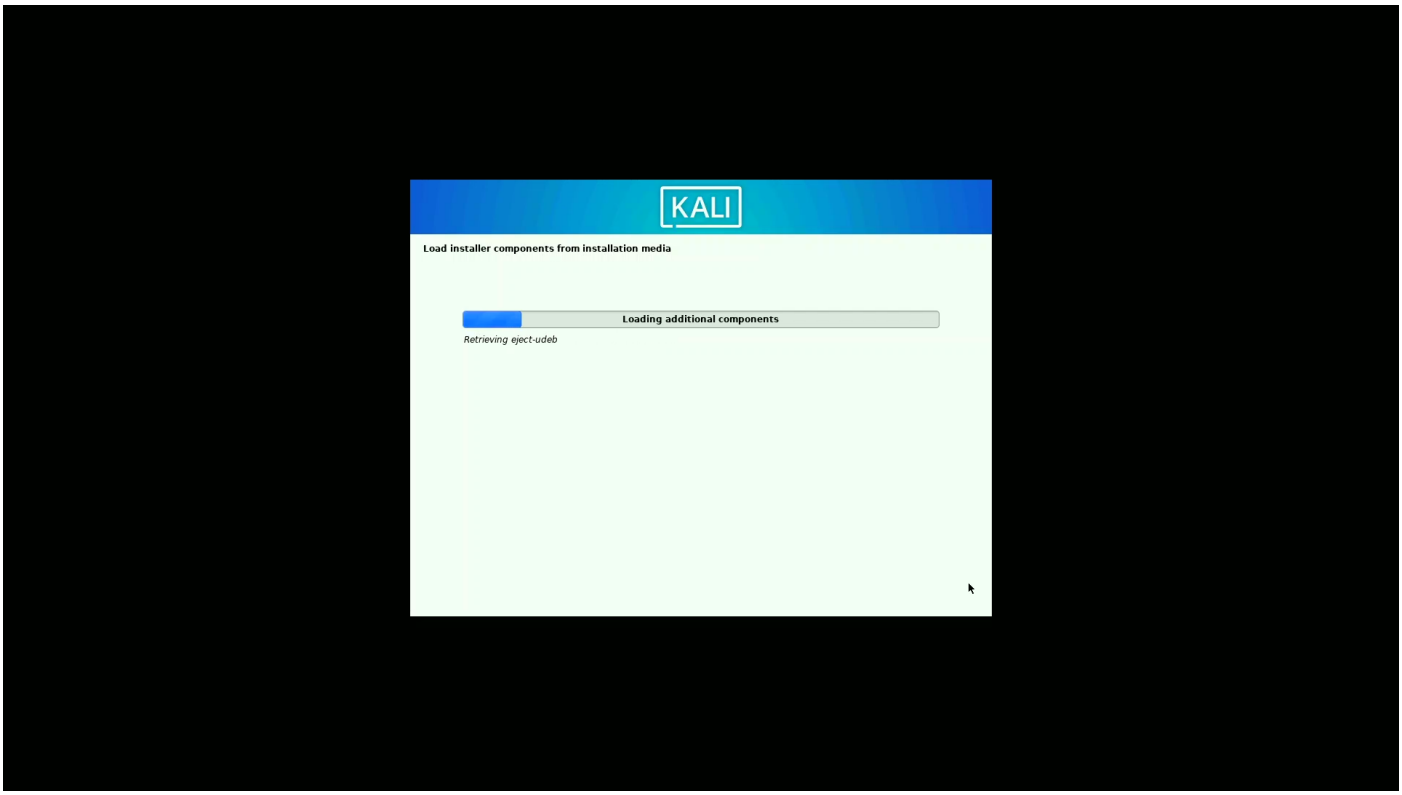
Step 3: Select region and click “Continue”.



Step 4: Select keyboard layout click “Continue”.



Step 5: Wait for the components to get loaded.

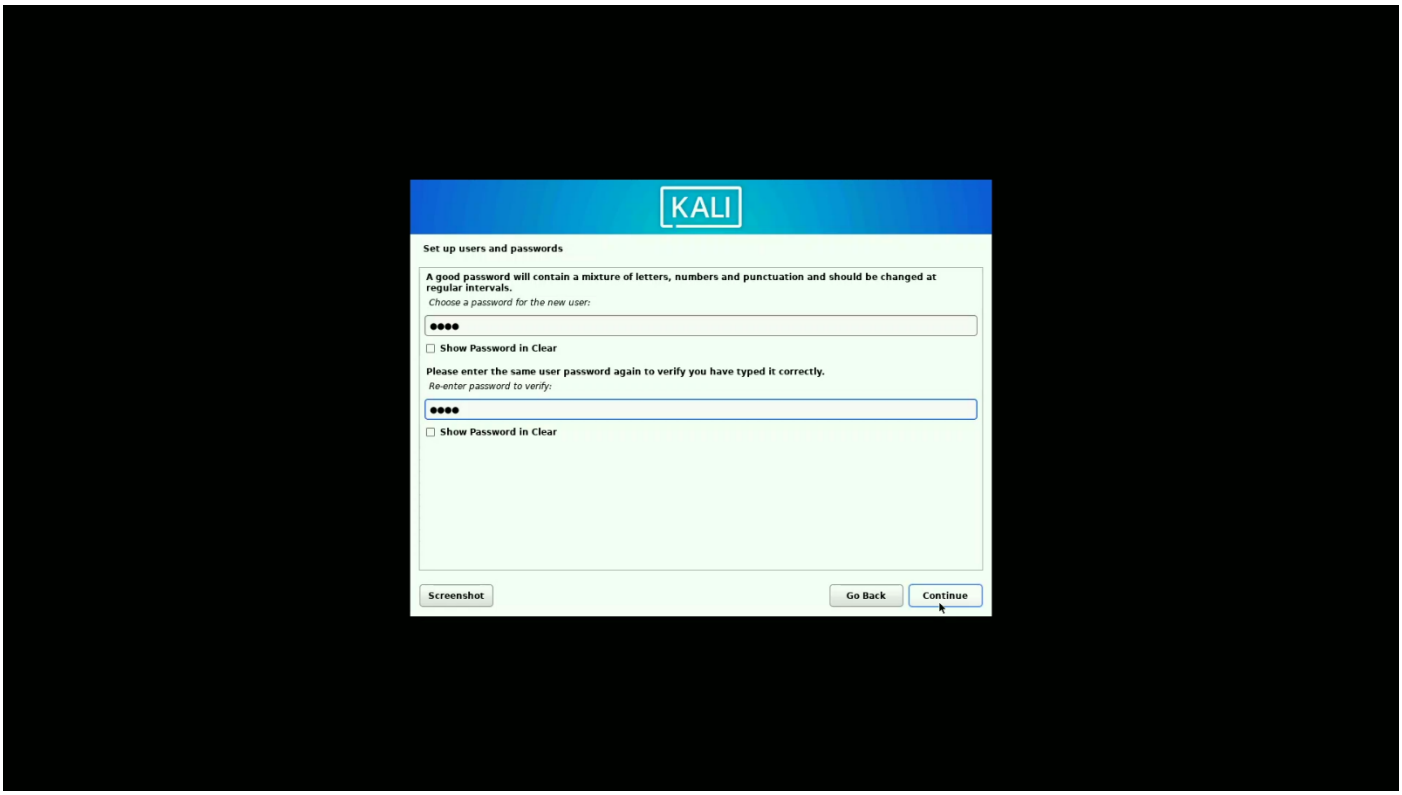


Step 6: Enter device name click “Continue”.

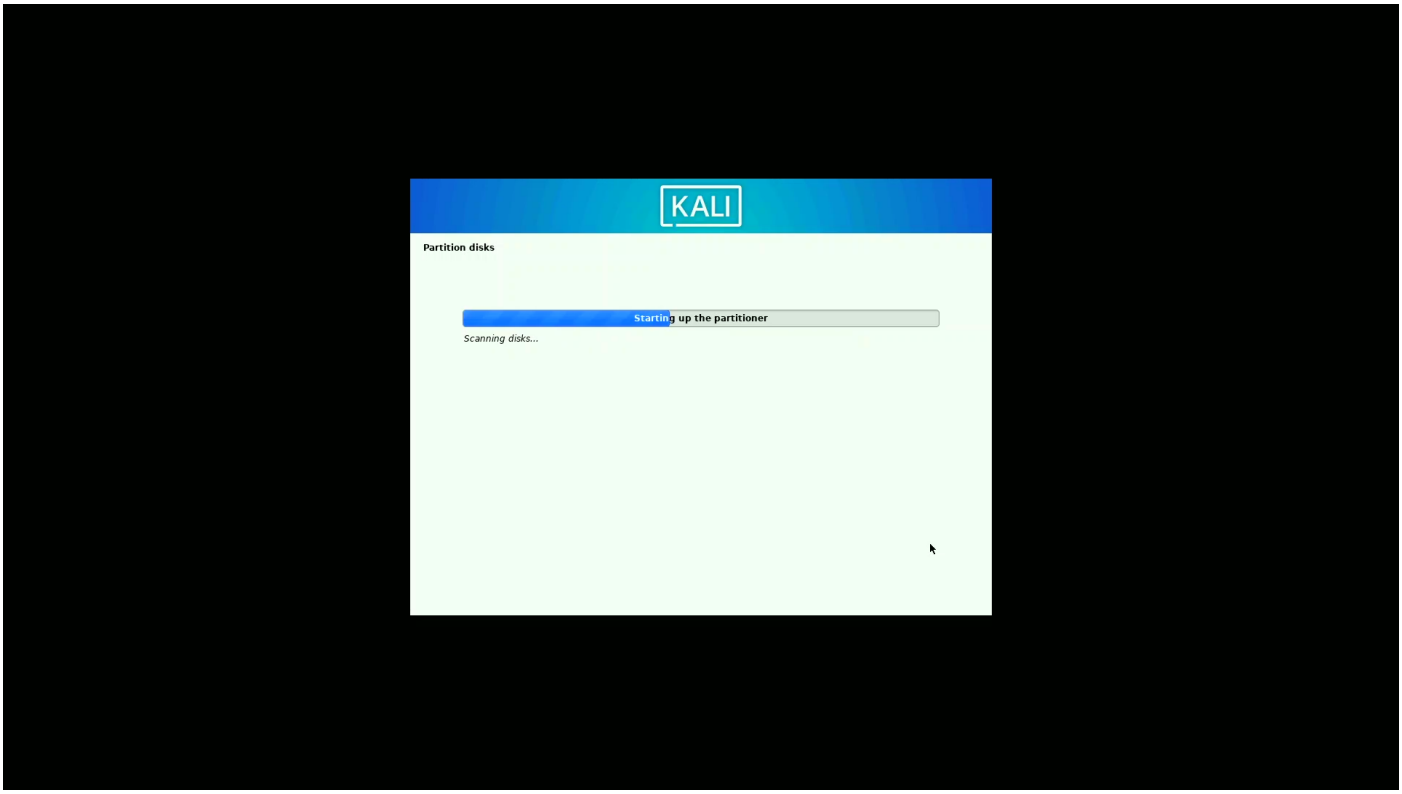
Step 7: Enter domain name click “Continue”.

Step 8: Enter your full name click “Continue”.

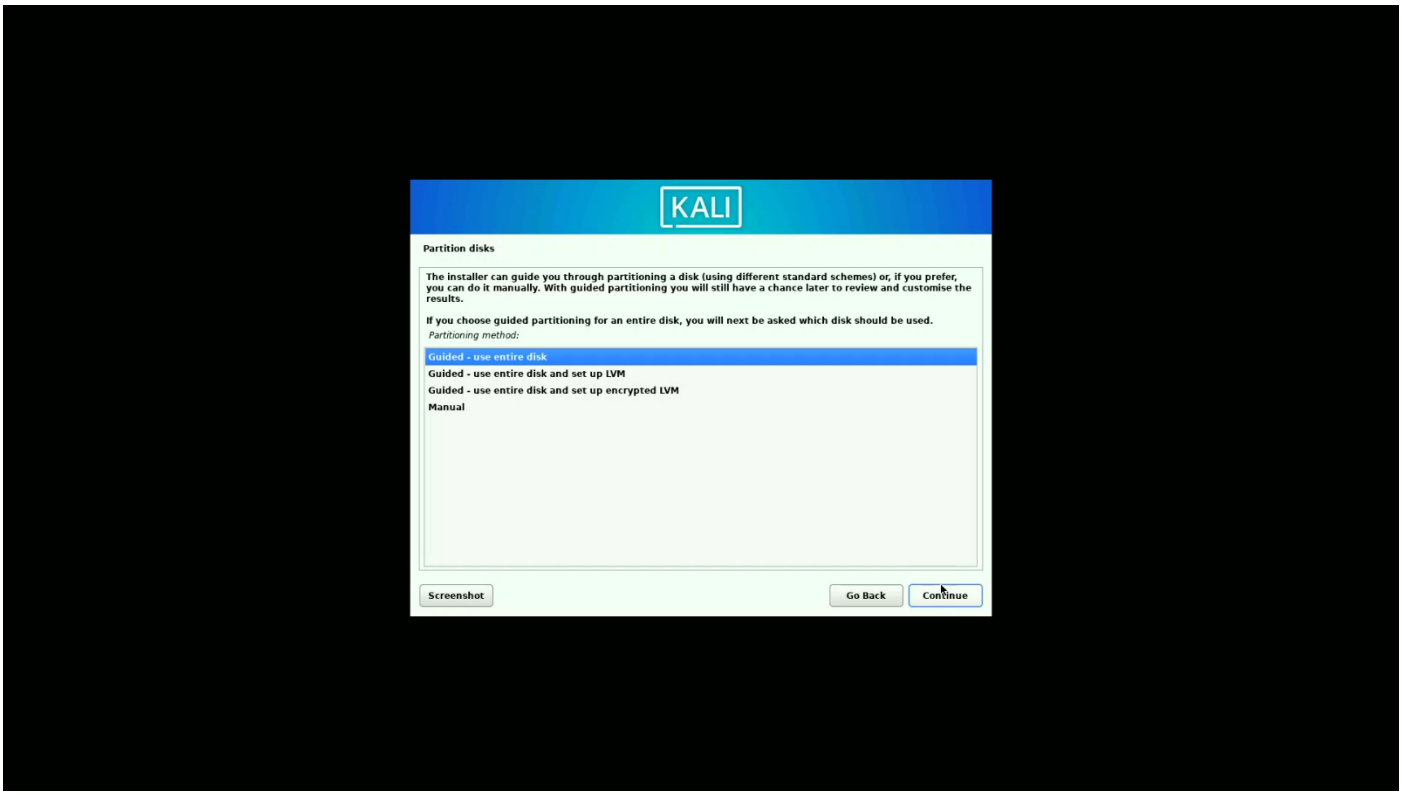
Step 9: Enter password click “Continue”.



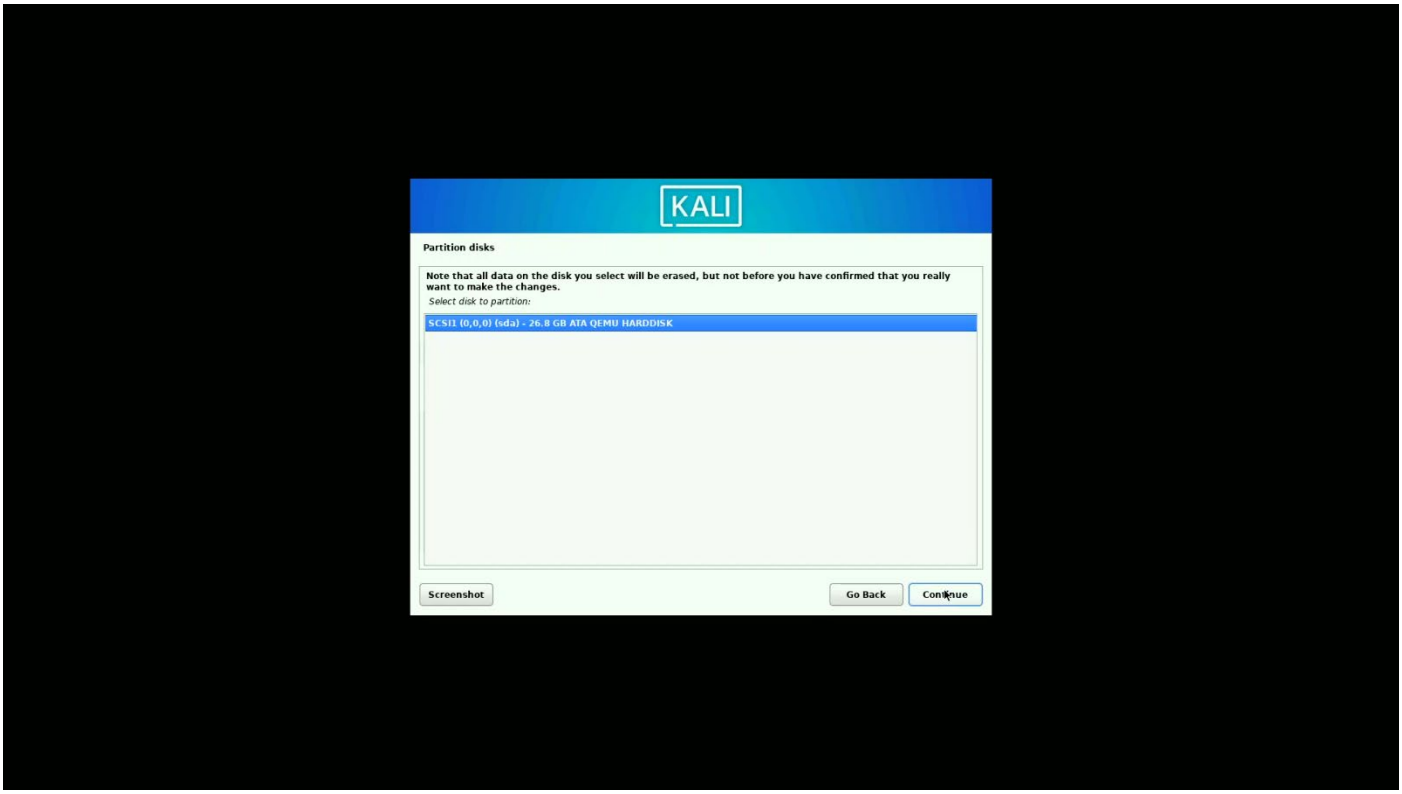
Step 10: Wait for the disk manager to get loaded.



Step 11: Select “Guided – use entire disk” and click “Continue”.

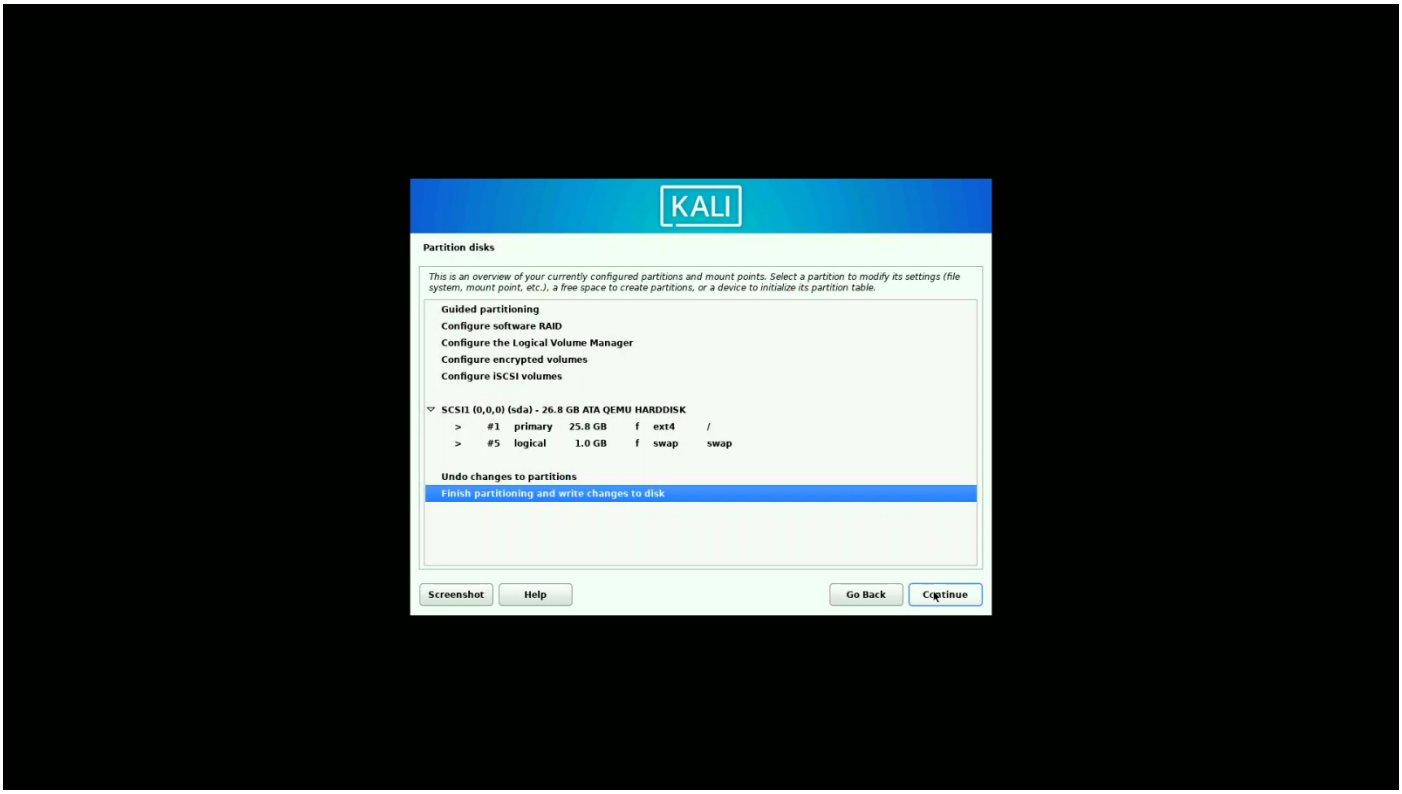


Step 12: Select Disk click “Continue”.



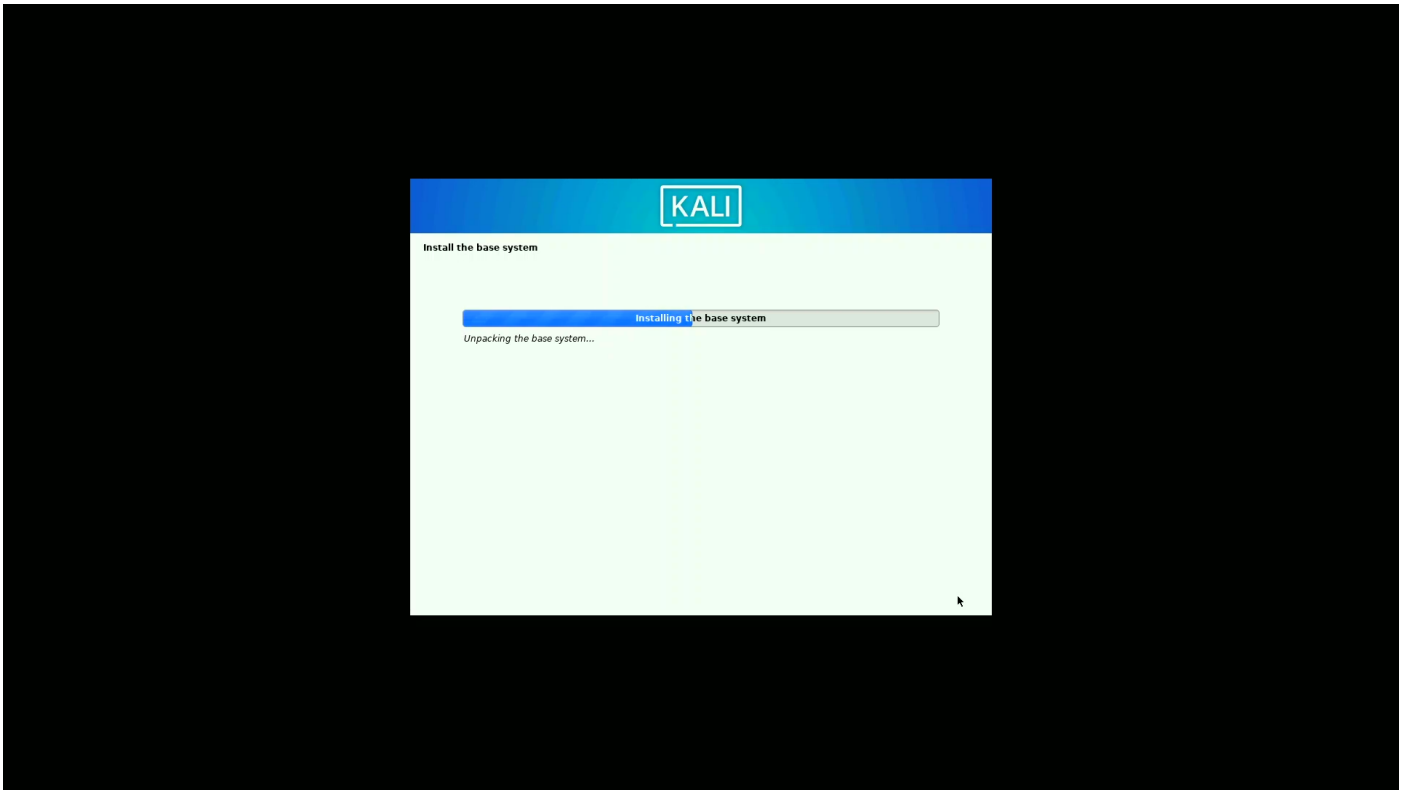
Step 13: Select “All files in one partition” and click “Continue”.

Step 14: Click “Continue”.

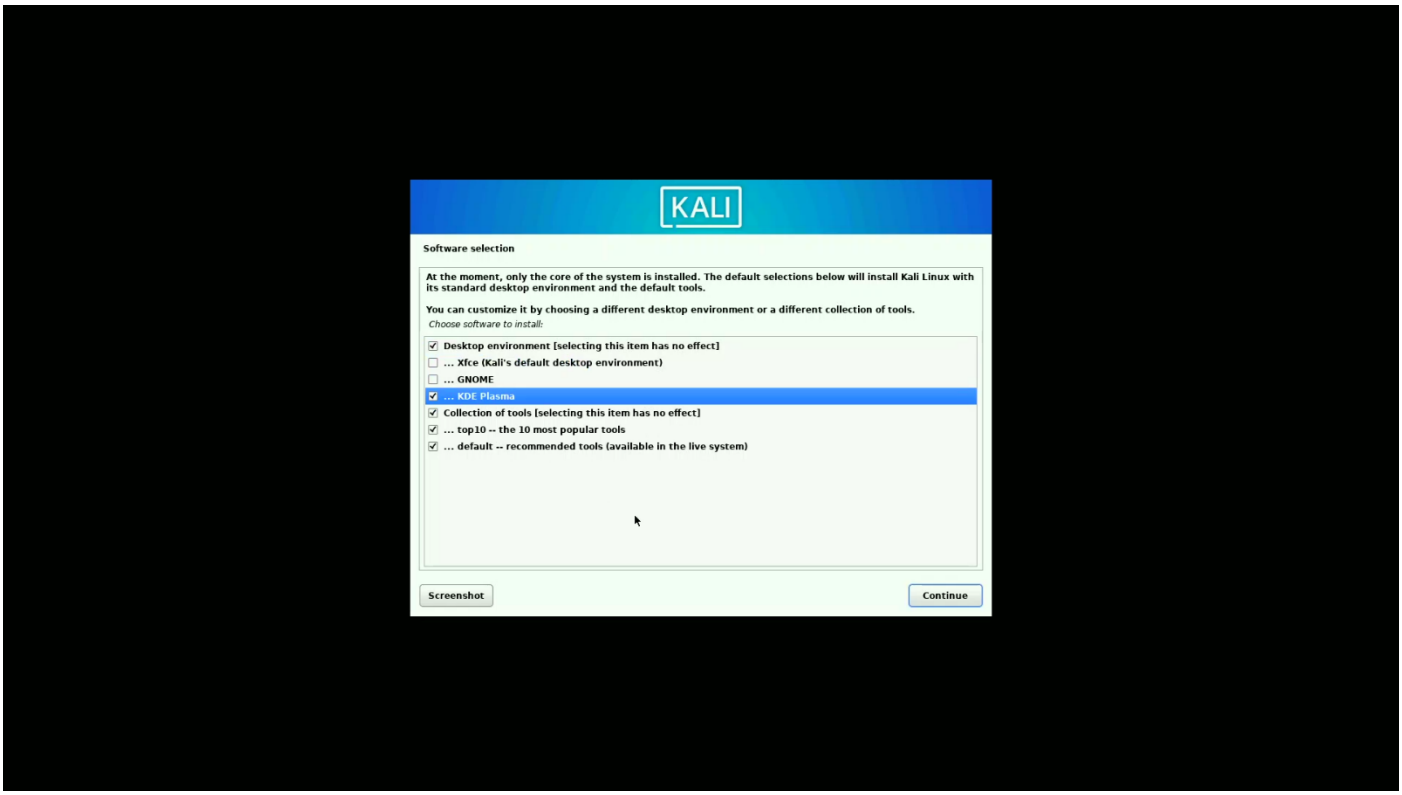


Step 15: Select “Yes” and click “Continue”.

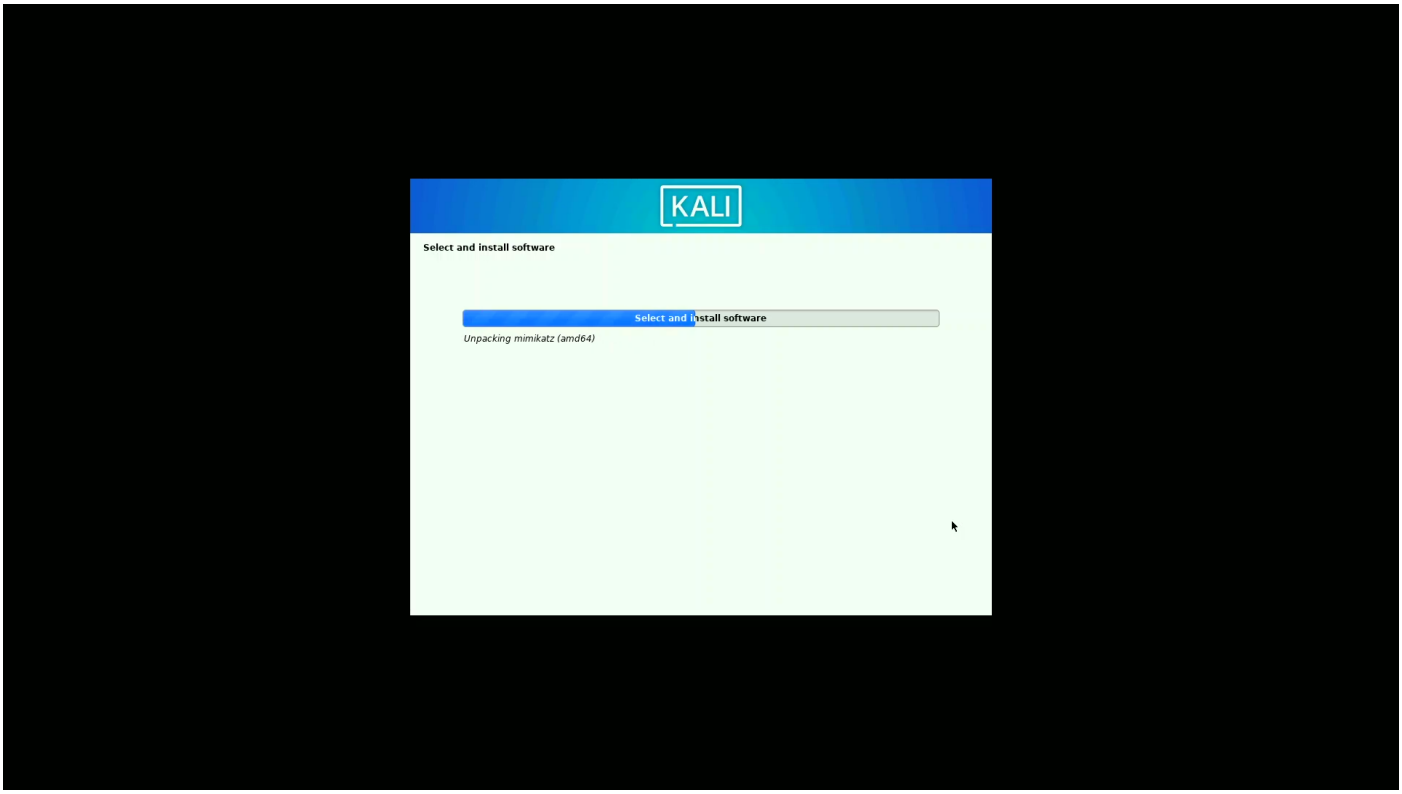
Step 16: Wait for the Base OS to get installed.



Step 17: Select the software's, Desktop Environment and click "Continue".

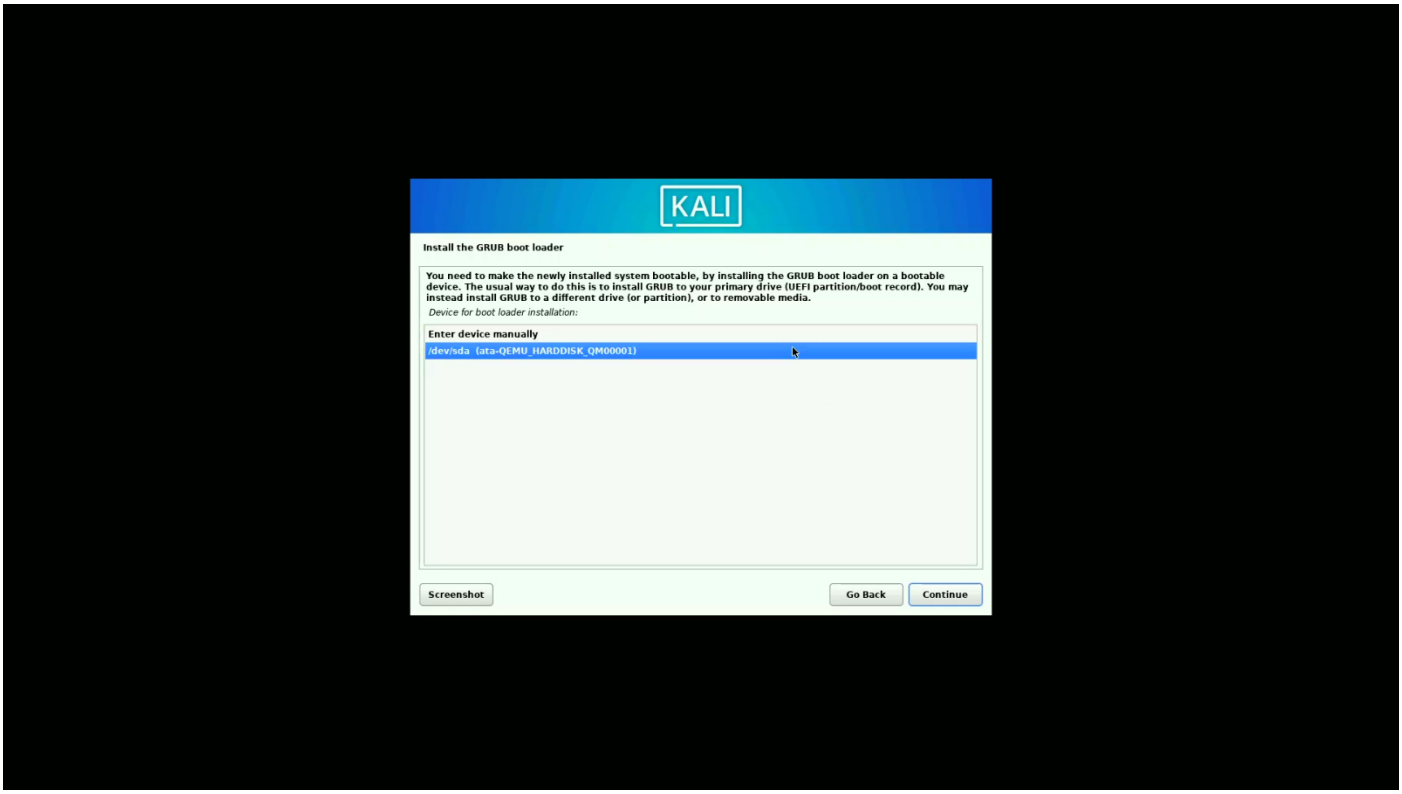


Step 18: Wait for the software's and DE to get installed.

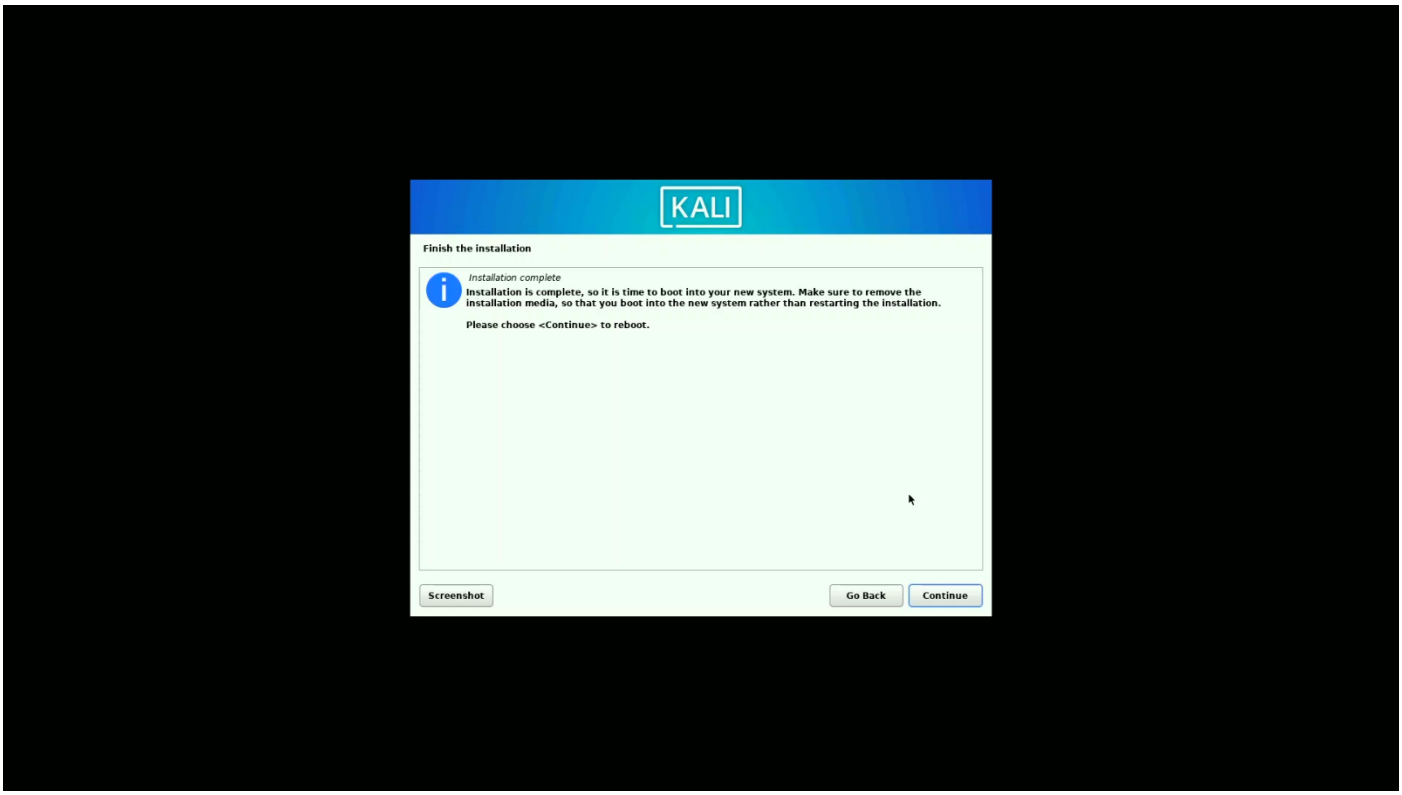


Step 19: Select “Yes” and click “Continue”.

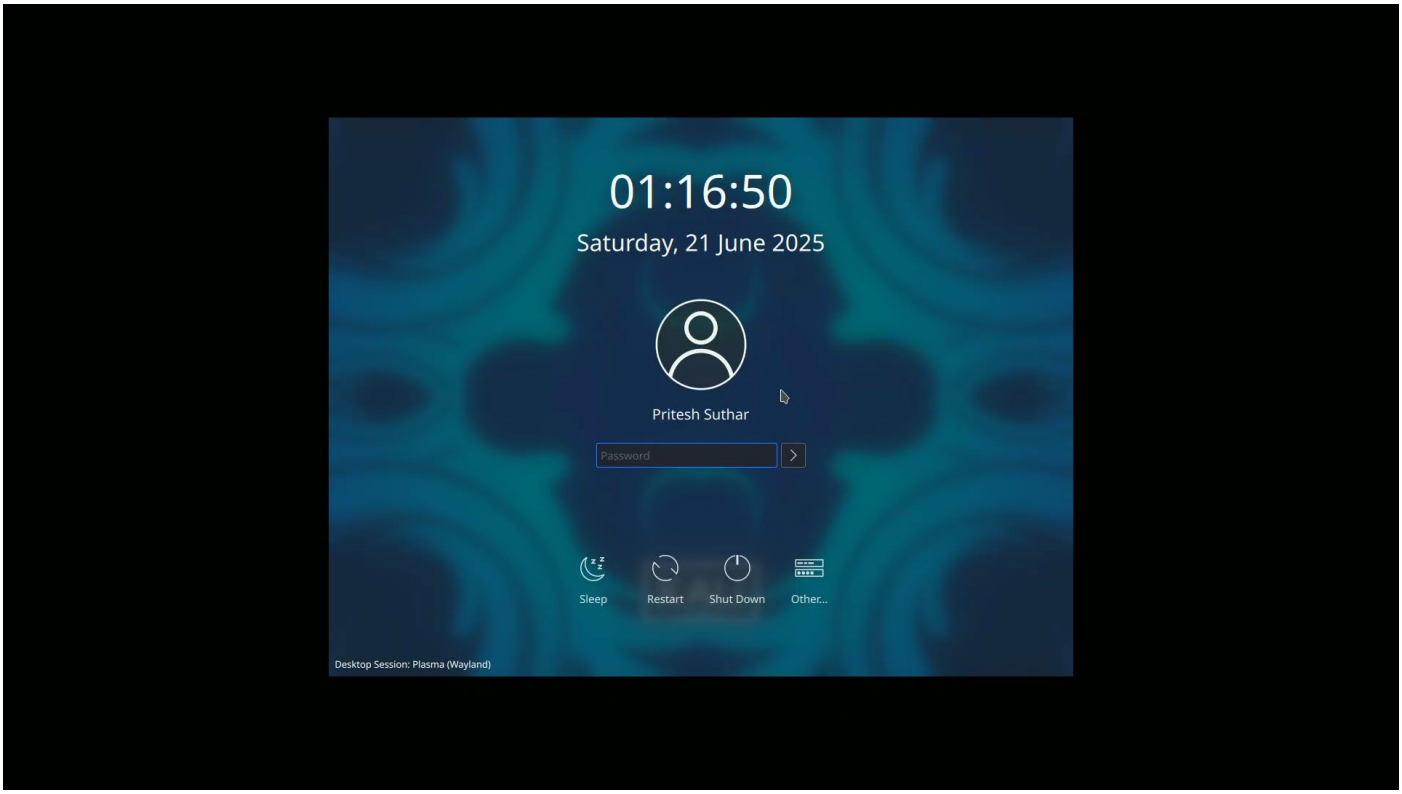
Step 20: Select disk and click “Continue”.



Step 21: Click “Continue”.



Step 22: Enter your password.

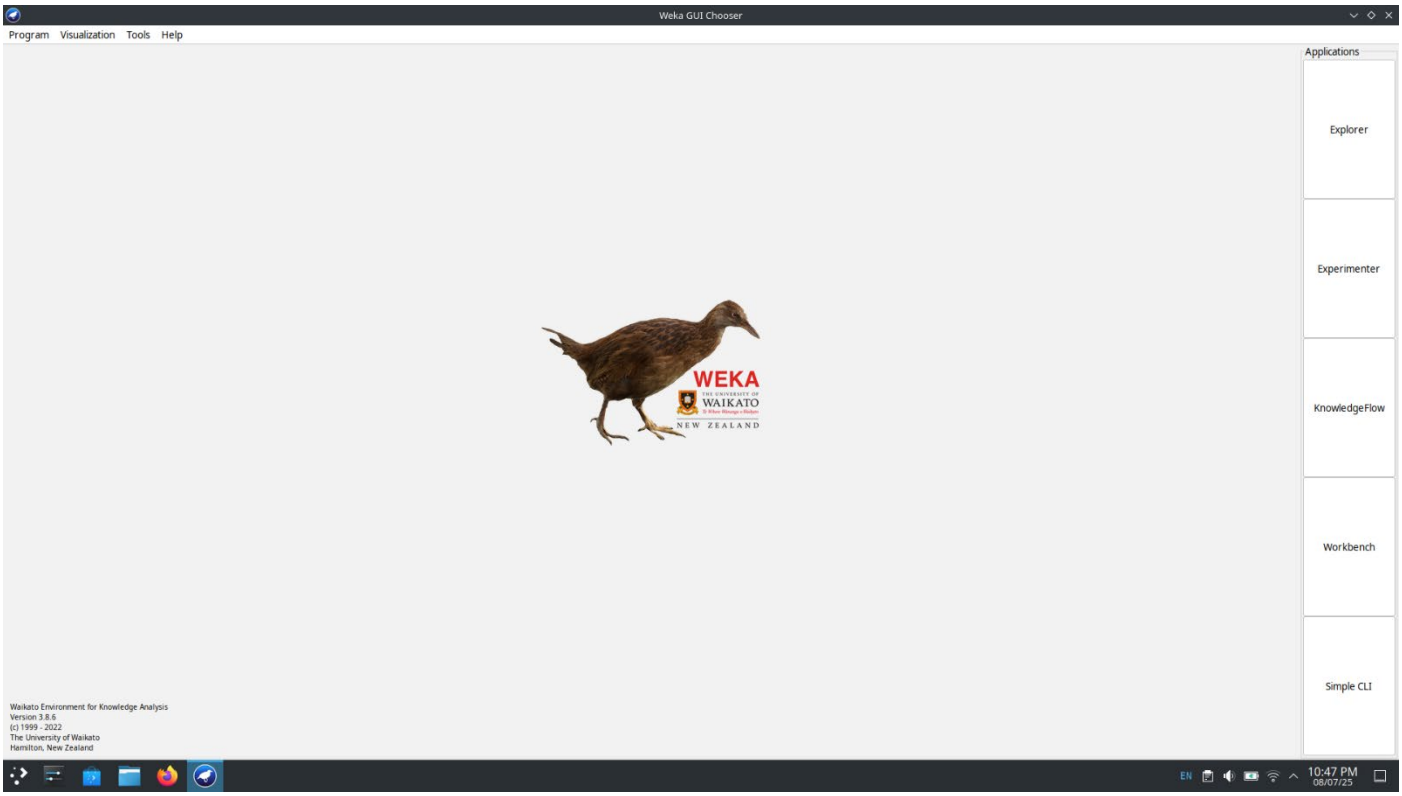


Step 23: Kali Linux is successfully installed.

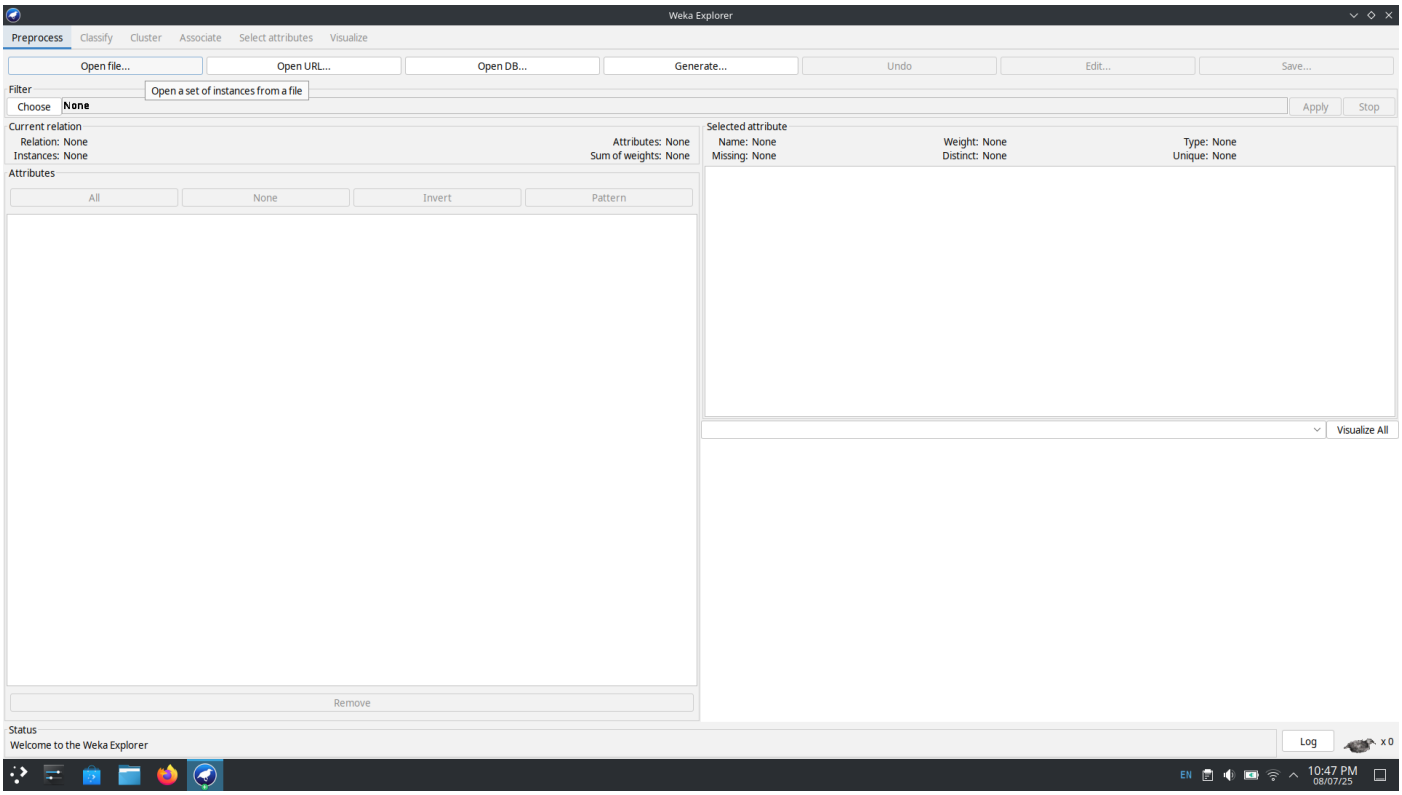
Practical – 2

Aim: Explore WEKA Tool.

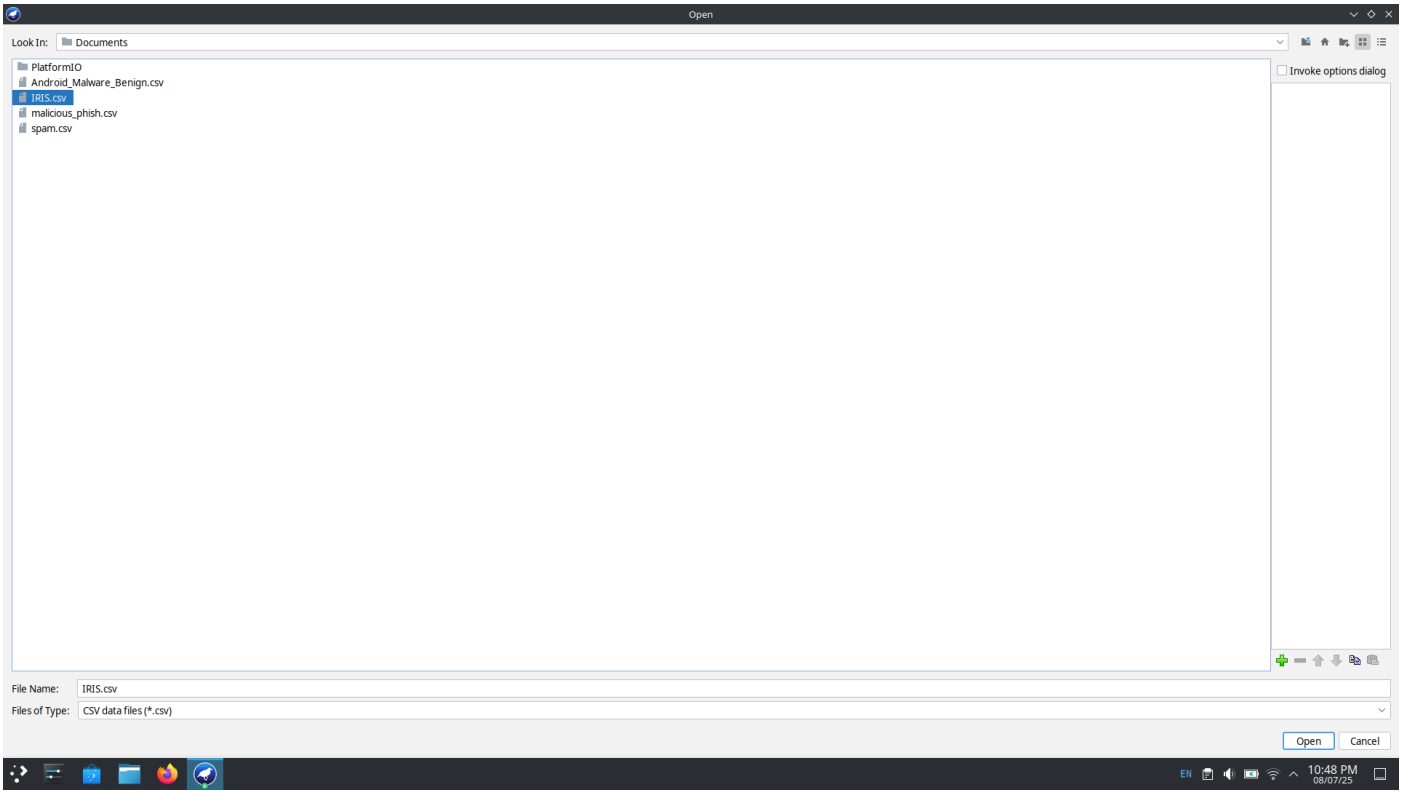
Step 1: Click on “Explorer”.



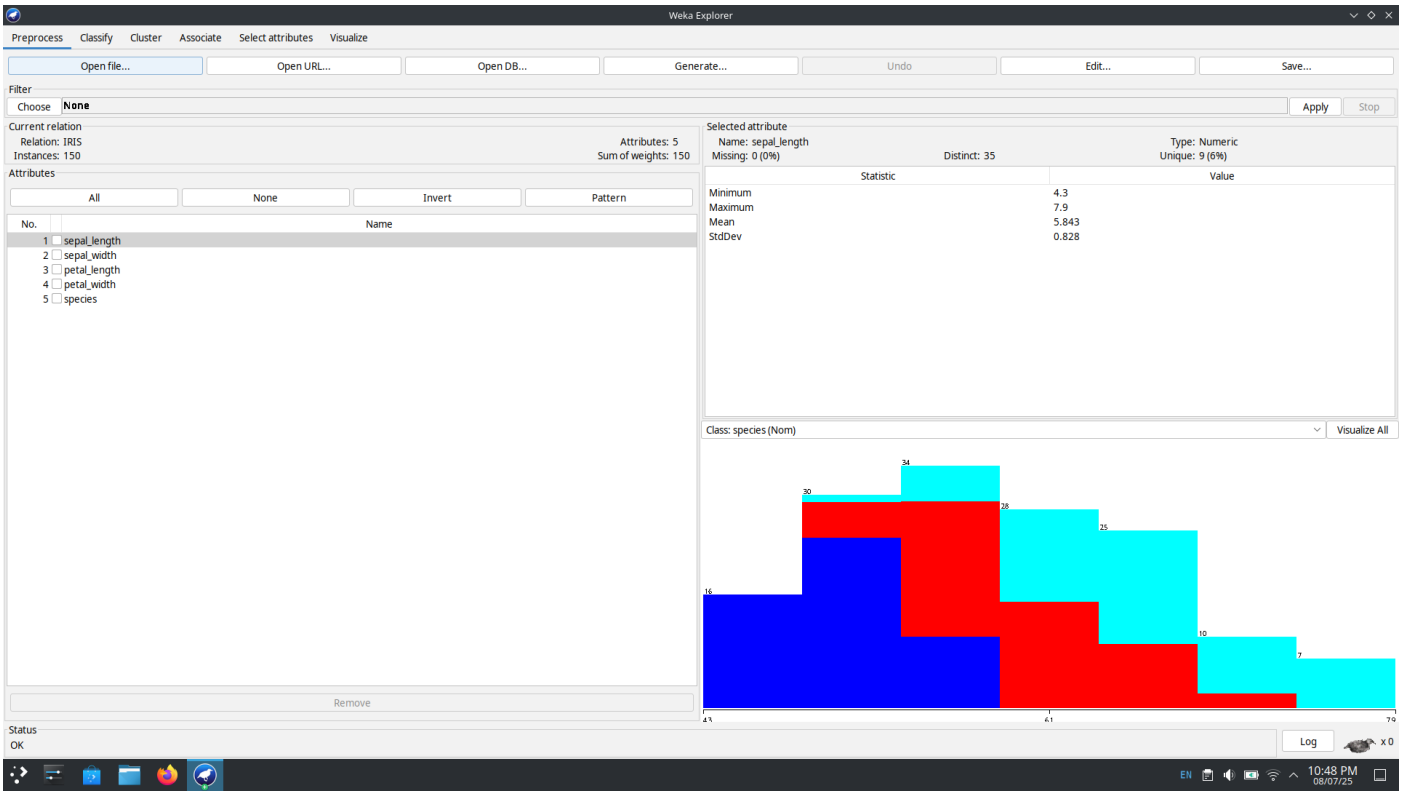
Step 2: Click on “Open File”.



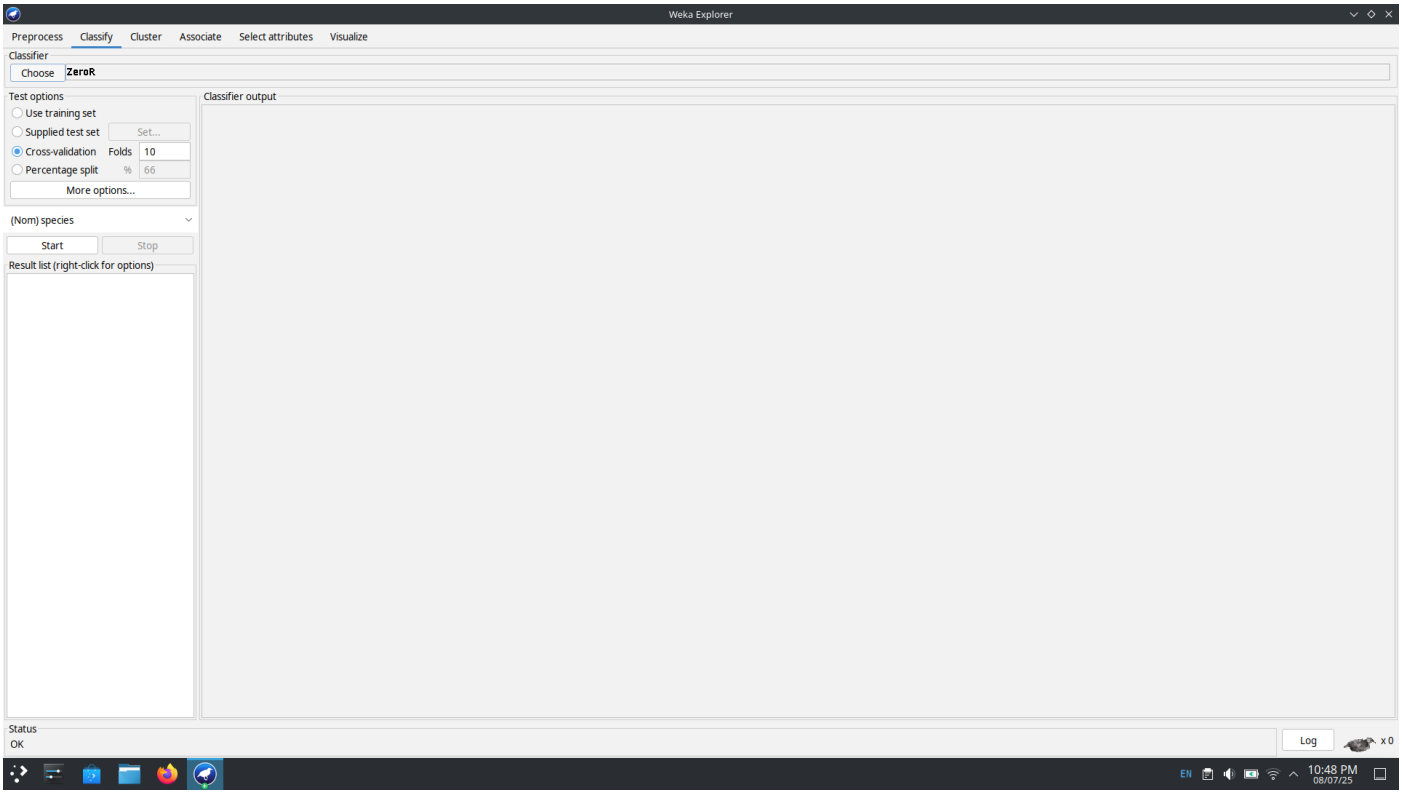
Step 3: Select and open “IRIS.csv”.



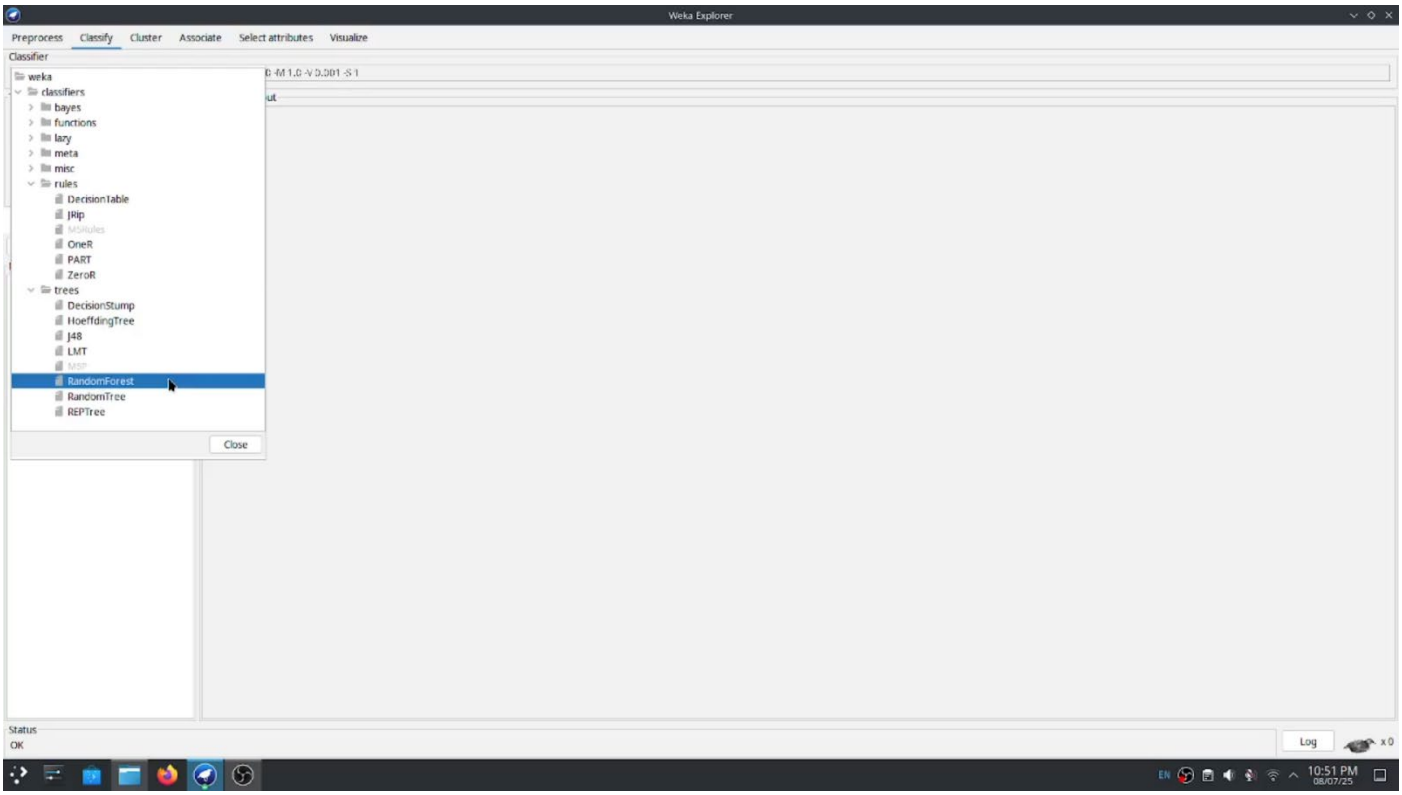
Step 4: Click on “Classify”.



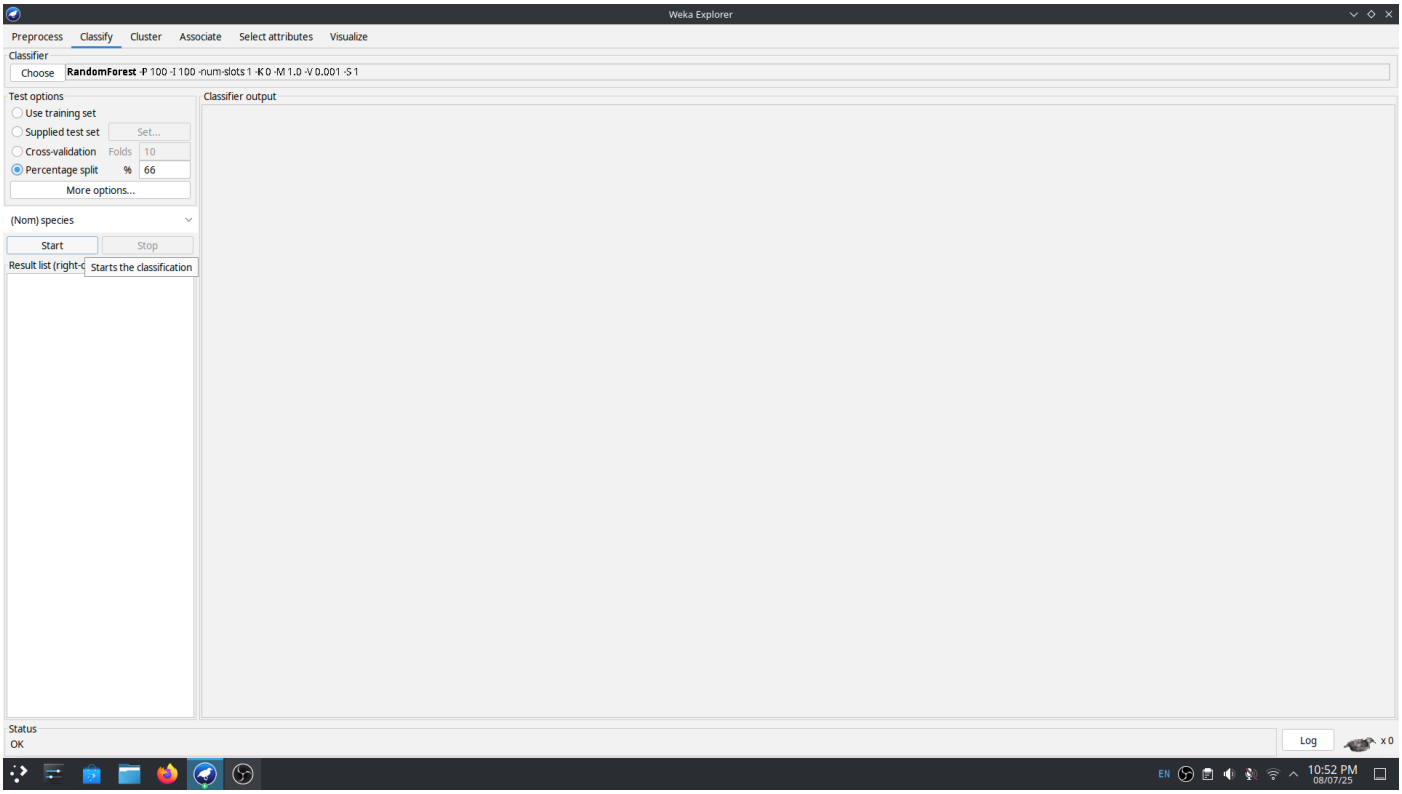
Step 5: Click on “Choose”.



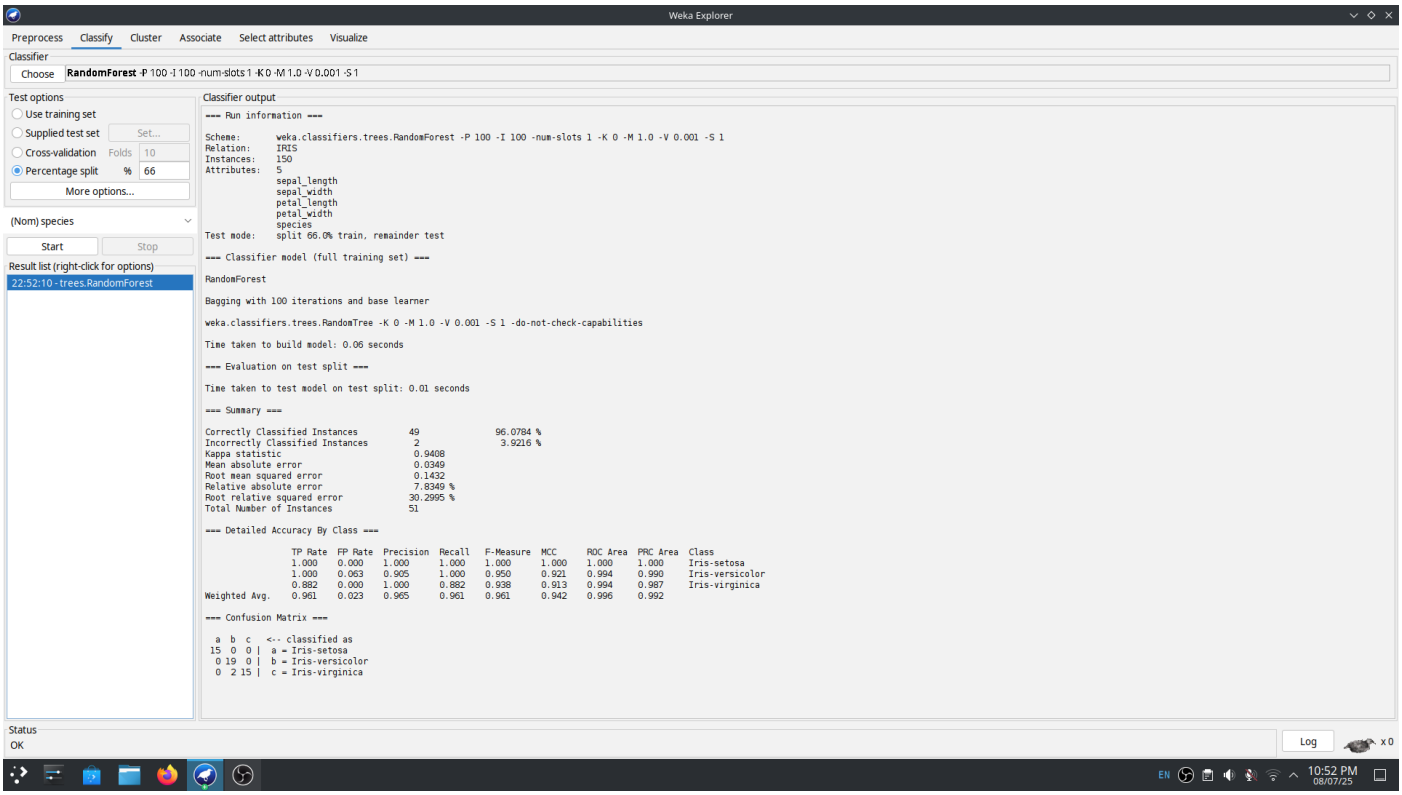
Step 6: Select “RandomForest” available under “trees”.



Step 7: Select “Percentage Split”.



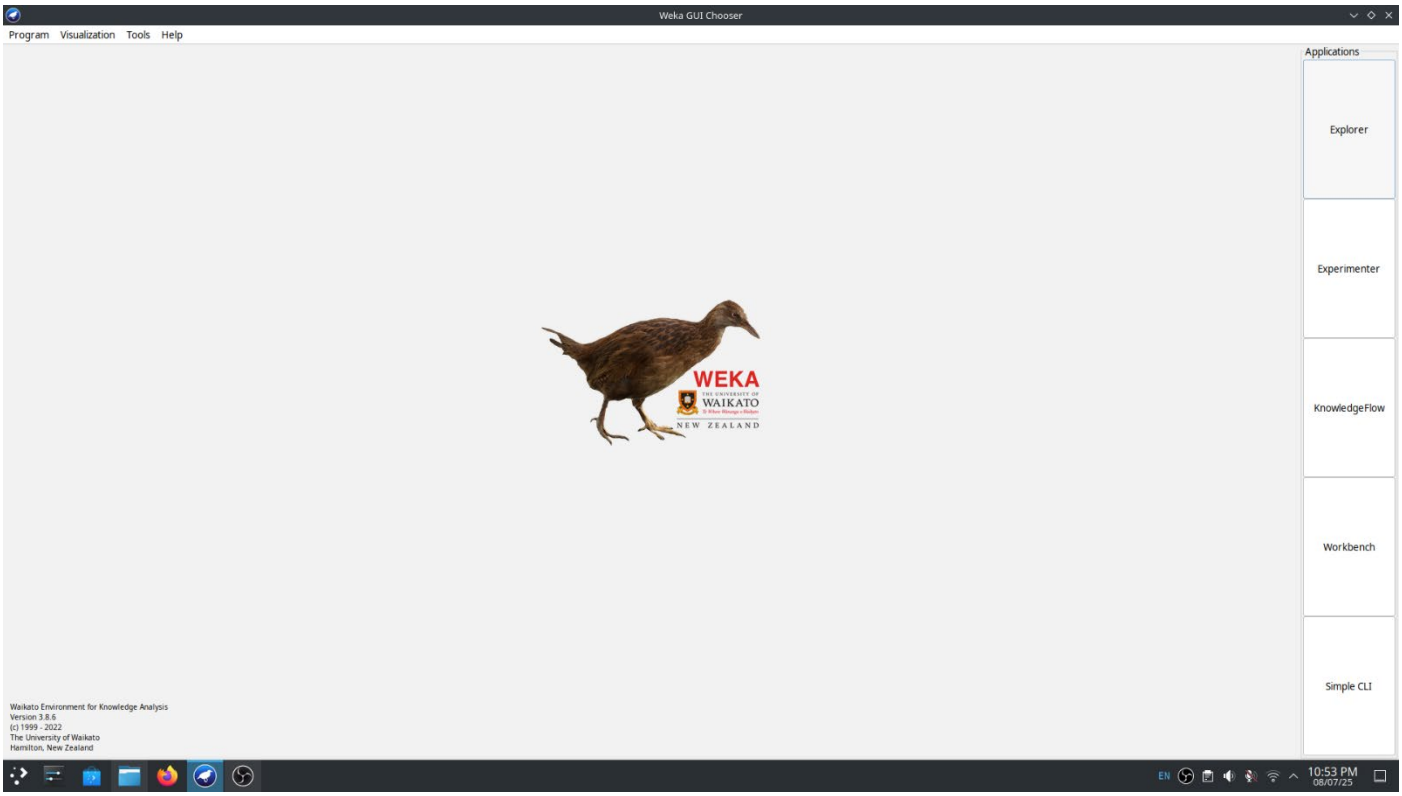
Step 8: Press on “Start” to train the model and review the output.



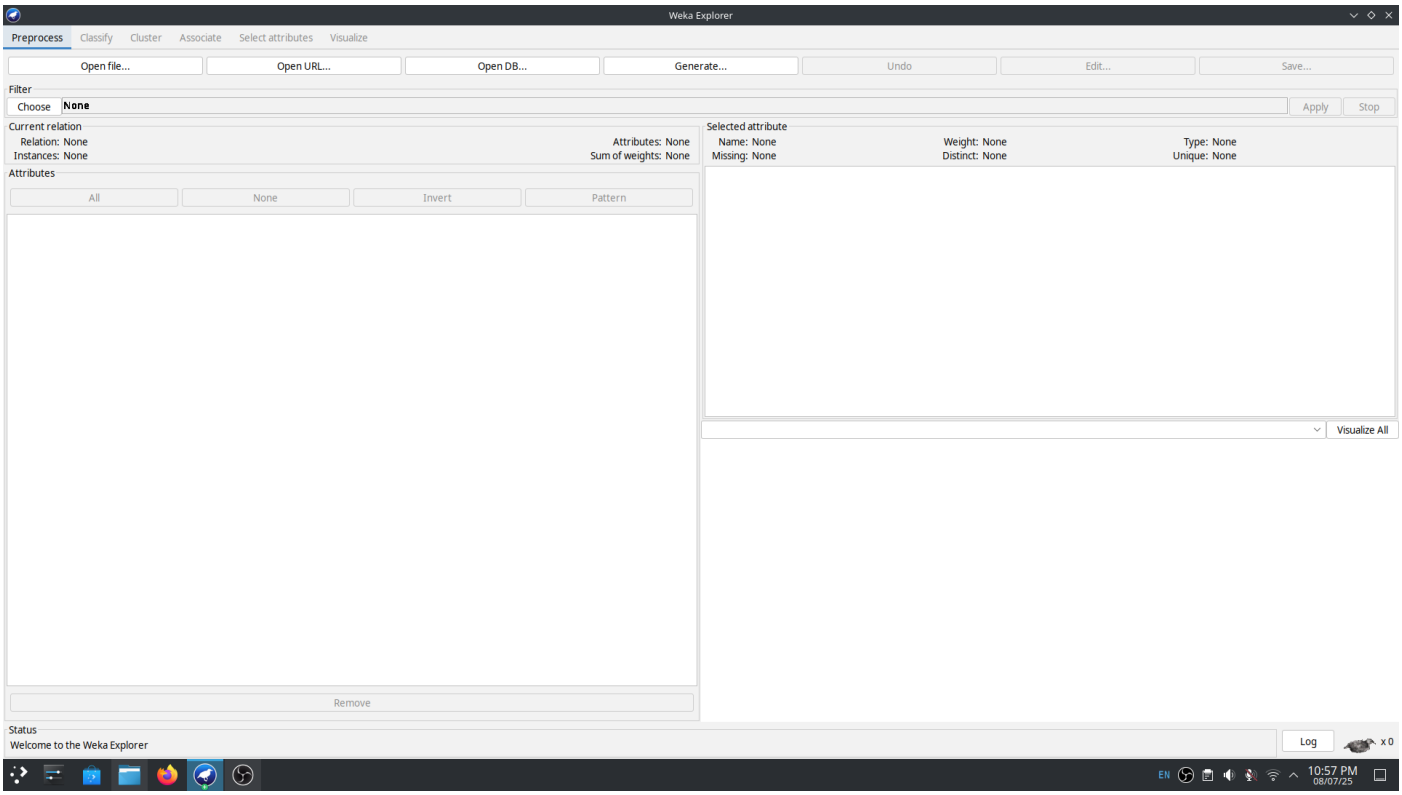
Practical – 3

Aim: Android malware detection using WEKA tool.

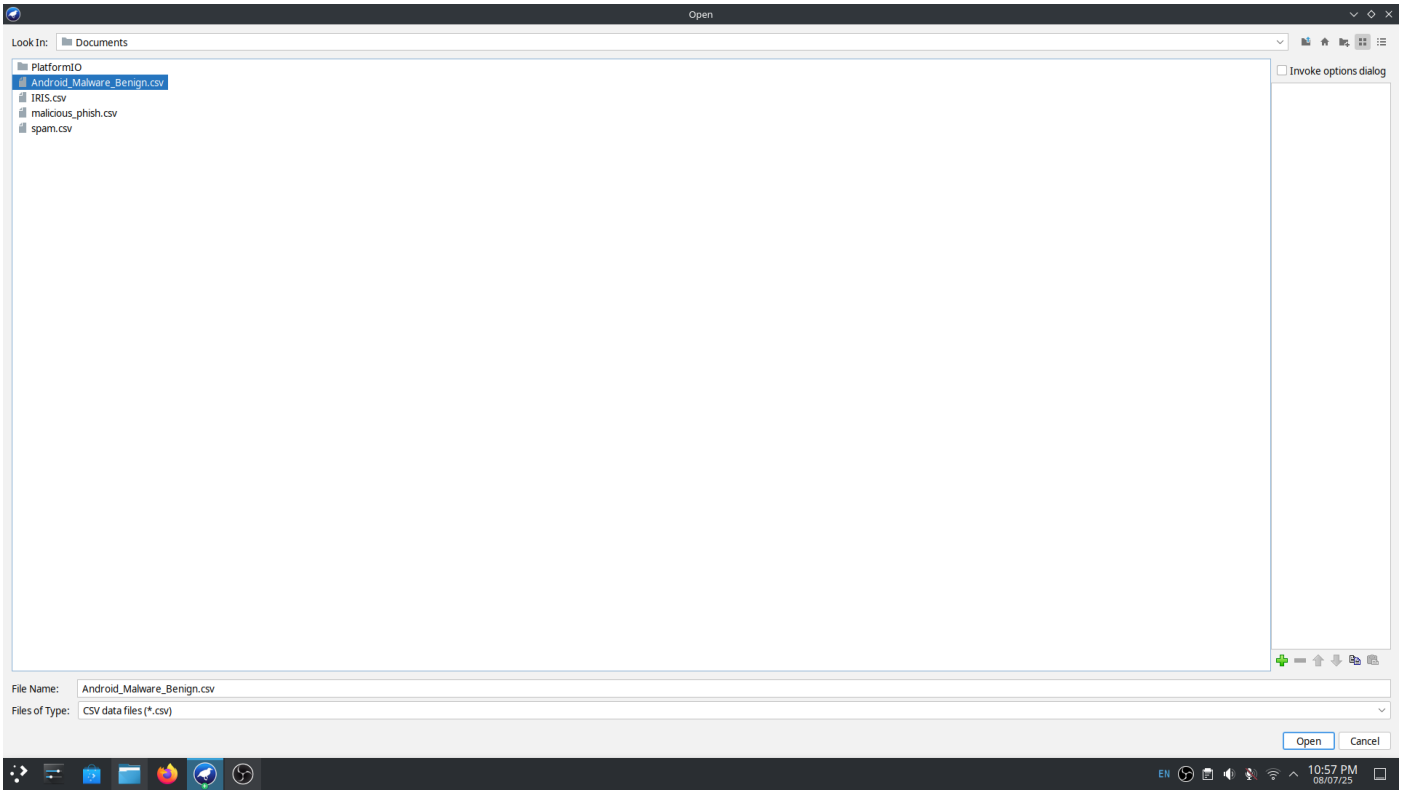
Step 1: Click on “Explorer”.



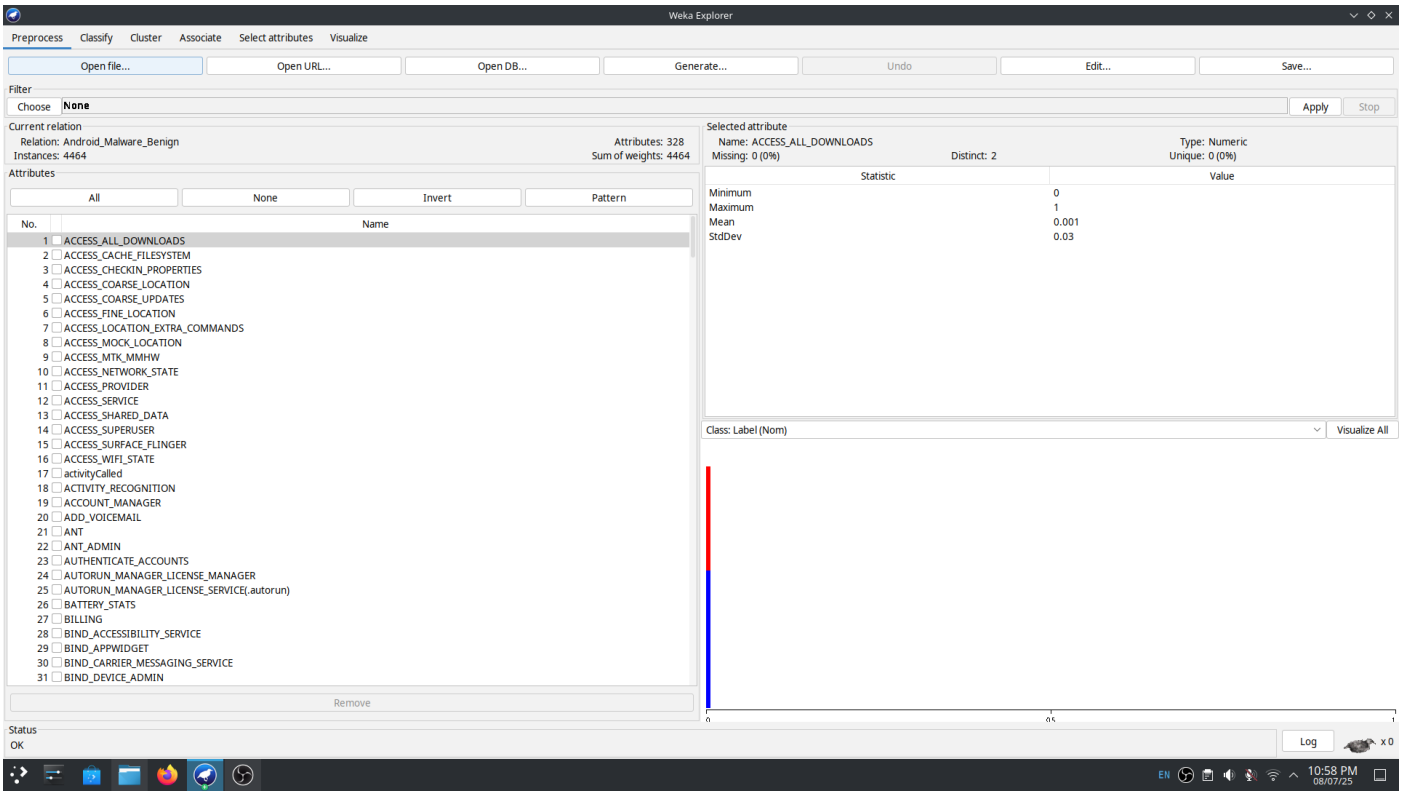
Step 2: Click on “Open File”.



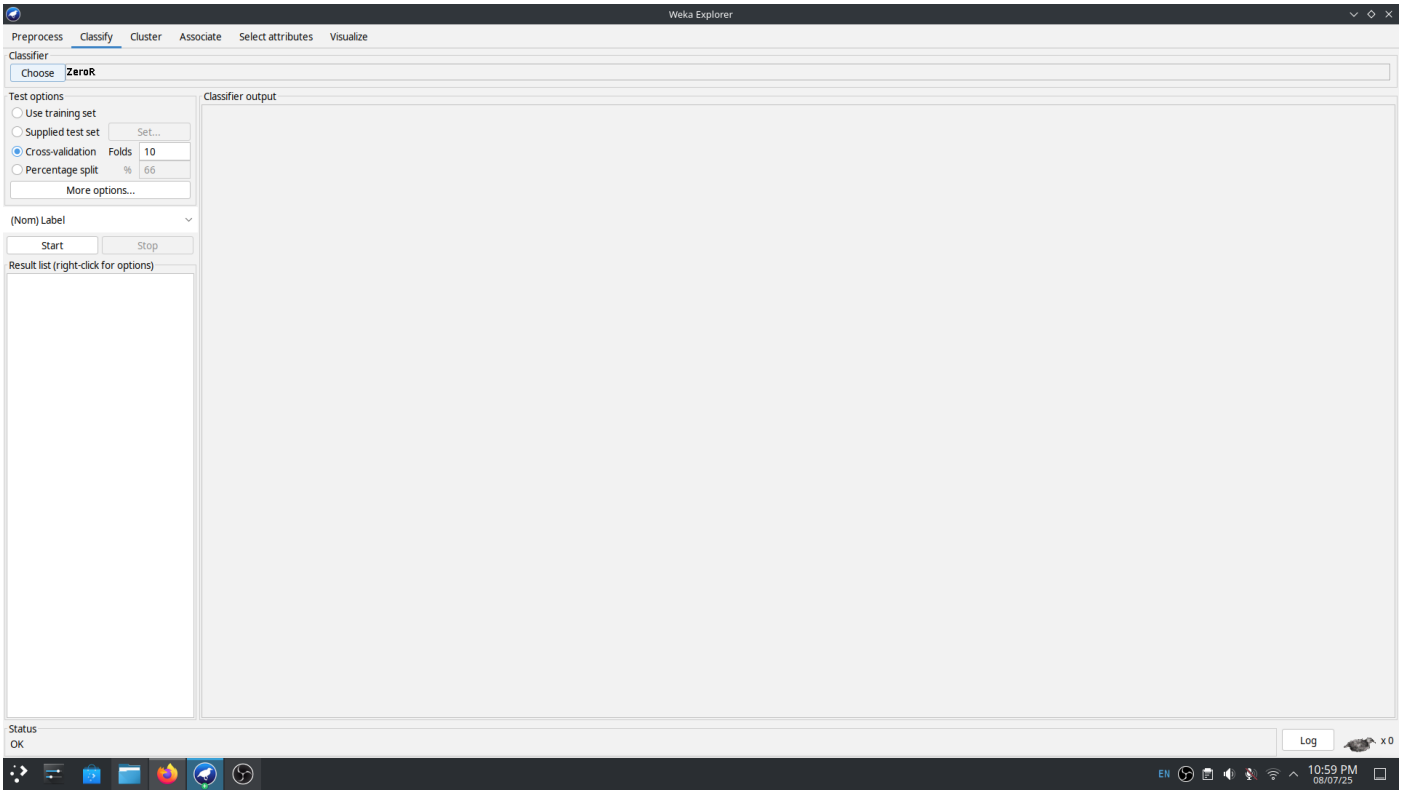
Step 3: Select and open “Android_Malware_Benign.csv”.



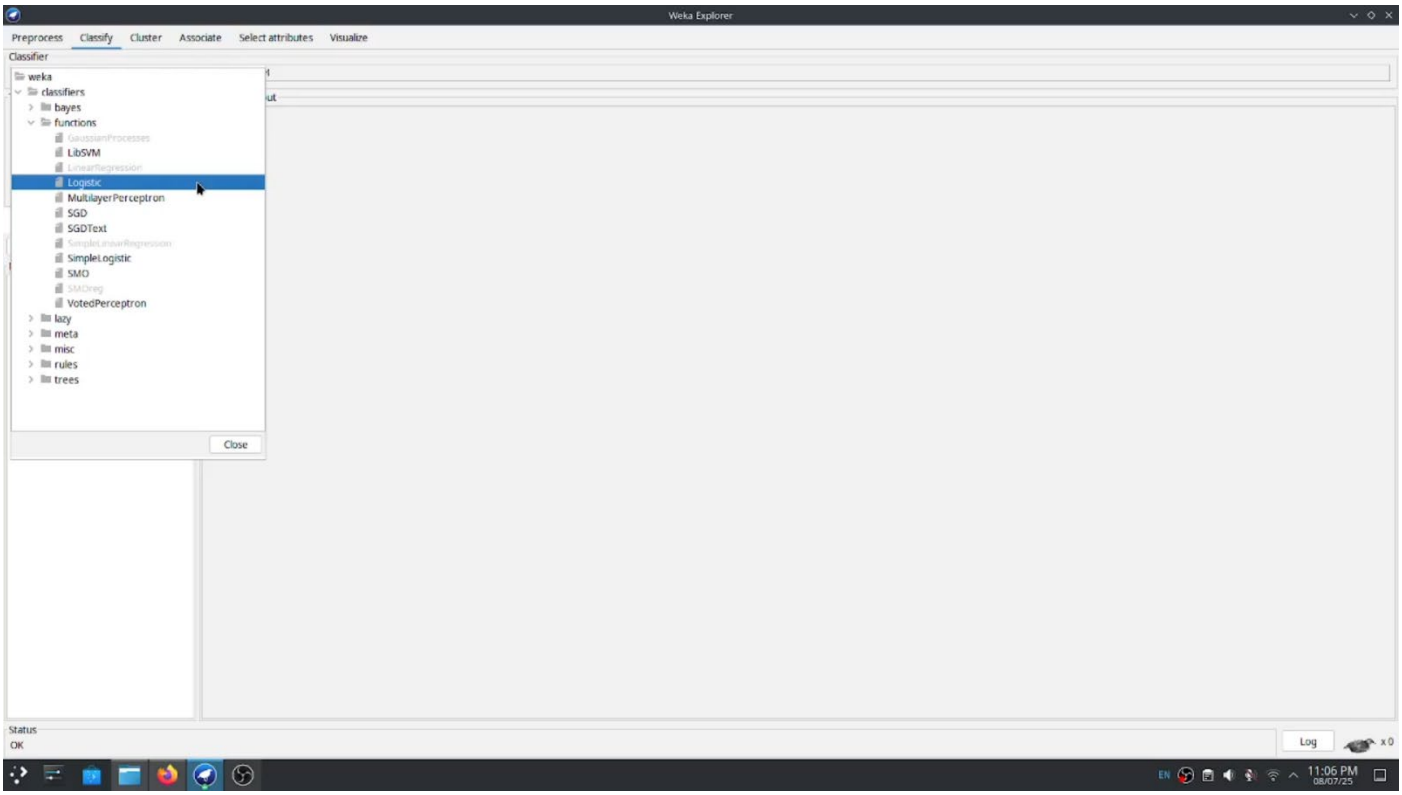
Step 4: Click on “Classify”.



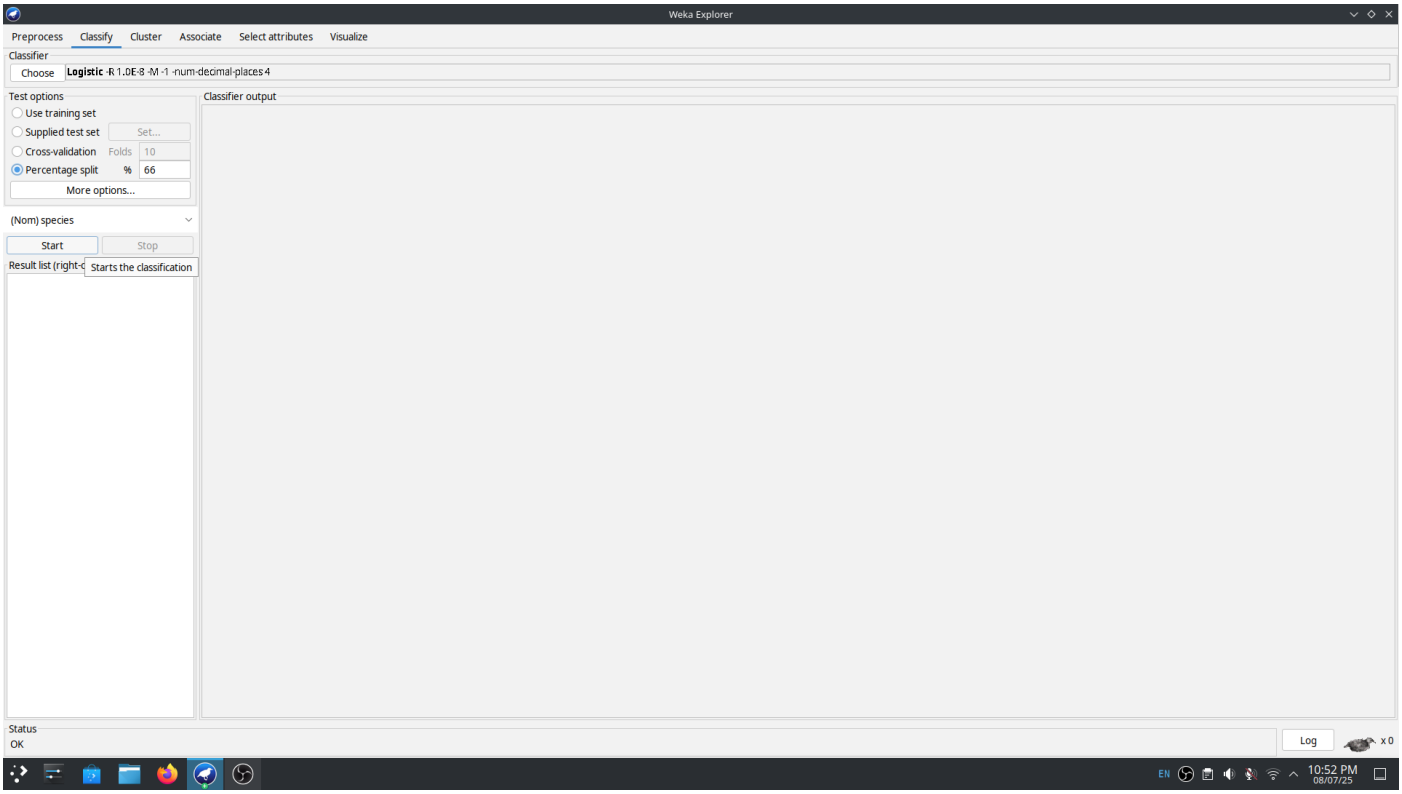
Step 5: Click on “Choose”.



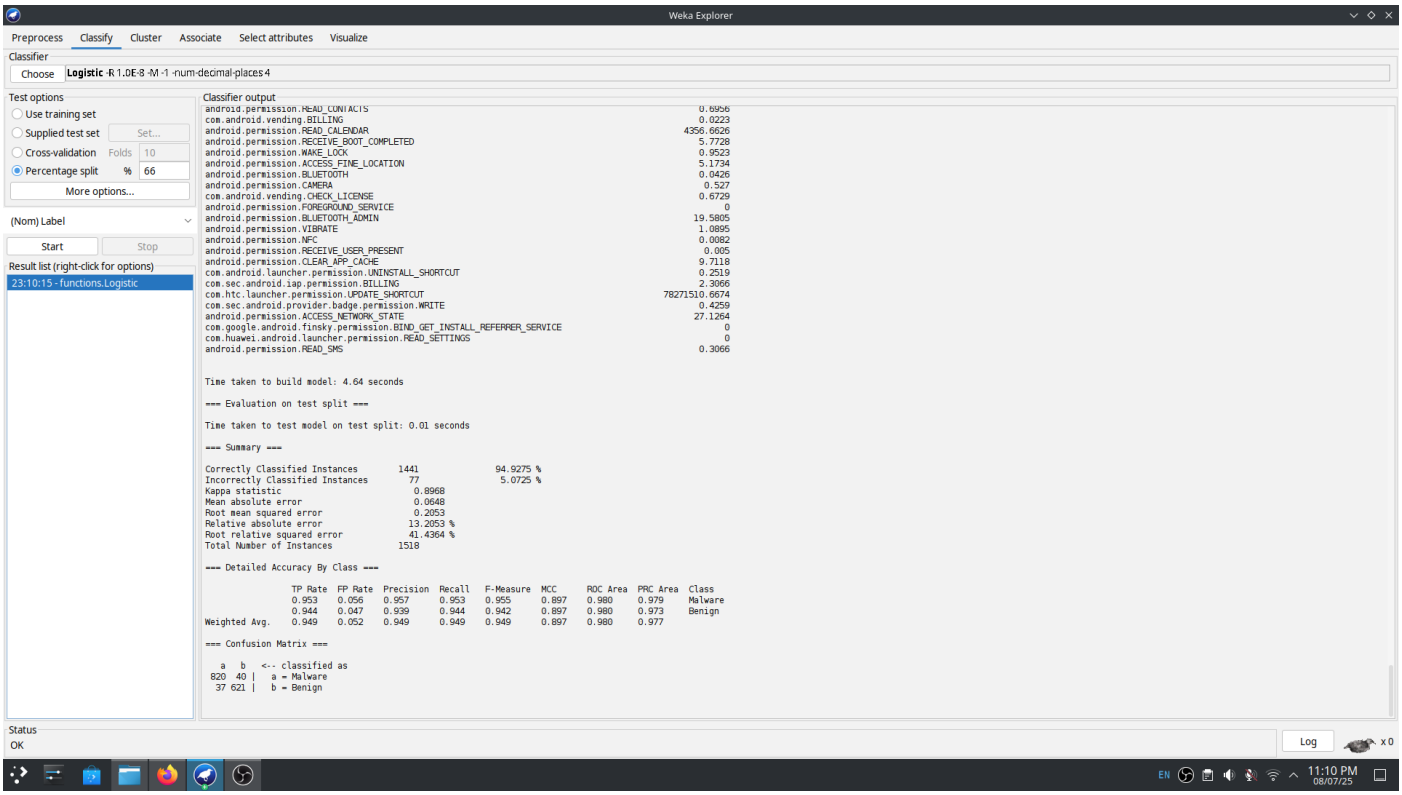
Step 6: Choose “Logistic” available under “functions”.



Step 7: Select “Percentage Split”.



Step 8: Click “Start” to train the model and review the output.



Practical – 4

Aim: Perform malicious URL detection using SVM algorithm.

Cell:

```
import pandas as pd
import numpy as np
from urllib.parse import urlparse
import re
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.preprocessing import StandardScaler
from tld import get_tld, exceptions
import matplotlib.pyplot as plt
import seaborn as sns
```

Cell:

```
df = pd.read_csv('malicious_phish.csv')

sample_size = 50000
df = df.sample(n=sample_size, random_state=42).reset_index(drop=True)

expected_url_col = 'url'
expected_label_col = 'type'

df.dropna(subset=['url', 'type'], inplace=True)
df['Label'] = df['type'].apply(lambda x: 0 if x.lower() == 'benign' else 1)

print("\nBinary Class Distribution (0=Benign, 1=Malicious):")
print(df['Label'].value_counts())
```

Output:

```
Binary Class Distribution (0=Benign, 1=Malicious):
Label
0    32914
1    17086
Name: count, dtype: int64
```

Cell:

```
def get_num_directories(url):
    try:
        path = urlparse(url).path
        return path.count('/')
    except:
        return 0

def is_ip_address_url(url):
    try:
        host = urlparse(url).netloc
        if re.match(r"^\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}$", host):
            return 1
        return 0
    except:
        return 0

def get_tld_length(url):
    try:
        tld_info = get_tld(url, as_object=True, fail_silently=True)
        if tld_info:
            return len(tld_info.tld)
        return 0
    except exceptions.TldDomainNotFound:
        return 0
    except:
        return 0

sensitive_words = ['login', 'bank', 'account', 'update', 'verify', 'paypal',
'ebay', 'amazon', 'security', 'webscr', 'confirm']
def count_sensitive_words(url):
    count = 0
    url_lower = url.lower()
    for word in sensitive_words:
        if word in url_lower:
            count += 1
    return count
```

Cell:

```
df['url_length'] = df[expected_url_col].apply(len)

df['num_dots'] = df[expected_url_col].apply(lambda x: x.count('.'))

df['has_at_symbol'] = df[expected_url_col].apply(lambda x: 1 if '@' in x else 0)

df['has_double_slash_in_path'] = df[expected_url_col].apply(lambda x: 1 if
x.count('//') > 1 else 0)

df['num_hyphens'] = df[expected_url_col].apply(lambda x: x.count('-'))

df['num_question_marks'] = df[expected_url_col].apply(lambda x: x.count('?'))

df['num_equals'] = df[expected_url_col].apply(lambda x: x.count('='))

df['num_directories'] = df[expected_url_col].apply(get_num_directories)

df['is_ip_address'] = df[expected_url_col].apply(is_ip_address_url)

df['hostname_length'] = df[expected_url_col].apply(lambda x:
len(urlparse(x).netloc))

df['tld_length'] = df[expected_url_col].apply(get_tld_length)

df['has_https'] = df[expected_url_col].apply(lambda x: 1 if x.startswith('https')
                                             else 0)

df['num_sensitive_words'] = df[expected_url_col].apply(count_sensitive_words)
```

Cell:

```
features = [
    'url_length', 'num_dots', 'has_at_symbol', 'has_double_slash_in_path',
    'num_hyphens', 'num_question_marks', 'num_equals', 'num_directories',
    'is_ip_address', 'hostname_length', 'tld_length', 'has_https',
    'num_sensitive_words'
]
X = df[features]
y = df['Label']
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Cell:

```
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
                                                    random_state=42, stratify=y)
print(f"\nTraining set size: {X_train.shape[0]}")
print(f"Testing set size: {X_test.shape[0]}")
print(f"Training set Malicious ratio: {y_train.sum() / len(y_train):.2f}")
print(f"Testing set Malicious ratio: {y_test.sum() / len(y_test):.2f}")
```

Output:

Training set size: 40000
Testing set size: 10000
Training set Malicious ratio: 0.34
Testing set Malicious ratio: 0.34

Cell:

```
svm_model = SVC(kernel='rbf', C=1.0, random_state=42, verbose=True,
                shrinking=False)

svm_model.fit(X_train, y_train)
```

Output:

[LibSVM].....*
optimization finished, #iter = 10916
obj = -7960.662193, rho = 0.242135
nSV = 8948, nBSV = 8515
Total nSV = 8948

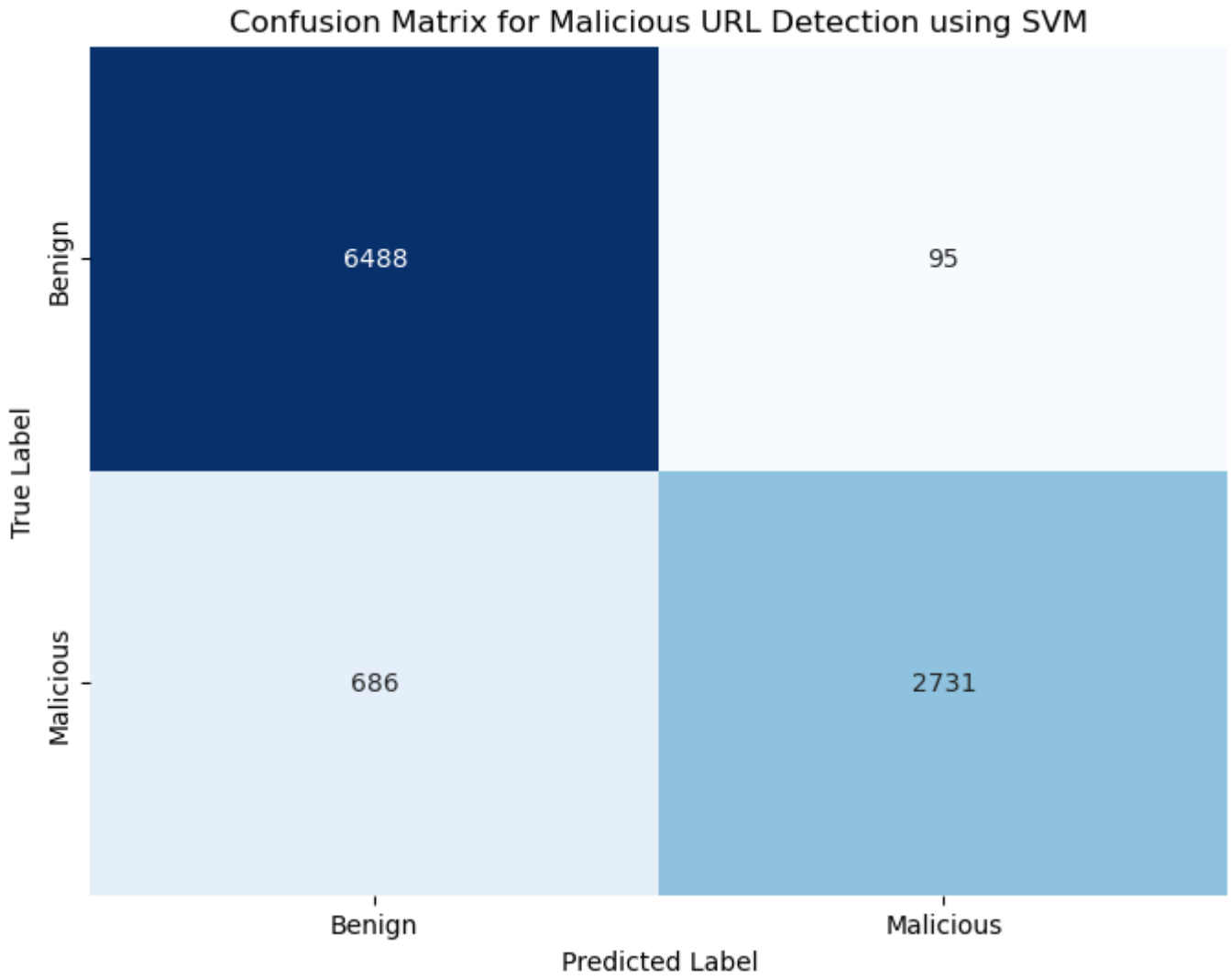
Cell:

```
y_pred = svm_model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print(f"\nAccuracy: {accuracy:.4f}")

print("\nConfusion Matrix:")
cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Benign', 'Malicious'],
            yticklabels=['Benign', 'Malicious'])
plt.title('Confusion Matrix for Malicious URL Detection using SVM')
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.show()
```

Output:

Accuracy: 0.9219

Confusion Matrix:



Practical – 5

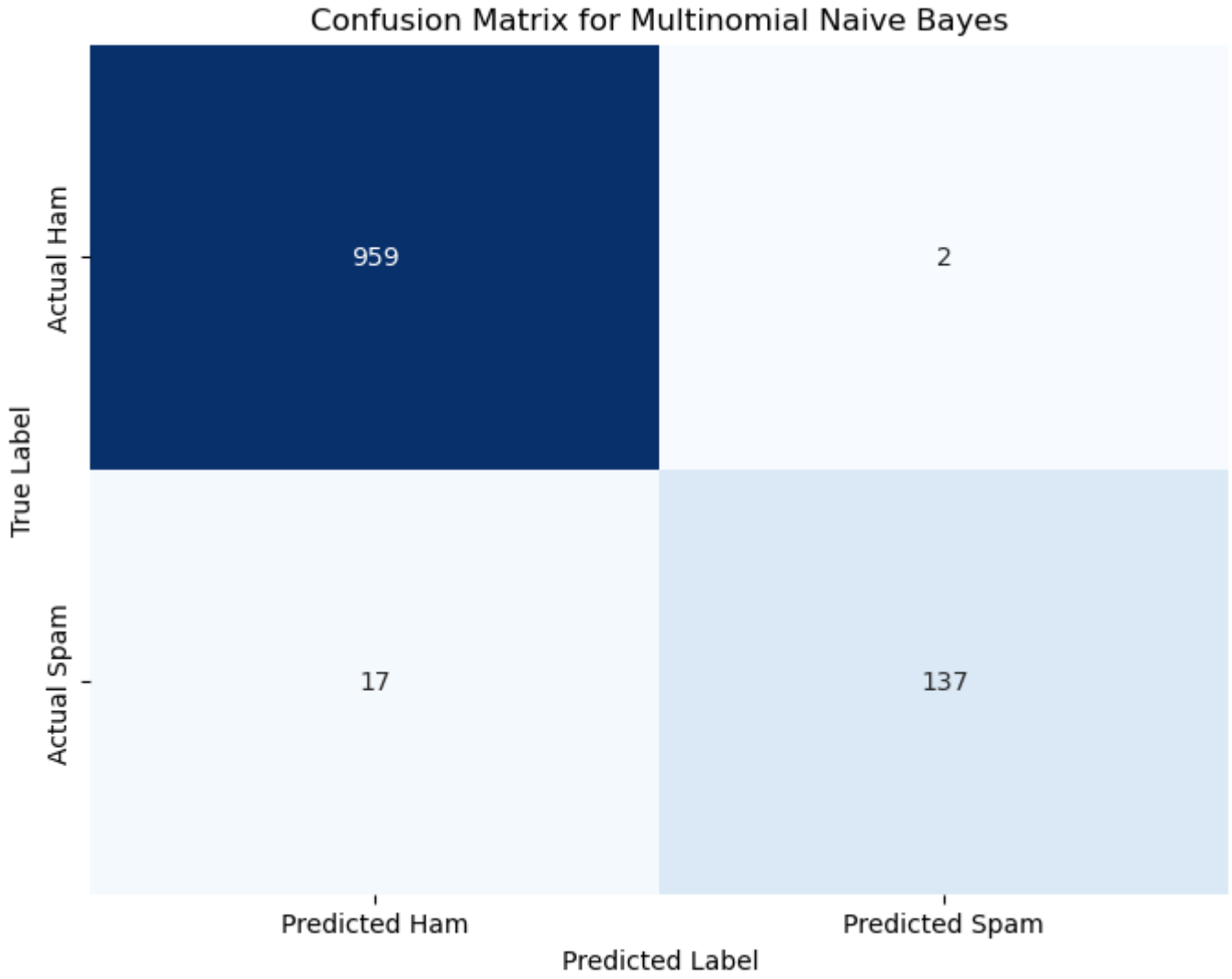
Aim: Perform SMS spam detection using Logistic regression algorithm.

Code:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
df = pd.read_csv('spam.csv', encoding='latin-1')[['v1', 'v2']]
df.columns = ['label', 'text']

df['label_num'] = df['label'].map({'ham': 0, 'spam': 1})
X_train, X_test, y_train, y_test = train_test_split(
    df['text'], df['label_num'], test_size=0.2, random_state=99
)

vectorizer = CountVectorizer()

X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
clf_lr = LogisticRegression(solver='liblinear', random_state=42)
clf_lr.fit(X_train_vec, y_train)
y_pred_lr = clf_lr.predict(X_test_vec)


accuracy = accuracy_score(y_test, y_pred_lr)
cm = confusion_matrix(y_test, y_pred_lr)

print(f'Accuracy: {accuracy:.4f}')

plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Predicted Ham', 'Predicted Spam'],
            yticklabels=['Actual Ham', 'Actual Spam'])
plt.title('Confusion Matrix for Multinomial Naive Bayes')
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.show()
```

Output:

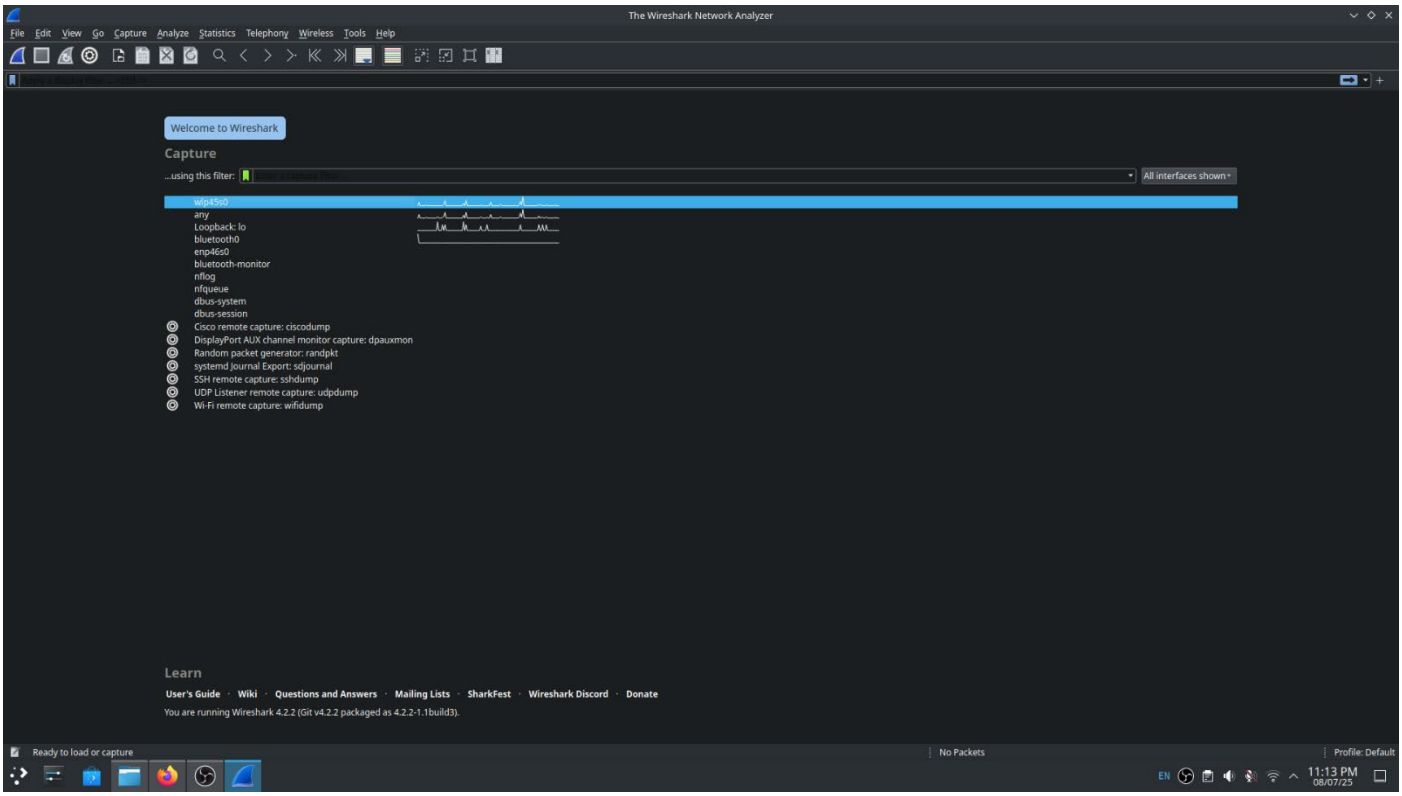
Accuracy: 0.9830



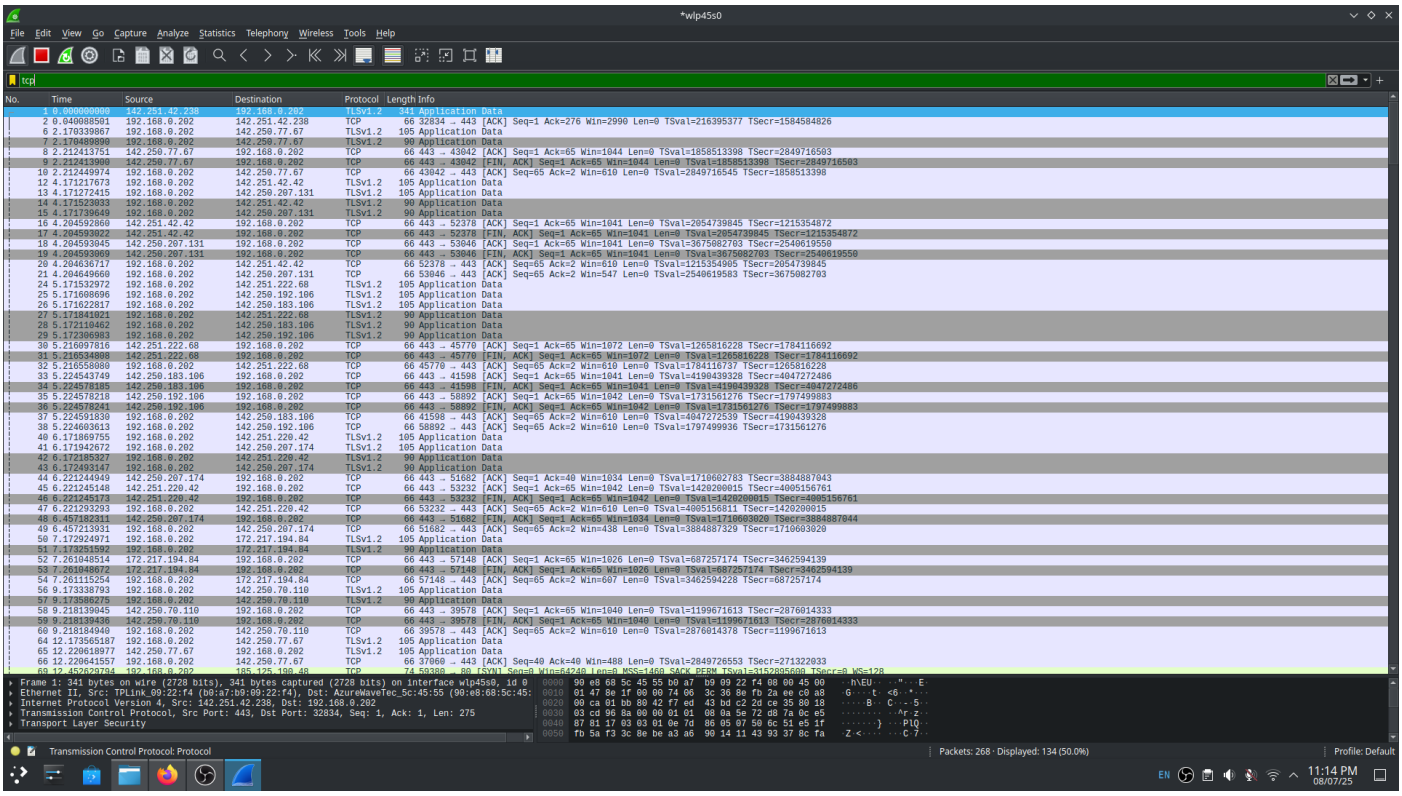
Practical – 6

Aim: Capture and analyse network traffic using packet capture tools

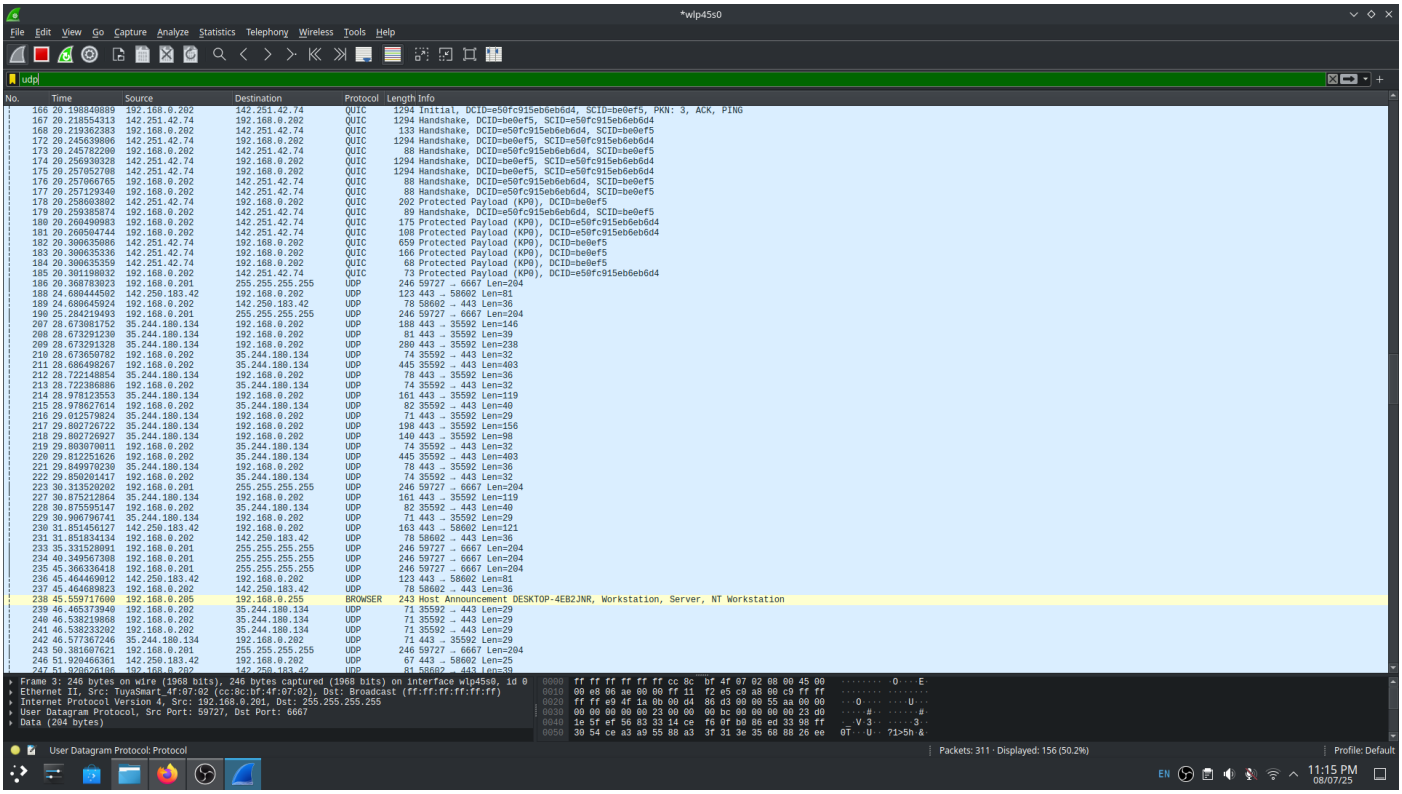
Step 1: Select and open “wlp45s0”.



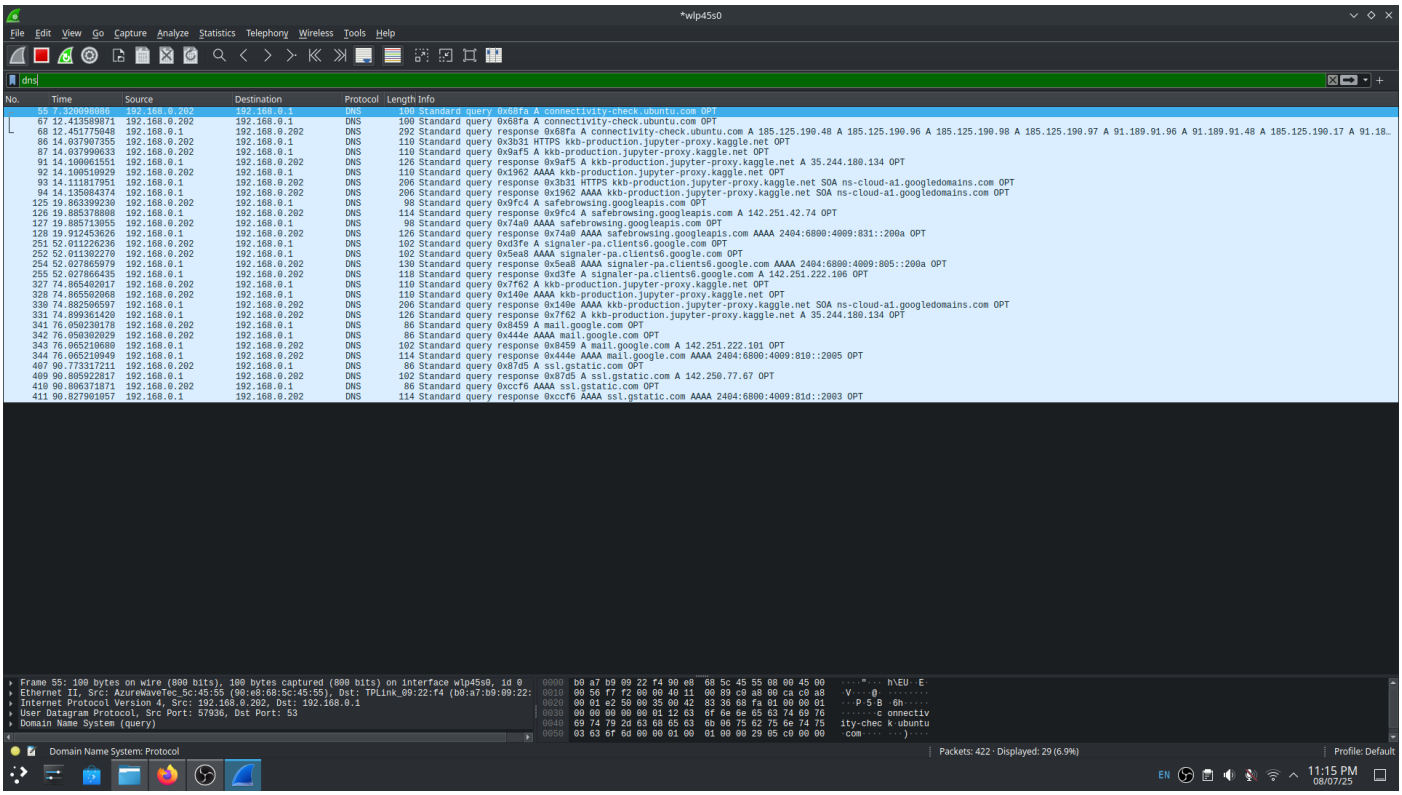
Step 2: Type “tcp” in filters section and hit Enter to view tcp packets.



Step 3: Type “udp” in filters section and hit Enter to view udp packets.



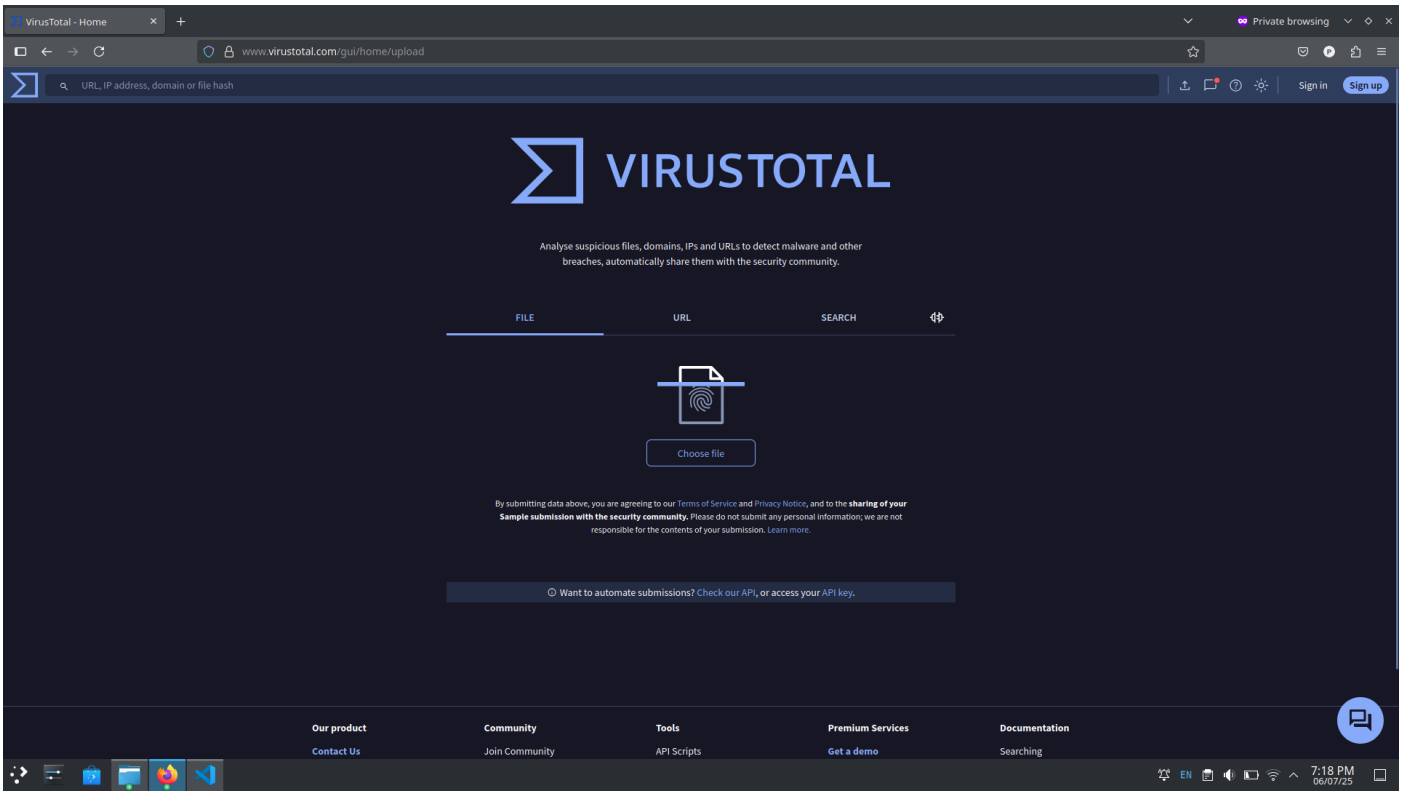
Step 4: Type “dns” in filters section and hit Enter to view dns packets.



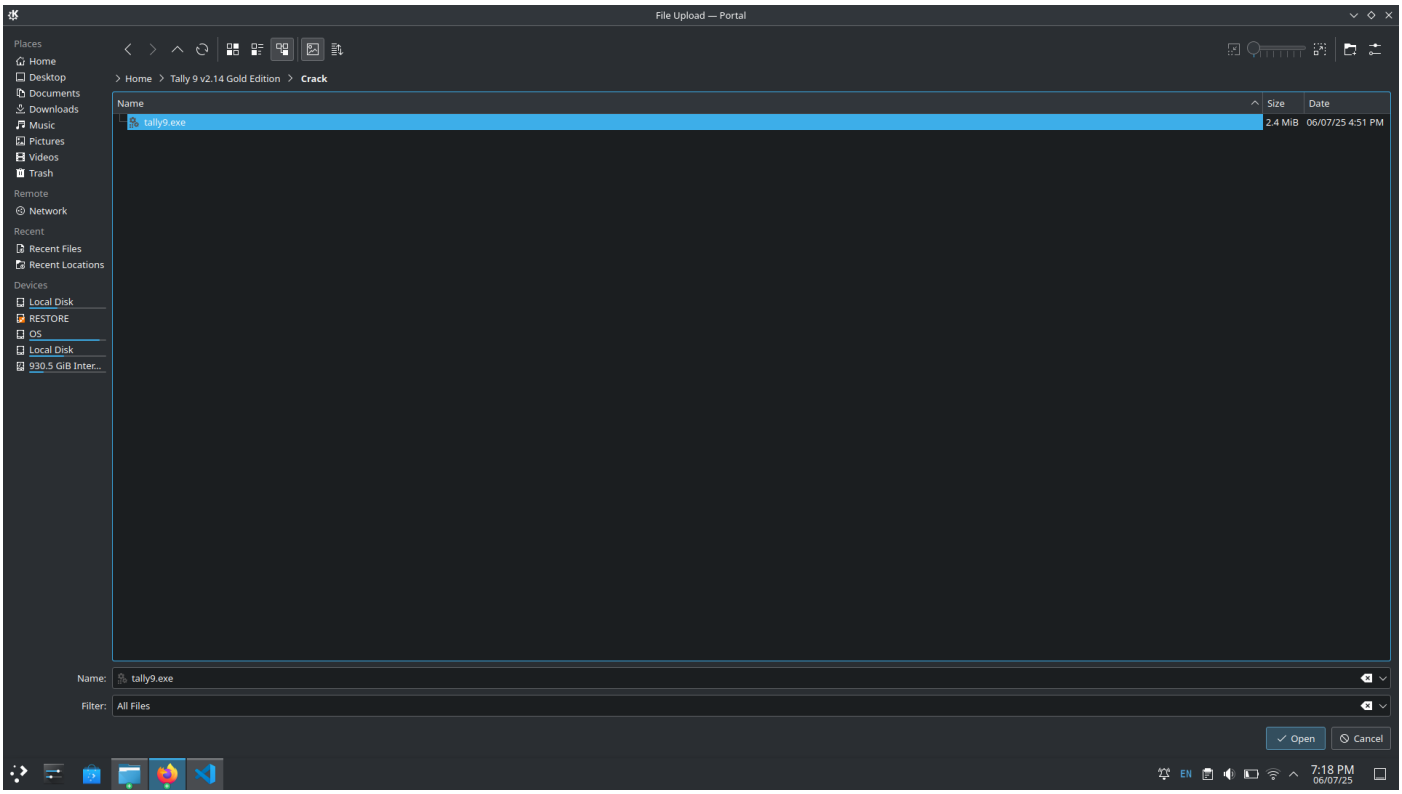
Practical – 7

Aim: Identification of Virus infected file using virustotal.com

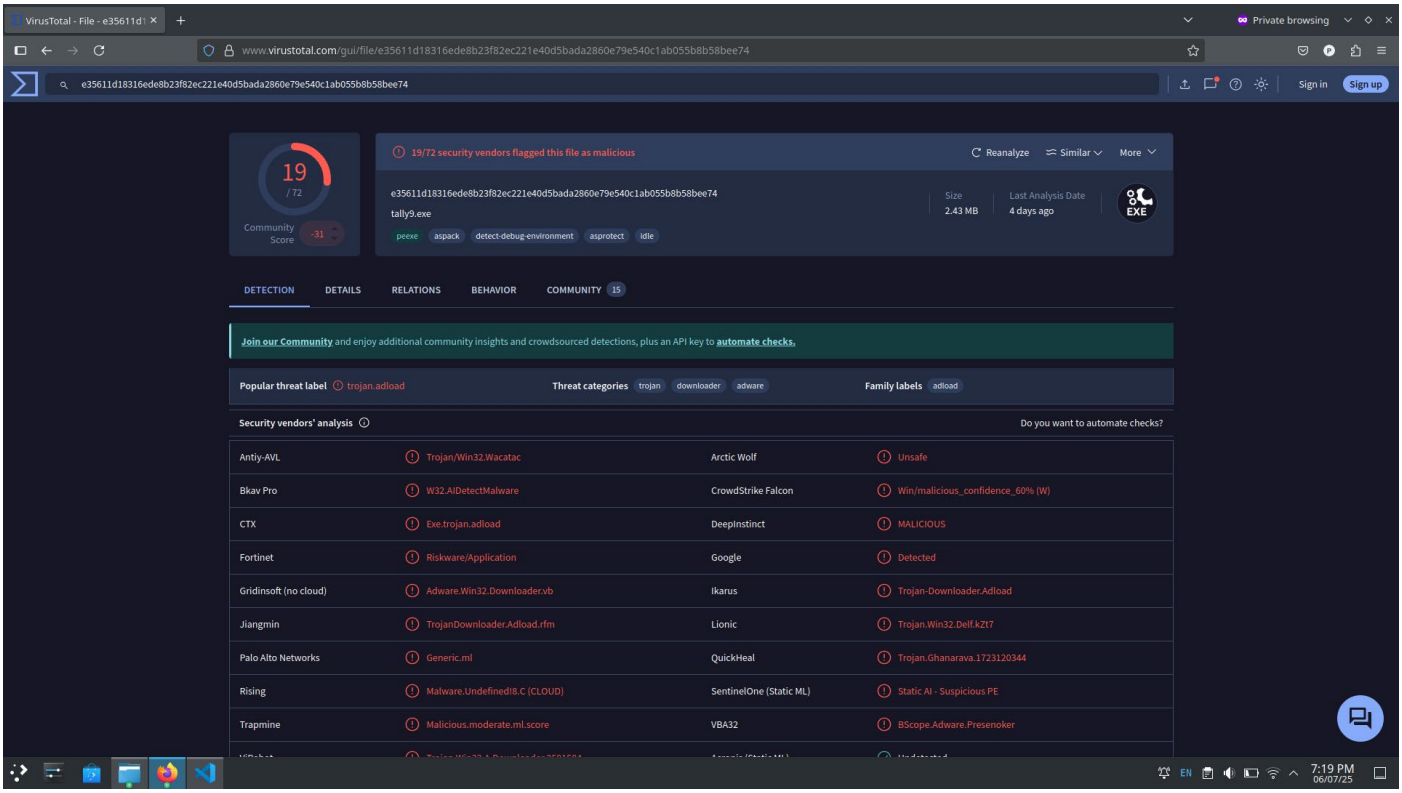
Step 1: Click “Choose file”.

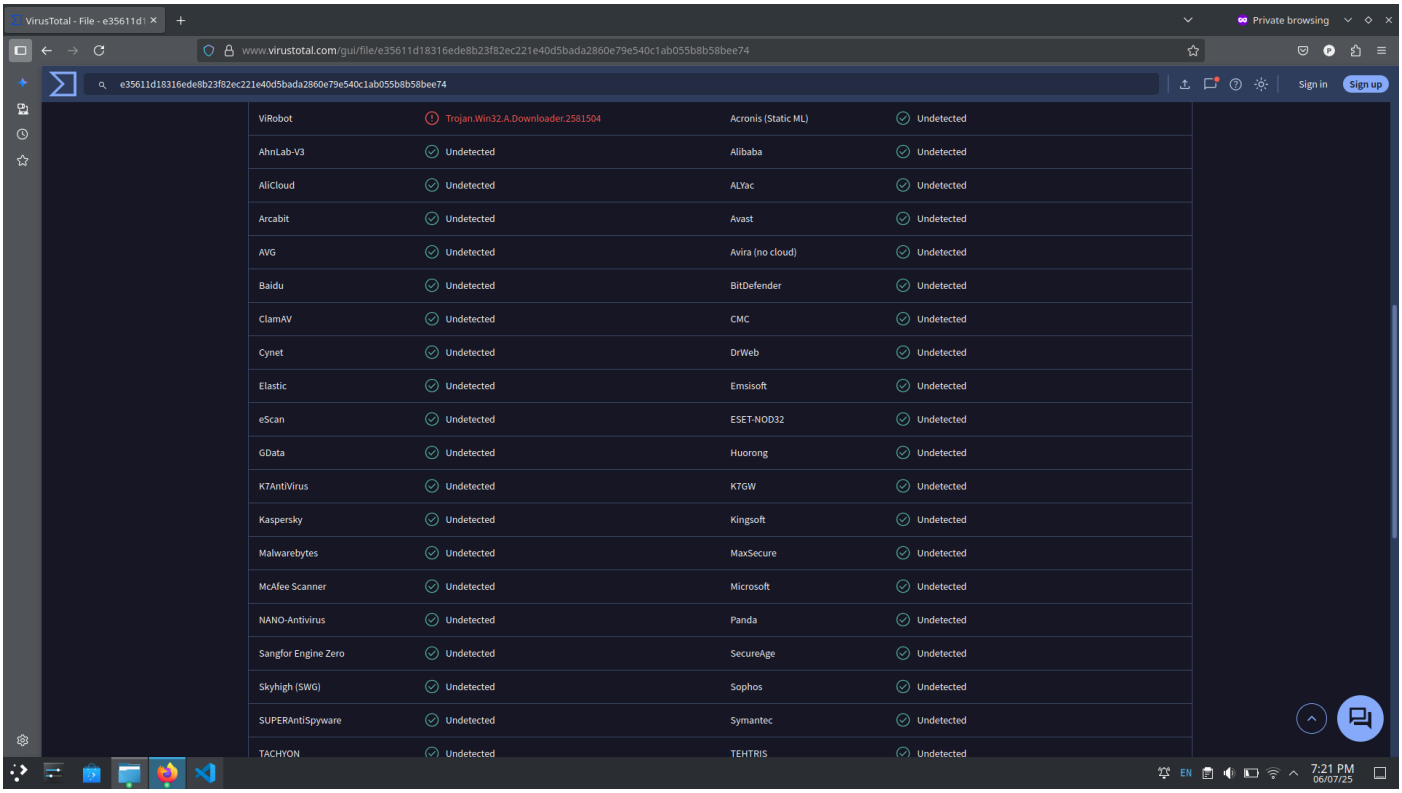


Step 2: Select and open any file you want to scan.



Step 3: Review the detected threats.



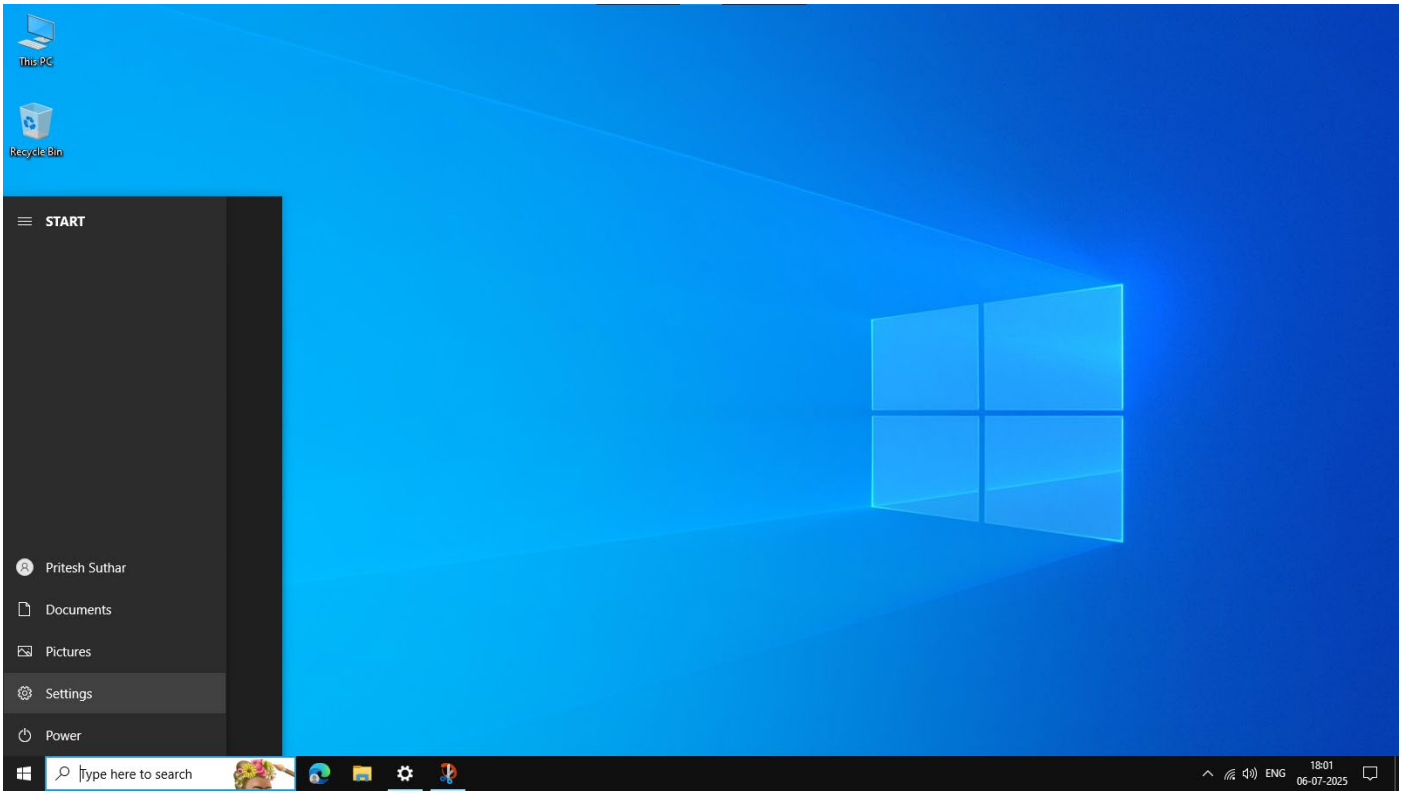

VirusTotal - File - e35611d1
Private browsing
www.virustotal.com/gui/file/e35611d18316ede8b23f82ec221e40d5bada2860e79e540c1ab055b8b58bee74
e35611d18316ede8b23f82ec221e40d5bada2860e79e540c1ab055b8b58bee74
Sign in
Sign up
ViRobot Trojan.Win32.A.Downloader.2581504 Acronis (Static ML) Undetected
AhnLab-V3 Undetected Alibaba Undetected
AliCloud Undetected ALYac Undetected
Arcabit Undetected Avast Undetected
AVG Undetected Avira (no cloud) Undetected
Baidu Undetected BitDefender Undetected
ClamAV Undetected CMC Undetected
Cynet Undetected DrWeb Undetected
Elastic Undetected Emsisoft Undetected
eScan Undetected ESET-NOD32 Undetected
GData Undetected Huorong Undetected
K7AntiVirus Undetected K7GW Undetected
Kaspersky Undetected Kingsoft Undetected
Malwarebytes Undetected MaxSecure Undetected
McAfee Scanner Undetected Microsoft Undetected
NANO-Antivirus Undetected Panda Undetected
Sangfor Engine Zero Undetected SecureAge Undetected
Skyhigh (SWG) Undetected Sophos Undetected
SUPERAntiSpyware Undetected Symantec Undetected
TACHYON Undetected TEHTRIS Undetected
7:21 PM
06/07/25

Practical – 8

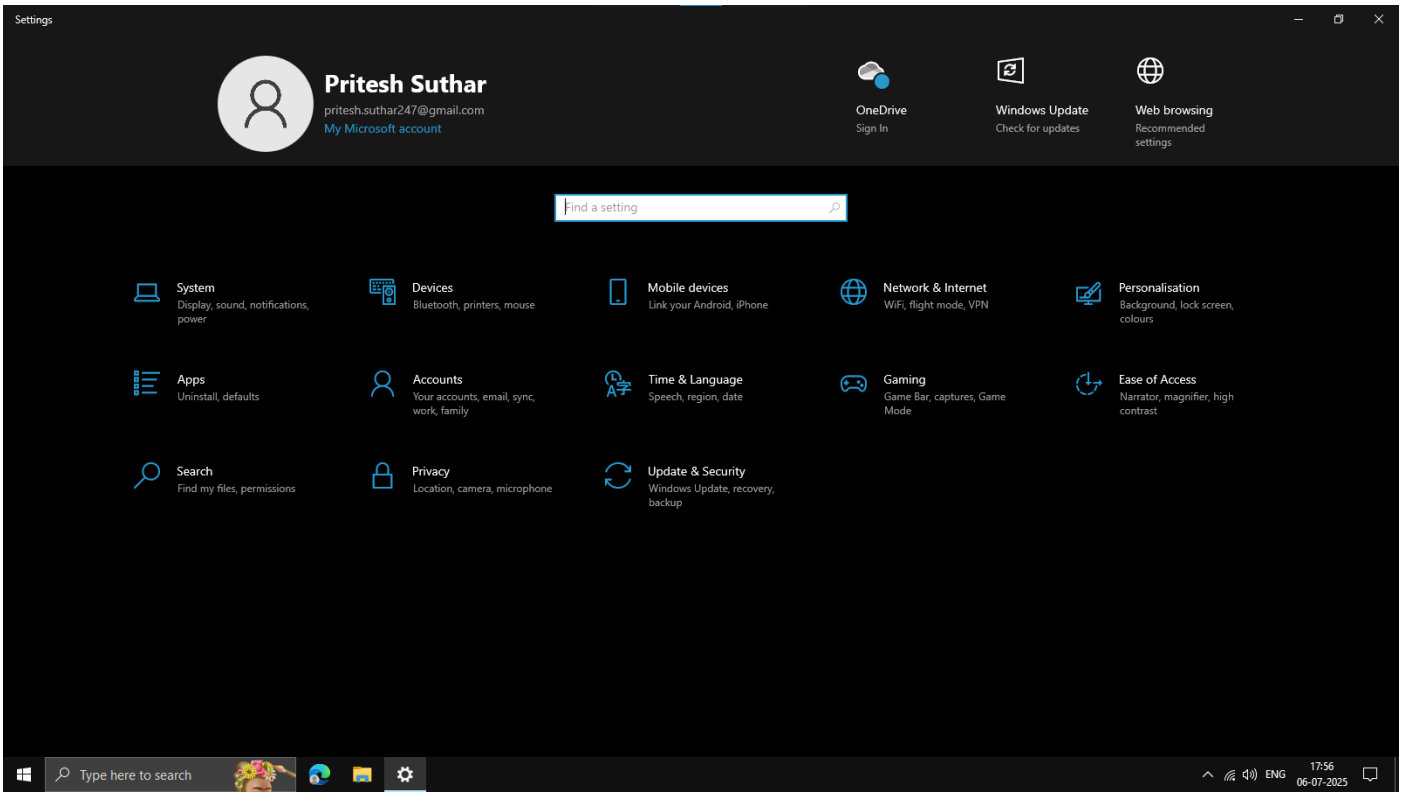
Aim: Keep operating systems and software up to date by applying patches and updates.

Windows 10:

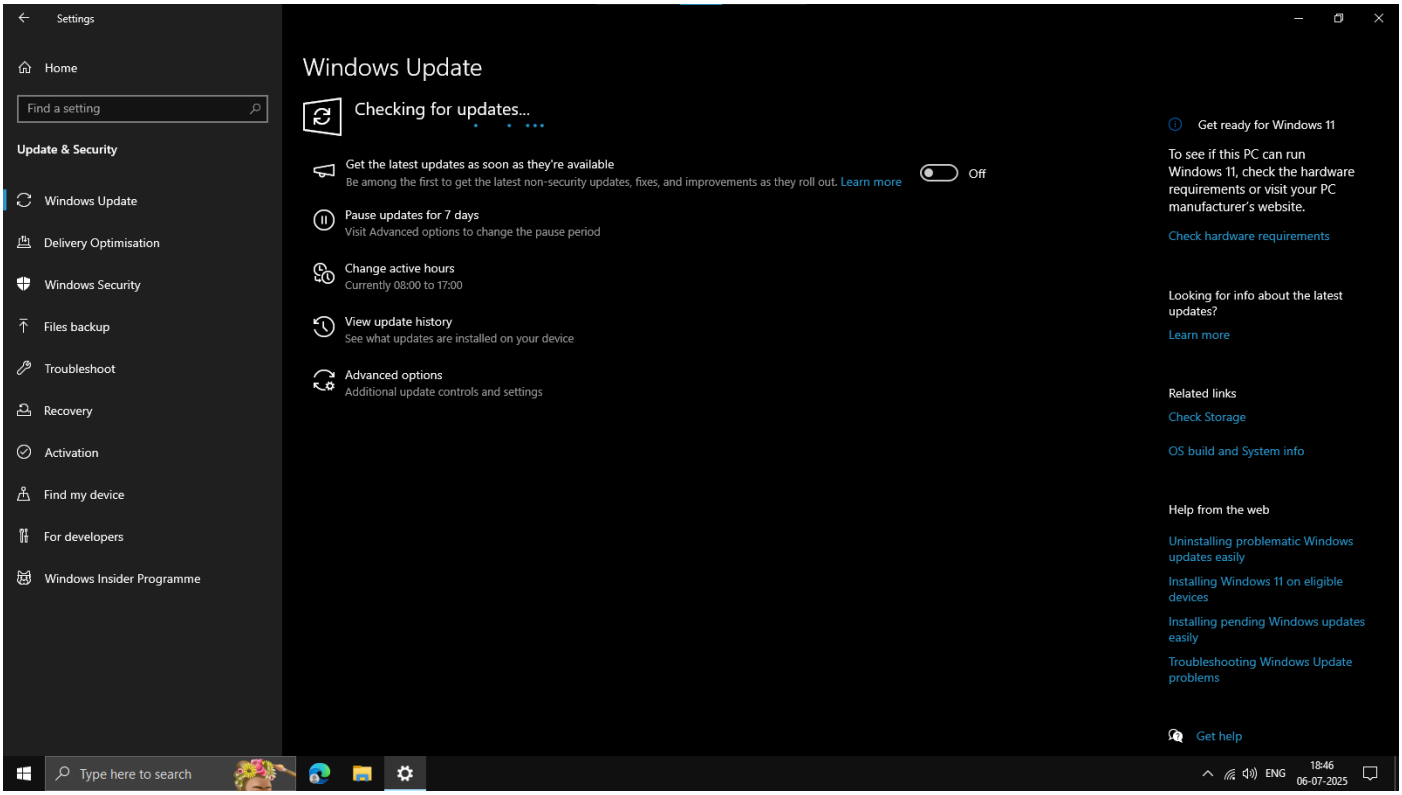
Step 1: Press the windows icon and click on “Settings”.



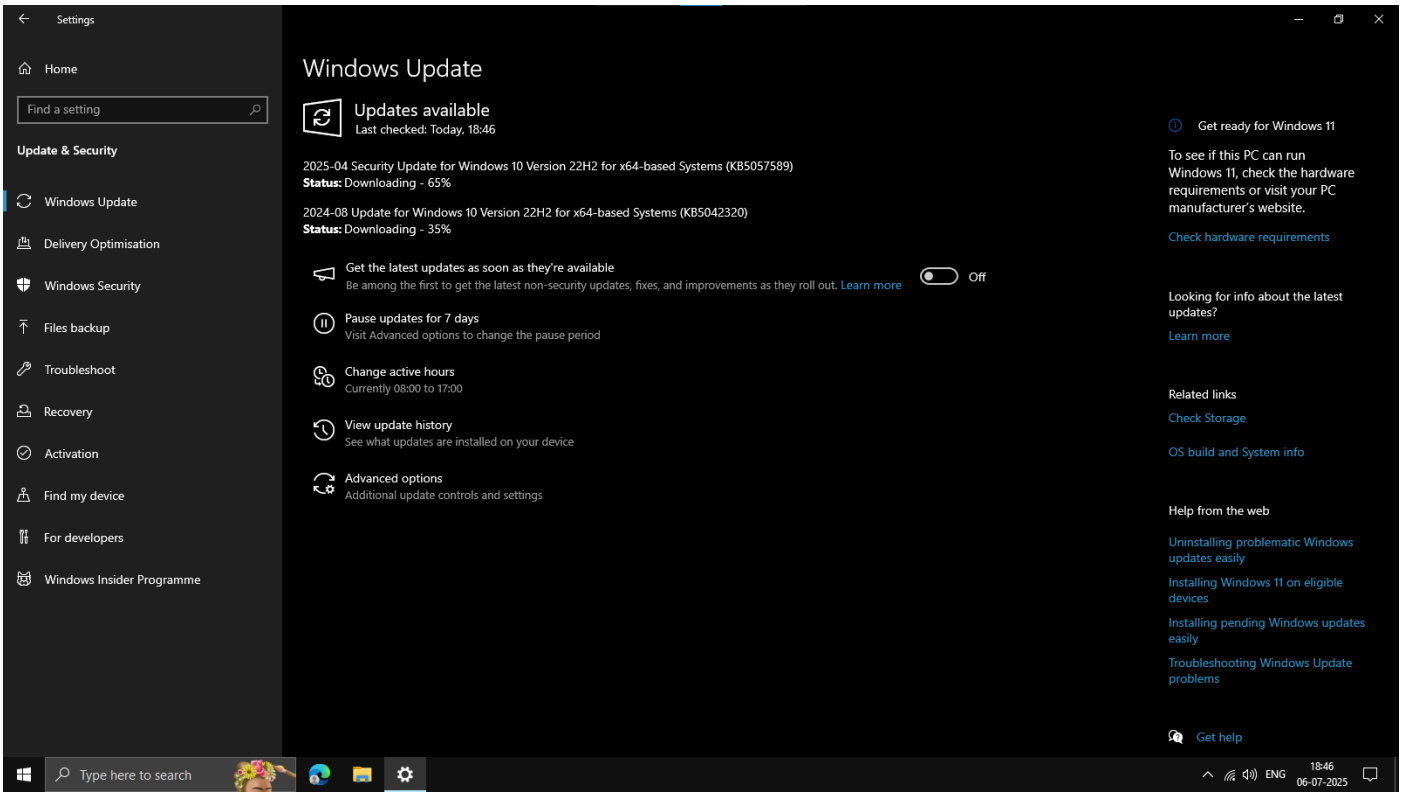
Step 2: Click on “Updates & Security”.



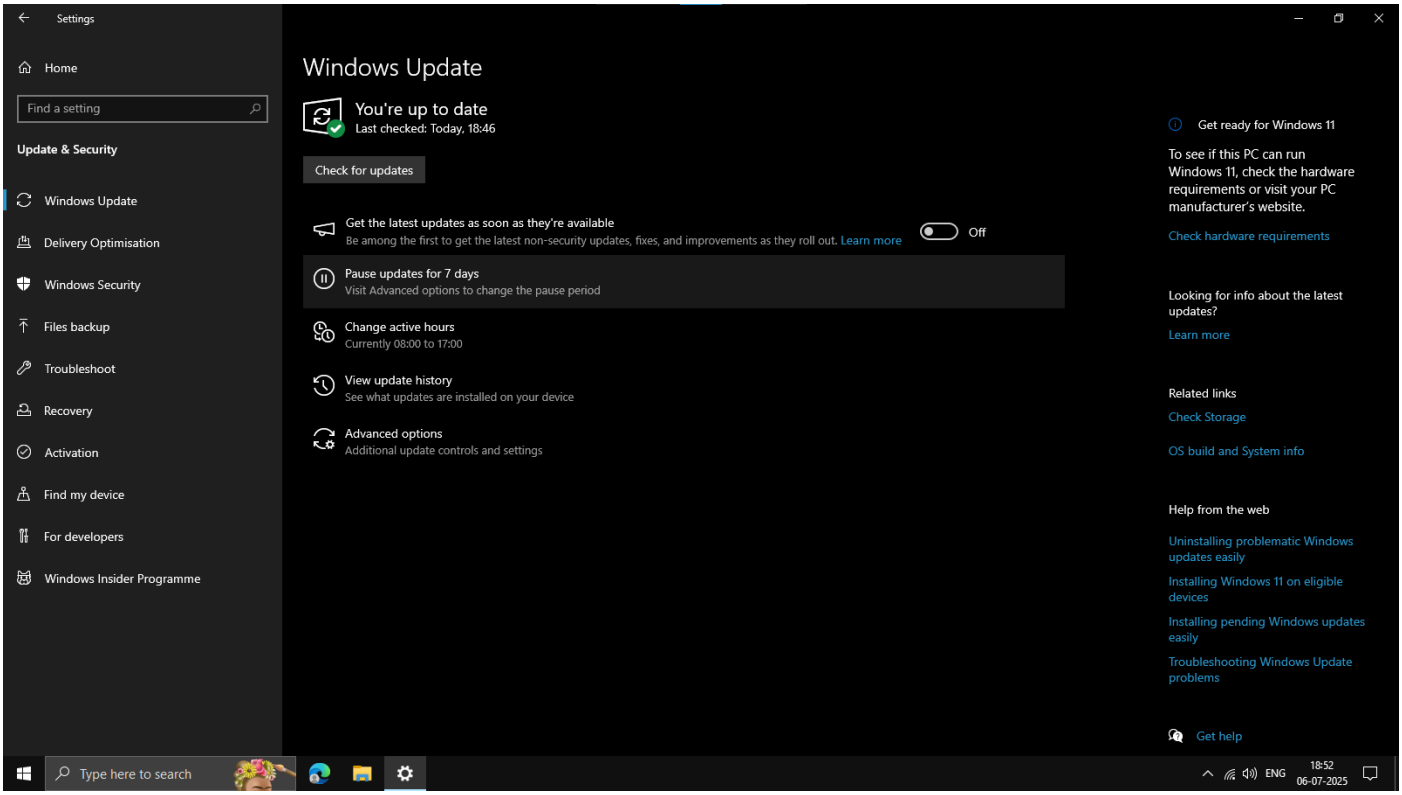
Step 3: Wait for the system to check for the updates.



Step 4: Wait for the updates to get downloaded and install.

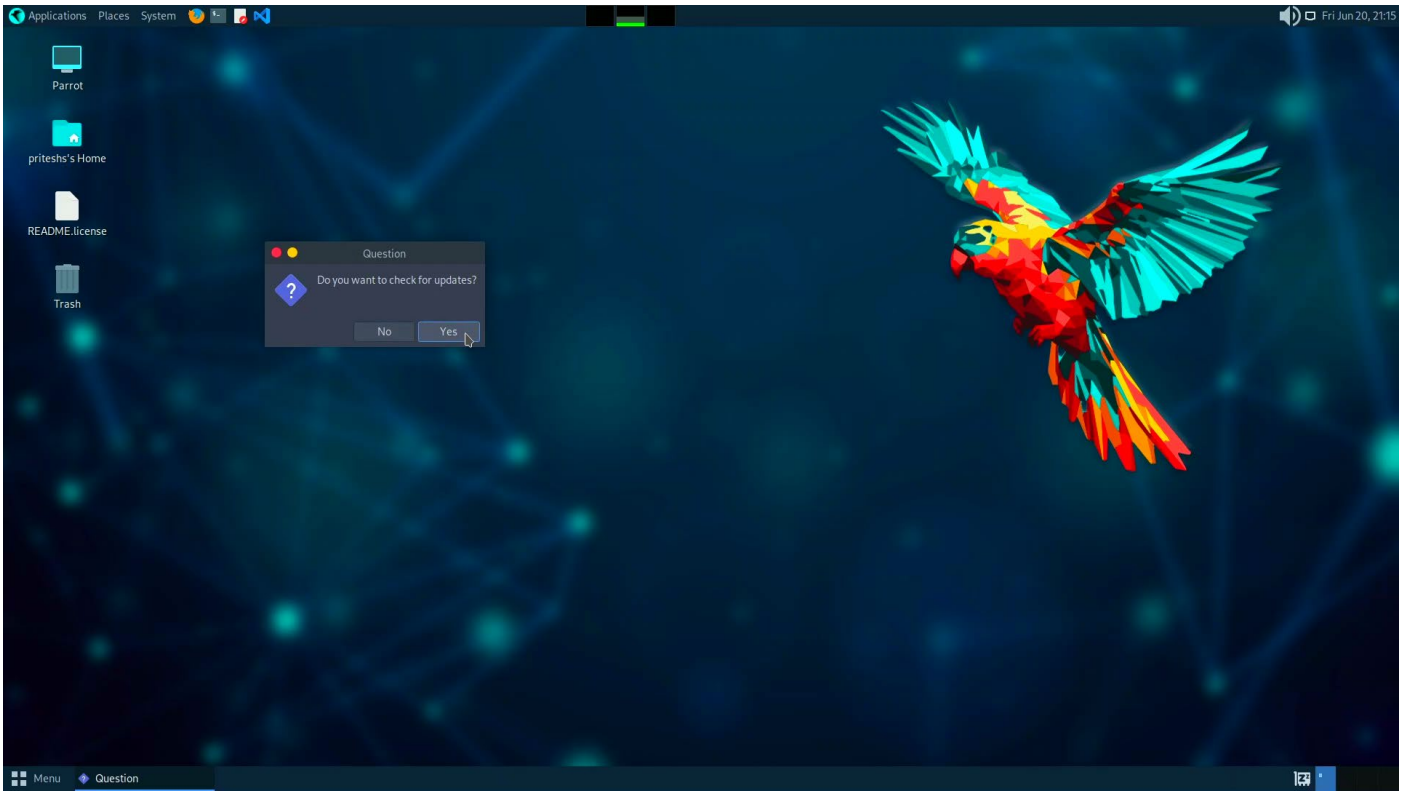


Step 5: Click on “Check for updates” to verify there aree no more pending updates.

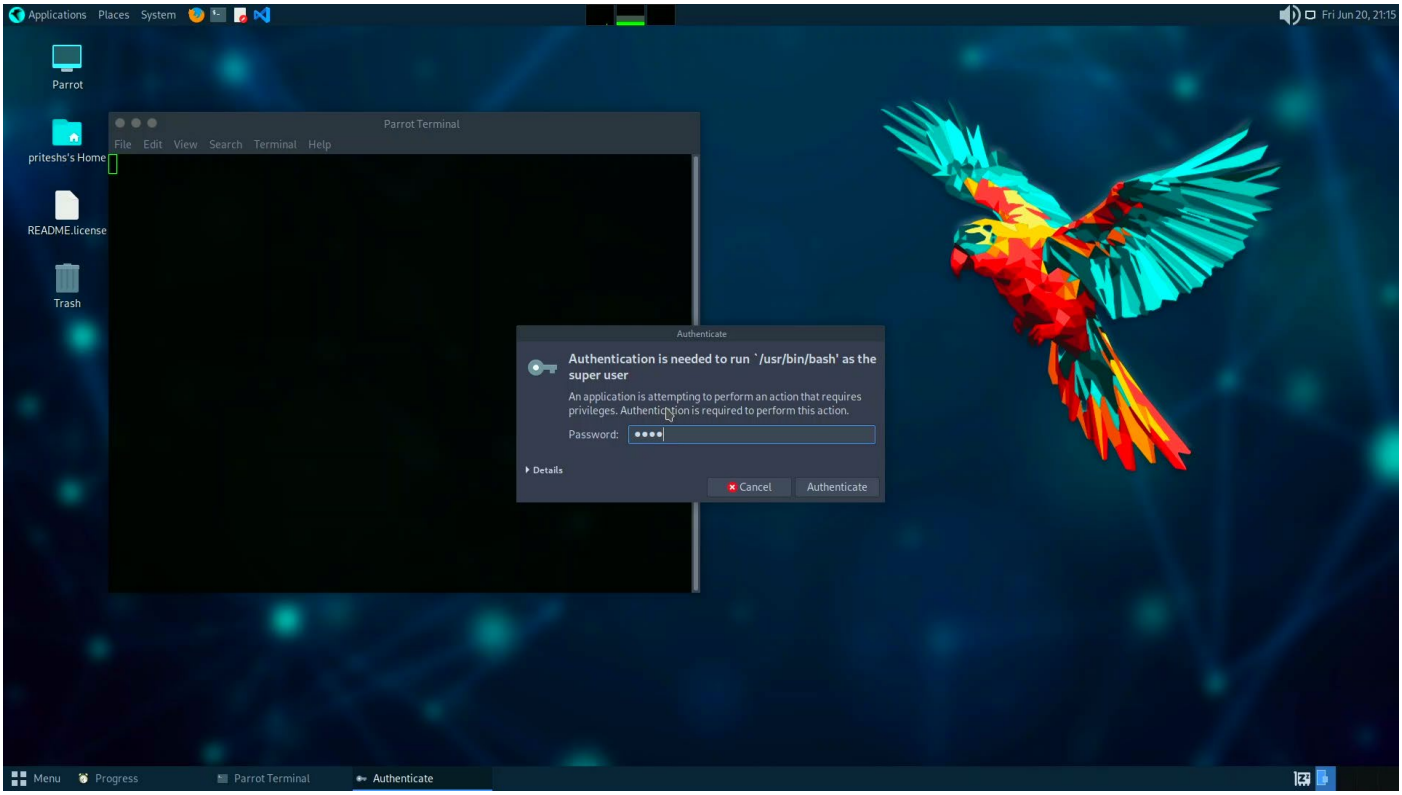


Parrot OS:

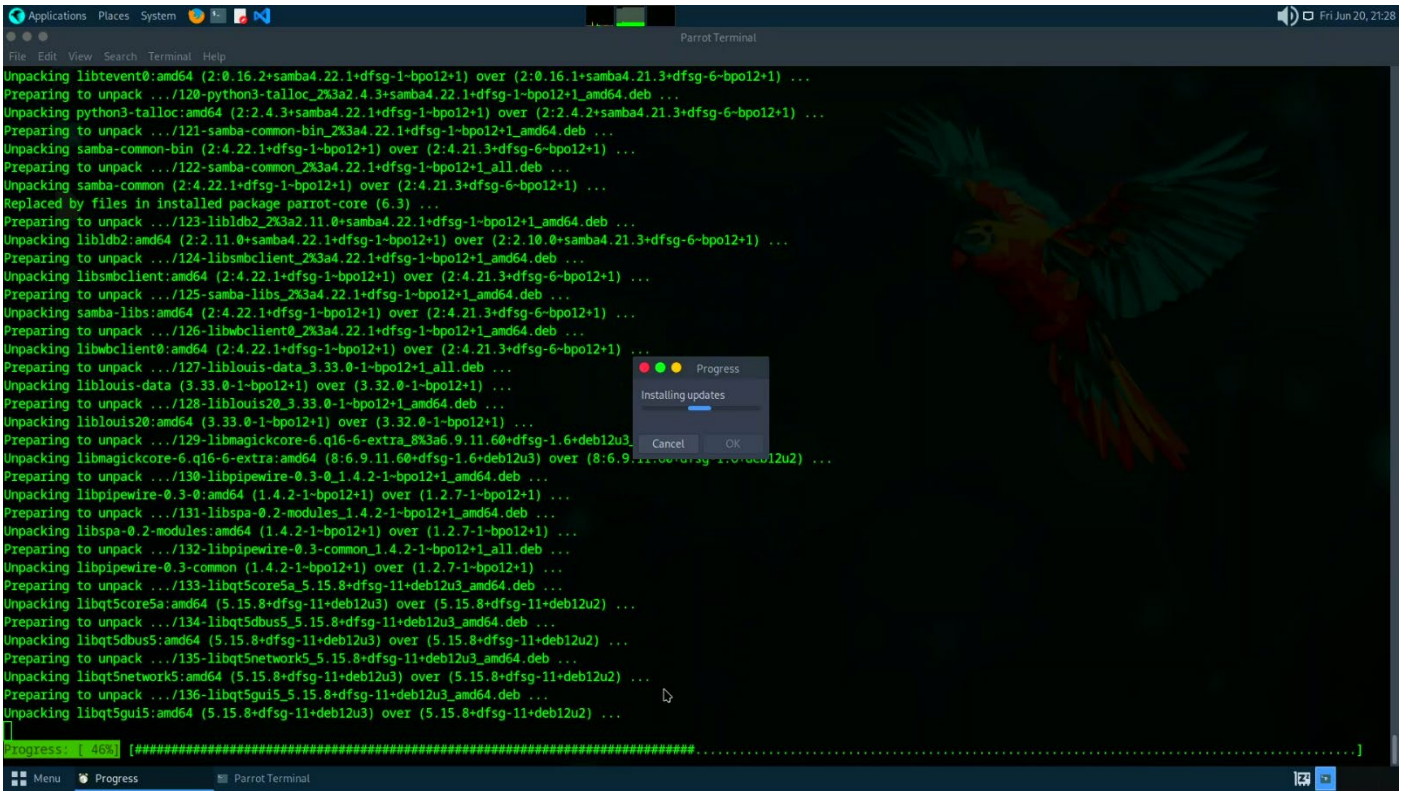
Step 1: Click on “Yes”.



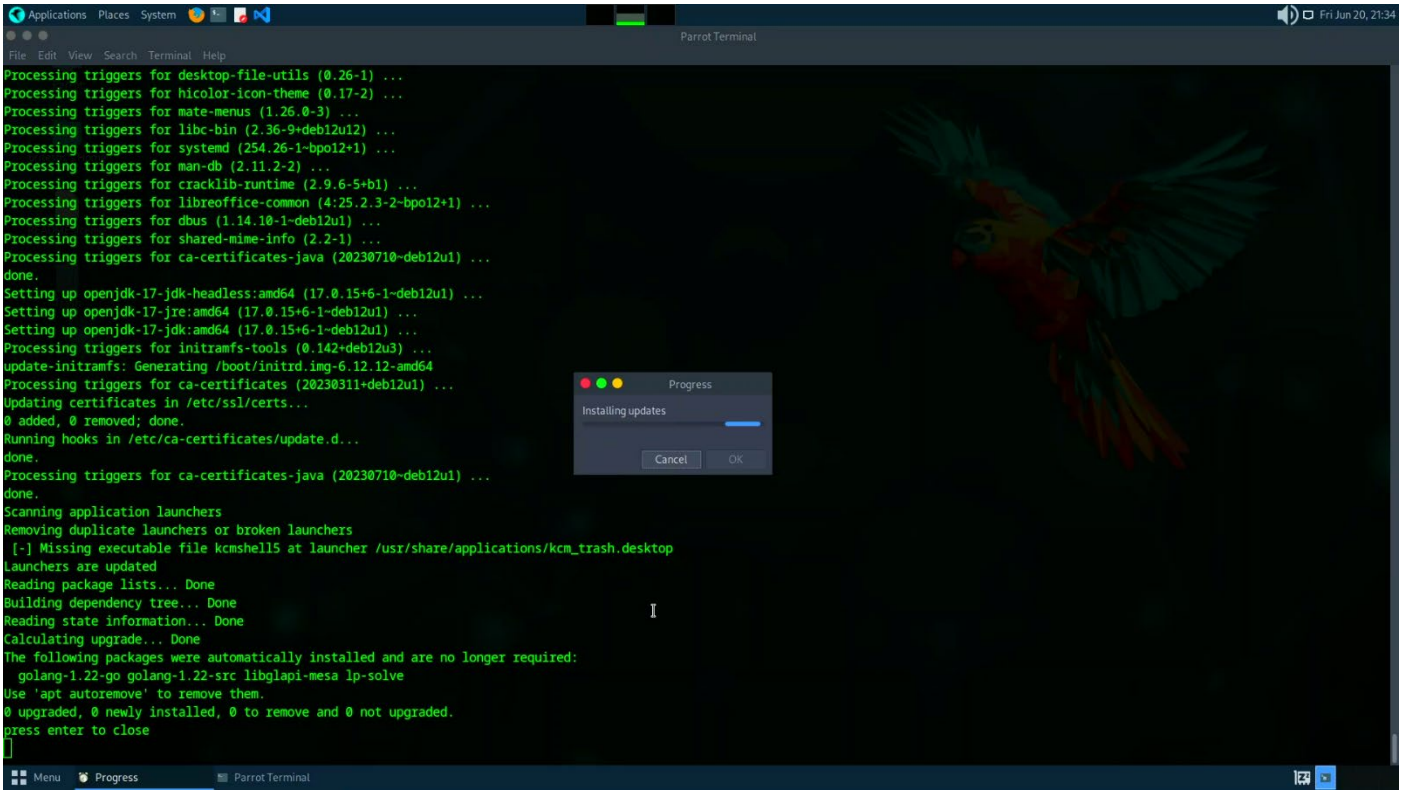
Step 2: Enter your password and hit Enter.



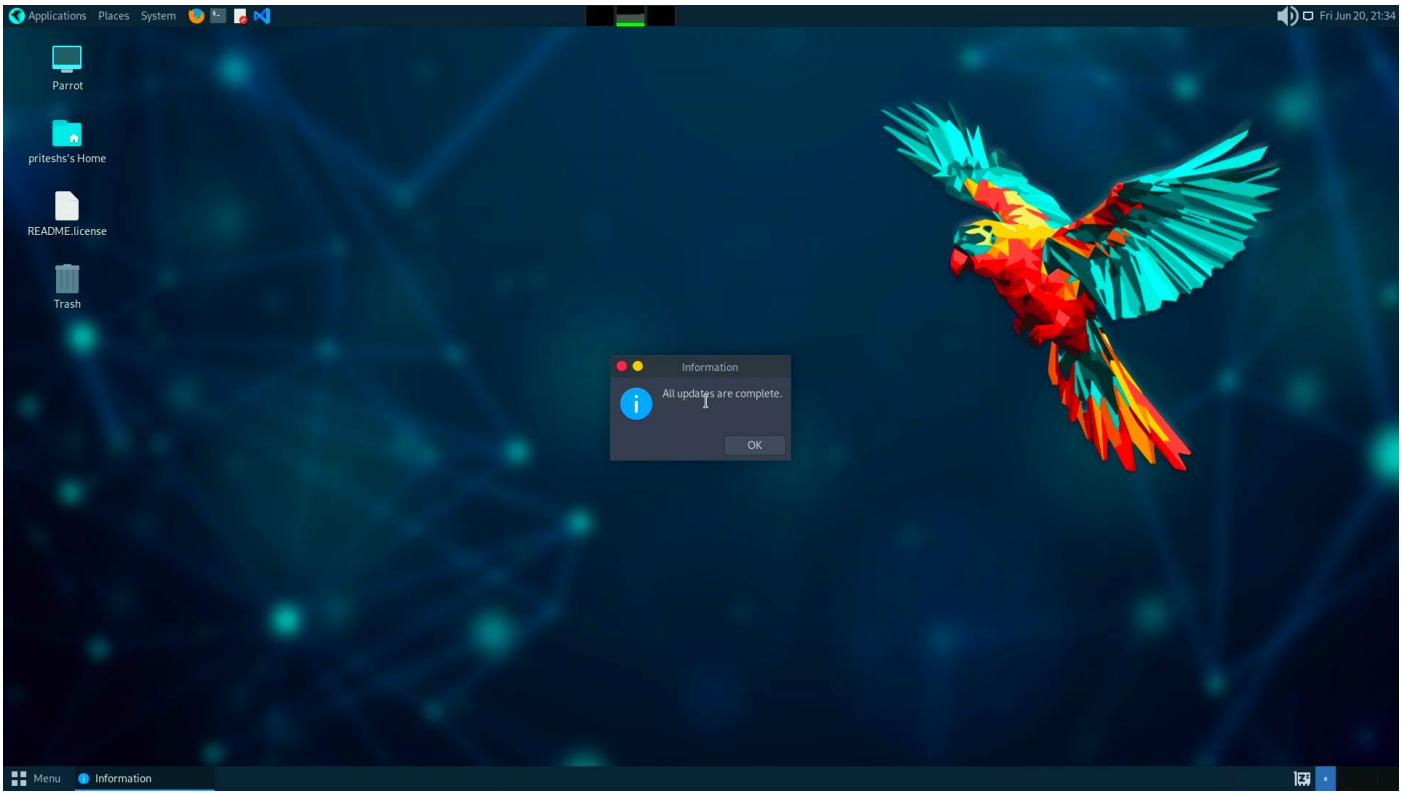
Step 3: Wait for the updates to get downloaded and installed.



Step 4: Press enter to close the terminal.



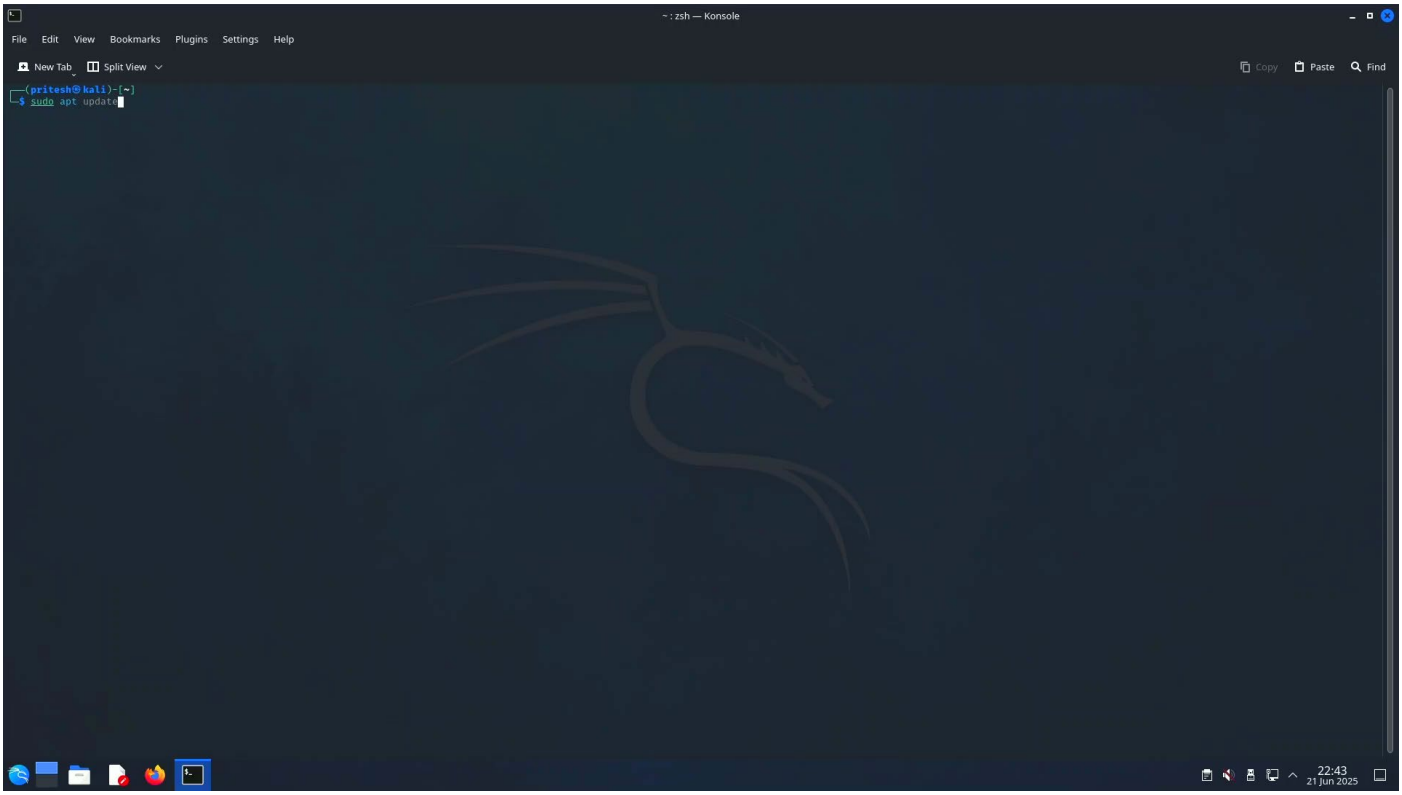
Step 5: Click “OK”.



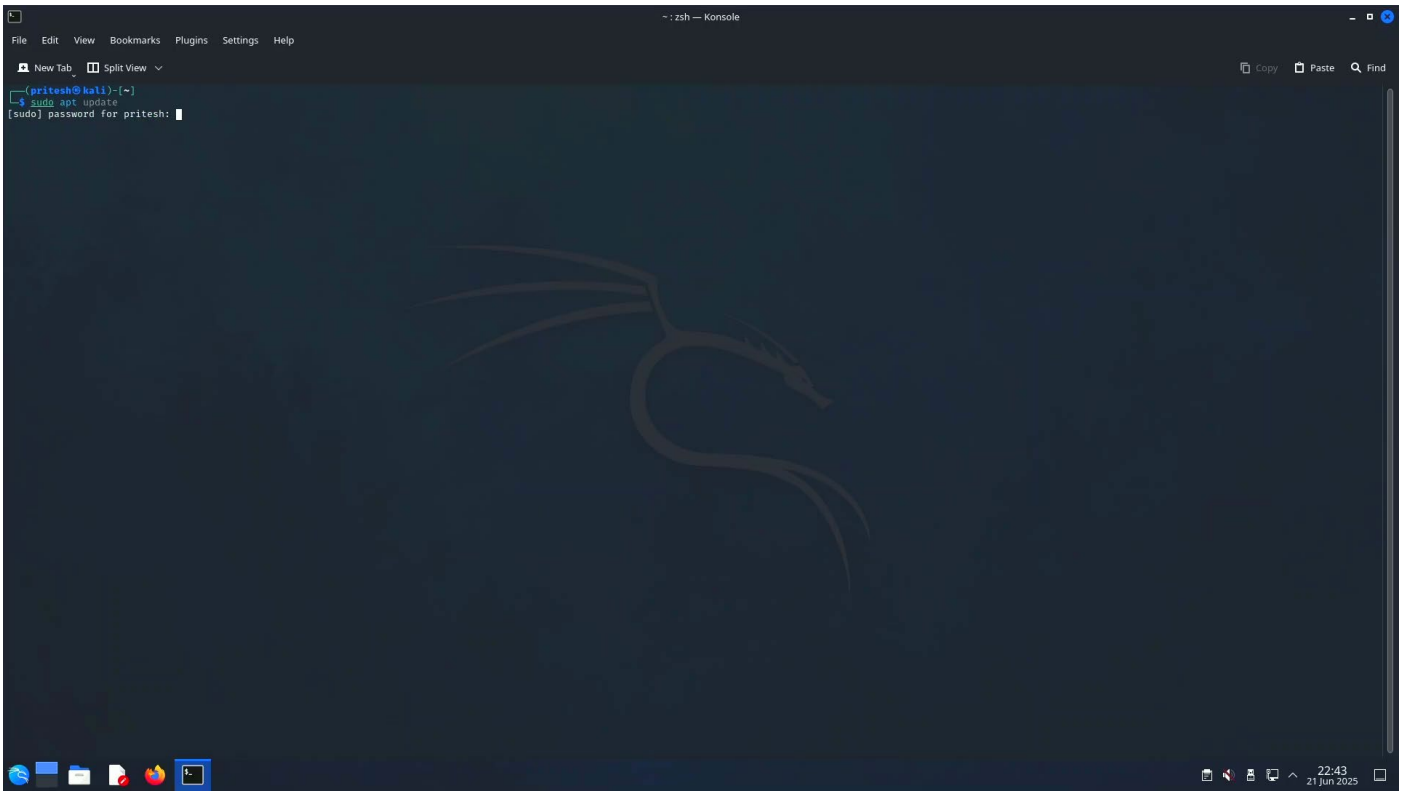
Kali Linux:

Step 1: Open Terminal by pressing “CTRL + ALT + T”.

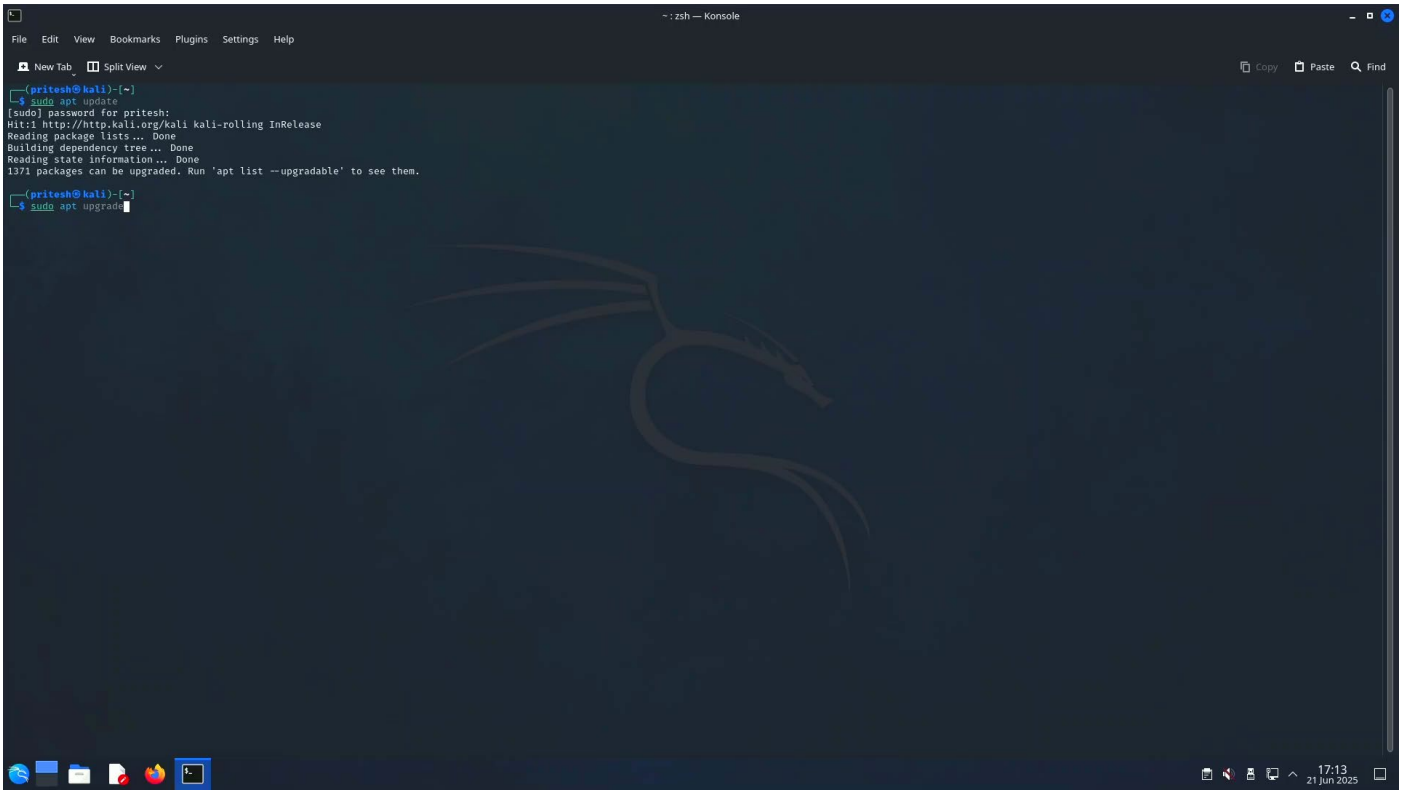
Step 2: Type “sudo apt update” and hit enter.



Step 3: Enter your password.

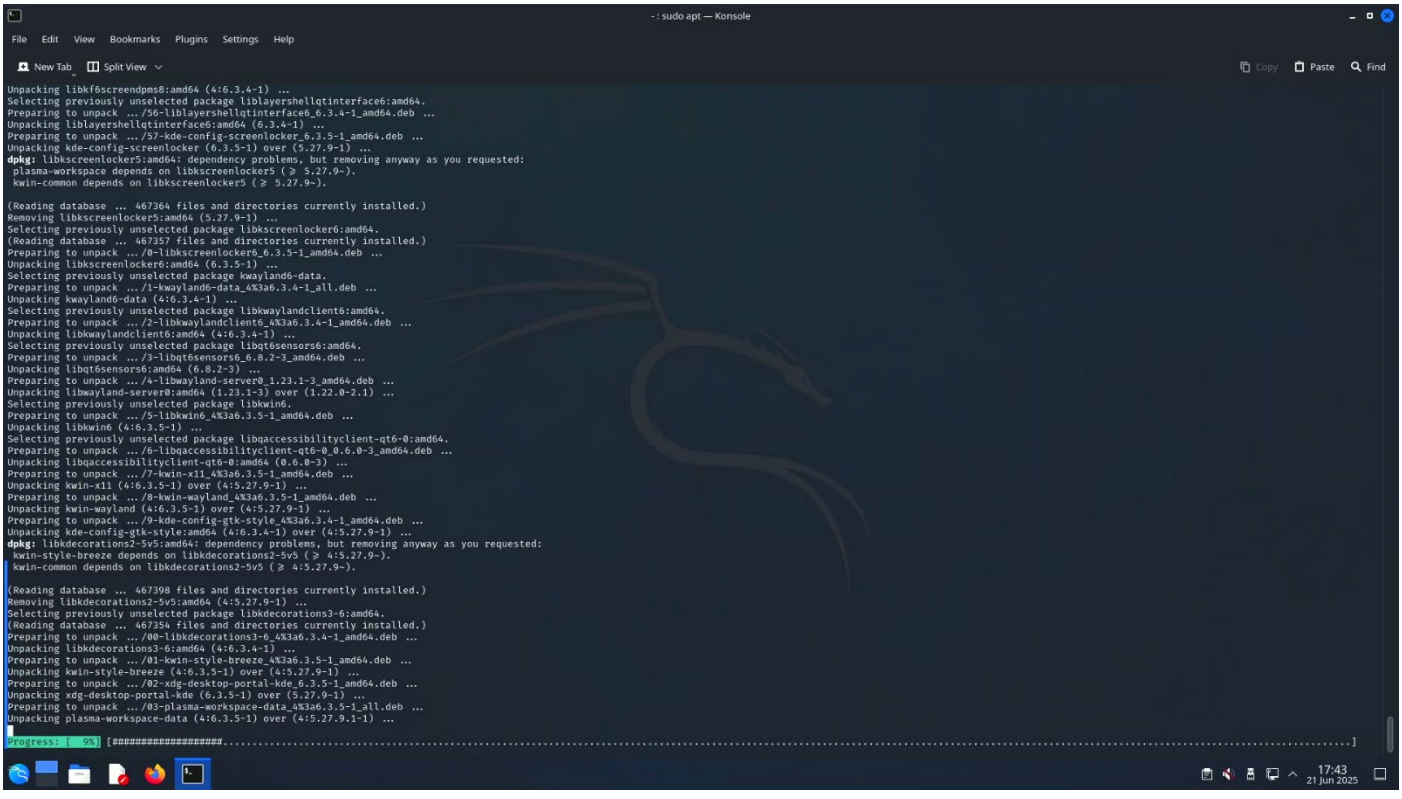


Step 4: Type “sudo apt upgrade” and hit Enter.

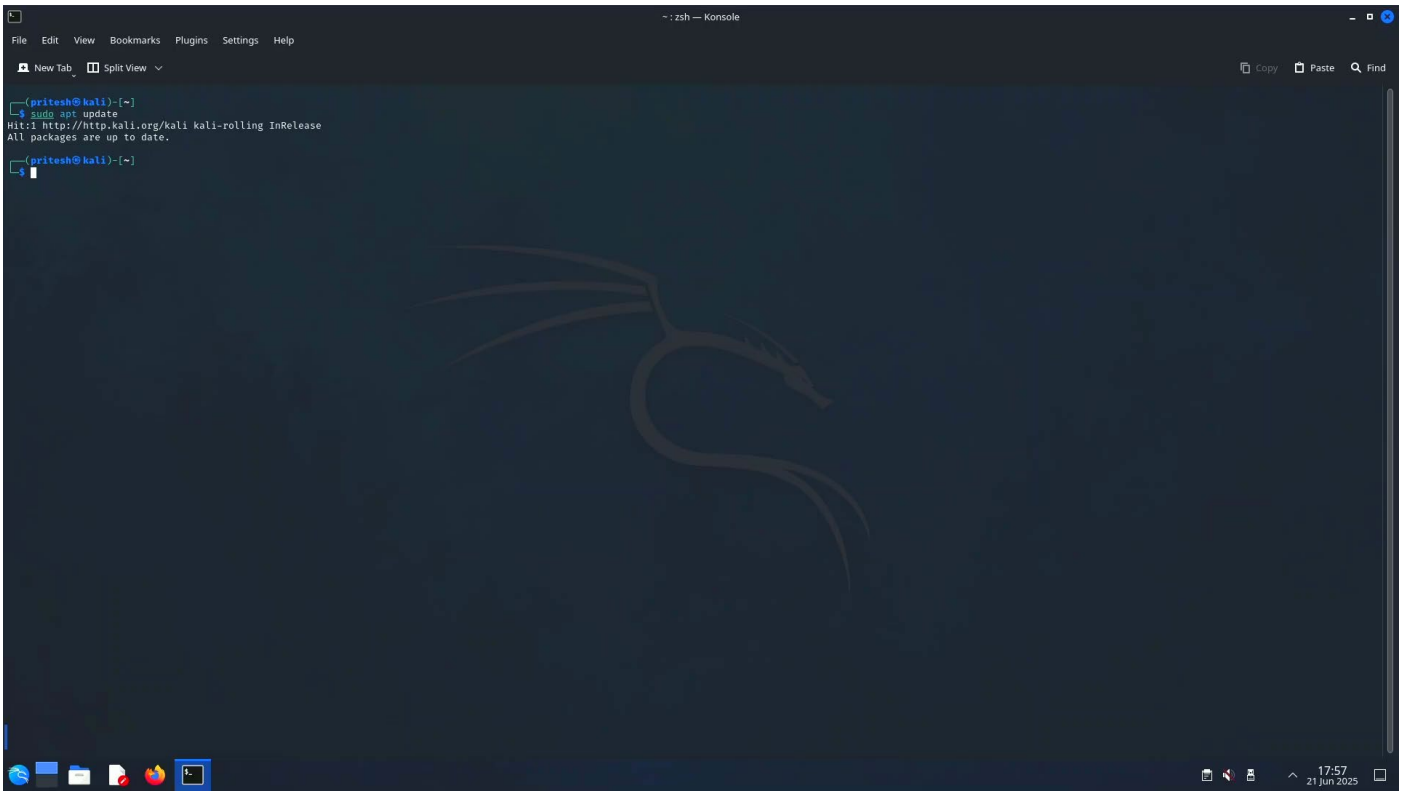


Step 5: Press “y” and hit enter.

Step 6: Wait for the updates to get downloaded and installed.



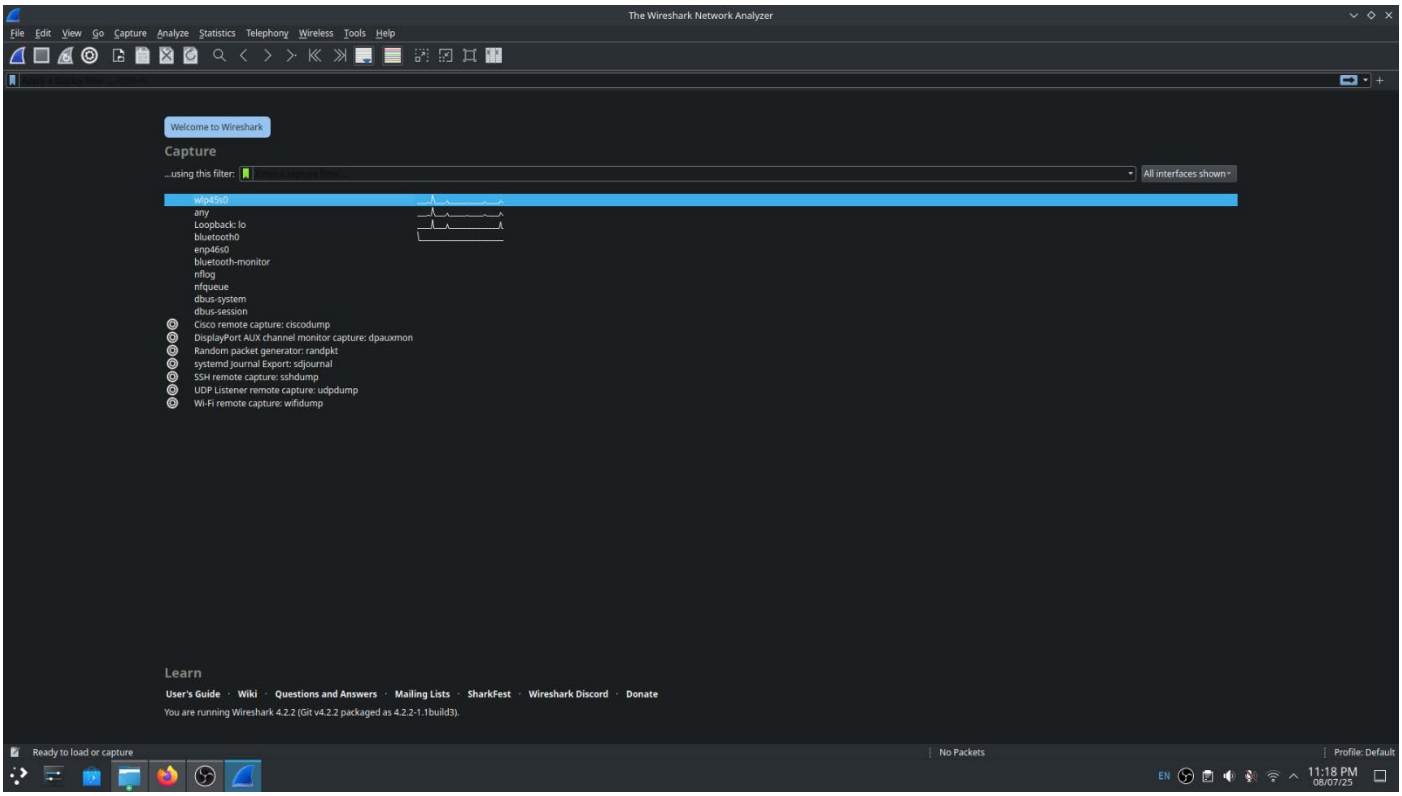
Step 7: Type “sudo apt update” to verify there no new updates.



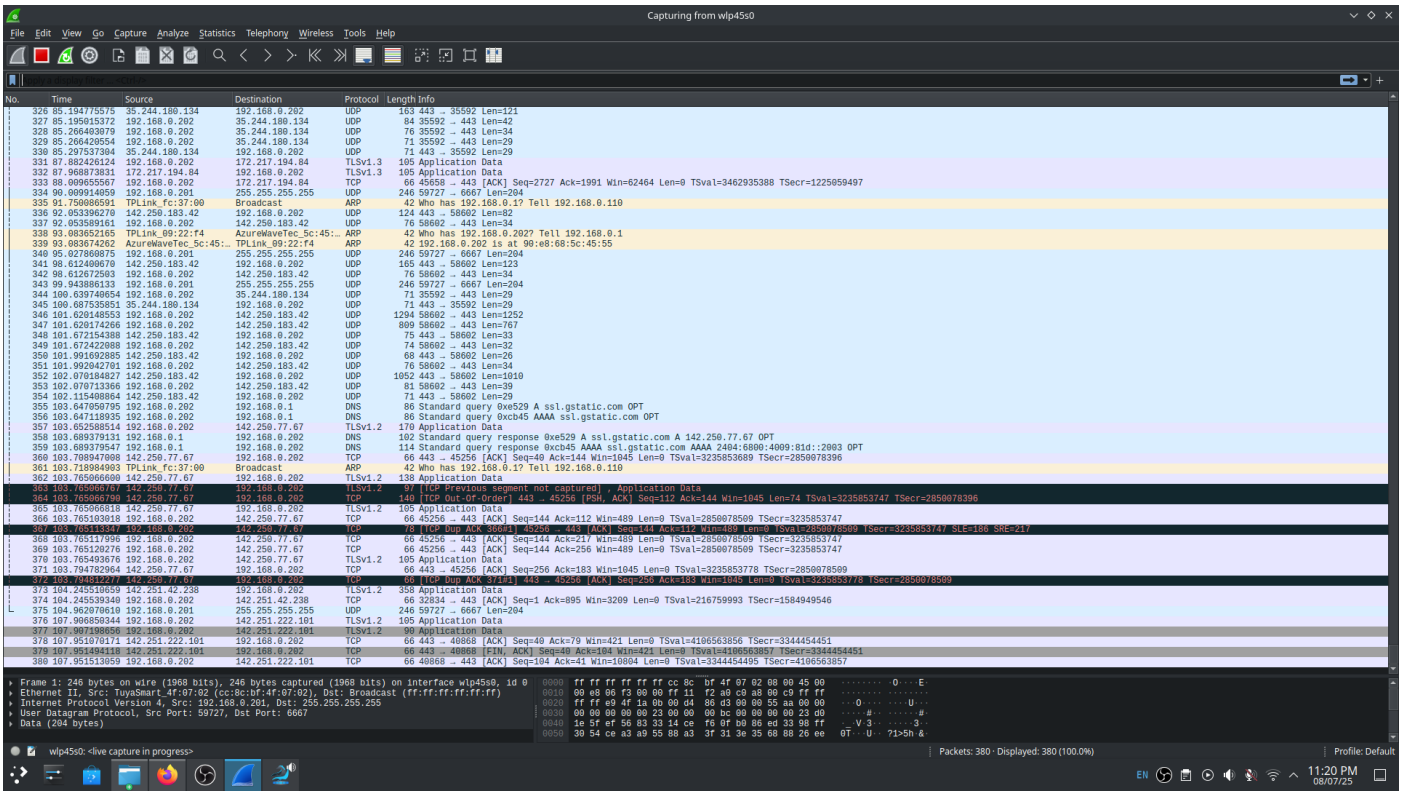
Practical – 9

Aim: Monitor network traffic and detect suspicious or malicious activity

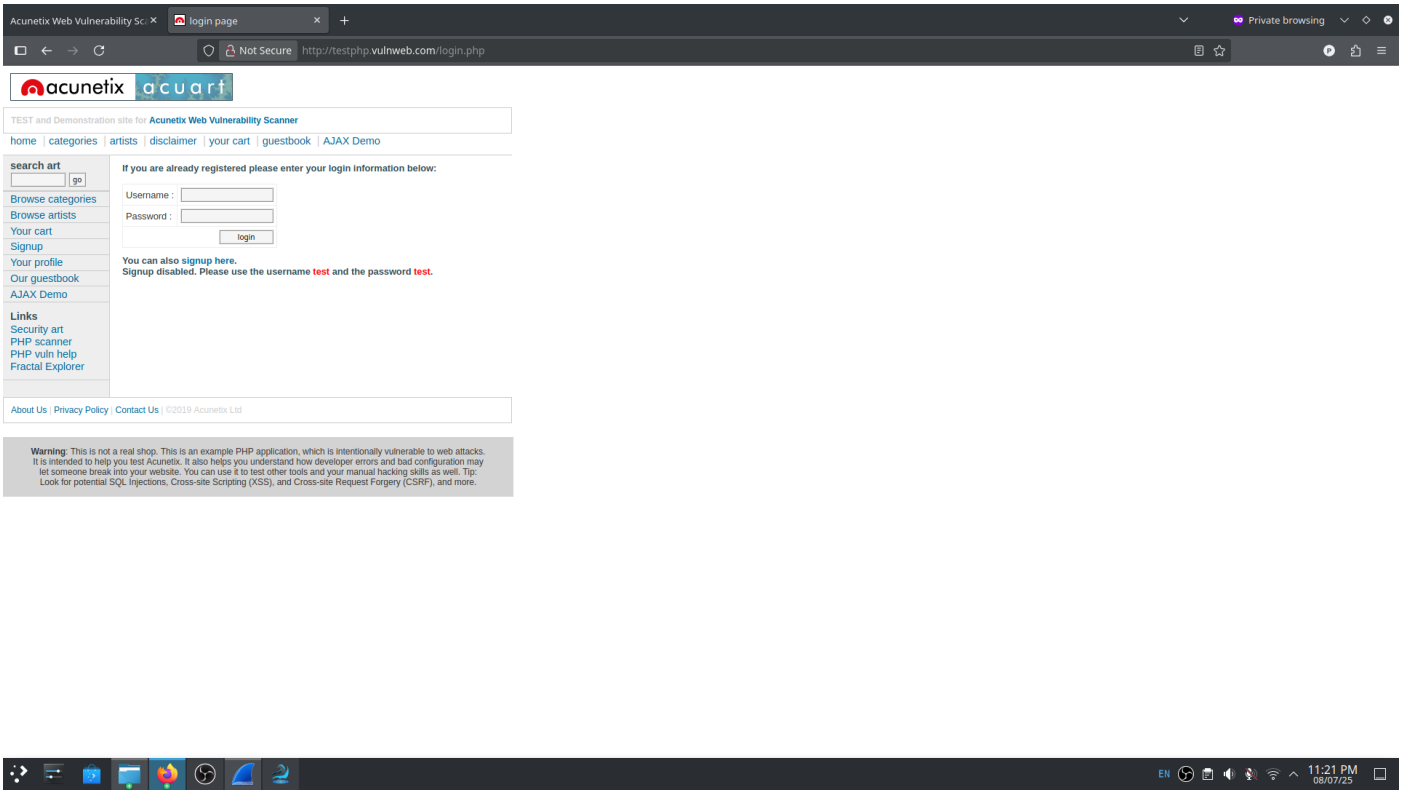
Step 1: Select and open “wlp45s0”.



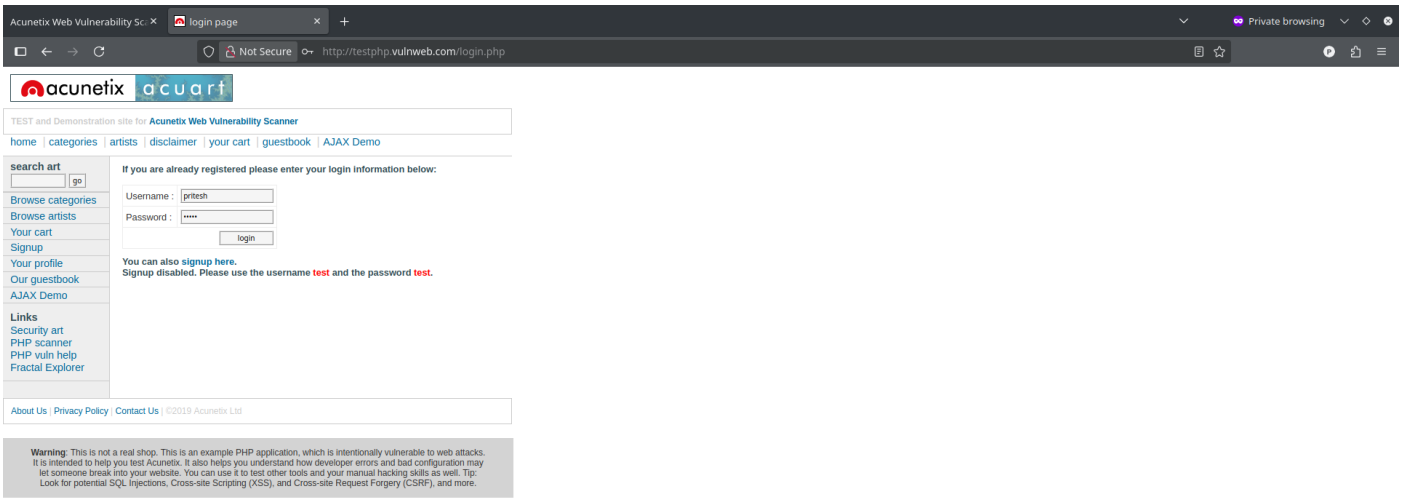
Step 2: Start network capturing.



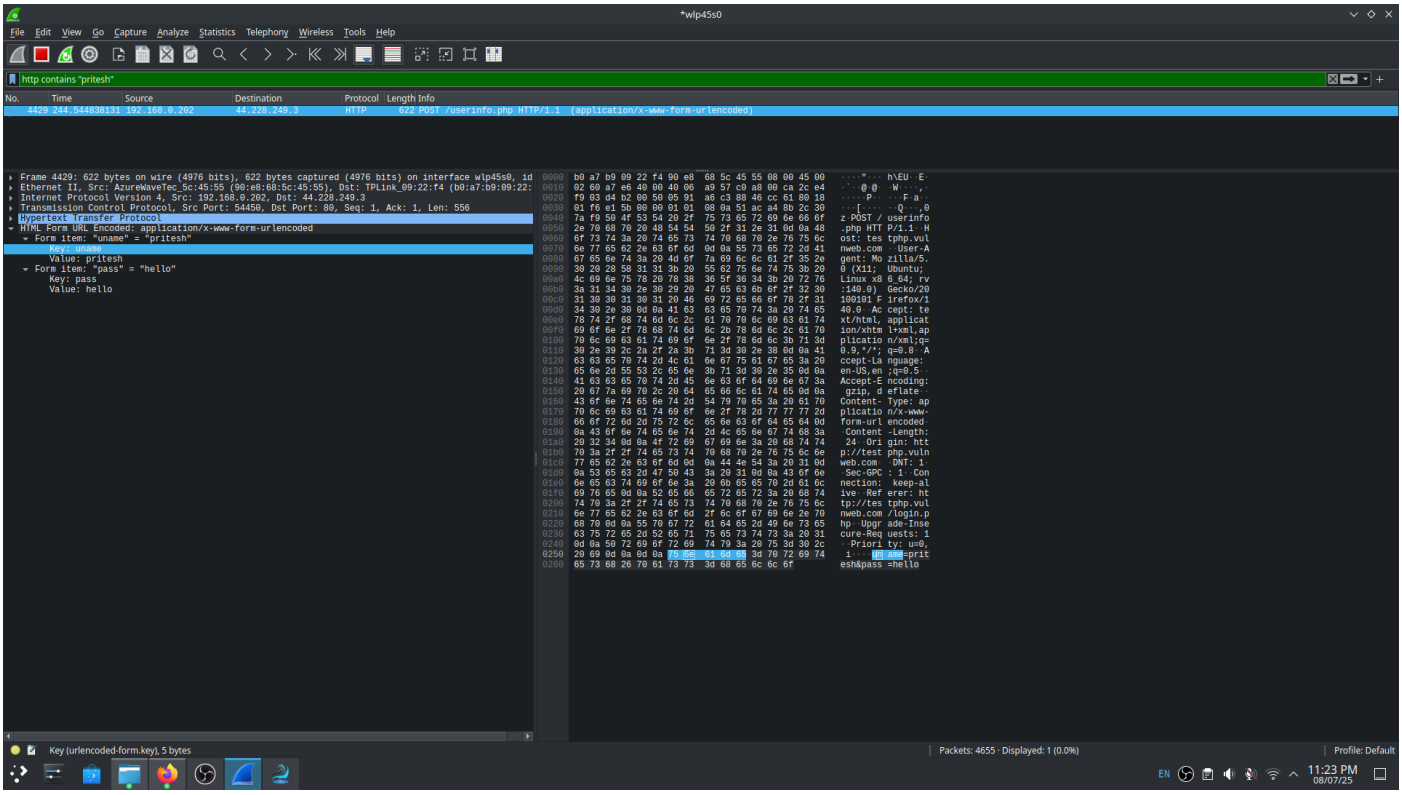
Step 3: Open "http://testphp.vulnweb.com/login.php" in browser.



Step 4: Enter Username and Password.



Step 5: In filters type “ http contains “uname” “ and hit Enter.



Practical – 10

Aim: Study AI related attacks on AI Devices.